Journal Pre-proofs

Technical note

Plain text passwords:- a forensic RAM-raid

Graeme Horsman

PII: S1355-0306(20)30013-7
DOI: https://doi.org/10.1016/j.scijus.2020.07.001
Reference: SCIJUS 889

To appear in: *Science & Justice*

Received Date: 22 January 2020
Revised Date: 26 June 2020
Accepted Date: 4 July 2020

Please cite this article as: G. Horsman, Plain text passwords:- a forensic RAM-raid, *Science & Justice* (2020), doi: https://doi.org/10.1016/j.scijus.2020.07.001

Plain text passwords:- a forensic RAM-raid

*Graeme Horsman

School of Health & Life Sciences,
Teesside University,
Middlesbrough,
North Yorkshire,
United Kingdom

Email: g.horsman@tees.ac.uk
Phone: +44 1642 738130

There are no conflicts of interest to declare.

Plain text passwords:- a forensic RAM-raid

Abstract
Despite many academic studies in the last 15 years acknowledging the investigative value of physical memory due to the potential sensitive nature of data it may contain, it arguably remains rarely collected at-scene in most criminal investigations. Whilst this may be due to factors such as first responders lacking the technical skills to do this task, or simply that it is overlooked as an evidence source, this work seeks to emphasise the worth of this task by demonstrating the ability to recover plain-text login credentials from it. Through an examination of logins made to 15 popular online services carried out via the Chrome, Edge and Mozilla Firefox browsers, testing shows that plain-text credentials are present in RAM in every case. Here, a transparent test methodology is defined and the results of test cases are presented along with 'string markers' which allow a practitioner to search their RAM captures for the presence of unknown credential information for these services in future cases.

Keywords: RAM; Physical Memory; Passwords; Digital Forensics; Investigation

1 Introduction

Random Access Memory (RAM), also frequently referred to as physical memory (both terms will be used interchangeably from here-on) is a core component of all computer systems. It facilitates the short-term storage of data allowing for the efficient retrieval of content which is frequently accessed by the device, where as Okolica and Peterson (2011, p.118) indicate that 'a computer's memory provides the most up to date snapshot of a machine's state'. The volatile nature of RAM means that its contents are lost when power is removed from the system (ENISA, 2015; Voelzow, 2017) placing emphasis on the need for first responders to collect this content at the first appropriate opportunity. Whilst there is the potential for content to be paged (on Microsoft operating systems or equivalent processes on other platforms) to the Pagefile.sys (which enables 'the system to remove infrequently accessed modified pages from physical memory to let the system use physical memory more efficiently for more frequently accessed pages' (Microsoft,

2019)), this file is not guaranteed to contain all data which was once present in RAM. In addition, the hiberfil.sys may also contain RAM data following a system hibernation, but again is not guaranteed to do so if this feature is disabled or hibernation is not activated.

RAM content will contain trace evidence of a user's short-term behaviours in regards to their actions on a given computer system (Stevens and Casey, 2010; Thomas *et al*., 2013; Hausknecht et al., 2015; Leimich et al., 2016), information which may not be present on less volatile system data storage forms such as the system disk drive. In criminal investigation contexts, the collection of RAM must be done at-scene when the device is still active (and if the device is even active in the first instance). In most cases, this involves the introduction and execution of specialist software on the suspect system itself, causing RAM content to be written to a form of sanitised external storage media. This 'RAM dump' can then be analysed via tools such as Volatility (found at https://www.volatilityfoundation.org/) which can parse known structures, via basic GREP keyword searches (an often utilised approach to RAM data extraction due to the simplicity of the method), through known signatures or in some cases specific memory locations can be queried (Malin et al., 2012). In the context of a forensic examination of a device, RAM has been shown to contain plain-text password strings, running process information, encryption keys, malware traces and Internet history remnants (It has been long since accepted that RAM can and often does contain sensitive information related to device usage (Hejazi et al., 2009; Maartmann-Moe et al., 2009; Wang et al., 2009; Casey and Aquilina, 2010; Balogun and Zhu, 2013; Barhate and Jaidhar, 2013; Rafique and Khan, M.N.A., 2013; Richard III and Case, 2014; Thongjul and Tritilanunt, 2015; Lillis et al., 2016). As a result, it can offer important information for supporting an investigation of suspect activity on a device. Due to both the potentially volatility and sensitivity of its content, it is important that practitioners and first responders now consider RAM as a core element of their data collection processes at-scene, making sure that it is acquired via a suitable forensic process and thoroughly examined (Schuster, 2008; Walters and Petroni, 2007; Caviglione et al., 2017; Schatz and Cohen, 2017; Schramp, 2017).

This work provides the results of RAM testing and analysis for the purpose of determining the presence of plain-text login credential information following logins made via an Internet browser to online services. Logins were made to 15 online services using Chrome, Edge and Mozilla Firefox. Subsequent RAM acquisition and analysis reveals that plain-text password information is stored in RAM following logins by all browsers for all services. However, only 10 services consistently store credential information in a format where GREP syntax can be identified to extract credential information from RAM for these services for unknown credential information across all three browsers. A transparent methodology for testing is described which is hoped will foster peer-review and facilitate further work in the area of RAM analysis and credential identification and extraction. Test results are offered and discussed with suggestions for key future works made.

