2014

Forensic examination and analysis of the Prefetch files on the banking Trojan malware incidents

Andri P. Heriyanto
Edith Cowan University

DOI: 10.4225/75/57b40250fb894

12th Australian Digital Forensics Conference. Held on the 1-3 December, 2014 at Edith Cowan University, Joondalup Campus, Perth, Western Australia.

FORENSIC EXAMINATION AND ANALYSIS OF THE PREFETCH FILES ON THE BANKING TROJAN MALWARE INCIDENTS

Andri P Heriyanto
School of Computer and Security Science, Edith Cowan University, Perth, Australia
aheriyan@our.ecu.edu.au

Abstract

Whenever a program runs within the operating system, there will be data or artefacts created on the system. This condition applies to the malicious software (malware). Although they intend to obscure their presence on the system with anti-forensic techniques, still they have to run on the victim's system to acquire their objective. Modern malware creates a significant challenge to the digital forensic community since they are being designed to leave limited traces and misdirect the examiner. Therefore, every examiner should consider performing all the forensics approaches such as memory forensic, live-response and Windows file analysis in the related malware incidents to acquire all the potential evidence on a victim's system. There is a challenge when an examiner only has an option to perform post-mortem forensic approach. It leads to a question: what is a forensic examination and analysis that available to obtain evidence in such incidents? The paper shows how the Prefetching process works on a system, common characteristics and the differences in the Prefetching process related to the various versions of Windows. Thus, the paper shows how the Prefetch files contain the evidentiary value which could answer what, how, where and when the banking Trojan malware infects the system. Finally, the paper shows that forensic examination and analysis of the Prefetch files can find the data remnants of banking Trojan malware incidents.

Keywords
Prefetch Files, Post-Mortem Forensics, Forensic Examination and Analysis, Banking Trojan Malware

INTRODUCTION

In essence, according to Locard's exchange principle, any interactions or contacts between two entities will result in exchange of material (Carvey, 2012). This principle applies to the digital forensic. As an example, when a user interacts with the system, there are traces of this activity, whether the user logs in locally or accesses the system remotely. The same condition happens whenever a program runs within the operating system there will be data created on the system. Many of these data or artefacts will exist only for a short time and some may persist until the system is rebooted. Other artefacts will persist well after the system is shut down and rebooted. Whatever the type of artefact is, at least one artefact will always be created (Carvey, 2012).

As operating systems advanced, paradoxically their user interface has an aim to be simple so that computers could be used easily by users with few computer skills. With the aim of ease of use, the operating system had to collect even more information about the user, such as their actions, preferences, and credentials. The result of such data storage is an environment that is loaded with artefacts, which take the form of logs, files, lists, passwords, caches, history, recently used lists, and other data. As a general category or label for this type of data or information is operating system artefacts. Most importantly, the digital examiner can use this data as evidence to identify users and their computing activities (Bunting, 2007).

Carvey (2009) suggests using the three different approaches in dealing with the Windows artefacts based on the order of volatility and other certain circumstances. The first approach is conducting the Windows memory analysis, the second approach is conducting the live response, and the third is conducting the Windows file analysis. The first and the second approach are used for analysis of volatile memory and the third approach is mainly used to analyse non-volatile memory or known as post-mortem forensics. However, there is a significant challenge for the examiner in dealing with modern malware since it is being designed to leave limited traces on the compromised host and to misdirect the forensics examiner. However, every examiner should perform a thorough and robust examination that might include all the approaches to extract the maximum amount of information relating to the malware incidents (Malin, Aquilina, & Casey, 2012).

Heriyanto (2012) reveals that the volatile memory forensics is the most effective approach in comparison with live-response and Windows registry analysis on banking Trojan malware incidents. The question is arise when examiner only has an option to perform post-mortem forensic approach: what technique is available and can be

used to obtain data remnants in such incidents? Hence, the main objective of the research is to propose the Prefetch file analysis as part of post-mortem forensic approach in banking Trojan malware incidents. Futhermore, the paper shows the process and comparison of the Prefetch files on various Windows OS and the evidentiary value of the Prefetch files as digital evidence.

