Forensic Analysis of Digsby Log Data to Trace Suspected User Activities

Muhammad Yasin, Muhammad Abulaish, Muhammad Nour Naeem Elmogy

Center of Excellence in Information Assurance
King Saud University,
Riyadh, Saudi Arabia
{mohyasin.c, mabulaish, melmougy}@ksu.edu.sa

Abstract. Digsby is a versatile nature application, which can be used for instant messaging, e-mailing and at the same time providing an opportunity to end users to communicate over online social networks. Due to providing versatile services and facilities, the usage of Digsby by end users has diverted the attention of intelligence agencies to look into its log data from digital forensics examination point of view. In this paper, we follow a current conventional approach to analyze Digsby log data, which considers Windows registry and directory files as a vital source of evidence related to cyber crimes, specifically in case of Digsby usage. We have also analyzed the password encryption method used by the Digsby developers and discussed the expected insider attack. In addition, we have developed an application to assist analysts in decrypting Digsby user password.

1 Introduction

Due to increasing popularity of social media, Digsby has emerged as a multi-purpose application that can be used for Instant Messaging (IM), emailing and social networking. Due to its multi-dimensional nature, the analysis of log data maintained by Digsby may have great significance from digital forensics perspective. Digsby supports multiple instant messaging protocols and its current version supports AOL Instant Messenger (AIM), MSN, Google Talk, Jabber, Yahoo and Facebook. It allows IM users to use aliases and merge multiple accounts to avoid duplicate buddies. All these features need special attention, especially, merging of multiple different IM accounts of same user demands a close investigation to confirm which account was used for chatting and which one is in current use. Digsby email manager manages Gmail, Yahoo mail, Hotmail, AOL/AIM mail, IMAP and POP accounts. It is vital to know, does it maintain the logs of sent and received emails. The concerning matter is that where and how it keeps track of every sent or received email. Digsby's social networking feature updates the users regarding activities on social and professional networks such as Facebook, Twitter, MySpace and LinkedIn.

Although, a number of researches have been carried out to collect forensics artefacts of Instant Messengers [1] such as Yahoo [2], MSN [3]/ Windows Live

Messenger [4], AOL/AIM [5], and Pidgin [6], this area is still in its infancy and there is a need to forensically analyze such tools to identify digital traces. In this paper, we present an analysis of Digsby log data to collect digital footprints related to suspected users activities. Different tools and techniques used for finding digital traces are also elaborated. Data analysis for forensic traces collection is performed on home and professional editions of Windows XP with service pack 2 and Windows 7. Administrator and limited user accounts have been created to analyze its behavior in different situations. Test scenarios in virtual machine `VMware` are created and both static data as well as live system analysis are performed. `Registrar Lite`[1] and Windows provided Registry Editor *"regedit.exe"* are used to conduct registry analysis. `WinHex` by X-Ways Forensics, a hexadecimal,disk and RAM editor, is used for critical examination of various curtail files such as *"logininfo.yaml"* and *"iconhashes.dat"* files. Finally uninstallation results are captured and examined.

The rest of the paper is structured as follows. Starting with the introduction of Digsby installation process in section 2, we discuss the login and authentication processes by Digsby in section 3. Section 4 and section 5 focus on the analysis of windows registry and chat logs, respectively. Section 6 throws light on files and folders containing artefacts from Digsby, whereas section 7 describes the file containing login credentials, and password decryption technique. The developed application to decrypt password and expected insider attack is also described in this section. Finally, section 8 concludes the paper with future directions of work.

2 Digsby Installation Process

Digsby installer provides a lot of flexibility for the end users to customize its installation. For personal or home user, it provides *"All Users"* option where user has full access rights to install and uninstall any program. But in corporate or shared computer scenario, user prefers to install under his/her Windows account (`Windows_ID`) by using *"Single User"* choice. If an organization has implemented a software installation policy, where a user does not have administrative rights to install and uninstall any program on his allocated computer, he/she can install Digsby on portable device with *"Portable"* preference.