2 Background

Despite Hoelz et al., in 2011 suggesting RAM capture at-scene was slowly increasing in prevalence, it is arguably still not undertaken at all available opportunities at-scene. This is despite both the Association of Chief Police Officers (ACPO, 2012) and Interpol (2019) indicating that this

should be one of the first processes considered by a first responder where collection is feasible, as opposed to older positions of 'pulling the plug' on a system for subsequent dead-box analysis (Sutherland, 2008). It should also be noted that RAM capture is not always possible (a device may be off for example) or feasible if there is not the equipment to carry out the task or that a first responder is not trained to do so.

Within the context of this work, focus will be maintained on the recovery of login credential information relating to services which a user has accessed via an Internet browser. Existing studies have shown that it is possible to retrieve password information from RAM (see Davidoff, 2008; Zhao and Cao, 2009; Simon and Slay, 2010; Xu and Wang, 2013), with Simon and Slay (2010) highlighting the value of this information to a forensic investigation.

> "Recovery of passwords may be of value for user profiling as the target may use the same password for other applications. Depending on the particular communication technology being used, passwords may be exploited to obtain additional information. For example, some systems provide a web based account information page that may provide access to information such as voicemail messages or contact history that would not otherwise be available. (Simon and Slay, 2010)"

Where a suspect has accessed an online service, likely through a browser, access to this information often requires knowledge of these credentials. Let's consider the hypothetical scenario that a suspect has accessed a cloud storage provider to hide illegal material. A digital forensic investigator may determine usage of the service via Internet History records but this often does not reveal the full extent of the usage of this service, where access to it may be required. An investigator may pursue access in a number of ways, first, self-disclosure of the account details and content from the suspect, second, a lawful disclosure request made to a service provider within the bounds of applicable laws, or finally, a lawful login could be made (subject to acquiring the appropriate legal permission) by an investigating authority if credentials can be obtained. Arguably the final of these options provides the quickest option to pursuing investigatory leads, allowing practitioners to control the extraction of data from the service themselves without relying on any third party disclosure processes. Yet this option requires identifying what these credentials are, where RAM analysis may help.

Acquiring access to a service often requires identifying both a username (often overlooked in RAM analysis studies) and an associate password for the account. These may be obfuscated if stored locally on a system hard drive, however as indicated in previous studies, credentials can be 'leaked' into RAM and shown in plain text. As a result, where a suspect has logged into an online service and their machine remains active, RAM capture may reveal credential content.

3 Methodology

As part of testing for the presence of credentials in RAM (particularly when repeat testing is involved), the prevention of test contamination is important to ensure the validity of any results. This section documents the testing methodology utilised in this work, where Figure 1 provides an overview of the experimental setup, to allow for transparency and effective scrutiny of the work.

To start, 15 services were utilised in testing (shown in Table 1). Each service examined as part of this work first had an account created using a lab PC (separate to the test PC where RAM acquisition occurred), with a unique string set as the account password. This prevented any contamination of RAM by the account credentials prior to a login attempt for that service. Once the account was setup, the service account was then accessed via a test PC (running OS Microsoft Windows 10 Enterprise, Version 10.0.17763, Build 17763, with 8GB of RAM), where a login would be made (only 1). The login process was left to complete until the homepage of the service website was reached and completely loaded. At this point, a RAM capture was initiated using FTK Imager (V. 4.2.0.13). This RAM capture was then searched using FTK Imager for the presence of both ANSI and Unicode plain-text login credentials that to carry out the login, with each instance manually examined and recorded. Finally, testing has focused on RAM captured content only where the Pagefile.sys was not covered within the experimentation.

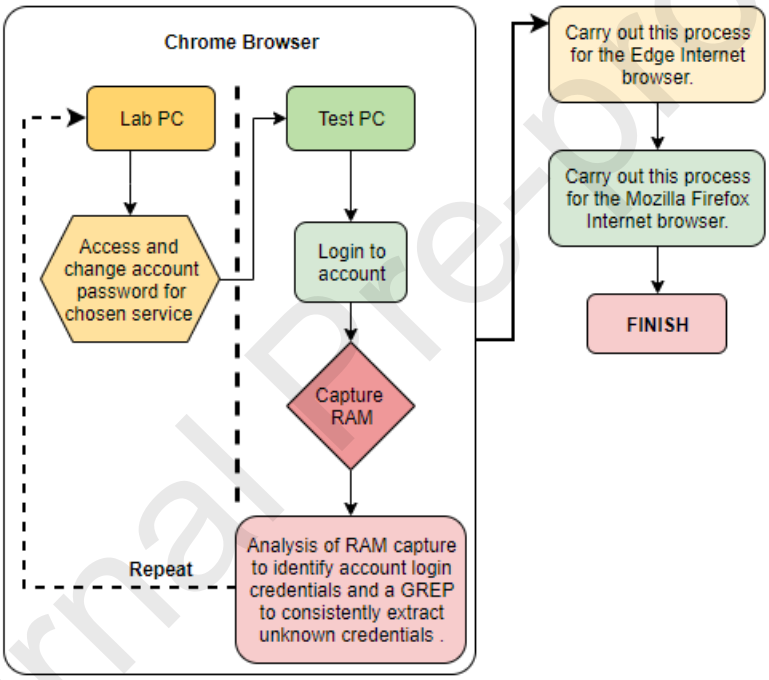


Figure 1: Experiment methodology.

