Automated Production of Predetermined Digital Evidence

Aniello Castiglione[1⋆], Giuseppe Cattaneo[1], Alfredo De Santis[1]
Giancarlo De Maio[1], and Mario Ianulardo[2]

[1] Dipartimento di Informatica *"R. M. Capocelli"*
Università degli Studi di Salerno, I-84084, Fisciano (SA), Italy
castiglione@acm.org, cattaneo@dia.unisa.it, ads@dia.unisa.it
demaio@dia.unisa.it
[2] IISFA Italian Chapter & Lawyer in Naples, Italy
marioianulardo@codicieleggi.it

Abstract. Recent legal cases have shown that the *digital evidences* are increasingly used in proceedings (by defense, accusation, public prosecutor, etc.). Digital tracks can be left on computers, phones, digital cameras and also on third party servers belonging to ISPs, telephone providers, companies that provide services via Internet such as YouTube, Picasa, Facebook, Gmail and DropBox.

It is possible to suppose that these digital tracks can be forged ad-hoc in order to set up a false alibi with the help of a collaborator. This work points out that a criminal could set up a false digital alibi without any collaborator. The key idea is that it is possible to leave digital evidences in a fully automatic way without any human intervention. The so forged digital tracks will be indistinguishable ex-post from any digital evidences eventually produced by a human potentially in the same place and in the same time.

A forensic investigation that will only consider digital evidences will never be able to establish if such tracks have been produced by a human behaviour or by an automated tool. These considerations emphasize the difference between digital and physical (traditional) evidences: in substance, the digital evidences should be considered relevant only if supported by evidences collected using the traditional investigation techniques. The considerations presented in this work should be considered by any entity involved in a digital investigation or digital forensics activity because, starting from now, each digital track should be considered not like "the solution" but must be correlated with any additional information provided by the various disciplines of the Forensics Sciences.

Keywords: Digital Evidence; Digital Investigation; Digital Forensics; Anti-Forensics; Counter-Forensics; False Digital Evidence; Automated Alibi; False Alibi; Digital Alibi

⋆ Corresponding author: Aniello Castiglione, castiglione@ieee.org, Phone: +39089969594, FAX: +39089969821

1 Introduction

1.1 The Digital Evidences

The use of digital technology is rapidly growing. The number of users in the world that is using the Internet is almost 2 billions, with a penetration of 28.7% of the world population [1]. As a consequence, more and more crimes are performed on the Internet or have something to do with digital equipments. For these reasons, there is a growing increase in digital evidences being brought in courts in the United States and elsewhere in the world.

Consequently, courts are becoming concerned about the admissibility and probative value of digital evidence. Although digital devices have not been directly used by an individual that have been indicted to have committed a crime, they can be subject to forensic investigations in order to collect useful tracks about the behavior of such person, either to be cleared of an accuse or to charged of an offense. Elements required to determine the liability for having committed a crime often consists of files stored in the memory of a PC, of photos on a digital camera, of information lying on a mobile phone, and of all information that can be found in many other digital devices.

Digital tracks are *ubiquitous*: they can be located anywhere in the world. In fact, digital tracks can be retrieved on mobile devices (phones, PDAs, laptops, GPS navigators, etc.) but especially on servers that provide services via Internet, which often register the IP address and other informations concerning the connected clients. These servers can be located Countries different than the one in which the crime has been committed, and different national legislations can be an obstacle in acquiring the digital evidences during the investigation.

Digital tracks are also *immaterial*: it is well known that all digital data present on a digital device is represented as one's and zero's. This data can be modified by a user having enough access privilege to such device. For example, on a PC running a multitask/multiuser Operating System, such as Microsoft Windows, Apple Mac OSX or Linux, which distinguish between "*user*" and "*kernel/superuser*" mode, it will exists a specific user — for example, the "*root*" in a Linux environment — that can execute every operation on the system including the access to whatever hardware/software resource, leading to the possibility of modifying the data stored in the PC memories (RAM and/or disks). Devices running simpler OSs with respect to PCs, such as GPSs, digital cameras, and, in part, mobile phones, often do not distinguish between access modes. In such cases, any person who has physical access to the device can modify its memory without the necessity of gaining superuser privileges on the system.

