

# Reconstructing People's Lives: A Case Study in Teaching Forensic Computing\*

Felix C. Freiling    Thorsten Holz    Martin Mink

Laboratory for Dependable Distributed Systems

University of Mannheim, Germany

{freiling|holz|mink}@informatik.uni-mannheim.de

**Abstract:** In contrast to the USA and the UK, the academic field of forensic computing is still in its infancy in Germany. To foster the exchange of experiences, we report on lessons learnt in teaching two graduate level courses in forensic computing at a German university. The focus of the courses was to give a research-oriented introduction into the field. The first course, a regular lecture, was accompanied by two practical exercises: (1) a live-analysis of a compromised honeypot, and (2) a dead-analysis of a set of hard disks purchased on the web. The second course was a laboratory course with extensive experiments including forensic analysis of mobile phones. We give an overview over these courses and pay special attention to the reports resulting from the exercises which clearly document the ubiquity of data available to forensic analysis.

## 1 Introduction

Forensic science is the application of science to questions which are of interest to the legal system. In the standard definition [15], *forensic computing* is the gathering, interpretation and presentation of evidence found on computers. Obviously, methods in forensic computing are very dependent on the kind of legal system they work with, and since the legal systems are very diverse worldwide so are the methods and process models in forensic computing.

The field of forensic computing research has developed primarily in the anglo-american area. This observation is easily validated by looking at the academic forums that have the most reputation in the field [2, 6, 17, 18, 20]. We focus here on the German legal system in which handling of digital evidence has not reached the mainstream portfolio of investigators' skills. In academic education it is therefore difficult to give students firm guidance on how to process digital evidence in a way conforming to the legal system, to interact with the legal system and to testify as an expert witness in front of a court. It is therefore still a question of much dispute how to teach forensic computing in an academic environment like Germany.

In this paper, we report on lessons learnt in teaching graduate level courses in forensic

---

\*Appears in Proceedings of the 4th International Conference on IT Incident Management & IT Forensics (IMF 2008), September 23–25, 2008, Mannheim, Germany.

computing at the University of Mannheim, a German institution of higher education with approximately 11,000 students enrolled in five schools. The courses were given within the School of Mathematics and Computer Science. In writing this report, we wish to motivate other universities and instructors to enter a discussion and an exchange of experiences in this area.

## 1.1 Our Courses

To give an orientation, the courses were held in the spring terms of 2007 and 2008. The *first course* (given in 2007) was a lecture entirely devoted to forensic computing. It had a regular attendance of about 30 students. The students were mostly in their 4th year of studies and took the course as part of their Diploma degrees in computer science (“Informatik”) and business informatics (“Wirtschaftsinformatik”). The final exam was taken by 21 students. The course was meant to give a research-oriented introduction into the field and explicitly did not want to train participants to immediately become investigators. For example, the common commercial tools like EnCase [4] or FTK [1] were only mentioned briefly and practical lab experiences were mostly unsupervised. This contrasts our course with those offered in many specific Bachelor degree programmes on forensic computing (often called *computer* or *digital forensics*) or the professional training offered by the IACIS organization [5].

Two practical exercises accompanied the course: (1) a live-analysis of a compromised honeypot, and (2) a dead-analysis of a set of hard disks purchased on the web. In this specific iteration of the course we focussed much resources on the design and evaluation of the second exercise. Since there exists (to our knowledge) no generally accepted standard of writing reports about the results of a digital investigation, we also place particular emphasis on this subject here in this paper.

The *second course* (given in 2008) was a laboratory course on computer security in which students undertook team projects in specific fields. The motto of the course was that teams simulate the actions of a CERT (Computer Emergency Response Team). Following this idea, the projects encompassed analysis of malicious websites, binary analysis of malware, tracking of botnets and investigation of phishing incidents. About one third of the course was also devoted to forensic computing. In this part of the course, students had to perform a forensic analysis of two floppy discs, two hard disks and at least one mobile phone. The results of analyzing the discs were similar to those of the first course, so we do not report them in detail. However, our experiences in the analysis of the mobile phones were interesting as they were our first experiences in this field. Since we are not aware of similar experiences from other German universities, we devote some attention to the lessons learnt here too.

Thirteen students participated in the second course. Like in the first course, they all were registered as students for a Diploma or Bachelor degree in computer science or business informatics.

