Lab 3 - Evaluation of forensic tools for mobile

Table of Contents

3.0 Objectives

The main objective of this lab is to get familiar with some tools for mobile forensics to the extent we can with the limited availability of them. Specific objectives of the lab are:

- To try out and evaluate the mobile device tools we have access to (see Appendix A) and compare the result with ordinary AccessData FTK for computers using a dd dump.
- To construct regular expressions that can find possible evidence in a binary dump using a hex-editor.
- Learn some basics about mobile forensics and a small intro to examination of SQLite databases.
- Optionally get practical or theoretical knowledge of SIM card forensics (see Appendix C).

Högskolan Dalarna
Röda vägen 3
781 88 BORLÄNGE

Telefon: 023-77 80 00
Telefax: 023-77 80 50
URL: http://www.du.se/

3.1 Methodology

3.1.1 Evaluation of tools

Use tools in appendix A to obtain a logical collection of user data and/or a physical memory dump of one of your own mobile devices or an Android emulator. Then "subjectively" grade the tools. To perform the grading use the instructions in Table 1.

In this lab we use the Android emulator when capturing a dd image. You can refer to the chapter 3 from the book "Android Forensics and Mobile Security"(included in the lab folder as a pdf) for more info about the Android SDK installation and setup. The main components that should be installed are ADB, a system image and an emulator. **Refer to Appendix D for more info.**

When done with the selected tools in appendix A capture a dd image of the /data/data partition of the Android emulator as described in appendix D. Then load the physical memory dump into FTK and give your opinion (Reporting and Accuracy) compared with tools from Appendix A.

Notes!

- For Android >= 7.0 (API-24) the location of some SQLite databases may have moved, see appendix D for more info.
- You may have to go back to this task after you have done the SQLite databases part.

Area	Weight	Description
Installation	10%	This cover installation, activation and updates of the forensic tool
Acquisition	10%	This covers the acquisition process
Reporting	20%	This covers the reporting process
Accuracy	60%	This covers the accuracy and completeness of the information acquired

Table 1. Grading table for evaluation of forensic software

If you like to have a good reading and check examples for how the grading can be performed I recommend the iPhone-Forensics-2009.pdf, which is included in the lab folder. In this report, you can also find other examples of test scenarios. **Note that you do not need to perform a similar thorough investigation as reported. In this case we just want to have an orientation of the field.** An example from a real world investigation of a cell phone can also be found in docs\sample_forensic_reports\ folder.

Högskolan Dalarna
Röda vägen 3
781 88 BORLÄNGE
Telefon: 023-77 80 00
Telefax: 023-77 80 50
URL: http://www.du.se/

3.1.2 Regular expressions

Download MSAB XACT at:

- [server]\embedded_forensics\MSAB.com\MSAB Forensic Pack v6.7**msab XACT 6.7 no installation.7z**

MSAB XACT is a powerful special Hex viewer for searching artifacts in mobile phone dumps via regular expressions. Please note that values usually are stored in hex in a reverse nibbled decimal format.

A web page (attached in the lab in the docs folder) that has very good info doing searches in phone dumps is: http://www.controlf.net/regexps/

Download the two low level compressed memory dumps of a phone with the name SonyEricsson_W800i_*.7z located at:

- [server]\embedded_forensics\cell_phone_dumps\

The dumps in the folder are either available as a XRY projects (.xry file), which have some limited system/user data in the dump decoded or as raw binary (.bin) files.

To find the unlock code and IMEI you need the XRY file, which can be viewed with both the XACT hex-tool and the XRY_READER_NOINST_* tool found at the same location.

If you want, make it possible to start up the phone during an investigation you need to find the Unlock Code and valid parameters for IMSI and ICCID to clone a SIM card.

Use XACT with the phone dumps to find these parameters. You need to double click on a memory range (subfolder to image) in order see the hex view.

An additional task for this lab is to construct two regular expressions that finds possible valid IMSI:s and ICCID:s candidates using the dumps with the XACT tool.

If you want to read more about regular expressions and abbreviations for telecoms a list of references are listed in Appendix B.

Högskolan Dalarna
Röda vägen 3
781 88 BORLÄNGE
Telefon: 023-77 80 00
Telefax: 023-77 80 50
URL: http://www.du.se/

3.1.3 Mobile forensics basics about images and SQLite databases

Unpack the sdcard-image.7z file which is found in the lab3 folder. Then answer some questions using Windows tools you have learned as FTK Imager, FTK etc. or Linux tools described below. Kali-Linux got all tools needed.

3.1.3.1 SD card image

Linux

Run $ mmls ~/Desktop/lab3/sdcard-image.dd

To get info about the image.

