

An Overview and Analysis of PDA Forensic Tools

Wayne Jansen, Rick Ayers
National Institute of Standards and Technology

Abstract: Mobile handheld devices are becoming evermore affordable and commonplace in society. When they are involved in a security incident or crime, forensic specialists require tools that allow proper extraction and speedy examination of any digital evidence present. This paper gives an overview of forensic software tools for Personal Digital Assistants (PDA). A set of generic scenarios was devised to simulate evidentiary situations and applied to a set of target devices to gauge how selected tools react under various situations. The paper summarizes those results, giving a snapshot of the capabilities and limitations of present day tools, and also provides background information on PDA hardware and software.

Keywords: Computer Forensics, Handheld Devices, Forensic Tools

Introduction

Handheld devices are commonplace in today's society, used by many individuals for both personal and professional purposes. While such devices have limitations, they are nonetheless extremely useful in such things as managing appointments and contact information, corresponding via electronic mail and instant messaging over wireless networks, and handling voice calls, as well as transporting and viewing documents. Increasingly, such devices also incorporate the ability to handle and even capture multimedia information (e.g., sound, images, and video). For cellular devices, such as smart phones, or GPS-enabled devices, additional evidence sources exist, for example, the last set of dialed numbers or the coordinates of waypoints to some destination. Even security minded individuals eventually succumb to entering sensitive information into such devices. Over the course of use, handheld devices can accumulate significant amounts of personal information that can provide a wealth of digital evidence when encountered during an investigation.

As with digital computers in general, both the functionality and information capacity of handheld devices are improving rapidly. Present day memory capacities can hold megabytes of information, easily extendable into the gigabyte range. Though an investigator can browse the contents of the device through its user interface to obtain evidence, the approach is highly impractical and problematic, and should be used only as a last resort. Instead, applying forensic tools is the preferred alternative. Forensic software tools facilitate the proper acquisition of data from a device and the examination, organization, and reporting of the evidence recovered. This article provides an overview of current forensic software designed for PDAs (Personal Digital Assistants), a popular type of handheld device.

PDAs in and of themselves are somewhat like miniature desktop computers. Significant differences from them exist, however, which affect the procedures used during seizure, acquisition, and examination. Some examples include the following items:

- Because PDAs are oriented toward mobility, they depend on battery power, emphasize wireless connectivity, and use specialized interfaces and media.

- PDAs typically use volatile memory versus non-volatile memory for user data, such that loss of battery power results in an immediate loss of data.
- PDAs normally use different operating systems from desktop computers, which accommodate mobility aspects such as power management, specialized file systems, automatic file compression, and execute-in-place programs.
- PDAs are always in an active state; when powered off or idle various degrees of hibernation occur to avoid a lengthy delay when powered on again or activity resumes.

Forensic software tools address only some of these issues. An investigator must have a good technical understanding and follow appropriate procedures to address the remaining issues when conducting an investigation [Jan04].

The remainder of this article discusses the characteristics of PDA hardware and software and surveys a number of PDA forensic tools, describing the functional range and facilities for acquiring and analyzing evidence contained therein. A set of generic scenarios was devised to mirror situations that might occur during a forensic examination of a PDA and its associated media. The scenarios were applied to a set of target devices to reveal how selected tools react under various situations. The scenarios and results are also presented in this paper, providing a snapshot of the capabilities and limitations of present day forensic tools.

PDA Hardware and Software

While different hardware components and platforms are used in PDA designs, they support an equivalent core set of Personal Information Management (PIM) capabilities, which include administration of address book, appointment book, mailbox, and memo data. Most devices also afford comparable abilities to communicate wirelessly, review electronic documents, and browse the Web. PIM data residing on a PDA can be synchronized with a desktop computer and automatically reconciled and replicated between the two devices using a synchronization protocol. Synchronization protocols can also be used to exchange other kinds of data such as text, image, audio, and archive file formats. Information not obtainable directly from the PDA can often be found on a desktop computer to which the device has been synchronized [Cas04].

Operating Systems

The various families of PDAs on the marketplace run different Operating Systems (OS), which among other things support different styles of interaction, types of file systems, and protocols for synchronizing information with a desktop computer. Windows Mobile, Palm OS and Linux platforms comprise the majority of the pure PDA devices currently available, and also a significant portion of the so-called smart phone category of cell phones.

Early Palm OS devices use 16- and 32-bit processors based on the Motorola DragonBall MC68328-family of microprocessors. More recent devices use StrongArm and XScale microprocessors. Versions of the Palm OS can be divided into three ranges: those before version 4.0, those from version 4.0 to 5.0, and those from 5.0 onward to version 6. Initially Palm OS supported simple multitasking whereby applications could run only one at a time and were

single-threaded [Kin01]. More recent versions support full multitasking and multithreaded applications.

The Palm OS and built-in applications are stored in ROM, while application and user data are stored in RAM [Kin01]. Add-on utilities are also available to back up PIM data (e.g., Address Book, Date Book, To Do List, Memo Pad) onto available ROM [Bob04, Pie99]. One of the latest Palm OS models, the Treo 650, breaks this pattern by using ROM instead of RAM for storage of PIM data. Palm OS divides the total available RAM store into two logical areas: dynamic RAM and storage RAM. Dynamic RAM is organized as a single heap and used as working space for temporary allocations, analogous to the RAM installed in a typical desktop system. The remainder of the available RAM is designated as storage RAM and configured as one or more storage heaps used to hold non-volatile user data. Storage RAM is analogous to disk storage on a typical desktop system. Storage heaps may also be ROM-based. Palm OS arranges storage memory into chunks called records, which are grouped into databases – the equivalent of files [PS04]. The architecture for Palm OS devices up to version 5.0 is organized into the following layers: Application, Operating System, Software API and Hardware Drivers, and Hardware. Figure 1 illustrates the relationship between layers.

