ISSN 2229-5518

Symbian phone forensics- An agent based approach

Deepa Krishnan, Satheesh Kumar.S, A.Arokiraj Jovith

Abstract— Smart phones like the older mobile phones are fast becoming a life style choice. These sleek devices with large amount of personal information in them make smart phone forensics, a key component in any criminal investigation. The paper presents a contrast between hardware and software approaches and highlights the key advantage of the software approach i.e. the speed at which actionable data can be made available with less technical knowhow. Moreover, the proposed plug-in based agent development provides an extensible framework to handle customizations that will matchup with each unique nuance of phone platform and model. The paper summarizes the finding by unveiling a prototype module development, using platform SDK and on-phone agent. The main focus is the simplicity and extensibility of proposed approach but at the same time the paper does warn about the possible impact to device memory and contrasts with other alternatives.

Index Terms— Cyber forensics, law enforcement, mobile computing, security.

1 INTRODUCTION

THERE is no device that has changed lives and has seen world wide adoption like our humble mobile phone. With the advent of smart phones and its integration with web and social networking we are at the cusp of another radical change. We are in the era where phones have horsepower equivalent to a PC, were phone based news reporting has brought about revolution and downfall corrupt of a regimes. These powerful devices in the wrong hands will be equally disruptive. In this changing background forensic analysis of phones has become even more challenging all when our law enforcement agencies need to process a wide range of handsets quickly and get all available information to the investigating officer.

Today smart phones have almost all the features of a laptop or a notebook computer.Analysis of such devices is a major agenda before the forensics community.The law enforcement agencies require sophisticated software as well as hardware tools for the proper analysis of digital evidence to bring the culprit before the court of law.

If you look at figure 1[9] it shows the current penetrations of mobile phones in relation to world population and how mobile phones usage stacks up in comparison with other technologies. As smart phones replace the current generation of phones we are looking at a massive redefinition of current process.

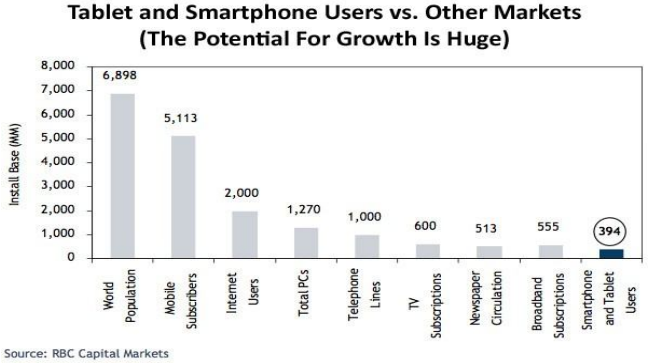

Fig 1 World mobile phone penetration and potential growth of smart phones

The capabilities and features of each handset determine what information could potentially be retrieved from each device. This is easier to understand if we look at this from the perspective of what particular tasks handset could perform. For example, older phones have limited memory so what we can expect to get is limited to data usually form the SIM card. Modern phones on the other hand have huge internal storage, which can further be extended by external memory card. Apart from a camera, many of the modern phones come equipped with GPS, compass, humidity sensor, proximity sensor, gyroscope and much more. The table 1 shows a comparison between different categories of phones.

TABLE 1
COMMON ATTRIBUES OF SMART PHONE

	Basic	Advanced	Smart
OS	Proprietary	Proprietary	Android,iOS, RIMOS, Palm OS, Symbian, Windows Phone7
Address Book	Basic Phone Book	Address Book with possible Calendar	Elaborate address book including special apps from app store.
Apps	Non existent	Basic pre- build App	Wide range of app selection from app store
eMail & Chat	None	SMS Chat	Wide range of chat & eMail app from the app store
Web	None	None or WAP Gateway	HTTP
Wireless	None	IrDA or Bluetooth	IrDA, Bluetooth, WiFi

Over the past few months the landscape of smart phone market share has drastically changed with Android and iOS making huge gains at the expense and in some cases downfall of the competition.

