Journal of Information Technology Education: Innovations in Practice **Volume 10, 2011**

Using Digital Logs to Reduce Academic Misdemeanour by Students in Digital Forensic Assessments

Harjinder Singh Lallie
University of Warwick, International Digital Laboratory, Coventry, UK
h.s.lallie@warwick.ac.uk

Philip Lawson
University of Derby, School of Computing and Mathematics, Derby, UK
p.lawson@derby.ac.uk

David J. Day
Sheffield Hallam University, Sheffield, UK
d.day@shu.ac.uk

Executive Summary

Identifying academic misdemeanours and actual applied effort in student assessments involving practical work can be problematic. For instance, it can be difficult to assess the actual effort that a student applied, the sequence and method applied, and whether there was any form of collusion or collaboration. In this paper we propose a system of using digital logs generated by selected software tools (such as FTK- Forensic Toolkit and EnCase) for the purpose of identifying the effort and sequence of events that students followed to complete their learning activities (say, arriving at conclusions relating to an assessment question) and, thereby, determining whether it is likely that an academic misdemeanour may have occurred. The paper elaborates on an assessment exercise conducted with a cohort of 67 students in a specific class of disciplinary learning, highlighting the process that students have to follow, and then proceeds to show in some details how selected logging facilities can be used to provide evidence that students may have committed an academic misdemeanour.

Keywords: Digital investigation, plagiarism, academic collusion

Introduction

In some practical assessment exercises it can be difficult to determine the effort that a student applied in completing his or her work; in particular, programming exercises and digital forensic investigations are two such examples. In the context of digital investigation, software tools such as FTK (Forensic Toolkit) (AccessData, 2011), EnCase (Guidance Software, 2011) and Autopsy Sleuthkit (Carrier, 2003) can be used to assess students' practical understanding of such knowledge as:

- Digital investigative process
- Tools involved in an investigation
- Planning stage of an investigation

Editor: Kam Vat

These tools provide investigators with low-level access to the operating system file system and the inner structure of some storage facilities, making possible the analysis of the structure and content of the underlying files, even if they have been erased or corrupted either purposefully or unintentionally. Some of these tools have been used by law enforcement agencies around the world and also by education institutes to teach file system analysis and digital investigation (AccessData, 2011; Guidance Software, 2011). It should be noted that of these tools, Autopsy Sleuthkit is the only *open-source* tool and is therefore effectively free for academic use.

Where students have been required to produce a report of a practical exercise, it can be difficult to assess whether the report is indeed the student's own work or the result of collusion, collaboration, or plagiarism (collectively known herein as academic misdemeanour). This is because the answers produced by most students to questions posed within the assessment task will tend to be similar. If the reports of such digital investigations were subjected to electronic anti-plagiarism software such as Turnitin, it is likely that a high degree of similarity could appear. This does not therefore provide a true account of the student's effort and efforts could be made by the assessor to reduce the likelihood of students committing an academic misdemeanour or at least being able to detect it more easily should the student commit an academic misdemeanour .

In the assessment exercises to measure and collect student evidence in learning programming and database development tasks, the actual answers are often not as important as the process and the true effort that students employed in producing their results. Throughout this paper we use the example of digital investigation as the context of our discussion.

We propose that more effective techniques are needed to identify student efforts in higher education learning scenarios, or at least for making it easier for the assessor to determine whether a misdemeanour may have occurred. In the context of digital investigation assessments, we propose that five particular techniques can be useful in this context:

- Focus on process rather than result
- Require the development of contemporaneous logs
- Require Demonstrations
- Unique Digital Forensic Images
- Bank of Assessment questions

Process Focused

The idea is to focus on the evidence rather than the answers: students are required to articulate precisely how they have discovered the answer. One problem with this approach is that answers can be reverse engineered, and it is easy to generate a discussion of how and where the evidence was found. Furthermore, for a given question in a case study, the method of finding the evidence will generally be one of a few options (as we shall show later in the "Results" section).

Contemporaneous Logs

Contemporaneous logs (also referred to as contemporaneous notes) are used to record the process of investigations in the law enforcement/legal domain (State of New Jersey, 2011; Your Rights, 2008) as well as in other domains (East Sussex County Council, 2011; NHS Scotland, 2011; The University Workplace, 2009). In an assessment exercise such as the one under present consideration, the student can be required to develop contemporaneous paper-based logs of the exercise in which they are required to record every event that they followed (searches, file analysis, etc.) cross referenced with a date/time for the event to demonstrate the efforts and sequence of events they followed to find particular evidence. In other words, it is important to require the students to take detailed notes throughout the exercise outlining the searches they performed (with journaling explanations) and summarise their results thereof with some indication of what they have book-

marked with reasons. The problem with this approach is that the assessor can never be sure that the notes are indeed contemporaneous and a precise record of the work done - it is possible to *doctor* these. This can be mitigated somewhat by requiring for the contemporaneous logs to be signed off on a weekly (per tutorial) basis. Similarly students could be expected to insert comments in the software at particular points in the investigation which are subsequently included in the delivered report – similar to the comments facility in documenting program design in programming. However, here again, these can be engineered and the inclusion of a comment or remark within the exercise doesn't demonstrate knowledge and effort.

