Digital Forensics (DIFO)
Stockholm University
Department of Computer and Systems Sciences

Lab Assignment I: Introduction to Computer Forensics

Prof. Oliver Popov &
Alaa Altorbaq &
Jesper Bergman

Version 0.1
Autumn 2017

1 Introduction

In this laboratory exercise you will learn how to use a few common forensic tools, forensic and anti-forensic techniques. When working with this lab exercise, you are expected to read up on how to use the tools and therefore there will be limited assistance from the staff regarding questions concerning that. You should regard the laboratory staff as non-technical colleagues that have consulted you to conduct a forensic investigation. You will be provided with manuals for all tools that you will be asked to use. In addition, any search engine of your choice could be useful for finding information on how to use certain tools.

Please do regard the labs as any other examination process. Plagiarism is not tolerated and the occurrence of plagiarism will face disciplinary actions in accordance with the code of honour of Stockholm University[1]. This means that pictures and text - or other work that is not your own - must be referred to using an accepted reference system. At the department of computer and systems sciences we use the following reference systems: Oxford, Vancouver, IEEE, and Harvard - hence you must use any of those reference systems if you want to cite any source that you include in your assignment report/reports. You are required to read and accept the group policy on iLearn2 before starting working on this laboratory assignment. Please ask the course staff if you do not know how to use a reference system.

During the work with this laboratory exercise, you will learn more about the following basic forensic tools and concepts:

1. Hashing tools and file integrity
2. File headers and file identification
3. Anti-forensic techniques
4. Acquisition
5. Cracking encrypted files
6. Steganography and hidden messages

The files that are needed for completing this lab assignment are available on the computers in the CS2Lab. They are also available on: https://people.dsv.su.se/~jesperbe/DIFO/Lab1.zip. Some of the exercises can be done on a basic Windows/Linux system after downloading the files from there, though that is on your own responsibility.

[1] http://dsv.su.se/en/education/study-information/regulations

1.1 Purpose

This lab assignment is worth 1.5 ECTS - that is equivalent to one week of full-time studies per person[2]. We expect you to spend that time on this exercise, and it is designed to be completed within that time frame.

1.2 Lab Environment

You will mostly be working with the exercises on the physical Windows 7 computers (also referred to as the "native OS" or the "host OS") in the lab if you are not taking this course as a distance course. If you are taking this course on distance, please read the document in iLearn2 that is called *Getting Started with the Virtual Environment for DIFO*.

During this lab assignment you will be working with the aforementioned Windows 7 system, but also with VirtualBox and a virtual machine named "DiFo Kali" (might also be referred to as "Kali") which you will find in the list of virtual machines when you open up VirtualBox. The lab files that you will need are located on the Desktop (C:\Users\cs2lab\Desktop\Shared Folder) folder on the Windows (host) OS.

There are several tools that might be useful when completing this assignment's exercises. Feel free to use any other tool that you find sufficient on the computers [3]. On the Windows machine, most of the tools are available in the (C:\Users\cs2lab\Desktop\Forensic_tools) folder, or as executable shortcuts on the desktop, for example the programs PRTK, FTK, and Encase can all be launched by double clicking their icons. On the Kali machine, however, most of the programs are command line based - that means that you have to launch them in a terminal. If you are not familiar with Linux and command line programs, please check iLearn2 for resources that could help you get started.

1.3 Supervision and Resources

You are advised to read the literature and information found on the iLearn2 page of Digital Forensics, Autumn 2017 and also make use of the references in this document for additional material that could be very handy while attempting this lab. Please be aware that the lab staff will not give you any correct answers to the questions. Supervision will be available online as well as on-site. Check the iLearn2 course website for more information.

[2] http://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/hogskoleforordning-1993100_sfs-1993-100

[3] Given that it is legally acquired and used.

1.4 Reporting

For this lab assignment you are required to document all answers and other relevant information from each exercise in a report. The report must be written in English. For exercises 1-3, and 5-6 you should use the report template A. For exercise 4 you must use report template B. Both these templates are available on iLearn2 under Laboratory assignment 1.