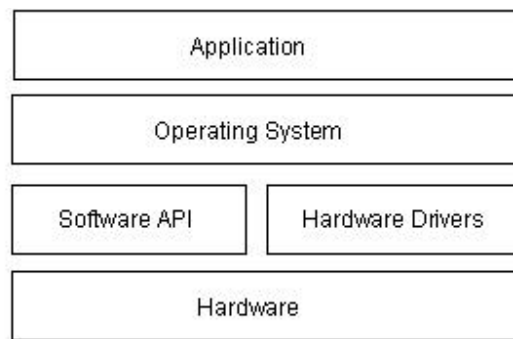


Figure 1: Palm OS Architecture

Windows Mobile, initially branded as Pocket PC, grew out of the success of the Palm PDA and the growing demand for similar devices that had more processing power and networking capabilities. Microsoft entered the handheld device market with the Windows CE (WinCE) operating system, which was later augmented with additional functionality to produce Pocket PC (PPC). Windows CE supports a multitasking, multithreaded environment, which is inherited by Pocket PC. Windows CE and PPC have evolved in tandem from versions WinCE 2.0/PPC 2000 to WinCE 3.0/PPC 2002 to WinCE 4.2/Windows Mobile 2003 (PPC 2003 was rebranded as Windows Mobile 2003). For simplicity, in this paper Pocket PC refers to the entire line of PDA operating systems.

Pocket PC runs on a number of processors, but primarily appears on devices having Xscale, ARM, or SHx processors. The operating system and support applications reside in ROM. PIM and other user data normally reside in RAM, along with but segregated from program memory, used to store the stacks and heaps of executing applications. Unused ROM can be allocated and made available as an additional file store for backing up data files from RAM. The Windows CE Object Store organizes persistent objects in one of three categories: the file system – for file and directory objects, the registry – for user preferences and configuration settings, and the database – for PIM application data [MSC04]. The architecture for Windows CE devices consists of four

layers: Application, Operating System, Original Equipment Manufacturer (OEM), and Hardware. A simplified diagram of the architecture of Windows CE is shown in Figure 2 below. Services are organized into modules, which can be included or excluded when building an image for a specific target system [Ges03]. OEM Layer allows an OEM to adapt Windows CE to a specific platform by writing functions related to system startup, interrupt handling, power management, profiling, timer and clock for any custom hardware present.

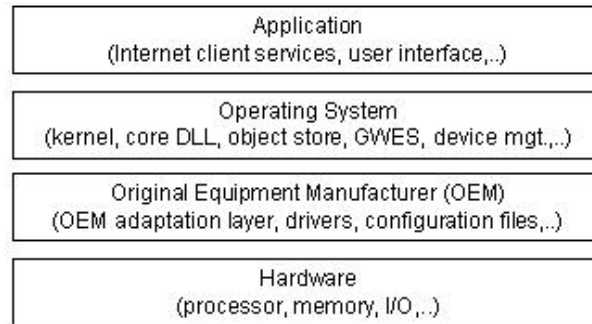


Figure 2: Windows CE Architecture

Linux, a popular open source operating system for servers, which is used on desktop computers, has also appeared on a number of PDA devices [Fae03]. Linux is a true multitasking, 32-bit operating system that supports multithreading. Besides commercial distributions that come preinstalled by PDA manufacturers (e.g., Sharp for the Zaurus PDA), Linux distributions are also available for a range of Pocket PC and Palm OS devices [Fae01, Hal01, Hon04, Zwi02]. The success of Linux-based PDAs rests on its open source model and ability to engage the software development community to produce useful applications. The Linux operating system is responsible for memory management, process and thread creation, interprocess communication mechanisms, interrupt handling, execute-in-place (XIP) ROM file systems, RAM file systems, flash management, and TCP/IP networking. While most Linux distributions include the same utilities, libraries, drivers, and windowing frameworks, differences occur in the associated patches and modules, the included utilities, and the facilities for installation, configuration and upgrade.

Often PIM and other user data reside in RAM, while the operating system and support applications reside in ROM. However, Linux supports file systems such as JFFS2 (The Journaling Flash File System, version 2), which was designed specifically to manage usage of flash memory resident files to minimize wear and postpone destruction of the media [Woo01]. For example, JFFS2 prevents the rewrite of an entire sector to erase a single byte and ensures that different areas of memory are used in rotation. Thus, some Linux distributions use such file systems to maintain user data in flash ROM along with the operating system and support application code, and use RAM exclusively for process execution. Figure 3 gives a conceptual architecture for the Linux operating system.

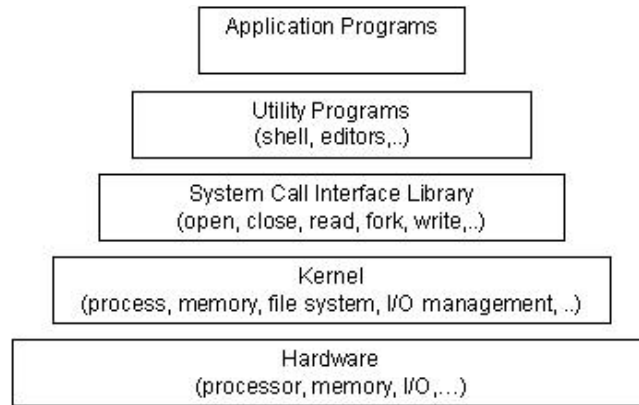


Figure 3: Linux Architecture

Hardware Characteristics

Just as with operating system and application software, most types of PDAs have comparable hardware features and capabilities. They house a microprocessor, read only memory (ROM), random access memory (RAM), a variety of hardware keys and interfaces, and a touch-sensitive, liquid crystal display. The operating system (OS) of the device is held in ROM. Several varieties of ROM are used, including Flash ROM, which can be erased and reprogrammed electronically with OS updates or an entirely different OS. RAM, which normally contains user data, is kept active by batteries whose failure or exhaustion causes that information to be lost.

