Overconfidence: Personal Behaviors Regarding Privacy that Allows the Leakage of Information in Private Browsing Mode.

Rodrigo de S. Ruiz [1], Fernando Pompeo Amatte, Kil Jin Brandini Park D. Sc. [2], Rogério Winter [3]
1 Malware Analysis Nucleus (NUCAM)
Renato Archer Information Technology Center (CTI)
Campinas – SP, Brazil.
2 Computer Faculty (FACOM) –Federal University of Uberlândia
Monte Carmelo – MG, Brazil
3 Brazilian Army
rodrigosruiz@outlook.com, famatte@gmail.com, kil@facom.ufu.br, rogwinter@gmail.com

Abstract — A growing concern of users about confidentiality and privacy in web related tasks presses companies to present more secure solutions that respect the right to individual privacy. However, as some sources show, the most common browsers on the market are not able to maintain adequate privacy, even with the adoption of private browsing mode. For law enforcement agents this vulnerability may give a chance to acquire evidence during an investigation. Information security or lack of it solidifies into issues that often are not technical. The first concept is the confidence. Conceptually, trust is the firm belief that one has in relation to another person or something. Certainly, several security incidents began in the confidence that software and hardware would not fail under certain conditions. This paper presents a data capture method of browsers related activities and argues that it is possible to recover text and graphics data related to pages visited during private browsing sessions. The observations, reported in this article, show a clear violation of the functional requirement to maintain user's privacy. Overall, it is important to assess and validate private browsing techniques.

Keywords: Privacy, Private browsing, Browser safety, Browser forensics.

1. INTRODUCTION

"Security is a feeling of protection, necessary and indispensable to a society and each of its members, against threats of any kind". Defense is the capable action to sustain security feeling [1]. From this concept, we can derive issues pertaining to technology, software, quality and reliability of the environments and systems that need to protect critical information. Information security depends on the reliable operation of the infrastructure, which in its nature is critical. Cyber threats exploit the growing complexity and connectivity of critical infrastructure systems, putting safety at risk.

Information security or lack of it solidifies into issues that often are not technical. The first concept is the confidence. Conceptually, trust is the firm belief that one has in relation to another person or something. Certainly, several security incidents began in the confidence that software and hardware would not fail under certain conditions.

On this point, the paper discusses a security breach in the private mode function of browsers, which starts as social process of confidence. Logically, the information provided by the developer is considered reliable. It is usually not questioned or simply accepted as true.

Therefore, this article is aimed at answering the following questions:
- Can privacy be guaranteed when the browsers are used in private mode?

- Can the data acquisition methodology used for testing be considered efficient for evaluating privacy aspects?

Our work focuses on checking the status of privacy provided by browsers. As mentioned earlier, trust is a social process that may at some point be abused. For the study, we collected statements of developers on the use of private browsing and the consequences of its adoption to user's privacy.

In this work, we present the following contributions in the privacy area of research: a collection and analysis method for data generated by browsers and web navigation and fault identification in the chain of privacy. The generic method of collection and analysis contributed significantly to the given conclusions about the limits of the private browsing functionality. Moreover, we could analyze and identify possible flaws in the chain of privacy, from aspects ranging to the knowledge collected by the operating system through the implementation and use of the private browsing mode in browsers.

This paper is an extended version of a work previously presented by the authors [2], with additional results and analysis, and it is structured in the following topics: introduction, our contribution, related works, method and tests, results, discussion, conclusion, and references.

2. CONCEPT ABOUT PRIVACY

Currently, privacy concerns have gained a prominent place in people's lives; however, the behavior in respect to privacy is different. Dienlin's work [3] discusses in depth the privacy concerns that people have and behaviors relating to privacy that are adopted.

The privacy issue on the internet is sometimes controversial and difficult to solve. Therefore, privacy is not only achieved with the use of software tools, but also with a change of attitude on how to access information on the internet.

This change in attitude is the cornerstone to achieve the desired privacy. When seeking anonymity or privacy on the internet the principle goes beyond the use of technological tools. A user when browsing on the internet is basically subject to the following elements that can monitor their habits and customs: LAN administrator, internet service provider, operating system and other applications on the computer and site services:

1. Local network administrator - a network administrator can identify internet users habits due to the available technological resources in a router and other net devices. This way, the administrator can trace user profiles, monitor network traffic and other operations.
2. Internet service provider – in the same way as the LAN administrator, the service provider can access information from users and their preferences.
3. Computer operating system - There is no software immune to errors caused by incorrect coding or hardware failure. With that in mind, software developers many times include software routines to measure and store telemetry related data based on the software execution. This information sometimes can be sent to the developer independent of the will of the user. A malicious developer may use this information in an incorrect manner or may even sell this information to interested third parties.
4. Site on the internet - for a service to be profitable on the internet, the owner of the system must constantly assess the profile of its customers. Thus, access monitoring is a mandatory activity to establish consumer habits and geographic users' location.

The goal of software testing is to show the presence of defects if they exist [4]. Similarly, the goal of our work is to identify possible weaknesses in browsers that can compromise the privacy of users.

On the one hand, such a feature, if operating perfectly aligned with security guidelines, provides the user privacy in their online activities. On the other hand, it is clear that in case of unlawful behavior, law enforcement officers have to deal with this layer of protection to obtain the necessary data to provide evidence during the course of an investigation.

In both cases, it is important to verify the actual functionality of such a feature, if available implementations actually provide the degree of confidentiality offered, or if there are flaws that allow the retrieval of online activity data.

3. RELATED WORKS

Aggarwal et al. [5], establishes a definition of the attack model between site attacker and web attacker. Moreover, the study is based on a technique where they discoverer how to remotely test if a browser is currently in private browsing mode. Finally, they describe an automated technique to identify failures in private browsing implementations and used it to discover a few weaknesses in the Firefox browser. The deepest analysis was conducted in Firefox 3.5. They primarily focus their analysis on the Firefox browser where the testing of private browsing mode has been done by conducting the MozMill tests.

In Mahendrakar et al. [6], the analysis was performed to collect evidence of some standard tests. They created a website that contained individual pages which required the browser to interact with some forms. The authors used virtual machine VMWare Workstation 6.5 to perform the tests. They analyzed the existing content in virtual memory after using the browsers Firefox, Internet Explorer, Chrome and Safari.

Chivers [7] presents a study based on Internet Explorer 10, particularly about the InPrivate Browsing. The author pointed out that this version of Internet Explorer marked a profound shift in the way internet history and cache memory data are stored within the file system. The system was replaced with a high performance database technology known as the Extensible Storage Engine (ESE). This paper reports the results of the experiment performed on the Windows desktop 8. He discusses some implications for seizure tactics where the evidence can be found on the increasingly complex data structures used to record the activity on the internet. The prospect of recovery of such evidence, together with its potential forensic importance, raises questions, including where and when such evidence can be retrieved, so you can prove that a recovered artifact originated of an InPrivate browsing session.