Hint, multiply the partitions starting sector (sector offset) by sector size to determine the byte offset of the partition from storage start.

If you would want to mount the partition in Linux the command is:

$ sudo mount -t vfat -o ro,loop,offset=BYTE-OFFSET ~/Desktop/lab3/sdcard-image.dd ~/Desktop/mount

$ sudo umount ~/Desktop/mount

3.1.3.2 Carving and timeline

Linux
Carving for evidence:
$ scalpel -c ~/Desktop/lab3/scalpel.conf ~/Desktop/lab3/sdcard-image.dd -o ~/Desktop/lab3/scalpel/

Using fls and mactime to create a timeline of the SD card image file:
$ mmls ~/Desktop/lab3/sdcard-image.dd (to find the offset)
$ fls -o SECTOR-OFFSET -z CST6CDT -s 0 -m '\' -f fat16 -r ~/Desktop/lab3/sdcard-image.dd > ~/Desktop/lab3/sdcard.body
$ mactime -b ~/Desktop/lab3/sdcard.body -z CST6CDT > ~/Desktop/lab3/timeline.csv

3.1.3.3 SMS database and deleted records

We will focus on locating evidence within a sms database file (sms.db) taken from a mobile phone.

Open the sms.db file in a SQLite database viewer of your choice. The method to view user data contained in the database depends of the viewer. SQLiteSpy and SQLite Database Browser are popular choices.

Select the table **message** where the SMS messages are located to browse logical SMS messages to get familiar how they are displayed.

Since the database viewer only will display records that have **not** been deleted we will need to use a hex editor to look for unallocated messages. SQLite stores deleted messages in free pages until garbage collection removes the records or database from the nand memory or they are overwritten with new data.

Linux

Högskolan Dalarna
Röda vägen 3
781 88 BORLÄNGE
Telefon: 023-77 80 00
Telefax: 023-77 80 50
URL: http://www.du.se/

Open the sms.db database in a hex editor (hexeditor) read-only:
$ hexeditor -r sms.db

To search for a word you have to first use the TAB key to get the cursor to the ASCII side of the file, then hit Ctrl-w to initiate the search. Once the search field is displayed simply type the search word and hit enter.
Now you can search for deleted records using for example parts of the phone number. Keep in mind that since the message may have been deleted that parts of the phone number might have been reallocated.

A good indicator on how many messages have been deleted is to look at the ROWID column in the SQLite database viewer. The ROWID is auto incrementing so it can be a helpful tool to reinforce that a message was deleted.

In this case (message table), you can see that some rows are missing. To determine whether any messages at the end are missing, select the "sqlite_sequence" table from sms.db. It will display the most recently assigned ROWID listed next to "message." In this case, 17 was the last assigned ROWID, so we can verify that there were no additional messages deleted after ROWID 17.

Another way (compared to hexeditor) to view the ASCII portion of the sms.db file is to perform a strings search on the db file.

Linux
$ strings sms.db > ~/Desktop/lab3/strings.txt
To determine if a record was deleted you will need to compare the SMS records within the strings output file with those found in the hex editor. The record should not be visible in the SQLite database viewer.

Use grep to find instances of a specific word.
$ grep SEARCHED-WORD ~/Desktop/lab3/strings.txt

Högskolan Dalarna
Röda vägen 3
781 88 BORLÄNGE
Telefon: 023-77 80 00
Telefax: 023-77 80 50
URL: http://www.du.se/

3.2 Reporting

3.2.1 Evaluation of tools

Compile the results in a small report where you will present the weighted grades for the tools, using the instructions in Table 1. For instance, you can use a scale ranging from 1 to 10 where 10 is the highest grade. You should also comment each area (installation, acquisition, etc.) for each tool. Andriller and MOBILedit (which you can run trial editions for 1-2 weeks) should have been installed and tested. Use FTK for with the dd image for verification. If Andriller stops change the bundled adb.exe to newer version.

3.2.2 Regular expressions

Compile the results in a small report where you will present the result of the investigation of the SonyEricsson_W800i_* dump:

a) unlock code,

b) IMEI,

c) regular expressions used to find IMSI:s candidates,

d) regular expressions used to find ICCID:s candidates for the MNC:s Telia, 3, Sweden 3G, Tele2 and Telenor.

Examples of IMSI and ICCID can be found in the lecture material.

3.2.3 Mobile forensics basics about images and SQLite databases

3.2.3.1 SD card image

a) What is the partition type (NTFS, FAT16, FAT32 etc.)?
b) What is the starting sector for the partition?
c) What is the offset of the partition?

