Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering -- CSEE 2014
Copyright © Institute of Research Engineers and Doctors. All rights reserved.
ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-34

Algorithm for Tracking Sensitive Information of Online Application in Computer Memory

Khairul Akram Zainol Ariffin, Ahmad Kamil Mahmood, Jafreezal Jaafar, Solahuddin Shamsuddin

Abstract— **With the advance in technology and the introduction of cloud computing, the usage of Internet application has been increased rapidly. In addition, the online storages with password protected such as Dropbox, Box Sync, Copy and Cloud Me are already available to all users. Hence, with a rapid development and introduction of this technology, the analysis on the hard drive has become obsolete in retrieving the information from those applications. The aim of this paper is to present an algorithm to track the sensitive information from online application for both active and exile process in the computer memory. The algorithm encompasses the signature search to find the possible processes in the memory, obtain the user and machine data and then some sensitive information with regard to the online application. The algorithm will be independent of address translation algorithm that has been frequently applied in the retrieving technique in the past.**

Keywords— Algorithms, Information Retrieval, Digital Forensics, Memory Analysis, Signature Search

I. Introduction

In these modern days, computer system has been an important part of our daily life. The introduction of computer and the technology within it give a benefit to the human and ease our daily routine. However, as it brings a bright advantage to us, it also introduces a negative effect such as cybercrime to our world. A report from ITU had stated that there were about 2, 749 millions of Internet users over March 2013 and out of that amount, it was expected about 625 million owned the personal cloud storage [1], [2]. As the amount of online users has been increasing over the years, there is a possibility that the quantum of cybercrime victim will be increased as well as the cybercriminal since the Internet has eased the method to conduct the cybercrime. In addition, the introduction of online storage will overtake the responsibility of the hard disk as the primary storage of data. The cybercriminal can use this available service as a medium to store the information with regard to the crime as it is protected by the password. Hence, this facility has become a problematic for the digital forensic investigator since the traditional approach (analysis on the hard drive) is no longer applicable in this situation.

As the technology in computer system merge, a field that is known as Digital Forensic has been established to handle the problem in cybercrime. This field is responsible in colleting, preserving, analyzing, documenting and presenting the contents of computer as evidence of the cybercrime [3]. In the past, the investigation on the computer contents was only towards the non-volatile drive whilst the computer memory analysis was only applied for investigating the malware behavior [4]. Until recently, and with the merged of online storage, the investigation on computer memory has become critical as it can still hold sensitive data such as username, password and decrypted version of data. Hence, in 2005, the importance of memory analysis in Digital Forensic has been outlined at Digital Forensics Research Workshops (DFRWS) where two tools had been designed and developed that were known as Memparser [5] and KntList [6].

K.A Zainol Ariffin
Dpartment of Digital Forensic,
CyberSecurity Malaysia,
Mines, Selangor, Malaysia

A.K Mahmood
Computer Information System Department,
University Teknologi Petronas,
Perak, Malaysia

J Jaafar
Computer Information System Department,
University Teknologi Petronas,
Perak, Malaysia

S Shamsuddin
Research Department,
CyberSecurity Malaysia,
Mines, Selangor, Malaysia

II. Literature reviews

The knowledge and theories that is required for memory analysis has been outlined in Dhamdhere [7] and Rusnovich and Soloman [8]. Both books discuss about the internal structure, address translation algorithm and the procedure of object creation that is an important knowledge for tracking the data in the computer memory. Further, the study in Amari, K [9] has demonstrated the importance of kernel as it is responsible to store the objects on the pools of memory. By depending on the criticality of the objects in the computer memory, most of the objects are stored in paged pool as it allows the swapping process to the hard disk when memory is running low in space [10]. Whereas the important structure such as processes and threads are stored on the non-paged pool as the kernel needs to access them frequently. This scenario justifies that some of the important information such as running objects can be obtained from memory analysis [11].

A tool known as VADTool has been developed in 2007 where the work of it is based on tracking the status of object space that is stored in the Virtual Address Description (VAD). This internal structure is maintained by the memory manager and it stores the information on the attributes of the object such as range of the address, inheritance of child and object's security [12]. Apart from that, XORSearch [13] is a

Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering -- CSEE 2014
Copyright © Institute of Research Engineers and Doctors. All rights reserved.
ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-34

tool for memory forensic where the work is based on the string search technique. It takes a keyword as an input, and then performs the search throughout the memory dump and with additional function, this tool is able to find the keywords that have already been obfuscated.