The latest PDAs include considerable memory capacity and come equipped with system-level microprocessors that reduce the number of supporting chips required. Built-in Compact Flash (CF) and combination Secure Digital (SD)/MultiMedia Card (MMC) slots support memory cards and peripherals, such as a digital camera or wireless communications card. Wireless communications such as infrared (i.e., IrDA), Bluetooth, and WiFi may also be built in. Figure 4 illustrates a system-level processor chip and the generic core components of most PDAs.

Different devices have different technical and physical characteristics (e.g., size, weight, processor speed, memory capacity). Devices may also use different types of expansion capabilities (e.g., I/O and memory card slots, device expansion sleeves, and external hardware interfaces) to provide additional functionality. Furthermore, PDA capabilities are sometimes combined with those of other devices such as cell phones, Global Positioning Systems (GPS), and cameras to form new types of hybrid devices. Characteristics for a wide range of PDAs can be found on manufacturer and vendor Web sites, as well as product review sites.

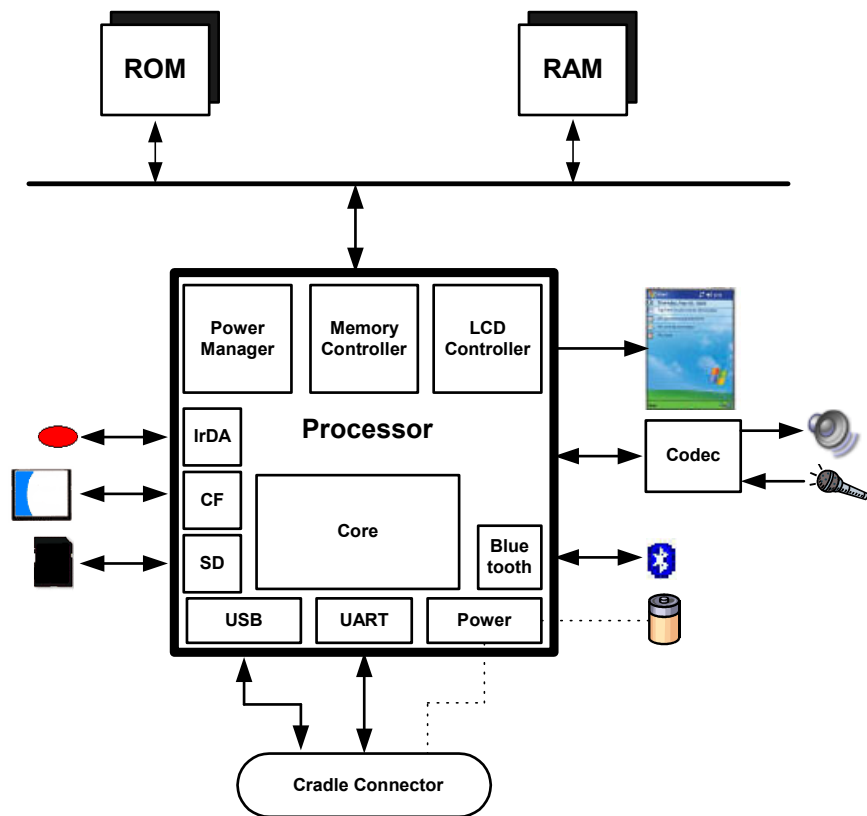


Figure 4: Generic Hardware Diagram

A wide array of memory cards exists on the market today for PDAs. Their storage capacity is substantial, ranging from 8MB to beyond 2GB, though their size is diminutive, no larger than a matchbook, and easy to overlook. Memory cards allow individuals to store additional files beyond the device's built-in capacity and provide another avenue for sharing data between compatible devices. Unlike the RAM of a PDA, removable media is non-volatile storage, normally flash memory, and does not require battery power to retain data. Fortunately, if a PDA forensic tool cannot handle such media, the memory card can be treated similarly to a removable disk drive, and imaged and analyzed with the aid of an external media adaptor using conventional forensic tools. Memory card adapters exist that support an Integrated Development Environment (IDE) interface, allowing removable media to be treated as a hard disk and used with a write blocker to ensure that the contents of the removable media remain unaltered during acquisition [Wie02]. Several common types of memory cards in use with PDAs are listed in Table 1.

Table 1: Memory Card Formats

Name	Characteristics
Compact Flash Card (CF)	Matchbook size (length-36.4 mm, width-42.8 mm, thickness-3.3 mm for Type I cards and 5mm for Type II cards) 50 pin connector, 16-bit data bus
Multi-Media Card (MMC)	Postage stamp size (length-32 mm, width-24 mm, and thickness-1.4 mm) 7-pin connector, 1-bit data bus
Reduced Size MMC	Fingernail size (length-18mm, width-24mm, and thickness-1.4mm) 7-pin connector, 1-bit data bus Requires a mechanical adapter to be used in a full size MMC slot
Secure Digital (SD) Card	Postage stamp size (length-32 mm, width-24 mm, and thickness-2.1mm) 9-pin connector, 4-bit data bus Features a mechanical erasure-prevention switch
MiniSD Card	Fingernail size (length-21.5 mm, width-20 mm, and thickness-1.4 mm) 9-pin connector, 4-bit data bus Features a mechanical erasure-prevention switch Requires a mechanical adapter to be used in a full size SD slot
Memory Stick	Chewing gum stick size (length-50mm, width-21.45mm, thickness-2.8mm) 10-pin connector, 1-bit data bus Features a mechanical erasure-prevention switch
Memory Stick Duo	Partial chewing gum stick size (length-31mm, width-20mm, thickness-1.6mm) 10-pin connector, 4-bit data bus Features a mechanical erasure-prevention switch

