Acquisition of a Symbian Smart phone's Content with an On-Phone Forensic Tool

Pontjho M Mokhonoana and Martin S Olivier
Information and Computer Security Architectures Research Group
Department of Computer Science
University of Pretoria
Email: pontjho@tuks.co.za

***Abstract*— A smart phone is a handheld device that combines the functionality of a cellphone, a personal digital assistant (PDA) and other information appliances such a music player. These devices can however be used in a crime and would have to be quickly analysed for evidence. This data is collected using either a forensic tool which resides on a PC or specialised hardware. This paper proposes the use of an on-phone forensic tool to collect the contents of the device and store it on removable storage. This approach requires less equipment and can retrieve the volatile information that resides on the phone such as running processes. The paper discusses the Symbian operating system, the evidence that is stored on the device and contrasts the approach with that followed by other tools.**

I. Introduction

The use of mobile phones has grown in recent years. This can be attributed to their increasing functionality, usability and affordability. An increasing number of mobile phones, in addition to the functionality to make and receive calls, have added PDA functionality and thus, are referred to as smartphones. The added functionality of smartphones includes Personal Information Management, Internet browsing and multimedia capabilities. More often than not, smartphones will also run a smartphone operating system such as Symbian, Windows Mobile or Linux. This smartphone OS is different from the embedded operating systems in that it allows native third party applications on the hardware.

The increasing use of these smartphones means that they are progressively more involved in digital investigations. A digital investigation, according to Palmer [Palmer, 2001],

> *is the use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation, and presentation of digital evidence derived from digital sources for the purpose of facilitation or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorised actions shown to be disruptive to planned operations*

A smartphone can be involved in a crime in a couple of ways, it can be an *instrument* of a crime, a *target* of a crime or a *storage* of evidence. For this reason, investigators are increasingly becoming aware of the importance of the data retrieved from these devices as they contain a lot of traces of the activities of the user. The right tools and procedures have to be employed in order to ensure the integrity of the acquired data, otherwise it may be damaged or rendered inadmissible in court.

A number of commercial and open-source forensic tools [Jansen and Ayers, 2005], [Williamson et al., 2006] exist for acquiring data from smartphones. The use of these tools also require a lot of computing equipment which makes it difficult to perform acquisitions outside the lab.

In this paper, another approach of placing the forensic tool on the device in order to acquire data is suggested. This approach does not require interfacing with the PC connectivity services as it accesses the data directly from the operating system. This approach is critically discussed and its strengths compared to the traditional methods used by existing tools.

The remainder of this paper starts with an overview of the Symbian operating system. Section 3 describes the different methods for retrieving data from a smartphone. This is followed by a description of the approach suggested in this paper. The paper ends with a critical discussion of the tool and a comparison with other forensic tools.

II. Symbian OS

This section describes Symbian OS to give the reader some background on the platform which is being discussed. Symbian OS [Mery, 2003], [Siezen, 2005] is an advanced operating system for mobile devices. The OS works with open standards and is licensed by leading mobile phone manufacturers [Vaughan-Nichols, 2003]. Symbian supports several user interfaces (UI) from a wide range of device categories. These include the Nokia Series 60, UIQ, and the NTT DoCoMo. The operating system employs a layered architecture as illustrated in figure 1. The different layers are discussed below.

A. Kernel and Hardware Integration

The Kernel and Hardware Integration layer is the lowest layer in the operating system and it provides the abstraction that allows design over multiple platforms, making it easier to port to new types of hardware. This layer ensures Symbian OS robustness, performance and efficient power management. It consists of kernel services which implement scheduling policy, does power management and allocates memory to processes and device drivers which provide software controllers for devices on the phone such as the keyboard, screen, etc.