Before getting into the process details it is appropriate to look at the basic information present in smart phones. These can include but not limited to:

1. Handset Setting (language, date & time, GPRS,

WAP, internet etc)
2. Phone Book
3. Call Logs (incoming, missed and outgoing)
4. SMS Messages
5. Tasks
6. Calendar Events
7. Stored Files (pictures, music, video, audio recording etc)

2 FORENSIC PROCESS

While dealing with a digital device the method used to acquire data must have little impact on device memory as possible. Impact if any should be well understood and documented. This is important to ensure that integrity of the acquired data and also to allow for a 3rd party verification if it is required at any stage.

Forensics on a cell phone is considerably different from a personal computer. Even though the number of platform we have to deal with is reducing there is still a wide selection of proprietary OS along with a short product release cycle. Hence we will always be dealing with a moving target even within a single platform. Methodology and sequence in which the phone is handled and data collected is critical [4]. Turning off the phone has the potential to alter its memory or data on the phone, but leaving the phone ON raises the possibility of new information arriving over the network and affecting the integrity. Ideally the phone should be placed in a radio shield environment and it should only be switched OFF if that's not possible. On the same lines removal of SIM card or battery from some phones could modify the contents of phone memory. We recognize the complexity of this process and are developing an application based on the plug-in model, which allows for extensive customization based on the phone platform, model and version. The application will identify the expected steps and walk the user though it.

The following section describes some of the things a crime scene technician should consider as he/she goes though the evidence. The application helps the technician with proper reminders and logging the finding.

2.1 Keeping Track of Yourself

General guidelines for forensics require that, investigators cannot change the digital data content of the device being analyzed. Moreover, an audit trail of the analysis and investigation process should be maintained at all times in such a way that it can be verified by multiple sources. Each step should be accurately documented, so that there is enough information for the process to be reviewed by independent third-party. Finally, the person in charge of the investigation should maintain compliance with the governing laws.

Forensic method used should minimize changes on the device, be able to retrieve the full set of data, and finally minimize user interaction with the device itself. Ideally, the full memory content of a generic embedded device should be collected, to preserve the full inner state and obtain a forensically sound acquisition.

It is also recommended to keep track of approach and progress, by means of an external recording device (e.g. camera) that will maintain visual breadcrumbs.

2.2 Background Data

During the intake and processing of the phone evidence the crime scene investigator from law enforcement inputs a bunch of contextual information. This includes but not limited to where the evidence was found, the crime file details, the technician name etc. Capturing this kind of context information that can be kept with the analysis report is the first step of the forensic extraction tool.

2.3 External Forensic Data Source

There is information that can be gleaned from outside vis-à-vis the network can be as important as what is in the phone.For e.g. in the GSM network environment,a great deal of information might be recovered from the service provider.The set of information,which can be successfully collected with this method,is related to the SIM data set,such as SMS,MMS,list of last called numbers,and the location of the subscriber.Clearly,information such as photos,videos,phone book,web browser logs,audio recordings,or user's notes cannot be gathered in such a way.

If external forensic data can be gathered then the request for the information from the service provider or notes regarding this are recorded by the technician.

2.4 Physical Data Extraction

Physical acquisition implies a bit-by-bit copy of the entire physical store.Physical acquisition [5] has its advantages since it allows deleted files and data remnants in unallocated memory to be analyzed.Once a bit-by-bit copy is made the extracted image need to be parsed and decoded manually.Logical extraction of data implies copying data in logical file system partition through OS frame work calls.This is a logical view not raw memory view, which has its disadvantages.

There are not many effective tools available to take an effective physical image and parse it to something meaningful;most forensic tools for cell phones and SIMs acquire more technical knowhow and training.At the minimum the technician should know how to hook up to diagnosis or debugging ports like JTAG or at the extreme level may require de-soldering the flash chip and connecting to the reader.NOTE:De-soldering the flash chip is the most invasive method for the equipment so may not be the right approach in all cases.But this is ideal when the phone is damaged.If physical hardware based extraction is deemed useful the technical records those though-

ts in the report log.