Require Demonstrations

The assessor can call for individual demonstrations wherein students show how they found the evidence relating to particular (randomly selected) questions in the investigation. The problem with this approach is that student responses can be rehearsed; while this may demonstrate that the student understands how to get to that evidence, it does not demonstrate the effort that the student originally applied in seeking the answers for him or herself. Furthermore, individual demonstrations can be quite time consuming.

Unique Digital Forensic Images

The idea is to develop unique digital forensic images for each piece of student work. The problems associated with generating digital forensic images are that the process requires a lot of time and the task of generating realistic images is problematic as indicated in Lallie (2010). Furthermore, the problem is exacerbated in a large crowd of cohorts. While solutions have been proposed (Jones, 2010) for generating multiple unique images for a given technical scenario, these solutions are not generally available to the academic community.

Bank of Assessment Questions

The idea is to generate different questions for each student or a bank of questions from which a separate set is selected for groups of students. For this to work, the case study and the digital forensic image must have sufficient scope and breadth, in other words, the digital forensic image that is generated must be complex enough so as to allow an assessor to be able to set multiple tasks. This solution may in fact be impossible for large cohorts, such as the one experienced in this study, because larger cohorts would require a much larger bank of questions

Of course – a combination of any of the above could be used; however this may make the assessment process more complex and time consuming. In this paper, we propose a technique that allows a more precise analysis of students' effort in conducting assessment for student work, thereby making the detection of academic misdemeanour easier and more in line with an evidence based approach to detecting academic misdemeanours. Using this technique the assessor is able to identify the sequence of events that a student followed in order to arrive at a given conclusion and is able to tell whether a student went directly to a complex answer (suggesting a misdemeanour) or whether a particular logical sequence was followed in order to achieve the same. Where the student submissions give rise to suspicion, the proposed system can be combined with face-to-face demonstration to allow the student to walk through their findings.

The remainder of this paper is structured as follows. In the next section we explore the background regarding available digital forensic images and introduce the nuances of the Greg Schardt (CFReDS (NIST), 2007) image which make it a useful digital forensic image for academic purposes. We proceed in the same section to present the methodology we used in the study and, in particular, highlight the resource requirements for such an exercise and also the instructions students have to follow in order to complete their tasks. We proceed in the section "Digital Logs" to

analyze the format and content of the log file in FTK 1.81 (AccessData, 2011) – particularly highlighting how this file could be affected by actions students take under the working of this software. Within the same section we also consider the logs contained under other popular digital investigation tools (such as EnCase and Autopsy Sleuthkit). We proceed thereafter to discuss the data mining exercise that we performed in order to reduce the resulting logs to a more manageable and meaningful size. In the "Results" section we present the results of our teaching and learning experiments and analyze particular students' log entries that raised concerns about their performance.

Using Logs to Highlight Academic Misdemeanour

To begin with, it would be useful to introduce our audience to some digital investigative concepts to be referred to throughout this paper. Digital investigation involves the investigation of data storage devices for the existence of evidence which can support a particular hypothesis relating to the incident being investigated (B. D. Carrier, 2007). Such investigations must conform to a process of accepted legal practice wherein the investigator must ensure that the investigation is conducted in a manner such that the results can be accepted by the judicial authorities, and in so doing must ensure that no changes are made to the original data storage system that is being investigated. Digital investigation tools allow a user to make bit-for-bit exact copy of a storage system and then store it as *digital image* (which we shall refer to as digital forensic images – DFI).

The investigation is conducted on the DFI and rarely, if ever, on the original storage system itself. This ensures that the original hard disk is not tampered with during the course of the investigation. In this case study, students were given access to a previously created DFI, i.e., they did not have to create a DFI. The DFI was portable and could be stored on a USB storage system.

DFIs for use in academia can be generally classified into two categories: specific skill DFIs and holistic skill DFIs. Specific skill DFIs can be used to teach and/or to assess very specific and particular concepts such as file system analysis, partition analysis, meta-data analysis or e-mail investigation. Holistic skill DFIs are designed to teach and/or to assess a collective range of specific skills including the overall investigation of a case possibly including the production of a final case report (Lallie, 2010).

