The ISC Int'l Journal of Information Security

ISeCure

July 2012, Volume 4, Number 2 (pp. 137–150)

http://www.isecure-journal.org

Analyzing Registry, Log Files, and Prefetch Files in Finding Digital Evidence in Graphic Design Applications

Enos K. Mabuto [1,*] and Hein S. Venter [1]

[1] *Department of Computer Science, University of Pretoria, Pretoria, 0002, South Africa*

ARTICLE INFO.

Article history:
Received: 20 November 2011
Revised: 08 March 2013
Accepted: 12 March 2013
Published Online: 10 June 2013

Keywords:
Digital Evidence, Digital Forensics, Digital Forensic Artifacts, Graphic Design Applications.

ABSTRACT

The products of graphic design applications, leave behind traces of digital information which can be used during a digital forensic investigation in cases where counterfeit documents have been created. This paper analyzes the digital forensics involved in the creation of counterfeit documents. This is achieved by first recognizing the digital forensic artifacts left behind from the use of graphic design applications, and then analyzing the files associated with these applications. When analyzing digital forensic artifacts generated by an application, the specific focus is on determining whether the graphic design application was installed, whether the application was used, and determining whether an association can be made between the application's actions and such a digital crime. This is accomplished by locating such information from the registry, log files and prefetch files. The file analysis involves analyzing files associated with these applications for file signatures and metadata. In the end it becomes possible to determine if a system has been used for creating counterfeit documents or not.

1 Introduction

Industries including but not limited to, advertising, newspaper printing, architecture, fashion and design, project management and manufacturing make use of graphic designs for their corporations. Graphic design applications have enhancing tools like paint brushing, vector drawing, digital pen and pencil drawing and many more. These graphic design applications are used to facilitate creating unique art for company logos, magazine advertising or computer-aided design, to mention only a few. Most industries make use of graphic design applications for visual presentations using pictorial expressions that aid communication and expressing of ideas.

The use of forged documents, however, is noticed all over the world. A report by Ilham Rawoot of the Mail and Guardian newspaper stated that terrorists target fake South African passports because of the ease with which one can be faked [15]. A similar report from the International Business times was also reported [31]. These reports show that counterfeit documents are in circulation all over the world. The same graphic design applications used in the industry today can also be used for illegitimate purposes like creating counterfeit documents. The problem is that, with the editing and design capabilities of these graphic design applications, they can be used to create counterfeit documents like ID's, passports or drivers licenses. Criminal activities such as these necessitate need for digital forensic investigations.

* Corresponding author.
Email addresses: nasbutos@yahoo.co.uk (E. K. Mabuto), hsventer@cs.up.ac.za (H. S. Venter).
ISSN: 2008-2045

The use of graphic design applications leaves behind traces that can be revealed during a digital forensic investigation. This paper identifies the digital traces left behind after using graphic design applications. In addition, a file analysis of files associated with these applications is conducted. To address the problem, the authors focus on the following three steps. First, the digital forensic information that shows whether the specific graphic design application was installed is identified. The second step entails querying whether the application was actually used for document editing. Lastly, it is determined whether an association can be made between the application's actions and such a digital crime. In so doing, an association with the potential criminal may be achieved. However, it is not the focus of this paper to link the crime to an actual person. After gathering the traces left behind, the authors focus on an analysis of files associated with these applications. This involves determining the file signatures and recognizing the metadata related to these files.

The remainder of the paper is structured as follows. In the second section, some background of digital forensics is given, followed by a brief background on graphic design applications. The third section, which is the contributing section, is divided into three parts. The first part highlights the potential evidence which the authors refer to as digital forensic artifacts. Digital forensic artifacts can be found in graphic design applications where the source of the evidence is mainly system-generated. The source of potential evidence referred to above equates to the results of the registry analysis, application log file analysis and system prefetch file analysis. The second part is an examination of user-generated files and a highlight of the potential evidence. The source of potential evidence referred to being results from content identification and content examination of files utilized by graphic design applications. The authors also name the tools that can be used in aiding the analysis where applicable. The last part of the third section is a methodological description of how to acquire the evidence contained in the paper. The fourth section is an evaluation of the evidence that is extracted from the graphic design applications. Lastly a conclusion is given to end this paper.

2 Background

In the following section the authors provide some brief background literature on digital forensics including an explanation of digital evidence. A definition of digital forensic artifacts and a discussion on image forensics is found thereafter. The second section of the background consists of a very brief literature survey on graphic design applications.

2.1 Digital Forensics

At the Digital Forensics Research Workshop (DFRWS) in 2001, digital forensics was defined as the use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations [11]. To reconstruct and understand what has happened in the past on a system, data has to be gathered and analyzed in a transparent manner.

A digital forensic investigator can use the digital forensic process which is made up of steps including acquisition, examination, analysis and reporting [12].

The goal of a digital forensic investigation on a system is to find out what happened and who was responsible for a particular incident or crime. Digital forensic investigations focus on finding digital evidence after a computer or network security incident has occurred or locating data from systems that may form part of some litigation, even if it has been deleted. In this context, evidence is the most critical in any case. Therefore any items that can be considered to be of evidential value should be identified and collected [6].

2.1.1 Digital Evidence

Computer evidence or digital evidence is defined as any hardware, software or any data that can be used to prove one or more of the "who, what, when, where, why and how" of a security incident [2]. Computer evidence furthermore consists of digital files and their contents left behind after an incident. Casey defined digital evidence as any data that can be used to establish that a crime was committed or can prove a link between a crime and its victim or an offender [3]. Digital evidence consists entirely of sequences of binary values called bits [7]. It is important to note, however, that the evidence should be presented in its logical form in court or disciplinary hearing [8, 23].

When investigating crime related to the use of an application the first question would typically be whether the particular application was installed, then whether the application was used and, lastly, whether there is any relationship between the actions of the application and the computer crime or incident being investigated. In responding to these queries, one or more of the "who, what, when, where, why and how" questions usually asked about a security incident has to

be proven. Traces that are left behind from the use of an application or from an operating system can be referred to as digital forensic artifacts.