2.5 Mobile Phone Communication Interface

Different interfaces [7], [12]like IrDA, Bluetooth or serial cable can be used to acquire logical content. Extracting data using serial cable is the recommended option; wireless options should be used only after understanding the potential forensic issues. E.g. Bluetooth requires the wireless antenna to be switched ON and requires key entries on the handset so that it is paired with the forensic workstation and a good connection is setup all of this generates integrity concerns.

2.6 Logical Data Extraction

This is the heart of the process and relies on multiple protocol and communication methods some of the things used are AT Commands, SyncML, FBUS, MBUS,OBEX,IrMC APDU.As you will see in further sections the phone OS platform SDK provides powerful options to extract data.Because each phone has its own unique approach;the plug-in model provides an extensible option to pick and choose what works best for the phone platform and model.

As you choose the phone model the correct plug-in that will do the job is called and used to extract the information.The extracted information and the final report is then run through a hashing algorithm(MD5) before saving it,to prevent tampering.

3 UNDERSTANDING SYMBIAN PLATFORM FOR PHONE FORENSICS

In the remainder of this paper we will dive deeper into developing the proposed module by choosing one of the smart phone platforms i.e. Symbian S60. We will start the discussion with an overview of the Symbian operating system and lay the groundwork of what we are dealing with. In sections following that describes different methods employed in retrieving data.

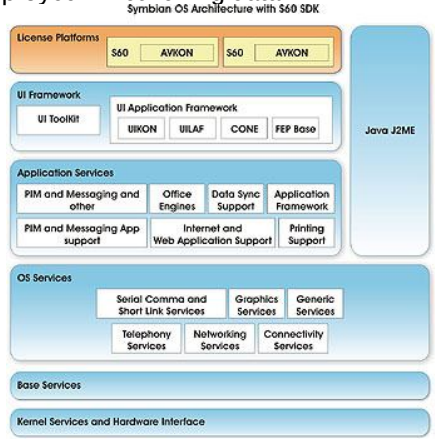

Fig 3 Symbian OS Architecture

3.1 Symbian OS Architecture

As can be seen in Figure 4, the architecture is modular, where operating system functionalities are provided in separate building blocks, and not in one monolithic unit. Being a single purpose phone OS Symbian [2] is single user with multi-tasking capability, being able to switch CPU time between multiple threads, giving the user of the mobile phone the impression that multiple applications are running at the same time.

The core OS is formed by a microkernel built as a personality layer on top of a real-time (RTOS) nanokernel. This block is responsible for primitives such as fast synchronization, timers, initial interrupts dispatching, and thread scheduling. Generally speaking, Symbian OS is intended to run on open, small, battery-powered portable computers, which are modern advanced state-of-the-art mobile phones.

3.2 Symbian File System

On a Symbian smart phone, the file system [5],[13] can be accessed by means of the file server component, also referred to as F32, which manages every file device. It provides services to access the files; directories and drives on those file mapped devices. The file server uses the client/server framework, by receiving and processing file-related requests from multiple clients. Moreover, it is able to deal with different file system formats, such as the FAT format used for removable disks, by using components that are plugged into the file server itself. In addition, it supports a maximum of 26 drives, each identified in DOS-like convention by a different drive letter, in the range A: - Z.

The main ROM drive, where the firmware resides, is always designated as "Z:". This drive holds system executables and data, which are referred to as XIP (eXecutable In Place) because they are directly launched without being loaded into RAM. Besides this, the firmware, or ROM image, is usually programmed into Flash memory, known also as EEPROM, the nonvolatile memory that can be programmed and erased electronically.

The C: drive is always designated as the main user data drive, which can be mapped onto the same Flash memory chip of the firmware, whereas any removable media device is generally designated as D: or E:. It is worth mentioning that every access from a client to file server (F32) takes place via a file server session, by means of RFs server session class, which implements all the basic services to interact with the file system.[3] We can obtain information about drives and volumes, act on directories, obtain notification about the state of files and directories, analyze file names, verify the integrity of the file system, and finally, manage drives and file systems.