Reporting is an essential part of a forensic investigation. Therefore you should, put effort into the report so that the person reading it can easily understand what you have done and also if they wish, repeat the investigation by following the report you have written; the report should be written in a scientific manner. Make sure to provide answers to the questions in an adequate way, that you use the correct template, and that you are taking into account the forensic soundness, the chain of custody etc. If have questions on how to answer the questions and/or write the report, please consult the course literature, the recommended reading material, or contact the course staff. The general structure of a forensic report is the following, and you are advised to follow this structure as far as possible when providing answers for, and information about, the exercises:

1. Description of the cases handled,
2. Description of the evidence and the chain of custody,
3. Method and tools used for answering the questions,
4. An explanation of what you found, and where you found it,
5. Conclusions.

1.5 Marking and Grading

This assignment is worth 1.5 ECTS (i.e. one week of full time studies) and is graded Pass or Fail. All exercises in this assignment are mandatory, unless they are marked: OPTIONAL (as exercise 2.2 is). You need 65 points out of 100 points to pass this assignment. Every question is marked with a number of points it is worth (all exercises are worth 15 points, except for exercise 4 "Acquisiton" which is worth 25 points). You must do all the exercises that are not marked as optional in order to pass the assignment. Generally, the questions' points are divided into: correctness (50%) and reflection and elaboration (50%), meaning you could get 7.5 points for a *correct* answer, but in order to receive 15 points, *elaboration* and *reflections* upon the answer and the question are needed. Of course, some questions ask for an answer which one cannot necessarily elaborate on; in that case all points will be granted for a correct answer. Please ask the course staff in the iLearn2 forum if this is unclear to you.

1.6 Lab Files

On the computers in the lab you will find a set of files that are listed below. If you prefer to work on your own computer at home or somewhere else, you can always download the files from https://people.dsv.su.se/~jesperbe/DIFO/Lab1.zip If you download it from the net, be sure to verify the SHA256 of it.

```
| DIFO_2017_Lab1.zip # SHA256: 9c5d0bfbeccd75858426cfc84345e0a68687b0fc5662b715153aa88ce
    | Lab1 # Main folder
        | Hashset  # Exercise 1
            | hello
            | hello(1)
            | hello(2)
            | hello(3)
        | Files # Exercise 2
            | 01
            | 02
            | 03
            | 04
            | 05
            | 06
            | 07
            | 08
            | 09
            | 10
            | 11
            | 12
            | 13
        | Antifiles  # Exercise 3
            | c.mp3
            | Suspicious_File
        | Acquisition  # Exercise 4
            | winxp.vdi
        | Cracking  # Exercise 5
            | casssh.pdf
            | ht.zip.tar.gpg
            | untitled.docx
            | Untitled 1.ods
            | wallet1.dat
        | Steganography  # Exercise 6
            | c1l.png
            | c2l.png
```

2 Exercises

Exercise I: Hashing (15 points)

As you should have learnt from the course literature and the lectures, hashing is an efficient technique for uniquely identifying files in order to maintain the chain of custody, and verify and maintain the integrity of the files. In this exercise you will learn how to hash files technically using different tools and different algorithms.

In the "Forensics tools" folder on the Desktop folder, you will find the tools that will help you with this. You can do this in Windows or Kali. Open up the Command prompt (or the Terminal if you use Kali) and cd into the Forensics_tools folder on the Desktop and run the hashing tools you find relevant in the md5deep-4.4\folder on all the files in the folder "Hashset" in the Lab1 (or other file that you would like to calculate the hash sum of). In order to confirm legitimacy (or non-legitimacy) of a file, its hash sum can be calculated and compared to a stored hash sum that has been confirmed to be benign (given that you can trust the stored hash sum of course). On Windows systems, you can find a system folder called System32 under C:\Windows\where the internal Windows calculator executable file and the command prompt executable files etc. are located. Calculate the hash sums of one of these two files and compare them to the hash sum database RDS_modern.txt that is available on the Desktop folder on the Windows and Kali machines. The hash sums database file - which really is a plain .txt file - is big, so it might not be the best idea to open it in a resource hungry graphical user interface program. Instead, it is recommended to run the findstr command[4] on the file to find matches for the hash sums that you have calculated from previous steps.

```
> cd C:\system32\calc.exe
> C:\Users\cs2lab\Desktop\Forensic Tools\md5deep-4.4\md5deep64.exe
C:\Windows\System32\calc.exe

> findstr <hash sum> C:\Users\cs2lab\Desktop\RDS.txt
```

Include the answers to the following questions in your report:

- What files (and/or folders) did you hash?
- Which algorithms did you choose?
- Suppose you would need to calculate the hash sums of several files and folders, how could you do that using the tools and resources provided? Provide a description and exact commands.