In their article Ohana and Shashidhar [8], also working with Internet Explorer 8 among other browsers discovered residual artifacts from private and portable web browsing sessions. Portable web browsing artifacts are primarily stored where the installation folder is located (removable disk). Their testbed was composed of Microsoft Internet Explorer, Mozilla Firefox, Apple Safari, and Google Chrome, but they used Microsoft Windows 7 Professional 64 bits.

4. METHOD AND TESTS

When testing a security feature, it is necessary to define its functional requirements and the profile of the attacker who will try to disable or override this feature.

A paper on the analysis of private browsing functionality [5], lists the profiles of potential attackers, security models to be checked and the objectives to be met by browsers that implement private browsing.

One must understand that when privacy is important, any element pertaining to the set of resources used could be responsible for leaking private data. In this respect, any browser plug in must be compliant with the security policies in use:

“Browser plug-ins and extensions add considerable complexity to private browsing. Even if a browser adequately implements private browsing, an extension can completely undermine its privacy guarantees” [5]

Also noteworthy is that attackers could be either local or remote. In the first case, one has physical access to the user´s machine while in the second one can only launch attacks through network connections.

Furthermore, according to [5] we can classify the changes caused by user´s navigation actions in four different categories:

1. Changes caused by web site independent of the user actions, e.g. caching.
2. Changes caused by web site but dependent of the user actions, e.g. adding a certificate.
3. Changes caused directly by user actions, e.g. adding data to a form field.
4. Changes caused by other sources, e.g. updating the browser.

Changes pertaining to any category could be the source of a breach on private browsing.

In this work, we start from the methodological framework presented by [5], for the construction of the following methodological model:

The profile of the attacker considered assumes that he has local access to the user machine. Consequently, attempts to circumvent the system of private browsing will occur from an image taken from the user's machine hard drive.

As the focus of the evaluation is the private browsing feature, we considered that the user does not adopt other security tools or techniques that could exert influence on the access of the data generated during navigation.

As an example of such influence, [9] considers the impacts on forensic evaluation caused by the adoption of cryptographic methods in the disk of the user's machine. In the case of file level cryptography (or the use of encrypted containers), as those mechanisms are not fully integrated with the operational system, they are not able to prevent activities from generating sensitive data outside the containers or files protected, such as application data found on temporary files or even on swapping and paging structures on the file system.

This level of protection is only achieved with the adoption of full disk encryption (FDE). In this case, turning off the target machine and cloning it´s hard drive may not be the best approach, because the entire hard drive contents will be protected and the forensic analyst will need the passphrase or cryptographic key to access then. Given the circumstances, [9] points that forensic analysts should consider performing live system forensic when possible in systems where FDE is applied.

When performing forensic analysis of browsers, one can consider the specific artifacts such as data structures and files or implementation characteristics of each one of them or perform a browser independent forensic analysis of the entire file system searching for significant data.

Proposing a new tool for browser forensic analysis, [10] present a list of browser structures that could be targeted, such as history, cookies, download lists, bookmarks, cache and index.dat file. Furthermore, they propose a methodology to extract search history of search engines used in the browsers by users through the application of signatures derived from the study of HTTP URL generated from those searches.

However, this paper focus on searching the user's machine for fragments of data from which text or images that brings information about pages visited could be extracted. Therefore, the specific analysis of changes to files used by browsers such as history, cookies, cache and certificates was not performed. Specific analysis of those characteristics can be found in [5], [10], and [6].

We performed two different test batches. In the first batch, four different set of actions were performed on the browsers Internet Explorer [11], Firefox [12], Google Chrome [13] and Safari [14].

We tested Internet Explorer browser on bare metal hardware with the use of four notebooks equipped with Windows 7 Pro SP1.

For the other tests performed, we created a standard guest virtual machine - with the operating system Windows 7 Pro - in the host operating system - Windows 7 Pro - using the virtualization software Virtual Box [15].

An export (snapshot) of the newly installed Windows machine was created, considering the possible need for future comparison of the base guest machine with guest machines running the different browsers tested.

The browsers tested were Internet Explorer 10, Firefox 24.0_1, Google Chrome 30.0.159969M_1 and Safari 5.1.7_1. The base guest virtual machine for each browser was replicated 4 times, each to be used in the four different tests performed on each browser.

Based on those configurations, the four different set of actions were applied for each browser in private browsing mode:

Table 1: Test Type

Test Type	Action
S (Shutdown)	Consists of visiting a web site available on the internet, making operations to interact with the site, finish the execution of the browser correctly and generating the virtual machine image for analysis. This test is the most favorable for both the operating system and the browser because the user follows the steps expected for the shutdown of the machine.
F (Freeze)	Consists of visiting a web site available on the internet, making operations to interact with the site and with the browser still active, generating the virtual machine image for analysis.
K (Kill process)	Consists of visiting a web site available on the internet, making operations to interact with the site, requesting that the operating system interrupt the browser execution and generating the virtual machine image for analysis.
P (Power down)	Consists of visiting a web site available on the internet, making operations to interact

	with the site, requesting the virtualizer to turn off the virtual machine - simulating a power outage - generating the virtual machine image for analysis.

In the second test batch, we only applied one set of actions, represented by Test S, for the browsers TOR Browser Windows 3.6.6 [16] and Safari 6.0.3 (8536.28.10).

The TOR browser test was conducted on a guest operating system Windows 7 Pro SP1 running over a Virtual Box [15] virtual machine.

The Safari test happened on MAC OS Mountain Lion, 10.8.3 running as guest on a VMWare [17] virtual machine.

For each test performed, the virtual machine image generated is analyzed through the application of the program Strings [18] found in many different Linux distributions.

This program is used for the search of strings inside the virtual machine images that could present relation to the webpage visited.

The images of the virtual machines are also analyzed for the search of graphic files associated with the visited webpage, through the usage of the Foremost program [19], a renowned forensic tool for extraction of files - "data carving" - of different formats.

A study about data analysis inside the Windows pagefile, [20] points out that for data extraction the two approaches represented by both tools (Strings and Foremost) have differences but are considered standard inside the forensic analysis field.

About the process of file carving, one can generalize the method as:

“By using a database of headers and footers (essentially, strings of bytes at predictable offsets) for specific file types, file carvers can retrieve files from raw disk images, regardless of the type of filesystem on the disk image.” [21]

In other words, the foremost tool works as follows: It reads a block of data - memory, disk or files - and looks for signatures (headers or footers) related to files of well-known formats. It is noteworthy that in the present research we investigated only the persistent memory (i.e. physical and virtual disk).

