CloudMe Forensics: A Case of Big-Data Investigation

Yee-Yang Teing[1,2], Ali Dehghantanha[2], and Kim-Kwang Raymond Choo[3,*,†]

[1]*Department of Computer Science, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, UPM Serdang, Selangor, Malaysia, 43400*
[2]*The School of Computing, Science & Engineering, Newton Building, University of Salford, Salford, Greater Manchester, United Kingdom, M5 4WT*
[3]*Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, USA, TX 78249-0631*

SUMMARY

The significant increase in the volume, variety and velocity of data complicates cloud forensic efforts, as such big data will, at some point, become computationally expensive to be fully extracted and analyzed in a timely manner. Thus, it is important for a digital forensic practitioner to have a well-rounded knowledge about the most relevant data artefacts that could be forensically recovered from the cloud product under investigation. In this paper, CloudMe, a popular cloud storage service, is studied. The types and locations of the artefacts relating to the installation and uninstallation of the client application, logging in and out, and file synchronization events from the computer desktop and mobile clients are described. Findings from this research will pave the way towards the development of tools and techniques (e.g. data mining techniques) for cloud-enabled big data endpoint forensics investigation.

KEY WORDS: Big data forensics, cloud forensics, CloudMe forensics, mobile forensics

1. INTRODUCTION

With continuing advances in broadband and pervasive media devices (e.g., smartphones and tablets), it is not uncommon to find the storage media in consumer devices to contain Terabytes (TB) of data. Federal Bureau of Investigation (FBI)'s 15 Regional Computer Forensic Laboratories, for example, reported that the average amount of data they processed in 2014 is 22.10 times the amount of data ten years ago, up from 22TB to 5060TB [1], [2]. The increase in storage capacity has a direct impact on cloud forensics and operational investigations; hence, it is inevitable that big data solutions will become an integral part of cloud forensics [3].

Due to the nature of cloud-enabled big data storage solutions, identification of forensic artefacts from the cloud hosting environment may be analogous to 'finding a needle in a haystack' [4]. The data could be segregated across multiple servers via virtualization [5]. Due to the lack of physical access to the cloud hosting environment, forensic examiners may need to rely on the Cloud Service Provider (CSP) for preservation of evidence at a lower level of abstraction. This may, however, not be viable due to service level agreements (SLAs) between a CSP and its users [6]–[14]. Even if the location of the data could be identified, traditional practices and approaches to computer forensic investigation are unlikely to be adequate [9]. For example, existing digital forensic practices generally require a bit-by-bit copy of an entire storage media [15]–[17], which is unrealistic and expensive on a large-scale dataset [12]. It has been demonstrated that it could take more than 9 hours to merely acquire 30GB of data from an Infrastructure as a Service (IaaS) cloud environment [18], [19]. Hence, the time required to acquire a significantly larger dataset could be considerably longer. These challenges are compounded in cross-jurisdictional investigations, which could prohibit the transfer of evidential data due to the lack of cross-nation legislative

agreements in place [16], [20]–[26]. Therefore, it is unsurprising that forensic analysis of cloud service endpoints remains an area of research interest [23], [27]–[33].

CloudMe (previously known as 'iCloud') is a Software as a Service (SaaS) cloud model currently owned and operated by Xcerion [34]. The free version of CloudMe offers up to 19 GB storage space (with referral program), and its premium version offers up to 500 GB storage space for individual users and 5 TB for business users [35]. CloudMe users may share contents with each other, as well as other public users, through email, text-messaging, Facebook and Google sharing. There are three modes of sharing in CloudMe, namely: WebShare, WebShare+, and Collaborate. WebShare only permits one-way sharing, where the recipients are not allowed to make changes to the shared folder. WebShare+ allows users to upload files/folders only, while collaborative sharing allows the recipients to add, edit or delete the content, even without the use of CloudMe client application [36]. The service can be accessed using the web User Interface (UI) as an Internet file system or the client applications, which are available for Microsoft Windows, Linux, Mac OSX, Android, iOS, Google TV, Samsung Smart TV, Western Digital TV, Windows Storage Servers, Novell's Dynamic File Services Suite, Novosoft Handy Backup etc. CloudMe is also compatible with third-party software and Internet services, enabling file compression, encryption, document viewing, video and music streaming etc. through the web/client applications [36].