2.1.2 Digital Forensic Artifacts

An examiner reveals the truth of an event by discovering and exposing the remnants of the event that have been left on the system. These remnants are known as artifacts, which can be referred to as digital evidence [22]. However, due to the loaded legal connotations binding the term "evidence" the term "artifacts" is used instead. Evidence is referred to as something to be used during a legal proceeding. Artifacts are traces left behind due to activities and events, which may or may not be innocuous. Trying to remove these artifacts leaves other artifacts. For example, in trying to remove log files from a system one has to use a removal tool, thus leaving additional traces that indicate that a log removal tool was used. The scattered evidence inside a system can indicate what has happened for a particular digital forensic investigation.

Application artifacts left by installed applications can be an excellent source of potential evidence when performing an analysis. An artifact does not become evidence unless its ability to prove a fact has been established [9]. Therefore it is necessary to reconstruct events that occurred by gathering all the possible digital information from a system.

The work covered in this paper continues from previously-published work by the authors on "User-generated digital forensic evidence from graphic design applications" [28]. The mentioned paper elaborates on gathering potential evidence on the actual files with counterfeit value created by the counterfeiter intentionally. The potential evidence referred is described by use of evidence identifiers such tags and prefixes that embed the evidence.

As opposed to the previous paper [28], the focus of this paper is on the files generated by the graphic design application itself, mostly for the purpose of metadata that would hold potential evidence. Several file types are then compared with regards to the type of metadata they contain. Furthermore this paper describes how the identified artifacts can be linked to identify counterfeiting.

2.1.3 Image Forensics

The amount of research and development that has been undertaken in this field has not, to date, focused on the skills and of graphic design software, which is a particular area that is nearly always exploited for the purpose of creating counterfeit documents and images. Most research work that has been undertaken up till now has concentrated on image forensics, which is the kind of investigation that is able to determine whether or not an image as been forged or tempered [32, 33].

Lien [32], proposed a method that uses a pre-calculated resampling weighting table to detect periodic properties in error distribution within an image. The errors in the distribution within an image are used to determine if the image has been forged. Stamm [33] proposed a method to detect contrast enhancement and addition of noise in *jpeg* compression images. Changes in contrast and noise within an image are determined through the use of an algorithm that calculates pixel values within the image. The values are then used to detect forgery within the image. Cohen [34] proposed a method that determines characteristics associated within digital still camera images to determine the origin of the image. The characteristics are compared to the exact replicas and derivates of other statistical images to detect forgery.

These, [32–34], and other related work focus on determining forgery using statistical data within the image [35–38].

Very little of the research carried out to date has specifically investigated the ways and means in which documents are counterfeited. These ways also include the methods and procedures that can be used to detect such activities from graphic design applications, which is the focus of this paper.

In an investigation, how and where evidence is located differs depending on the crime being investigated, the platform (operating systems) and the application used to commit the crime.

2.2 Graphic Design Applications

Many graphic design applications are currently available in the industry; however, Adobe Systems Incorporated is regarded as the largest software maker in the graphic design software category [1, 5]. Adobe Systems Incorporated owns software technologies that are used for online transactions, business applications and social technologies [10].

Therefore, for this research, a case study was conducted with Adobe graphic design applications. The following are Adobe applications used for graphic design purposes.

2.2.1 Adobe Acrobat

Adobe Acrobat is an application used for viewing, creating, manipulating, printing and managing files in the portable document format (PDF). PDF files are usually read-only documents that cannot be altered without leaving an electronic footprint [19].

2.2.2 Adobe Photoshop

Adobe Photoshop is a professional industry-standard application for digital image editing and creation. Adobe Photoshop has an interactive platform to change the picture format, join pictures, split pictures, and change the color and appearance of photos among the many features it can offer.

2.2.3 Adobe In-Design

Adobe In-Design is a professional layout and design application that delivers production workflows, complex graphics and typography. Adobe In-Design is also used for designing magazines, printing page layouts and facilitating digital distribution using built in creative typography tools, to name a few.

2.2.4 Adobe Illustrator

Adobe Illustrator is an application used for vector artwork in planning projects. It has drawing tools and brushes that can be of use in designing graphic art consisting of rigid shapes and various line drawings, to mention only a few.

Any one of these applications can be used for document editing. Therefore it is necessary to conduct an exclusive examination for potential digital forensic evidence.

3 Digital forensic evidence in graphic

Various experiments were carried out in order to search for pertinent evidence in graphic design applications. Experiments were conducted on Adobe applications capable of graphic designing namely Adobe Acrobat, Adobe Photoshop, Adobe In-Design and Adobe Illustrator. The experiments were conducted in two parts. The first part highlights digital forensic artifacts found in graphic design applications where the source of the potential evidence is mainly system-generated with results mostly from registry analysis, application log file analysis and system prefetch file analysis. The second part of the experiments, which involves examination of user-generated files, highlights results from file content identification and examination.

Early 2011 software reviews revealed that the Windows operating system is still ranked the most popular operating system [4, 10, 13]. Therefore, the analysis for forensic artifacts was conducted on a Windows 7 platform.

To respond to the problem stated earlier, that graphic design applications can be used for creating counterfeit documents, firstly three techniques are used to gather digital forensic information related to graphic design applications. These techniques are the registry analysis, application log file analysis and system prefetch file analysis. From the experiments conducted it was recognized that an offender can deny any of the following; running the application, installing the application or using the application for counterfeiting. Therefore the analysis is formulated by asking three questions for each of the techniques listed above. The first question is can one identify digital forensic evidence that shows that the application was installed? Secondly, the question is asked, was the application actually used for document editing? The third question determines whether there is an association between the application action's and the alleged digital incident or crime. By following these queries an investigator is able to conduct an investigation in a uniform manner. For example, if the application was not installed, then there is no need to ascertain whether the application was used. Furthermore to respond to the same problem, a user-generated file analysis section follows, with two sub-sections dealing with content identification and content examination respectively. A summary of results is tabulated at the end of the section.

Experimental results gleaned from asking the three questions about registry analysis, application log file analysis and system prefetch file analysis are applied to each of the subsections to follow.

3.1 System-generated digital forensic artifacts from graphic design applications

"System generated digital forensic artifacts" refers to those artifacts created by the application without user intervention, while "user generated digital forensic artifacts" refers to artifacts created by the user intentionally.

For the experiments conducted, the following section describes the techniques used on Adobe graphic design applications. Three sub-sections follow in this section, namely registry analysis, application log file analysis and system prefetch file analysis.