3.2.3.2 Carving and timeline

a) Based on the information contained in the timeline, what happened on February 17 around 3:15 PM?
b) On February 14, between 5:00 and 5:05 PM, a file was deleted. Find the name of this file and try to recover it.
c) At what time was the Projections.txt file initially created?

3.2.3.3 SMS database and deleted records

a) How many deleted messages (sms and mms) were you able to recover and what text did they contain?
b) Search for the word “Italian” (case sensitive). How many instances of the word Italian did you find?

Högskolan Dalarna
Röda vägen 3
781 88 BORLÄNGE

Telefon: 023-77 80 00
Telefax: 023-77 80 50
URL: http://www.du.se/

3.3 Appendix A - Other professional tools that may be available (restrictions may arise)

Some tools and images are available at the digitalbrott share. To access the digitalbrott share follow this guide: https://wiki.du.se/%C3%84mnen_-_Subjects/Datateknik_-_Computer_Engineering/Software_and_distance_access_to_software.

3.3.1 Free tools

- Andriller - http://andriller.com/ (you can run trial for two weeks)
- MOBILedit - http://www.mobiledit.com/ (you can run trial for one week)
- Magnet Forensics got some free tools - https://www.magnetforensics.com
- Santoku Community Edition (rather old) – http://santoku-linux.com/

3.3.1.1 Deprecated free tools

- NowSecure Forensics Community Edition (NFCE) – [server]:\embedded_forensics\NowSecure.com (viaforensics)\
- NowSecure Lab/Apptesting Community Edition (NLACE) – [server]:\embedded_forensics\NowSecure.com (viaforensics)\

To activate use online activation, create a license container and then use the license key found in the info.txt file. If it do not work start again and use the offline activation with the RaU and RaC license files that were downloaded from the server. Then select license update.

3.3.2 Non-free tools

- AccessData MPE+ which require a license. The current release of MPE+ can be found at: http://accessdata.com/product-download. If you got problems to get a result from the NFCE tool with your physical phones you can use some of the investigator images found in the MPEInvestigator_Images.zip archive at our school server: [server]\embedded_forensics\Mobile Phone Examiner Plus\.
- If you perform the lab at school you may also use the MSAB office box: http://www.msab.com/ > Products > XRY to obtain a logical collection of user data and/or physical memory dump of your own phone (or a school phone). Use the MSAB Forensic Pack on [server]\embedded_forensics\MSAB.com\.
- There may also be a possibility to try out Cellebrite: http://www.cellebrite.com/ in the same way as MSAB if you are doing the lab at school.
- You may also try out the Oxygen forensic tool found on [server]\embedded_forensics\oxygen-forensic.com\ or: http://www.oxygen-forensic.com/.
- MOBILedit Forensic Express - http://www.mobiledit.com/
- If you got time you may try some other tools as Paraben Device Seizure tools, MOBILedit! Forensic or other related tool found on: [server]\embedded_forensics\common-tools\ or the internet.

Högskolan Dalarna
Röda vägen 3
781 88 BORLÄNGE

Telefon: 023-77 80 00
Telefax: 023-77 80 50
URL: http://www.du.se/

- Check out the folders in the [server]\embedded_forensics\docs\ directory where forensic methods are given for each mobile OS, see the README.txt file for more info. You may need to consult some of these resources in order to do your investigation in a forensic sound way.
- If you perform the lab at school and cannot use your own phone by some reason, you can use one of the schools phones as earlier mentioned. Available equipment at school is given in this file: [server]\embedded_forensics\README-list of available equipment.pdf. I have a ZTE Blade Android Phone as well.

3.4 Appendix B - Regular Expressions & Search Terms for Phone Examiners

- Link to regular expressions: http://www.controlf.net/regexps/

Abbreviations and explanations of the terms:

- MCC: http://en.wikipedia.org/wiki/List_of_mobile_country_codes
- MNC: http://en.wikipedia.org/wiki/Mobile_Network_Code
- IMSI: http://en.wikipedia.org/wiki/International_mobile_subscriber_identity
- IMEI: http://en.wikipedia.org/wiki/International_Mobile_Station_Equipment_Identity
- ICCID: http://en.wikipedia.org/wiki/Subscriber_identity_module#ICCID

3.5 Appendix C – Optional SIM card forensics exercise

This part of the lab can be really hard, if not impossible to do at home without a supported smartcard reader, which is provided if doing the lab at school.

Do the same evaluation as in task 3.1.1 but now for SIM forensic tools. You can use the MSAB equipment, MPE+, Oxygen Forensics, Paraben or other tool with the SCR smartcard reader. Also CHIPDRIVE Smartcard Commander 1.07 ([server]\embedded_forensics\common-tools\) can be fun to play with. There is at least around 10 SIM cards available for you to use if you do not want to use your own.