This process was iterative where the Lab PC would be used to re-access the service, change the password credentials to another unique string allowing the Test PC to re-access the account with a subsequent login using updated credentials. To stress, the first time that each specific set of login credential strings were entered into the Test PC was the first time a login was made with this information. This process was repeated five times for each service. Each set of five RAM captures was examined to establish the presence of any consistent marker strings which could be used to identify the credentials.

**To note*:- Identifying that credentials are stored in RAM in the first instance is only valuable to the digital forensic practitioner if there is a way to extract these. Marker strings allow unknown credentials to be searched for in RAM captures (see Figure 2). In most investigations, a

practitioner will not know the password strings used by a suspect (as if they did, there would be no requirement for such password extraction processes). As a result in order to attempt to extract plain text passwords accurately and on a consistent basis, marker strings which either directly precede or follow a password string must be established. This allows practitioners to search any RAM captures, where the presence of a marker string may indicate also that a plain text credential is also retrievable. If a marker string is unique to a particular service, the presence of it in RAM may also reveal the service in which a password belongs to.

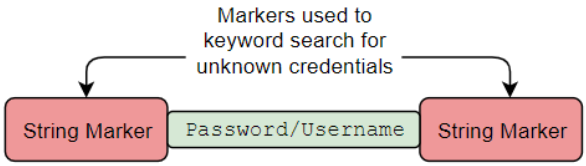


Figure 1: Marker strings for credential extraction.

To place the value of string markers in context, where a DF practitioner in a given case has both an image of a device and a RAM capture, it may be possible to both identify the services used by a suspect, and the credentials they have used to access it. Through an analysis of Internet history on the system, use of specific services by the suspect may be identifiable through URL analysis (for example Twitter login pages). By searching an associated RAM capture for marker strings, it may be possible to extract an associated plaintext password. However, in cases where the service does not maintain a unique marker string and multiple services have been used by a suspect (discussed in the results later), a practitioner may face trial and error in terms of which password correlates to which service.

The aforementioned process was repeated across the Chrome (v79.0.3945.117), Edge (v44.17763.1.0) and Firefox (v72.0.1) Internet browsers to determine whether traits of the login process differ in RAM between applications.

**To note*:- In all cases, the identification and extraction of both username and password credentials has been attempted.

4 Results

Table one provides the results of RAM credential extraction for 15 online services across three Internet browsers.

Table 1: Ram extraction results (Table submitted as a seperate .xlsx file).

It is important to note that in every test undertaken in all browsers tested, a plaintext password could be found. However, as noted in Table 1, only instances where a consistent string marker could be used to identify credential information were recorded. Where no consistent marker could be identified for a set of tests, 'NO Consistent GREP' is recorded.

**To Note*: String marker searches should be conducted in both ASCII and Unicode as both formats may appear in RAM.

For all 15 services, credentials entered via the Chrome Internet browser could be extracted during testing with the string markers noted in Table 1. Instagram, Reddit, Flickr and Dropbox maintained no consistent string markers when a login was made in the Edge and Firefox browsers. Ask.FM maintained no consistent string marker when a login was made in the Edge browser.

It can be seen from the results that multiple services maintain the same string marker for their password credentials in RAM (for example, Yahoo! and MySpace). As a result, it is important that practitioners examine thoroughly the Internet history on a device in order to identify which online services are likely to have been accessed. Where multiple services have been accessed which deploy the same password marker strings, practitioners may be faced with having to try multiple passwords should they exist in RAM. Whilst time-wise this should not be too much of a burden, they should be aware that some accounts implement security measures for too many incorrect password inputs. One way to help here is to consider whether a matching username string marker exists, as this will help to distinguish what service a password belongs to, as with Yahoo! and MySpace, they both utilize different username string markers.

5 Conclusions and talking points

This work has demonstrated that not only are online service credentials present in RAM in plaintext, but that they are also stored in a way which allows their extraction by a practitioner using the GREP syntax recorded in Table 1. However, following testing there are three main areas requiring further discussion.

1. *Persistence*: The persistence of data in RAM must be considered following general usage of a system over time (Burdach, 2006; Solomon et al., 2007). The methodology in the work aims to establish the recoverability of credentials from RAM and does not explore persistence as a testing variable. Given that it has been shown that credentials are present, an examination of the persistence of this content following prolonged system usage is the next logical step to work in this area. Such work would provide an insight into the window of opportunity a practitioner may have in order for extracted RAM to contain credentials from historic logins. In addition, it will also allow paging to be considered and examined, where it is expected that the string markers noted in Table 1 should also be of value in this context, but testing is required to confirm this.