In this paper, we seek to identify, collect, preserve, and analyze residual artefacts after using CloudMe cloud storage service on a range of end-point devices, as these devices are typically available to forensic investigators. Evidence recovered from these end-point devices could also be used to inform further investigation in a big data environment. The questions we seek to answer in this research are as follows:

1. What residual artefacts remain on the hard drive and physical memory after a user has used CloudMe desktop client application, and web application?
2. Where are such data remnants located on a Windows, Ubuntu, and Mac OS client device?
3. What Cloudme residual artefacts remain on the internal memory, and where are such data remnants located on an Android and iOS client device?

The structure of this paper is as follows. In the next section, we describe related work. Section 3 describes the experiment environment setup. In Section 4, we discuss the traces from the storage media and physical memory dumps of the desktop clients. Section 5 presents the findings from mobile clients and network traffic, respectively. Finally, we conclude the paper and outline potential future research areas in Section 6.

2. Related Work

The National Institute of Standard and Technology (NIST) defines cloud computing as

> *[a] model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*" [37].

The key aspects are to provide on-demand self-service, broad network access, resource pooling, rapid edacity, and measured services. There are three cloud computing service models [37], namely: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). NIST [37] also defined four deployment models as part of the cloud computing definition, which are public, private, community, and hybrid clouds. The public cloud is owned and operated by a provider organization. Users can subscribe to the service for a fee, based on the storage or bandwidth usage. On the other hand, the private cloud is tailored to a single organization's needs. The cloud infrastructure that is administered by organizations sharing common concerns (e.g., mission, security requirements, policy, and compliance considerations) are called community cloud.

Cloud computing is not without its own unique forensics challenges [39]. Jurisdiction differences, loss of data control, physical inaccessibility of evidences, multi-tenancy, and lack of tools for large scale distributed and virtualized systems are often cited as key cloud forensic challenges [40]–[43]. Other related

challenges include diverse range and types of digital media storage, decentralization, and utilization of anti-forensic and encryption techniques [42], [44], [45]. For example, Fahdi et al. [46] found that the top three cloud forensic challenges according to digital forensic practitioners are volume of data, legal aspect, and time, while the top three challenges raised by digital forensic researchers are time, volume of data, and automation of forensic analysis.

In the review of the 2011 Australian Federal Government's Cybercrime Bill amendment on mutual legal assistance requests, Hooper et al.[24] concluded that laws amendment in a single jurisdiction is unlikely to be adequate in addressing multi-jurisdiction investigation issues, such as in cloud computing environments. Martini and Choo [7], Taylor et at. [47], and Daryabar et al. [9] echoed the need for harmonize relevant legislation across jurisdictions, although realistically it is challenging due to the different judicial and legal systems internationally. Simou et al. [42] and Pichan et al. [43] added that our dependence on CSP also compound the challenges in all stages of cloud forensics (e.g., identify, preserve, analyze, and report [48], [49]). Therefore, Farina et al. [50] and Damshenas et al. [3], [11] suggested mitigating such limitations through clearly-defined Service Level Agreements (SLA) between CPSs and users.