RELATED WORKS

There is research and work related to the Prefetching process and the Windows Prefetch analysis in regard to the digital forensic procedure and process. Tank and Williams (2008) examined the information from the Prefetch folder in the case of U3 smart drive that may assist in forensic investigation. The work shows that the Prefetch folder can proof that the U3 devised has been used and when it was used on the target machines. Thus, it shows what software has been executed from U3 smart drive, at what time and what files has been created or modified or saved to U3 drive.

Harrel (2010) (2011a) (2011b) has analysed three different exploits including CVE-2010-2883 (PDF Cooltype) Vulnerability, CVE-2010-0094 (RMIConnectionImpl) and CVE 2010-1885 (Windows Help Center URL Validation Vulnerability). Results show that potential artefacts can be found on Windows Prefetch files that related with the presence of the three exploits on the victim's machine. As an addition, Harell (2012) found the advancement of NTOSBOOT as one of the Prefetch file on the malware investigation process.

There are softwares and techniques which claimed and can be used as the anti-forensics techniques. Primarily, users wants to hide their activity or certain files on the system for avoiding the artefacts or evidence that could alleged them for such illegal activities. Pomeranz (2012) and Casey, Fellows, Geiger, and Stellatos (2011) show that the Prefetch files can reveal the artefacts on an encrypted drive (True Crypt). Zax and Adelstein (2009) finds the certain activities on the Prefetch file although someone has used the Steganography (FAUST). This presented the traces left behind after a number of freely available steganography tools were installed, run, and uninstalled. Tilbury (2009) shows that the Prefetch files could present the artefact of certain activitities on the defragmentation process as a part of an anti-forensic technique.

Geiger (2005) examined the performance of six commercial counter-forensic tools which designed to irretrievably erase files and records of computer activity to eliminate the evidence. The paper shows that the five tools have a failure area on the Prefetch files. It means that there is still data remnants of the wiped files and directory tree referenced in the Prefetch files. Geiger (2006) expanded his examination to different thirteen six commercial counter-forensic tools. The result almost the same with the previous work: most of the tools ignored the Prefetch files which still contained the information such as the full path and names of many of the files in the wiped directory.

Atkinson (2013) proposed the development of tools that remotely parsing file based forensic artifacts such as the Prefetch files. The remote parsing tool could provide many advantages including a capability called Least Frequency of Occurrence (LFO). The organisation could aggregate data from every the Prefetch files on every host in a large network and use LFO to detect any anomalies which might turn out to be a malicious activity.

PREFETCHING PROCESS AND PREFETCH FILES ON WINDOWS OS

Prefetching and SuperFetch Processing

The detail description and purpose of the Prefetching process are described below:

> The Prefetching process tries to speed the boot process and application startup by monitoring the data and code accessed by boot and application startups and using that information at the beginning of a subsequent boot or application startup to read in the code and data. The Prefetching process monitors the first 10 seconds of application startup. For boot, the Prefetching process by default traces from system start through the 30 seconds following the start the user's shell (typically Explorer), or failing that, up through 60 seconds following Windows service initialization or through 120 seconds, whichever comes first. Further optimization and Prefetching is performed by another component called SuperFetch. The SuperFetch service (which hosts the logical Prefetcher, although it is a completely separate component from the actual SuperFetch functionality) performs a call to the internal NtQuerySystemInformation system call requesting the trace data (Russinovich, Solomon, & Ionescu, 2009, p. 823).

Type and Naming Convention

According to Wade (2010), there are three types of the Prefetch files: **Boot Trace, Application** and **Hosting Application**. The naming convention is unique for each of the three types of the Prefetch files which stated above: boot trace, application, and hosting application. There is only one boot trace the Prefetch file which its name will be static: **NTOSBOOT-B00DFAAD**. NTOSBOOT is short for NT Operating System Boot, which is used by the Windows operating system when the system is booting up. This Prefetch file is always named the same with the trailing hash BAADF00D, which is used to represent uninitialized data. Thus, this is the largest of the Prefetch files in term of size.