## 1.2 Other Courses and Related Work

The field of forensic computing has not yet reached the mainstream of academic teaching in Germany. This is evident from the small number of courses offered on this subject in Germany compared to the numbers in the anglo-american world. To our knowledge, the first course devoted entirely to forensic computing offered by a German university was given by Dornseif at RWTH Aachen University in the winter term 2004/2005 [16] (see also the discussion by Anderson et al. [9]). As some of us co-organized Dornseif's course, we drew strongly from this course when designing ours.

Starting in 2007, at several German universities instructors began lecturing in the field of forensic computing. Among these were Baumgartl at TU Chemnitz [10], Hahndel at FH Ingolstadt and Hammer at FH Offenburg [19]. While an increasing number of other universities is starting to discuss aspects of forensic computing within a general course on computer security, we are not aware of any other courses entirely devoted to the topic of forensic computing.

## 1.3 Paper Outline

The paper is structured as follows: We give more details about the motivation and the contents of the first course in Section 2. We then present the results of the dead analysis exercise from the first course in Section 3. After that we turn our attention to the second course and the results of the mobile phone investigation in Section 4. We summarize the lessons learnt in Section 5 and conclude in Section 6.

# 2 Outline of First Course

## 2.1 Definition of Forensic Computing

Our goal was to give a research-oriented introduction into the field of forensic computing to technically interested students. This meant to abstract mostly from concrete legal regulations and therefore to broaden the definition of forensic computing to a more computer science-like definition. We understand forensic computing not primarily as a tool for the legal system, but also as a tool for understanding security in general. Sound engineering principles dictate a thorough analysis of failures to learn the workings of a system and avoid subsequent failures of the same kind in the future. We define forensic computing therefore as *the discipline to reconstruct the events which lead to a security policy violation in an information system*. Thus forensic computing also includes the analysis of security incidents to learn the tools, tactics and techniques of the attackers and to gather facts needed to improve security in the future.

## 2.2 Course Content Overview

The course on computer forensics is taught with two lecture hours (in total 90 minutes) per week over a complete semester lasting 12 weeks. It was accompanied by three on-demand non-periodic meetings to hand out and explain the practical exercises and discuss problems that evolved during their pursuit. A final (half-day) meeting was used to present and discuss the results of the dead-analysis exercise that will be presented later in Section 3.

The goal of the course was to provide students with the necessary knowledge to understand digital evidence at a very deep level. A large part of the lecture consisted of deepening topics from operating systems and systems programming courses in areas of specific interest to forensics. For example, the course gave advanced background in computer networks, process management and filesystems with a strong bias towards the latter.

Based on the understanding of how relevant parts of information systems work, we aimed at teaching how to extract and interpret evidence from such systems and to evaluate the validity of the information gathered. The focus was specifically on a low-level, fundamental view on how the extraction of evidence from IT systems works, enabling the students to conduct forensic analysis without anything but the most basic tools. The idea was to follow the insight that investigators should not be restricted by their tools. Instead, investigators should be able to develop tools they need to support them in an optimal way. As mentioned above, commercial software like EnCase were only covered briefly in the lectures. We are convinced that the fundamental knowledge acquired during class enables students to quickly understand the commercial tools available on the market.

The twelve weeks roughly covered the following sequence of topics:

1. Course organization; overview: forensic science and digital evidence
2. Attack patterns and common computer crime; forensic mindset
3. Process models for forensic computing
4. Hard disk technology, imaging, integrity preservation
5. Disk volumes and disk partitions (DOS partition system)
6. File system analysis: Carrier's reference model for file system data [12]
7. File system analysis: FAT
8. File system analysis: NTFS
9. File system analysis: Ext2/3
10. Network and Internet forensics
11. Commercial tools and legal aspects
12. Theoretical forensic computing: Carrier's hypothesis-based approach [14]

Name	Affiliation	Topic of the talk
Steven Wood	Alste GmbH	Performing large digital investigations
Andreas Körner	PricewaterhouseCoopers	Forensics and white-collar crime
Andreas Schuster	Deutsche Telekom	Main memory analysis of Windows systems
Knut Eckstein	ESA	Advanced file system forensic analysis

Table 1: Talks accompanying the first course.

