




May 25th, 1:00 PM

Forensics Analysis of Privacy of Portable Web Browsers

Ahmad Ghafarian

Department of Computer Science and Information Systems, Mike Cottrell College of Business,
ahmad.ghafarian@ung.edu

Follow this and additional works at: <https://commons.erau.edu/adfsl>

 Part of the [Aviation Safety and Security Commons](#), [Computer Law Commons](#), [Defense and Security Studies Commons](#), [Forensic Science and Technology Commons](#), [Information Security Commons](#), [National Security Law Commons](#), [OS and Networks Commons](#), [Other Computer Sciences Commons](#), and the [Social Control, Law, Crime, and Deviance Commons](#)

Scholarly Commons Citation

Ghafarian, Ahmad, "Forensics Analysis of Privacy of Portable Web Browsers" (2016). *Annual ADFSL Conference on Digital Forensics, Security and Law*. 9.
<https://commons.erau.edu/adfsl/2016/wednesday/9>

This Peer Reviewed Paper is brought to you for free and open access by the Conferences at Scholarly Commons. It has been accepted for inclusion in Annual ADFSL Conference on Digital Forensics, Security and Law by an authorized administrator of Scholarly Commons. For more information, please contact commons@erau.edu.

EMBRY-RIDDLE
Aeronautical University™
SCHOLARLY COMMONS

(c)ADFSL



FORENSICS ANALYSIS OF PRIVACY OF PORTABLE WEB BROWSERS

Ahmad Ghafarian

Department of Computer Science and Information Systems

Mike Cottrell College of Business

University of North Georgia, Dahlonega, GA 30597, USA

Ahmad.ghafarian@ung.edu

ABSTRACT

Web browser vendors offer a portable web browser option which is considered as one of the features that provides user privacy. Portable web browser is a browser that can be launched from a USB flash drive without the need for its installation on the host machine. Most popular web browsers have portable versions of their browsers as well. Portable web browsing poses a great challenge to computer forensic investigators who try to reconstruct the past browsing history, in case of any computer incidence. This research examines various sources in the host machine such as physical memory, temporary, recent, event files, Windows Registry, and Cache.dll files for the *evidential information regarding portable browsing session*. The portable browsers under this study include Firefox, Chrome, Safari, and Opera. Results of this experiment show that portable web browsers do not provide user-privacy as they are expected to do.

Keywords: computer forensics tools, RAM forensics, volatile memory, forensics artifacts, Registry

1. INTRODUCTION

When surfing the web, browsers save information about the surfing activities in various locations. In an attempt to maintain privacy of web browsing, most major web browsers have a portable version of their browsers. A portable browser is a browser that can be saved on a removable storage media such as a USB flash drive. The browser can then be launched from the flash drive without the need for its installation on the host machine (Choi, et al, 2012). From computer-forensics point of view, browsing artifacts can be saved on the portable browser flash drive, server and the host machine. The local machine saves browsing data in both static media such as hard drive as well as random access memory (RAM), also known as volatile

memory (Aggarwal and Jackson, 2010). The data that is contained within the two types of sources varies significantly. Static media is primarily used for long term storage and contains data such as executables, images, documents, and browser history. On the other hand, physical memory is a temporary working space for data that are being used by the system. The major difference between the data sources in relation to a computer-forensic investigation is that the latter is a less tangible source of evidence (Simmons & Slay, 2009).

Forensics artifacts left after a portable browsing session can be retrieved from sources such as caches, history, cookies, and download file lists (Davis, 2009). On the other hand, retrieving portable browsing forensics artifacts left behind from main memory have recently

attracted some attention (Oh et al, 2011) (Shashidhar, 2013). The authors have used limited memory forensics to retrieve forensics data left after a portable browsing session. They argue that memory forensics is very promising in establishing a link between the suspect and the retrieved data.