2. *Remember me*: Most online services offer a ‘remember me’ function (an example is shown in Figure 3) which allows automatic logins to services or sustained open sessions. Where a ‘remember me’ option is engaged, the user is not required to type their password into the required field like the actions tested within this work.

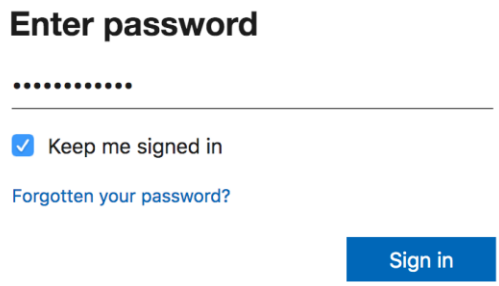


Figure 3: The 'remember me' function.

The impact of the '*remember me*' function must be considered as it is common for users to engage this when utilising their own personal devices. As a result, if a device is seized by a practitioner, it may be the case that any login to an online service has occurred in this way as opposed to a typed-in login. To provide an initial assessment, the following methodology was deployed within our test PC. A unique password string was set for a sub-set of chosen services (Facebook, Titter, Yahoo and Pinterest) and their '*remember me*' (or equivalent') option was initialised. The PC was then turned off to ensure a RAM contents purge, where the device was then powered back on and a login to each service was carried out (no password typing) followed by RAM acquisition. In all cases, no credential information was present in the RAM dump for any services.

3. *Vicinity Searching*: Vicinity searching is a concept coined here to describe a search which can get a practitioner within an 'area of a RAM dump' where credentials are stored, but not consistently close to it. This scenario has been noted by Davidoff (2008) with regards to clear text password recovery from Linux memory and the difficulties with identifying consistent GREP string search criteria. A visual representation of vicinity searching is presented in Figure 4, where perhaps the best example to explain this concept is a comparison between the highlighted string markers for extracting Facebook credentials and the lack of a consistent string marker for Dropbox credentials for the Firefox and Edge browsers.

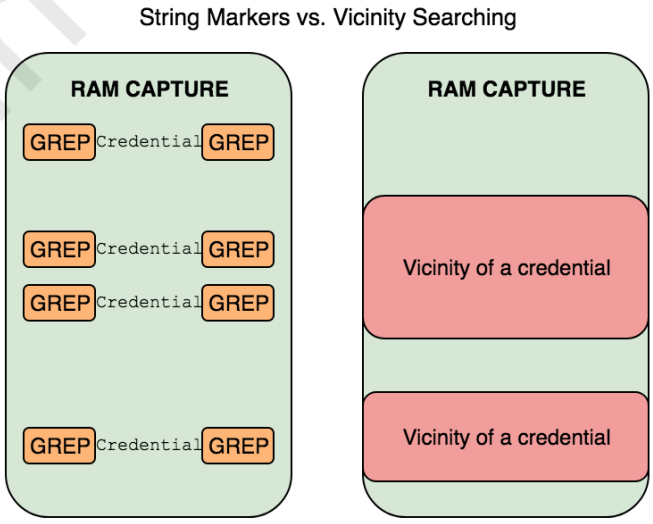


Figure 4: The concept of a vicinity search.

In relation to Facebook, the marker <`&pass=`> is directly before a user's plaintext password string. However, following testing, no such markers could be established for a Dropbox password in RAM when a login is made via Edge and Mozilla. Yet, plaintext passwords are present, and following an analysis of their position in RAM, testing indicated that the password appears consistently in the vicinity of the string marker <`l.o.g.i.n._.p.a.s.s.w.o.r.d`> (`0x 6C 00 6F 00 67 00 69 00 6E 00 5F 00 70 00 61 00 73 00 73 00 77 00 6F 00 72 00 64`). This is shown in Figures 5 and 6 where a Dropbox password string is within 21 bytes (Figure 5) and 405 bytes (Figure 6).

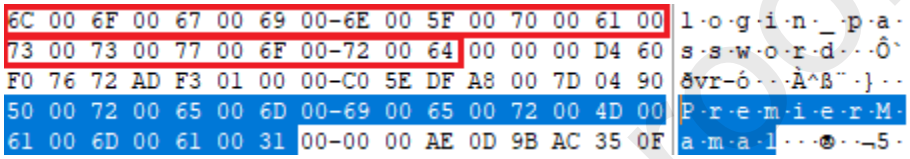


Figure 5: Dropbox vinivity search password (Part 1).