### 2.3 Invited Talks

The course was accompanied by a series of four invited talks by forensic computing practitioners from industry and law enforcement. The presenters and talk topics can be seen in Table 1. A fifth talk by a law enforcement practitioner from the German Federal Police (BKA) had to be cancelled due to health reasons.

### 2.4 Exercises

The exercises accompanying the lecture aimed at giving the students opportunity to gather experience with different forensic techniques themselves. We prepared two exercises:

1. Live-analysis of a compromised computer system.
2. Dead-analysis of a real hard disk.

#### 2.4.1 Exercise 1: Live Analysis

The material for the first exercise consisted of a paused VMware image of a compromised machine. The image was taken from material prepared by the HoneyNet Project as part of their forensic challenges [21]. Briefly spoken, honeypots are electronic bait, e.g., computers deployed to be probed, attacked, and compromised. Honeypots run special rootkit-like software which permanently collects data about the system and greatly aids in post-incident forensic analysis. Honeypots therefore not only provide a way to gather a large number of individually compromised machines over time, they also deliver a “true” story of how the system was compromised through their monitoring functionality. This is fortunate to instructors who have a better chance to grade investigation results of students.

More specifically, the image consisted of a Red Hat Linux 7.2 system compromised in August 2003 [22]. The skill level required for this challenge was estimated by the HoneyNet Project to be “intermediate to advanced”. The system contained many traces of compromise, beginning with the network interface being in promiscuous mode and ending with an installed rootkit plus a multitude of deleted files, some of them in Romanian. We considered this to be an ideal image for our course since the technical skills of the students were very diverse. So while some of the students only found little or unconvincing evidence that

the system had been compromised, other students were able to identify the attack vectors to an extent that superseded the “official” solution [13].

#### 2.4.2 Exercise 2: Dead Analysis

To make the second exercise as realistic as the first one, we had planned to have students image and analyze *real* hard disks. Since the focus of this exercise was not primarily on incident response but rather on data recovery analysis, we did not choose Honeypot hard disk images. Instead, we acquired a large amount (about 50) of pre-used hard disks through online auction platforms and asked the students to image and analyze these. Roughly half of the hard disks has IDE and the other half SCSI interfaces. The task for the students was to find out as much as possible about the former owner of the disk.

Because the acquisition process of the disks had taken place several years ago, many of the disks turned out to be broken. Only about 20 disks were readable and could be analyzed. Most of these disks had a capacity below 500 MB which made them slightly easier to handle than modern large volume disks. The disks also contained many different file systems, ranging from FAT16 over FAT32 and NTFS to Ext3. We asked the students to prepare a copy of the disk image on CD or DVD so that they can be re-used in the future even if the hard disk itself would stop functioning.

We set up one of the server rooms in our building which have strict access control checks to play the role of the room where a court or police keep court exhibits (see left side of Figure 1, the set of disks is shown on the right). To access a hard disk, people had to register formally with a lab member and the “evidence” was handed out. This was meant as a starting point for the documentation of the chain of custody. We prepared an investigation workstation in our lab which had many analysis tools pre-installed, but students could also perform the investigation at home.

As a result of this exercise, students had to write an investigation report. Following best-practices we recommended students to use the following structure when writing the report:

1. Formalities: name of investigator, date, reference/file number.
2. List of evidence to be investigated (serial number), documentation of chain of custody.
3. Task description (“find out as much as possible about former owner”).
4. Overview over evidence found.
5. Details of acquisition process of evidence.
6. Summary of used tools.
7. Summary of the implications of the evidence found.
8. Appendix: log files, screenshots, hand-written notes, etc.

Most students followed these suggestions. Details of these reports will be discussed in the following section.



Figure 1: Room keeping “court evidence” (left) and hard disks serving as “evidence” in second exercise (right).

### 3 Dead Analysis Results

In this section we describe in more detail the results of the second exercise in our course on forensic computing: the dead-analysis of hard disks purchased through a large auction platform on the web. Since we placed much resources on the design and evaluation of this exercise, we devote an entire section to it in this paper.