Common Characteristics

The characteristics of the Prefetch files can contain evidentiary value for the examiners. Metz (2014) and Koepi (2013) show the common characteristics of Windows Prefetch file (.pf) on Windows XP, Windows Vista, Windows 7 and Windows 8 as shown on Table 1.

Table 1: Common Characteristic of the Prefetch Files (.pf) on Windows OS

No	Characteristics	Description
1	**Byte Order**	Little-endian
2	**Date and time values**	Filetime in UTC
3	**Character String**	Unicode strings are stored in UTF-16 little-endian without the byte order mark (BOM)
4	**Location**	C:\Windows\Prefetch\
5	**File Name (Naming Convention)**	The application and hosting application Prefetch file name, except for the extension, is commonly in upper case and structured as: <executable filename>-<Prefetch hash>.pf. Where "executable filename" is the filename of the original executable truncated to 29 characters, and "Prefetch hash" is calculated based on the original filename. The Prefetch hash value for hosting application Prefetch file has a different calculation which using the application's path of execution and the command line used to start the application.
6	**File Header**	Offset 04, length of 4 bytes SCCA (0×53, 0×43, 0×43, 0×41)
7	**Unicode filename**	Offset 16, length of 30 bytes
8	**Last executed time**	Offset 128, Length of 8 bytes (LE), Windows Filetime format.
9	**VolumeID**	Offset 108, length of 4 bytes points to the offset of section D of the Prefetch file. Volume ID is located at Offset of section D + 16 bytes, for a length of 4 bytes

Different Characteristics

In contrast, there are four variable differences among the various Windows OS as shown on the Table 2.

Table 2: Differences of Characteristic of the Prefetch Files (.pf) (.pf) on Windows OS

No	Variables or Condition	Description
1	**Format version**	Value 17 used in Windows XP and Windows 2003. Value 23 used in Windows Vista and Windows 7. Value 26 used in Windows 8.1. Every format version on each Windows version has different file information (Metz, 2014).
2	**Executed count**	On Windows XP: Offset 144, length of 4 bytes (LE); Windows Vista: Offset 152, length of 4 bytes (LE); Windows 7: Offset 152 length of 4 bytes (LE) and Windows 8: Offset 208, length of 4 bytes (LE) (Koepi, 2013)
3	**Last Access Timestamp Count**	According to Atkinson (2013), the Prefetching process captured the last 8 executed time starting at offset 128. It gives the examiner several additional timestamps to help build a timeline of events on a system (McQuaid, 2014).
4	**Prefetching Enabled by Default?**	By default, server systems (Windows 2003, 2008, 2008R2) have boot Prefetching enabled only, whilst workstation systems (XP, Vista, Win7)

		have application Prefetching enabled as well (Carvey, 2012). In Windows 7, Microsoft automatically disabling Superfetch and Prefetch when a fast SSD was detected. In Windows 8, however, the operating system tries to analyse the performance characteristics of the system's storage and intelligently enable or disable Superfetch/Prefetch as needed (Tanous, 2014).
5	**Amount of PF Files**	At any given time the system can keep up to 128 (Windows XP/2003/Vista/7/2008) or 1024 (Windows 8/8.1/2012) individual Prefetch files (Each one correlates to a single application) (Atkinson, 2013).

Absense of the Prefetch Files

Before conducting the examination of the Prefetch files, the examiner should examine the certain configuration on the victim's system to identify whether the system has been enabled the Prefetching process. At first, the examiner should examine the status of enable/disable of the Prefetch file. The Prefetcher behavior is controlled by the Windows registry value "EnablePrefetcher" located in the following registry path: *HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SessionManager\MemoryManagement\PrefetchParameters*. The value for "EnablePrefetcher" can have one of the following values: "0" means "disabled", "1" means "application launch Prefetching enabled", "2" means "boot Prefetching enabled" and "3" means "application launch and boot enabled (default)"(LLC, 2013). Secondly, the examiner should examine the Enable/Disable Superfetch service on *Start > Control Panel > System and Security > Administrative Tools*; Select Services; Double-click the Superfetch service; on General Tab check the startup type.