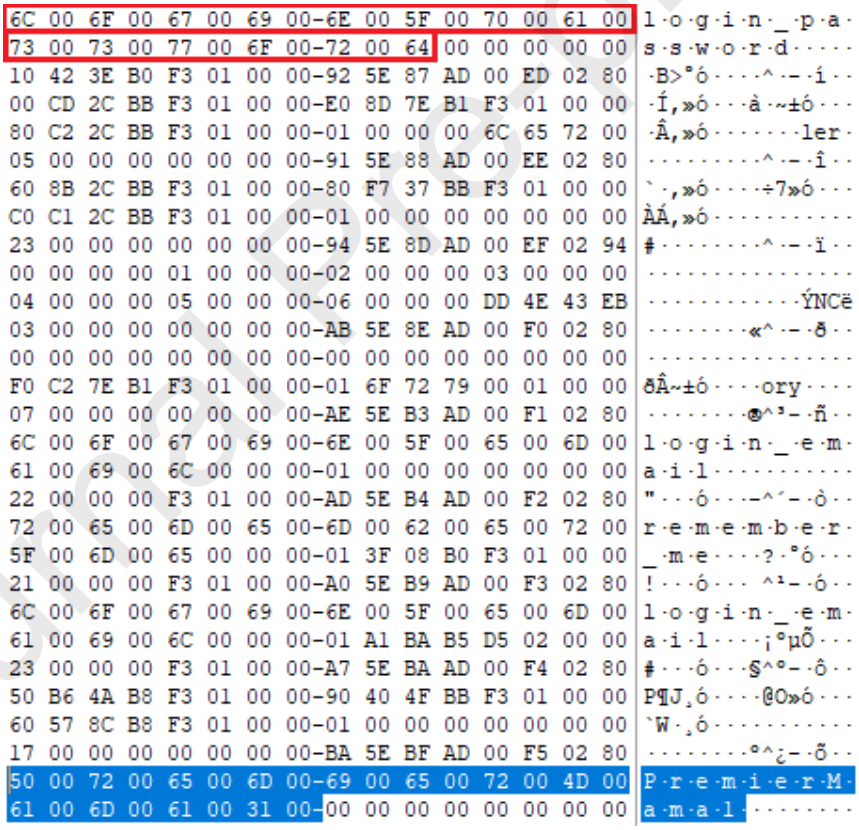


Figure 6: Dropbox vicinity search password (Part 2).

The idea behind a vicinity search is that it does not provide the practitioner with a defined set distance (in terms of number of bytes) away from a set of credentials, but can place them near to one, at which point the practitioner is required to manually review surrounding data to determine if a potential password may be present. The value of this concept is that in cases where no string marker exists, given that plain text passwords were present in

RAM for all services and all browsers, this concept has the potential to be used to offer a chance of password recovery.

Vicinity searching requires further exploration to determine its worth in regards to identifying the extent to which it can be utilised for credential recovery and to determine its margin of error (how close a search can get to a credential). In addition, the development of a formalised vicinity search methodology is required and as a result, this concept is identified as an important element of future work.

References

ACPO, 2012. ACPO Good Practice Guide ACPO Good Practice Guide for Digital Evidence Available at: https://www.digital-detective.net/digital-forensics-documents/ACPO_Good_Practice_Guide_for_Digital_Evidence_v5.pdf (Accessed: 17 January 2020)

Balogun, A.M. and Zhu, S.Y., 2013. Privacy impacts of data encryption on the efficiency of digital forensics technology. *arXiv preprint arXiv:1312.3183*.

Burdach, Mariusz, 2006. 'Physical Memory Forensics' Available at: https://www.blackhat.com/presentations/bh-usa-06/BH-US-06-Burdach.pdf (Accessed: 17 January 2020)

Casey, E. and Aquilina, J.M., 2010. *Malware Forensics Field Guide for Windows Systems*. Elsevier Science & Technology.

Caviglione, L., Wendzel, S. and Mazurczyk, W., 2017. The future of digital forensics: Challenges and the road ahead. *IEEE Security & Privacy*, *15*(6), pp.12-17.

Davidoff, S., 2008. Cleartext passwords in linux memory. Massachusetts institute of technology, pp.1-13.

ENISA, 2015. Electronic evidence - a basic guide for First Responders' Available at: https://www.enisa.europa.eu/publications/electronic-evidence-a-basic-guide-for-first-responders (Accessed: 17 January 2020)

Hausknecht, K., Foit, D. and Burić, J., 2015, May. RAM data significance in digital forensics. In *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 1372-1375). IEEE.

Hejazi, S.M., Talhi, C. and Debbabi, M., 2009. Extraction of forensically sensitive information from windows physical memory. digital investigation, 6, pp.S121-S131.

Hoelz, B., Ralha, C. and Mesquita, F., 2011, January. Case-based reasoning in live forensics. In *IFIP International Conference on Digital Forensics* (pp. 77-88). Springer, Berlin, Heidelberg.

Interpol (2019) 'GLOBAL GUIDELINES FOR DIGITAL FORENSICS LABORATORIES' Available at: https://www.interpol.int/content/download/13501/file/INTERPOL_DFL_GlobalGuidelinesDigitalForensicsLaboratory.pdf (Accessed: 17 January 2020)

Leimich, P., Harrison, J. and Buchanan, W.J., 2016. A RAM triage methodology for Hadoop HDFS forensics. *Digital Investigation*, *18*, pp.96-109.