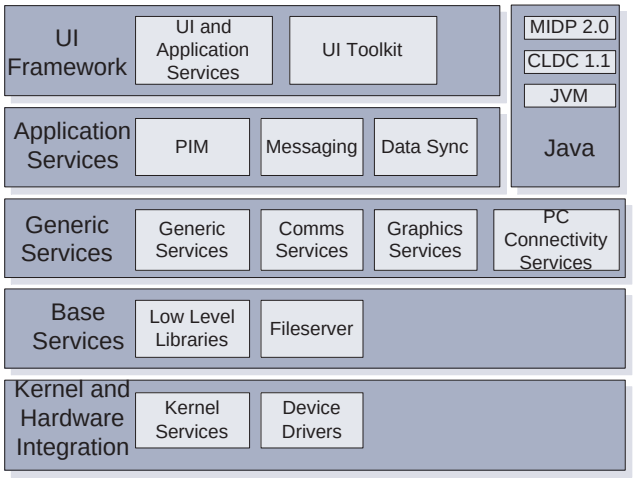

Fig. 1. Overview of Symbian OS

B. Base Services

The base subsystem provides the programming framework for all other Symbian OS components. It consists of the fileserver and low level libraries. The fileserver provides other components shared access to the file systems and a framework for mounting dynamic file systems. The low level libraries provide generic libraries utilities needed by other components and third party applications. They include character coding support, database engines and the cryptography library.

C. OS Services

The OS Services layer provides components vital to the OS infrastructure. These include the generic, Comms, Graphics and PC Connect services. The generic services consist of the multimedia and cryptography services. Multimedia services support the camera, Bluetooth headsets, OpenGL ES and provide media device framework. The Cryptography services enable data confidentiality, integrity and authentication by providing underlying support for secure communications protocols.

D. The Comms Infrastructure subsystem

The Comms Infrastructure subsystem provides important frameworks and system services for communications and networking. It consists of the Telephony, Serial and Shortlink and Network Services. The telephone system provides an API that allows communications using global network standards such as GSM, GPRS and EDGE. The Serial and Shortlink services provide point to point communications with other devices and peripherals. The Networking Services provide frameworks for implementing protocols over the socket interface.

The Graphics services provide applications with shared access to the keyboard, screen, pointing devices and fonts. PC Connect Services provide a toolkit which allows the phone's services to be accessed from the PC.

E. Application Services

The Application Services provide the application engines for the core mobile phone applications such as contacts, calendar, browsing, messaging and phone synchronisation.

F. UI Framework

The UI Framework is a powerful environment that allows Symbian OS licensees to create different user interfaces while maintaining maximum compatibility for developers.

III. Data Acquisition Methods

It is very important that evidence be collected in a forensically sound manner and this entails following principles that ensure the integrity of the data. A set of principles [Ayers et al., 2005] guiding the process were suggested and are quoted below.

1) *No actions performed by investigators should change data contained on digital devices or storage media.*
2) *Individuals accessing original data must be competent to do so and have the ability to explain their actions.*
3) *An audit trail or other record of applied processes, suitable for independent third-party review, must be created and preserved, accurately documenting each investigative step.*
4) *The person in charge of the investigation has overall responsibility for ensuring the above-mentioned procedures are followed and in compliance with governing laws.*

The first principle states that no action performed should change the data contained in digital devices. This is however not possible with mobile phones since the phone has to be kept on in order to be able to acquire data from it. Switching on the phone or connecting the phone to a computer will very likely change some data, even without explicitly doing so. This means that in the best case, data must be modified as little as possible [Carrier, 2002]. A number of approaches exist for capturing that data from the devices so that it can be analysed. These approaches are discussed below.

A. Manual Examination

A manual examination is the simplest but most tedious means of capturing data from a smartphone device because of all the manual work involved. The examiner manually copies information through the device's native applications. This is performed if the investigator requires only a particular piece of evidence from the device [Casey, 2004, pg345] or when there is no other way to access that information. It is advisable that the examiner be familiar with the operation of the device in order to reduce room for error. This can be achieved by practising on an identical test device or an emulator. It is important to record all actions taken on that device in order to allow the verification of the actions taken by others.

B. Connectivity Services

The most commonly used method of acquiring data from a mobile device is through the use of command-response protocols to interact with the Connectivity Services. Tools which employ this method either use open protocols such as the AT Command Set [SMG, 1999], SyncML [OMA, 2001] or OBEX [Megowan et al., 1999], or proprietary protocols like the Nokia FBUS [Kot and Zoltan, 2006].
There exists wrappers or PC connectivity SDK's [McDowall, 2004] which enable application developers to write programs which make use of mobile phone services without having to implement the underlying protocols.