The Digital Forensics Science is constantly subject to change and evolution because it is mainly influenced by technological innovations. It make necessary to constantly upgrade not only the tools for detecting and reporting digital tracks, but also the analysis methodologies and the personnel that manage such analysis tools. Judges, juries, and attorneys are more and more aware of the presence and of the relevant value of digital evidences. However, much of the digital evidence

is never get seen by judges, juries, and the public. They assume that any digital evidence that the investigators find could have been produced only by the users and in particular by the accused person. The digital forensic techniques have to be upgraded in order to meet the growing demand of scientific evidences in legal cases: it is known as the "*CSI effect*" [35]. The term most often refers to the belief that jurors have come to demand more forensic evidence in criminal trials, thereby raising the standard of proof for prosecutors. Although this belief is widely held among American legal professionals, several studies have shown that crime shows are unlikely to cause such an effect.

1.2 The Digital Alibi

Computers can be used in the commission of crimes and can contain evidence of crimes, but can also be an *alibi* for the defense of an accused person. In the Latin language the word "*alibi*" is an adverb meaning "*in or at another place*". According to the Webster's NewWorld dictionary, alibi is "the defensive plea or fact that an accused person was elsewhere than at the scene of the crime with which the person is charged."

There are two examples of legal proceedings in which the digital evidence has been considered an alibi that contributed to exonerate the accused.

Rodney Bradford, 19 years old resident of New York, was arrested on October 18, 2009 for suspicion of armed robbery at the Farragut Houses in Brooklyn, where he lives [2], [3], [4]. His defense lawyer, Robert Reuland, claimed the innocence of Mr. Bradford asserting that he was at his father's house, located in the Harlem quarter, at the time of the crime. The evidence offered in support of this thesis was a message posted by the suspected to his girlfriend - "*On the phone with this fat chick... where my IHOP.*" - on his facebook page having timestamp "October 17 - 11:49 AM", exactly one minute before the robbery. The status update would take place from his father's PC. The subsequent investigation confirmed that the connection was established from an apartment located in the 71 West, 118th Street of Manhattan, i.e. the father's house, which was far more than thirteen miles from the scene of the crime. Rodney Bradford was released 12 days after his arrest.

This is probably the first case in which a status update on Facebook has been used as an alibi. It is clear that anyone who knew the appropriate username and password could modify the Facebook profile. For example, these actions may have been made by a partner. However, according to defense attorney Reuland, this possibility was remote because it would imply a level of criminal genius unusual in one so young.

Another court case, extremely interesting in terms of assessing the digital alibi, is the case Garlasco ([4] [5]). It ended with the acquittal of Alberto Stasi, the main suspect in the murder of his girlfriend Chiara Poggi. The defendant proclaimed his innocence claiming a digital alibi: when his girlfriend was murdered he was writing the thesis on his computer. This court case is characterized by a close comparison between the results of analysis performed on each type of specimen, such as DNA traces and digital evidence on the PCs of the victim

and the suspected. These findings were complemented by traditional techniques. However, the attention of the investigators still focused on verifying that digital alibi claimed by Stasi was true or false. While noting the presence of errors committed by the experts at the stage of retrieving and analysis of digital evidence, the Court directed an acquittal of the accused person.

This means that the digital alibi, although undermined by mistakes and although it was made necessary the use of analysis of metadata for reconstructing the parties affected, has proved to the court that the suspect was working on his laptop during the time of the crime.

— gi dette? —

Differences between "classic" alibi and digital alibi-

———————

The problem of user identification.

These court cases make you think about the origins of digital evidence, as it is not always possible to trace the identity of the author, which may be a person or an automated program. This work will shown that it is possible to set up a series of automated actions in order to leave digital traces that would be completely indistinguishable from evidences leaved by a person. The scope of this paper is to show the strengths and weaknesses of a new anti-forensic technique in order to implement digital forensic tools to detect this threat.