[4]The arrow (>) indicates a command prompt command in a Windows system

- Come up with an efficient way to match hash sums - so that you can automatically identify files that might (or might not be) of interest. Describe how you did this and which exact programs you used, and reflect upon why this might be useful for a forensic examiner in their work life.

2.1 Comparison of Hashing Algorithms

In order to compare the execution time of different hashing algorithms, you can use the Unix time command in a similar way as follows[5]. Choose a large file (100-200MB) of your choice (for example: https://ftp.eu.openbsd.org/pub/OpenBSD/6.1/amd64/install61.iso) and run (at least two) different hashing algorithms on it. This only works in the Kali machine:

```
$ time <hashingalgorithm> <file>
real    0m20,552s /* This is the process' total execution time */
user    0m4,667s /* This is the user's execution time */
sys     0m0,826s /* This is the system's execution time */

/* Exempli gratia */
$ time sha256sum /bin/bash
```

If you for some reason prefer to do this on a Windows based system, you could run the following command in Powershell (please note that md5deep64.exe could be substituted with something else):

```
> Measure-Command {start-process C:\Users\<user>\Desktop\Forensic Tools\md5deep64.exe
<file> -wait}
```

Include the following information in your report:

- Which hashing algorithms did you use? How long did each algorithm take to run?
- Provide the exact times for all hashing algorithms used.
- Provide the name, size, and other relevant information about the file.

[5] Please note that the dollar sign ($) only indicates a regular user under a GNU/Linux environment. A hash sign (#) would have indicated a root user terminal in the same environment.

2.2 Chaining Hashes (OPTIONAL) (0 points)

The idea of hash functions is to provide a unique hash sum for a unique input - this was something that two researchers started to elaborate on in the early 1990s, in order to timestamp digital documents [1]. Their idea was that given that a hashing algorithm always produces the same unique hash sum for each unique input, one could combine a digital document (e.g. "Yearly_Budget.pdf") with a date, hour, and second timestamp (e.g. "20170814180804") and calculate the hash sum of those together, in order to get a unique hash sum.

- What do you think about this way of time-stamping a digital document?

Now, choose another document and do the same thing with that, so that you have two hash sums from two documents that have been hashed with their respective timestamp. Next, calculate the hashsum for the hash sums of these two document to create a "root hash" (as illustrated in figure 1 below).

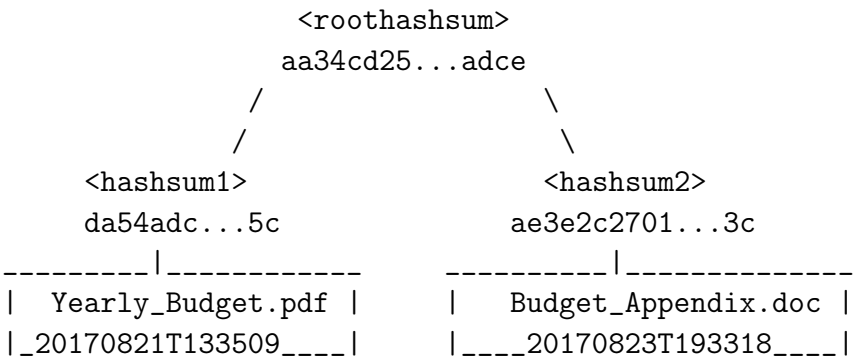

Fig.1 - An example of what a hash tree could look like.

- Explain in your own words what you have just done here. Also explain how this (think freely) possibly could be used for general purposes, but also specifically for forensic purposes.

Exercise II: File Headers (15 points)

You have been given a set of unidentified files that have been extracted from a disk image. These files are located under the Files folder in the Lab1 folder. You are called to identify their type by using different tools and cross-checking their results. Please investigate the files using at least three tools, where HexEdit[6] is one of them. You can find more information about file header and file footers on Gary Kessler's website[7]. Other tools that can be used are: Encase, FTK, TrID, file, or Exiftool. Feel free to use the Kali virtual machine in VirtualBox (or any other GNU/Linux or Unix system you fancy), where many of these tools are preinstalled. The lab files are available in the SharedFolder folder on the Desktop of the Kali virtual machine. Make sure that you document your results as well as the step-by-step methodology and include them in the report.

- Describe in detail which files you could identify and how you identified them.