The most common tools which can access the device's memory are the manufacturer tools. These are software packages for data synchronisation of the device and computer. These applications are however not designed for forensics and in using them, there is a risk of altering the data on the phone if not used properly. Other non-forensic third party applications which can perform the same function also exist.

The most widely accepted method for obtaining data from the mobile is the use of mobile forensic tools such as XRY, Oxygen Phone Manager - Forensic Edition and Paraben Device Seizure. These tools use the same protocols as the non-forensic tools but do not implement commands that explicitly modify the contents of the phone. This does not guarantee that their actions will not modify the contents of the phone [McCarthy, 2005] since a command, even though it is requesting data, may cause the operating system to change the item being requested. Table I list the commonly used mobile forensic tools. Some of the tools are explained in more detail later in the paper.

Tool	*PDA*	*GSM*	*CDMA*	*SIM*
PDA Seisure	X	X	X	-
Pilot Link	X	X	X	-
EnCase	X	-	-	-
Device Seizure	X	X	X	X
GSM XRY	-	X	X	X
Oxygen PM	-	-	X	-
MobilEdit	-	-	X	X
BitPIM	-	X	X	-
TULP 2G	-	X	X	X
SIMIS	-	-	-	X
Forensic SIM	-	-	-	X
Forensic CR	-	-	-	X
SIM Con	-	-	-	X

TABLE I
OVERVIEW OF MOBILE PHONE TOOLS

C. Connection Agent

Another method of retrieving data from a mobile phone is the use of a connection agent. A connection agent is a small program that is placed on the target device to enable connection establishment and data exchange between the phone and the tool. This approach uses a client-server architecture with the agent acting as the server. Without the agent, the data retrieval tool is unable to retrieve the data from the device. This approach is similar to the one above, except instead of relying on the phone's connectivity services, custom connectivity services are used to gain access to the device's data. The one problem with this method is that it requires a piece of software to be loaded on the device, thus modifying the target device.

D. Direct Access

Directly accessing the phone's memory is the most forensically sound but also the most challenging [Willassen, 2005] of the methods of retrieving data from a mobile phone. This method enables the imaging of the entire phone's memory, making it possible to access deleted or partially overwritten entries. It also bypasses the phone's security measures which would otherwise prevent access to the phone's contents. Another advantage to this method is that there is no reliance on the operating system to provide the correct results. This is not the case with the above mentioned approaches.
Willasen [Willassen, 2005] proposes two methods which are forensically sound:

- Remove the phone's memory chip so it can be accessed directly.
- Connect the phone to a motherboard to access the memory.

These proposed methods require a very high level of technical expertise which is usually not available to the average law enforcement agency. They require detailed knowledge of the phone's inner workings; therefore a fair amount of research is required in order to be able to perform proper analysis. It is therefore infeasible that these methods could be used by the average investigator since he will not possess the required knowledge.

E. Service Provider

An alternate means of getting information about a mobile phone is obtaining it from the service provider [Goode, 2003]. This is because a mobile phone typically functions in a GSM [Croft, 2004] network environment. When the network subscriber accesses a service from the network, it will be recorded by the service provider for billing purposes. If the examiner has the authority, he can request this information directly from the service provider. The information from the service provider will be more reliable than that from the mobile since it is more difficult to tamper with the service provider's records. It can also serve as a way to validate the data obtained from the mobile during an initial examination. The big drawback with this source of data is the limited amount of data that can be retrieved using this method. This is because there is a lot of data which will not be captured by the service provider either because of cost constraints or because it is not transmitted over the network.

IV. TOOL REQUIREMENTS

A forensic tool for mobile devices should meet the criteria stated below for it to be most effective. The requirements given are a measure of the ability of the tool to retrieve and only to retrieve data. It is for this reason that issues concerning the presentation of results, compression, and integrity verification

are not considered. The requirements are summarised below. This summary is followed by a more detailed explanation of each requirement.