Instructions how to fill the SIM with data is found in the README-list of available equipment.pdf file. An example of a SIM report is found in the [server]\embedded_forensics\docs\common_documents\sample_forensic_reports\ folder.

If you got problems to get result from the tools you can use the SIM investigator image found in the MPEInvestigator_Images.zip archive at: [server]\embedded_forensics\Mobile Phone Examiner Plus\.

3.6 Appendix D - Use the Android Emulator for forensics

3.6.1 Preparation of the AVD

- First create an AVD (Android Virtual Device) with a SD card of around at least 2 GB storage (it depends on the size of the emulators or device data partition). It is **It is**

Högskolan Dalarna
Röda vägen 3
781 88 BORLÄNGE

Telefon: 023-77 80 00
Telefax: 023-77 80 50
URL: http://www.du.se/

good if the SD card can be able to store a dd image of the data partition. You can check the size of you data partition with the commands given at the end of section 3.6.3. In some cases the reported free size may lie since the virtually partitions are shared on newer devices.

- You should have an AVD with API-19 (Android 4.4.x) **or higher** since from this version **hw.useext4 = yes** is set in the AVD:s **hardware-qemu.ini** file. You can find the hardwareqemu.ini file at: "C:\Users\<username>\.android\avd\<AVD-name.avd>" on a Windows computer.

- If you use an emulator you probably **must** use an **ARM emulator** for some of the tasks to work in later labs (not needed in this lab). Therefore when we run native programs they need to be cross-compiled (since we are running them on the ARM architecture).

- If we use an older emulator or set the **useext4** variable to "no" we are stuck with the YAFFS2 file system which make extraction and parsing a lot more difficult. Most, if not all phones and tablets use the ext4 file system since 2011.
  Some limited devices from 2014 and beyond have been using the F2FS (Flash-Friendly File System) initially developed by Samsung Electronics: http://en.wikipedia.org/wiki/F2FS.

3.6.2 Preparation of user account and evidence

We are going to add/generate all possible kinds of user data to the AVD. Most of the data, if not all is going to be located in SQLite databases in /data/<package name> in the dd image, (/data/data/<package name> on a live system).

We should at least try to deal with the possible data that are present in the emulator as:

- Calendar
- Chrome web browser
- Camera
- Contacts
- Downloads
- Email
- Maps
- Messenger (SMS/MMS)
- Phone (Call Logs)

from the Android device.

A problem can be to know where the SQLite database files (*.db) which contains the forensic info is located on the user partition. Some examples are:
/data/data/com.android.providers.contacts/databases/contacts2.db
/data/data/com.android.providers.telephony/databases/mmssms.db

The package com.android.providers.contacts for example got the contacts2.db and the calllog.db. Some manufacturers however use non-standard locations.

Högskolan Dalarna
Röda vägen 3
781 88 BORLÄNGE
Telefon: 023-77 80 00
Telefax: 023-77 80 50
URL: http://www.du.se/

Note! For Android >= 7.0 (API-24) the location of some SQLite databases may have moved to the: /data/user_de/0/<package name> folders instead.

3.6.3 Capture the user data partition

Short instructions for how-to get the user data partition as an ext4 dd image from the Android emulator with netcat.

- Remember that you may have to open up the used netcat port in your personal firewall on your computer.
- The transfer may take some time, so start with a small partition as the cache partition first to know that you doing it right.

1. Start an administrator console on your local computer and get an ADB shell in the emulator.
C:\> adb shell

2. Become root and list the available partitions with some of the commands in the command reference below.
su

3. Start an another administrator console with a netcat listener which can create a dd image saved to your local computers drive.
C:\utils\netcat> nc -l localhost –p 31337 > c:\tmp\dump.dd

4. Dump the user data partition from the emulator, now the actual transfer begins.
dd if=/dev/block/vdc bs=4096 | nc 10.0.2.2 31337

5. When emulator reports something like the strings below you have successfully created and transferred the dd image.
204800+0 records in
204800+0 records out
838860800 bytes transferred in 66.028 secs (12704622 bytes/sec)

6. The dd dump can now be used in FTK as with any normal forensic investigation.

Command reference
Command list for getting info about partitions in Android.

1. **List the available partitions -** `# cat /proc/partitions`
2. **Disk free and file system block size -** `# df`
3. **Check the file system mount points -** `# mount`

3.7 Lab feedback

a) Were the labs relevant and appropriate and what about length etc?

b) What corrections and/or improvements do you suggest for these labs?

Högskolan Dalarna
Röda vägen 3
781 88 BORLÄNGE

Telefon: 023-77 80 00
Telefax: 023-77 80 50
URL: http://www.du.se/