AccessData [14] group has designed a tool that is known as Forensic Toolkit (FTK) which relies on the Process Environment Block to locate and retrieve the executable file and DLL's path from the memory. It will parse all the active processes in the memory and enumerate all the contents within them. In addition to that, Windows Memory Forensic Toolkit has stated the role of Directory Table Base (DTB) in the address translation algorithm for tracking the active processes and other linked objects [15]. In 2009, Ruichao Zhang, Lianhai Wang, Shuhui Zhang [16] had demonstrated the data extraction from memory dump by using Kernel Processor Control Region (KPCR). On the same year, S.M Hejazi, C. Talhi, M. Debbabi [17] has outlined the use of fingerprint to track username and password for email and messenger application from memory dump. Meanwhile, FTFinder [18] is a tool that applies the file carving where this technique is done linearly to recover the contiguous file. The technique is used in this tool due to the theory that the operating system will convert the file to be contiguous file instead of fragment file.

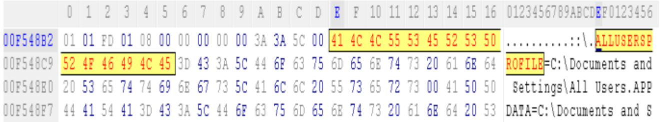
III. METHODOLOGY

The main aim of this paper is to provide an algorithm that can justify the existence of the online application that run on the machine and then listing the method to retrieve some interesting information from them. The algorithm will start by identifying the user, the architecture and some relevant information about the machine. Then, all the processes in the machine's memory are captured by using the proã signature search technique. This technique is chosen against all the other techniques that has been discussed in previous section because of less information required for process tracking and the advantage of detecting both exile and hidden process. Finally, if the online application is detected from the process list, a unique signature or protocol is applied as a string search to locate and capture the content of them. The interested contents to be captured are the uniform resource locator (url) from the tabs of internet browser, username and password. This algorithm will not include the address translation algorithm which has been frequently applied for object tracking in memory in the past. The overall work of algorithm is based on the following rules:

RULE 1: Capturing the user and the machine information.

In theory, the information about the user and machine occur a number of times in the computer memory. It acts as an identifier of the machine for most applications that run on the machine. This information is important for forensic investigation as it is not only for the validation of the machine as true exhibit, but it also gives an idea about the architecture of the machine which is critical if the analyst wants to apply address translation algorithm to retrieve objects.

The information can be obtained by searching for the ALLUSERPROFILE signature in the computer memory image. The size of this object is about 95 blocks of byte which may contain the directory, processor architecture, system root, computer name and other relevant information. Figure 1 show the representation of this signature in the memory image.



```
          0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  10 11 12 13 14 15 16 0123456789ABCDEF0123456
00F548B2  01 01 FD 01 08 00 00 00 00 00 3A 3A 5C 00 41 4C 4C 55 53 45 52 53 50 ..........::\.ALLUSERSP
00F548C9  52 4F 46 49 4C 45 3D 43 3A 5C 44 6F 63 75 6D 65 6E 74 73 20 61 6E 64 ROFILE=C:\Documents and
00F548E0  20 53 65 74 74 69 6E 67 73 5C 41 6C 6C 20 55 73 65 72 73 00 41 50 50  Settings\All Users.APP
00F548F7  44 41 54 41 3D 43 3A 5C 44 6F 63 75 6D 65 6E 74 73 20 61 6E 64 20 53 DATA=C:\Documents and S
```

Fig 1 Presentation of the signature to obtain the user and machine information

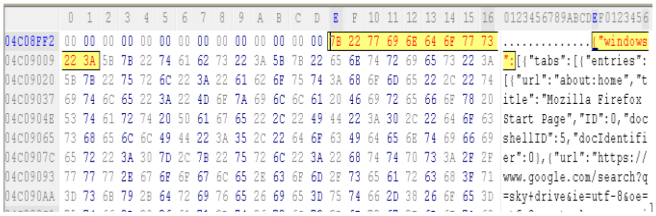
RULE 2: Identify, capture and extract the general information about all the processes in the computer memory image.

In order to identify the online application that has been run on the machine, it is necessary to capture all the processes that have been loaded on the memory. Proã signature search technique is selected for the processes tracking because of its ability to capture both active and exile processes. It starts by capturing all possible entities that may represent the process block, and then select only the true process blocks. Finally, all the processes are arranged in the Process Block Tree (PBT) according to its seniority (parent or child process). Once the PBT has been constructed, the online application such as web browser and cloud storage can be identified and categorized as active or exile process. Overall, this technique has been discussed in detail in [19], [20].

RULE 3: Identify the online application from PBT and capture the sensitive information about them.