A number of DFIs are available that can be used for investigation by students in tutorials or formal assessments, some of these have been developed by the National Institute of Standards and Technology (NIST), (CFReDS (NIST), 2007), Mueller (2010) and Digital Corpora (2011). This study is based on the *Greg Schardt* DFI published by NIST as part of the CFReDS data sets. CFReDS are a series of DFIs that can be used for the training and teaching of digital investigation. The *Greg Schardt* image is a holistic skill DFI involving an individual accused of hacking. A number of model questions have been posted by CFReDS; however, the answers to these questions are also posted. These answers are available through a password which can be accessed very easily (there are no security checks).

The Greg Schardt DFI is a very adequate resource which can be used in academia to test the learning of undergraduate students. This is because the solutions are available, the image is freely available, and it contains sufficient depth for the assessor to define further questions which can be used to lower the risk of students committing potential academic misdemeanours. The availability of solutions should not raise a concern because a digital investigation exercise should be focussed on process and not the answers.

The Cohort

The cohort of students to which this paper relates were studying the *Digital Forensics* module (the equivalent of a course in the US) and were registered in three degree programmes: BSc

Computer Forensics and Security (47 students), BSc Information Technology (17 students) and BSc Mathematics with Computing (7 students). The cohort consisted of 44 male students and 23 female students. Each of the degrees consists of 8 x 15 credit modules of which there is a combination of core and elective (optional) modules. The *Digital Forensics* module is a core module for the Computer Forensics and Security programme and an elective on the other two. None of the students had any prior curricular experience of digital forensics as a discipline, and for students who studied the Computer Forensics and Security degree; this was the first of 5 related modules in the specific discipline of digital forensic investigation.

Digital Forensics is an introductory year 1 module that introduces students to the core concepts of conducting investigations on digital storage systems. The core content of this module was covered in 12 x 1 hour lectures and includes topics such as incident management and response, the ACPO (Association of Chief Police Officers) guidelines, crime scene management, the handling, management and reporting of evidence, storage system structures, acquiring data from popular operating systems such as Windows, UNIX and LINUX, obtaining system logs and other important configuration files, and recovering allocated and unallocated data.

Following the lecture, the cohort of students was split into groups of 18 for the 12 x 3 hour laboratory sessions, and each laboratory session was dedicated to a practical investigation of a hard disk using FTK 1.81. At the time of writing, FTK is at version 3.3; however the version used within this case study was version 1.81. The module was delivered in the second semester of year 1 of the respective programmes which meant that students had already received exposure to fundamental IT concepts such as programming, databases, and networks management. Three academic tutors/assessors were involved in the delivery of the module although only one of them led all the lectures.

The intended learning outcomes (ILOs) of this module include the following:

- Demonstrate an understanding of the core concepts relating to the digital forensics and investigation of computer systems. This was assessed through a formal examination.
- Formulate and present technical arguments pertaining to the digital forensics investigation of a computer system in a coherent and clear manner. This was assessed through the practical coursework to which this paper relates.

Formative assessment takes place throughout this module in two ways: by way of online tests (in our case using the *blackboard* system (Blackboard, 2011)) linked to particular hard disks that have been investigated by students throughout the tutorials and also through feedback provided by tutors within the assessment.

Lab Equipment and Resource Requirements

Encase and FTK are the two of the most popular software tools used by law enforcement agencies around the world (AccessData, 2011; Guidance Software, 2011). The University supports both software; however, FTK 1.81 is less 'resource hungry,' i.e., it requires less RAM and a smaller hard disk space on which to operate. Furthermore, FTK is more established than subsequent versions, and it also provides a more easily manageable logging facility. By a 'manageable logging facility' we mean that:

- Certain logging items can be turned on/off so that the resulting log file sizes are manageable. By manageable we mean that if particular options are not turned off, the resulting log file can be hundreds of pages in length.
- The application produces a log in a file format (for example text format) which can be analysed independently of the application system. This feature was important so as to allow us to data mine the log files.

Encase does not provide such a facility and the logging facilities within FTK 3.3 were much more difficult to manage – particularly because the logs could not be analyzed independently, therefore version 1.81 was selected as the vehicle for this study.

The minimum 'recommended' requirements to run FTK 1.81 are Windows XP/Windows Server 2000/Windows Server 2003; Intel Core 2 Duo, or equivalent processor; 2 GB RAM; 4 GB for program files and XGA (1024 x 768) or higher resolution. In our case, the students completed the tutorials/assessment in one of two laboratories. The computer machines in each of these laboratories had the following specification: Windows 7 (64 bit); Quad core Intel CPU Q9300; 8GB RAM; 1TB hard drive and an NVidia 9800 graphics card. Collectively this was far better than the recommended specification. The computer systems do not have to be networked as the software can run on stand-alone unconnected systems. In addition to the FTK software, the laboratory machines also had Encase 6.16.2 (64bit) & Oxygen 2011 (a mobile phone investigation software).