Forensic Tools

Unlike the situation with desktop computers and workstations, the number and variety of toolkits for PDAs and other handheld devices are limited. Not only do fewer specialized tools and toolkits exist, but also the range of devices over which they operate is typically narrowed to only the most popular families of PDA devices – those based on the Pocket PC and Palm OS. Linux-based devices can be imaged with the dd utility, somewhat analogously to a Linux desktop, and analyzed with the use of a compatible tool (e.g., EnCase). Since Palm OS devices have been around the longest, more forensic tools are available for them than for other device families. Table 2 lists open-source and commercially available tools for PDAs known to the authors and the facilities they provide: acquisition, examination, or reporting. The abbreviation NA means that the tool listed at the left of the row is not applicable to the device listed at top of the column. With one exception (i.e., versions of Palm OS prior to 4.0 [Kin01]), these tools require that the examiner have unobstructed access to acquire contents (i.e., no authentication technique need be satisfied to gain access).

Table 2: PDA Forensic Tools

	Palm OS	Pocket PC	Linux
pdd	Acquisition	NA	NA
Pilot-Link	Acquisition	NA	NA
POSE	Examination, Reporting	NA	NA
PDA Seizure	Acquisition, Examination, Reporting	Acquisition, Examination, Reporting	NA
EnCase	Acquisition, Examination, Reporting	NA	Examination, Reporting
dd	NA	NA	Acquisition

Forensic tools acquire data from a device in one of two ways: physical acquisition or logical acquisition. Physical acquisition implies a bit-by-bit copy of an entire physical store (e.g., a disk drive or RAM chip), while logical acquisition implies a bit-by-bit copy of logical storage objects (e.g., directories and files) that reside on a logical store (e.g., a file system partition). The difference lies in the distinction between memory as seen by a process through the operating system facilities (i.e., a logical view), versus memory as seen in raw form by the processor and other related hardware components (i.e., a physical view).

Physical acquisition has advantages over logical acquisition, since it allows deleted files and any data remnants present (e.g., unallocated RAM or unused file system space) to be examined, which otherwise would go unaccounted. Physical device images are generally more easily imported into another tool for examination and reporting. However, a logical structure has the advantage that it is a more natural organization to understand and use during examination. Thus, if possible, doing both types of acquisition on PDAs is preferable.

Tools not designed specifically for forensic purposes are questionable and should be thoroughly evaluated before use. Non-forensic software tools generally focus on logical acquisition, using an available protocol for device synchronization and management to communicate with the device, as opposed to a debugging or testing protocol that could be used to acquire a memory image. Such tools support two-way flow of information, unconcerned with blocking changes, and avoid taking hashes of acquired content for integrity purposes. Documentation also may be limited and source code unavailable for examination, respectively increasing the likelihood of error and decreasing confidence in the results. As with any tool, forensic issues might be associated with the usage of a non-forensic tool. For example, a tool may inadvertently alter a time stamp on the device, such as the date and time of last synchronization, incorrectly decode and display information, or alter the state of the device. For example, some tools for Palm OS devices require the device to be placed in console debug mode, causing a soft reset after collection. The soft reset triggers heap compaction and deleted records to be overwritten [Cas04]. On the one hand, non-forensic tools might be the only means to retrieve information that could be relevant as evidence. On the other, they might overwrite, append, or otherwise cause information to be lost (e.g., through a reset) during use.

Palm dd (pdd)

Palm dd (pdd) is a Windows-based command line tool that performs a physical acquisition of information from Palm OS devices [Gra02].¹ pdd is designed to work with most PDAs running the Palm OS in console mode. During the acquisition stage, a bit-for-bit image of the device's memory can be obtained. The data retrieved by pdd includes all user applications and databases. pdd is strictly a command line driven application without features such as graphics libraries, report generation, search facilities, and book-marking capabilities. Once the information has been acquired, two files are generated: one that contains device-specific information (e.g., OS version, processor type, sizes of RAM and ROM), and another that contains a bit-by-bit image of the device. Examiners face the challenge of carefully examining the output, which is in binary form, some of which happens to be ASCII characters. Files created from pdd can be imported into a forensic tool, such as EnCase, to aid analysis; otherwise, the default tool is a hex editor. pdd does not provide hash values for the information acquired. However, a separate procedure can be used to obtain needed hash values. As of January 2003, pdd is no longer supported, however, version 1.11 source code is available and should remain available for use, as defined in the included license. Paraben has integrated elements of the pdd engine into PDA Seizure [Cas04].

Pilot-Link

Pilot-link is an open source software suite originally developed for the Linux community to allow information to be transferred between Linux hosts and Palm OS devices.² It runs on other desktop operating systems besides Linux, including Windows and Mac OS. About thirty command line programs comprise the software suite. Unlike pdd, which uses the Palm debugger protocol for acquisition, pilot-link uses the Hotsync protocol. The two programs of interest to forensic specialists are pi-getram and pi-getrom, which respectively retrieve the contents of RAM and ROM from a device, similar to the physical acquisition done by pdd. Another useful program is pilot-xfer, which allows the installation of programs and the backup and restoration of databases. pilot-xfer provides a means to acquire the contents of a device logically. The contents retrieved with these utilities can be manually examined with either POSE, a compatible forensic tool such as EnCase, or a hex editor. Pilot-link does not provide hash values of the information acquired. A separate step must be carried out with an appropriate utility to obtain them.