[6] http://hexedit.com/downloads.aspx
[7] http://www.garykessler.net/library/file_sigs.html

Exercise III: Anti Files Forensics (15 points)

In the folder Anti-files, there are three files that have been extracted from a disk image. The other members of the forensic team have reported that these files seem either corrupted or malformed. You have been called to identify the "real" type of these files as well as extract their content. As a hint one of your colleagues has suggested you to employ file carving techniques in case that there are hidden file content or decryption techniques in case they seem encrypted. An interesting reference tool that can prove very useful is Malwaretracker[8] and Virustotal[9] - feel free to use any other tool or service that you fancy. On Windows, FTK and EnCase can carve files well. On Kali, Foremost and Scalpel are capable of file carving. Include answers to the following qustions in your report

- Could you identify the files? If so, what type of files were they?
- Could you find anything peculiar about the files?
- How did you proceed your examination and analysis of the files? Describe methodology and your results.

Exercise IV: Acquisition (25 points)

As taught in course literature and the lectures, an acquisition - a bit-by-bit copy of evidence is always done in order not contaminate the (original) evidence. In this exercise, you will learn how to make an acquisition of an evidence hard drive, not physically, but by acquiring a virtual machine hard drive. Be careful and do it in a forensically sound manner; minutely describe every step in the report. You can use any tool you would like for this (FTK Imager, dd, dc3, dfdd are a few examples. Do some research and use the one you find most suitable).

Include answers to the following questions (do not include the questions themselves) in the report (template B). Document your step-by-step methodology and result of this exercise. The idea with this exercise is to get you to 1) familiarise yourselves with the acquisition process and 2) to familiarise yourselves with the different tools that are available - and choose one that you find most fitting, and argue for your choice of tool (or tools if more than one). Acquire the file winxp.vdi in the folder "Acquisition" and include the answers to the questions below in your report (using report template B).

```
Filename: winxp.vdi
MD5 hash sum: c965a5e2236d60624c07c8233ed0aeb3
SHA1 hash sum: a8d7b2a8ebffc3905ab8b04edfe7e6fa92076fce
```

[8] https://www.malwaretracker.com/doc.php
[9] https://www.virustotal.com/

- Which program did you use to acquire the drive?
- Which file format was the acquired evidence files saved as?
- What was the hash checksums (SHA1 & MD5) of the result file?
- What are the SHA1 and MD5 hash sums of the file acquired from the mentioned.vdi file?
- What is the disk image file's acquisition date?
- What device (including CPU architecture) and what operating system were used for the acquisition?
- How many sectors and how many bytes per sector did the drive contain?
- Is the image file associated with a hash value (MD5/SHA1)? Can you use the tool to verify the integrity of the image file?
- Is there any other relevant information needed to include in the report? If so, include it.

Exercise V: Obtaining Access to Encrypted Files (15 points)

Encryption is one of the most evident challenges that digital forensic investigators face. Sometimes it is possible to obtain access to encrypted files - commonly known as "cracking", the files. The success of the decryption attempts depends on a number of factors. In this exercise you will learn more about those factors and some of the tools that can be used for this.

As you probably have learnt from your previous studies, there are different ways of cracking passwords and/or encryption keys: brute force cracking and dictionary attacks are the most common ones, and the ones that you will get to know more during this exercise. You will be given tools, dictionaries and files to crack - but the success of your result is also dependent on your creativity in using these tools and dictionaries, as you will learn.

In the folder "Cracking" you will find the following files:

- Unnamed 1.xls
- casssh.pdf
- Untitles.docx
- hr.gpg.tar
- wallet.dat

On the Desktop of the Windows machines you will find the password recovery tool PRTK. Open up PRTK. If it says that "Secure device was not found", make sure the IP address is set to 10.11.130.3 and the port is set to 6921. Once opened, click on File, then New case - create a case and click OK. Then click File, Add files and add the files you want to crack. If you want to use a dictionary attack, choose the dictionary file by clicking Tools and then Dictionary utility. You can find a dictionary file under Desktop\Shared Folder\Wordlist - please note that you might have to unzip that file before you can import it to PRTK.

If you want to use a more powerful computer for this, you log on to one of our password cracking servers. You can reach them by pressing the Windows key and the R key simultaneously and then type "mstsc" (without quotation marks). Then type pwd1.cslab.dsv.su.se (for groups 1-10) or pwd2.cs2lab.dsv.su.se (for groups 10-35) as the host and click connect. Log in with your username "groupXX", where XX is your group number and password "dsvcs".