In the normal approach, the data that links with the process block can be tracked by using address translation algorithm. This method work perfectly with the active objects, however for hidden and exile processes, the chance to retrieve the linked objects will depend on whether their pointer still resides in the memory. By depending on the size of the computer memory and the activity that is run on the machine, the availability of the pointer is random and there is a chance that new objects have overwritten it [21]. Due to this scenario, the process enumeration technique will not work perfectly and thus, in this paper the unique signature is used to capture the information of the exile online application (process).

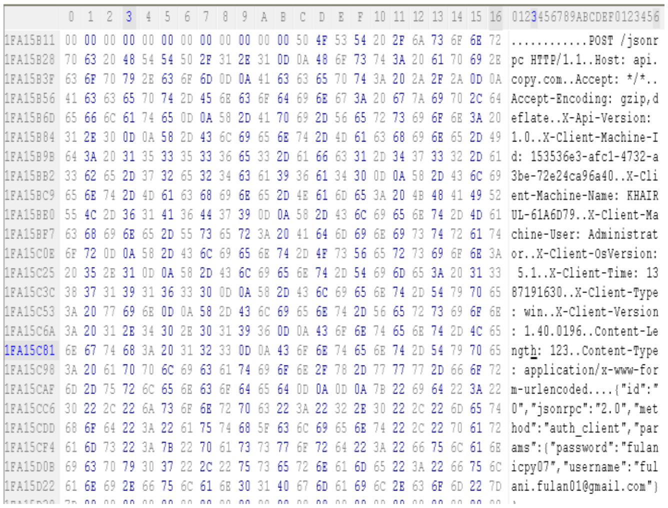
In theory, most of the tabs in Internet browser use the similar structure to store the information about the open url link. This structure can be retrieved from the computer memory by using *"window":* keyword as shown in figure 2. By searching for this keyword, the information about the tabs of Internet browser can be retrieved as the information is stored in the entry. Each of the entry will consist of the url name, ID, docShellID, doc ID, title and the referrer of the url links.



```
          0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  10 11 12 13 14 15 16  0123456789ABCDEF0123456
04C08FF2  00 00 00 00 00 00 00 00 00 00 00 00 00 00 7B 22 77 69 6E 64 6F 77 73  ..............{"windows
04C09009  22 3A 5B 7B 22 74 61 62 73 22 3A 5B 7B 22 65 6E 74 72 69 65 73 22 3A  ":[{"tabs":[{"entries":
04C09020  5B 7B 22 75 72 6C 22 3A 22 61 62 6F 75 74 3A 68 6F 6D 65 22 2C 22 74  [{"url":"about:home","t
04C09037  69 74 6C 65 22 3A 22 4D 6F 7A 69 6C 6C 61 20 46 69 72 65 66 6F 78 20  itle":"Mozilla Firefox 
04C0904E  53 74 61 72 74 20 50 61 67 65 22 2C 22 49 44 22 3A 30 2C 22 64 6F 63  Start Page","ID":0,"doc
04C09065  73 68 65 6C 6C 49 44 22 3A 35 2C 22 64 6F 63 49 64 65 6E 74 69 66 69  shellID":5,"docIdentifi
04C0907C  65 72 22 3A 30 7D 2C 7B 22 75 72 6C 22 3A 22 68 74 74 70 73 3A 2F 2F  er":0},{"url":"https://
04C09093  77 77 77 2E 67 6F 6F 67 6C 65 2E 63 6F 6D 2F 73 65 61 72 63 68 3F 71  www.google.com/search?q
04C090AA  3D 73 6B 79 2B 64 72 69 76 65 26 69 65 3D 75 74 66 2D 38 26 6F 65 3D  =sky+drive&ie=utf-8&oe=
```

Fig 2 Representation of the starting of information for Internet Browser tabs in the Windows memory image