3.1.1 Registry Analysis

The Windows registry is a collection of data files that stores vital configuration data for the system including user activity [16]. The Windows registry contains a plethora of valuable information including, user activity history, system configurations and information about installed applications. Potentially all the registry information can be of use to an analyst attempting to establish a timeline of activity on a system. Registry information is organized in the form of key entries. Registry information retrieved from different

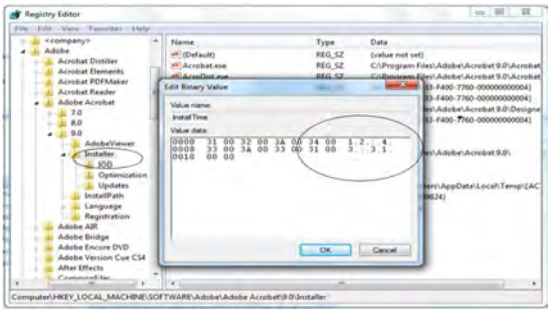

Figure 1. Registry view of Acrobat installation time

keys can be correlated for a better understanding. Besides the default regedit tool available in Windows systems, other tools that can be used to analyze the registry are Registry Lite [17] and Registry Viewer [18].

An in-depth search was executed for keys associated with graphic design applications. It can also be noted that a single registry key can reveal more than one value. In establishing whether the application was installed, registry keys containing values for application settings, the installation time, installation date and the installation path for Adobe Acrobat can be obtained from key, `HKEY_CURRENT_USER\Software\Adobe\Adobe Acrobat\9.0\Installer` as shown in Figure 1 and for Adobe Photoshop it can be obtained from key `HKEY_LOCAL_MACHINE\SOFTWARE\Adobe\Photoshop\11.0\ApplicationPath`. Thus, if these keys are found in the registry, it answers the first question that the application was installed.

To query whether the application was actually used for document editing, values for the visited directories are acquired from the registry key, `HKEY_CURRENT_USER\Software\Adobe\Photoshop\11.0\VisitedDirs` and values for the home path of the application, as well as the name of the computer used to login (titled login server in registry) are obtained from the key, `HKEY_USERS\<user-id>\Volatile Environment`. These registry entries answer the second query that the application was actually used for document editing.

To query whether there is an association between the application's actions and a particular digital crime registry keys were obtained with values indicating the following: who used the application, the email address, the name of the department, the domain name and the name of the corporation. All these values are obtained from registry key `HKEY_CURRENT_USER\Software\Adobe\Adobe Acrobat\9.0\Identity` and similar values as above from `HKEY_CURRENT_USER\Software\Adobe\Adobe Acrobat\9.0\Security\cMain`.

The registry keys contain a last used directory, which is created when the application is used. This establishes that the application was actually used to create a document.

In general, when a registry key is deleted, much like a file, it really does not disappear. In actual fact, when it is deleted, the size value is changed to a positive value [24]. In 2008, Jolanta Thomassen released a perl script known as "regslack" which uses this property to parse through a hive file which is the hierarchical file structure and retrieve deleted keys. It, therefore, comes to our attention that when an offender has deleted these keys a digital forensic investigator is able to retrieve the keys.

3.1.2 Application Log File Analysis

Application log files are files related to events from a particular application. Besides these, Windows also maintains system log files of events and actions that can be essential to an investigation. System log files contain important information about recently viewed documents, saved data, personal user information and other temporary data files. The focus of this paper is on log files created by the graphic design applications in question. Winhex from XWays [20] is used as the hex editor for analyzing data files, but any other hex editor can also be used.

In establishing whether the application was installed (the first query), a folder is created in the following path `C:\Users\<user>\AppData\Roaming\Adobe\`. The time stamp on the folder denotes the date of installation. It should be noted that the `AppData` folder is hidden by default.

To query whether the application was actually used for document editing (the second query), a history of viewed documents, history of file searches and other temporary files are obtained from the following location `C:\Users\<user>\AppData\Local\Microsoft\Windows\History\<week or day>\Computer`. This location contains the actual files saved after document editing, for example a `*.psd` file saved from Adobe Photoshop.

To query whether an association exists between the application's actions and a digital crime a log file titled *InDesignSavedData* in the location `C:\Users\<user>\AppData\Local\Adobe\InDesign\Version 6.0\en_GB\Caches` provides an answer. The file contains data indicating which actions were taken during document editing like alignment, clearing text, moving an object, joining, importing files all starting at hex offset C544 and the location of any imported files at hex offset 10D7F7 as illustrated in Figure 2. An imported file can be any file for example a fingerprint photo attached to the

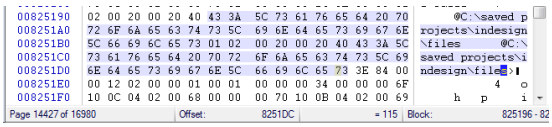

Figure 2. Log file hex editor extract indicating location of imported files and an alignment action

file being created. Figure 2 illustrates the file location of imported files used during document editing and alignment actions carried out as an example. The location `C:\Users\<user>\AppData\Roaming\Adobe\Acrobat\9.0\Security` contains a file titled shared data events, which shows that a digital signature was created with values supplied for email, department, corporation and name of user. This information helps a digital forensic investigator to establish a possible link to the criminal.

To further explain Figure 2; the first column is the byte count, also known as the byte offset, in base sixteen (hex). The proceeding paired columns are the hexadecimal representation of the file content. Each column represents two bytes. The last column on the far right represents the ASCII text rendition of the file. Non printable or non ASCII characters are displayed as dots as seen in the last column.

Also the location `C:\Users\<user>\AppData\Roaming\Adobe\Adobe Illustrator CS4 Settings\` contains a *ins* file extension titled *Recently used optimizations* which contains the format last used for document editing and the previous changes made to file type. The location `C:\Users\<user>\AppData\Roaming\Adobe\Adobe Photoshop CS4\Adobe Photoshop CS4 Settings\` contains a `Actionspallette.psp` file containing information relating to saving actions that took place during document editing. It also contains information about the file extension used to save documents and any messages displayed while saving. The `Prefs.psp` file in the same location contains objects used for editing like brushes used, shapes used, and the recent file location at offset 31AFE.