BANKING TROJAN MALWARE

Overview

Banking Trojan malware is another variant of Trojan Horse malware that the main objective is to steal the private data of online bank application such as system information, passwords, banking credentials or other financial details. This malware uses many techniques or schemes to infect the target which includes email phishing, drive-by-download or could be from fake Microsoft Word which containing malicious VBA macros. After infecting the target, the banking Trojan can steal the bank credential by man-in-the middle browser attack, encrypt the stolen information and send it to the attacker's specified servers or known as the command-and-control (C&C) server. Finally, the cybercriminal can launch their main objective: make a financial transaction on behalf the user bank account and send the money to the mule account (Donohue, 2013; Neagu, 2014).

FORENSIC EXAMINATION AND ANALYSIS

Definition

The paper uses the examination terminology as a process to extract and prepare data for analysis. The analysis terminology is used to express the process that involves critical thingking, assessment, experimentation, fusion, correlation, and validation to gain an understanding of and reach conclusion abouth the incident on the basis of available evidence. In general, the aim of analysis proces is to gain insight into what happenned, where, when and how, who was involved and why (Casey, 2011).

Characteristics of Malware and Indirect Artefact

Carvey (2013) suggests the understanding of four characteristics of malware to detect their presence. The first is an initial infection vector (IIV). It refers to how the malware originally made it's way on to the victim's system. Second is the propagation mechanism. This characteristic refers to how the malware moves between system , if it does. Third is the artefacts. According to Kornblum (2006), rootkits want to remain hidden and at the same time they want to run. It called the **Rootkit Paradox**. The same condition might be applied on the other malware. Therefore, the malware interact with their environment and it will leave the artefacts on the victim's system. The fourth is a persistence mechanism. It refers that malware utilizes to survive during the reboots. Moreover, the persistence mechanism is also an artefact of the related malware.

According to Carvey (2012), there are two types of artefacts that can be found during the examination: direct and indirect artefacts. A direct artefact is something as a direct result of an incident, such as malware infection

or an intrusion. An indirect artefact is something as a result of the ecosystem or environment in which the incident occur and is not a direct result of the incident. Based on the Prefetching process, it can be concluded that Prefetch files are indirect artefacts since they can be created by the Windows operating system during the course of an incident.

Methodology, Tools and Test Environment

Methodology

The research uses post-mortem forensic approach whereby the data remnants will be examined and analysed after the system has been shut down. There are three different conditions were deployed for the Prefetch files examination and analysis to acquire more robust findings:

1. Before the system has been shut down;
2. After the system has been shut down; and
3. After all the Prefetch files on C:\Windows\Prefetch were deleted, emptied the recycle bin and shutting down the system.

The main purpose of this methodology is to answer the question whether the Prefetch files still consists the pertinent data remnants after the system has been shut down and even after the Prefetch files has been deleted.

Forensic Tools

Three forensic tools have been used for examining and analysing the Prefetch files on the system:

1. The WinPrefetchView v1.12;
2. The Encase ver. 6 with two EnScripts: PFDump (v2.5.0) and Find & Parse Prefetch Files in Unallocated Clusters;
3. The X-Ways ver.17.

Test Environment

As the victim's system, the work uses the several applications:

1. VMWare Workstation ver 9.0.2;
2. Virtual machine with Windows 7 Ultimate SP1 32-bit, RAM: 2048 MB; and
3. Virtual machine with Windows XP Professional version 2002 SP 3 32-bit, RAM: 2048 MB.

Detail information and source of malware samples is shown on Table 4.