Since these signatures are a sequence of bytes, there is the chance of occurrence of false positives and therefore the capture of incorrect file.

Furthermore, it is important to note the existence of several known problems associated with the use of tools aiming for "data carving”, for example, limitations to the treatment of non-contiguous data. Thus, it is possible for an image whose sequence of bytes is dispersed to not be fully recovered, despite its possible existence in the block of data analyzed.

About the Foremost software, it is also possible to measure its acceptance and support in the digital forensic community through the analysis of many studies where it was applied.

In one such paper about the forensic analysis of the XBOX videogame system, [22] discusses that the use of Foremost could speed up the carving of data of XBOX executable files (“xbe” type).

While discussing different forensic techniques for mobile Windows phone analysis, [23] compares the carving performance (with comparing metric given by numbers of artifacts detected, partially or fully recovered) of the data carving programs Scalpel, Foremost, Simple File Carver and Phone Image Carver.

To test a proposed methodology to compare file carvers, [24] chooses the following tools: Foremost, Scalpel, PhotoRec and Adroit. Their conclusion based on the results obtained was that the best approach is to use various tools in order to explore the strong points of each one and to perform file validation tests after the data carving tools processing.

One last important comment about the choice of Foremost is that it is an open source tool. As any forensic evidence that could potentially be used in a court of law, it is of paramount importance for the evidence to be deemed as legally valid. To that end, the tools used in the process should clearly follow any legal guidelines pertinent to evidence reliability. And as one study points out, under the Daubert test guidelines, “…open source tools may more clearly and comprehensively meet the guideline requirements than would closed source tools.” [25]

The WinHex tool [26] was also used to search for keywords found in the navigated webpage.

5. RESULTS

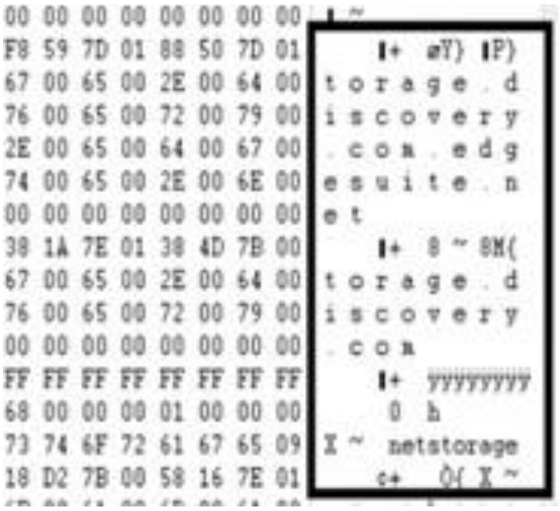
Aiming to simulate an actual visit to any website available on the internet, a random selection was made, and the sites chosen for the experiment was [27] and [28]. Since some site information is proprietary, the figures recovered during the test will be only partially reproduced in the present work. We would like to acknowledge that those information are copyright of their respective owners.

First Batch of Tests – Four Different Set of Actions

SAFARI Browser

For the Safari browser, the following results were obtained:

F test (freeze)

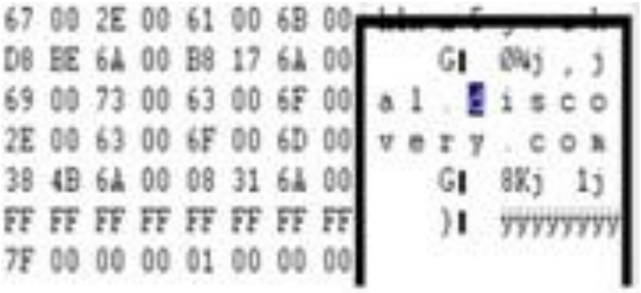


```
00 00 00 00 00 00 00 00  ▌ ~
F8 59 7D 01 88 50 7D 01     ▌+  øY} ▌P}
67 00 65 00 2E 00 64 00  t o r a g e . d
76 00 65 00 72 00 79 00  i s c o v e r y
2E 00 65 00 64 00 67 00  . c o m . e d g
74 00 65 00 2E 00 6E 00  e s u i t e . n
00 00 00 00 00 00 00 00  e t
38 1A 7E 01 38 4D 7B 00     ▌+  8 ~ 8M{
67 00 65 00 2E 00 64 00  t o r a g e . d
76 00 65 00 72 00 79 00  i s c o v e r y
00 00 00 00 00 00 00 00  . c o m
FF FF FF FF FF FF FF FF     ▌+  ÿÿÿÿÿÿÿÿ
68 00 00 00 01 00 00 00      0  h
73 74 6F 72 61 67 65 09  X ~  netstorage
18 D2 7B 00 58 16 7E 01     c+   Ò{ X ~
```

Figure 1 - "storage.discovery.com" string located in virtual machine´s image.

No image fragments were found on the virtual machine´s hard disk image.

K Test (kill process)



```
67 00 2E 00 61 00 6B 00
D8 BE 6A 00 B8 17 6A 00      G▌  ØNj , j
69 00 73 00 63 00 6F 00  a l . E i s c o
2E 00 63 00 6F 00 6D 00  v e r y . c o m
38 4B 6A 00 08 31 6A 00      G▌  8Kj  1j
FF FF FF FF FF FF FF FF      )▌  ÿÿÿÿÿÿÿÿ
7F 00 00 00 01 00 00 00
```

Figure 2 - "discovery.com" string located in virtual machine´s image.

Images related to the webpage visited were found on the virtual machine hard disk image analysis:

Figure 3 – Image recovered on hard disk image analysis and found on Discovery.com website.

The strings utility could also recover text references in the virtual machine hard disk image analysis that indicated the webpage visited:

p://dsc.discovery.com/videos
http://store.discovery.com/?ecid=PRF-DSC-101345&pa=PRF-DSC-101345

P Test (Power down)

Images related to the webpage visited were found on the virtual machine hard disk image analysis:

Figure 4 – Image recovered on hard disk image analysis and found on Discovery.com website.

The strings utility could also recover text references in the virtual machine hard disk image analysis that indicated the webpage was visited:

http://store.discovery.com/discovery/layout/favicon.ico
http://dsc.discovery.com/
http://games.dsc.discovery.com/
http://dsc.discovery.com/tv-shows
http://store.discovery.com/discovery/layout/favicon.ico

S Test (Shutdown)

Images related to the webpage visited were found on the virtual machine hard disk image analysis:

Figure 5 – Image recovered on hard disk image analysis and found on Discovery.com website.