POSE

POSE (Palm OS Emulator) is a software program that runs on a desktop computer under a variety of operating systems, and behaves exactly as a Palm OS hardware device, once an appropriate ROM is loaded into it.³ The free emulator program imitates the hardware of a DragonBall processor. Built-in PIM applications (e.g., Datebook, Address Book, To Do, etc.) run properly and the hardware buttons and display react accurately. ROM images can be obtained from the PalmSource Web site or by copying the contents of ROM from an actual device, using pdd, Pilot-Link, or a companion tool provided with the emulator. POSE is limited to Palm OS versions 4.x and below.

¹ More information can be found at <http://www.atstake.com/research/tools/forensic/>

² More information can be found at <http://www.pilot-link.org>

³ More information can be found at <http://www.palmos.com/dev/tools/emulator/>

Loading actual RAM-based databases into the emulator, extracted using pilot-link or another tool, allows an examiner to view and operate the emulated device in a similar fashion as having the original. Though originally developed to run, test, and debug Palm OS applications without having to download them to an actual device, POSE also serves as a useful tool for doing presentations or capturing screen shots of evidence found on the emulated device from within the databases loaded from a seized device. POSE can be configured to map the Palm OS serial port to one of the available serial ports on the desktop computer or to redirect any TCP/IP calls to the TCP/IP stack on the desktop. With some experimentation, the HotSync protocol can even be run between the desktop computer and device it is emulating, over a looped back serial connection or a redirected TCP/IP connection.

PDA Seizure

Paraben's PDA Seizure is a commercially available forensic software toolkit that allows forensic examiners to acquire and examine information on PDAs for both the Pocket PC (PPC) and Palm OS platforms.⁴ Paraben's product currently supports Palm OS up to version 5, Pocket PC 2000-2003 (up to Windows CE 4.2), ActiveSync 3.7, and HotSync. PDA Seizure's features include the ability to acquire a forensic image of Palm OS, Pocket PC, and BlackBerry devices, to perform examiner-defined searches on data contained within acquired files, generate hash values of individual files and to generate a report of the findings. PDA Seizure also provides book-marking capabilities to organize information, along with a graphics library that automatically assembles found images under a single facility, based on the graphics file extension of the acquired files.

During the acquisition stage of a PPC device, the connectivity of the device via ActiveSync is required. A guest account must be used to ensure data is not synchronized after a successful connection has been established. Before acquisition begins, PDA Seizure places a small program on the device in the first available block of memory to access unallocated regions of memory. To access the remaining information, PDA Seizure utilizes the Remote API (RAPI) protocol, which provides a set of functions for desktop applications to communicate with a device and logically access information. For Palm OS devices, the PDA must first be put into a debug mode, commonly referred to as console mode, and all active HotSync applications must be closed. Once the memory image of a Palm OS device is acquired, the user is prompted to select the HotSync button on the device to acquire the logical data separately. The logical data is also represented in the RAM image file that was acquired through physical acquisition. Palm's HotSync protocol is used to gain communication with the device to perform a logical acquisition.

EnCase

EnCase is a commercially available forensic software toolkit that provides acquisition of suspect media, search and analytical tools, hash generation of individual files, data capture and documentation features.⁵ Although more widely used for examining PCs, EnCase also supports Palm OS devices. Currently, support for Pocket PC is not available, but the ability to import a data dump of Linux-based PDAs exists. EnCase allows for the creation of a complete physical bit-stream image of a Palm OS device. Throughout the process, the integrity of the bit-stream image is continually verified by CRC (Cyclical Redundancy Check) values, which are calculated

⁴ More information can be found at <http://www.paraben-forensics.com/pda.html>

⁵ More information can be found at <http://www.guidancesoftware.com/products/EnCaseForensic/productinfo.shtm>

concurrent to acquisition. The resulting bit-stream image, called an EnCase evidence file, is mounted as a read-only file or “virtual drive” from which EnCase proceeds to reconstruct the file structure using the logical data in the bit-stream image. This allows the examiner to search and examine the contents of the device using either a logical or physical perspective.

EnCase allows for files, folders, or sections of a file to be highlighted and saved for later reference. These marks are called bookmarks. All bookmarks are saved in case files, with each case having its own bookmark file. Bookmarks can be viewed anytime and can be made from anywhere data or folders exist. Reporting features allows examiners to view information from a number of perspectives: all acquired files, single files, results of a string search, a report, or the entire case file created.

Duplicate Disk (dd)

The duplicate disk (dd) utility is similar to pdd insofar as it allows examiners to create a bit-by-bit image of the device. As one of the original Unix utilities, dd has been around in one form or another for decades. Unlike the other tools described above, dd executes directly on the PDA. An image of the device can be obtained by connecting to the PDA, issuing the dd command, and dumping the contents elsewhere, for example, to auxiliary media such as a memory card or across a network session to a forensic workstation. Caution should be exercised, since dd may destroy parts of the file system (e.g., overwriting data) if used incorrectly. As with pdd, dd produces binary data output, some of which contains ASCII character information. Images created from dd may be imported for examination into a forensic tool, such as EnCase, if the file system is supported. A dd created image may also be mounted in loopback mode on a file system-compatible Linux machine for analysis. The standard version of dd does not provide hash values for the information acquired. However, a separate procedure can be used to obtain needed hash values. Modified versions of dd exist that incorporate hash value computation, but would require cross compilation and installation to use.

Forensic Tool Analysis

A simple methodology was followed to gauge the capabilities of the forensic tools under study. First, a set of target devices was assembled. Second, a set of prescribed activities was performed for each device. Third, after each activity, the contents of the device were acquired using one of the available tools and examined to see if the results of an activity could be recovered as expected. Finally, an assignment was made as to whether the tool met, surpassed, fell below, or completely missed stated expectations

Each scenario defines a generic set of activities. The scenarios begin with content acquisition and move progressively toward more interesting situations involving common applications, file formats, and device settings. The scenarios were not intended to be exhaustive or to serve as a formal product evaluation. However, they attempt to cover a range of situations commonly encountered when examining a device (e.g., data obfuscation, data hiding, data purging) and are useful in determining the features and functionality afforded an examiner.