When we are dealing with portable browsing artifacts, memory forensics would be challenging because once we remove the portable browser flash media from the suspect machine, the portable browser-related data content in the main memory will gradually be deleted.

The focus of this research is the examination of various sources such as main memory, *temporary files*, *recent files*, *event files*, *Windows Registry*, and *Cache.dll* file in the suspect machine looking for residual artifacts left behind after private portable web browsing activities. The research experimentally analyzes both static media sources such as hard drive as well as volatile memory for their evidential potential related to portable web browser activities. The portable browsers under this study include Firefox, Google Chrome, Opera and Safari. To evaluate the effectiveness of browser closure after a browsing session, the experiment is carried out in two cases: 1) portable browser flash drive left attached to the suspect machine after a browsing session and 2) portable browser flash drive was removed from the suspect machine after a browsing session. The results will be tabulated for comparison purposes.

The remainder of this paper is organized as follows: Section 2 gives literature review, section 3 provides research methodology, results appear in section 4, section 5 covers conclusion and future research is presented in section 6.

2. LITERATURE REVIEW

In this section, we review the previous work on portable browsing forensics. We study portable browsing artifacts retrievable from static media and from main memory.

2.1 Static Media Forensics

Static media forensics related to the artifacts left due a portable browsing session has been studied by various researchers. For example, Marrington et al (2013) has examined privacy of Google Chrome portable web browser using conventional forensics by taking an image of the hard drive after a portable browsing session and analyzed the image. The details of the analysis are not clear in their paper. They reported that portable Google Chrome does leave traces of browsing activities on the hard drive. In another research, privacy of the portable Google Chrome has been studied by Adautin and Meeran (2015). The researchers examined the content of the *IconCache.db* database as well as Windows Registry and reported that they found evidence of portable browsing activities. They claim that they examined the content of volatile memory; however, other than making some general recommendations, the authors provided no details of their RAM forensics process. It is worthwhile to notice that in both cases of hard drive forensics and volatile memory forensic, they left the portable flash drive connected to the suspect machine during their experiment.

Dharan and Meeran (2014) have reported that portable web browsing activities can be obtained by searching the Windows Registry and *Prefetch* files. The researchers performed both live and offline forensics and reported evidence of portable web browsing activities in both cases. However, their experiment description is very fuzzy and they did not disclose the portable browser with which they experimented.

2.2 Memory Forensics

Memory forensics involves two steps, memory capture and analysis of the captured memory. RAM capture is the process of making an image of the physical memory and saving it as a file on an external storage media. Memory analysis involves parsing the data structure tree of the captured memory file, looking for processes that were running when the memory was taken as well as other data such as passwords, downloaded files, SSL Certificates, URLs, etc. To facilitate memory forensics, several open-source and proprietary RAM forensics tools have been developed. Some examples include Volatility (2015), Redline (2015), and WinHex (2015).

One of the most comprehensive portable browsing forensics researches is the work of Ohana and Shashidhar (2013). Along with other forensics investigation methods, the researchers performed RAM forensics with three portable web browsers, namely Mozilla Firefox portable, Google Chrome portable, and Opera portable. They conclude that the best way to recover residual data is to obtain the evidence from RAM, but that is not always possible for investigators. Also, they did not disclose whether, during the RAM capture, the portable flash drive was connected to the suspect machine or not. Based on our own results, we believe the researchers captured RAM while the portable flash drive was still attached to the suspect machine.

Oh, et al. (2012) demonstrated that web browsing activities can be obtained from the web browser's log file. They suggest that current tools are not adequate for this task. Consequently, they developed a tool called WEF. This tool provides an integrated analysis function for various web browsers in various time zones. In addition, online user activity, search words, and URL parameters, which are significant information for digital forensics, can

be confirmed. In special cases, if the search word information is encoded in unfamiliar characters, this tool provides a decoding function.

Other researchers have explored memory forensics in relation to private browsing mode; however, in both cases of private browsing and private portable browsing, the process of RAM forensics and the objective remain the same, which is maintaining privacy of the user. For this reason, we briefly review some of the important memory forensics research findings here.