Digsby default installation path for *"All User"* is `"C:\Program Files\Digsby"` and for *"Single User"* is under local settings `"C:\Documents and Settings\<Windows_ID>\Local Settings \Application Data\Digsby\App"`. Similarly, for *"Portable"* devices it uses `"<Directory Path>\Digsby\App"` as the default path. These paths are not fixed, rather they are flexible and can be changed by the user manually. In order to find the traces of Digsby usage and installation paths (if installed), an examiner can search for generic patterns *"Digsby\cache"*, *"Digsby\temp"*, *"logininfo.yaml"* and *"Digsby\App"* on suspected system or portable device. These files/paths show that Disgby was installed by the end user.

[1] http://www.resplendence.com/download/RegistrarLite.exe

3 Login and Authentication Process

Fig.1 illustrates the steps of Digsby login and authentication process. In order to use Digsby, firstly user has to sign up on it. During registration process, Digsby checks the availability of requested *"User ID"* and send confirmation message after successful completion of registration process. In the meanwhile, it generates the hash of user password and stores it in Digsby database . Digsby servers store the passwords neither encrypted nor in clear text. Only hash of the user password is stored on server to enhance the end user security. Once a user gets registered, he/she supplies login credentials for login to Digsby application. During this process, Digsby checks for application updates and tries to establish a connection with the server for authentication purpose. After successful authentication, server synchronizes with the requested client and updates his/her status to the online group members. Finally, user supplies login credentials for instant messengers, email accounts and social networks which he/she wants to use through Digsby ID. All supplied passwords are encrypted with Digsby password and then stored in Digsby database[2]. Thereafter, whenever user login to its Digsby account, his/her login credentials for all protocols are confirmed automatically.

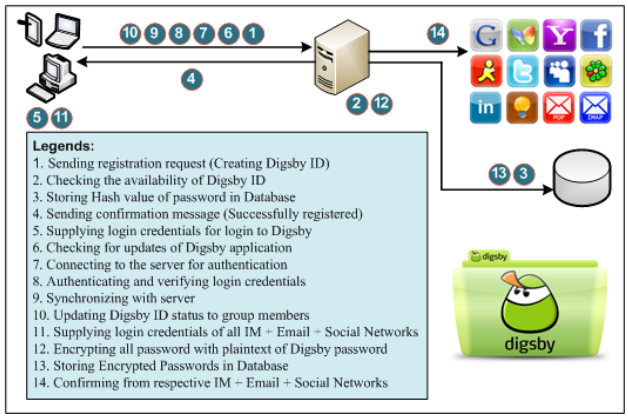

Fig. 1. Digsby login and authentication process

4 Windows Registry Examination

In contrast to other instant messengers[1][7] and download managers [8][9][10], Windows registry does not maintain the history of performed activities such as instant messages sent or received, email messages and activities performed on

[2] http://wiki.digsby.com/doku.php?id=security

social networks through Digsby. The only information maintained by Windows registry is the execution path, uninstall location, search bar and recent usage, i.e. whether the user has recently used Digsby or not as discussed in subsections.

During installation process, Digsby *"search bar"* checkbox is checked by default, which is generally opted by the Digsby users. The history of Digsby search toolbar is maintained under `"HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\SearchScopes\{0633EE93-D776-47 2f-A0FF-E1416B8B2E3A}"` key, which includes display name and URL address

Examiner can collect the information whether Digsby is recently used or not, if used when the activity was performed. The *"MUIChache"* key located at `"HKEY_USERS\S-1-5-21-1757981266-1708537768-725345543-500\Software\Microsoft\Windows\ShellNoRoam\MUICache"` path contains the footprints of Digsby recent usage which describes that user has performed some activity via Digsby. Registry Editor, the default registry viewer provided by Microsoft, does not provide the timestamp. Examiner can collect the timestamp with the help of `Registrar Lite` or `FTK Registry Viewer` by AccessData. A snapshot of the MUI Cache in Windows registry is shown in Fig. 2, which gives clear indication to examiner that Digsby application has been used recently.