Martini and Choo [51] proposed the first cloud forensic investigation framework, which was derived based upon the frameworks of McKemmish [52] and NIST [49]. The framework was used to investigate ownCloud [53], Amazon EC2 [18], VMWare [54], and XtreemFS [55]. Quick et al. [23] extended and validated the four-stage framework using SkyDrive, Dropbox, Google Drive, and ownCloud. Chung et al. [56] proposed a methodology for cloud investigation on Windows, Mac OSX, iOS, and Android devices. The methodology was then used to investigate Amazon S3, Google Docs, and Evernote. Scanlon et al. [57] outlined a methodology for remote acquisition of evidences from decentralized file synchronization networks and utilized it to investigate BitTorrent Sync [58]. In another study, Teing et al. [27] proposed a methodology to investigate the BitTorrent Sync application (version 2.0) or any third party and Original Equipment Manufacturer (OEM) applications. Do et al. [59] proposed an adversary model for digital forensics and demonstrated how such an adversary model can be used to investigate mobile devices (e.g. Android smartwatch – Do et al. [60] and apps). Ab Rahman et al. [61] proposed a conceptual forensic-by-design framework to integrate forensics tools and best practices in the design and development of cloud systems.

Marty [62] and Shields et al. [63] proposed a proactive application-level logging mechanism designed to log information of forensics interest. However, Zawoad and Hasan [64] argued that the proposed solutions may not be viable in real world scenarios. Forensic researchers such as Dykstra and Sherman [65], Gebhardt and Reiser [66], Quick et al. [23], and Martini and Choo [54], on the other hand, presented methods and prototype implementations to support the (remote) collection of evidential materials using Application Programming Interfaces (API). Quick and Choo [67] and Teing et al. [29] studied the integrity of data downloaded from the web and desktop clients of Dropbox, Google Drive, Skydrive, and Symform and determined that the act of downloading files from client applications does not breach the evidence integrity (e.g., no change in the hash values), despite changes in file creation/modification time.

In addition to remote collection of evidences, scholars also studied the potential of on-device collection of cloud artefacts such as from Evernote [56], Amazon S3 [56], Dropbox [33], [56], Google Drive [31], [56], Microsoft Skydrive [32], Amazon Cloud Drive [68], BitTorrent Sync [27], [69], SugarSync [70], Ubuntu One [30], huBic [71], Mega [72], Syncany [28], SpiderOak, JustCloud, pCloud [73], as well as different mobile cloud apps [20], [74]. Quick and Choo [31]–[33] also determined that data erasing tools such as Eraser and CCleaner may not completely remove the data remnants from Dropbox, Google Drive, and Microsoft SkyDrive. We are not aware of any published research focusing on forensic investigation of CloudMe SaaS cloud, and this is the gap we aim to contribute to in this research.

3. Research Methodology

We adopted the research methodology of Quick and Choo [31]–[33] and Teing et al. [16], [26]–[29] in the design of our experiments. The first step was to set up the test environments for the desktop and mobile

clients. The desktop environment consisted of three Virtual Machines (VMs) with following configurations:

- Windows 8.1 Professional (Service Pack 1, 64-bit, build 9600) with 2GB RAM and 20GB hard drive.
- Ubuntu 14.04.1 LTS with 1GB RAM and 20GB hard disk.
- Mac OS X Mavericks 10.9.5 with 1GB RAM and 60GB hard drive.

The VMs were hosted using VMware Fusion Professional version 7.0.0 (2103067) on a Macbook Pro running Mac OS X Mavericks 10.9.5, with a 2.6GHz Intel Core i7 processor and 16GB of RAM. As noted by Quick and Choo [31]–[33], it would have been laborious to replicate the experimental environment setup on a physical workstation. The client mobile devices comprised a factory restored iPhone 4 running iOS 7.1.2 and an HTC One X running Android KitKat 4.4.4, which were jailbroken/rooted with 'Pangu8 Version 1.1' and 'Odin3 Version 185' (respectively) to enable root access to the user's partition. We then created a set of sample files for the file transfer experiments, which consisted of copies of the 3111th email message of the Berkeley Enron email dataset (downloaded from *http://bailando.sims.berkeley.edu/enron_email.html*) that were saved in .RTF, .TXT, .DOCX, .JPG (print screen), .ZIP, and .PDF formats. This provides a basis for replicating the experiment in future.