Mahendrakar, et al. (2010) examined various popular web browsers in private mode to determine traces of browsing activities that remain in physical memory. They created a website which contained individual pages that required the browser to interact with various types of data including SSL certificates, form passwords, form text entries, HTML files, JPEG files, and cookies. Since they used their own memory parser tool, which is not publicly available, and their experiment was performed in a controlled research setting environment, their result cannot be replicated.

Said, et al. (2011) examined the content of the volatile memory after a private browsing session and found artifacts left in memory about user activities. Private-mode browsing has also been studied by Satvat, et al. (2014). In their experiment, after navigating a few websites in private mode and closing the session, they discovered traces of private navigation in RAM. The researchers did not disclose the details of RAM forensics tools and methodologies and thus their findings cannot be proved by replication.

In a study of physical memory forensics, Hejazi, et al. (2009) proposed a new technique for extracting sensitive information from physical memory. Their technique is based on analyzing the Call Stack and the security

sensitive Application Program Interfaces (API). They implemented this technique as part of memory analysis plug-in, which takes a memory image file and analyze the file..

A theoretical discussion of RAM forensics tools, techniques and guidelines can be found in (Simmons, 2009) and (Amari, 2009). The authors discuss the way physical memory works in Windows and Linux operating systems as well as the types of forensically valuable data that can be extracted from physical memory.

3. RESEARCH METHODOLOGY

In this section, we provide the tools, techniques and the forensics investigation methodologies.

3.1 Technology and Setup

In preparation for the forensics experiment, the following tools were used.

Hardware:

- One Desktop PC (4GB RAM) for forensics workstation activities
- Four other Desktop PC (4GB RAM) for suspect activities
- Four USB Flash Drive (8GB) for portable browsers
- Four USB External Drive (8GB) for captured RAMs
- SATA to USB adaptor
- USB write blocker

Software:

- Microsoft Windows 7, Pro 32 bits, SP1
- DaemonFS- file integrity monitoring software
- Paragon DiskWipe v 12

- Nirsoft Internet Tools- history, cache, and cookie Viewers
- Firefox Portable 33.0, Google Chrome portable 42.0.2311.90, Opera portable 12.7, and Safari portable 5.1.7
- FTK Imager Lite- portable version
- SQLite Maestro software
- WinHex
- Mandiant Redline
- DumpIt memory capture software

3.2 Experiment

We started by uninstalling the OS from all four PCs and installed Windows 7 fresh. Then we installed DaemonFS (2015), which is a tool that monitors in real-time files on the hard disk. We installed several tools from NirSoft (2015) on the PCs for viewing history, cache, and cookie. Next, we used Paragon Disk Wiper (2015) to wipe all USB flash and external drives. The flash drives were installed with a free utility program called PortableApps (2015). This utility allows you to run different programs from a flash drive, similar to an OS Start menu. Subsequently, we installed the portable web browsers on the USB flash drives and connected the flash drives to the suspect PCs. We also attached the write-blocker to the suspect machines. We should note that the only browser on each machine was the portable browser, and there were no installed browsers. At this point we were ready to do the web-browsing activities. Each portable browser was individually launched in private-mode followed by the same series of web-activities, i.e. log in to email and bank account, sending/receiving email, searching for images and videos, uploading and downloading files and streaming videos.

Using the DeamonFS (2015) and NirLauncher, a Nirsoft tool (2015), we examined temp, recent, Cache.dll, and cookies with the aim of finding footprints of portable browsing activities. Our experiment showed that in all four browsers, after the USB flash drive was removed from the suspect machine, most of the browsing activities information was created, modified, and then deleted from the host machine (see Table 1). This observation is consistent with the results reported by Ohana and Shashidhar (2013). Table 1 entries show that portable Firefox and portable Opera provide slightly more privacy than portable Chrome and portable Safari. This is because with portable Chrome we were able to see some account login information. Similarly, use of Safari leaves traces of email communication activities. We repeated the RAM forensics process to verify the validity of the results but the same results were obtained the second time as well.