Table 2 gives an overview over all analyzed disks and the written reports. For sake of anonymization and brevity we assigned each disk a unique letter from the alphabet. In total, 14 disks were analyzed (letters A to N). The table shows the disk manufacturer and the size in MB. It also lists the number and page sizes of the reports written by students. Reports are identified by the letter of the disk plus a consecutive number. For example, report G2 is the second report written about evidence disk G. It can be seen that reports usually have a size of at most 20 pages but extreme cases with around 200 pages also exist (I1 and M1). The core part of these reports however was also at most 30 pages long. The remainder of the document consisted of log files, screenshots and recovered user files. Of all the reports submitted a selected number is presented here.

#	Manufacturer	Size (MB)	Reports and their size (in pages)
A	Western Digital	170	A1 (9), A2 (10), A3 (16), A4 (56), A5 (7)
B	Seagate	545	B1 (52)
C	Conner	412	C1 (13)
D	IBM	4330	D1 (19)
E	IBM	30700	E1 (14), E2 (13)
F	Conner	210	F1 (39), F2 (18)
G	Conner	420	G1 (65), G2 (48)
H	Seagate	545	H1 (14)
I	Western Digital	325	I1 (186)
J	Seagate	546	J1 (29)
K	Seagate	8400	K1 (15)
L	Fujitsu	1700	L1 (17)
M	Quantum	170	M1 (211)
N	Conner	406	N1 (13)

Table 2: Overview over analyzed hard disks.

### 3.1 Disk A, Report A1

Disk A contained a single FAT16 partition which appeared to be empty. Investigating the partition table with a disk editor, the student found traces of a prior Windows98 installation. Using the file carving tool foremost [3] the student tried to reconstruct data but the reconstruction failed. So the student reverted to the low-level Unix tool `strings` to identify readable characters on the raw hard disks. This resulted in recovery of a large number of text files containing invoices for special steel constructions manufactured by a well-known German steel processing company.

Particularly interesting in this report was the fact that the student always used two tools to recover and cross-check evidence found, whenever possible. For example, recovery and analysis of the partition table was performed using both `mmls` and `foremost`.

### 3.2 Disk A, Report A4

Report A4 also documents that the disk had been used by the same company as reported by A1. However, A4 used a demo-version of FTK [1] to investigate removed or overformatted files. FTK was able to recover many bitmap files from the Windows System folder as well as three Microsoft Excel files with calculations of the total work hours of 10 employees of the company in the year 1995. All evidence is documented with exact offsets on the disk where it was found.

Particularly interesting in report A4 was that the author did a partial web-search on the company and the names of the employees. The company itself had declared bankruptcy recently and now is owned by a Japanese technology company.

### **3.3 Disk B, Report B1**

Disk B was also empty, but turned out to have been reformatted before being sold. The student was able to recover a FAT16 partition and recover half a dozen MS Word and Excel files. Interestingly, they belonged to the same steel processing company which seems to have been the source of disk A. We assume that both disks were bought at the same time (as part of a larger “batch” of disks).

Report B1 is a typical example of a very technical approach. It contains a lot of numbers and hardcopy technical details from log files but hardly any explanation *why* the investigation was performed in the way it was done. However, the report contains a well-readable executive summary of the found evidence and the implications.

### **3.4 Disk F, Report F2**

The author of report F2 used a demo-version of FTK to analyze the disk. It also contained a FAT16 partition, but this time it had not been formatted before the disk had been sold. The installation appeared to be a Windows for Workgroups 3.11 based on MS-DOS 6.20. FTK was able to recover over 1600 files on the disk.

It turned out that this disk had belonged to a notary from Eastern Germany. The system installation information also gave hints on the remaining network infrastructure (network drives, attached printers, etc.) of the prior user.

The report was the only one that clearly indicated on the title page that the report contained confidential personal data. F2 also was unique in its attempt to guide the investigation through hypotheses. For every step in the investigation, F2 listed a couple of hypotheses that seemed possible based on the previously collected evidence. These hypotheses were then scrutinized in a systematic fashion, leading to new hypotheses etc.

### **3.5 Disk M, Report M1**

Report M1 was by far the longest and largest report submitted for this exercise. This seems only partly to be the result of the skill and determination of the student, because disk M contained an overwhelming wealth of information. The disk had been used in a computer of a small company selling and building kitchen interiors. The main user was a carpenter who also used the computer for private purposes. The student was able to recover more than 5000 files that seem to have been created and used between the years 1992 and 2000.