Lillis, D., Becker, B., O'Sullivan, T. and Scanlon, M., 2016. Current challenges and future research areas for digital forensic investigation. *arXiv preprint arXiv:1604.03850*.

Maartmann-Moe, C., Thorkildsen, S.E. and Årnes, A., 2009. The persistence of memory: Forensic identification and extraction of cryptographic keys. *digital investigation*, *6*, pp.S132-S140.

Malin, C.H., Casey, E., Aquilina, J.M. and Rose, C.W., 2012. Malware Forensics Field Guide For Windows Systems. 225 Wyman Street, Waltham, MA 02451.

Microsoft, 2019. 'Introduction to page files' Available at: https://docs.microsoft.com/en-us/windows/client-management/introduction-page-file (Accessed: 17 January 2020)

Okolica, J. and Peterson, G.L., 2011. Extracting the windows clipboard from physical memory. *Digital investigation*, *8*, pp.S118-S124.

Rafique, M. and Khan, M.N.A., 2013. Exploring static and live digital forensics: Methods, practices and tools. *International Journal of Scientific & Engineering Research*, *4*(10), pp.1048-1056.

Richard III, G.G. and Case, A., 2014. In lieu of swap: Analyzing compressed RAM in Mac OS X and Linux. *Digital Investigation*, *11*, pp.S3-S12.

Schatz, B. and Cohen, M., 2017. Advances in volatile memory forensics. *Digital Investigation*, *100*(20), p.1.

Schramp, R., 2017. Live transportation and RAM acquisition proficiency test. *Digital Investigation*, *20*, pp.44-53.

Schuster, A., 2008. The impact of Microsoft Windows pool allocation strategies on memory forensics. *Digital Investigation*, *5*, pp.S58-S64.

Simon, M. and Slay, J., 2010, February. Recovery of skype application activity data from physical memory. In 2010 International Conference on Availability, Reliability and Security (pp. 283-288). IEEE.

Solomon, J., Huebner, E., Bem, D. and Szeżynska, M., 2007. User data persistence in physical memory. *Digital Investigation*, *4*(2), pp.68-72.

Stevens, R.M. and Casey, E., 2010. Extracting Windows command line details from physical memory. *Digital investigation*, *7*, pp.S57-S63.

Sutherland, I., Evans, J., Tryfonas, T. and Blyth, A., 2008. Acquiring volatile operating system data tools and techniques. *ACM SIGOPS Operating Systems Review*, *42*(3), pp.65-73.

Thomas, S., Sherly, K.K. and Dija, S., 2013, April. Extraction of memory forensic artifacts from windows 7 ram image. In *2013 IEEE Conference on Information & Communication Technologies* (pp. 937-942). IEEE.

Walters, A. and Petroni, N.L., 2007. Volatools: Integrating volatile memory into the digital investigation process. *Black Hat DC*, *2007*, pp.1-18.

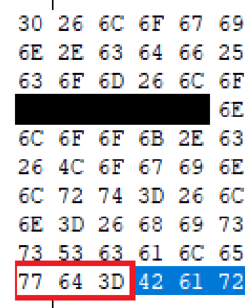
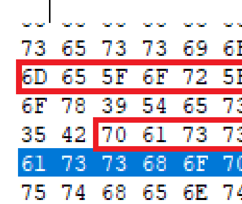
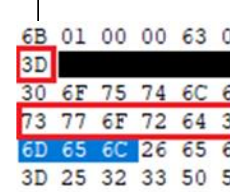
Thongjul, S. and Tritilanunt, S., 2015, July. Analyzing and searching process of internet username and password stored in Random Access Memory (RAM). In *2015 12th International Joint Conference on Computer Science and Software Engineering (JCSSE)* (pp. 257-262). IEEE.

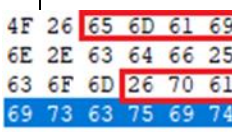
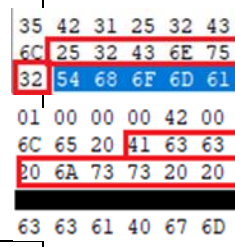
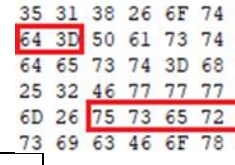
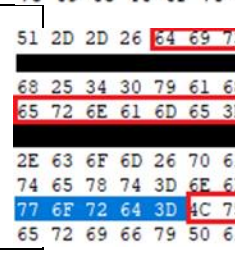
Voelzow, Victor 2017 'Blog' Available at: https://www.coe.int/en/web/octopus/blog/-/blogs/live-data-forensics-or-why-volatile-data-can-be-crucial-for-your-cases/ (Accessed: 17 January 2020)

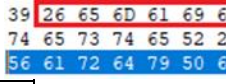
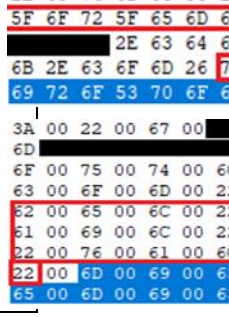
Wang, L., Zhang, R. and Zhang, S., 2009, December. A model of computer live forensics based on physical memory analysis. In *2009 First International Conference on Information Science and Engineering* (pp. 4647-4649). IEEE.