The remainder of this work is organized as follows: in Section 2 we introduce ... In Section 2.1 we describe ... In Section 3 we describe ... In Section 4 we report the results obtained in an experimental evaluation of the indistinguishability of our proposed technique... In Section 5 ...

2 Creation of a False Digital Alibi

In order to forge his digital alibi, a criminal can use different strategies depending on three main factors: risks, costs and implementation complexity (tools to use, necessary skills, trace to delete a-posteriori, etc.). In general, there is three options.

- Accomplice
  It is the traditional anti-forensic technique. The criminal can engage one or more *accomplices* which use digital devices (PC, cellphone etc.) in order to leave traces in his stead. This technique do not require the involved people to have particular skills, but can be very dangerous and expensive. In fact, it is not so simple for a criminal to find a trustworthy accomplice, furthermore the traditional investigation techniques can reveal evidences belonging to the accomplice(s).
- Remotization
  The criminal can access his digital devices staying in another place. For example, he can access his PC using an IP connection over Internet. Generally, remote accesses leave traces on the involved devices and data about the connections can be stored by the routers along the path on the network. For

these reasons, the criminal is required to be skilled enough to delete every inconvenient evidence. Moreover, the devices used for the remotization can be very expensive.
- Automation
  The criminal can use fully automated tools to generate digital evidences on his devices. As shown later in the paper, this technique requires the criminal to have intermediate skills in using computers, have no costs and intermediate risks.

2.1 Remotization

In this section three techniques to forge an alibi setting up a personal computer to be remotely controlled have been discussed.
distant place.

Remote connection through KVM over IP A criminal can use a KVM over IP switch (iKVM)[3] to control his PC remotely from a mobile station. This technique do not require any suspect software to be installed on the PC. However, the criminal must take some precautions to limit the amount of traces left. For example, he must configure the iKVM with a static IP address to avoid requests of the device to DHCP servers which can register its MAC. Moreover, if a local router is present, the criminal may configure it to allow the communication with the iKVM.

While assuming that the criminal took all reasonable precautions to cover their suspicious traces, an accurate investigation at the ISP can reveal the unusual IP connection persisting for the overall duration of the alibi.

Remote connection through Remote PC Control software A criminal can use a Remote PC Control software to access his PC from a mobile station. To limit suspicious traces, he can use a portable software installed on a USB pendrive connected to the PC. For example, TeamViewer Portable for Windows (free for personal use use) can accomplish this task. However, such software leave some traces on the Windows Registry and prefetch files on the hard disk. Moreover, the IP connection used by the mobile station to access the Remote PC Control software can leave traces on ISP and routers along the network path.

In both the iKVM and Remote PC Control software case, the criminal should obfuscate the auxiliary hardware such as the iKVM switch and the USB pendrive.

[3] A KVM switch (with KVM being an abbreviation for keyboard, video or visual display unit, mouse) is a hardware device that allows a user to control multiple computers from a single keyboard, video monitor and mouse. KVM over IP devices use a dedicated microcontroller and potentially specialized video capture hardware to capture the video, keyboard, and mouse signals, compress and convert them into packets, and send them over an Ethernet link to a remote console application that unpacks and reconstitutes the dynamic graphical image.

3 Undistinguishable automated production of Digital Evidence

The anti-forensics technique discussed in this paper consists of forging a digital alibi using fully automated tools that can simulate the actions performed by an human during the time of a crime. As explained in 1.2, such digital evidences can lead to the acquittal of an accused person.

The typical actions performed by an ordinary person on his PC, which may be simulated by automated tools, are mouse clicks, pressure of keyboard keys, writing of texts, use of specific software, all interleaved with random timings.

There are a lot of computer applications capable to perform these tasks automatically, for example:

- AutoHotKey (for Windows) [13];
- AutoIt (for Windows) [14];
- Windows Host Script (for Windows) [15];
- DoThisNow (for Windows and Linux) [16];
- GNU Xnee (for Linux) [17];
- Automator (for Mac Os X) [18].