Given the portability of the DFI, students could effectively work on any machine in the laboratory as long as they saved their case on their external USB storage or on a network storage system. AccessData (vendor of FTK) recommends that an additional storage space of 75% in comparison with the size of the DFI is available on the storage system to which the case will be saved. So for instance if the DFI (including the evidence files) is 8GB, an additional spare 6GB is required on the storage system (AccessData, 2011).

Student Task

The student assessment consisted of three particular elements: the DFI, the case study (explaining the background to the case, and rationale for investigation), and the specific questions that the student had to answer. The questions in this case were largely based on the questions posed by CFReDS – although a large number of additional questions were added by the assessors. The questions were given to the students one week before the digital forensic image was presented – so as to help them begin the 'preparation' to the investigation. The first week of the assessment (guided through tutorials) was to be spent in pre-planning for the searches that they would need to conduct once they received the DFI.

The assessment consisted of two deliverables:

- A report in which students were to answer the questions very specifically whilst highlighting where and how they found the required evidence, this was not the FTK report
- The FTK generated case log

For the student to begin the actual investigation, they must:

- Create a 'case' using the software. This involves assigning certain details to the case such as investigator name, case name, case number, and other miscellaneous notes (all of which can be added/modified subsequently)
- Following this, the evidence files (DFI) must be added to this case

Once this is done students are able to investigate the evidence. The actual investigation involves a number of steps which are beyond the scope of this paper but, in brief, such details include the following:

- **Direct file system analysis** wherein students can visually explore the file system, individual files, and contents of files, locate existence of deleted files (including files that have been removed from the recycle bin or corrupted)
- **Keyword searching**. This is by far the more efficient method of investigation and involves developing search terms (based on what the student knows about the background of the case) and then running these searches on the DFI. Digital investigation software

incorporates powerful GREP/mask based search facilities to allow for the searching of 'complex' terms such as zip codes, credit card numbers, and telephone numbers.

- **Bookmarking**. This is the process of highlighting important evidence and bookmarking it using the inbuilt facility so that it can be retrieved directly in future – thus avoiding having to conduct another search to achieve the same.

For further details regarding the digital investigative process, the reader is referred to Baryamureeba and Tushabe (2004) or Carrier (B. D. Carrier, 2007).

A properly organized investigation (in the context of a particular assessment) would demonstrate that the student had understood the background of the case and had conducted sufficient research so as to plan the search terms and thereby make the investigation more efficient.

Students were advised that following the assessment of the report and consideration of the log files, selected students may be called in to provide case demonstrations of their findings if the assessor felt that this was necessary (this is in fact standard practice). Four students were subsequently invited to provide demonstrations. During this demonstration, the assessor asked a series of questions which were selected randomly from the original pool of questions in the case study. The student was required to clearly demonstrate how he or she found the evidence.

Digital Logs

Digital logs are used in a variety of scenarios, most particularly in operating systems (Schuster, 2007) and in network devices such as intrusion detection systems and firewalls. Kenneally (2004) refers to digital logs as a *digital eyewitness* that can replicate the perception, memory, and cognition which are used in eyewitness events. However for a digital log to be of use in a given context, it has to conform to particular requirements. Essentially the logs have to be, amongst other things:

a. **Trustworthy.** The application that produced them has produced a correct and accurate record; namely, the application 'works'

b. **Authentic.** Shown not to be tampered with

c. **An accurate reflection.** A rule that digital forensics takes further by requiring it to be identical

d. **Self-authenticating.** For instance, we can trace the log activity back to the specific device that generated it.

Most investigative tools provide an option to generate a log of investigator activity. The information contained within the FTK 1.81 log includes details of evidence files added to the case (with the parameters/options that the investigator selected), searches performed, and bookmarks recorded - all supported with dates and times. The assessor can then scrutinize/analyze this data and build a picture of the process that the student followed in performing the investigation. While there seem to be no studies that have 'tested' the FTK 1.81 log against Kenneally's criteria, we can assert that the FTK 1.81 log does not satisfy criteria (b) which we shall explore later in the "Results" section. The FTK log is not intended to be a system log in the same spirit as logs in IDS (intrusion detection systems) or firewall systems. For instance, an IDS log would record very specific system events and generally not have the facility to record details of every single transaction that takes place. The FTK log on the other hand does have the function to record details in such depth.

