An Innovative Approach in Digital Forensic Education and Training

Primož Cigoj, Borka Jerman Blažič

An Innovative Approach in Digital Forensic Education and Training

Primož Cigoj[1,2], Borka Jerman Blažič[2]

[1] Jožef Stefan International Postgraduate School, Jamova cesta 39,
1000 Ljubljana, Slovenia
[2] Jozef Stefan Institute, Laboratory for Open Systems and Networks, Jamova cesta 39,
1000 Ljubljana, Slovenia
{primoz, borka}@e5.ijs.si

Abstract. This paper present a novel approach to education in the area of digital forensics based on a multi-platform cloud-computer infrastructure and an innovative computer based tool. The paper presents the tool and describe the different levels of college and university education where the tool is introduced. The tool provides multi-level training that is initiated with the educational levels of the exercises and the content applied. The assessment of the achieved results is provided by the tool at the end of the training session.

Keywords: digital forensic, training environment, education, cloud computing, forensic tool

1 Introduction

E-learning has now been around for more than 10 years. During this time it has changed from being a radical idea, the effectiveness of which was yet to be proven, to something that is now widely regarded as mainstream in modern education. It is also considered as being the core of numerous business plans and services offered by many colleges and universities [1]. Currently, e-learning tends to take the form of online courses. These courses differ in terms of their technology and content, ranging from the resources distributed by MIT's OpenCourseWare project [2] to the design of learning materials offered by colleges and universities from all around the world, acting as the basic unit for the organization of curriculums. The dominant learning technology employed for the management of e-learning is a type of system that organizes, delivers the online courses, and then follows the accomplishments of the learning objectives; it is called the Learning Management System (LMS). This piece of software, which is usually part of the university’s network infrastructure, has become almost ubiquitous in most of the known e-learning environments. Other recent technologies, for example, cloud-computing platforms, are not yet being heavily exploited in the area of e-learning training. This is especially the case for education in the area of digital forensic engineering, which is a field associated with cyber security and the fight against cybercrime and cyber terrorism. These fields are very specific and require intelligent, adaptable approaches that respond to user requirements, coming mainly from officers and members of LEAs (law-enforcement agencies), or private investigators, prosecutors and other cybercrime combatants. Criminal justice education, where digital forensic engineering belongs, is still conducted in a very traditional manner, especially when it comes to capturing digital forensic evidence and its subsequent analysis. The training environment for computer forensics and the methods for fighting cybercrime in most traditional institutions are carried out mainly as a static type of training and closely follow traditional matters and approaches in the criminal justice curriculum. The practical exercises and the challenges presented to the students are usually refreshed slowly [3][4][5]. The exercises that accompany the curriculum are mainly carried out in a class studying classic examples. The practical work is focused on solving a task: "find the flag", and the flag to be found is always at the same location as where the training takes place. In a real-life situation, digital investigators are faced with much more complex tasks, where they are forced to use a wide range of methods to solve the problem [6].

In this paper we describe a novel approach to educating and training in the area of digital forensic engineering. The training is based on the use of a dynamic tool developed within the E-Forensics Educational Community that acts within the D-FET project consortium (Digital Forensic Education and Training project). The tool is installed and offered over the cloud-based infrastructure of the D-FET project [7]. Cloud computing introduces an efficient mechanism for a wide range of services that offer real-life environments in the area of cybersecurity and digital forensics, which is not being sufficiently well exploited for on-line education. The cloud-based infrastructure enables the construction of on-dynamic e-learning systems and tools, making this training very close to reality and to real-life situations [8]. The DFET project is funded within the EU's ISEC program [9] and the main objective is the development of an innovative educational approach in the area of digital forensics.

The paper is organized as follows: the next section introduces the cloud-based infrastructure and the basic features of the EduFors tool. The section that follows describes the technical details of the tool and the educational approach applied during the training process. The collected experiences of the performed training and the received feedback are presented in the fourth section, which is followed by a brief discussion and concluding remarks.

2 The D-FET Cloud-Computing Training Environment

The D-FET project is a training environment that consists of a virtual cloud-based platform that enables the sharing of courses and the use of laboratory training material. The training environment is built up from virtual machines that are generated to an extent that depends on the number of enrolled trainees, meaning that the number of virtual machines is sufficient for the requirements of the training. They can be accessed remotely. The training environment is dynamic and follows the evolving nature of the analytical cybercrime methods and approaches, as well as the technology development of cyberspace. The D-FET training environment caters for a range of educational levels, enabling education for Information and Communications Technology (ICT) students and as well as for law-enforcement professionals. Currently, the cloud infrastructure is owned and shared by the institutions of the participating countries (Slovenia, UK, Ireland, and Sweden).