Among these tools, the most suitable for anti-forensic purposes are Automator and AutoIt. Automator is part of the software bundle of Mac OS X from version 10.4 (Tiger). AutoIt is a freeware software for Windows environment. Both the tools can receive in input script files which contains the actions to perform. Alternatively, scripts can be compiled into standalone executables which can be executed on the target system without the necessity that the tools are installed. With respect to Automator which is part of the Mac OS X software bundle, latter feature is more important for AutoIt, because it does not come pre-installed on Windows systems and the presence of such software can be suspicious.

Recently, because of their growing usefulness, many resources like tutorials, online communities, tools, downloads, and books on automation tools are becaming available [32].

The automation tools are useful to avoid boring, manual, repetitive, and error-prone tasks. That is, they speed up otherwise tedious, time-consuming tasks, avoiding the possibility of errors while doing those tasks. Applications of automator tools include Data analysis, Data munging, Data extraction, Data transformation, and Data integration [32].

In this paper we point out a new potential application of automation tools: How to use them to construct a digital alibi.

Automation tools have the possibility to perform simple operations like:

- simulate keystrokes and mouse gestures,
- manage windows (e.g., activation, opening, closing, resizing),
- get information on and interact with edit boxes, check boxes, list boxes, combos, buttons, status bars,

– control time for operation (e.g., choose time to schedule each operation or choose time delay between consecutive tasks).

Automation tools usually provide much powerful functions. There are also libraries, modules and a high-level scripting language available for users and developers.

However the basic and simple operations listed above are sufficient to automate tasks for our purpose of constructing a digital alibi. The list of tasks include:

– *Web navigation.* Opening new tabs, new windows, new URL. Inserting username, password, text. Uploading or downloading files. These include interaction with social networks, and popular websites like Picasa, Dropbox, Gmail.
– *Files and folders.* Processing specific files, renaming them, working with folders.
– *Photos and images.* Processing photos, cropping images, creating thumbnails.
– *Music and audio files.* Play an audio file. Adjusting audio controls. Converting audio to text.
– *Compound files.* Create new text files, modifing (inserting and deleting) them, saving them. These include Office documents, processed by Word, Excel, and Powerpoint.
– *Computer applications.* Launching any application. For example, launching a browser or using email by opening unread messages and sending new messages with attached just created files.
– *Phone calls.* While it would be easy to simulate a phone call using IP telephony like Skype/VoIP, it is possible to make a phone call over the PSTN circuit or GSM mobile network by using additional hardware, as well as send SMS.

3.1 Detecting Digital Evidence of automation

The construction of the digital alibi consists of two phases: one is the developing and testing of the executable and then the execution of this script. Several unwanted traces can be produced during both these phases. The criminal may obfuscate them in order to avoid any suspect during the forensics analysis.

Script development and testing Before an executable script is launched in order to automatically simulate the human behavior, the script itself must be developed and eventually tested to verify that it will acts correctly.

Writing the script can leave some unwanted traces: the OS, in fact, typically records recently opened files and applications. For example, Windows stores these informations in the Registry, which can be only modified by the Administrator user and the modifications take effect only after a system reboot. Therefore, an accurate (live) forensic analysis can reveal these evidences.

In most cases the executable script must be extensively tested before it will be used for a so sensible task as forging a digital alibi. In fact, executable scripts created using tools like AutoIt and Automator are strictly connected to the running environment. For example, the mouse movements and clicks must be specified using absolute coordinates (x,y), therefore the different position of an element on the screen results in a different behavior of the automation. Due to these considerations, the script must be tested on a system that has the same appearance as the target system (screen resolution, windows position, desktop theme, icon size, etc.).

A criminal can employ some workarounds to avoid most of suspicious traces about the development and testing of the script.