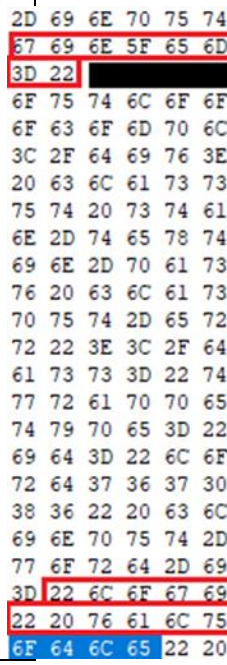
Xu, L. and Wang, L., 2013, December. Research on extracting system logged-in password forensically from windows memory image file. In 2013 Ninth International Conference on Computational Intelligence and Security (pp. 716-720). IEEE.

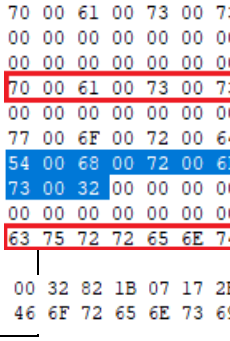
Zhao, Q. and Cao, T., 2009. Collecting Sensitive Information from Windows Physical Memory. *JCP*, *4*(1), pp.3-10.

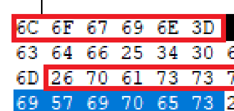
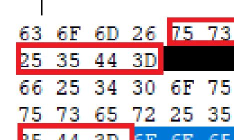
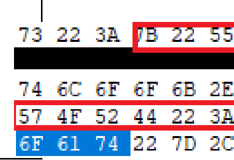
	Chrome				Edge				Mozilla Firefox				
Service	Password Search Term (ASCII)	Password Search Term (HEX)	Username Search Term (ASCII)	Username Search Term (HEX)	Password Search Term (ASCII)	Password Search Term (HEX)	Username Search Term (ASCII)	Username Search Term (HEX)	Password Search Term (ASCII)	Password Search Term (HEX)	Username Search Term (ASCII)	Username Search Term (HEX)	Example
Outlook	<=&passwd=>	0x3D 26 70 61 73 73 77 64 3D	<login=>	0x6C 6F 67 69 6E 3D	<=&passwd=>	0x3D 26 70 61 73 73 77 64 3D	<login=>	0x6C 6F 67 69 6E 3D	<=&passwd=>	0x3D 26 70 61 73 73 77 64 3D	<login=>	0x6C 6F 67 69 6E 3D	
Twitter	<password%5D=>	0x70 61 73 73 77 6F 72 64 25 35 44 3D	<username_or_email%5D=>	0x75 73 65 72 6E 61 6D 65 5F 6F 72 5F 65 6D 61 69 6C 25 35 44 3D	<password%5D=>	0x70 61 73 73 77 6F 72 64 25 35 44 3D	<username_or_email%5D=>	0x75 73 65 72 6E 61 6D 65 5F 6F 72 5F 65 6D 61 69 6C 25 35 44 3D	<password%5D=>	0x70 61 73 73 77 6F 72 64 25 35 44 3D	<username_or_email%5D=>	0x75 73 65 72 6E 61 6D 65 5F 6F 72 5F 65 6D 61 69 6C 25 35 44 3D	
Instagram	<&password=>	0x26 70 61 73 73 77	<username=>	0x75 73 65 72 6E 61	**NO Consistent GREP**	**NO Consistent**	**NO Consistent GREP**	**NO Consistent**	**NO Consistent GREP**	**NO Consistent**	**NO Consistent GREP**	**NO Consistent**	

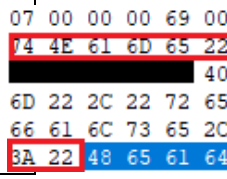
		6F 72 64 3D		6D 65 3D		**GREP**		**GREP**		**GREP**		**GREP**	
Facebook	<&pass=>	0x 26 70 61 73 73 3D	<&email=>	0x 26 65 6D 61 69 6C 3D	<&pass=>	0x 26 70 61 73 73 3D	<&email=>	0x 26 65 6D 61 69 6C 3D	<&pass=>	0x 26 70 61 73 73 3D	<&email=>	0x 26 65 6D 61 69 6C 3D	
Google	<%2Cnull%2C%5B%22>	0x 54 68 6F 6D 61 73 46 61 69 72 79 31	<Account: test jss (>	0x 41 63 63 6F 75 6E 74 3A 20 74 65 73 74 20 6A 73 73 20 20 0A 28	<%2Cnull%2C%5B%22>	0x 54 68 6F 6D 61 73 46 61 69 72 79 31	<Account: test jss	0x 41 63 63 6F 75 6E 74 3A 20 74 65 73 74 20 6A 73 73 20 20 0A 28	<%2C null %2C% 5B%2 2>	0x 54 68 6F 6D 61 73 46 61 69 72 79 31	<Account: test jss	0x 41 63 63 6F 75 6E 74 3A 20 74 65 73 74 20 6A 73 73 20 20 0A 28	
Redit	<=&password=>	0x 3D 26 70 61 73 73 77 6F 72 64 3D	<&username=>	0x 26 75 73 65 72 6E 61 6D 65 3D	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	
Yahoo	<&password=>	0x 26 70 61 73 73 77 6F	<displayName=> or <&user	0x 64 69 73 70 6C 61 79	<&password=>	0x 26 70 61 73 73 77 6F	<displayName=> or <&user	0x 64 69 73 70 6C 61 79	<&password=>	0x 26 70 61 73 73 77 6F	<displayName=> or <&user	0x 64 69 73 70 6C 61 79	