Similar to previous studies [29], [32], [70], [75], the fourth step of the methodology involved conducting a predefined set of experiments such as installation and uninstallation of the CloudMe client applications as well as uploading, downloading, viewing, deleting, unsyncing, sharing, and inactivating sync files/folders to simulate various real world scenarios of using the CloudMe desktop, mobile, and web applications. The web application was accessed using the Google Chrome client for Windows version 51.0.2704.103m. Before each experiment, we made a base snapshot of each VM workstation to serve as the control case. After each experiment, we created a snapshot of the VM workstations before taking a copy of the virtual memory and disk file (after system's shutdown) in bit-stream (dd) and Encase Evidence (E01) formats, respectively. It was considered that undertaking analysis on the virtual memory (.VMEM) and disk (.VMDK) file would prevent the memory/image acquisition tools from altering the data in the storage media and physical memory [31]–[33]. As for the mobile clients, we made binary images using 'dd' over SSH/ADB Shell.

In the fifth step, we set up a forensic workstation with the tools in Table I. We then collected data relevant to the CloudMe investigation in the sixth step, before analysing the data in the seventh step. In the former, we extracted data that matched the terms 'cloudme', 'xcerion, and 'Enron3111′ from the forensic images for analysis using the tools of relevance in the latter. These included SQLite database files, PLIST files, prefetch files, event logs, shortcuts, thumbnail cache, $MFT, $LogFile, $UsnJrnl, as well as web browser files (e.g., in *%AppData%\Local\Google, %AppData%\Local\Microsoft\Windows\WebCache, %AppData%\Roaming\Mozilla, %AppData%\Local\Microsoft\Windows\Temporary Files\index.dat*). The artefacts from the physical memory dump were collected using Volatility, Photorec file carver, and HxD Hex Editor; network traffic using Wireshark and NetMiner. For all the data collected, both MD5 and SHA1 hash values were calculated and subsequently verified. All experiments were repeated thrice (at different dates) to ensure consistency of findings.

4. CloudMe Analysis on Desktop Clients

The installation of the CloudMe desktop clients created the data directory at *%AppData%\Local\CloudMe, /home/<User Profile>/.local/share/CloudMe, /Users/<User Profile>/Library/Application Support/CloudMe* on the Windows, Ubuntu, and Mac OS desktop clients. The sync (download) folders were located in the OS' Documents directory, such as *%Users%\[User Profile]\Documents, /home/[User Profile]/Documents, /User/[User Profile]/Documents/* on the Windows, Ubuntu and Mac OS clients by default. When the sync folders with the option "When delete folder in the cloud and all its content is selected." in the client applications were deleted, we observed that the sync folders remained locally but

were removed completely from the server. In all scenarios, the data and download directories remained after uninstallation of the client applications.

4.1 Cache.db Database

The file synchronization metadata and cloud transaction records could be predominantly located in the */%CloudMe%/cache.db* database (in the data directory). The tables of forensic interest are 'user_table', 'syncfolder_table', 'syncfolder_folder_table', and 'syncfolder_document_table'. The 'user_table' holds the property information of users who had logged in from the desktop client applications; 'syncfolder_table' maintains a list of metadata associated with the sync folder(s) added by or download to the local device; 'syncfolder_folder_table' keeps track of the tree structure for the sync folder(s); and 'syncfolder_document_table' records the metadata associated with the synced files in the sync folder(s). Table columns of forensic interest are presented in Table II.

To construct a meaningful file synchronisation timeline, we threaded the data fields in the four tables to provide the information such as: Which are the synced files? Where are the locations? Who is the owner of the files? What time was the files created? What is the last sync time. **Error! Reference source not found.** shows the SQL query used to parse Cache.db and produce synchronization history shown in **Error! Reference source not found.**.

4.2 CloudMe Registry, Sync.conf, and com.CloudMe.Sync.plist Files

Examination of the Windows registry revealed the username for the currently logged in user and the device name in *HKEY_USERS\<SID>\Software\CloudMe\Sync\startup\me* and *HKEY_USERS\<SID>\Software\CloudMe\Sync\<Username>_xClientId* (respectively). The username can be a useful identifying information for the cache.db database's remnants i.e., locating copies of the 'user_table' data in physical memory dumps. The client ID is a unique 32-character alphanumeric string used to identify a CloudMe device, which can be used to correlate residual evidences.