Incidentally, the online storage can also run separately from the Internet browser. Therefore, it is not possible to capture the sensitive information such as username and password by relying only on the information of the Internet browser tabs. In general, the application will still use the HTTP or HTTPS protocol to transmit the information between server and client. If the application is using the HTTPS protocol, the process of encrypting the data that happen in the computer memory will give a slightly good percentage for tracking this sensitive information. This is due to the fact that the decrypted version of the data still exists in the memory if it is not been overwritten. This can be captured through the communication between server and client. Apart from the password and username, other important information such as ID, name and machine can also been attached together which can be used for verification purposes (verify that it come from the same machine). Figure 3 show the presentation of the username and password location on memory image for Copy application that is captured by using HTTP/1.1 fingerprint search.



```
          0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  10 11 12 13 14 15 16  0123456789ABCDEF0123456
1FA15B11  00 00 00 00 00 00 00 00 00 00 00 00 50 4F 53 54 20 2F 6A 73 6F 6E 72  ............POST /jsonr
1FA15B28  70 63 20 48 54 54 50 2F 31 2E 31 0D 0A 48 6F 73 74 3A 20 61 70 69 2E  pc HTTP/1.1..Host: api.
1FA15B3F  63 6F 70 79 2E 63 6F 6D 0D 0A 41 63 63 65 70 74 3A 20 2A 2F 2A 0D 0A  copy.com..Accept: */*..
1FA15B56  41 63 63 65 70 74 2D 45 6E 63 6F 64 69 6E 67 3A 20 67 7A 69 70 2C 64  Accept-Encoding: gzip,d
1FA15B6D  65 66 6C 61 74 65 0D 0A 58 2D 41 70 69 2D 56 65 72 73 69 6F 6E 3A 20  eflate..X-Api-Version: 
1FA15B84  31 2E 30 0D 0A 58 2D 43 6C 69 65 6E 74 2D 4D 61 63 68 69 6E 65 2D 49  1.0..X-Client-Machine-I
1FA15B9B  64 3A 20 31 35 33 35 33 36 65 33 2D 61 66 63 31 2D 34 37 33 32 2D 61  d: 153536e3-afc1-4732-a
1FA15BB2  33 62 65 2D 37 32 65 32 34 63 61 39 36 61 34 30 0D 0A 58 2D 43 6C 69  3be-72e24ca96a40..X-Cli
1FA15BC9  65 6E 74 2D 4D 61 63 68 69 6E 65 2D 4E 61 6D 65 3A 20 4B 48 41 49 52  ent-Machine-Name: KHAIR
1FA15BE0  55 4C 2D 36 31 41 36 44 37 39 0D 0A 58 2D 43 6C 69 65 6E 74 2D 4D 61  UL-61A6D79..X-Client-Ma
1FA15BF7  63 68 69 6E 65 2D 55 73 65 72 3A 20 41 64 6D 69 6E 69 73 74 72 61 74  chine-User: Administrat
1FA15C0E  6F 72 0D 0A 58 2D 43 6C 69 65 6E 74 2D 4F 73 56 65 72 73 69 6F 6E 3A  or..X-Client-OsVersion:
1FA15C25  20 35 2E 31 0D 0A 58 2D 43 6C 69 65 6E 74 2D 54 69 6D 65 3A 20 31 33   5.1..X-Client-Time: 13
1FA15C3C  38 37 31 39 31 36 33 30 0D 0A 58 2D 43 6C 69 65 6E 74 2D 54 79 70 65  87191630..X-Client-Type
1FA15C53  3A 20 77 69 6E 0D 0A 58 2D 43 6C 69 65 6E 74 2D 56 65 72 73 69 6F 6E  : win..X-Client-Version
1FA15C6A  3A 20 31 2E 34 30 2E 30 31 39 36 0D 0A 43 6F 6E 74 65 6E 74 2D 4C 65  : 1.40.0196..Content-Le
1FA15C81  6E 67 74 68 3A 20 31 32 33 0D 0A 43 6F 6E 74 65 6E 74 2D 54 79 70 65  ngth: 123..Content-Type
1FA15C98  3A 20 61 70 70 6C 69 63 61 74 69 6F 6E 2F 78 2D 77 77 77 2D 66 6F 72  : application/x-www-for
1FA15CAF  6D 2D 75 72 6C 65 6E 63 6F 64 65 64 0D 0A 0D 0A 7B 22 69 64 22 3A 22  m-urlencoded....{"id":"
1FA15CC6  30 22 2C 22 6A 73 6F 6E 72 70 63 22 3A 22 32 2E 30 22 2C 22 6D 65 74  0","jsonrpc":"2.0","met
1FA15CDD  68 6F 64 22 3A 22 61 75 74 68 5F 63 6C 69 65 6E 74 22 2C 22 70 61 72  hod":"auth_client","par
1FA15CF4  61 6D 73 22 3A 7B 22 70 61 73 73 77 6F 72 64 22 3A 22 66 75 6C 61 6E  ams":{"password":"fulan
1FA15D0B  69 63 70 79 30 37 22 2C 22 75 73 65 72 6E 61 6D 65 22 3A 22 66 75 6C  icpy07","username":"ful
1FA15D22  61 6E 69 2E 66 75 6C 61 6E 30 31 40 67 6D 61 69 6C 2E 63 6F 6D 22 7D  ani.fulan01@gmail.com"}
```

Fig 3 Representation of information for Copy (Cloud application), which includes ID, Machine name, password and username.