The *"Uninstall"* branch directs all the installed programs in the computer. The *"Digsby"* key contains registry key values which mainly include: execution path, uninstall path and publisher at `"HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\Digsby"` path.

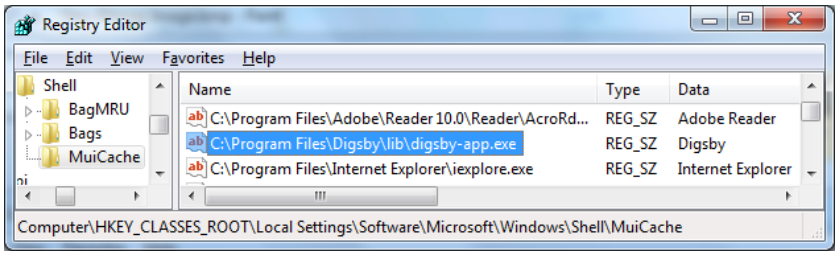

Fig. 2. A snapshot of the MUI Cache in Windows registry

5 Chat Logs

Forensics examiner can collect valuable footprints of chat when user is using *default* settings of Digsby. Digsby does not implement any hashing or encryption algorithm to hide user's chat logs, it is fairly easy for examiners to get chat logs. These chat logs are kept in HTML files under *"My Documents"* at *"\Digsby Logs\<Digsby ID>\<Protocol Name>\<Protocol ID>\<Contact ID>_<Protocol Name>"* path in both Windows XP and Windows 7. Digsby

maintains chat logs on daily basis of each contact person in a separate folder named using the *"contact ID"*. A pre-specified file naming convention, *"yyyy-mm-dd.html"*, is used to create files for maintaining chat logs data. This pattern can be exploited by chat examiner to search log data for a particular date on which the activity is performed.

Digsby maintains history of all supported protocols in separate folders. For example, in case of using Facebook chatting service, its chatting logs are maintained under *"fbchat"* folder. Inside this folder, Digsby creates a folder named *"Protocol ID"*, which is initially used by the user to get access. The complete path from the Digsby user ID can be "<Digsby ID>\fbchat\<Protocol ID>\571xxxxx3_fbchat". This folder path gives crucial information such as sender ID, receiver ID, protocol ID, and date and time stamp of on-going communication. It also provides numeric Facebook ID of contact user that can be used to trace the actual identity of the user by using search engines like Google. In this way, examiner can trace of all those users with whom the user was chatting. Digsby records all update information at "C:\Documents and Settings\<Windows_ID>\My Documents\Digsby Logs\<Digsby ID>\digsby\<Digsby ID>@digsby.org\digsby.org_digsby" path.

An uninstallation process is performed on both Windows XP and Windows 7. It is noted that Digsby does not provide *"program utility"* and *"complete"* options to its clients during uninstallation process. It has been found that Digsby does not remove any artefacts from Windows directories and as a result examiner can collect each and every artefacts even after the uninstallation of Digsby application. Examiner may not be able to get the artefacts from Windows registry uninstallation key entries after successful completion of Digsby uninstallation process. But, investigator can collect artefacts of recent usage from Windows registry when investigation started with in a short period of time after uninstallation.

6 Digsby Files/Folders Examination

Digsby maintains crucial information under various paths in Windows . Table 1 present the artefacts and their corresponding directory paths in Windows 7. The artefacts of Digsby installation and its executable files and can be collected from *"Digsby\App"* folder under multiple paths as shown in Table 1. The most crucial information which include the Digsby login ID's and respective passwords, proxy and configurations settings, cache and contact icons are found at "C:\Documents and Settings\<Windows_ID>\Local Settings\Application Data\Digsby" path in Windows XP and at "C:\Users\<Windows_ID>\AppData\Local\Digsby" path in Windows 7, respectively.