Table 1
Retrieved portable browsing artifacts

Portable Browser	Suspect machine Activity
Google Chrome	<i>temp, recent, and Cache.dll</i> created and then deleted. some account login info and <i>downloaded</i> files created but not deleted
Firefox	<i>temp, recent, and Cache.dll</i> created and then deleted
Safari	<i>temp, recent.Cache.dll</i> created and then deleted for email login we noticed that some <i>Appdata/Ntuser.dat</i> modified on host machine but not deleted
Opera	<i>temp, recent, and Cache.dll</i> created and then deleted

Next, we used a Registry editor to examine Windows Registry. Table 2 shows portable browsing information retrieved by examining the Registry.

Table 2
Portable browsing artifacts retrievable from Registry

Portable Browser	Registry report of host machine activity
Google Chrome	Flash drive vendor ID, product ID, version, serial number, drive letter, URLs visited were retrievable. Some registry keys was created but deleted after the browsers was removed
Firefox	Flash drive vendor ID, product ID, version, serial number, drive letter, URLs visited was retrievable. The time/date the browser launched was also visible
Safari	Flash drive vendor ID, product ID, version, serial number, drive letter, URLs visited were retrievable.
Opera	Flash drive vendor ID, product ID, version, serial number, drive letter, URLs visited was retrievable. The time/date the browser launched was also visible

Table 2 entries show information such as flash drive vendor ID, product ID, serial number, the URL history and date/time the browsers were launched. These are important evidential information for computer forensics investigators. This data establishes a link between the suspect and browsing activities; however, we were not able to see the details of browsing activities such as browsing history, cookies, search items, etc. This indicates that although examination of the Registry data is very useful, it is not sufficient.

We also analyzed the SQLite database which stores user-defined records in large tables. Examination of this database shows no details of web-surfing activities. Table 3 shows the activities on the host machine reported by SQLite database. As can be seen from the entries, there are no privacy differences

between the portable web browsers with which we experimented.

Table 3
SQLite report of portable browsing session

Portable Browser	Suspect machine Activity
Google Chrome	cookies.sqlite-wal, places.sqlite-shm, and webappsstore.sqlite-shm were deleted profile/*.db were modified
Firefox	cookies.sqlite-wal, places.sqlite-shm, and webappsstore.sqlite-shm were deleted profile/*.db were modified
Safari	cookies.sqlite-wal, places.sqlite-shm, and webappsstore.sqlite-shm were deleted profile/*.db were modified
Opera	cookies.sqlite-wal, places.sqlite-shm, and webappsstore.sqlite-shm were deleted profile/*.db were modified

For RAM forensics we followed the framework suggested by Ghafarian (2015). We chose an open source memory forensics tool called Redline (2015) for the following reasons:

- Graphical User Interface
- Selection option which allows user to choose only browsing related processes and disabling all the other processes and files
- Allow to import memory analysis results to a file such as MS Word for offline processing
- Easy to user and having a comprehensive user manual.

In Redline, the RAM capture tool is called ‘Collector’ and the RAM analysis is called ‘Memoryze.’ We created the Collector software and saved it on a USB external drive. We also created the Memoryze and saved it on the Forensics workstation machine. The details of

creating these tools can be found in Redline user manual.

The experiment consisted of two parts. In the first part, after a portable browsing session, we left the portable browser flash drive attached to the suspect machine, captured, and analyzed the RAM. In the second part, we removed the portable browser flash drive, captured the RAM, and analyzed it. RAM capture in the latter part is very time sensitive and it depends on the time gap between removal of portable browser flash drive and RAM capture. Since Redline Collector cannot collect information about terminated processes and closed files, we also used WinHex (2015) Hexadecimal editor.