Table 4: Detail Information of the Malware Samples with the Link Source

No	Malware Samples	Link Source
1	Cridex Banking Trojan MD5: e92de5cc06a361575d24adbde4bf0e81 SHA1: 29fc820e7e989f961cf7eab24a4f553488a60307	http://oc.gtisc.gatech.edu:8080/search.cgi?s earch=cridex
2	ZeuS Banking Trojan MD5: fb4d991644686160625eafe0c589392b SHA1: 944810e76932d83e338d25711175fc66903c8c0a	http://oc.gtisc.gatech.edu:8080/search.cgi?s earch=zeus
3	SpyEye Banking Trojan MD5: 79ac48be8de57d54764fdd22c0fe3f16 SHA1: 38f0f5d3849e78a1e0fb6f83e9fedf8f45d1cffb	http://oc.gtisc.gatech.edu:8080/search.cgi?s earch

Results

The result of examination process with certain the forensic tools is shown on Table 5. The result of examinations and analysis shows that there is no different condition before and after shut down of the system. Therefore, the post-mortem forensics on the Prefetch files has the same result with the live response approach. On the third condition where all the Prefetch files have been deleted, emptied the recycle bin and shut down the system, the only artefact of the malware only resisted on the NTOSBOOT file. This results are persist on all of the three banking Trojan malware incidents.

In accordance with the four characteristics of malware, the results show that the Prefetch files contain the data remnants such as the initial infection vector (IIV) which is show how the malware originally infected the system, the artefacts which is the Prefetch files itself, and the persistence mechanism. On Zeus incident, there is

the application Prefetch file named MALWARE.EXE-1EEA6A1B.pf with its detail application named MALWARE.EXE and NTOS.EXE. This Prefetch files contain evidentiary value information such as created time, process EXE, process path, run counter, last run time, full path and device path. Although those the Prefetch files have been deleted, there is persistence mechanism of the malware that persists on the boot trace Prefetch files named NTOSBOOT-B00DFAAD.pf. This Prefetch file contains the file named NTOS.EXE with its information such as full path and device path.

On Cridex incident, the result shows two application Prefetch files named: MALWARE.EXE-CE4FE371.pf and KB00062397.EXE-F8DC7213.pf with detail application named KB00062397.EXE. The persistence mechanism of the malware could be found on the NTOSBOOT-B00DFAAD.pf with the file named KB00062397.EXE. On SpyEye incident, the result show the Prefetch file named NTVDM.EXE-1A10A423.pf with its detail application named MALWARE.EXE. The persistence mechanism of the malware has shown on the NTOSBOOT-B00DFAAD.pf with the file named NTVDM.EXE.

The forensic examination and analysis results of the Prefetch files on banking Trojan malware incidents are aligned with the results from volatile memory forensic and live-response approach on the previous work (Heriyanto, 2012). Particularly the data remnant of NTOS.EXE on Zeus incident and KB00062397.EXE on Cridex incident. The only inconsistency occured on the SpyEye incident. On the previous work, the result shows the data remnant of CLEANSWEEP.EXE on the victim's system. Instead, on the current work, the artefacts of the malware is shown by NTVDM.EXE.