- *The same system* - The criminal can use the same system that will be used to forge his alibi to develop and test the automation. These tasks leave several traces that can be difficult to retrieve and remove: Registry entries, prefetch data and virtual memory informations on the filesystem, repetitive execution of specific operations and so on.
- *Virtual machine* - A virtual machine running an identical copy of the OS of the target system can be used in order to write, compile and run the script for testing. This technique do not leave any unwanted trace on the target system except the files containing the virtual machine image and traces that the virtual machine itself has been launched. However, as discussed in **??**, it can be very difficult to delete any trace of the big-sized virtual machine image from the hard disk.
- *Live OS* - A live CD or live USB version of the target OS can be used in order to develop and test the automation. This technique do not leave any unwanted trace on the hard disk because the live OS only use the central memory for all his operations. However, it can be an hard task the cloning of the target OS on a live media.
- *Another system* - The criminal can simply develop and test the automation on another PC running an identical OS. Subsequently, he can copy the script on a removable media and run it directly from there. In this case, the entire secondary PC must be obfuscated in order to avoid any forensic analysis on it.

It is supposed that traces recorded by the OS as mounting an external drive are not considered suspicious.

Script execution traces For any Operating System (OS) *process* is considered as the basic execution unit [**?**]), and even the simplest OS provides mechanisms to trace the execution of each process it runs saving data such as executable name, the time was started, the amount of CPU was allocated during the execution, max resident size for virtual memory and so on. We generically define these records *accounting data*.

Depending on the OS, the execution of a script generated with tools as AutoIt and Automator also leaves this kind of traces. For example, Windows implements a mechanism to speed up the startup of programs called Prefetch[**?**],

which records informations about executed applications on the filesystem. Windows also implements the Virtual Memory technique[**?**], which transfers data about processes on the filesystem in order to provide a virtual memory space wider than the real one. More traces about the executed programs are also memorized in the Registry.

It is important to note that erasing all data about the automation from a drive is an hard task. In fact, the simple deletion/modification of a file using the OS specific system calls do not completely removes the presence of the old data from the drive. It mostly happens on magnetic devices as hard disks, where electromagnetic traces remain even after the data deletion. These hidden traces are often revealed by digital forensics analysis. A more detailed discussion on how it is possible to remove these traces is addressed in **??**.

In order to forge a perfect alibi, the criminal should delete all these unwanted traces deriving from the execution of the automation. In section 4 it is discussed how this task can be performed on a Windows environment.

Logon traces Besides the data related to the process execution, another dedicated OS module is in charge of storing each user access to the system *logon data*; normally this is done during login-logout phases and the module is supposed to record data such as local login time, local logout time, source address of the connection (if the operation was performed through the net) or the tty (the serial line) the user used to connect to the terminal both for local or modem access.

There were many historical reasons for an OS to keep trace of such data. For example in the old time sharing OSs this was necessary for billing purposes, but actually it holds steady even with the advent of the personal workstations and the personal OSs. In fact the accounting modules have been considered useful to provide the user with statistical data about the system and the CPU usage.

The accounting data are collected in memory by a kernel module (process management) and stored on several files when the process terminates its execution. On the other hand logon data are immediately stored on disk when the operation is performed. Data on disk are stored in multiple formats in order to efficiently face with the big amount of records (an OS can run thousands processes per day), therefore binary (vs textual) representation is required and usually once per day the files are compressed, keeping only average values.

As far Digital Forensics concerns this approach produces an interesting side effect, making hard to edit this files for normal users. Binary record cannot be modified by a usual editor and statistical data represents a sort of checksum of the current file. Moreover, these files are owned by the super user and normally have the read-only flag true. This does not mean that during a forensic analysis we should trust the content of these files but it is possible to verify (with several tool) the integrity of such files and in this case they should be considered meaningful.

Substantially, since it is hard for a non-skilled person to modify the logon informations stored by the OS, the only way to guarantee that the automation

tool do not leave suspicious traces is to simulate the typical access pattern of the user. For example, it can be be suspicious if at the day of the crime the logon time will be very different from the previous average logon times.

There exists some simple tricks that the criminal can use to automatically login and logout at his system. For example, the most of BIOSes can be programmed in order to automatically turn on the computer at a specific time. Regarding the logout action, the automation script can be programmed to send a shutdown signal to the OS.

Detection of the evidences — già detto? —

Discuss now what is the digital evidence that the automation process itself leaves.

Non ho capito tanto
Magari metterlo dopo?

Detecting traces eventually leaved by the automation, discussed in 3.1 and 3.1, is in practice not easy. There exists some digital investigation methodologies that can accomplish this task.