Figure 1 outlines the generalised training infrastructure and the approach used in creating real-life-based instances of cybercrime attacks or fraud, where an instructor has the possibility to generate an active instance for the training challenge, such as one related to finding data associated with an act of crime on a PC or on a smartphone. These active instances are generally based on known criminal use cases, such as the investigation of a computer for financial fraud, or a denial of service, where an investigation must create an evidence bag and handle the evidence correctly according to the legislative rules, making it intelligible for use in a court. The cloud platform contains an incorporated educational tool known as EduFors, which is designed to provide several cyber forensics images, as instances originating from an attacked operating system or network server.

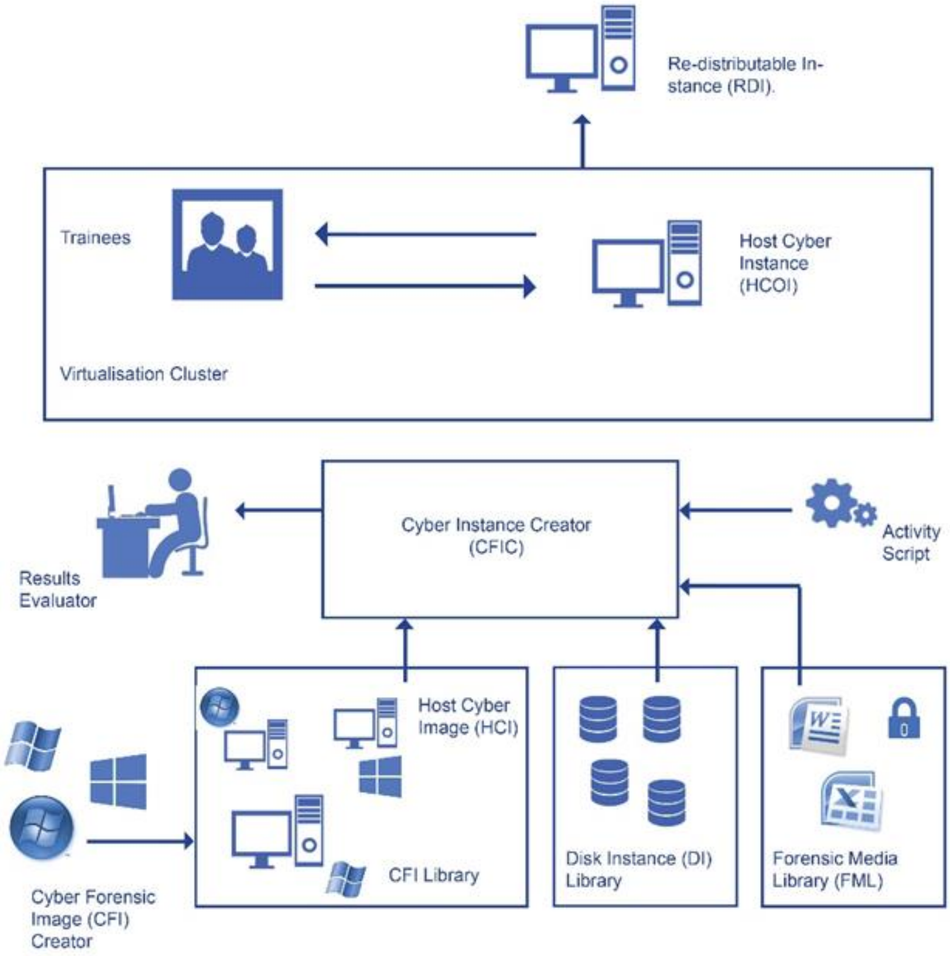

Re-distributable Instance (RDI).
Trainees
Host Cyber Instance (HCOI)
Virtualisation Cluster
Results Evaluator
Cyber Instance Creator (CFIC)
Activity Script
Host Cyber Image (HCI)
CFI Library
Disk Instance (DI) Library
Forensic Media Library (FML)
Cyber Forensic Image (CFI) Creator

Fig. 1. The general training infrastructure

Using this tool, multiple instances across the network are created and the challenge presented to the trainee is to collect the evidence from the criminal attack across a number of network-connected devices. The tool then adds the required disk instances for selected scenarios of different types of criminal act and prepares them to be analysed by a digital forensic tool, such as the X-Ways software packet. These forensic analytical tools are stored in the Forensics Media Library built within the DFET project and are triggered for use during the training process. Another key part of the process intended for education and training is the in-built metrics for the assessment of the trainee's performance. These metrics are related to the following:

- The time necessary to find the required evidence (which is limited in accordance with some difficulty level that is specified in advance);
- The investigation method used;
- The application of the right parameters/indicators within the applied forensic tool.