IV. Experiment

The algorithm was tested on Cloud_Online_Text_Process memory image which was acquired using an open source tool, DumpIt. It was an image of computer memory with XP Operating System and with the size of 512 MB. The memory image, once available will pass through the algorithm as shown in figure 4.

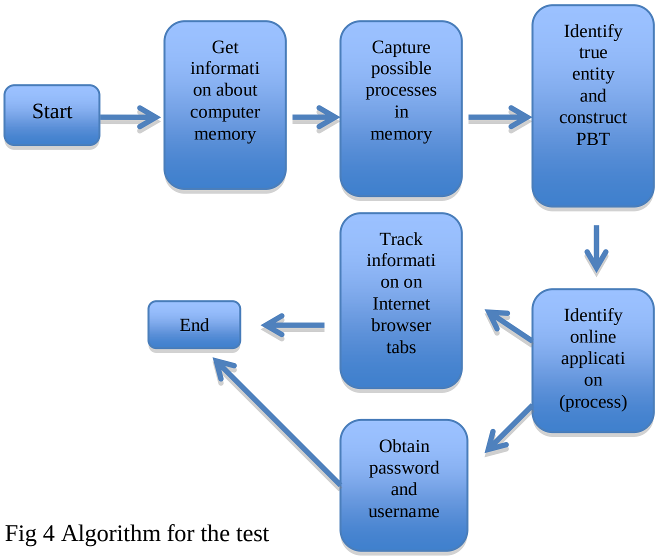

Start
Get informati on about computer memory
Capture possible processes in memory
Identify true entity and construct PBT
Identify online applicati on (process)
Track informati on on Internet browser tabs
Obtain password and username
End

Fig 4 Algorithm for the test

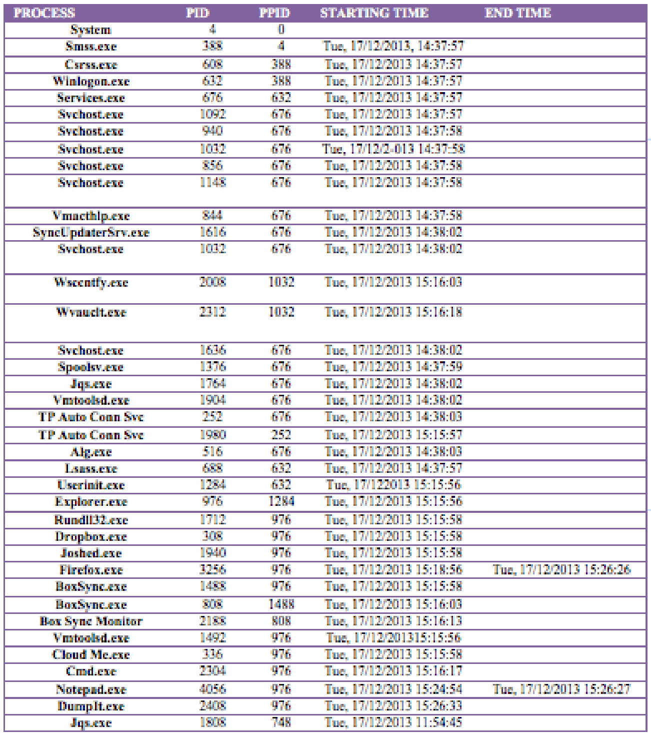
V. Result and Discussion

From the test, the information with regard to the machine has been retrieved by using the algorithm and listed in Table 1. From the table, it is noted that the memory was acquired from 32-bit system with Page Address Extension architecture after referring to the information in Processor Architecture and Identifier. Further, the computer name, operating system version and other important directory are also visible. Thus, with this knowledge, the analyst can use the process enumeration technique if he is interested in active process whilst the information such as computer name can be used as verification medium on the objects that reside in the memory so as to justify that the object come from the same source.