- *System monitoring*: it is possible to monitor the system processes in order to detect anomalies. However, it requires that a "monitor process" has been secretly installed by the investigators on the suspected's system before he run the automation program.
- *Live forensic tools*: it is possible to use Live forensic tools in order to recognize the value of the volatile data of the running system, for example, the informations about recently executed processes contained into the RAM. However, this approach requires that the system has never been turned off before the analysis.
- *Traditional digital forensic tools*: this methodology consists of analyzing the system's drives in order to detect traces leaved by the execution of the automation program or its development and testing. —- che possono fare?

3.2 Removing unwanted Digital Evidence of automation

As discussed in 3.1 and 3.1, the development/testing and execution of the automated program leave several unwanted evidences. Evidence of automation created by the executable can be removed/avoided employing three different approaches:

- Manual deletion.
- Semi-automatic method.
- Automatic method.

Manual deletion. The criminal can manually delete the unwanted evidences from the system. He must exactly know which traces about the executable of the

automation the OS records. He can disable some OS specific features, as the prefetch and the virtual memory allocation, in order to minimize data stored about the automation on the filesystem. Finally, he must delete the files constituting the automation executable itself using a kind of wiping technique **??**.

Semi-automatic method. It is possible to further minimize the unwanted data that will be stored on the drive running the automation executable from a removable device (e.g. an USB pendrive, a CD-ROM and so on). Using this approach, the criminal do not have to wipe the executable file(s) of the automation from the drive. However, he must also remove all suspicious evidences recorder by the OS about its execution. Moreover, the removable device containing the automation program is itself an unwanted evidence and must be obfuscated to the investigators.

Automatic method. The process of unwanted evidences deletion can be itself part of the automation program. It requires that the criminal is skilled enough to create a shell script that firstly runs the automation program, then deletes all unwanted traces about its execution recorded by the OS, and eventually wipes itself. In this work a case study 4 is hereinafter discussed which makes use of this technique.

3.3 Additional cautions

A recent paper [12] explain how it is possible to recognize the persons who have used a computer analyzing the bacteria left by their fingertips on the keyboard and mouse. The imprint left by the bacteria on the keys and mouse persists for more than two weeks. This is potentially a new tool for forensic investigation. Obviously, investigators should use gloves before examining the computer. This kind of analysis can be exploited by a criminal to validate his digital alibi. If the suspected made sure of being the only one to use the computer, the defender Advocate can request a forensic analysis within two weeks, which will confirm that bacterial traces on the keyboard and mouse are those of the suspect.

People have their habits and then follow a predictable pattern. For example, it may be usual for the suspect to connect to the Internet during the morning, access his mailbox, browse some websites and work on his thesis. In practice, the behavior of the suspected inferred from his digital alibi must be not very different from his typical behavior. That is, suspicious traces must not be discovered by an hypothetical Anomaly Detection [20] analysis. The connection time, the amount of Transmitted and Received bytes, the amount of access to social networks, and other actions must be similar to those of the previous days according to the habits of the accused. The same behavior inferred from the digital evidences may be repeated on other days with some randomization. The testing phase of the automation 3.1 can already give regularly to the behavioral pattern of the suspect and therefore may be useful in order to guard against eventual anomaly detection analysis [34].

4 Case Study

Strumenti utilizzati

Realizzazione, sperimentazione ed analisi

Microsoft Windows contains significant amounts of digital evidence that enables an investigator to reconstruct events that took place on the machine before it was seized. The Windows Registry in particular contains a wealth of information about the configuration and use of a computer. [33]

4.1 AutoIt

AutoIt is a freeware automation language for Microsoft Windows. The syntax of AutoIt is like the BASIC family of languages. An AutoIt automation script can be compiled into a compressed, stand-alone executable which can be run on computers that do not have the AutoIt interpreter installed.

...

The function

Sleep (delay)

Determines a pause of a number of milliseconds equal to the delay.

5 Digital Evidence as an Alibi

Discuss the issues related to the use of digital evidence as an alibi in digital forensics investigations.