From these files it was possible to reconstruct an extensive personal profile of the former owner and his family. Figure 2, taken from report M1, for example shows the complete family tree of the former owner. The disk also contained (among other data) telephone numbers, social security and insurance numbers, bank account numbers, registration numbers of cars, and details about medical treatment for almost all (living) members of the

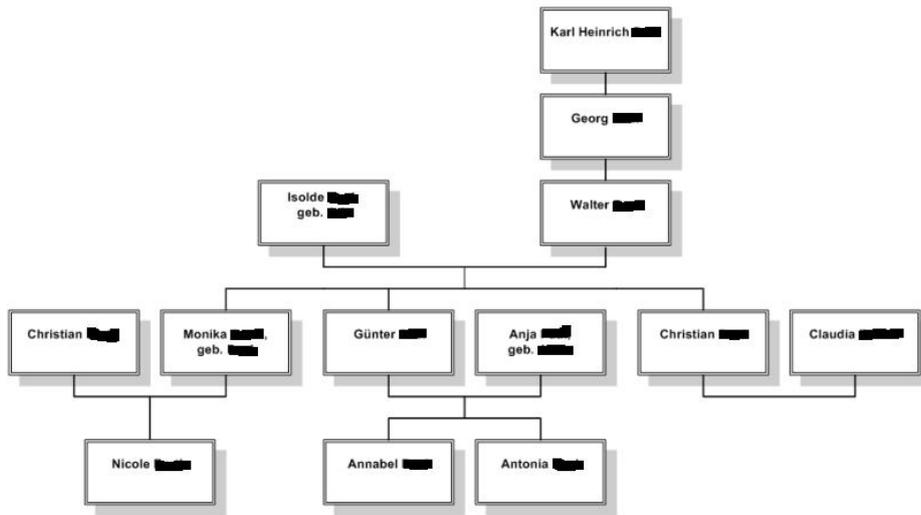


Figure 2: Reconstructed family tree taken from report M1.

family. In summary, it was possible to reconstruct the life of many individuals from this report.

### 3.6 Disk G, Report G1

Students sometimes make mistakes in data acquisition and analysis. For example, analysis of disk G showed traces of data from two totally different sources: relatively old data from a user called Mike D. from Bochum, and relatively new data from a student from the university of the authors. It seems that this disk has been acquired some time ago and was used in the meantime by a student for writing a project report for another professor at our university.

## 4 Second Course

We now report on our experiences with the second course, a laboratory course offered in the spring term of 2008.

## 4.1 Course Outline

The laboratory course was organized as a simulated CERT (Computer Emergency Response Team). A CERT is an organizational entity responsible for handling computer security incidents in the organization. Thirteen students formed 4 CERT teams (3 teams with 3 participants and one with 4). All had to investigate the same set of incidents and write up a final report.

The areas touched in the course were malware, botnets, phishing and forensic analysis. Students had to “apply for a position” within the CERT and were selected after a short interview. The course ran for a full semester (13 weeks) and consisted of a sequence of projects which were disguised as “real incidents”. While all projects had to be handled with a forensic mindset, only the final couple of projects was really concerned with forensic analysis. These projects consisted of a forensic analysis of two floppy disks, two hard disks and at least one mobile phone. The floppy disks were meant as an initial exercise and had been prepared from scratch by copying onto and deleting some files on them. The hard disks had been acquired from an administrative unit of University of Mannheim which regularly collects old and broken hardware and makes it available for other use within the university. The mobile phones were second hand phones and had been bought through a forum website on the Internet.

Since the results of the floppies and hard disks were similar to those described in Sect. 3, we focus on the results of the mobile phone analysis here.

## 4.2 Results of Mobile Phone Analysis

As confirmed by the practitioners who gave talks in the first course, mobile phones are one of the prime sources of digital evidence today. They contain (increasingly) large portions of flash memory that remain in the phone even if the SIM card is removed. The challenge with mobile phone analysis is not only handling the digital evidence from memory but also acquiring the evidence from within the phone.

To enable the acquisition of memory images, we bought a *twister box*, a commercial device usually used to install new firmware on the phone. It can also be used in read-only mode to access the memory chip in the system. A twister box usually comes with an impressive set of cables and adapters since every manufacturer has his own way of connecting peripherals to his phone.