The strings utility could also recover text references in the virtual machine hard disk image analysis that indicated the webpage visited:

http://dsc.discovery.com/tv-shows
http://dsc.discovery.com/
http://store.discovery.com/discovery/layout/favicon.ico
http://dsc.discovery.com/videos
america.discovery.com.edgesuite.net
velocity.discovery.com
metrics.discovery.com
orate.discovery.com
animal.discovery.com.edgesuite.net

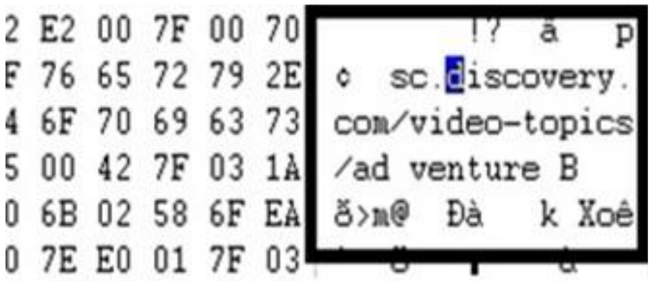
The results obtained for the Safari browser tests are grouped in table 1:

Table 2 – Results for Safari Browser

	F Test	K Test	P Test	S Test
Page address recover	Yes	Yes	Yes	Yes
Picture recover	No	Yes	Yes	Yes

FIREFOX browser:

F Test (freeze)

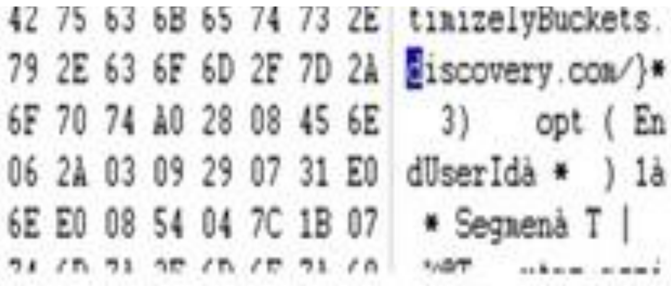


```
2 E2 00 7F 00 70        !?  à   p
F 76 65 72 79 2E  ¢   sc.discovery.
4 6F 70 69 63 73  com/video-topics
5 00 42 7F 03 1A  /ad venture B
0 6B 02 58 6F EA  ŏ>m@  Đà   k Xoê
0 7E E0 01 7F 03
```

Figure 6 – "sc.discovery.com/video-topics" string located in the virtual machine´s image.

No images related to the webpage visited were found on the virtual machine hard disk image analysis.

K Test (kill process)



```
42 75 63 6B 65 74 73 2E  timizelyBuckets.
79 2E 63 6F 6D 2F 7D 2A  discovery.com/}*
6F 70 74 A0 28 08 45 6E    3)    opt ( En
06 2A 03 09 29 07 31 E0  dUserIdà *  ) 1à
6E E0 08 54 04 7C 1B 07   * Segmenà T |
```

Figure 7 – "discovery.com" string located in the virtual machine´s image.

Images related to the webpage visited were found on the virtual machine hard disk image analysis:

Figure 8 – Image recovered on hard disk image analysis and found on Discovery.com website.

The strings utility could also recover text references in the virtual machine hard disk image analysis that indicated the webpage visited:

C:\Program Files\Mozilla Firefox\firefox.exe
ttp://games.dsc.discovery.com/
/ttp://dsc.discovery.com/videos
http://games.dsc.discovery.com/word-games
http://games.dsc.discovery.com/sport-games
https://securestore.discovery.com/cart.php
https://securestore.discovery.com/cart.php
store.discovery.com
http://games.dsc.d

P Test (Power down)

Images related to the webpage visited were found on the virtual machine hard disk image analysis:

Figure 9 – Image recovered on hard disk image analysis and found on Discovery.com website.

The strings utility could also recover text references in the virtual machine hard disk image analysis that indicates the webpage visited:

investigation.discovery.com.edgesuite.net
netstorage.discovery.com.edgesuite.net
netstorage.discovery.com
netstorage.discovery.com.edgesuite.net
netstorage.discovery.com.edgesuite.net
netstorage.discovery.com.edgesuite.net
games.dsc.discovery.com

S Test (Shutdown)

Images related to the webpage visited were found on the virtual machine hard disk image analysis:

Figure 10 – Image recovered on hard disk image analysis and found on Discovery.com website.

The strings utility could also recover text references in the virtual machine hard disk image analysis that indicated the webpage visited. A fraction of strings retrieved in this test follows:

```
Fdsc.discovery.com%2Fvideo-topics%2Fadventure&u=oeu1381760545360r0.4355827774372748&wxhr=true&t=1381760579498&f=340937086
http://dsc.discovery.com/
h;e++)if(a[e].name=="keywords")if(b=="")b=a[e].content;else b+=", "+a[e].content;else if(a[e].name=="description")c=a[e].content;if(!(b.length+c.length>eb)){z("dmk",b);z("dmd",c)}}function ub(){var a="__cmb",b=[];for(var c in aa)c.indexOf(a)==0&&b.push(c
s_sess=%20s_cc%3Dtrue%3B%20s_campaign%3DPRF-DSC-101345%3B%20s_sq
```

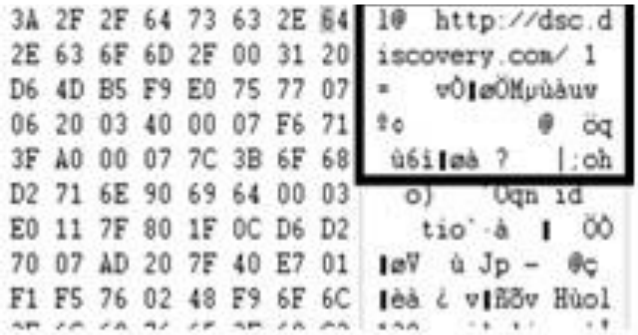
Table 3 – Results for FireFox Browser

	F Test	K Test	P Test	S Test
Page address recover	Yes	Yes	Yes	Yes
Picture recover	No	Yes	Yes	Yes