In our investigation, the case log is used as a means of assessing student efforts; however, the understanding of case logs can itself be quite useful from a pedagogic point of view for students. Consider the following example: A case investigator (in this case the academic/assessor) investi-

gates an image and answers specific questions throughout the investigation. Yet, some of the answers are found to be incorrect - deliberately or accidentally. The investigator has inserted comments (using the *insert comments* feature) some of which are indicative of what an investigator would normally do, whereas others are red herrings - perhaps indicating a conspiracy.

Students are asked to analyze the case logs and the digital forensic image itself to highlight the mistakes made by the original case investigator and to identify the correct courses of action that should have been taken at those points. This activity would give students a better understanding of the digital investigative process, of the importance of planning, and also of logging systems in IT (information technology).

The FTK Log

The FTK case log is configured when a new case is created and can be deactivated at any point. The log is a text file which holds a record of each *event* that the application performs. With the log being a text file, this means it does not comply with the authenticity requirements stated by Kenneally (2004). The issue of trustworthiness and authenticity of digital logs can normally be solved relatively easily by encrypting the logs as proposed by Waters, Balfanz, Durfee, and Smetters (2004); however, in this case it was probably never the intention of AccessData to use the log as an audit log as such.

The FTK log records two types of event: user driven and system driven. User driven events include the following:

- Management of evidence files (digital forensic images)
- Error generation
- Bookmarking (Figure 1)
- Searching - including the resulting hits of those searches (Figure 5)
- Data carving and Internet searching. For instance Searches of Internet keywords
- File management (copying, viewing, etc.)
- Altering visual presentation of evidence (Figure 3)

System driven events include, for instance, points at which FTK processes an evidence file (**Figure 2**). For both event types, FTK also logs the result of that event (for instance the results of searches).

31/03/2011 13:15:16 -- File Properties viewed for: 4Dell Latitude CPi\Part_1\NONAME-NTFS\My Documents\FOOTPRINTING\NT\Nmapnt\DRIVERS

31/03/2011 13:15:24 -- File Properties viewed for: 4Dell Latitude CPi\Part_1\NONAME-NTFS\My Documents\FOOTPRINTING\NT\Nmapnt\DRIVERS\Packet2K

31/03/2011 13:15:52 -- Added bookmark: "Question 3.M"

Bookmark comment:

Include in report: Yes

Export files: No

Remember file position/selection: No

Added file: 4Dell Latitude CPi\Part_1\NONAME-NTFS\My Documents\FOOTPRINTING\NT\Nmapnt\DRIVERS

Figure 1. Bookmarking

```
31/03/2011 15:21:54 -- Initializing thumbnail view
31/03/2011 15:21:54 -- Resetting search terms list
31/03/2011 15:21:54 -- Building the indexed search results tree...
31/03/2011 15:21:54 -- Building the live search results tree...
31/03/2011 15:21:54 -- Building the bookmark tree
```

Figure 2. FTK System driven events

```
01/04/2011 14:34:31 -- Column settings changed to:  Default File List Column Setting
01/04/2011 14:34:33 -- Column settings changed to:  All Columns
```

Figure 3. User driven column change event

Throughout the forensic case management, the user has the option of adding specific manual entries (comments) into the case log. This function can be used by students to explain particular searches or to highlight problems they encounter in finding particular evidence.

Tellingly, all events in the log are time-stamped; this allows the assessor to determine the time scope of the assessment, such as when the student began the case and how the student spent time on the case. This can give a better insight as to how the students balanced their time on the assessment – for instance whether they spent the majority of the time on the work at the beginning of the time period (front loaded) or towards the end (end loaded). It is possible to determine somewhat the total time spent on the case through analyzing all the timestamps.

While we noted that flexible management of FTK 1.81 logs made this version of the software a suitable choice for our experiments, there is a downside to this: namely, the logs can be forged. It is conceivable that a student could forge the timestamps so as to suggest that more work was spent on the task than it actually was. While recognizing this problem, we did not see this as an important issue because the time and efforts required to modify a whole sequence of timestamps would far outweigh the benefit of showing how the time was spent on the task.

Logging in Other Digital Investigation Tools

FTK 3 has similar logging facilities which differ particularly in the way logs are accessed. A series of log files are generated for each case, one such file of importance is the *JobInformation.log* file, the output of which is shown in **Figure 4**.