Table 5: Forensic Examination Results from WinPrefetchView v1.12

Malware	After Shut Down	After Deletion and Shut Down
Zeus	*a. Application Prefetch file:* ========== Filename : MALWARE.EXE-1EEA6A1B.pf Created Time : 7/1/2014 5:11:15 AM Modified Time : 7/1/2014 5:11:15 AM File Size : 16,546 Process EXE : MALWARE.EXE Process Path : C:\DOCUME~1\COMPUTER\LOCALS~1\TEMP\TEMPORARY DIRECTORY 1 FOR ZEUS MALWARE SAMPLE.ZIP\MALWARE.EXE Run Counter : 1 Last Run Time : 7/1/2014 5:11:14 AM ========== *b. Detail of application Prefetch files:* ========== Filename : MALWARE.EXE Full Path : C:\DOCUME~1\COMPUTER\LOCALS~1\TEMP\TEMPORARY DIRECTORY 1 FOR ZEUS MALWARE SAMPLE.ZIP\MALWARE.EXE Device Path : \DEVICE\HARDDISKVOLUME1\DOCUME~1\COMPUTER\LOCALS~1\TEMP\TEMPORARY DIRECTORY 1 FOR ZEUS MALWARE SAMPLE.ZIP\MALWARE.EXE Index : 5 ========== ========== Filename : NTOS.EXE Full Path : C:\WINDOWS\system32\sortkey.nls Device Path : \DEVICE\HARDDISKVOLUME1\WINDOWS\SYSTEM32\NTOS.EXE Index : 23 ==========	*a. Boot trace Prefetch file:* ========== Filename : NTOSBOOT-B00DFAAD.pf Created Time : 7/1/2014 5:57:56 AM Modified Time : 7/1/2014 5:57:56 AM File Size : 385,078 Process EXE : Process Path : Run Counter : 1 Last Run Time : 7/1/2014 5:56:20 AM ========== *b. Detail of boot trace Prefetch file:* ========== Filename : NTOS.EXE Full Path : C:\WINDOWS\system32\drivers\kmixer.sys Device Path : \DEVICE\HARDDISKVOLUME1\WINDOWS\SYSTEM32\NTOS.EXE Index : 362 ==========
Cridex	*a. Application Prefetch file:* ========== Filename : MALWARE.EXE-CE4FE371.pf Created Time : 6/30/2014 2:57:19 AM Modified Time : 6/30/2014 2:57:19 AM File Size : 16,050 Process EXE : MALWARE.EXE Process Path : C:\USERS\~\APPDATA\LOCAL\TEMP\TEMP1_CRIDEX MALWARE SAMPLE.ZIP\MALWARE.EXE Run Counter : 1	*a. Boot trace Prefetch file:* ========== Filename : NTOSBOOT-B00DFAAD.pf Created Time : 6/30/2014 9:40:30 AM Modified Time : 6/30/2014 9:40:30 AM File Size : 735,282 Process EXE : Process Path : Run Counter : 0 Last Run Time : 7/1/2014 5:56:20 AM

	Last Run Time : 6/30/2014 2:57:17 AM ========== ========== Filename : KB00062397.EXE-F8DC7213.pf Created Time : 6/30/2014 2:57:19 AM Modified Time : 6/30/2014 2:57:19 AM File Size : 12,754 Process EXE : KB00062397.EXE Process Path : C C:\Users\~\AppData\Roaming\KB00062397.EXE Run Counter : 1 Last Run Time : 6/30/2014 2:57:18 AM *b. Detail of application Prefetch files:* ========== Filename : KB00062397.EXE Full Path : C:\Users\~\AppData\Roaming\KB00062397.EXE Device Path : \DEVICE\HARDDISKVOLUME1\USERS\~\APPDATA\ROAMING\KB00062397.EXE Index : 5 ==========	*b. Detail of boot trace Prefetch file:* ========== Filename : KB00062397.EXE Full Path : C:\Users\~\AppData\Roaming\KB00062397.EXE Device Path : \DEVICE\HARDDISKVOLUME1\USERS\~\APPDATA\ROAMING\KB00062397.EXE Index : 683 ==========
SpyEye	*a. Application Prefetch file:* ========== Filename : NTVDM.EXE-1A10A423.pf Created Time : 12/14/2012 3:35:24 PM Modified Time : 7/1/2014 6:50:28 AM File Size : 19,842 Process EXE : NTVDM.EXE Process Path : C:\WINDOWS\system32\ntvdm.exe Run Counter : 3 Last Run Time : 7/1/2014 6:50:20 AM b. *Detail of application Prefetch files* ========== Filename : MALWARE.EXE Full Path : C:\Malware\SpyEye Malware Sample\SpyEye Malware Sample\malware.exe Device Path : \DEVICE\HARDDISKVOLUME1\MALWARE\SPYEYE~1\SPYEYE~1\MALWARE.EXE Index : 33 ==========	*a. Boot trace Prefetch file:* ========== Filename : NTOSBOOT-B00DFAAD.pf Created Time : 7/1/2014 7:42:09 AM Modified Time : 7/1/2014 7:42:09 AM File Size : 380,716 Process EXE : Process Path : Run Counter : 1 Last Run Time : 7/1/2014 7:40:35 AM *b. Detail of boot trace Prefetch file:* ========== Filename : NTVDM.EXE Full Path : C:\WINDOWS\system32\ntvdm.exe Device Path : \DEVICE\HARDDISKVOLUME1\WINDOWS\SYSTEM32\NTVDM.EXE Index : 99 ==========