The parameters identifying the factual, conceptual, procedural and metacognitive learning outcomes are introduced in the assessment part of the educational tool. Here, we present only the main training process by providing an explanation of the properties of the EduFors tool.

3 The EduFors Tool

3.1 Basic Architecture Description

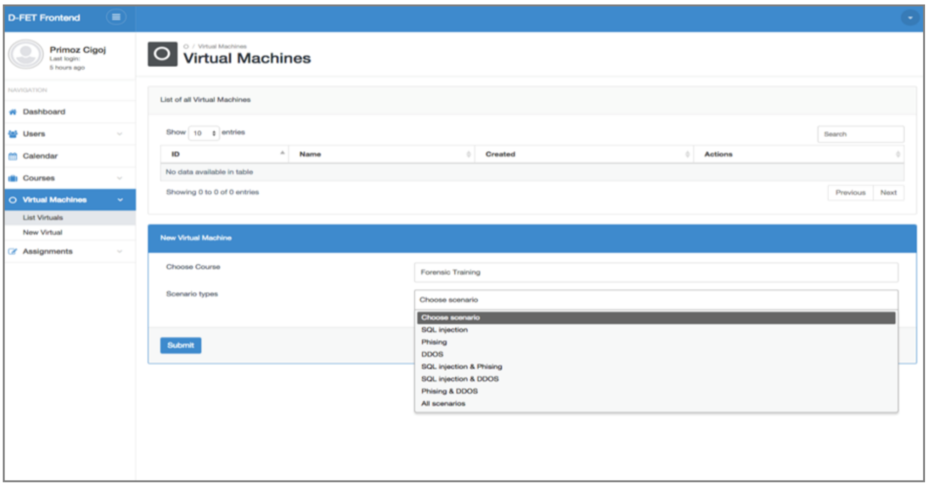
The tool is designed to generate instances based on known cybercrime scenarios and to present them to the trainee. The required levels of skills and understandings to solve the challenge are accommodated in such a way that it reflects the different educational levels to be employed for the different trainee groups. The EduFors tool consists of two parts: the front end, which communicates (using an API – Application Interface) with the backend, which is responsible for the generation of the instances representing different situations and for the management of the training process. The front end allows the trainees to enrol in the system and to gain access to all the available courses for their level of education. The appointed administrator creates paths for accessing the courses, giving assignments to the trainees and manages the virtual machines generated in different platforms available in the cloud. This part of the system is also responsible for the presentation of the log files, the generation of the images and the injection of standardized templates for each of the crime scenarios that is selected to be examined by the trainee. The tool is also responsible for managing the virtual machines on the remote cloud platforms. Representational State Transfer (REST) access is used (REST over the HTTP protocol) for communications with the remote terminal used by the trainee. The data are exchanged in the JSON format.

The registration of the trainee is enabled by direct self-registration or via a social network account (e.g., Facebook). Adding a new course to the system is fairly straightforward, as just the title and the description of the course are required to be entered, together with the duration of the course, accompanied by the course material. Once the trainees are registered and the courses saved in the system, the administrator starts with the preparation of the virtual forensic machines. Six combinations of cybercrime scenarios are currently available in the EduFors tool, but there are plans to add more. Every time the administrator creates a new virtual machine for each of the existing scenarios, a different dynamic attack is applied to this machine. Figure 2 shows the list of pre-created virtual machines and the available crime scenarios, stored on the forensic disks, with the crime-attack data that are afterwards embedded in each template presented to the trainee. The dynamic templates for each type of criminal attack and for each trainee are unique because of the different instances and the injected data.

Each of the criminal attacks is presented to the trainee as a standard template, stored on the virtual machine with the respective forensic disk. The templates presented to the trainee differ in terms of the injected data from the templates presented to another trainee and regarding the required level of skills and knowledge that a particular cybercrime case requires to be solved and for the production of evidence.