GOOGLE CHROME Browser

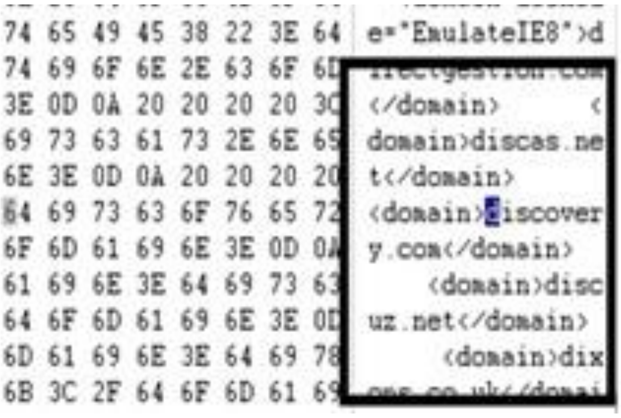
F Test (freeze)

Some strings related to the webpage were also found in hard disk analysis:



```
3A 2F 2F 64 73 63 2E 64  1@  http://dsc.d
2E 63 6F 6D 2F 00 31 20  iscovery.com/ 1
D6 4D B5 F9 E0 75 77 07  =   vÔ▮øÖMµùàuv
06 20 03 40 00 07 F6 71  ‡c          @  öq
3F A0 00 07 7C 3B 6F 68   ù6i▮øà ?    |;oh
D2 71 6E 90 69 64 00 03    o)    Òqn id
E0 11 7F 80 1F 0C D6 D2     tio`·à  ▮  ÖÒ
70 07 AD 20 7F 40 E7 01  ▮øV  ù Jp –  @ç
F1 F5 76 02 48 F9 6F 6C  ▮èà ¿ v▮ÑÕv Hùol
```

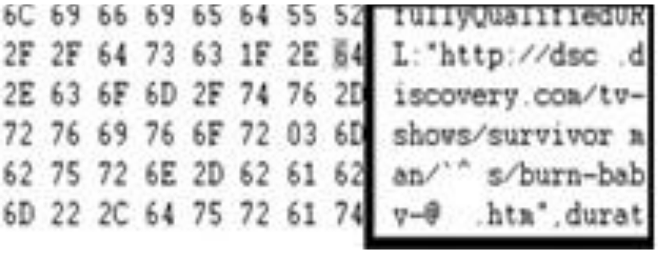
Figure 11 - "http://dsc.discovery.com" string located in the virtual machine´s image.



```
74 65 49 45 38 22 3E 64  e="EmulateIE8">d
74 69 6F 6E 2E 63 6F 6D  irectgestion.com
3E 0D 0A 20 20 20 20 3C  </domain>      <
69 73 63 61 73 2E 6E 65  domain>discas.ne
6E 3E 0D 0A 20 20 20 20  t</domain>
64 69 73 63 6F 76 65 72  <domain>discover
6F 6D 61 69 6E 3E 0D 0A  y.com</domain>
61 69 6E 3E 64 69 73 63      <domain>disc
64 6F 6D 61 69 6E 3E 0D  uz.net</domain>
6D 61 69 6E 3E 64 69 78       <domain>dix
6B 3C 2F 64 6F 6D 61 69  ons.co.uk</domai
```

Figure 12 - "discovery.com< /domain>" string located in the virtual machine´s image.

K Test (kill process)



```
6C 69 66 69 65 64 55 52  fullyQualifiedUR
2F 2F 64 73 63 1F 2E 64  L:"http://dsc .d
2E 63 6F 6D 2F 74 76 2D  iscovery.com/tv-
72 76 69 76 6F 72 03 6D  shows/survivor m
62 75 72 6E 2D 62 61 62  an/`^ s/burn-bab
6D 22 2C 64 75 72 61 74  v-@  .htm",durat
```

Figure 13 – "discovery.com/tv-shows" string located in the virtual machine´s image.

Images related to the webpage visited were found on the virtual machine hard disk image analysis:

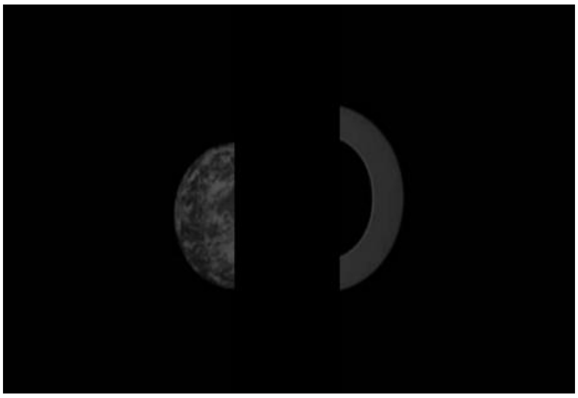


Figure 14 – Image recovered on hard disk image analysis and found on Discovery.com website.

Figure 15 – Image recovered on hard disk image analysis and found on Discovery.com website.

P Test (Power down)

Images related to the webpage visited were found on the virtual machine hard disk image analysis:

Figure 16 – Image recovered on hard disk image analysis and found on Discovery.com website.

The strings utility could also recover text references in the virtual machine hard disk image analysis that indicated the webpage visited. A fraction of strings retrieved in this test follows:

//dsc.discovery.com/
://static.ak.facebook.com/connect/xd_arbiter.php?version=27#cb=fdde13148&domain=dsc.discovery.com&origin=http%3A%2F%2Fdsc.discovery.com%2Ff2a7e0cd34&relation=parent&error=unknown_user
/dsc.discovery.com/tv-shows
://dsc.discovery.com/
://dsc.discovery.com/
://dsc.discovery.com/
http://dsc.discovery.com/tv-shows
http://dsc.discovery.com/tv-shows
http://dsc.discovery.com/

S Test (Shutdown)

Images related to the webpage visited were found on the virtual machine hard disk image analysis:

Figure 17 - Image recovered on hard disk image analysis and found on Discovery.com website.

The strings utility could also recover text references in the virtual machine hard disk image analysis that indicated the webpage was visited. A fraction of strings retrieved in this test follows:

":"Survivorman Videos","srtUrl":"","uuid":"8e18dcd9-8d1d-11e2-a7b7-06a90ff35868","bdat":"must watch","keywords":"survivorman,10 days,ten,days,must watch,mexico,tiburon,deserted,island,les stroud,survival,survivor,man,water,pool,algae,fresh,cane,reed,sludge","mediaType":"lift","mp4":[{"bitrate":"110k","src":"http://discsmil.edgesuite.net/digmed/hdnet/07/a7/13776400801197_102MissingPiece-110k.mp4"}f.akamaihd.net/i/digmed/hdnet/98/9a/13776401201197_104Stove-,400k,110k,200k,600k,800k,1500k,3500k,.mp4.csmil/master.m3u8","networkId":"DSC","thumbnailURL":"http://netstorage.discovery.com/feeds/brightcove/asset-thumbnails/dsc/0a5dbdfa893fec1f556a7d81c