3.3 Platform Security

Platform security [6],[10],[11] on Symbian OS v9 prevents applications from having unauthorized access to hardware, software and system or user data. The intention is to prevent malware, or even just badly written code, from compromising the operation of the phone, corrupting or stealing confidential user data, or adversely affecting the phone network. Every Symbian OS process is assigned a level of privilege through a set of capabilities, which are like tokens. A capability grants a process the trust that it will not abuse the services related to the associated privilege. The Symbian OS kernel holds a list of capabilities for every running process and checks it before allowing a process to access a protected service.

There are four different types of platform security capability, when digital signing is considered. The differences arise because of the sensitivity of the data or system resources the capabilities protect, and the requirements that are placed on the developer before they are given permission to use them. The capabilities of a process cannot be changed at runtime. The Symbian OS loader starts a process by reading the executable and checking the capabilities it has been assigned. Once the process starts running, it cannot change its capabilities, nor can the loader or any other process or DLL that loads into it affect the capability set of the process. A process can only load a DLL if that DLL is trusted with at least the same capabilities as that process.

The Symbian OS file system is partitioned to protect system files (critical to the operation of the phone), application data (to prevent other applications from stealing copyrighted content or accidentally corrupting data) and data files personal to the user (which should remain confidential). This partitioning is called data caging. It is not used on the entire file system; there are some public areas for which no capabilities are required.

Directory		Capabilities			
		None	AllFiles	TCB	AllFiles+TCB
\resource	Read	✓	✓	✓	✓
	Write	×	×	✓	✓
\sys	Read	×	✓	×	✓
	Write	×	×	✓	✓
\private\<ownSID>	Read	✓	✓	✓	✓
	Write	✓	✓	✓	✓
\private\<otherSID>	Read	×	✓	×	✓
	Write	×	✓	×	✓
\<anyOther>	Read	✓	✓	✓	✓
	Write	✓	✓	✓	✓

Fig 4 DataCaging Capabilities

However, some directories in the file system can only be accessed using certain capabilities. Each Symbian OS process has its own private folder, which can be created on internal memory or removable media. The folder name is based on the Secure Identifier (SID) of the process. A SID is required to identify each EXE on the phone and is used to create its private directory. With the previous Nokia phone generations, for instance the S40 series, the logical acquisition of the device content was possible, by means of Symbian OS APIs, which was able to copy the entire file system content on an external memory device. With S60 the access is restricted by data caging, the figure 4 shows table with folder security access (data caging) based on the application capabilities.

Interestingly, the file system restriction policy is fully contained in the file known as SWIPOLICY.INI [1], located in the folder z:\system\data\. The original policy file related to a Nokia Symbian based smart phone is illustrated as follows.

```
AllowUnsigned = false
MandatePolicies = false
MandateCodeSigningExtension = false
Oid = 1.2.3.4.5.6
Oid = 2.3.4.5.6.7
DRMEnabled = true
DRMIntent = 3
OcspMandatory = false
OcspEnabled = true

AllowGrantUserCapabilities = true
AllowOrphanedOverwrite = true
UserCapabilities = NetworkServices LocalServices
ReadUserData WriteUserData UserEnvironment
AllowPackagePropagate = true
SISCompatibleIfNoTargetDevices = false
RunWaitTimeoutSeconds = 600
AllowRunOnInstallUninstall = false
DeletePreinstalledFilesOnUninstall = true
```

It is interesting to observe that the capability set defined in the previous file is limited, and it restricts the interaction with the file system of the mobile platform. According to the standard documentation, the various parameters can appear in any order. Moreover, UserCapabilities set might be changed, by adding the required capabilities such as those illustrated in the following modified version of policy file.

```
AllowUnsigned = true
MandatePolicies = false
MandateCodeSigningExtension = false
Oid = 1.2.3.4.5.6
Oid = 2.3.4.5.6.7
OcspMandatory = false
OcspEnabled = true
AllowGrantUserCapabilities = true
UserCapabilities = AllFiles DiskAdmin NetworkSer-
vices
LocalServices ReadUserData WriteUserData
UserEnvironment MultiMediaDD NetworkControl
CommDD ReadDeviceData WriteDeviceData
SISCompatibleIfNoTargetDevices = false
AllowRunOnInstallUninstall = true
AllowPackagePropagate = true
DeletePreinstalledFilesOnUninstall = true
```