TABLE I
INFORMATION ABOUT USER

Structure	Information
HOMEDRIVE	C:
COMPUTER NAME	KHAIRUL-61A6D79
OS	Window_NT
SystemRoot	C:\WINDOWS
PROCESS ARCHITECTURE	x86
PROCESSOR IDENTIFIER	X86 FAMILY 6 MODEL 58 STEPPING 9, GENUINE INTEL

By referring to figure 5, there were about 40 processes that have been identified and it was noted that the Internet browser, firefox.exe, had been loaded on computer at 15:18:56 on 17/12/2013 but exited on the same day at 15:26:26. Although the application had already exiled, there were separate structures that hold the data on Firefox tabs that were located at offset of 0x04c09f15 (H), 0x048a011 (H), 0x1b78b084 (H) and 0xe9cc06f (H). There were about 69 entries that hold the information about the tabs of Firefox and in Table II, it listed one of the entries that stored the information on the Gmail Inbox. By referring to Table II, the information about the Gmail Inbox such as the title, contents and ID can be obtained directly. In addition, the username and password for the Box Sync can be obtained at offset of 0x08bc1c28 where the information about it is attached in figure 6.



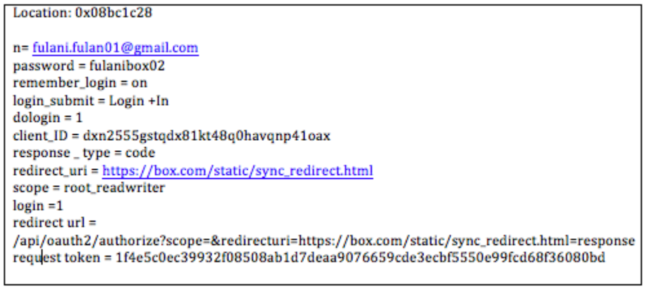
PROCESS	PID	PPID	STARTING TIME	END TIME
System	4	0		
Smss.exe	388	4	Tue, 17/12/2013, 14:37:57	
Csrss.exe	608	388	Tue, 17/12/2013 14:37:57	
Winlogon.exe	632	388	Tue, 17/12/2013 14:37:57	
Services.exe	676	632	Tue, 17/12/2013 14:37:57	
Svchost.exe	1092	676	Tue, 17/12/2013 14:37:57	
Svchost.exe	940	676	Tue, 17/12/2013 14:37:58	
Svchost.exe	1032	676	Tue, 17/12/2-013 14:37:58	
Svchost.exe	856	676	Tue, 17/12/2013 14:37:58	
Svchost.exe	1148	676	Tue, 17/12/2013 14:37:58	
Vmacthlp.exe	844	676	Tue, 17/12/2013 14:37:58	
SyncUpdaterSrv.exe	1616	676	Tue, 17/12/2013 14:38:02	
Svchost.exe	1032	676	Tue, 17/12/2013 14:38:02	
Wsccntfy.exe	2008	1032	Tue, 17/12/2013 15:16:03	
Wvuaclt.exe	2312	1032	Tue, 17/12/2013 15:16:18	
Svchost.exe	1636	676	Tue, 17/12/2013 14:38:02	
Spoolsv.exe	1376	676	Tue, 17/12/2013 14:37:59	
Jqs.exe	1764	676	Tue, 17/12/2013 14:38:02	
Vmtoolsd.exe	1904	676	Tue, 17/12/2013 14:38:02	
TP Auto Conn Svc	252	676	Tue, 17/12/2013 14:38:03	
TP Auto Conn Svc	1980	252	Tue, 17/12/2013 15:15:57	
Alg.exe	516	676	Tue, 17/12/2013 14:38:03	
Lsass.exe	688	632	Tue, 17/12/2013 14:37:57	
Userinit.exe	1284	632	Tue, 17/122013 15:15:56	
Explorer.exe	976	1284	Tue, 17/12/2013 15:15:56	
Rundll32.exe	1712	976	Tue, 17/12/2013 15:15:58	
Dropbox.exe	308	976	Tue, 17/12/2013 15:15:58	
Joshed.exe	1940	976	Tue, 17/12/2013 15:15:58	
Firefox.exe	3256	976	Tue, 17/12/2013 15:18:56	Tue, 17/12/2013 15:26:26
BoxSync.exe	1488	976	Tue, 17/12/2013 15:15:58	
BoxSync.exe	808	1488	Tue, 17/12/2013 15:16:03	
Box Sync Monitor	2188	808	Tue, 17/12/2013 15:16:13	
Vmtoolsd.exe	1492	976	Tue, 17/12/201315:15:56	
Cloud Me.exe	336	976	Tue, 17/12/2013 15:15:58	
Cmd.exe	2304	976	Tue, 17/12/2013 15:16:17	
Notepad.exe	4056	976	Tue, 17/12/2013 15:24:54	Tue, 17/12/2013 15:26:27
DumpIt.exe	2408	976	Tue, 17/12/2013 15:26:33	
Jqs.exe	1808	748	Tue, 17/12/2013 11:54:45	