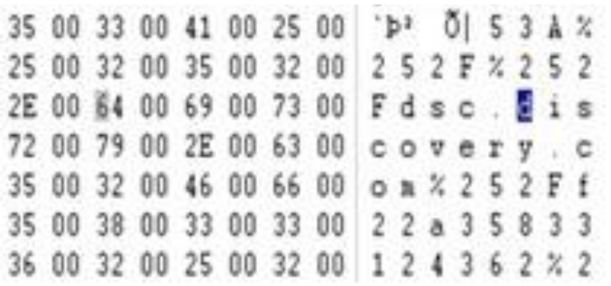
5b28bc470ecbb0e_0a5dbdfa893fec1f556a7d81c5b28bc470ecbb0e.jpg"

Table 4 – Results for Chrome Browser

	F Test	K Test	P Test	S Test
Page address recover	Yes	Yes	Yes	Yes
Picture recover	No	Yes	Yes	Yes

INTERNET EXPLORER Browser

F Test (freeze)

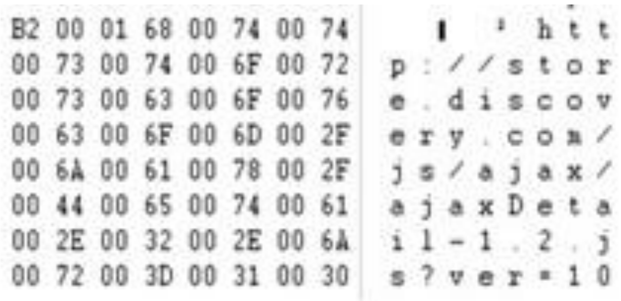


```
35 00 33 00 41 00 25 00   `Þ¹  Õ| 5 3 À %
25 00 32 00 35 00 32 00   2 5 2 F % 2 5 2
2E 00 64 00 69 00 73 00   F d s c . d i s
72 00 79 00 2E 00 63 00   c o v e r y . c
35 00 32 00 46 00 66 00   o m % 2 5 2 F f
35 00 38 00 33 00 33 00   2 2 a 3 5 8 3 3
36 00 32 00 25 00 32 00   1 2 4 3 6 2 % 2
```

Figure 18 – “discovery.com” string located in the virtual machine´s image.

No images related to the webpage visited were found on the virtual machine hard disk image analysis.

K Test (kill process)



```
B2 00 01 68 00 74 00 74      ▌   ¹ h t t
00 73 00 74 00 6F 00 72   p : / / s t o r
00 73 00 63 00 6F 00 76   e . d i s c o v
00 63 00 6F 00 6D 00 2F   e r y . c o m /
00 6A 00 61 00 78 00 2F   j s / a j a x /
00 44 00 65 00 74 00 61   a j a x D e t a
00 2E 00 32 00 2E 00 6A   i l - 1 . 2 . j
00 72 00 3D 00 31 00 30   s ? v e r = 1 0
```

Figure 19 – “http://store.discovery.com/js/ajax/” string located in the virtual machine´s image.

No images related to the webpage visited were found on the virtual machine hard disk image analysis.

P Test (Power down)

Images related to the webpage visited were found on the virtual machine hard disk image analysis:

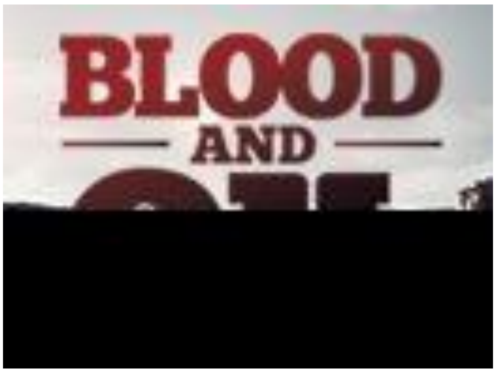
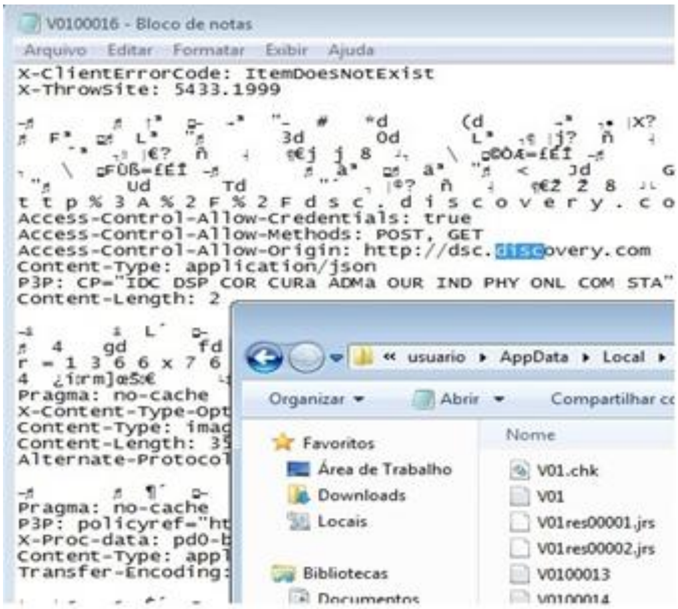


Figure 20 – Image recovered on hard disk image analysis and found on Discovery.com website.

S Test (Shutdown)

On this test, another step taken was the analysis of log files generated by the Internet Explorer browser. It is easy to see that the page address is easily visible inside a log file:



```
X-ClientErrorCode: ItemDoesNotExist
X-ThrowSite: 5433.1999

t t p % 3 A % 2 F % 2 F d s c . d i s c o v e r y . c o
Access-Control-Allow-Credentials: true
Access-Control-Allow-Methods: POST, GET
Access-Control-Allow-Origin: http://dsc.discovery.com
Content-Type: application/json
P3P: CP="IDC DSP COR CURa ADMa OUR IND PHY ONL COM STA"
Content-Length: 2

r = 1 3 6 6 x 7 6
4  ¿i:rm]œS:€
Pragma: no-cache
X-Content-Type-Opt
Content-Type: imag
Content-Length: 3
Alternate-Protocol

Pragma: no-cache
P3P: policyref="ht
X-Proc-data: pd0-b
Content-Type: appl
Transfer-Encoding:
```

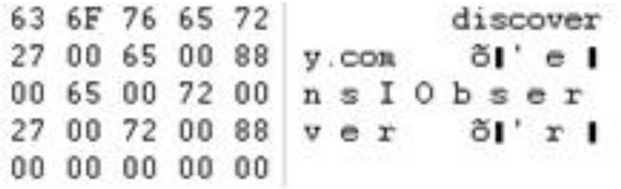
Figure 21 – Log file found using only the explorer and notepad. They demonstrate the system failure (string http://dsc.discovery.com found) in the private-IE10.