Recovery the deleted Prefetch files

Every examiner might consider to recover the deleted Prefetch files to find any potential data remnants regarding with the incident. The work also examine the recovery process of deleted Prefetch files on the test environment. The forensic tools use the search strings based on the file header which is *SCCA (0×53, 0×43, 0×43, 0×41)* on allocated and unallocated cluster and parsing them out. On the Zeus incident, Encase with its PFDump Enscript could parsing the application Prefetch file named MALWARE.EXE. The same result is persist on the Cridex incident, the PFDump Enscript on Encase tools could parsing two application Prefetch files named MALWARE.EXE and KB00062397.EXE. Therefore, it has been suggested for every examiner to recover any deleted Prefetch files that might be relevant with the incident.

CONCLUSION

Heriyanto (2012) uses the live-response approach, the memory forensic approach and Windows live analysis approach to investigate the banking Trojan malware incidents to find what is the proper forensics approach for such incidents. Although all three approaches can find the related data of interest, the work reveals that memory forensic approach can obtain the most robust findings. On the other hand, the forensic examination and analysis result of the Prefetch files are consistent with and can support the result findings from the previous work. Furthermore, the paper has demonstrated the Prefetch files as the indirect artefact on Banking Trojan malware incidents and has evidentiary value as the digital evidence on the related incidents. Finally, the paper shows that the examiners can conduct forensic examination and analysis of the Prefetch files if they only have an option to perform post-mortem forensic in banking Trojan malware incidents.

However, the examiner should be aware that the Prefetching process on the system can be disabled by users or disabled by the default setting on Windows 7 with SSD and could be disabled on Windows 8 with SSD. This setting will create the absense of the Prefetching process and the Prefetch files on the system. Although the Prefetch file is not the only source of evidence on the Windows file analysis, but the paper shows the advancement and significant of the Prefetch files on the banking Trojan malware incidents.

REFERENCES

Atkinson, Jared. (2013). What's New in the Prefetch for Windows 8?? , from http://www.invoke-ir.com/2013/09/whats-new-in-Prefetch-for-windows-8.html

Bunting, Steve. (2007). EnCase Computer Forensics: The Official EnCE: EnCase Certified Examiner Study Guide (Second ed.): Wiley.

Carvey, Harlan. (2009). Windows forensic analysis DVD toolkit: Syngress.

Carvey, Harlan. (2012). Windows Forensic Analysis Toolkit: Advanced Analysis Techniques for Windows 7: Elsvier.

Carvey, Harlan. (2013). HowTo: Malware Detection, pt I. Retrieved 03/08/2014, 2014, from http://windowsir.blogspot.com.au/2013/07/howto-malware-detection-pt-i.html

Casey, Eoghan, Fellows, Geoff, Geiger, Matthew, & Stellatos, Gerasimos. (2011). The growing impact of full disk encryption on digital forensics. digital investigation, 8(2), 129-134. doi: http://dx.doi.org/10.1016/j.diin.2011.09.005

Community, CybOX. (n.a.). Main schema Win_Prefetch_Object.xsd. from https://cybox.mitre.org/language/version2.0.1/xsddocs/objects/Win_Prefetch_Object_xsd.html

Donohue, Brian. (2013). The Big Four Banking Trojans. from http://blog.kaspersky.com/the-big-four-banking-trojans/

Geiger, M. (2005). Evaluating commercial counter-forensic tools. Paper presented at the Proceedings of the 5th Annual Digital Forensic Research Workshop.