Application log files record information about the documents created, their names and if any images or objects have been used to create these documents. The list of the created documents can then be used to search for the potential counterfeit documents created. Furthermore the inserted images can be used to identify if indeed the created document is counterfeit or not. By analysing them individually thereby determining if it's a human face, fingerprint or barcode that was inserted into the document.

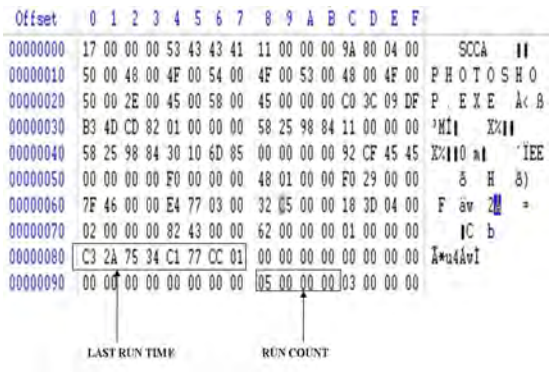

Figure 3. Hex editor extract of Adobe Photoshop prefetch file

3.1.3 System Prefetch File Analysis

Prefetching was developed to improve the systems performance [14]. The purpose of prefetching is to allow regularly used applications to load faster by prestaging segments of loaded code in a specific location so that instead of searching for it (resulting in page faults), the operating system knows exactly where it is. It means when an analyst finds a prefetch file for a particular application, it indicates that the particular application was indeed run on the system. The creation date of that file will indicate when the application was first run, although assuming that a previous prefetch file wasn't deleted and a new one created in its place. This is because prefetch files are actually temporary files that can be deleted or overwritten by the operating system at any time. The prefetch file contains a 64 bit time stamp indicating when it was last run, as well as a count of how many times it was run. On Windows 7, the 64 bit last run time stamp is at offset 80 (128 bytes) within the binary contents of the prefetch file and the run count 4 bytes at offset 98 (156 bytes) as illustrated in Figure 3.

Once the data is processed, it is written to a `*.pf` file in the systems prefetch directory. The `*.pf` file will be referenced later when the program is run again. The file name is created using the application's name followed by a dash and then by a hexadecimal representation of the hash of the path of the application for example `ACROBAT_SL.EXE DC4293F2.pf`. That means the same program run from different locations will create different `.pf` files. In this way, the next time an application is launched, the prefetch directory is checked for a prefetch file. If it exists, the code within the `*.pf` file is used to launch the application. If, however, the prefetch file is not present the application will still be launched but will load slowly.

Prefetch files are located in the folder: `%systemroot%prefetch`. It should also be noted that one needs administrative privileges to access the prefetch folder. Within the prefetch file are values that correspond to the number of times the application was launched

Table 1. Adobe prefetch files

Application Name	File Name
Adobe Acrobat	ACROBAT_SL.EXE DC4293F2.pf
Adobe Distributor	ACRODIST.EXE 1C2D8F2D.pf
Adobe Reader	ACRORD32.EXE DE3ACC1.pf
Adobe Collaboration	ADOBECOLLABSYNC.EXE 621E7FA.pf
Adobe Updater	ADOBEUPDATER.EXE 9AAD898.pf
Adobe Service Manager	CSXSERVICEMANAGER.EXE B80CD935.pf
Adobe Indesign	INDESIGN.EXE C8D4FD6C.pf
Adobe Tray	VERSIONCUECS4TRAY.EXE D4DE4E1A.pf
Adobe Photoshop	PHOTOSHOP.EXE 4545CF92.pf

and a value containing the last time the application was launched. This information is obtained from analyzing the prefetch file with a hex editor as illustrated in Figure 3.

Therefore, prefetch files establish that the application was installed and that the application was used indicated by last run time and run count respectively. However, there is no established relationship between the application's actions and the digital crime in the prefetch files but the information found can be correlated to information gathered from the registry and log files. The operating system generates several different prefetch files. It is necessary for an investigator to know all prefetch files generated, for in some cases the name of the prefetch file will not be similar to the name of the application. Table 1 shows Adobe prefetch files that are obtained from `%systemroot%prefetch`.

It should also be noted that any deleted log files or prefetch files could be recovered using any popular forensic tool like FTK and Encase.

3.2 User generated artifacts from file examination

In order to conduct an exclusive examination on a crime conducted from an application the investigator has to understand the nature of the files that are generated from that particular application, in this case, graphic design applications. This is so that digital forensic examiners are able to uncover and exploit any digital forensic artifacts present in the identified files [22].

As previously stated, user-generated digital forensic artifacts refers to files created by the user intentionally. User generated file artifacts are divided into two distinct categories, which are, content identification and content examination. Content identification is the process of determining or verifying the type of a specific file. Content examination is the retrieval of any embedded metadata that may be present in a given file.

In the case of the examination of counterfeit documents the digital forensic investigator might need to identify potential changes inside files consistently, for example, the involvement of a fingerprints, barcodes or human faces embedded inside graphic design application file formats. The four graphic design applications discussed above are associated with more than thirty nine file types. However, for this research the authors focus was only on file types that are specific to the four graphic design applications, thus ignoring well-known file types like *jpeg*, *bitmap*, *tag*, *tiff*, *tga*, *etc*. Gary Kesler and Martin Reddy keep a list of these common file signatures online, which is a continuing work in progress database [25, 26]. An online metadata extraction tool is also available for extracting metadata from these common file types [30].

3.2.1 Content Identification

As previously stated content identification involves verifying the identity of a file extension. An offender can alter the file extension of a particular file in order to promote ambiguity. Therefore there is need to identify a files integrity by file signature analysis. An investigator needs to know what a particular file type is. A file is normally analyzed within its first bytes to determine the specific signature [14]. The file signature is therefore located at specific offsets usually in the beginning of a file.

It can be noted from the research conducted that known digital forensic tools like FTK can detect various file types but not for graphic design applications discussed in this paper. For example, digital forensic tools can verify file types like *tga*, *bmp*, *gif*, *tif*, and *png* amongst others, but not the file types of graphic design applications as discussed in this paper.

The analysis to determine a graphic design file signature was also conducted using a hex editor. These values are generally hexadecimal values. Table 2 contains the list of file signatures identified and specific to the graphic design applications previously discussed. The file type in Table 2 represents the named form of the particular graphic design file. Proof of the real file content resides within the content of the file, usually known as the file signature. The file extension is merely a suffix that represents the encoding of a file's content, usually three or four characters separated by a dot from the file name. However, the file extension should never be trusted as it can be renamed to anything else. One should rather focus on the file signature to determine the correct file type. The ASCII column in Table 2 represents the entry in text-readable format.