1) The method should minimise changes to the device.
2) The method should retrieve as much data as possible.
3) The method should minimise user interaction with the device.

The method of examining a mobile phone should minimise the loss or change of data on that device [McCarthy, 2005, pg23]. The method used should preferably not place or remove data from the device. In addition, it should modify as few files as possible.

The data retrieved by a forensic tool should mirror that on the device as possible to reduce or eliminate the need to access the original [Shinder, 2002, pg 522]. This means that the tool must get as much data as possible from the device and that retrieved data should closely match the data on the device.

The forensics analyst who examines a digital device must prove that his actions did not compromise the evidence when it is presented in court. This is achieved by documenting all the actions performed on the target device. It can later be verified that the sequence of actions did not compromise the integrity of the evidence. It can be seen that the more interaction with the device, the more complex it would be to prove the integrity of the evidence as the examiner could omit the documentation of certain steps or even make mistakes. For this reason, it is important that the method employed to extract the data should require as little interaction as possible.

V. On-Phone Tool

This paper suggests a different approach to getting data from the device. Instead of using connectivity protocols to connect and extract data to a computer, this paper suggests using a tool that executes directly on the device. This approach is similar to the one used by duplicate disk (dd) for Linux based PDA devices [Kai Wee and Wong, 2006]. The criteria described above were used as a guide for developing the tool.

The first problem that had to be tackled is the deployment of the tool onto the device. A number of ways to place the on-phone tool on the device were considered. The tool can be packaged as a SIS installer, so it can be sent to the phone using Bluetooth, infra-red or file transfer using the PC suite. A SIS file is a special software installer for the Symbian platform. This method of using an installer however is not preferable since it requires that data be placed on the phone. This violates the first requirement that the method should change the target device as little as possible. The more ideal method is to place the tool on a memory card which is then inserted into the phone. Even though it may change certain parts of the operating system, the changes are very little compared with placing an entire installer which still has to be extracted.

Another consideration that had to be made is the platform to develop on. There are basically two options for the platform: the Java or the native platform. The use of Java as platform has a portability advantage; the same byte-code is guaranteed to run on all devices with a compatible runtime environment. Alternatively, one can write a native application using Symbian C++. Even though this locks the program to the Symbian operating system, it allows a more lower level access to operating system interfaces, thus allowing access to more data. With reference to the second requirement, writing a native application is preferable.

In order to minimise interaction with the device, the tool was designed such that it auto-starts either when the removable card with the program is inserted, or when the device boots. Recognizers [Bustarret, 2003] are used to auto-start the application. Because the application starts automatically, it is able to bypass the security password which would otherwise prevent access to the phone provided the phone was already powered. The phone's security mechanisms will not be bypassed if the phone was off. Note that not all Symbian handhelds have a slot for a memory card but most of the newer models will have a removable memory slot [Symbian, 2007].

As mentioned above, the tool is only able to perform acquisitions of data from the device. The retrieved data is placed on the same memory card on which the tool resides. The memory card can then be acquired and analysed using other forensic tools. It is important for the tool to be acquired with the files since this will allow the workings of the tool to be verified later.

The tool is only able to perform a logical copy of the file system, meaning that deleted files are not recoverable. Another problem with the tool is that it cannot copy files which have open handles. Since system critical processes cannot be terminated, it is not possible to gain access to certain files. Typically, the following files will not be copied:

- C:/system/data/CallLog.dat
- C:/system/data/CallScreening.dat
- C:/system/data/cdbv3.dat
- C:/system/data/Contacts.cdb
- C:/system/data/settings.db
- C:/system/data/smsreast.dat
- C:/system/data/smssegst.dat
- C:/system/data/TotalCallCounters.dat
- C:/system/data/VCStore.dat
- C:/system/data/wapreast.dat

Some of the files which could not be copied are very important. The *CallLog.dat* file stores information about recently made calls while *Contacts.cdb* stores contact information. Even though information on recently made calls cannot be retrieved, contact information stored on the device can be retrieved by using the PIM application services. Data which could be copied by the tool includes:

- Messages (SMS, MMS and Email)
- Multimedia (Audio, Video and Pictures)
- Applications
- Internet cache
- User Files

The biggest challenge preventing access to the files, and direct access to the entire disk is the difficulty of releasing the file handles. There was a problem with the specific phone that was used for testing (P800) in that when the backup API is used to temporarily release the file handles of system critical process, the entire operating system crashed. It has not been

determined if this problem is specific to the model or in all Symbian OS v7 phones.