EnCase 6 does not incorporate a similar logging functionality – better functionality is however included in the EnCase SAFE module (Guidance Software, 2010). *EnCase 7* incorporates something similar in the guise of an *evidence processor log*. *Autopsy Sleuthkit* (V1.70), also incorporates a reasonably complex logging facility (B. Carrier, 2003) that stores details of particular user activity in a series of log files which include:

- **Autopsy.log** which contains details of when Autopsy was started/stopped
- **Case.log** when the case was created, opened
- **Host.log** details of when images are created, opened
- An investigator log file which is perhaps more important in this context. This file has the following filename format: <investigator_name.notes> and contains details such as directory listings, files viewed, keyword searches performed, creation of timelines etc.

```
2011-06-19 21:17:12 [14] INFO  -
        Processing Manager Version: 3.2.0.32223
        Starting Job: 99c66fb28b9b41519990048e9ff19707
        Started by Phil-MSI\Phil on PHIL-MSI
        Processing manager: PHIL-MSI
        FTK Username: Phil
        CaseID: 1020
        CasePath: C:\Users\Phil\Forensics\FTK\Cases\MyWorkPenDrive\MyWorkPenDrive
        JobType: LiveSearch
        Database: PHIL-MSI:1521 - FTK2
        Rangeset: <objids><range start="1000" stop="11861"/><range start="12000" stop="17769"/></objids>

2011-06-19 21:17:17 [PEEventQueueThread] INFO  - 99c66fb28b9b41519990048e9ff19707 - Using engine localhost
2011-06-19 21:17:17 [PEEventQueueThread] INFO  - 99c66fb28b9b41519990048e9ff19707 - Processing started
2011-06-19 22:04:26 [PEEventQueueThread] INFO  - 99c66fb28b9b41519990048e9ff19707 - Processing finished
2011-06-19 22:04:26 [PEEventQueueThread] INFO  - 99c66fb28b9b41519990048e9ff19707 - Job finished
2011-06-19 22:04:27 [PEEventQueueThread] INFO  -
99c66fb28b9b41519990048e9ff19707 - Job finished with state Finished
Job had 0 failed items
Searched: 16630
Total wall clock time: 0 days 00 hours 47 minutes 14 seconds
Total post-processing time: 0 days 00 hours 47 minutes 14 seconds
```

Figure 4. Sample log from *JobInformation.Log* in FTK3

Data-mining the Log Files

FTK log files can be very large, and in this case they varied considerably in size from 19 pages to 448 pages. This is because the log file contains details of each and every event with a corresponding *<drive>:\filepath\filename* listing. A manual analysis of this would have been difficult and time consuming; therefore, two macros were created and applied to each student log separately:

- One to highlight important user events such as searches (normal and live searches) and bookmarking.
- A second which reduced the log down to important timestamps that indicated the amount of time a student had spent on the work as well as the pattern of their activity.

The *summarised logs* were saved as separate files and then analyzed. Arasteh, Debbabi, Sakha, and Saleh (2007) have proposed that there is a lot more work to be done in the area of automated log analysis and propose a system of formally reducing and representing events in a log. Murphy (2007) shows that log files can often be very complex and can require a lot of skills to analyze. Our own experience supports this view and has shown that, in hindsight, this operation may have been more easily performed with each file being processed by a program at receipt rather than having to manually process each file in this way.

Results

Two of the skills assessed in this assessment were the ability of the student to plan their investigation and their understanding of the investigation process. There is a particular process that stu-

dents should follow in order to find the answers to the questions posed in the case study. This process varies depending on the question being asked; however, the student must identify where the particular item of evidence is likely to be found. Many evidence items, for instance, can be found within particular operating system *registry keys*. Students in this case need to understand the registry and be able to search directly within the registry system.

Following the investigation process highlighted previously, we were able to identify a number of student submissions which warranted further investigation. In such cases, while most of the answers had been found, the students did not follow the process we would have expected. Answers were sometimes found through a very small number of searches, some of which were committed haphazardly and in no particular logical sequence. We could not determine within this experiment as to from where the students may have acquired the correct results – for instance it may be that students approached previous cohorts who had completed a similar assessment in a previous year.

We give three examples herein, two of which were potential academic misdemeanours, the third demonstrates poor planning.

Consider the following question: "**What are the IP address and MAC address of the computer?**" The correct answers to this question in this example are "**192.168.1.111**" and "**0010a4933e09**" respectively and the answers could be found in one of two ways:

- Search through the appropriate registry keys which point specifically to the IP address/MAC address of the machine
- Build a 'masked'/GREP search which would return all the IP addresses or MAC addresses recorded on the system, the student should then be able to deduce the correct IP/MAC address.

However in certain cases, students were seen to be searching directly for the 'correct' IP Address (see **Figure 5**). In this case, it is likely that the student was told the correct IP address resulting in the direct search as shown in the figure. It was also shown in this particular submission that the student did subsequently bookmark the correct registry entry which pointed to the IP address (after analysing the 68 hits).