In total, we had purchased 10 phones for a price of around 130 Euros; 7 of them were analyzed in the course. Since we wanted to be sure that common analysis software as well as our cables were compatible with the acquired phones, we restricted our purchase to popular models. The result was that all purchased phones were manufactured by Nokia: 6 model 3510i (see Fig. 3), two 6510, one 6030 and one 6800. Students prepared memory images using the twister box and then started data analysis on the image.

Since mobile phones usually use standard file systems on their flash memory, the challenge



Figure 3: Nokia 3510i analyzed by Team 1.

in data analysis lies rather in the proprietary file formats used by different manufacturers. Fortunately, the analyzed phones belonged to a popular and already rather old series of phones and therefore the data formats for short messages and other types of data was documented. Most students started data analysis using the same low-level tools used for dead-analysis of hard disks and then turned to the use of file carvers. File carvers allowed to reconstruct several graphics files, most of which were screen savers or logos of the mobile operators. Only one picture could be reproduced that seemed to stem from a mobile phone camera. Since none of the investigated phones had a built-in camera, it is unclear where this photo came from.

Since the open source tools did not help in analysing proprietary data, students turned to the use of commercial tools. Since we had no budget for these tools, they were restricted to evaluation versions. Independent research by all teams lead them to use one particular tool called *Cell Phone Analyzer* [11]. Using this tool it was very easy to recover the full phone book, the full call history as well as all active and all deleted short messages (SMS). Using this information it was relatively easy to identify the previous owner of the phone.

The evaluation version of *Cell Phone Analyzer* obfuscates the output of recovered SMS with dots instead of letters in random places of the SMS. The students quickly analyzed this behavior, verified that dots were placed in truly random places, and quickly wrote short scripts that merged the output from different invocations of the program. By this technique they were able to construct an (almost) perfectly readable transcript of SMS traffic.

A particularly interesting discovery was made by Team 2, that recovered a long SMS conversation (in German) discussing the pleasures of bodily union in some detail (see Fig. 4). Team 2 concluded that the former owner of the phone would not be amused seeing this information being made public. The team also discovered a series of SMS on that same phone, partly in a foreign language and dated older than the ones already mentioned. So it seems this phone had (at least) two former owners.

Würde dich niemals enttäuschen. Habe dich und möchte nichts anderes... Bis .  
 päter Schatz... Liebe dich..  
 Möchte am liebsten jeden Abend mit dir einschlafen. Nach hause ko.men und immer  
 wissen, du bist da oder hier leben wir gemeinsam!  
 Und, schon Männer kennen gelernt?  
 Das ist wirklich was besonderes. Habe noch nie so gefühlt für jemand, wie für  
 dich... Es ist einfach wahnsinnig schön...  
 Nein, so nie... Das mit uns ist einzigartig und werde ich nie mehr hergeben!  
 Ich Liebe dich auch... Kann garnicht in Worten ausdrücken wie sehr...  
 Wir auch... Ist alles nicht so schön ohne dich...  
 Nie wieder so lange getrennt... Wie lange macht ihr noch?  
 Danke auch so...  
 Warum jetzt schon? Was ist los? Hoffe du gehst alleine aufs Zimmer!  
 Ich hoffe .u betrügst mich niemals....  
 Ich verspreche es Dir.... Niemals...  
 Was hättest du nie gedacht?  
 Anruf-Info von Mailbox: [REDACTED] at k.ine N.c.rie... hinte.las...n [REDACTED].  
 Schon gut, hörte sich nur so an schatz... Noch 32 stunden... Dann kann ich  
 meinen lieben schatz endlich küssen  
 Und ich vermisse es wie du mich in den arm nimmst, mich liebevoll auf die stirn  
 küss., mich streichelst und deine hände mich überall .erühren....  
 Oh ja... Küss mich morgen bitte überall...  
 Als erster natürlich auf den mund, ganz lange und ganz innig... Danach den.hals  
 runter [REDACTED] und dann [REDACTED] zwischen [REDACTED]...  
 Oh, gan. ganz lange... Wie lange möchtest du mich küssen?  
 Weiß nicht, sehr lange... Wieviele [REDACTED] bekommst du denn innerhalb sagen  
 wir mal einer nacht?  
 Weiß nicht, w.rden wir sehen wenn wir es.probieren. Was meinst du wie oft du  
 kommst in fünf stunden?  
 Ich glaube drei stunden. Bin zweimal [REDACTED]... Ich stelle es mir schön vor so  
 lange mit dir schönen [REDACTED] zu haben... Gehen wir mal wieder spazieren? ;-)  
 Das werden wir schatz... Wie oft hast du [REDACTED]??  
 Einmal... Wann war das und wo?  
 Wo zu hause? Im bett oder auf dem sofa? [REDACTED]  
 mal... An was hast du gedacht?  
 Die nachricht kam nur halb an...  
 Auch an das gleiche... [REDACTED]  
 Wenn du mir sagst [REDACTED], das [REDACTED]... Laß dir was  
 einfallen... [REDACTED]?  
 Magst du das wort [REDACTED]? [REDACTED]?  
 Ist okay... Keine ahnung, du wirst die richtigen finden... Mag es wenn du mir  
 sagst wie [REDACTED]  
 Wenn du magst... Du muß anfangen...  
 Wenn das so ist... [REDACTED]  
 . @e@e tseichshst.  
 Wenn du [REDACTED]... Und ihn  
 [REDACTED]  
 Ja sehr sogar... Magst du es wenn [REDACTED]  
 wird?  
 Laß uns morgen all das machen was du [REDACTED]... Willst du  
 morgen [REDACTED]  
 Überall in der wohnung...oh ja... [REDACTED]... [REDACTED]...  
 [REDACTED]...