The illustrated policy file can be written directly in the original firmware of the phone and, subsequently, up-

loaded by means of re-flashing. As a result, the complete C: disk content might be collected, with standard self-signed APIs, and thus analyzed, to extract the full set of probatory data which might be usually found on a mobile platform. This is the usual approach for other mobile platforms as well, where the primary image, the one which contains the entire set of evidence, can be obtained without any restrictions. Unfortunately, such a scenario is far from the reality, and we need to evaluate others ways to access the digital data content of the smart phone.

So far, if an application needs to have the complete access to the phone file system, it has to be authorized by means of the Symbian signing procedure with AllFiles capabilities, which requires a special certificate released by TC Trust Center. Three steps are required to sign an application. Initially, the installation file generator, makesis.exe, creates the installation files (extension.sis) from information specified in the package file (extension.pkg). After that, if the application is for the international market, it will be signed with an ACS Publisher ID, by means of the Symbian Signed service. Conversely, it will be signed with a user-generated certificate, which might be created with makekeys.exe. Finally, the Installation File Signer (signsis.exe), digitally signs the installation files with the proper certificate, by generating, as a result, a .sisx file.

4 PROPOSED DATA COLLECTION SCHEME FOR SYMBIAN SMARTPHONE

This paper suggests a distinct approach both in development and extracting of data from the device. Application is developed as a composable part —A part provides services to other parts and consumes services provided by other parts. Each plug-in exposes a contract identifier so that it can talk to other parts in the application. The data extracting is based on client server architecture. During acquisition, the tool should have full access to the object store, as discussed there are very severe constraints to obtain a full physical image, so a logical copy of the object store is proposed. The client part is installed on the desktop PC the server part is copied into the mobile device giving us API access to extract a copy. The first problem that had to be tackled is the deployment of the tool onto the device. A number of ways to place the agent-based 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. Even though it may change certain parts of the file system, the changes are very little.

4.1 Data Acquisition

Data acquisition is the major step in forensics process. According to the forensics principle, the original data cannot be used for any forensics analysis. So we need to create a copy of the logical data present in the mobile device. This is achieved by developing an agent, which is to be installed on the target device. The module uses Symbian SDK, AT Command Set (SIM card commands), FBUS and Connectivity SDK to read the file system. The module is capable of establishing a connection and exchanging data with an externally connected host computer. Figure 5 shows setting screen of the acquisition GUI which allows the investigator to select the phone model, which further enables or disables the features available.

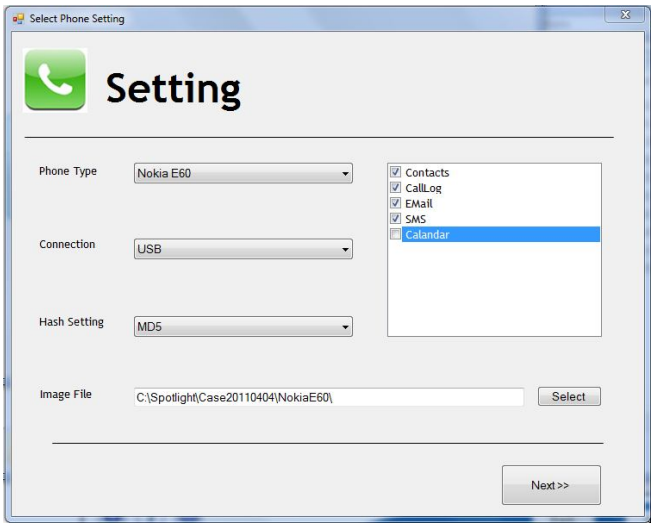

Fig 5 Symbian Acquisition Process

Since the tool uses standard APIs to access the file system and uses an agent, which is the code signed by the signing authority,
we can reasonably believe that these APIs will not change the device content. The main issue with this approach is that it requires a piece of software, called the agent, to be loaded on the device to acquire the content.