The *default* path to receive the files from the sender is receiver's desktop. Examiners can search for emoticons and message styles at "Application Data\Digsby" folder in Windows XP and "AppData\Roaming\Digsby" folder

in Windows 7. An emotion is the textual representation of a writer's mood or facial expression. By default, Digsby maintains chat history in user documents at "<Windows_ID>\<My Documents/Documents>\Digsby Logs" path in Windows XP and in Windows 7, respectively. Digsby does not allow its users to change the location of chat logs, which is quite helpful for investigators. Digsby also maintains "systemlog.txt" file at "C:\Users\<Windows_ID>\AppData\Local\Digsby \Logs" location, which can be used by the forensics examiners to collect the last login timestamp value.

Digsby contains *"digsbylocal.ini"* file at "C:\Documents and Settings\<Windows_ID>\Local Settings\Digsby" location, which holds proxy settings, chatting log directory location information and download/transfer files location information, as shown in Fig. 3(a). The proxy username and password are stored en-clair. The variable *"chatlogdir"* contains the complete path of chat history directory. In the same way *"save_to_dir"* maintains the complete path for received files.

Table 1. Artefacts and their directory paths in Windows 7

Artefacts	**Directory Path**
Default installation path, Executable	C:\Program Files \Digsby C:\Users\<Windows_ID>\AppData\Local\Digsby \App <Directory Path>\Digsby\App
Cache, Password, Temporary data, Icons, Last usage timestamp, Aliases	C:\Users\<Windows_ID>\AppData\Local\Digsby
File Transfer	C:\Users\<Windows_ID>\Desktop
Emoticons, Message Styles	C:\Users\<Windows_ID>\AppData\Roaming\Digsby
Chat and Update History	C:\Users\<Windows_ID>\Documents\Digsby Logs

Digsby contains *"iconhashes.dat"* file at "C:\Documents and Settings\<Windows_ID>\Local Settings\Digsby\cache\iconhashes.dat" path, which has information about all the contact ID's and icons of added contacts. These icons can be traced in *"Digsby\Cache"* folder. The format used to store information in *"iconhashes.dat"* file is illustrated in Fig. 3(b). It maintains information according to the protocols for example MSN and Facebook. The current version of Digsby stores SHA-1 Hash as a 20 byte value in hexadecimal format. But, its previous versions use a mixture of hexadecimal and ASCII values along with special characters to store SHA-1 Hash; for example "\x18\xc3\xec\x0f\xb2\x9d8\xa8 O|\xce#\n\xc3\xef\xeb\xcax\xc3s" is a SHA-1 Hash values in which *"\x"* represents that the next two characters are hexadecimal values.

The information contained in *"iconhashes.dat"* file is basically linked with icons that are stored in "C:\Documents and Settings\<Windows_ID>\Local Settings\Digsby\cache" folder. This folder contains icons of each contact ID in a separate subfolder on the name of protocol used for communication such as fbchat, gtalk, msn and yahoo, as shown in Table 2. These icons can be viewed using *"Windows Photo Viewer"*, a Windows application. Facebook icons are stored with *"580xxx436_ICON.dat"* name, in

(a) Proxy settings, chat log and received/downloaded files directory path

(b) Iconhashes.dat file format

Fig. 3. Digsbylocal.ini and iconhashes.dat files

which *"580xxx436"* represents the Facebook ID. Yahoo icons are kept with *"alxxxxxkh_ICON.dat"* name in which *"alxxxxxkh"* is the Yahoo ID of the contact person *"alxxxxxkh@yahoo.com"*. Google Talk icons are stored with complete address of the Gmail users for instance *"yasxxxxxxns@gmail.com_ICON.dat"*. Hotmail icon names such as *"faxxxxxxab@hotmail.com_ICON.dat"* are kept in the same format as Google Talk icons are stored. Table **??** shows the folders containing icons information, icons filename format, protocol ID. The protocol ID's can be found in *"iconhashes.dat"* files under their respective protocols. Another folder *"cache\webcache"* contains the Facebook ID's icon files, which has the same name as in *"iconhashes.dat"* file. Facebook user ID is found as a folder name under "C:\Documents and Settings\<Windows_ID>\Local Settings\Digsby\cache\<Digsby ID>_cache\fbchat\<protocol ID>" folder, which contains the *"cookiejar"* file that stores all the cookies of last logins with expiry timestamps.