Figure 4: Excerpts of short messages recovered by Team 2, personal details and obscene language removed.

## 5 Lessons Learnt

We now discuss several lessons we learnt from our courses.

### 5.1 Tool Support

Following the bias towards low-level open source tools in both courses, most students started their analysis by using open source tools like the Sleuthkit [7] and foremost [3] in the beginning. In the first course, roughly 6 students used evaluation copies of the commercial software FTK [1] and X-Ways Forensics [8]. One student was able to use a licensed version of EnCase [4]. Regarding those disks for which multiple reports are available we did not see any fundamental difference in the depth and breadth of results achieved between open source and commercial tools. The main difference seems to have been that the analysis results were obtained *faster* using commercial tools, but no real data exists to measure the effort.

In the second course, no real open source tools exist to analyze memory images of mobile phones. After a first scan of the images using hex editors or command line tools like `strings` students reverted to evaluation copies of commercial tools. An interesting point here is that, although we did not give advice towards specific tools, all teams ended up using the same tool [11] because it was the only commercial tool offering a free evaluation version. Noteworthy is also that they were quickly able to circumvent the restrictions built into the evaluation copy by writing scripts to merge multiple analysis results and thereby mask out the random blinds. This is a clear indication to us that programming experience is a powerful skill to overcome restrictions of analysis tools (be they artificial or not).

### 5.2 Experiences with Documentation

In the first course, the recommended structure of the reports turned out to always lead to good results in documentation. Most students forgot to document the chain of custody in a rigorous fashion. Also, only half of the students documented sufficiently their analysis environment and argued that it was trustworthy.

Participants in the second course had partly followed the first course and so the quality of the reports in the second course was on average higher than in the first course.

From the level of detail of the personal data found, it also makes sense to add a general initial notice to the structure of the document that the document may contain sensitive personal data and that the report is not meant for public distribution. At the moment, we store all reports only in our lab's internal network. They are not accessible from outside.

Reports need to follow a quality control process too, i.e., they should contain a version number and a change log. The names of the authors responsible for individual parts of the document must also be clearly marked. Furthermore we will demand a short executive

summary “for the attorney” at the beginning.

In general we found that we need to motivate students more to focus on the prospective readers (CEOs, attorneys, forensic examiners). For example, the technical parts of the report must be written for an expert reader who is trying to validate the documented results. This helps to follow the report and allows to quickly reproduce the results from the same initial evidence. For this it is better, for example, to give exact disk sector numbers as reference for a particular piece of evidence on the disk rather than giving a screenshot of the evidence “on screen” during the investigation. Reproducibility is also improved by clear language. For example, instead of writing “the evidence disk is in a bad state”, an examiner should rather write: “the tool dd produced read errors in the following disk sectors”.

Finally, the report should follow standard academic practices, i.e., it should have a clear structure, clear statement and a set of bibliographic references if necessary. This is what students should learn anyway.