Table 2. Graphic design file signatures

File type	File extension	ASCII	File signature
In-design	indd	íõØFå½1ïçþt·DOCUMENTp	06 06 ED F5 D8 4D 46 E5 BD 31 EF E7 FE 74 B7 1D 44 4F 43 55 4D 45 4E 54 01 70 0F
In-design XML Interchange document	incx	<?xml version="1.0" encoding="UTF-8" standalone="yes"?>	3C 3F 78 6D 6C 20 76 65 72 73 69 6F 6E 3D 22 31 2E 30 22 20 65 6E 63 6F 64 69 6E 67 3D 22 55 54 46 2D 38 22 20 73 74 61 6E 64 61 6C 6F 6E 65 3D 22 79 65 73 22 3F 3E
In-design template	indt	íõØFå½1ïçþt·DOCUMENTp	06 06 ED F5 D8 4D 46 E5 BD 31 EF E7 FE 74 B7 1D 44 4F 43 55 4D 45 4E 54 01 70 0F
Photoshop	psd	8BPS	38 42 50 53 00 01
Illustrator file	ai	%PDF-1.5% âãÏÓ1 0 obj	25 50 44 46 2D 31 2E 35 0D 25 E2 E3 CF D3 0D 0A 31 20 30 20 6F 62 6A
Illustrator template	ait	%PDF-1.5% âãÏÓ1 0 obj	25 50 44 46 2D 31 2E 35 0D 25 E2 E3 CF D3 0D 0A 31 20 30 20 6F 62 6A
Encapsulated post script	eps	ÅÐÓÆ	C5 D0 D3 C6

The file signature columns represent the entry in hexadecimal format. Both these entries appear exactly as shown in the hex editor. The digital forensic examiner can use the information in Table 2 to identify the particular files for the graphic design applications in question.

3.2.2 Content Examination

Content examination involves determining the metadata of files, in this case, graphic design application file types. Metadata refers to data about data [22]. On Windows systems this includes modified, accessed and creation times .The same hex editors, as previously stated, are used to examine the content of files associated with graphic design applications. Metadata is essential during an investigation as this reveals what useful information can be extracted from a particular file, for example this can be time stamps or name of the user who created the file. Table 3 shows the metadata acquired from graphic design file types. The offset is the address pointer of the described metadata. In other words, if an investigator searched for a certain offset, the hex editor would skip to the particular metadata. However, several experiments reveal that the offset can slightly differ by plus or minus 780 bytes per metadata, which is usually in the same page view depending on the size of the file and quantity of metadata present in the file. Therefore the tabulated values can still be used on graphic design files of different sizes. The metadata is embedded in Extensible Metadata Platform (xmp) tags, which is Adobe's way of embedding metadata in its various file types [27].

3.3 Methods to gather digital evidence

Digital forensic investigators should be able to identify digital evidence from graphic design applications and interpret the evidence appropriately. In the subsections that follow, the authors describe a method to identify the evidence presented in this paper.

3.3.1 Examine the system

As discussed in section 3.1, an investigator has to recognize digital evidence from the system. This enables one to identify the particular graphic design application installed on a system using the registry and prefetch files. The identified graphic design applications can then be examined for log files embedded within the system. The log files are examined to recognize the documents that were created by that application. Recognizing the particular graphic design application also enables one to be able to recognize the file types associated with the application. The files types referred to in this case being user-generated artifacts discussed in this paper.

3.3.2 Examine file types

As discussed in section 3.2, an investigator next task would be to identify all the file types associated with the graphic design application. For example, *psd*, *indd*, *ait*, *inx* file types from Adobe graphic design applications. The identified file types are examined for file signatures as described in section 3.2.1 content identification. After the files signatures are noted, the examination continues to determine the contents of the graphic design file types as described in section 3.2.2.

Table 3: Graphic design file types related metadata

File Type	File Extension	Description of Metadata	Offset (Address Pointer to Metadata)	Example of the Metadata (As presented in a hex editor)
Indesign document	indd	File location for any imported image files	D9EB	file:C:/Users/<username>/Pictures/dvd%20picture%20sleeves/Capture_005%20%282%29.JPG
		Name of application that created the file	E510B or E6E16	<stEvt:softwareAgent>Adobe InDesign 6.0</stEvt:softwareAgent>
		String events of saving history	F0D0C to F12FE	<stEvt:action>created</stEvt:action> <stEvt:when>2011-05-04T15:13:25+02:00</stEvt:when>stEvt:action>saved</stEvt:action><stEvt:when>2011-05-04T15:15:43+02:00</stEvt:when>
		Date file was created	F5263	CreateDate>2011-05-04T15:13:25+02:00
		Metadata Date	F52A7	MetadataDate>2011-05-04T15:18:24+02:00</xmp:MetadataDate
		Modify Date	FD2EA	<xmp:ModifyDate>2011-05-04T15:18:24+02:00</xmp:ModifyDate>
Illustrator Postscript file	eps	Name of application that created the file	57	%%Creator: Adobe Illustrator(R) 14.0
		Date file was created	8E	%CreationDate: 9/17/2011
		Login name of user that created the file	73	%%For: <username>/ %
Illustrator file	ai	Metadata Date	3A7	<xmp:MetadataDate>2011-05-04T15:51:17+02:00</xmp:MetadataDate>
		Date file was modified	3ED	<xmp:ModifyDate>2011-05-04T15:51:17+02:00</xmp:ModifyDate>
		Date file was created	431	<xmp:CreateDate>2011-05-04T15:51:17+02:00</xmp:CreateDate>
		Name of application that created the file	476	<xmp:CreatorTool>Adobe Illustrator CSX</xmp:CreatorTool>
Photoshop file	psd	Name of application that created the file	1A9	<xmp:CreatorTool>Adobe Photoshop CSX Windows</xmp:CreatorTool>
		Date file was created	1F0	<xmp:CreateDate>2011-05-04T14:39:08+02:00</xmp:CreateDate>
		Date file was modified	234	<xmp:ModifyDate>2011-05-04T14:50:23+02:00</xmp:ModifyDate>
		Metadata date	27A	<xmp:MetadataDate>2011-05-04T14:50:23+02:00</xmp:MetadataDate>
		String events of saving history	6FF to 717	<stEvt:instanceID>xmp.iid:DE0657134D76E011B00EFDC555D228CB</stEvt:instanceID> <stEvt:when>2011-05-04T14:50:23+02:00</stEvt:when>
Illustrator template	ait	Name of application that created the file	1F3 or 452	<xmp:CreatorTool>Adobe Illustrator CSX</xmp:CreatorTool>
		Metadata Date	383	<xmp:MetadataDate>2011-05-04T15:51:17+02:00</xmp:MetadataDate>
		Date file was modified	3C9 or 16323	<xmp:ModifyDate>2011-05-04T15:51:17+02:00</xmp:ModifyDate>