Geiger, M. (2006). Counter-forensic tools: Analysis and data recovery. Paper presented at the 18th FIRST Conference.

Harell, Corey. (2012). NTOSBOOT Prefetch File. from http://journeyintoir.blogspot.com.au/2012/12/ntosboot-Prefetch-file.html

Harrel, Corey. (2010). CVE 2010-1885 (Windows Help Center URL Validation Vulnerability) Exploit Artifacts. from http://journeyintoir.blogspot.com.au/2010/12/cve-2010-1885-windows-help-center-url.html

Harrell, Corey. (2011a). CVE-2010-0094 (RMIConnectionImpl) Exploit Artifacts. 2014, from http://journeyintoir.blogspot.com.au/2011/03/cve-2010-0094-rmiconnectionimpl-exploit.html

Harrell, Corey. (2011b). Exploit Artifacts for CVE-2010-2883 (PDF Cooltype) Vulnerability. 2014, from http://journeyintoir.blogspot.com.au/2011/01/cve-2010-2883-pdf-cooltype-exploit.html

Heriyanto, Andri P. (2012). What is the Proper Forensics Approach on Trojan Banking Malware Incidents?

Koepi, David. (2013). Prefetch Forensic. from http://davidkoepi.wordpress.com/tag/windows-8/

Kornblum, Jesse D. (2006). Exploiting the Rootkit Paradox with Windows Memory Analysis International Journal of Digital Evidence, 5(1).

LLC, TZWorks. (2013). Windows Prefetch Parser (pf). from https://tzworks.net/prototype_page.php?proto_id=1

Malin, Cameron H, Aquilina, James M, & Casey, Eoghan. (2012). Malware Forensics Field Guide for Windows Systems: Digital Forensics Field Guides: Elsevier.

McQuaid, Jamie. (2014). Forensic Analysis of Prefetch files in Windows. 2014, from http://www.magnetforensics.com/forensic-analysis-of-Prefetch-files-in-windows/

Metz, Joachim. (2014). Windows Prefetch File (PF) format: Analysis of SCCA. https://a7d066c1b08934631d956e84d1dcc86e6397e428.googledrive.com/host/0B3fBvzttpiiSbl9XZGZzQ05hZkU/Windows%20Prefetch%20File%20(PF)%20format.pdf

Neagu, Aurelian. (2014). The Top 10 Most Dangerous Malware That Can Empty Your Bank Account. Retrieved 16/11/2014, 2014, from Neagu, Aurelian. (2014). The Top 10 Most Dangerous Malware That Can Empty Your Bank Account. Retrieved 16/11/2014, 2014, from https://heimdalsecurity.com/blog/top-financial-malware/

Pomeranz, Hal. (2012). Tales From the Crypt! No Keys? No Problem! http://computer-forensics.sans.org/summit-archives/2012/tales-from-the-crypt-truecrypt-analysis.pdf

Tank, Ravi, & Williams, Patricia AH. (2008). The Impact of U3 Devices on Forensic Analysis.

Tanous, Jim. (2014). How to Disable Superfetch and Prefetch in Windows 8. from http://www.tekrevue.com/tip/disable-superfetch-Prefetch-windows-8/

Tilbury, Chad. (2009). De-mystifying Defrag: Identifying When Defrag Has Been Used for Anti-Forensics (Part 1 - Windows XP). from computer-forensics.sans.org/blog/2009/08/05/de-mystifying-defrag-identifying-when-defrag-has-been-used-for-anti-forensics-part-1-windows-xp/

Wade, Mark A. (2010). Decoding Prefetch Files for Forensic Purposes. http://download.harris.com/app/public_download.asp?fid=2325

Zax, R., & Adelstein, F. (2009). FAUST: Forensic artifacts of uninstalled steganography tools. digital investigation, 6(1), 25-38.