28/03/2011 11:53:26 -- Performed Search Query: (192.168.1.111)

Search Results: 68 Hits in 8 Files

Search Query: (192.168.1.111)

Search Results: 68 Hits in 8 Files

28/03/2011 11:53:30 -- Displayed Search Results for QUERY: (192.168.1.111)

Figure 5. A direct search for the IP Address 196.168.1.111

In a second question students were asked to "**List the network cards used by this computer**". The correct answer was: "**Xircom Card Bus Ethernet 100+**" and "**Modem 56**" and the answer could have been found in a number of ways:

- An analysis of the appropriate keys in the registry through a search for those keys rather than the 'contents' of those keys
- An analysis of configuration files generated by the installation of the network cards. This is less likely unless the student already knows the make and model of the network card. This method should follow the above wherein the student identifies the card first and then searches for the appropriate configuration files generated by its' installation
- Build and execute a search of the more popular network cards. The search in this case would be very lengthy with no guarantee that it would even find the card as there are many proprietary cards which may not have been included in the search.

However, in this case again, some students searched directly for the network card as shown in **Figure 6** suggesting that they were informed of the correct answer and they then searched for the network card to find the appropriate evidence for inclusion in the report. Ironically this example posed a problem for the student as it returned 550 hits which would take a long time to analyze before the student is able to bookmark the correct entry.

```
05/04/2011 21:20:52 -- Performed Search Query: (xircom)
        Search Results:  550 Hits in 58 Files
        Search Query: (xircom)
        Search Results:  550 Hits in 58 Files
05/04/2011 21:20:58 -- Displayed Search Results for QUERY: (xircom)
```

Figure 6. A direct search for the network card (Xircom)

In a third question, students were asked to "**identify the operating system used on the notebook**", detail whether there was "**another operating system installed on this machine prior to that**" and identify the "**the installation date of the operating system**". There are a number of ways in which to find the answers to these questions, for instance:

- The registry contains installation details/configurations – this would point to installation dates as well.
- An analysis of the folder structure which sometimes points to previous operating system directory structures. This is not a reliable method – particularly as it does not accurately point to an installation date.
- There are a limited number of potential operating systems (Win XP, 98, Win 95, WinME, Linux etc). If one can understand their installation/upgrading processes and in particular artefacts left behind when they are upgraded, it is possible to search for those particular artefacts.

Figure 7 demonstrates an example of very poor planning but perhaps not an academic misdemeanour. The student attempts to identify the previous *operating system* on the hard disk by searching for the term "operating system". This results in too many search hits (374 to be precise) and the log (the rest of which is not shown in this example) also shows that the student pursued a number of 'seemingly random' searches before finally realizing how to discover evidence of previous operating systems (a few days later in this case).

```
01/04/2011 15:13:08 -- Performed Search Query: (operating System)
  Search Results:  3848 Hits in 374 Files
  Search Query: (operating System)
  Search Results:  3848 Hits in 374 Files
```

Figure 7. A poorly planned search

Of course we cannot jump to conclusions regarding the conduct of a student throughout the task based solely on isolated examples such as the ones presented herein. In each submission a variety of activity was taken into account before students were called in to demonstrate their investigation.

Conclusions and Further work

We have demonstrated that the FTK case log can be used to determine student effort and, in particular, highlight potential academic misdemeanours. Through this process we were able to more easily identify student work which required further attention on our part.

The biggest challenge throughout this work was in being able to analyze a large number of very large log files. This work can be taken further and in particular a formal method of log analysis such as that proposed by Arasteh et al. (2007) could be explored with a view to reducing and mining the log into a precise set of formally defined events, removing file-path/file-names, and also providing timelines of activity. Another area for further work is in a comparative analysis of log files to identify areas where students may have colluded/collaborated.

Our study is clearly linked to the domain of digital forensic investigation, and it may be prohibitive for colleagues to directly implement these ideas within their own areas of specialization; however there is ample room for this work to be expanded into other domains such as programming and the development of web and database systems. We would suggest that this work be considered further by peers to establish further contributions of expanding this work into those domains. An area for future development or research is that of a logging system for programming tasks wherein the entire programming project is automatically logged from the moment the student begins working on the programme to the moment it is completed.

References

AccessData. (2011). AccessData. Retrieved 19th October, 2011, 2011, from http://accessdata.com/

Arasteh, A. R., Debbabi, M., Sakha, A., & Saleh, M. (2007). Analyzing multiple logs for forensic evidence. *International journal of digital investigation, 4s*, 82-91.

baryamureeba, v., & tushabe, f. (2004). *the enhanced digital investigation process model.* retrieved 3rd november, 2011, from http://www.forensicfocus.com/enhanced-digital-investigation-model

Blackboard. (2011). *Blackboard*. Retrieved 3rd November, 2011, from http://www.blackboard.com/

Carrier, B. (2003). *The sleuth kit informer*. Retrieved 8-7-11, 2011, from http://www.sleuthkit.org/informer/sleuthkit-informer-2.txt