Table 3 gives an overview of the scenarios used. For each scenario listed, a description of its purpose, method of execution, and expected results are summarized. Note that the expectations are comparable to those an examiner would have when dealing with the contents of a hard disk

drive as opposed to a PDA. Though the characteristics of the two devices are quite different, the recovery and analysis of information from a hard drive is a well-understood baseline for comparison and pedagogical purposes. Note too that none of the scenarios attempt to confirm whether the integrity of the data on a device is preserved when applying a tool – that topic is outside the scope of this paper.

Table 3: Scenarios

Scenario	Description
Device Content Acquisition	<p>Determine if the tool can successfully acquire the contents of the device.</p> <ul style="list-style-type: none"> • Initiate the tool on a forensic workstation, attempt to connect with the device and acquire its contents, verify that information has been obtained. • Expect that information residing on the device can be successfully acquired.
PIM Applications	<p>Determine whether the tool can find information, including deleted information, associated with Personal Information Management (PIM) applications such as calendar, contacts, e-mail synched with a PC, and task lists.</p> <ul style="list-style-type: none"> • Create various types of PIM files on the device, selectively delete some entries, acquire the contents of the device, locate and display the information. • Expect that all PIM-related information on the device can be found and reported, if not previously deleted. Expect that remnants of deleted information can be recovered and reported.
Web/E-mail Applications	<p>Determine whether the tool can find a visited web site and exchanged e-mail message information obtained by a wireless network enabled device through an 802.11b access point.</p> <ul style="list-style-type: none"> • Use the device to visit specific web sites and exchange e-mail, acquire the contents of the device, selectively delete some e-mail, locate and display the URLs of visited sites, headers of e-mail messages, and any associated data acquired (e.g., images, text, etc.). • Expect that information about most recent web, web-mail, and e-mail activity can be found and reported. Expect that remnants of deleted e-mail information can be recovered and reported.
Graphics File Formats	<p>Determine whether the tool can find and display a compilation of the graphics formatted files acquired from the device.</p> <ul style="list-style-type: none"> • Load the device with various types of graphics files, acquire the contents of the device, locate and display the images. • Expect that all files with common graphics files formats (i.e., .bmp, .jpg, .gif, .tif, and .png) can be found, reported, and collectively displayed.

Scenario	Description
Compressed File Archive Formats	<p>Determine whether the tool can find text, images, and other information located within compressed-archive formatted files (i.e., .zip) residing on the device.</p> <ul style="list-style-type: none"> • Load the device with various types of file archives, acquire the contents of the device, find and display selected filenames and file contents. • Expect that text, images, and other information contained in common compressed archive formatted files can be found and reported.
Other Compressed Archive Formats	<p>Determine whether the tool can find text, images, and other information within other, less common, archive formats (i.e., .tar, .tar.gz, .tgz, .rar, and self-extracting .exe).</p> <ul style="list-style-type: none"> • Load the device with various types of file archives, acquire the contents of the device, find and display selected filenames and file contents. • Expect that text, images, and other information contained in the compressed archive formatted files can be found and reported.
Deleted Files	<p>Determine if the tool can recover files deleted from the device. Two variants exist: recovery attempted before and after synchronizing the device with a PC.</p> <ul style="list-style-type: none"> • Create one or more files on the device; delete a file, acquire the contents of the device, and attempt to locate the deleted file. • Expect that all deleted files can be recovered, reported, and, if an image, displayed.
Misnamed Files	<p>Determine whether the tool can recognize file types by header information instead of file extension, and find common text and graphics formatted files that have been misnamed with a misleading extension.</p> <ul style="list-style-type: none"> • Load the device with various types of common text (e.g., .txt) and graphics files (e.g., .bmp, .jpg, .gif, and .png) purposely misnamed, acquire the contents of the device, locate and display selected text and images. • Expect that all misnamed text and graphics files residing on the device can be recognized, reported, and, if an image, displayed.
Peripheral Memory Cards	<p>Determine whether the tool can acquire individual files stored on a removable memory card inserted into the device and whether deleted files would be identifiable and recoverable.</p> <ul style="list-style-type: none"> • Insert a memory card containing a populated file system into an appropriate slot on the device, delete some files, acquire the contents of the device, find and display selected files and file contents, including deleted files. • Expect that the files on the memory card, including deleted files, can be properly acquired, found, and reported in the same way as expected with on-device memory.
Cleared Devices	<p>Determine if the tool can acquire any user information from the</p>

Scenario	Description
	device or peripheral memory, after a hard reset has been performed. <ul style="list-style-type: none"> • Perform a hard reset on the device, acquire its contents, and find and display previously available filenames and file contents. • Expect that no user files, except those contained on a peripheral memory card, if present, can be recovered.
Password Protected Devices	Determine whether the tool can obtain the user's password to acquire the contents of the device. <ul style="list-style-type: none"> • Enable the password on the device, apply any utilities to crack the password, acquire the contents of the device. • Expect that the user's password cannot be obtained, except for those devices with older, more vulnerable operating systems.

To apply the various forensic tools against the scenarios, several PDAs from different device families served as the target device under examination. Table 4 summarizes the various operating system and target device combinations available. The entries indicate the operating system version and processor type of the target devices used with a tool. All the devices listed, except one, come with the respective operating system preinstalled by their manufacturer. The Linux column, however, includes a modified device (i.e., an iPaq 3970) on which a distribution of Linux for handheld devices was installed for this exercise, replacing the manufacturer's preinstalled operating system.