3.3 RAM Forensics Process

To make data extracting less cumbersome, we cleared all cookies, cache, history, bookmarks, etc that may have been left on the suspect machines from our earlier experiment. We installed Memoryze software on the forensics workstation. To simplify analysis, we disabled physical address extension mode on Redline. We ran Redline, created the RAM capture software Collector, and saved it on a wiped USB external drive. Then we followed the below steps:

1. Attached the portable browser flash drive to the suspect machine and configured the browsers as the default browser with extensions and plug-ins disabled. Then we performed a browsing session, attached the Collector external drive to the suspect machine, captured RAM, saved the file onto the external drive and removed external drive for RAM analysis.
2. Step 1 was repeated for all the other suspect machines with different portable browser.
3. We repeated steps 1 and 2 above, but this time we removed the portable

browser flash drive and immediately captured the RAM and saved it to the external drive.

4. Configured Memoryze to retrieve only browsing-related information and processes. This action reduced the amount and time of data analysis. We imported the memory parsed data to a MS Word file for offline analysis. We should note that Redline only provide information about running processes and programs that were running before memory was captured. We also used WinHex to retrieve residual data on these processes and files.
5. Step 4 was repeated for the other three captured RAM files.

Over all, we had four RAM captured files for the cases when portable browsers flash drives were still attached to the suspect machine during the RAM capture process. And four captured RAM files for the case when portable storage flash drives were removed after each browsing session. The total memory files that we captured were eight. Considering each RAM capture took on average one hour, we spent eight hours to capture the memory of the suspect machines. The process of memory capture and analysis were performed according to the forensics investigations rules and regulations. The results are discussed in the next section.

4. RESULTS

Retrievable computer forensics artifacts after a portable browsing session via memory forensics for all four browsers are summarized in Table 4. The blue columns represent artifacts retrieved when the portable browsers flash drives were still attached to the machines when RAMs were captured. The pink columns represent retrieved data when the portable browser flash drives were removed from the

suspect machines and immediately RAMs were captured.

The blue entries in Table 4 show that with the exception of email password everything else was retrievable. That means that if the portable USB flash drive is attached to the machine during RAM capture, portable browser provides no privacy at all. In this case, the information that was retrieved from memory is enough to conclude browsing activities and establishing link between the web browsing activities and the suspect. For example, browsing history, search history, and file downloads were retrieved from memory for all of the portable browsers we studied. These are important evidential information for computer-forensics investigators.

However, when we removed the portable browser from the PC and then captured RAM, the forensics artifacts retrieved from main memory slightly varies among various browsers. This variation is discussed below.

For Mozilla Firefox, analysis of the memory dumped file showed considerable browser-related entries in memory indicating web browser activity. We were able to detect email communication details, browsing and URL history, search history, and downloaded files (documents, images, and videos). On one hand, some information such as cookies, email password, timelines, and process ID could not be retrieved. We also used Winhex to analyze the captured RAM, but did not find any of aforementioned data. This indicates that when the portable browser flash drives were removed, some of the browsing information was removed from the memory.

Similar results were observed for Opera. Analysis of the RAM showed that portable browser flash drive removal had an effect on the amount of data retrievable from memory. Similar to Firefox cookies, timelines, email passwords, and process ID were deleted before

we captured RAM. This is because once the portable browser flash drive is removed, the data structure tree that handles cookies, for example, are not accessible. On the other hand, we were able to identify HTML data containing various types of information including the SSL certificate for accessing a secure website, URL, file downloaded and more.

Analysis of physical memory for portable Google Chrome revealed forensically valuable artifacts such as Certificate, HTML text file, URL history, and files downloaded. We should note that only Google Chrome saved process ID in memory. Nevertheless, similar to Firefox, we were not able to see cookies, email passwords, and timeline. Analysis of the retrieved artifacts indicates that Google

Chrome portable left the most residual artifacts on the host machine.