Table 5 – Results for IE10

	F Test	K Test	P Test	S Test
Page address recovery	Yes	Yes	No	Yes
Picture recovery	No	No	Yes	No

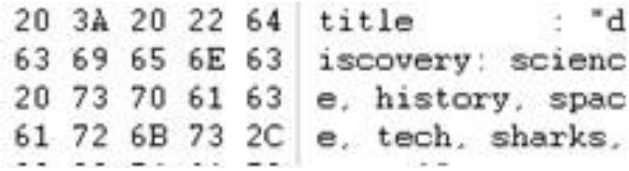
Second Batch of Tests – One Set of Actions

TOR Browser



```
63 6F 76 65 72          discover
27 00 65 00 88  y.com     õ▌' e ▌
00 65 00 72 00  n s I O b s e r
27 00 72 00 88  v e r     õ▌' r ▌
00 00 00 00 00
```

Figure 22 – “discovery.com” string located in the virtual machine´s image.



```
20 3A 20 22 64  title         : "d
63 69 65 6E 63  iscovery: scienc
20 73 70 61 63  e, history, spac
61 72 6B 73 2C  e, tech, sharks,
```

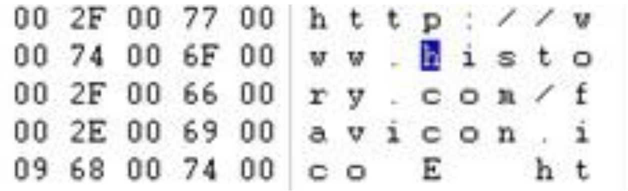
Figure 23 – “discovery: science, history, space, tech, sharks” string located in the virtual machine´s image.

Images related to the webpage visited were found on the virtual machine hard disk image analysis:

[redacted]

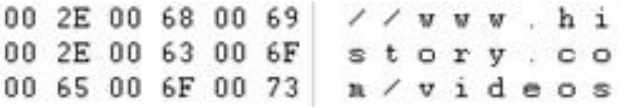
Figure 24 – Image recovered on hard disk image analysis and found on Discovery.com website.

SAFARI Browser



```
00 2F 00 77 00  h t t p : / / w
00 74 00 6F 00  w w . h i s t o
00 2F 00 66 00  r y . c o m / f
00 2E 00 69 00  a v i c o n . i
09 68 00 74 00  c o   E     h t
```

Figure 25 – “http://www.history.com/favicon.ico” string located in the virtual machine´s image.



```
00 2E 00 68 00 69   / / w w w . h i
00 2E 00 63 00 6F   s t o r y . c o
00 65 00 6F 00 73   m / v i d e o s
```

Figure 26 – “http://www.history.com/videos” string located in the virtual machine´s image.

No images related to the webpage visited were found on the virtual machine hard disk image analysis. Further analysis to prospect the files and directories involved in the data leakage generated the following results:

In all browsers, some of the data associated with the navigation could be extracted from the file pagefile.sys. This proves that part of the data is leaking through the paging process´s storage mechanism used by the operating system.

In Internet Explorer´s case, more data could be found in a file located at the directory:

> *\user\<username>\appdata\local\microsoft\windows\temporary internet files\low\content.ie5\ndm4l4gv\*

On Chrome´s case, more data could be found in the file:

> *\user\administrador\appdata\local\microsoft\windows\webcache\webcachev01.dat*

Those files points to the fact that navigation data is leaking from cache files used by the browsers.

In Table 5, we can see a summary of all tests.

Table 6 – Summary

Recovery	F Test	K Test	P Test	S Test	Browser
Page address	Yes	Yes	No	Yes	Safari
	Yes	Yes	Yes	Yes	Firefox
	Yes	Yes	Yes	Yes	Chrome
	n/a	n/a	n/a	Yes	Tor Browse
	n/a	n/a	n/a	Yes	Chrome/ Android 4.0.3
Picture	No	Yes	Yes	Yes	Safari
	No	Yes	Yes	Yes	Firefox
	No	Yes	Yes	Yes	Chrome
	n/a	n/a	n/a	Yes	Tor Browse
	n/a	n/a	n/a	Yes	Chrome/ Android 4.0.3

6. DISCUSSION

After the tests carried out using the proposed methodology, we can return to the issues that gave rise to the current research. Can privacy be guaranteed when the browsers are used in private mode? Can the data acquisition methodology used for testing be considered efficient for evaluating privacy aspects?

In this context, we can discuss two possibilities: the effects of operating system in private mode browsing and the particularities on the implementation of the functionality itself. We understand that certain beliefs may be proven wrong under those two approaches.

In the first case, the software is built upon the operating system abstraction layers, and various functions and system calls required for browsers are imported from the operational system itself. In this way, memory management and I/O operations are under the domain of the operational system removing the browser's power to determine what should be recorded and where on. Without full control of those actions, the browser is dependent on the OS to maintain user's privacy.

In the second case, the browser's domain, the developer creates expectation of privacy in users when they declare that their software has features that are able to prevent others to reconstruct the steps the users took during their online activities. As an example Firefox and TOR Bundle browsers rely on functions that are specific to the Windows operating system.

However, with the possibility of user's privacy loss, as shown in the results gathered on this paper, the statements by the developers about the insurance of user's privacy seem misleading and, therefore, can destroy the trust between the parties.

On the other hand, in the case of the IE, Chrome and Safari browsers developers are the same developers of the operational system, Windows, Android and Mac OS, respectively. For this reason, the developers have condition to fully control and change the system behavior. However, what we see is a situation similar to that covered on the previous paragraph, because even in favorable condition browsers behavior is the same, leaving residues that could allow some form of identification of web browsing habits of users.

All things considered and returning the assumptions that led to the present research, private mode browser functionality is not sufficient to guarantee users' privacy when tested with the article proposed method and boundary conditions.

Finally, we argue that the methodology used in collecting and analyzing the data is valid to evaluate the implementation aspects of private browsing. It allowed the construction of a privacy model that supported the discussions and elucidated the key aspects and assumptions analyzed that ultimately proved problems not only in the implementation of various browsers private browsing functions but also on the management of resources by the operational systems.

7. CONCLUSION

In all four types of tests performed, it is possible to verify that all browsers tested presented flaws in their private browsing feature.

Those flaws generates data that remain available in the system and allow not only the identification of pages visited but in some cases also to partially rebuild them.

Browsers promises to leave no traces of the navigation activities of users. This work proves that privacy as advertised is not provided.

In face of the results obtained, we would like to recommend the developers to explicitly alert the users about the limitations of the private browsing functionality implementation.