Please note that you might not be able to crack all passwords due to time limitation but you must provide a forensically sound description of how you at least tried to crack all the files and how far you got.

2.3 Advice

In order to get you started with some of the above mentioned tools, we advise you to play around with the commands presented below, and of course to read the manuals and "man pages" of tools in question. Also, please note that password cracking might take some time, so a good idea is to let the password cracking run in the background while moving on to another exercise.

2.3.1 PRTK

AccessData's PRTK is a graphical user interface program that is easy to use, yet still configurable. You can add your dictionary source (it is predefined on the computers in the lab) by clicking File/Dictionary source. You can define rules for brute forcing under File/Rules. On the desktop of the Windows computers you will also find the a .pdf manual for PRTK, which you can use to fine tune your cracking process.

2.3.2 Hashcat

If you prefer wokring with open source tools that are command line based, a password recovery tool called hashcat is installed on the pwd1 and pwd2 computers in the lab. You can access them via RDP on pwd1.cs2lab.dsv.su.se and pwd2.cs2lab.dsv.su.se. Under

```
C:\Programs\hashcat\
```

you will find hashcat64.exe, which is a GPU (and CPU) cracking program that is capable of cracking many different types of files. Here are a few examples of how to get started with hashcat (in Windows environment):

```
> ./hascat64.exe --help /* Read the manual */

> ./hashcat64.exe -m 14400 wallet.dat C:\Users\cs2lab\Desktop\wordlist\unique.dict
/* Example dictionary attack with the dictionary file
unique.dict, cracking a Bitcoin wallet file */

> ./hashcat64.exe -m 14400 wallet.dat -b ?m?d?d
/* Example of brute force with digits and lower case letters */
```

Exercise VII: Steganography (15 points)

When doing steganalysis, you might encounter the difficulty of finding the right tool for your steganalysis; this is the tricky part with steganography: you need the right tool to be able to extract the hidden data[10]. There are a number of tools for hiding data in images, text files, video files, and other file formats. On the Kali machine, you will find three steganography tools in the Downloads directory. These are called Rizzy, Stepic, and pngcheck. Below you will can see how you can get started with each of these tools in order to find out if any steganography has been used on either or the files in the directory Lab1/Steganography.

```
# pngcheck -7cfpqstvx /media/sf_shared_folder/<file1>.png
/media/sf_shared_folder/<file2.png>
```

Or for Rizzy, this is how you get started:

```
# cd /root/Downloads/Rizzy-master
# python rizzy.py /* Click "Get Image", choose image and then click start */
```

Or for Stepic, this is how you get started:

```
# cd /root/Downloads/stepic-0.3
# stepic -d -i /media/sf_shared_folder/<steganography-file> -o <outputfile>
```

Describe which tools you tested, how you used them, and what your results were. Did you find any hidden message? If so, what was the message? Include it in the report. Please also answer the following questions in that case:

- Identify the two files; what do they depict? What are their hash sums? What are their META data?
- Which tool/tools did you use to find out if there was any message in the image/images?
- How was the message hidden in the image?
- Could you have found the message in any other way? Hint: Open both files named c2lab.png in two separate HexEditor windows and check if you can spot the differences and draw any conclusion from this.

[10] Or you need to be able to identify anomalities in the data of the file, where hidden messages can be found.

3 Resources

The following section presents sources that will be useful for succeeding with this lab assignment.

3.1 Windows Tools

- AccessData FTK
- AccessData PRTK
- AccessData Imager
- Guidance Software Encase
- TRiD
- hashdeep-4.4
- IrfanView

3.2 Linux Tools

- Foremost
- Scalpel
- Stepic
- Hexedit
- dd, dflcdd, dc3dd
- The Sleuth Kit (TSK)
- file, exiftool
- md5deep, sha1deep

3.3 Reading Material

- National Computer Forensics Institute - Network Intrusion Responder Program (NITRO), Volume 1 of 2: https://info.publicintelligence.net/NITROstudentV1.pdf

- National Computer Forensics Institute - Network Intrusion Responder Program (NITRO), Volume 2 of 2: https://info.publicintelligence.net/NITROstudentV2.pdf
- Forensics Wiki - http://forensicswiki.org
- E. Casey, *Handbook of Digital Forensics and Investigation*, Academic Press, 2010.

References

[1] Haber and Stornetta, "How to time-stamp a digital document," *Advances in Cryptology CRYPTO '90*, vol. 3, no. 2, pp. 437–455, 1991.