In Ubuntu client, both username and clientID were located in the */home/<User Profile>/.config/CloudMe/Sync.conf* file, by looking at values for entries 'me' (of the 'startup' property) and '_xClientId' (of the 'Username' property) respectively. In the Mac OSX client, Username and ClientID were located in the 'startup.me' and '[Username].xClientId' properties of the */Users/<User Profile>/Library/Preferences/com.CloudMe.Sync.plist* file.

4.3 Cloudme Log Files

Log files play a vital role in an incident investigation [13]. The CloudMe log files are located in the 'logs' subdirectory and created daily and named as [Year-Month-Day].txt. Although the log file only recorded application errors, it was possible to identify the file synchronization time alongside the sync path from the log entries such as "*2016-03-15 14:52:02: CloudMeUnthreaded: Request error: "/Users/alice/Documents/UbuntuShareFolder /UbuntuSubFolder/UbuntuSubFolder/Enron3111.docx" | "Error downloading https://os.cloudme.com/v1/ users/12886417622/favorites/112112/webshare/UbuntuSubFolder/UbuntuSubFolder/Enron3111.docx - server replied: Not Found" Error number: 203*", "*2016-03-15 14:56:30: onSyncRequestFailed: "WindowsSubFolder/WindowsSubFolder/Enron3111.pdf" | Type: "Uploading" | Error: "7"*"", "*2016-03-15 14:56:30: SYNC_FILE_NOT_FOUND— SYNC_FOLDER_NOT_FOUND: (0) "WindowsSubFolder/WindowsSubFolder/Enron3111.pdf*" :" and "2016-03-15 14:51:52: addRemoveLocalFolder:Fail: "/home/suspectpc/UbuntuSyncFolder/UbuntuSubFolder "". We could also recover the login time alongside the logged in username from the log entry "*2016-03-15 13:48:22: Logged in as: "adamthomson"*".

4.4 Web Browser Artefacts

Web browsing activities history is a critical source of evidence [29], [31]–[33], [53]. Our analysis of the web browsing history found unique identifying URLs associated with the user actions. For example, when accessing a sync folder in the CloudMe web application, we observed following URLs:

- *https://www.cloudme.com/en#files:/Documents/<Folder name>*,
- *https://www.cloudme.com/en#files:/f:<Folder ID>*,
- *https://www.cloudme.com/en#sync:/f: <Folder ID>*,
- *https://www.cloudme.com/en#sync:/<Folder ID>*, and
- *https://www.cloudme.com/en#sync:/f:<Folder ID>, <Folder name>*.

Accessing or downloaded a sync file produced following URL:

- *https://www.cloudme.com/v1/documents/<Folder ID>/<Document ID>/1/<Filename>*.

When we accessed the folders shared with other users, we observed the following URL:

https://www.cloudme.com/en#webshares:/<Folder name>.

Accessing the folder shared by other users produced the URL:

- *https://www.cloudme.com/en#following:/<Folder name>*

The download URL for the shared file could be discerned from:

- *https://www.cloudme.com/v1/documents/<Folder ID>/<Document ID>/1/<Filename>?dl=<Filename>*.

The web client's logout action generated from the following URL:

- *https://www.cloudme.com/en?r=1458192365602&logout=1*.

Rebuilding the web browsing caches produced the root directory for the web application at *www.cloudme.com/v1*. In particular, within the */%v1%/folders* directory, we recovered a list of metadata files for the sync folders accessed by the user, which could be differentiated by the folder ID. **Error! Reference source not found.** illustrates the metadata information associated with the sync folders; each of which creates a 'folder' subtag to house the folder ID and name, and a 'tag' subtag to hold the folder sharing information such as the webshare ID and folder sharing type i.e., in the 'group' property.