Table 3: Graphic design file types related metadata (continued)

		Date file was created	40D	<xmp:CreateDate>2011-05-04T15:51:17+02:00</xmp:CreateDate>
		String events of saving history	D02B or D5D3	<stEvt:action>saved</stEvt:action><stEvt:instanceID>xmp.iid :FF7F117407206811B628E3BF27C8C41B</stEvt:instanceID> <stEvt:when>2011-05-22T16:23:53-07:00</stEvt:whe n>
		Name of user that created the file	17FB9	%%For: (Pinchers) ()
		File path for any imported images	D727	%%DocumentFiles:C:/Users/<username>/Pictures/Sizzla-Soul De ep-Front.jpg %%+C:/Users/<username>/Pictures/Tulips.jpg
		List of previous files names used	180A8	/Title(illustrator .ait template)
Indesign template file	indt	File path for any imported images	CF1E0 or D4F03	%%DocumentFiles:C:/Users/<username>/Pictures/Sizzla-Soul De ep-Front.jpg %%+C:/Users/<username>/Pictures/Tulips.jpg
		Date file was created	D72AB	<xmp:CreateDate>2011-05-04T15:17:21+02:00</xmp:CreateDate>
		Metadata Date	D72F1	<xmp:MetadataDate>2011-05-04T15:17:21+02:00</xmp:MetadataDa te>
		String events of saving history	D3DBA to D3F46	<stEvt:instanceID>xmp.iid:972E234B5076E011AAFBC6ED1F893037< /stEvt:instanceID> <stEvt:when>2011-05-04T15:17:21+02:00</stEvt:when>
		Name of application that created the file	D400C or D737C	<xmp:CreatorTool>Adobe InDesign 6.0>
Indesign interex-change file	incx	Date file was created	BFD3	<xmp:CreatorTool>Adobe InDesign 6.0</xmp:CreatorTool>
		Metadata Date	C019	<xmp:MetadataDate>2011-05-04T15:17:21+02:00</xmp:MetadataDa te>
		Date file was modified	C05F	<xmp:ModifyDate>2011-05-04T15:17:21+02:00</xmp:ModifyDate>
		Date file was created	BD3A	<xmp:CreateDate>2011-05-04T15:17:21+02:00</xmp:CreateDate>
		Name of application that created the file	C0A4	<xmp:CreatorTool>Adobe InDesign 6.0
		String events of saving history	108C2 or 115F7	<stEvt:instanceID>xmp.iid:972E234B5076E011AAFBC6ED1F893037< /stEvt:instanceID><stEvt:when>2011-05-04T15:17:21+02:00</st Evt:when>
		Last file path used	119D8 or 11C4d	%%DocumentFiles:C:/Users/<username>/Pictures/Sizzla-Soul De ep-Front.jpg %%+C:/Users/<username>/Pictures/Tulips.jpg
		Previous file format used	15BD2	<xmpGImg:format>JPEG</xmpGImg:format>

Table 4. Summary of gathered digital forensic artifacts

Technique	Query	Artifact Type	Details of Contents
Registry analysis	Installed	Key	Path, time, date
	Used	Key	Visited directory
	Link	Key	Epic name, server name
Log file analysis	Installed	Folder	Temporary files
	Used	Cache list	Saved data
	Link	File	Security policy name
Prefetch file analysis	Installed	File	Program name
	Used	File	Hash of path location

3.3.3 Co-relate the evidence

The final task for an investigator would be to identify the artifacts obtained from the system and from the file types. This includes determining the names of the counterfeit documents obtained from system generated artifacts. The names can then be searched for using any application or operating system. The last task would be to view these created documents using any image viewers or any application capable of viewing graphic images to visualize the products of graphic design applications. In the end an investigator would be able to tie the evidence and recognize if the documents produced are counterfeit or not.

3.4 Summary

The analysis for digital forensic artifacts can be summarized in Table 4. To briefly explain the table, only one technique is discussed in detail. The remainder of the table can be read in a similar fashion. From the second technique (Log file analysis) the query "was the application installed" (indicated by "installed" in the "Query" column in Table 4) comprised of an artifact consisting of a folder with temporary files created from the application. The query "was the application used" (indicated by "used") included a cache list consisting of saved data actions made during document editing. For the same technique the query of "establishing an association with the crime concerned" (indicated by "Link") reflected a file relating to a security policy file and the name of the user. The remainder of the results is self explanatory in Table 4.

For user-generated file analysis all graphic design application file types analyzed have timestamps as part of their metadata. However, only a few of them have the user name of the creator of the file as part of the metadata. Table 5 summarizes the user-generated file types. "Yes" indicates that the described metadata is present and "No" denotes that the file type does not contain the described metadata. The headings of the columns are brief names of descriptions of the metadata tabulated in Table 3.

4 Discussion

The objective of the paper is to determine if a system was used for counterfeiting. However, based on possible offender deniability the questions are formulated to respond to such circumstances.

If it is recognized that the application was not installed, it becomes possible that another computer system was used to create the documents. From analyzing the log files, such information can be derived from the counterfeit document itself, this is the log in name on the computer, which is obtained by analyzing the suspect counterfeit document illustrated in Table 2. This can lead to identifying the name of the system that the counterfeit documents were created on.

An application can be uninstalled after editing counterfeit documents. The registry entries illustrated in this paper are under normal circumstances left behind after installation and un-installation. If however the offender has used some tool or has manually deleted these entries, an investigator can use a tool called "regslack" [24], which is used to recover deleted registry entries.