Fig 5 PBT for Cloud_Online_Text_Process memory image

TABLE II
INFORMATION ABOUT GMAIL INBOX

Structure	Information
Url	https://mail.google.com/mail/u/0/?shva=1…./14318f436105cb35
ID	42
docShellID	16
Title	Just one more step to get started on Facebook-
Contents	Formdata: Id = hist_state: inbox /14318f438d99fa09 Xpath: Scrool : "0,0" Last Accessed 0 Index: 10 hidden: false Attributes { } Image: https://mail.google.co/mail/u/0/images/favicon2.ico Storage: https://www.google.com Web::c:\5d2c8e28c93f0c0 Web::v:\21_c9c918f01 Web:c5df2c8e28c93f0c0 . . open TrueType :Software\Microsoft\Windows NT\CurrentVersion\FontLink\SystemLink

In general, the ability to retrieve this sensitive information is unpredictable and random. This is due to the nature of computer memory which is volatile. Hence, depending on the activity and the size of the memory, this information will be overwritten easily if the memory is running low in space. However, as the technology advances and the total memory storage in the computer keeps on increasing, it is necessary to take it to a new step in tracking that information as the size increase will increase the possibility of chances to retrieve them.



```
Location: 0x08bc1c28

n= fulani.fulan01@gmail.com
password = fulanibox02
remember_login = on
login_submit = Login +In
dologin = 1
client_ID = dxn2555gstqdx81kt48q0havqnp41oax
response _ type = code
redirect_uri = https://box.com/static/sync_redirect.html
scope = root_readwriter
login =1
redirect url =
/api/oauth2/authorize?scope=&redirecturi=https://box.com/static/sync_redirect.html=response
request token = 1f4e5c0ec39932f08508ab1d7deaa9076659cde3ecbf5550e99fcd68f36080bd
```

Fig 6 Contents of metadata for Box Sync in memory image

VI. Conclusion

From the result of the experiment, it shows that the algorithm can work well in retrieving information about the online application if the representative of the object still resides in the memory. The purpose of the paper is also to provide a technique that requires less expertise since the personnel involved in Digital Forensic come from different background and competency.

For future work, the study will be towards the analysis on the cookies and cache of the computer memory to obtain the sensitive information as there is a chance that those components may save some information related to the online application.

Acknowledgment

The author would like to present his gratitude towards CyberSecurity Malaysia and Universiti Teknologi Petronas for the help and support that had been accorded during the research period.

References

[1] Internet World Stats: Usage and Population Statistics (published report), www.internetworldstats.com/emarketing.htm, Jan 14, 2014.

[2] ElephantDrive, Cloud storage subscription growth statistics are huge, Sept 2012.

[3] Hill, C.E., *"What is the Definition of Digital Forensics? "*, in *eHow, How to do just about everything*(Unpublished work sytle). Unpublished

[4] Jesee, K., "*Using every part of the buffalo in Windows memory analysis*(Published Journal style)". Journal Digital Investigation, vol **4**: pp. 24-29, 2007.

[5] DFRWS." *Memparser Analysis Tool by Chris Betz".*(Unpublished work sytle). Unpublished

[6] DFRWS. "*Kntlist Analysis Tool by George M. Garner Jr."* . (Unpublished work sytle). unpublished

ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-34

[7] Dhamdhere, D.M., "*Operating Systems: A Concept based Approach*."(Book style) 1 ed., McGrawHill, 2009.

[8] Russinovich, M.E., D.A. Solomon, and A. Ionescu, "*Windows®Internals Covering Windows Server® 2008 and Windows Vista®*" (Book style). J. Pierce, Editor., Microsoft Press, 2009.

[9] Amari, K., "*Techniques and Tools for Recovering and Analyzing Data from Volatile Memory*"(unpublished work style), SANS Institute, 2009.

[10] Carrier, B.G., J.," *A Hardware Based Memory Acquisition Procedure for Digital Investigations"*. (Published Journals style). Journal of Digital Investigation, 2004, March.

[11] Schuster, A., "*Searching for processes and threads in Microsoft Windows memory dump"*(Published Journal style).Journal Digital Investigation, vol **3,** pp. 10-16, 2006.

[12] Dolan-Gavitt, B.," *The VAD tree: A process eye view of physical memory"*(Published Journal style).Journal Digital Investigation, pp. s62-s64, 2007

[13] Stevens, D."*XORSearch*", (unpublished work style).2007, January 30.

[14] AccessData Corporation, " *Importance of memory Search and Analysis*" (Published White Paper) Lindon, UT, 2006.

[15] Burdach, M. "*An Introduction to Windows memory forensic"*(unpublished work style). Unpublished.