3.2 Training Scenarios and Assignments

Currently, the EduFors tool generates dynamic forensic templates for the three most frequent cybercrime scenarios: phishing, SQL-based data leakage and a distributed denial-of-service (DDOS) attack. However, in the next version of the tool other cybercrime scenarios will be added according to the definition in the ISO standard [10]. We present all of them briefly, as follows.



List of all Virtual Machines

Show 10 entries
Search

ID	Name	Created	Actions
No data available in table			

Showing 0 to 0 of 0 entries
Previous Next

New Virtual Machine

Choose Course: Forensic Training

Scenario types: Choose scenario

- Choose scenario
- SQL injection
- Phising
- DDOS
- SQL injection & Phising
- SQL injection & DDOS
- Phising & DDOS
- All scenarios

Submit

Fig. 2. The EduFors forensic virtual machines

Phishing. In this scenario, the client Adam discovers that his bank account has been compromised using a phishing method. The scenario is constructed with the use of two virtual machines (A and C) and a bank server. The attacker has obtained access to the server C by exploiting weak password protection, as he/she has created a fake website as an imitation of the client bank's server. By sending forged emails to Adam, and inviting him to access his bank, the attacker tricks him into believing that he is actually accessing his trusted bank website. That causes the client A to send personal information to the fake host, residing on the attacker server C, and misleading him into believing that he was exchanging information with the bank (server B). The implementation of the phishing scenario and the data capture for forensic analyses require a website that retrieves and stores the entered credentials, the MySQL database for storing the retrieved credentials and various server log files. The log files contain the data of the random accesses to the phishing site by the victims and by the attacker. The instances and the template data are provided by a script, written in PHP language, that uses two parameters – the test name and the test sequence number, for example, "phishing 1". The script then runs the attack scenario by picking up a random date for the time when the attack occurred. A list of template logs is then preloaded from the template folder, which is then used to generate the victim logs in the Apache OS and for the MySQL log files. This script is executed on a special virtual machine that has access to the VMware platform of the cloud and the data storage. The attacker's template is selected and the placeholders are replaced with randomly selected data. These random data are built up from the IP address, the date and the hour of the accesses. The IP address is unique to the whole log and is not repeated for any other victim logs. The populated template, based on these data, is then stored within the output file for the trainee. Each training day can have a random number of events (between 6 and 24). To make the attacker's footprint slightly harder to find by the trainee, additional data lines are added at the top of the log files (adding data for several days up to 10). The same is applied to the end of the log files (for 1 to 10 days). The days are generated incrementally and de-incrementally from the remainder of the generated log, so the resulting logs are listed in chronological order. The results are stored in the local MySQL database (the attacker's IP address, the date and the hour of the attack and the type of scenario). A prepared, empty, virtual disk image is copied and mounted as a local file system using the libguestfs tool [11]. The logs are then injected inside the virtual disk image, the virtual machine template is cloned on the VMware server and the modified disk image is uploaded to the data storage. In this way it is possible to enable a new virtual machine to be prepared for inspection of the forensic image. The results, along with the virtual-machine identification, are then returned as an output in the JSON format. The trainee receives a template of Adam's PC and the templates from the compromised server. In this scenario, the assignment given to the trainee consists of the following tasks:

- find the IP source address of the attacker,
- explain how the attacker has exploited the bank server,
- locate the directory where the fake website of the bank is hidden,
- explain how the attacker stored the required data for the phishing scenario,
- locate the other victims' IP address(es).

The answers to these assignments are then stored, the correctness is checked, and an evaluation is then provided.

Data leakage and SQL exploit. In this scenario, a website is the victim of an SQL injection attack. The attacker has used the search bar to access the website's database. When the administrator of the attacked web server comes to the conclusion that the web data are compromised, he immediately contacts the police. The server is then disabled to prevent any further exploitation by the attackers and the logs are brought to the trainee.

The crime scenario for the data leakage and the SQL exploitation are implemented using the same script as for the phishing scenario, but the task generation is based on different log templates and website structures. The websites are modified so that they become vulnerable to SQL injection, which allows the attacker to run a multi-query MySQL statement, including uploading a binary file and saving it directly within a web-accessible folder. This attack is accomplished with an open-source tool known as SQLMap, which implements the SQL injection method and allows the execution of the remote SQL statements. Several security features of the configured machine for training must be disabled as the most recent OS Ubuntu software versions contain patches that solve the so-far identified security vulnerabilities. We mention some of them here: prevention of writing MySQL data inside the Apache's /var/www/html folder, and removal of the MySQL access to the group www-data, etc. In the Ubuntu OS other security features were also introduced, such as the prevention of the use of the multi-query MySQLi function, which allowed the execution of additional full SQL statements (as opposed to generally used, single-query, MySQLi functions that are not vulnerable to SQL injection attacks).