VI. Comparison with other Tools

The next section describes other forensic tools which support Symbian OS based devices. The information provided is based on the versions at the time of writting. The list of tools listed is not complete and other tools may have been omitted.

A. XRY

XRY is a forensic software toolkit that supports a variety of mobile phone devices. It uses SyncML and OBEX for Series60 based devices and a connection agent for UIQ based Symbian devices.

For the purposes of this paper, XRY version 3 was tested and it is able to retrieve the following data when using SyncML and OBEX:

- Contacts
- Notes
- Videos
- Audio
- Pictures
- A subset of the filesystem.

SyncML was used to retrieve all the above items except for the filesystem. OBEX was used to retrieve the files and only a very small subset of the filesystem was retrieved.
When the connection agent was used with a UIQ based device, most of the files were retrieved. Some files however could not be retrieved and this is probably because of operating system restrictions. All the files which could not be retrieved using the on-phone tool, also could not be retrieved using the XRY connection agent. The connection agent, like the on-phone tool, could not retrieve deleted information.

B. Oxygen Phone Manager II: Forensic Edition for Symbian Devices

The original Oxygen Phone Manager (OPM) is a cellphone management tool; the difference between the standard and the forensic version is that the forensic version prohibits modifications to the target device. OPM uses a connection agent [Software, 2007] to enable interaction with the device. There are six versions of the agent, one for each flavour of the operating system. OPM allows the extraction of the following data:

- Call information
- Messages
- Connection logs (Wi-Fi, CSD and GPRS)
- PIM related data
- Installed applications and games information
- Multimedia
- Flash card files

When evaluated against the requirements for a mobile phone examination tool, OPM and XRY do not fare very well in fulfilling the requirements. They both require an agent to be placed on the device, which violates the first requirement; and they also require a lot of interaction with the device to get the agent to connect with the workstation. This is especially true for OPM, which requires the most amount of interaction to successfully establish a connection.

There is one drawback to the on-phone tool that the other tools also suffer from. Even though the tool does not explicitly initiate writes to the device, it does not prevent the operating system or other system processes from modifying the device's content. For example, if the device is not set to flight mode, it can still receive new messages. This, in most cases is not desired since the state of the device must be preserved by as much as possible after taking it into custody.

VII. Conclusions

In this paper, different methods of retrieving data from a Symbian mobile phone were discussed. In addition, the use of an on-phone tool is presented. This approach is similar to the connection agent approach and the kind of data retrieved by the prototype and the tools mentioned is the same. In all cases, there was some data which could not be obtained because of operating system restrictions. The biggest advantage to using a standalone program is that it does not rely on a computer for operation. This means that devices can be acquired at the scene, which is advantageous in certain cases and it avoids the possibility of the device being compromised during transportation. Another advantage, is the fact that nothing has to be installed on the device, meaning that there are fewer modifications made to the target mobile device.

When compared to desktop forensics, mobile device forensics is lagging behind. It is still not possible, using the current generation of tools to access the entire memory chip, meaning that deleted files or fragments thereof cannot be recovered. This approach has a lot of potential to achieve that goal and future research will be directed at developing better methods and tools of getting data from mobiles.

References

[Ayers et al., 2005] Ayers, R., Jansen, W., Cilleros, N., and Daniellou, R. (2005). Cell phone forensic tools: An overview and analysis. Technical Report NISTIR 7250, National Institute of Standards and Technology.

[Bustarret, 2003] Bustarret, E. (2003). Writing a recognizer. http://newlc.com/Writing-a-recognizer-The-EZBoot.html.

[Carrier, 2002] Carrier, B. (2002). Open source digital forensic tools - the legal argument. Research report, At Stake.

[Casey, 2004] Casey, E. (2004). *Digital Evidence and Computer Crime*. Elsevier Academic Press, 2nd edition edition.