Furthermore, other tools can be obtained to clean registry entries. It is thereby important to mention that the fight between forensics and anti-forensics is beyond the scope of this paper. The papers objective is to present work for digital forensic investigators to be able to find and interpret evidence related to document counterfeiting.

Recalling that computer evidence is defined as any hardware, software or any data that can be used to prove one or more of the "who, what, when, where, why and how" of a security incident. The registry analysis proves the "who, when, where and how" of the digital evidence definition. The registry analysis also answers all three queries: (1) was the application installed, (2) was the application actually used, and (3) is there any link to the digital crime? Application log files prove the "where, who, and when" of a piece of digital evidence and respond to all three queries. Prefetch files prove the "when and how" part and answer the queries; was the application installed and was the application used? By following these three queries an investigator is able to conduct an investigation in a

Table 5. Summary of user-generated file analysis

File format extension	Date of creation	Date of modification	Metadata date	Creator username	Creator tool	Location of importations	String events
indd	Yes	Yes	Yes	No	Yes	Yes	Yes
indt	Yes	Yes	Yes	No	Yes	Yes	Yes
incx	Yes	Yes	Yes	No	Yes	Yes	Yes
ai	Yes	Yes	Yes	No	Yes	No	No
ait	Yes	Yes	Yes	Yes	Yes	No	Yes
psd	Yes	Yes	Yes	No	Yes	No	Yes
eps	Yes	No	No	Yes	Yes	No	No

step-by-step uniform manner.

For content identification, the digital forensic investigator can use the recognized file signatures and the corresponding ASCII text representation to determine the file type of the graphic design applications in question. The file signatures can also be used when searching files from a formatted hard drive. Also an in-depth analysis of user-generated files can assist an investigator in knowing which particular metadata to acquire from graphic design file types and at what offset address.

By reviewing all the artifacts gathered the definition of digital evidence can be confirmed. This is so because all the six questions, "who, what, when, where, why and how" of the digital evidence definition are validated from the results acquired. Briefly clarifying the results: the "who" was specified by an artifact with the user name, the "what", specified by identifying the particular files types from the application, the "when", specified with a registry artifact indicating time of incident, the "where" specified with an artifact showing the file location, the "why" specified with a file metadata extraction revealing the file contents and the "how" with an artifact indicating which application was used for document editing. These results are essential for a digital forensic investigator to know where to look for digital forensic information, guided by knowing what information to find at a named particular location. This speeds up the process of an investigation where graphic design applications were used.

This approach is appreciated in addressing cases where document editing is largely associated with a particular application. The approach only addresses case studies involving Adobe products but the same can be done for similar graphic design applications. However, the approach doesn't tackle issues where the user only edits a hard copy, scans and prints without using any pre-installed application. The techniques discussed can be incorporated in bigger digital forensic tools like FTK and Encase or possibly the design of a crime specific tool similar to a porn detection stick created by Parabens software [21], which is a thumb drive device that will scan and detect pornographic content on a computer.

5 Conclusion

Registry keys, log files and prefetch files each reveal information that can be of digital forensic value. All this digital information can be correlated to constitute the digital evidence related to graphic design applications. Overall the three queries - was the application installed, was the application used, and is there any link between the crimes being investigated - have been responded to. By responding to all the three queries, the investigator eliminates doubts about whether an application was installed or used before establishing a possible link to the crime in question.

Moreover, it is possible for a digital forensic investigator to conduct an in-depth analysis of files generated from graphic design applications. For user generated file examination the investigator is able to verify the identity of a file type through content identification using file signatures. Also an investigator is able to know which metadata can be extracted from user generated files from graphic design applications.

Revisiting the problem "graphic design applications can be used to create fraudulent documents" and having acquired the necessary digital forensic artifacts, a digital forensic investigator is able to deduce activities associated with the creating of fraudulent documents.

The experiments were conducted using the most used graphic design applications, so that the evidence illustrated can be of use to most digital forensic investigations. The work presented is suitable in cases were digital document counterfeiting has been exercised. The work does not cover cases in which hard copy documents have been counterfeited.

Aside from the five techniques, registry analysis,

application log file analysis, system prefetch analysis, content identification (signature verification) and content examination (metadata extraction) discussed above, more techniques can be tested for future work to gather digital forensic information related to the use of graphic design applications. The work contained in this paper can be incorporated into OpenCV [29] for use in detecting inserted images for example fingerprints, bar codes in counterfeit documents. Also, future work can be conducted by carrying out this exercise on other graphic design applications.

References

[1] Bloomberg News, "Stocks weaken after Fed Statements, The New York Times", 12 June 2011.

[2] M. G. Solomon, D. Barett, and N. Broom, Computer Forensics Jumpstart, Sybex, London, 2005, pp. 51.

[3] E. Casey, Digital evidence and computer crime, London, Academic Press, 2000, pp. 10.

[4] Gartner Research, "Which operating system will be 2011's bestseller", Accessed 11 August 2011.

[5] D. Jones, "Adobe 2Q Net Up 54% On Broad Sales Gains, Higher Margins", The Wall Street Journal, Accessed 21 June 2011.

[6] A. Jones, C. Valli, Building a digital forensic laboratory, Burlington, Elsevier, 2008, pp. 285.

[7] F. Cohan, "Towards a science of digital forensic investigation", IFIP Advances Digital Forensics VI, China, 2010, pp. 17-35.

[8] J. Grama, "Legal issues in information security", MA, USA, Jones and Bartlett, 2011, pp. 460-471.

[9] M. V. Zelkowitz, Advances in computers; information security. Academic Press-Elsevier, 2009.

[10] Tech Specs, www.adobe.com, Accessed 22 June 2011.

[11] "A roadmap for Digital Forensic Research", Digital Forensic Research Workshop, 2001, pp. 16.

[12] U.S. National Institute of Justice, Electronic Crime Scene Investigation Guide: A guide for First Responders, 2001.

[13] Top Tech News, "Windows 7, Office Drive Record Microsoft Revenue", Accessed 23 July 2010.

[14] H. Carvey, Windows Forensic Analysis Dvd Toolkit, 2nd Ed., Elsevier, 2009, pp. 296.

[15] I. Rawoot, "Terrorists favour 'easy' fake SA passports", Mail and Guardian, 17 June 2011.