A search for the filenames of the sample files recovered files viewed on the web application in cache at */%v1%/documents/<Folder ID>/<Document ID>/1/*. We also recovered thumbnails for the viewed files in */%v1%/documents/<Folder ID>/<Document ID>/<Thumbnail ID>*. Notice that the */%v1%/documents* directory will always contain at least one folder i.e., holding the metadata files associated with the sync devices at */%v1%/documents/<Folder ID>/<Document ID for device-specific metadata file>\1*. **Error! Reference source not found.** shows the recovered device name and client ID from the 'dName' and 'clientId' properties of the 'sync' tag in the metadata file. Each sync folder creates a 'syncfolder' subtag to define the folder name, directory path, folder ID, last sync time, and information about whether the sync folder has been synchronised and if it is a favourite folder in the 'name', path', 'folderId,' lastSync', 'hasSynchronized', and 'favoriteFolder' properties respectively.

Another directory of interest within the 'v1' directory is the user-specific *%v1%\users\<User ID>* directory, which maintains a list of OpenSearch [72] description documents containing a wealth of folder metadata of forensic interest about the sync folders [73]. For example, the *%v1%/users/<User ID>/favorites/extended=true&order=favoritename&count=1000&offset=0&_=1458191.xml* document holds the OpenSearch description for the favourite folders. The metadata of interest recovered from this document include the folder IDs, folder names, folder sharing passwords, webshare IDs, as well as usernames and user IDs for the favourite folders in the 'folder_id', 'name', 'password', 'webShareId', sharingUserId', 'sharingUserName' properties of the sync folder/file-specific 'favorite' subtags (see **Error! Reference source not found.**). The *%v1%/users/<User ID>/webshares/order=name&desc=false&count=1000&offset=0&resources=true&_=145.xml* document defines the OpenSearch property of the shared folders/files, such as the update time, creation time, passwords, creators' IDs, webshare IDs in the 'updated', created', 'password', 'userId', and 'id' properties of the sync folder/file-specific 'webshare' subtags. The folder name and ID could be discerned from the 'name' and 'id' properties of the 'folder' subtag (see **Error! Reference source not found.**). Further details of the folder/file sharing could be located in the *%v1%/users/<User ID>/lifestream* document, such as the senders' user ID, senders' group ID, senders' username, receivers' user ID,

receivers' group ID, receiver's username, favourite IDs (for favourite folders), and whether the sharing has been seen in the 'senderId', 'senderGroupId', 'senderName', 'receiverId', 'receiverGroupId', 'receiverName', 'parentFolder', and 'seen' properties in the 'event' subtags.

4.5 Physical Memory Analysis

For all investigated client applications, analysis of the physical memory dumps using the 'pslist' function of Volatility resulted in the recovery of the process name, process identifier (PID), parent process identifiers (PPID), and process initiation time (which echoed the observation of [16]). We determined that the CloudMe process could be differentiated using the process names 'CloudMe.exe', 'cloudme-sync' and 'CloudMe' on the Windows, Ubuntu and Mac OS clients, respectively.

Undertaking data carving of the memory image of the CloudMe process determined that the files of forensic interest such as cache.db, sync.config, and CloudMe logs could be recovered. When CloudMe was accessed using the web client, we recovered copies of the OpenSearch description documents containing the folder sharing passwords from the web browser's memory space intact. Unsurprisingly, we also recovered copies of the database, configuration, and log files in plain text. For the cache database, a search for the username for the user located the data [74] of the 'user_table', which holds the user ID in the row ID variant field of the cell header section [74] in hex format. Once the user ID is identified, a practitioner may locate the file offsets contained between the cell data section of the 'syncfolder_document_table', 'syncfolder_folder_table' and 'syncfolder_table' tables, and work backwards to read the header field type varints [74] to recover the remaining data fields.