The file that generates the template data for this type of attack is a PHP script in the form of a web file that allows simpler uploading for larger additional scripts. This scenario uses a publicly accessible PHP shell and a multipurpose script known under the acronym "cyb3r sh3ll". By exploiting these scripts, the attacker can retrieve various types of information from the server database, and each case of access generates a different set of data that is retrieved by the attacker. The script that generates the log files for the trainee uses another type of template. The template matches the specific filenames of the attack scripts and the SQL queries that they execute. As an addition to the MySQL log, the script attaches the SQL injection query that has executed the attack. The assignments for this scenario given to the trainee are as follows:

- Decide whether it has come to an SQL injection on the particular identified server;
- Identify the IP source address of the attacker;
- Find which data were compromised;
- Fix the attacked website.

Distributed Denial-Of-Service (DDOS). A DDOS scenario is implemented using the same script as phishing and data leakage, but the log templates presented to the trainee are different. This attack is implemented by simulating a small-scale DDOS attack that is initiated from different machines. Some of them are just normal computer-machine clients, but the others are virtual. The network traffic for the required evidence is captured on the victim’s server. The script for this scenario picks a random date to indicate the date when the attack has occurred. A list of template logs is then preloaded from the template folder to the Apache server log files and presented to the trainee. The assignments given for this type of tasks are as follows:

- Decide whether it has come to a DDOS attack;
- Identify the IP source address of the attacker(s);
- Find if a botnet network was involved in the attack;

– Assess the damage, taking into account the time for which the server was disabled;

The processes run by the EduFors tool during the laboratory training are presented in Figure 3.

3.3 The training exercise

Each training includes laboratory work constructed from the assignments generated by the EduFors tool. The assignments are presented to the student as a choice of the available scenarios prepared for a particular educational level and the associated laboratory training. The student selects a one-by-one scenario as a result, and finishes the training after all the task assignments have been solved successfully. After the student finishes the assignment tasks, the educator evaluates the used time and the correctness of the solved forensic problem(s) provided by the EduFors tool.

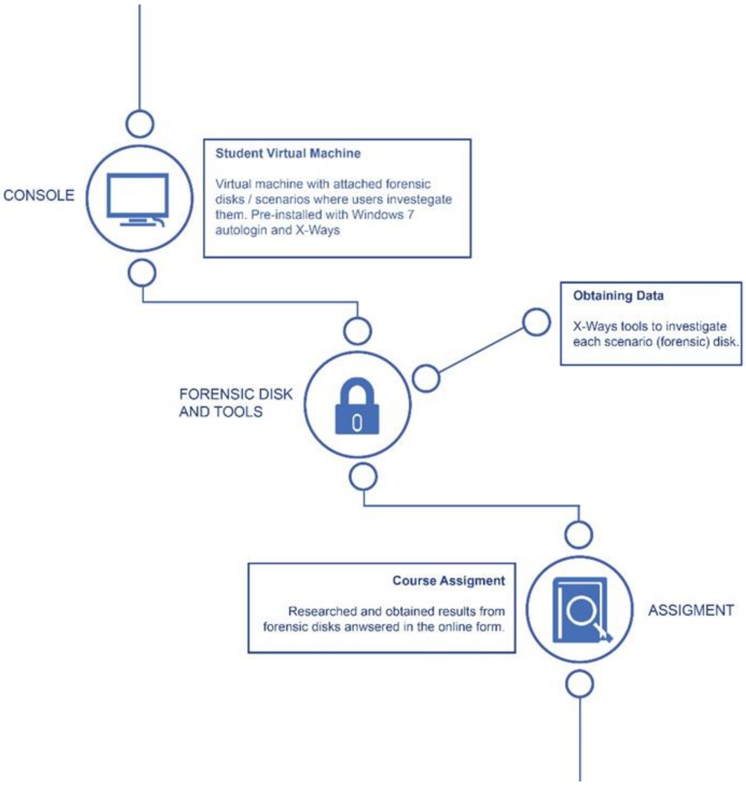

CONSOLE
Student Virtual Machine
Virtual machine with attached forensic disks / scenarios where users investigate them. Pre-installed with Windows 7 autologin and X-Ways
Obtaining Data
X-Ways tools to investigate each scenario (forensic) disk.
FORENSIC DISK AND TOOLS
0
Course Assigment
Researched and obtained results from forensic disks anwsered in the online form.
ASSIGMENT