Table 2. Icons artefacts w.r.t. different protocols

Protocol	Folder name	Icons name format	Protocol ID
Facebook	Fbchat	580xxx436_ICON.dat	580xxx436
Google	Gtalk	yasxxxxxxns@gmail.com.dat	yasxxxxxxns@gmail.com
Yahoo	Yahoo	alxxxxxkh_ICON.dat	alxxxxxkh
Msn	msn	faxxxxxxab@hotmail.com_ICON.dat	faxxxxxxab@hotmail.com

The information about alternative names can be found in *"alias_cache_v1.db"* file located at "C:\Users\<Windows_ID>\AppData\Local\Digsby\cache\<Digsby ID>_cache" path. This file can be open in SQL LITE format supported software to view the aliases of contact IDs of respective protocols. In order to protect user's privacy some letters in their names and aliases are replaced by the special character (?).

Digsby only maintains the artefacts of Instant messages. It does not maintain the records of sent and received emails and activities performed at social

networks. Although, examiner can collect the footprints of these activities from the web browser history as discussed in [11] and related information can also be found in Windows registry[3].

7 Analysis of Digsby Password

Digsby maintains login credentials detail in *"logininfo.yaml"* file located at `"C:\Documents and Settings\<Windows_ID>\Local Settings\Application Data\Digsby"` and `"C:\Users\<Windows_ID>\AppData\Lo- cal\Digsby"` paths in Windows XP and Windows 7, respectively. Fig. 4 presents a snapshot of the content of *"logininfo.yaml"* file, which contains creation time, last modification time, and directory path. Examiner can predict the installation date and user's first time use of Digsby application using the creation timestamp value. The last modification timestamp represent last time the password was changed or the time when a second user has used Digsby on same PC. Digsby does not permit users to change the location of this file, and consequently when a user is using Digsby, this file will be located at the aforesaid directories.

The *"logininfo.yaml"* file contains the Digsby ID's of all users who use it, but passwords are stored in an encrypted form only if a user selects *"Save password"* checkbox. The current version of Digsby supports thirty one characters long password string, but in previous versions it supports only sixteen characters long password string. The *"Exposing the Password Secrets of Digsby"*[4] article describes the password cracking method. Digsby uses RC4 encryption algorithm to encrypt user password and generate Base-64 code of the encrypted password before storing it into the designated file.

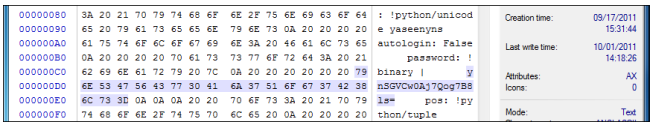

Fig. 4. Login Credentials

Encryption and decryption key of RC4 algorithm is a 20 byte SHA-1 hash of *"system product ID"*, *"install date"*, and *"Digsby ID"*. We have developed an application, which performs the following steps to decrypt an encrypted password. A snapshot of our application is shown in Fig. 5.

1. Search for *"logininfo.yaml"* file in directory paths.
2. Decode the Base-64 encoded password found in the file.

[3] http://accessdata.com/media/en_us/print/papers/wp.Registry_Quick_Find_Chart.en_us.pdf
[4] http://securityxploded.com/digsbypasswordsecrets.php

3. Collect system product ID, install date and Digsby ID. Digsby ID can be collected from the *"logininfor.yaml"* file. System Product ID and Install date are found in Windows registry under `"HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion"` key with their corresponding names *"DigitalProductId"* and *"InstallDate"*.
4. Make a complete string of system product ID, install date and Digsby ID in a sequence as written. The generated SHA-1 hash of the string is the decryption key of encrypted password.
5. Decrypted the encrypted password using RC4 encryption algorithm with the 20 byte generated SHA-1 hash key.