5. CLOUDME ANALYSIS ON MOBILE CLIENTS

Our examinations of the CloudMe mobile clients determined that the data directory is located in */private/var/mobile/Applications/<Universally Unique Identifier (UUID) for the CloudMe iOS app>/* and */data/data/com.excerion.android* on the iOS and Android clients. Although the mobile clients did not keep a copy of the sync folders from the user's account (like the desktop clients), it was possible to recover copies of the viewed files from *%<Universally Unique Identifier (UUID) for the CloudMe iOS app>%/Documents/persistentCache/* and */storage/sdcard0/Android/data/com.xcerion.android/cache/files/Downloads/* of the iOS and Android clients by default.

5.1 com.xcerion.icloud.iphone.plist and user_data.xml Files

A closer examination of the files in the directory listings located the username and password in plaintext in the 'username' and 'password' properties of the *%<Universally Unique Identifier (UUID) for the CloudMe iOS app>%/Library/Preferences/com.xcerion.icloud.iphone.plist* and *%com.excerion.android%/shared_prefs/user_data.xml* files. The former also held the last upload time in datetime format in the '<username_LastUploadTime>' property.

5.2 db.sdb Database

Analysis of the Android client revealed the cache database at */storage/sdcard0/Android/data/com.xcerion.android/cache/db.sdb*. The tables of interest with the cache database are 'files' and 'folders'. The 'files' table maintains a list of metadata of the sync files viewed by the user, while the 'folders' table holds the metadata of the sync folders associated with the user's account. db.sdb Database shows the table fields of interest from the db.sdb database. We also proposed using a SQL query to thread the table fields of interest from the tables to present the records in a forensically-friendly format as shown in **Error! Reference source not found.**.

5.3 Cache.db Database

Further examination of the iOS client recovered copies of the responses for the web API queries in the *%<Universally Unique Identifier (UUID) for the CloudMe iOS app>%/Library/Caches/com.*

xcerion.icloud.iphone/nsurlcache/Cache.db database. Specifically, we located the cached items in the 'receiver_data' table column of the cfurl_cache_receiver_data table in Binary Large OBject (BLOB) including metadata files and OpenSearch documents for the sync folders. Within the cfurl_cache_response table, we located the corresponding URLs and timestamps in datetime format, in the "request_key" and "time_stamp" table columns, respectively. By threading the data fields us ing the SQL query "*SELECT cfurl_cache_receiver_data.receiver_data, cfurl_cache_response.request_key, cfurl_cache_response.time_stamp FROM cfurl_cache_receiver_data, cfurl_cache_response WHERE cfurl _cache_receiver_data.entry_ID=cfurl_cache_response.entry_ID*", it was possible to correlate the cached items with the URLs and timestamps.

6. Conclusion and Future Work

In this paper, we examined the client residual artefacts left by CloudMe SaaS cloud as a backbone for big data storage. Our research included installing the client applications as well as uploading, downloading, deleting, sharing and activating/inactivating the sync folders/files using the client and web applications. We determined that a forensic practitioner investigating CloudMe cloud application should pay attention to the cache database, web caches, and log and configuration files, as highlighted in TABLE IV. Unlike cloud applications such as BitTorrent Sync [27] and Symform [29], the CloudMe client applications did not create any identifying information (e.g., configuration file and cache folder) in the sync folders; hence, a practitioner cannot identify the sync directories from the directory listing. This also indicates that the cache database is a critical source of evidence for the synchronization metadata and cloud transaction records, and hence should not be overlooked.

Analysis of the mobile clients determined that the findings were not as conclusive in comparison with the desktop clients, and only the viewed files could be recovered. This indicated that the iOS and Android mobile clients are merely a UI for the web application. Our examination of the web browsing activities identified unique URLs that can aid in identification of the user actions made to the web application, such as login, logout, and accessing and downloading sync files/folders. Although the application layer was fully encrypted (using HTTPS), we were able to recover the root directory for the web application from the web browser's caches unencrypted, which included viewed files and metadata files and OpenSearch documents for the sync files/folders that contain the timestamp information and sharing passwords for the sync folders/files. However, a practitioner should note that the availability of the cached items depends on the API requests made to the web application; hence, the artefacts may not be consistent across different occasions.