[16] H. Carvey, Windows Registry Analysis, 2nd Ed., Elsevier, 2009, pp. 194.

[17] Reglite software, www.resplendence.com/reglite, Accessed 14 July 2011.

[18] Regview, www.accessdat.com/support, Accessed 14 July 2011.

[19] T. Padova, "Adobe Acrobat 9 PDF Bible", Indianapolis, Wiley, 2008.

[20] Winhex, www.x-ways.net/forensics, Accessed 13 June 2011.

[21] Porn detection stick, www.paraben-sticks.com/porn-detection-stick, Accessed 9 August 2011.

[22] C. Altheide, H. Carvey, Digital Forensics with Open Source tools, Elsevier, MA USA, 2011, pp. 2.

[23] J. Ingram, "Criminal evidence", 11th ed., John C Klotter Justice Administration legal Series, USA, Elsevier, 2012, pp. 846.

[24] Regslack, Downloads, www.regripper.net, Accessed September 2011.

[25] G. Kesler, File signatures, http://www.garykessler.net/library/file_sigs.html, Accessed 19 December 2012.

[26] M. Reddy, Graphic design file format database, http://www.martinreddy.net/gfx/2d-hi.html, Accessed 19 December 2012.

[27] Adobe XMP, http://www.adobe.com/ products/ xmp/index.html, Accessed 19 December 2012.

[28] E. K. Mabuto, H. S. Venter, "User-generated evidence from graphic design applications", International conference on cyber security, cyber warfare and digital forensics, CyberSec2012, pp. 195-200.

[29] Open Source Computer Vision (OpenCV), www.opencv.org, Accessed 11 September 2012.

[30] Metadata Extraction Tool, www.extractmetadata.com, Accessed 11 July 2012.

[31] J. Bargas, "Brazilian man attempted to open a bank account using a fake Jack Nicholson ID", International Business Times, http://au.ibtimes.com/, 2 March, 2012.

[32] C. C. Lien, "Fast forgery detection with the intrinsic resampling properties", Journal of information security, vol. 1, no. 1, 2010, pp. 11-22.

[33] M. C. Stamm, "Forensic detection of image tampering using intrinsic statistical fingerprints in histograms". APSIPA Annual summit and conference, Japan, 2009, pp. 563-572.

[34] K. Cohen, "Digital Still Camera Forensics", Small scale digital device forensics Journal, vol. 1, no. 1, June 2007, pp. 2-8.

[35] H. Farid, "Image forgery detection", IEEE Signal Processing Magazine, 2009, pp. 16-25.

[36] S. Bayram, I. Avcibas, B. Sankur, N. Memon, "Image manipulation detection", Journal of Electronic Imaging, vol. 15, no. 4, 2006, pp. 41-52.

[37] J. Wang, "Image forensics based on manual blurred edge detection", Multimedia informa-

tion networking and security (MINES), 2010, pp. 907-911.
[38] N. Memon, "Photo Forensics", International workshop on information security, NYU, 2012, pp. 1-27.

E. K. Mabuto is a digital forensic researcher at the University of Pretoria, South Africa and a member of the Information and Computer Security Architectures research group in the Computer Science Department. He holds a BS (Hons) in Computer Science from the Midlands State University (Zimbabwe). He conducted this research while completing his MS research at the University of Pretoria's Information and Computer Security Architectures research group. His research interests include: computer security, digital forensics, computer graphics, and computer vision.

H. S. Venter is an Associate Professor and research group leader at the Information and Computer Security Architectures research group at the University of Pretoria's Department of Computer Science. He holds a PhD in Computer Science from Rand Afrikaans University (now University of Johannesburg). His research interests include network security, intrusion detection, information privacy, and digital forensics. Prof. Venter is a member of the IFIP Working Group 11.9 (Digital Forensics) and is also a member of the organizing committees of the Information Security for South Africa conference and the South African Institute of Computer Scientists and Information Technologists national conference.

The ISC Int'l Journal of Information Security
ISeCure

July 2012, Volume 4, Number 2 (p. 171)
http://www.isecure-journal.org

Persian Abstract

تحلیل رجیستری، فایل‌های لاگ، و فایل‌های پری‌فِچ به منظور یافتن مدارک دیجیتال در برنامه‌های طراحی گرافیکی

اِنوس کِی. مابوتو و هِین اس. وِنتِر
دانشکده‌ی علوم کامپیوتر، دانشگاه پرتوریا، پرتوریا، آفریقای جنوبی

خروجی برنامه‌های طراحی گرافیکی، ردپاهایی از اطلاعات دیجیتال به جا می‌گذارند که می‌تواند در بررسی صحنه‌ی جرم دیجیتال، و به ویژه در مواردی که اسناد جعلی تولید شده است، مورد استفاده قرار گیرد. این مقاله به تحلیل فرایند بررسی صحنه‌ی جرم دیجیتال می‌پردازد که در آن تولید اسناد جعلی رخ داده است. این هدف، ابتدا با تشخیص مصنوعات صحنه‌ی جرم که پس از استفاده از برنامه‌های طراحی گرافیکی بر جای مانده‌اند، و سپس با تحلیل فایل‌های منتسب به این برنامه‌ها حاصل می‌شود. وقتی به تحلیل مصنوعات صحنه‌ی جرم که توسط برنامه‌ای تولید شده است پرداخته می‌شود، توجه خاص روی تعیین این موارد ضروری است که آیا برنامه طراحی گرافیکی نصب شده است؛ آیا برنامه استفاده شده است؛ و آیا می‌توان رابطه‌ای میان فعالیت‌های برنامه و جرم دیجیتالی برقرار کرد. پاسخ به این سؤالات، با یافتن اطلاعاتی در این خصوص در رجیستری، فایل‌های لاگ و فایل‌های پری‌فِچ انجام می‌شود. تحلیل فایل نیز شامل بررسی فایل‌های منتسب به این برنامه‌ها به منظور یافتن امضاهای فایل و فراداده می‌باشد. در انتها می‌توان تعیین کرد که آیا یک سیستم برای ساختن اسناد جعلی مورد استفاده قرارگرفته است یا خیر.

واژه‌های کلیدی: مدرک دیجیتال، بررسی صحنه‌ی جرم دیجیتال، مصنوعات صحنه‌ی جرم، برنامه‌های طراحی گرافیکی.