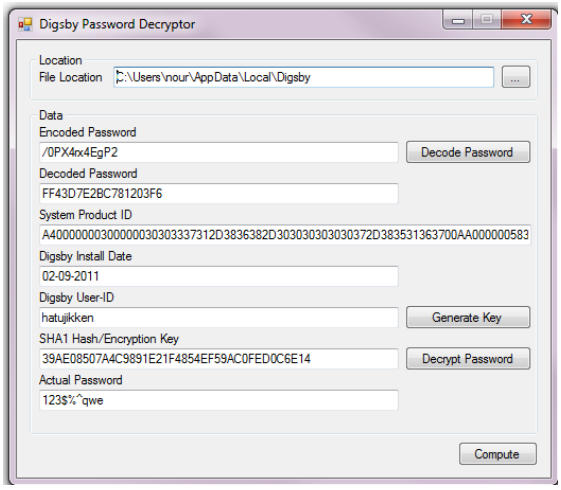

Fig. 5. GUI of Digsby password decryptor

7.1 Insider Attack

Digsby does not store the login credentials of IM, email accounts and social networks at user's computer. It stores the hash of password of Digsby ID at server[5] to avoid insider attacks (their own employees should not be able to view the passwords of users). The login credentials of IM, email accounts, social networks are encrypted with plaintext of password used to login at Digsby. After the analysis of Digsby artefacts at client end, it is noticed that it uses SHA-1 hash function to generate hash for decryption of stored passwords and contact icons. There is a possibility that Digsby is also using SHA-1 hash to store the password hash on server. There are few methods to get access to these login

[5] http://wiki.digsby.com/doku.php?id=security

credentials. Insider who has rights to view the hash passwords stored on the server can copy that password hash and put it on various websites [6] to generate a password online. Once he gets the password of a user in plaintext, there is a chance for intruders to decrypt the login credentials of IM, email addresses and social network, if Digsby is using RC4 encryption algorithm at server side. Thereafter, attacker can get the encrypted password from the database and consequently he/she can generate all the saved passwords within a short period of time.

8 Conclusion and Future Work

In this paper, we have presented an analysis of Digsby log data that can be used by digital forensics examiners to trace user's activities performed using Digsby. We have also analyzed the password encryption method used in Digsby and developed an application to assist forensics examiners to decrypt passwords of the suspected users. Presently, we are working on finding digital traces from portable installations of Digsby. We are also working towards analyzing RAM and swap files to identify volatile, but possibly relevant and crucial digital traces.

References

1. H. Carvey, "Instant messaging investigations on a live windows xp system," *Digital investigation*, vol. 1, no. 4, pp. 256–260, 2004.
2. M. Dickson, "An examination into yahoo messenger 7.0 contact identification," *Digital investigation*, vol. 3, no. 3, pp. 159–165, 2006.
3. ——, "An examination into msn messenger 7.5 contact identification," *Digital investigation*, vol. 3, no. 2, pp. 79–83, 2006.
4. W. Van Dongen, "Forensic artefacts left by windows live messenger 8.0," *Digital Investigation*, vol. 4, no. 2, pp. 73–87, 2007.
5. M. Dickson, "An examination into aol instant messenger 5.5 contact identification," *Digital investigation*, vol. 3, no. 4, pp. 227–237, 2006.
6. W. van Dongen, "Forensic artefacts left by pidgin messenger 2.0," *Digital Investigation*, vol. 4, no. 3, pp. 138–145, 2007.
7. D. Farmer, "A forensic analysis of the windows registry," 2007.
8. M. Yasin, M. Wahla, and F. Kausar, "Analysis of download accelerator plus (dap) for forensic artefacts," in *Proceedings of International Conference on IT Security Incident Management and IT Forensics (IMF'09)*, 2009, pp. 142–152.
9. ——, "Analysis of free download manager for forensic artefacts," *Digital Forensics and Cyber Crime*, pp. 59–68, 2010.
10. M. Yasin, A. Cheema, and F. Kausar, "Analysis of internet download manager for collection of digital forensic artefacts," *Digital Investigation*, vol. 7, no. 1, pp. 90–94, 2010.
11. S. Pyne, "Internet explorer forensics: Reconstructing internet activity using pasco and galleta," 2007.

[6] http://www.md5decrypter.co.uk/sha1-decrypt.aspx, http://hashcrack.com/