For Safari, the amount of retrieved data from portable browsing session is identical to Firefox and Opera. That means cookies, timeline, and email password were not retrievable from main memory. However, like Firefox, we were able to see forensically-valuable information such as history, file downloads, SSL certificates, etc.

In an attempt to validate the retrieved data from main memory, we used another open source software tool from Microsoft called DumpIt to capture the physical memory after a browsing session. Then we used WinHex to analyze the captured images. The results for both Redline and DumpIt were identical.

Table 4
Results of Installed Browser, and Portable Browsers

Data Item	Firefox Removed	Opera Removed	Chrome Removed	Safari Removed	Firefox Attached	Opera Attached	Chrome Attached	Safari Attached
Browser process	-	-	√	-	√	√	√	√
URL History	√	√	√	√	√	√	√	√
Cookies	-	-	-	-	√	√	√	√
File downloads	√	√	√	√	√	√	√	√
Timelines	-	-	√	-	√	√	√	√
Browser history	√	√	√	√	√	√	√	√
Email password	-	-	-	-	-	-	-	-
Email ID	√	√	√	√	√	√	√	√
SSL Certificate	√	√	√	√	√	√	√	√
Search history	√	√	√	√	√	√	√	√

4.1 Analysis of the Results

Interpretation of the data captured from memory indicate that portable browsing mode does leave browsing evidence, even after the browser flash drives were removed from the machine in all four web browsers under this experiment. The type and the amount of data

varied slightly among the browsers. For example, Table 4 above shows that the Timeline and process ID are retrievable with portable Google Chrome. Also, Figure 1 shows the date, time, and the site that was visited. Among all the browsers in our study, Google Chrome portable left the most residual artifacts on the host machine.

Examination of *temp*, *recent*, *Cache.dll* showed browser activity, but all the data were deleted immediately (see Table 1). Windows Registry showed flash drive information such as vendor ID, product ID, serial number, etc. This information establishes a link between the suspect and the browsing activities. Examination of the SQLite database also showed some information about cookies but not sufficient to establish conclusion. SQLite database revealed other data such as *profile/*.db*.

Additionally, we have employed the *Ipsconfig/displaydns* command. Issuing this command will generate the site address and the IP addresses of the sites visited even after the browser media is removed but keeps changing. For example, Figures 1 and 2 show the site visited with its IP address; however, closure of the browser causes the time-to-live to be reduced from 42 to 7 seconds and thus the forensics investigator should be quick on recording the data. This observation indicates that the speed of capturing RAM after the portable browser flash drive removal from the PC is important.