Table 4: Target Devices

Pocket PC	Palm OS	Linux
Jornada 548 (PPC 00)	Palm III (3.0)	Zaurus SL-5000 (Lineo 2.4.6)
iPaq 3875 (PPC 00)	Palm Vx (3.3)	iPaq 3970 (Familiar version 2.4.19)
iPaq 3970 (PPC 02)	Visor Platinum (3.5)	
iPaq 5455 (PPC 03)	Tungsten C (5.2.1)	

To determine the extent to which a given tool meets the expectations listed, the scenarios were applied to the target devices appropriate for each tool. The result of each scenario exercised was then compared against the predefined expectations defined in the scenario descriptions and an assessment assigned. The entry "Meet" indicates that the software met the expectations of the scenario for the device in question. Similarly, "Below" indicates that the software fell short of fully meeting expectations; "Above" indicates that the software surpassed expectations; and "Miss" indicates that the software unsuccessfully met any expectations. "NA" indicates scenarios that were not applicable for the device. The remainder of this section summarizes the results for PDA Seizure version 2.5.0.0, EnCase version 4.15, and dd. More detailed information can be found in [Aye04].

PDA Seizure demonstrated results shown in Table 5 that met or exceeded expectations in the majority of cases. On Palm OS devices, for those scenarios where outcomes were below expectations, all except one, involving memory cards on the Tungsten C, were related to deleted file recovery. Recovery of deleted files on PDAs depends on a number of factors including available memory space on the device, storage recovery algorithms, and hardware

characteristics. Similarly, deleted file recovery was the situation that plagued Pocket PC devices, affecting all but two scenarios, Graphics File Formats and Misnamed Files. For the former scenario, .tif and .png graphic formatted files were not handled automatically as with other graphic formats and displayed in the graphics library. For the latter, misnamed file extensions could not be identified by file header/footer signatures.

Table 5: PDA Seizure - Scenario Outcome

Scenario	Device					
	Joranda 548	iPaq 38/39xx	iPaq 5400	Palm III/Vx	Visor	Tungsten C
Device Content Acquisition	Meet	Meet	Meet	Meet	Meet	Meet
PIM Applications	Meet	Meet	Miss ^a	Meet	Meet	Meet
Web/E-mail Applications	NA	NA	Meet	NA	NA	Meet
Graphics File Formats	Below ^b	Below ^b	Below ^b	Meet	Meet	Meet
Compressed File Archive Formats	Meet	Meet	Meet	NA	NA	NA
Other Compressed Archive Formats	Meet	Meet	Meet	NA	NA	NA
Deleted Files	Below ^c	Miss ^d	Miss ^d	Below ^e	Below ^e	Below ^e
Misnamed Files	Miss ^f	Miss ^f	Miss ^f	NA	NA	NA
Peripheral Memory Cards	Below ^g	Below ^g	Below ^g	NA	NA	Miss ^h
Cleared Devices	Meet	Meet	Meet	Meet	Above ⁱ	Above ⁱ
Password Protected Devices	NA	NA	NA	Meet	Meet	NA

^a No PIM data was recovered

^b Not all graphics formats were supported for display

^c Some file-related information was recovered, but no file content

^d No information about deleted files was recovered

^e Some file-related information and file content was recovered, but not all

^f Files with altered extensions were not recognized

^g Deleted files were not recovered, but were recoverable through other means

^h Memory card was not discovered and its contents acquired

ⁱ Surprisingly, some information was recovered after a hard reset of the device

EnCase produced results for Palm OS devices comparable to PDA Seizure, as shown in Table 6. The one notable exception was that EnCase failed to acquire information from the Palm III device. However, the device was able to be imaged with pdd and the output imported into

EnCase to exercise the remaining scenarios. Recovery of deleted files again were the main cause of unmet expectations.

Table 6: Encase - Scenario Outcome

Scenario	Device				
	Palm III	Palm Vx	Visor	Tungsten C	Zaurus
Device Content Acquisition	Miss ^a	Meet	Meet	Meet	NA ^b
PIM Applications	Meet	Meet	Meet	Meet	Meet
Web/E-mail Applications	NA	NA	NA	Meet	NA
Graphics File Formats	Meet	Meet	Meet	Meet	Meet
Compressed File Archive Formats	NA	NA	NA	NA	Meet
Other Compressed Archive Formats	NA	NA	NA	NA	Meet
Deleted Files	Below ^c	Below ^c	Below ^c	Below ^c	Below ^c
Misnamed Files	NA	NA	NA	NA	Meet
Peripheral Memory Cards	NA	NA	NA	Miss ^d	NA
Cleared Devices	Meet	Meet	Above ^e	Above ^e	Meet
Password Protected Devices	NA	NA	NA	NA	NA

^a Acquisition was performed using @stakes pdd utility, after EnCase was unsuccessful

^b Acquisition was performed using the duplicate disk (dd) utility

^c Some file related information was recovered, but not all

^d Memory card was not discovered and its contents acquired

^e Surprisingly, some information was recovered after a hard reset of the device

The results for dd when applied to the Zaurus and an iPaq with the Familiar distribution of Linux installed were mixed. Because dd is no more than an acquisition tool, other utilities and techniques were used to examine the resulting image. The lack of a proper toolkit not only required more effort to perform the examinations, but also to track and report results. The output of dd for the Zaurus was able to be imported into EnCase and examined more readily. Deleted files were also a problem here, however, since not all file-related information and file content could be recovered. In the case of the iPaq the results had to be obtained manually, since no automated means were available to interpret and decode the JFFS2 file system image for examination. Manual examination proved to be difficult as reflected in the poor results attained for most scenarios, as shown in Table 7.