		72 64 3D	name= >	4E 61 6D 65 3D or 0x 26 75 73 65 72 6E 61 6D 65 3D		72 64 3D	name= >	4E 61 6D 65 3D or 0x 26 75 73 65 72 6E 61 6D 65 3D		72 64 3D	name= >	4E 61 6D 65 3D or 0x 26 75 73 65 72 6E 61 6D 65 3D	
Myspace	< &pas swor d=>	0x 26 70 61 73 73 77 6F 72 64 3D	<&emai l=>	0x 26 65 6D 61 69 6C 3D	< &pass word= >	0x 26 70 61 73 73 77 6F 72 64 3D	<&emai l=>	0x 26 65 6D 61 69 6C 3D	< &pas swor d=>	0x 26 70 61 73 73 77 6F 72 64 3D	<&emai l=>	0x 26 65 6D 61 69 6C 3D	
Pinterest	< &pas swor d=>	0x 26 70 61 73 73 77 6F 72 64 3D	<usern ame_or _email =>	0x 75 73 65 72 6E 61 6D 65 5F 6F 72 5F 65 6D 61 69 6C 3D	<,.".l .a.b.e .l.".: ." E.m.a. i.l.". }.].}. ,.". v.a.l. u.e.". :.>	0x 2C 00 22 00 6C 00 61 00 62 00 65 00 6C 00 22 00 3A 00 22 00 45 00 6D 00 61 00 69 00	SAME	SA M E	< &pas swor d=>	0x 26 70 61 73 73 77 6F 72 64 3D	<usern ame_or _email =>	0x 75 73 65 72 6E 61 6D 65 5F 6F 72 5F 65 6D 61 69 6C 3D	

						6C 00 22 00 7D 00 5D 00 7D 00 2C 00 22 00 76 00 61 00 6C 00 75 00 65 00 22 00 3A 00 22						
Dropbox	<"login_password" value=">	0x 22 6C 6F 67 69 6E 5F 70 61 73 73 77 6F 72 64 22 20 76 61 6C 75 65 3D 22	<login_email " value=" ">	0x 6C 6F 67 69 6E 5F 65 6D 61 69 6C 22 20 76 61 6C 75 65 3D 22	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	

Ask.FM	<current-password> **or** <p.a.s.s.w.o.r.d>	0x6375727265**or**0x70006100730073	<login>	0x6C6F6769 6E	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	<&password=>	0x26706173737765	<login=>	0x6C6F6769 6E3D	

		0077006F00720064											
Box	< &password=>	0x2670617373776F72643D	<login=>	0x6C6F67696E3D	< &password=>	0x2670617373776F72643D	<login=>	0x6C6F67696E3D	< &password=>	0x2670617373776F72643D	<login=>	0x6C6F67696E3D	
Tumblr	<password%5D=>	0x70617373776F72642535443D	<user%5Bemail%5D=>	0x75736572253542656D61696C2535443D	<password%5D=>	0x70617373776F72642535443D	<user%5Bemail%5D=>	0x75736572253542656D61696C2535443D	<password%5D=>	0x70617373776F72642535443D	<user%5Bemail%5D=>	0x75736572253542656D61696C2535443D	
Flickr	<"PASSWORD":">	0x2250415353574F524422	<"USERNAME":">	0x22555345524E414D4522	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	**NO Consistent GREP**	

		3A 22		3A 22									
				0x 7B 22 61				0x 7B 22 61				0x 7B 22 61	
		0x 22 70 61 73 73 77 6F 72 64		63 63 6F 75 6E 74 4E 61 6D 65		0x 22 70 61 73 73 77 6F 72 64		63 63 6F 75 6E 74 4E 61 6D 65		0x 22 70 61 73 73 77 6F 72 64		63 63 6F 75 6E 74 4E 61 6D 65	
iCloud	<"password":">	22 3A 22	<{"accountName":">	22 3A 22	<"password":">	22 3A 22	<{"accountName":">	22 3A 22	<"password":">	22 3A 22	<{"accountName":">	22 3A 22	

- A RAM examination if offered for service access via Edge, Chrome and Firefox.
- 16 services examined for credential recovery in RAM.
- String markers are identified for credential recovery.