## **6 Conclusions and Open Questions**

The first course was evaluated at the end of the semester. Of roughly 30 students who regularly attended, 15 gave feedback via a questionnaire. The results were very encouraging: On a scale between 1 (best) and 6 (worst), the course scored a total of 1.27 (standard deviation 0.44). Regarding the specific question whether the course has improved their skills as computer scientists, the students unilaterally voted 1.0. So from this feedback it seems that students liked the course.

There are a couple of open research questions coming out of this work. First of all, there is the legal question whether we are allowed to perform forensic analysis on hard disks which have been intentionally erased. Who is the owner of the data found? Who are we allowed to pass it to? What steps should students take when they find personally identifiable information during analysis? To investigate this question, a bachelor thesis currently is conducted at our Lab.

Connected to this question is another one: How can we create a large set of hard disk images for analysis automatically without using actual personal data? This is a rather technical problem which we wish to tackle in the future.

### **Acknowledgments**

We wish to thank Maximillian Dornseif for helpful discussions.

### **References**

- [1] AccessData Corporation. <http://www.accessdata.com>.

- [2] Digital Forensics Research Workshop. Internet: <http://www.dfrws.org>, 2007.
- [3] foremost. <http://foremost.sourceforge.net/>.
- [4] Guidance Software, Inc. <http://www.guidancesoftware.com>.
- [5] IACIS Homepage. Internet: <http://www.cops.org>, 2007.
- [6] International Journal of Digital Evidence. Internet: <http://www.ijde.org>, 2007.
- [7] The Sleuthkit. <http://www.sleuthkit.org>.
- [8] X-Ways Software Technology AG. <http://www.x-ways.net>.
- [9] Philip Anderson, Maximillian Dornseif, Felix C. Freiling, Thorsten Holz, Alastair Irons, Christopher Laing, and Martin Mink. A comparative study of teaching forensics at a university degree level. In Oliver Göbel, Dirk Schadt, Sandra Frings, Hardo Hase, Detlef Günther, and Jens Nedron, editors, *IT-Incidents Management & IT-Forensics - IMF 2006, Conference Proceedings, October, 18th-19th, 2006, Stuttgart*, volume 97 of *LNI*, pages 116–127. GI, 2006. ISBN 978-3-88579-191-1.
- [10] Robert Baumgartl. Vorlesung Computerforensik. Internet: <http://rtg.informatik.tu-chemnitz.de/lehre/cf/cf.php>, 2007.
- [11] BKForensics. Cell Phone Analyzer by BKForensics. Internet: <http://www.bkforensics.com/CPA.html>, 2008.
- [12] Brian Carrier. *File System Forensic Analysis*. Addison-Wesley, 2005.
- [13] Brian Carrier. Honeynet project - scan of the month #29. Internet: <http://www.honeynet.org/scans/scan29/sol/carrier/index.html>, October 2003. Solution to Honeypot Challenge 29 [22].
- [14] Brian D. Carrier. *A Hypothesis-Based Approach to Digital Forensic Investigations*. PhD thesis, Purdue University, 2006. Available as CERIAS Technical Report 2006-06.
- [15] Eoghan Casey. *Digital Evidence and Computer Crime - 2nd Edition*. Academic Press, 2004.
- [16] Maximillian Dornseif. Lecture on computer and network forensics (course material). Internet: <http://md.hudora.de/presentations/forensics/>, 2005.
- [17] Elsevier. Digital Investigation. Internet: <http://www.elsevier.com/locate/diin>, 2007.
- [18] GI e.V., Fachgruppe SIDAR. IMF 2007 – Conference on IT Incident Management and IT Forensics. Internet: <http://www.gi-ev.de/fachbereiche/sicherheit/fg/sidar/imf/imf2007/>, 2007.

- [19] Daniel Hammer. Computer Forensik. Internet: <http://security.fh-offenburg.de/forensics.php>, 2008. Course homepage at FH Offenburg, in German.
- [20] IEEE. IEEE Transactions on Information Forensics and Security. Internet: <http://www.ieee.org/organizations/society/sp/tifs.html>, 2007.
- [21] The HoneyNet Project. HoneyNet challenges. Internet: <http://www.honeynet.org/misc/chall.html>, 2008.
- [22] The HoneyNet Project. Scan of the month: Scan 29. Internet: <http://www.honeynet.org/scans/scan29/>, September 2003.