Carrier, B. D. (2007). *Basic digital forensic investigation concepts.* Retrieved 3rd November, 2011, from http://www.digital-evidence.org/di_basics.html

CFReDS (NIST). (2007). *Hacking case*. Retrieved 6-7-11, 2011, from www.cfreds.nist.gov/Hacking_Case.html

Digital-Corpora. (2011). *Digital Corpora.* Retrieved 8-7-11, from http://digitalcorpora.org/

East Sussex County Council. (2011). Guidance on keeping of contemporaneous notes. Retrieved 3rd November, 2011, from http://www.eastsussex.gov.uk/NR/rdonlyres/32C474F7-C479-49AA-BA0D-A9C48997490F/0/GUIDANCE_ON_KEEPING_OF_CONTEMPORANEOUS_NOTES.pdf

Guidance Software. (2010). *EnCase Version 6.16*. SAFE Administration Guide.

Guidance Software. (2011). *EnCase forensic*. Retrieved 20th October 2011, 2011, from http://www.guidancesoftware.com/forensic.htm

Jones, M. (2010). *A digital forensics case generator*. Paper presented at the 4th International Conference on Cybercrime Forensics Education and Training.

Kenneally, E. E. (2004). Digital logs - Proof matters. *International Journal of Digital Investigation, 1*, 94-101.

Lallie, H. S. (2010). *Developing usable hard disk images for forensic training, education and research.* Paper presented at the 4th International Conference on Cybercrime Forensics Education and Training, Canterbury Christchurch University.

Mueller, L. (2010). *Forensic practical exercise*. Retrieved 8-7-11, from http://www.lancemueller.com/blog/evidence/Forensic_Practical_3.E01

Murphey, R. (2007). Automated Windows event log forensics. *International Journal of Digital Investigation, 4S*, 92-100.

NHS Scotland. (2011). *Contemporaneous notes.* Retrieved 3rd November, 2011, from http://www.advancedpractice.scot.nhs.uk/legal-and-ethics-guidance/documentation-and-record-keeping/contemporaneous-notes.aspx

Schuster, A. (2007). Introducing the Microsoft Vista event log file format. *International Journal of Digital Investigation, 4S*, 65-72.

State of New Jersey. (2011). *Attorney General directive regarding retention and transmittal of contemporaneous notes of witness interviews and crime scenes.* Retrieved 3rd November, 2011, from http://www.nj.gov/lps/dcj/agguide/directives/dir-2011-2-RetentionTransmittal.pdf

The University Workplace. (2009). Contemporaneous notes. Retrieved 3rd November, 2011, from http://www.universityworkplace.com/2009/01/contemporaneous-notes.html

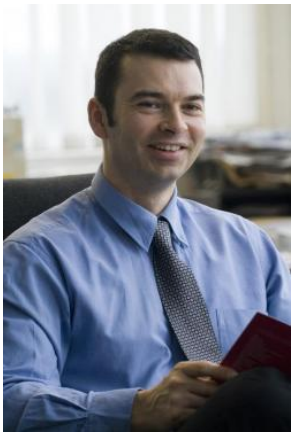
Waters, B. R., Balfanz, D., Durfee, G., & Smetters, D. K. (2004). *Building an encrypted and searchable audit log.* Paper presented at the 11th Annual Network and Distributed System Security Symposium, California.

Your Rights. (2008). Making notes. Retrieved from http://www.yourrights.org.uk/yourrights/the-rights-of-suspects/the-rights-of-suspects-in-the-police-station/making-notes.html

Biographies

Harjinder Singh Lallie (BSc., MSc., MPhil, ABCS) is a senior teaching fellow in Cybersecurity at the University of Warwick (International Digital Laboratory, WMG). He has previously led courses successfully in Digital Forensics and Security at the University of Derby. His research focus is in the area of Digital Forensics and Information Security and is currently studying towards his PhD. Harjinder's research interests include registry forensics, social structure/network analysis, cybercrime investigation and information security.

Philip Lawson (BSc) is a specialist technician in the Computing and Mathematics department at the University of Derby. He specialises in Digital Forensics software and hardware to support the needs of students and staff.



Dr David J. Day is a commercially aware Academic and Consultant specialising in Computer Networks and Information Security. David gained 10 years' experience in the IT sector as a Network Consultant before entering academia in 2005, subsequently completing an MSc and PhD. David's research focuses on Network Intrusion Detection and forensic analysis. Blending academic and commercial experience David has a rare insight into how research can be most appropriately and effectively utilised to ensure the successful commercialisation of projects. At present he is a senior lecturer and consultant in Information Security and Forensics at Sheffield Hallam University.