[16] Ruichao Zhang, L. W., Shuhui Zhang. "*Windows Memory Analysis Based on KPC"*(Published Conference Proceedings sytle) . In *Proc of the 2009 Fifth International Conference on Information Assurance and Security*, IEEE, Xi'An China.

[17] S. M. Hejazi, C. T., M. Debbabi "*Extraction of forensically sensitive information from windows physical memory*".(Published Journal Style) Journal digital investigation vol 6, pp.S121–S131, 2009

[18] Schuster, A." *PTFinder"*. 2006.(Unpnlished work style).unpublished.

[19] Khairul Akram Zainol Ariffin, Ahmad Kamil Mahmood, Jafreezal Jaafar, and Solahuddin Shamsuddin, "Object Signature Search for Capturing Processes Memory," *International Journal of Computer and Communication Engineering* vol. 2, no. 6, 2013.

[20] Khairul Akram Zainol Ariffin, Ahmad Kamil Mahmood, Jafreezal Jaafar, and Solahuddin Shamsuddin, *"Holistic Approach for Memory Analysis in Windows System"*, Open Access Journal of Information System (OIJIS), ISBN: 978-979-18985-7-7, pp 687-693, 2013

[21] Garfinkel, T., Pfaff, B., Chow, J., & Rosenblum, M. "*lifetime is a systems problem* (Published Conference Proceedings style)," In *Proc of the ACM SIGOPS European Workshop*, ACM, 2004

About Authors :

Khairul Akram earned his Bachelor and Master degrees with First Class Honours in System Engineering with Computer Engineering from University of Warwick, United Kingdom in 2008 and 2009 respectively. He later joined Universiti Teknologi PETRONAS (UTP) in 2010 to pursue his journey towards academic research and teaching courses to earn his PhD in Information System. During his time in UTP, a number of journal articles and conference papers have been produced and published internationally. Currently, he is appointed as Researcher in Digital Forensic Department, CyberSecurity Malaysia and has been entrusted with the research on embedded system forensics. His passion in research is towards algorithms, embedded system, image processing and audio authentication. He is a member of IET professional group.

Ahmad Kamil Mahmood earned his Bachelor and Master degree in Actuarial Science and Statistics from the University of Iowa, Iowa City, USA in 1986 and 1988 respectively. He later joined UUM, Bank Negara Malaysia, and Public Service Department and PETRONAS. After 10 years in the industry, he continued his journey in the academia serving the Universiti Teknologi PETRONAS in 1998 teaching courses for the Bachelor Degree in ICT and BIS. He earned his PhD in Information Systems from the University of Salford, UK in 2005. With his research team, a number of journal articles and conference papers have been produced and published internationally. Currently, his industrial collaboration research projects keep him occupied while supervising 7 postgraduate students and assuming the Dean of Faculty of Science and IT.

Jafrezal Jaafar obtained B.Sc in Computer Science from Universiti Teknologi Malaysia in 1998, MAppSc. (IT) from RMIT University (Australia) in 2002 and PhD from University of Edinburgh (Scotland, UK) in 2009. Previously he works as System Engineer for several years. He is currently the Head of Department for the Computer &Information Sciences Department, Universiti Teknologi PETRONAS, Malaysia. His research interests are in the area of Soft Computing and HCI. He is actively involved in a number of research works and secured research grants in these areas. He has also produced numerous journal, conference and workshop papers.

Solahuddin received his PhD from University of Bradford, United Kingdom in Network Security in 2008. He received a post-graduate Diploma in Systems Analysis from UiTM in 1991. He started his career with the Malaysian Armed Forces after completing his first degree in Electrical Engineering from Wichita State University, USA in 1986. He served in the Royal Signal Regiment of the Malaysian Army for 10 years holding various posts as communications engineer and IT manager before joining the industry after the completion of his stint with the Malaysian Armed Forces.

In 1997 he joined Softlabs Technologies Sdn Bhd as the General Manager. He was entrusted to manage and lead system development projects with various industries such as oil and gas, defence, telecommunications and local governments. In 2002 he joined National ICT Security & Emergency Response Centre (NISER) now known as CyberSecurity Malaysia as the Expert Service Manager. Later on, he was entrusted to be the manager for Malaysia Emergency Response Team (MyCERT). He has earned 4 professional certifications namely CWNA, CISSP, CEH and BS7799Lead Auditor.

With his knowledge and skills in various security domains, he is now entrusted to be the Chief Technology Officer at CyberSecurity Malaysia. He is also the research coordinator for CyberSecurity Malaysia.