[Croft, 2004] Croft, N. (2004). Secure interoperation of wireless technologies. Master's thesis, University of Pretoria.

[Goode, 2003] Goode, A. (2003). Forensic extraction of electronic evidence from gsm mobile phones. *Secure GSM and Beyond*, 10059(9):1–6.

[Jansen and Ayers, 2006] Jansen, W. and Ayers, R. (2006). Guidelines on cell phone forensics. Technical report, National Institution of Standards and Technology.

[Jansen and Ayers, 2005] Jansen, W. A. and Ayers, R. (2005). An overview and analysis of pda forensic tools. *Digital Investigation*, 2(2):120–132.

[Kai Wee and Wong, 2006] Kai Wee, C. and Wong, L. W. (2006). Forensic image analysis of familiar-based ipaq. *Forensic Focus*.

[Kot and Zoltan, 2006] Kot, P. and Zoltan, B. (2006). gnokii project. Website.

[McCarthy, 2005] McCarthy, P. (2005). Forensic analysis of mobile phones. Master's thesis, University of South Australia.

[McDowall, 2004] McDowall, I. (2004). *Programming PC Connectivity Applications for Symbian OS*. Wiley.

[Megowan et al., 1999] Megowan, P., Suvak, D., and Kogan, D. (1999). Irda object exchange protocol. Technical report, Infrared Data Association.

[Mery, 2003] Mery, D. (2003). Symbian os version 7.0 functional description. Technical Report 1.5, Symbian Ltd.

[OMA, 2001] OMA (2001). Syncml sync protocol. Technical Report 1.0.1, Open Mobile Alliance.

[Palmer, 2001] Palmer, G. (2001). A road map for digital forensic research. Technical report, First Digital Forensic Research Workshop.

[Shinder, 2002] Shinder, D. L. (2002). *Scene of the Cybercrime - Computer Forensics Handbook*, volume 36655. Syngress - Shinder Books.

[Siezen, 2005] Siezen, S. (2005). Symbian os version 9.1 product description. Technical Report 1.1, Symbian Ltd.

[SMG, 1999] SMG (1999). Digital cellular telecommunications system (phase 2) - at command set for gsm mobile equipment (me). Technical Report ETS 300 642, European Telecommunications Standards Institute, 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE.

[Software, 2007] Software, O. (2007). *Oxygen Phone Manager II Forensic Edition v2.13 for Symbian OS smartphones*. Oxygen Software, http://www.oxygensoftware.com/en/, 2.13 edition.

[Symbian, 2007] Symbian (2007). Symbian phones. http://www.symbian.com/phones/index.html.

[Vaughan-Nichols, 2003] Vaughan-Nichols, S. J. (2003). Oss battle in the smart-phone market. *Computer*, 36(10-12):3.

[Willassen, 2005] Willassen, S. (2005). Forensic analysis of mobile phone internal memory. In *IFIP Int. Conf. Digital Forensics*, pages 191–204.

[Williamson et al., 2006] Williamson, B., Apeldoorn, P., Cheam, B., and McDonald, M. (2006). Forensic analysis of the contents of nokia mobile phones. In *Australian Digital Forensics Conference*. Edith Cowan University.

BIOGRAPHY

Pontjho M Mokhonoana is a researcher at the CSIR, DPSS. His field of study is digital forensics with a focus on mobile devices. He completed his honours in Computer Science at the University of Pretoria in 2005 and is now studying towards his Masters Degree in Computer Science. He is also an active member of the ICSA Research Group at the University of Pretoria.

Martin S Olivier is a professor at the Department of Computer Science in the School of Information Technology at the University of Pretoria. In addition to normal teaching and research duties, he is the research coordinator of the School of Information Technology. His current research interests include privacy and digital forensics as well as database, application and system security.

PM Mokhonoana and MS Olivier, "*Acquisition of a Symbian smart phone's content with an on-phone forensic tool*," Proceedings of the Southern African Telecommunication Networks and Applications Conference 2007 (SATNAC 2007), Sugar Beach Resort, Mauritius, September 2007 (Published electronically)

Source: http://mo.co.za