Fig. 3. EduFors process

The tool collects the following data: the time slot between the opening of an assignment and its closing, and the time the student has used to accomplish the task or to answer the questions. If the time spent for solving the tasks is longer than the pre-allocated time for each task, this is acknowledged to the educator and a negative point is entered into the calculated scores for the correct answers. The allocated time per assignment differs and depends on the difficulty level required for the task solving. If 50% of the answers submitted by the student are correct and the time is not exceeded, then a positive score is given by the tool to this trainee. However, this percentage can be changed by the educator, depending on his/her requirements. To each participant in the training, e.g., trainee, educator and administrator, a different dashboard view of the tool EduFors is provided and this feature allows a visible follow up of the performed tasks and the achieved level of success for each of them.

The EduFors tool was recently launched and used in one of the seminars offered by the MSc program of the International Postgraduate School Jožef Stefan, so exhaustive experience cannot be reported yet. However, the first feedbacks from the educators and the trainees are very positive.

4. Conclusion

Teaching digital forensics is a demanding area of education as it involves intensive, hands-on exercises that require students to follow potentially tedious procedures that demand a long and focused span of attention. Due to these challenges, current forensics courses are often designed for advanced students that are capable of following a variety of demanding disciplines from OS, IP networking and traffic monitoring, file system analysis, basic web applications or protocols. The knowledge of these disciplines is a prerequisite for digital forensic students, so that they are capable of absorbing advanced, abstract concepts used in discovering acts of cybercrime and frauds. The experiments used for training are usually tedious, static and are not frequently applied during the study. In this paper we have proposed an innovative idea to overcome some of the difficulties associated with digital forensics education, based on a successful combination of the visualization technologies and the dynamic generation instances in a real cloud-based computing environment. The current experiences point to the conclusion that this approach will be very effective in the teaching of digital forensics and in other advanced, computer-based fields that involve an understanding of abstract concepts and hands-on practice. Future work includes upgrading the EduFors tool with game elements that will contribute to the attractiveness of EduFors applications and for triggering more attention from the learners and educators. We also plan to work on a further assessment and evaluation for measuring the effectiveness of the presented educational approach.

References

1. Downes, S.E.: Learning 2.0. The eLearn Magazine. (2005) http://www.elearnmag.org/subpage.cfm?section=articles&article=29-1, accessed 12th December 2014
2. Massachusetts Institute of Technology: Open Course Ware project. http://ocw.mit.edu/index.htm, accessed 1st Decemeber 2014

3. Pollitt, M., Nance, K., Hay, B., Dodge, R. C., Craiger, P., Burke, P., & Brubaker, B. Virtualization and digital forensics: a research and education agenda. Journal of Digital Forensic Practice, 2(2), 62-73. (2008)
4. Gottschalk, L., Liu, J., Dathan, B., Fitzgerald, S., & Stein, M. Computer forensics programs in higher education: a preliminary study. In ACM SIGCSE Bulletin (Vol. 37, No. 1, pp. 147-151). ACM (2005)
5. Taylor, C., Endicott-Popovsky, B., & Phillips, A. Forensics education: Assessment and measures of excellence. In Systematic Approaches to Digital Forensic Engineering, 2007. Second International Workshop on (pp. 155-165). IEEE (2012)
6. Hartel, P., Junger, M., Wieringa, R.: Cyber-crime Science = Crime Science + Information Security. Twente University Report. http:// eprints.eemcs.utwente.nl/18500/ accessed 1st Decemeber, 2014
7. D-FET project. http://www.d-fet.eu/project-overview/, accessed 5th Decemeber, 2014
8. Laisheng, X., Zhengxia, W.: Cloud Computing a New Business Paradigm for E-learning. In: Proceeding of International Conference on Measuring Technology and Mechatronics Automation. (2011) 716–719
9. ISEC: Prevention and fight against crime. http://ec.europa.eu/dgs/home-affairs/financing/fundings/security-and-safeguarding-liberties/prevention-of-and-fight-against-crime/index_en.htm, accessed 18th December, 2014
10. ISO, 2012. ISO 27037:2012 Information technology — security techniques — guidelines for identification, collection, acquisition, and preservation of digital evidence accessed 2nd Match, 2015
11. Diagnostics for libguestfs. http://libguestfs.org/libguestfs-test-tool.1.html accessed 5th December, 2014