Figure 1. Process Time To Live shows 42

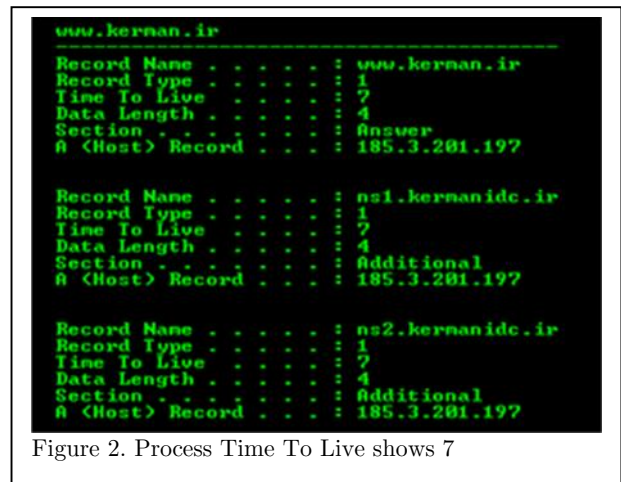


Figure 2. Process Time To Live shows 7

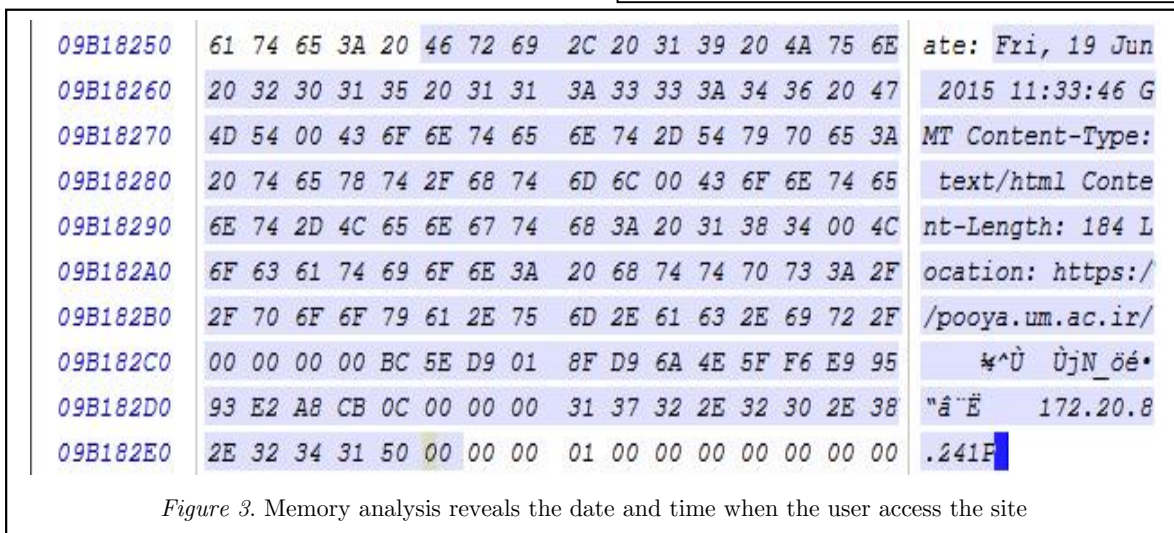


Figure 3. Memory analysis reveals the date and time when the user access the site

5. CONCLUSIONS

We used both static media forensics and RAM forensics to experimentally examine privacy feature of portable Firefox, Opera, Chrome, and Safari browsers when they are used in private mode. We found that through a combination of RAM forensics, Registry, SQLite database, temp, recent, Cache.dllfile and folders, we can retrieve forensically-valuable information about suspect's activity, such as sites visited, Internet searches, secure sites login credentials, and traces of email communication even after the portable browsers flash drive were removed from the machine. The artifacts such as flash drive vendor ID, product ID, version, serial number, drive letter, and URLs visited are enough to constitute a link between the data and the suspect. Our experiment shows that the vendor's claim of privacy can be nullified through a combination of various computer-forensics investigations. Among portable browsers under our experiment, Google Chrome portable left the most residual artifacts on the host machine.

Due to the dynamic nature of physical memory, the time gap between removing the portable browser flash media from the machine and capturing RAM is very important. The greater the time gap spent causes a greater chance of losing data in RAM.

Also, when the browsers are closed, we can retrieve the last information saved on the clipboard and analyze it for possible evidential information.

Finally, we believe the Registry is a good source for retrieving portable browsing artifacts when it is used along with memory forensics.

6. FUTURE WORK

This research can be extended in several ways; first, determine better tools and methodologies

for analyzing the volatile memory for data about terminated processes and closed files and programs. Second, repeat the same experience with different tool such as Volatility. Third, extract information over an extended period of time instead of one specified browsing session.

ACKNOWLEDGMENTS

This research was supported by the 2015 UNG Presidential semester award. The author would like to thank all the individuals who were involved in the establishment and the implementation process of this award.