4.2 Data Analysis

The tool creates a logical copy of the data present in the mobile device as a file at the desktop PC, where the client programme is running. The tool also supports to generate the hash value, which will prove the authenticity of the acquired data. The image created can be loaded in an analyzer so that the data present in the mobile device can be viewed for further analysis. The tool provides important forensic information like Contacts, Call logs, SMS etc. This information will help the investigation agencies to get some cues for further investigation. The analysis tool shows all these information in separate file viewers. The incoming outgoing and missed call details are displayed separately. Also the Inbox, Outbox, Draft, Sent and Deleted SMS are categorized in separate viewers. The analysis tool is also provided with keyword and file search facility, which is the key feature of a forensics tool. User can add any keywords and file extensions in the box provided and the tool will search the entire

image for the string entered. It shows the search hits in a separate viewer.

5 CONCLUSION

To summarize, standardizing the process of digital forensic methodologies for mobile phones are still in their infancy stage, the kind of data we need to look for, the security paradigm are new. As the platforms evolve and mature we should see more robust imaging tools e.g. VMWare tools for Android platform. For now, the hardware approach seems the only one which should be able to give a bit-by-bit image of the flash memory content thus preserving the content of the investigated phone. But software approach works for acquiring specific items. For instance; it is certainly possible to extract the entire set of probatory data, such as SMS, MMS, pictures, video clip and phone book, by using application APIs.

ACKNOWLEDGMENT

We would like to thank Mr. Thomas K L, Joint Director, at Resource Centre for Cyber Forensics (RCCF), Centre for Development of Advanced computing (CDAC) Trivandrum, for his valuable suggestions and support. This work was done at the RCCF, CDAC, Trivandrum, Kerala, India.

REFERENCES

[1] Symbian-Ltd. Symbian OS library for application developers. Available at: http://www.symbian.com.

[2] Morris B. The Symbian OS Architecture Sourcebook: Design and Evolution of a Mobile Phone OS. John Wiley & Sons, Ltd, 2007

[3] Michael Aubert, with Alexey Gusev ... [et al.], Quick Recipes on Symbian OS, Mastering C++ Smartphone Development. John Wiley & Sons, Ltd, 2008. pp. 529–551.

[4] Svein Yngvar Willassen, Forensics and the GSM mobile phone system, The International Journal of Digital Evidence, Spring 2003, Volume 2 Issue 1

[5] Richard Harrison, Mark Shackman, Symbian OS C++ for Mobile Phones Volume 3. John Wiley and Sons, Ltd. pp. 204-206

[6] Michael Auburt, Quick Recipies on Symbian OS Mastering C++ Smartphone Development, John Wiley and Sons, Ltd. pp. 60-63

[7] Breeuwsma M., Jongh M. D., Klaver C., et al. Forensics data recovery from flash memory. In Small Scale Device Forensics Journal, 2007, 1.

[8] Nielsen 2011, Who is Winning the U.S. Smartphone Battle?[Online]Available: http://blog.nielsen.com/nielsenwire/online_mobile/who-is-winning-the-u-s-smartphone-battle/

[9] Dan Frommer 2011, businessinsider.com, RBC Capital Markets CHART OF THE DAY: 99.7% Of People Still Haven't Bought A Tablet Yet [Online] Available: http://www.businessinsider.com/chart-of-the-day-heres-how-huge-the-tablet-market-could-get-2011

[10] Jo Stichbury Symbian OS Explained, , John Wiley and Sons, Ltd. 2004, 1.

[11] Iain Campbell, Symbian OS Communications, John Wiley and Sons, Ltd. , 2007.

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

[13] Symbian-Ltd. Carbide.c++: Introductory White Paper. Forum Nokia, Version 1.1; 2007

- *Deepa Krishnan is currently Assistant Professor in PIIT,Mumbai University,India ,PH-91 977342 3043 ,E-mail:deepa@pointingarrow.com*
- *Satheesh Kumar.S is currently Scientist at CDAC Trivandrum, India,PH-91-9995090885,E-mail:satheeshks@cdactvm.in*