The fact that it is possible to construct a digital alibi using automation tools, as shown in this paper, does not imply that every digital alibi have no probative values.

Observe that a digital alibi can be constructed also with the help of an accomplisher.

The same point is true also in the physical world, non only in the digital one. There have been various cases in which a false testimony have provided a "physical" alibi to the accused person. Serve qualche altro esempio ...

The fact that malicious people can construct a fake alibi has not invalidated his probative value.

... elaborare anche su [30] [33]

...

6 Conclusions

Computers are becoming more and more important in our society. People use PCs to accomplish a large set of activities, related to their work or personal purposes. A PC can contain a lot of informations about the person(s) which use it, for example personal data such as logon times, used applications, visited websites and so on. As a result, the number of court cases involving digital evidence is increasing.

Defense attorneys are becoming more knowledgeable about computer forensics and digital evidence, ... and possibly increased suppression of evidence. — che si vuole dire?

The discipline of Computer Forensics cannot survive for long if it relies on the lack of technical and scientific understanding by the courts.

Cum grano salis — che si vuole dire ?

Parallel to the evolution of digital forensics techniques, tools and methodologies to ... are becoming more dangerous.

Digital evidence is probably the strongest when it can be shown to be part of a larger pattern of behavior.
Alcuni spunti da inserire nelle conclusioni sono i seguenti: ...

- i giudici imparino a dare il giusto peso alle evidenze digitali;
- i tecnici acquisiscano le competenze necessarie per la corretta raccolta delle evidenze digitali;
- non si faccia affidamento esclusivo sulle evidenze digitali al fine del raggiungimento del verdetto finale e tali prove digitali non sostituiscano le prove "classiche".

Ringraziamenti

Qui vanno i ringraziamenti ...

References

1. *Internet world stats*, June 30, 2010, http://www.internetworldstats.com/stats.htm
2. Msnbc News, *Facebook message frees NYC robbery suspect*, November 12, 2009. http://www.msnbc.msn.com/id/33883605/ns/technology_and_science-tech_and_gadgets/
3. The New York Times, *I'm Innocent. Just Check My Status on Facebook*, November 12, 2009. http://www.nytimes.com/2009/11/12/nyregion/12facebook.html?_r=1
4. CNN, *Facebook status update provides alibi*, November 12, 2009. http://www.cnn.com/2009/CRIME/11/12/facebook.alibi/index.html
5. Stefano Vitelli, GUP presso il Tribunale di Vigevano, *Sentenza del processo Stasi*, http://static.repubblica.it/laprovinciapavese/pdf/SENTENZA_STASI.pdf, 17 Dicembre 2009
6. Fabio Bravo, *La computer forensics nelle motivazioni della sentenza sull'omicidio di Garlasco* http://internetsociety.wordpress.com/2010/03/16/la-computer-forensics-nelle-motivazioni-della-sentenza-sullomicidio-di-garlasco/, 16 Marzo 2010
7. Japan Electronic Industries Development Association (JEIDA), *Exchangeable Image File Format*, http://en.wikipedia.org/wiki/Exchangeable_image_file_format
8. U.S. Department of Defense, *DoD Directive 5220.22, National Industrial Security Program (NISP)*, http://www.dtic.mil/whs/directives/corres/html/522022m.htm, 28 February, 2010