REFERENCES

- Adautin, E.D. and Meeran, N. (2015). Forensic Reconstruction and Analysis of Residual Artifacts from Portable Web Browse. *International Journal of Computer Applications* (0975 – 8887), 128 (18), 19-24.
- Aggarwal, G., Bursztien, E., Jackson C., & Boneh, D. (2010). An analysis of private browsing modes in modern browsers. Proceedings of the 19th Usenix Security Symposium.
- Amari, K., (2009). Techniques and Tools for Recovering and Analyzing Data from Volatile Memory. SANS Institute InfoSec Reading Room.
- Choi, J. H., K.G. Lee, J. Park, C. Lee, and S. Lee (2012). Analysis framework to detect artifacts of portable web browser. Center for Information Security Technologies
- DaemonFS, Retrieved on May 27 from <http://sourceforge.net/projects/daemonfs/>
- Davis, N. (2009). Live memory forensics for Windows Operating Systems. Eastern Michigan University, IA 328. Retrieved, January 2015 from <https://www.emich.edu/ia/pdf/research/Live%20Memory%20Acquisition%20for%20Windows%20Operating%20Systems,%20Naja%20Davis.pdf>
- Dharan, D.G. and Meeran, N.A.R. (2014). Forensic Evidence Collection by Reconstruction of Artifacts in Portable Web Browser. *International Journal of Computer Applications* (0975 – 8887), 91(4), 32-35.
- Ghafarian, A. and Hosseini, S.A. (2015). Analysis of Private Browsing Mode through Memory Forensics. *International Journal of Computer Applications* (0975 – 8887), 132(16), 27-34
- Hejazi, S.M., Talhi, C. & Debbabi, M. (2009). Extraction of Forensically Sensitive Information from Windows Physical Memory. *Digital Investigation*, 6, 121-131. Elsevier publishing Co.
- Koepi, D. (2010). *Firefox Forensics*. Retrieved November 2014 from <http://davidkoepi.wordpress.com/2010/11/27/firefoxforensics>
- Mahendrakar, A., Irving, J., and Patel, S., (2010). Forensic Analysis of Private Browsing Mode in Popular Browsers. Retrieved August 2014 from <http://mocktest.net/paper.pdf>
- Mandiant Redline User Manual (2014). Retrieved February 2015 from https://dl.mandiant.com/EE/library/Redline1.7_UserGuide.pdf
- Marrington, A., I. Baggili, T. AI Ismail, A. AI Kaf (2013). Portable Web Browser Forensics: A forensic examination of the privacy benefits of portable web browsers. *IEEE Journal*.
- NirSofer. (2013). *NirSoft Freeware Utilities*. Retrieved February 2015 from <http://nirsoft.net>
- Oh, O., Lee, S., and Lee, S. (2011). Advanced evidence collection and analysis of web browser activity. *Journal of digital investigation* 8, 62-70
- Ohana, D.J. and Shashidhar, N. (2013). Do private and portable web browsers leave incriminating Evidence?: a forensic analysis

- of residual artifacts from private and portable web browsing sessions. *EURASIP J, on Inf. S.* 201(6), 1-13
- Paragon Disk Wiper, retrieved January 2015 from <http://www.paragon-software.com/home/dw-professional/download.html>
- PortableApps. (2013). retrieved March 2015 from <http://portableapps.com/>
- Ruff, N. (2008). Windows Memory Forensics. *Journal in Computer Virology*, 14, 83-100.
- Said, H., Mutawa, A.H., Awadhi, A.I., Guimaraes, M. (2011). Forensic analysis of private browsing artifacts. *International Conference on Innovations in Information Technology (IIT)*.
- Satvat, K., Forshaw, M., Hao, F. and Toreini E. (2014). On the Privacy of Private Browsing – A Forensic approach. *Journal of Information Security and Application*, 19, 88-100.
- Simons, M. and Slay, J. (2009). Enhancement of Forensics Computing Investigations through Memory Forensics Techniques. *International Conference on Availability, Reliability and Security*.
- Volatility Foundation
<http://www.volatilityfoundation.org/>
- WinHex: retrieved January 2015 from <http://www.x-ways.net/winhex/>