If on one hand this is a negative point for the user, on the other hand those flaws facilitate the work of law enforcers in cases where there is need for the data recovery related to the navigation activity.

8. REFERENCES

[1] Escola Superior de Guerra, “CAMPOS DE ATUAÇÃO DO PODER NACIONAL,” em *Manual Básico da Escola Superior de Guerra - Elementos Fundamentais Volume I*, Rio de Janeiro, Biblioteca General Cordeiro de Farias, 2013, pp. 66 - 84.

[2] R. d. S. Ruiz, F. P. Amatte e K. J. B. Park, “Tornando Pública a Navegação “InPrivate”,” em *Proceedings of the IcoFCS2012*, Brasília - Brazil, 2012, pp. 67-75.

[3] T. Dienlin e S. Trepte, “Is the privacy paradox a relic of the past? An in-depth analysis of privacy attitudes and privacy behaviors,” *European Journal of Social Psychology*, 31 July 2014.

[4] M. E. Delamaro, J. C. Maldonado e M. Jino, Introdução ao Teste de Software, 1 ed., vol. 1, Rio de Janeiro: Elsevier, 2007.

[5] G. Aggarwal , E. Bursztein, C. Jackson e D. Boneh, “An Analysis of Private Browsing Modes in Modern Browsers,” em *Proceedings The Advanced Computing Systems Association*, Washington, DC, 2010, pp. 6-6.

[6] A. Mahendrakar, J. Irving e S. Patel, “Forensic Analysis of Private Browsing Mode in Popular Browsers,” 25 April 2010. [Online]. Available: http://mocktest.net/paper.pdf. [Accessed at:30 November 2014].

[7] H. Chivers, “Private browsing: A window of forensic

opportunity," *Digital Investigation,* vol. 11, nº 1, p. 20–29, 2014.

[8] D. J. Ohana e N. Shashidhar, "Do Private and Portable Web Browsers Leave Incriminating Evidence? A Forensic Analysis of Residual Artifacts from Private and Portable Web Browsing Sessions," EURASIP Journal on Information Security - Springer Open Journal, 21 November 2013. [Online]. Available: http://jis.eurasipjournals.com/content/2013/1/6. [Accessed at:03 December 2014].

[9] E. Casey e G. J. Stellatos, "The Impact of Full Disk Encryption on Digital Forensics," *ACM SIGOPS - Operating Systems Review (OSR),* vol. Vol 42, nº Issue 3, pp. 93-98, 2008.

[10] J. Oh , S. Lee e S. Lee , "Advanced evidence collection and analysis of web browser activity," em *The Proceedings of the Eleventh Annual DFRWS Conference,* New Orleans - USA, 2011, pp. s62-s67.

[11] Microsoft, "What is InPrivate Browsing?," Microsoft, 15 November 2014. [Online]. Available: http://windows.microsoft.com/en-us/windows/what-is-inprivate-browsing#1TC=windows-7. [Accessed at: 15 November 2014].

[12] Mozilla Contribuitors, "Private Browsing," Mozilla, 15 November 2014. [Online]. Available: https://support.mozilla.org/en-US/kb/private-browsing-browse-web-without-saving-info. [Accessed at: 15 November 2014].

[13] Google Inc., "Chrome," Google Inc., 15 November 2014. [Online]. Available: https://www.google.com.br/chrome/browser/desktop/index.html. [Accessed at: 15 November 2014].

[14] Apple Inc., "Defending your online privacy and security," Apple Inc., 15 November 2014. [Online]. Available: http://www.apple.com/safari/. [Accessed at: 15 November 2014].

[15] Oracle , "Oracle VM VirtualBox," Oracle, 15 November 2014. [Online]. Available: http://www.oracle.com/technetwork/server-storage/virtualbox/downloads/index.html. [Accessed at: 15 November 2014].

[16] Tor Project, "What is the Tor Browser?," Tor Project, Inc., 15 November 2014. [Online]. Available: https://www.torproject.org/projects/torbrowser.html.en. [Accessed at: 15 November 2014].

[17] VMmare, "All Downloads," VMware, 20 November 2014. [Online]. Available: https://my.vmware.com/web/vmware/downloads. [Accessed at: 20 November 2014].

[18] Die.net, "strings(1) - Linux man page," 10 may 2009. [Online]. Available: http://linux.die.net/man/1/strings.. [Accessed at: 30 June 2012].

[19] J. Kornblum, . K. Kendall e N. Mikus, "Foremost website," 06 June 2002. [Online]. Available: http://foremost.sourceforge.net/. [Accessed at: 12 October 2012].

[20] S. Lee , A. Savoldi, S. Lee e J. Lim, "Windows Pagefile Collection and Analysis for a Live Forensics Context," em *Proceedings of Future Generation Communication and Networking (FGCN 2007),* Jeju-Island, Korea, 2007, pp. 97-101.

[21] G. G. Richard III e V. Roussev, "A Frugal, High Performance File Carver," em *Proceedings of the 2005 Digital Forensic Research Workshop (DFRWS),* New Orleans - USA, 2005, pp. 97-101.

[22] C. Vaughan, "Xbox Security Issues and Forensic Recovery Methodology (Utilising Linux)," *Digit. Investigation,* vol. 1, nº 3, pp. 165 -172, September 2004.

[23] G. Grispos, T. Storer e W. B. Glisson, "A comparison of forensic evidence recovery techniques for a windows mobile smart phone," *Digital Investigation,* pp. 23-26, 20 July 2011.

[24] T. Courrejou e S. L. Garfinkel, "A COMPARATIVE ANALYSIS OF FILE CARVING SOFTWARE," 12 September 2011. [Online]. Available: http://www.dtic.mil/dtic/tr/fulltext/u2/a550119.pdf. [Accessed at: 20 November 2014].

[25] B. Carrier, "Open Source Digital Forensics Tools: The Legal Argument," 10 October 2002. [Online]. Available: http://dl.packetstormsecurity.net/papers/IDS/atstake_opensource_forensics.pdf. [Accessed at: 25 November 2014], pp. 1-10.

[26] X-Ways Software Technology AG, "WinHex: Computer Forensics & Data Recovery Software, Hex Editor & Disk Editor," X-Ways Software Technology AG, 10 November 2014. [Online]. Available: http://www.x-ways.net/winhex/. [Accessed at: 10 November 2014].

[27] Discovery Channel, "Discovery website," Discovery Channel, 07 October 2012. [Online]. Available: http://dsc.discovery.com. [Accessed at: 07 October 2012].

[28] History Channel, "History.com website," History Channel, 10 September 2014. [Online]. Available: http://www.history.com. [Accessed at: 10 September 2014].