9. Peter Gutmann, *Secure Deletion of Data from Magnetic and Solid-State Memory*, Sixth USENIX Security Symposium Proceedings, San Jose, California, http://www.cs.auckland.ac.nz/ pgut001/pubs/secure_del.html , July 22-25, 1996.
10. Peter Gutmann, *Data Remanence in Semiconductor Devices*, 2001 Usenix Security Symposium, Washington DC, http://www.cypherpunks.to/ peter/usenix01.pdf , August 13-17, 2001.
11. US NIST, *Guidelines for Media Sanitization*, NIST Special Publication 800-88, September 2006. http://csrc.nist.gov/publications/nistpubs/800-88/NISTSP800-88_rev1.pdf ,
12. Noah Fierer, Christian L. Lauber, Nick Zhou, Daniel McDonald, Elizabeth K. Costello, and Rob Knight, *Forensic identification using skin bacterial communities*, Proceedings of the National Academy of Sciences, Abstract, http://www.pnas.org/content/early/2010/03/01/1000162107.abstract , March, 2010.
13. AutoHotKey website, *AutoHotKey*, http://www.autohotkey.com/ , March 2010.
14. Jonathan Bennett, *AutoIt v3.3.6.0*, http://www.autoitscript.com/autoit3/ , March 7, 2010.
15. Microsoft Corporation MSDN, *Windows Script Host*, http://msdn.microsoft.com/en-us/library/9bbdkx3k(VS.85).aspx , 2010.
16. Radical Breeze, *DoThisNow*, http://radicalbreeze.com/?page_id=50 , 2010.
17. Henrik Sandklef, *GNU Xnee*, http://www.sandklef.com/xnee/ , 2010.
18. Apple Inc., *Apple Automator*, http://www.macosxautomation.com/automator/ , 2010.
19. DropBox Developer Team, *DropBox*, http://www.dropbox.com/ , 2010.
20. F. Maggi, S. Zanero, and V. Iozzo, *Seeing the Invisible - Forensic Uses of Anomaly Detection and Machine Learning*, ACM Operating Systems Review, vol. 42, no. 3, pp. 52-59, April 2008.
21. A. Castiglione, A. De Santis and C. Soriente, *Taking advantage of a disadvantage: digital forensics and steganography using document metadata*, Journal of Systems and Software, Elsevier 80 (5), pp. 750-764, May 2007.
22. A. Castiglione, A. De Santis and C. Soriente, *Security and Privacy Issues in the Portable Document Format*, Journal of Systems and Software, Elsevier, Accepted Paper, April 2010.
23. R. A. Joyce, J. Powers, F. Adelstein, *MEGA: A tool for Mac OS X operating system and application forensics*, Journal of Digital Investigation, Elsevier, 5, pp. 83-90, 2008
24. 3GPP, *Timing Advance*, http://en.wikipedia.org/wiki/Timing_advance
25. M. Geiger, *Evaluating Commercial Counter-Forensic Tools*, http://www.dfrws.org/2005/proceedings/geiger_couterforensics.pdf
26. T. Bosschert, *Battling Anti-Forensics: Beating the U3 Stick*, Journal of Digital Forensic Practice, 1556-7346, Volume 1, Issue 4, pp. 265-273, 2006
27. A. Smith, *Describing and Categorizing Disk-Avoiding Anti-Forensics Tools*, Journal of Digital Forensic Practice, 1556-7346, Volume 1, Issue 4, pp. 309-313, 2006
28. G. Fellows, *WinRAR Temporary Folder Artefacts*, Journal of Digital Investigation, Elsevier, article in press, March 2010
29. D.-Y. Kao, S.-J. Wang and F. Fu-Yuan Huang, *SoTE: Strategy of Triple-E on solving Trojan defense in Cyber-crime cases*, Journal of Computer Law and Security Review, Elsevier, Volume 26, Issue 1, pp. 52-60, January 2010
30. Caloyannides, M. A.; *Forensics Is So "Yesterday"*, IEEE Security & Privacy, March-April 2009, vol. 7, Issue: 2, pp. 18 - 25

31. Carrier, B.D.; *Digital Forensics Works*, IEEE Security & Privacy, March-April 2009 vol. 7, Issue: 2, pp. 26 - 29
32. Thomas Myer; Apple[unkch] Automator with AppleScript Bible, Wiley Publishing, Inc., 2010
33. Vivienne Mee, Theodore Tryfonas, Iain Sutherland; *The Windows Registry as a forensic artefact: Illustrating evidence collection for Internet usage*, Digital Investigation, vol. 3, pp. 166 [unkch] 173, 2006
34. Varun Chandola, Arindam Banerjee, Vipin Kumar; *Anomaly detection: A survey*, ACM Computing Surveys vol. 41, n. 3, July 2009
35. Honorable Donald E. Shelton; *The 'CSI Effect': Does It Really Exist?*, National Institute of Justice Journal No. 259, March 17, 2008