Table 7: Duplicate Disk (dd) - Scenario Outcome

Scenario	Device	
	Zaurus SL-5000	IPaq 3970
Device Content Acquisition	Meet	Meet
PIM Applications	Meet	Below ^a
Web/E-mail Applications	NA	NA
Graphics File Formats	Meet	Below ^b
Compressed File Archive Formats	Meet	Meet
Other Compressed Archive Formats	Meet	Below ^c
Deleted Files	Below ^d	Below ^d
Misnamed Files	Meet	Meet
Peripheral Memory Cards	NA	NA
Cleared Devices	Meet	Above ^e
Password Protected Devices	NA	NA

^a Only Partial PIM information was found

^b File related information was discovered, but unable to display graphics except for .png files

^c Some file related information was discovered, but unable to extract text and graphics files from .tar and .exe archives

^d Some file-related information and file content was recovered, but not all

^e Since the device maintains the file system in ROM, all information was recovered after a hard reset of the device

Conclusions

Forensic examination of PDA devices is a small, but a growing subject area in computer forensics. Consequentially, PDA forensic tools are a relatively recent development and in the early stages of maturity. While the tools discussed in this paper performed well and have adequate functionality, new versions can be expected to improve and better meet investigative requirements.

As with any software, it is important for an examiner to understand the functionality and scope of forensic tools being used. It is especially critical for a forensic examiner to understand a tool's

limitations and when to turn to other means of examination. Proper detailed documentation of the functionality of a tool is essential. However, practice with a tool in mock examinations can help the examiner gain an even better understanding of the tool's capabilities and limitations, which often involve subtle distinctions. Practice examinations also provide the opportunity to customize facilities of the tool for later use.

References

- [Aye04] Rick Ayers, Wayne Jansen, PDA Software Tools: Overview and Analysis, NIST Interagency Report (IR) 7100, August 2004, <URL: <http://csrc.nist.gov/publications/nistir/nistir-7100-PDAForensics.pdf>>.
- [Bob04] Tanker Bob, JackSprat and JackFlash for Palm OS, PDA Buyer's Guide, May 2004, <URL: http://www.pdabuyersguide.com/software/JackSprat_JackFlash.htm>.
- [Cas04] Eoghan Casey, Chapter 13: Forensic Examination of Handheld Devices, Digital Evidence and Computer Crime, 2nd edition, Academic Press, March 2004.
- [Fae01] Nils Faerber, You Sexy Thing: Compaq iPaq on test, Linux Magazine, Issue 3, December 2000, <URL: <http://www.linux-magazine.com/issue/03/iPAQ.pdf>>.
- [Fae03] Nils Faerber, Pocket Power: Three new Linux PDAs in test, Linux Magazine, Issue 36, November 2003, <URL: http://www.linux-magazine.com/issue/36/Linux_PDAs_Tested.pdf>.
- [Ges03] Windows CE Embedded PC: Developer's Documentation, Version 3.0, Gesytec GmbH, August 2003, <URL: <http://www.gesytec.de/common/pdf-downloads/epc/embedded-pc.pdf>>.
- [Gra02] Joe Grand, pdd: Memory Imaging and Forensic Analysis of Palm OS Devices, Proceedings of the 14th Annual FIRST Conference on Computer Security Incident Handling and Response, June, 2002, <URL: <http://www.first.org/events/progconf/2002/d3-04-grand-paper.pdf>>.
- [Hal01] Chris Halsall, Linux on an iPAQ, Linux DevCenter, O'Reilly Media, Inc., June 2001, <URL: http://www.linuxdevcenter.com/pub/a/linux/2001/06/01/linux_ipaq.html>.
- [Hon04] Martyn Honeyford, Running Linux on an iPAQ: Put a penguin in your pocket, IBM developerWorks, September 2004, <URL: <http://www-106.ibm.com/developerworks/linux/library/l-ipaq.html?ca=dgr-lnxw25iPaq>>.
- [Jan04] Wayne Jansen, Rick Ayers, Guidelines on PDA Forensics, Special Publication 800-72, The National Institutes of Standards and Technology, November 2004, <URL: <http://csrc.nist.gov/publications/nistpubs/800-72/sp800-72.pdf>>.
- [Kin01] Joe Grand (Kingpin) and Mudge, Security Analysis of the Palm Operating System and its Weaknesses Against Malicious Code Threats, August 2001, pp. 135-152, Proceedings of the 10th Usenix Security Symposium, <URL: http://www.usenix.org/events/sec01/full_papers/kingpin/kingpin_html>.
- [MSC04] Files, Databases, and Persistent Storage - Microsoft Windows CE 3.0, Microsoft Corporation, February 2004, <URL: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/wcefiles/html/wcesdk_Files_Databases_and_Persistent_Storage.asp>
- [PS04] Palm OS Programmer's Companion, vol. I, PalmSource, Inc., November 2004, <URL: <http://www.palmos.com/dev/support/docs/palmos/PalmOSCompanion/CompanionTOC.html>>

- [Pie99] Claire Pieterek, How to get an extra 824K using FlashPro, PalmPower Magazine, May 1999, <URL: <http://www.palmpower.com/issues/issue199905/flashpro001.html>>.
- [Wie02] Officer Fred J. Wiechmann, Processing Flash Memory Media, New Technologies Inc., November 2002, <URL: <http://www.forensics-intl.com/art16.html>>.
- [Woo01] David Woodhouse, JFFS : The Journalling Flash File System, Ottawa Linux Symposium, July 2001, <URL: <http://sources.redhat.com/jffs2/jffs2.pdf>>.
- [Zwi02] Thomas Zwinger, Leif Laaksonen, Linux on an iPAQ PDA, @CSC, CSC - Finnish IT Center for Science, Issue 3, 2002 <URL: <http://www.csc.fi/lehdet/atcsc/atcsc3-2002/ipaq.pdf>>.