Our analysis of the physical memory captures revealed that the memory dumps may provide potential for alternative methods for recovering applications cache, logs, configuration files and other files of forensic interest. It was also possible to recover the folder sharing password from the web cache in plain text, but not for the login password. This suggested that a practitioner can only obtain the login password from the mobile clients, using WebBrowserPassView when manually saved in the web browsers, through an offline brute-force technique, or directly from the user. However, it should be emphasized that the data in physical memory may be overwritten on low memory and system's shut down [16]. Therefore, obtaining the memory snapshot as quickly as possible increases the likelihood of preserving the artefacts.

We suggest future work extend this study to other popular and contemporary cloud storage services to provide further understanding of the big data artefacts from different cloud deployment models, which could lay the foundation for the development of data reduction techniques (e.g., data mining and intelligence analysis) for these technologies [79], [80].

References

[1] U.S. Department of Justice and Federal Bureau of Investigation, "Regional Computer Forensics Laboratory Annual Report for Fiscal Year 2014." U.S. Department of Justice, 2014.

[2] D. Quick and K.-K. R. Choo, "Data Reduction and Data Mining Framework for Digital Forensic Evidence: Storage, Intelligence, Review, and Archive.," *Trends Issues Crime Crim. Justice*, vol. 480, pp. 1–11, Sep. 2014.

[3] M. Damshenas, A. Dehghantanha, R. Mahmoud, and S. bin Shamsuddin, "Forensics Investigation Challenges in Cloud Computing Environments," in *Proceedings of 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec)*, 2012, pp. 190–194.

[4] S. Watson and A. Dehghantanha, "Digital forensics: the missing piece of the Internet of Things promise," *Comput. Fraud Secur.*, vol. 2016, no. 6, pp. 5–8, Jun. 2016.

[5] F. Daryabar, A. Dehghantanha, F. Norouzi, and F. Mahmoodi, "Analysis of virtual honeynet and VLAN-based virtual networks," in *Proceedings of 2011 International Symposium on Humanities, Science and Engineering Research*, 2011, pp. 73–77.

[6] K.-K. R. Choo, "Cloud Computing: Challenges and Future Directions," Australian Institute of Criminology, report, 2010.

[7] B. Martini and K.-K. R. Choo, "Cloud Forensic Technical Challenges and Solutions: A Snapshot," *IEEE Cloud Comput.*, vol. 1, no. 4, pp. 20–25, Nov. 2014.

[8] K.-K. R. Choo, "Organised Crime Groups in Cyberspace: A Typology," *Trends Organ. Crime*, vol. 11, no. 3, pp. 270–295, Jul. 2008.

[9] F. Daryabar and A. Dehghantanha, "A Review On Impacts Of Cloud Computing and Digital Forensics," *Int. J. Cyber-Secur. Digit. Forensics IJCSDF*, vol. 3, no. 4, pp. 183–199, 2014.

[10] F. Daryabar, A. Dehghantanha, N. I. Udzir, N. F. binti M. Sani, and S. bin Shamsuddin, "A Review on Impacts Of Cloud Computing and Digital Forensics," *Int. J. Cyber-Secur. Digit. Forensics IJCSDF*, vol. 2, no. 2, pp. 77–94, 2013.

[11] M. Damshenas, A. Dehghantanha, and R. Mahmoud, "A survey on digital forensics trends," *Int. J. Cyber-Secur. Digit. Forensics*, vol. 3, no. 4, pp. 209–235, 2014.

[12] S. Nepal, R. Ranjan, and K. K. R. Choo, "Trustworthy Processing of Healthcare Big Data in Hybrid Clouds," *IEEE Cloud Comput.*, vol. 2, no. 2, pp. 78–84, Mar. 2015.

[13] N. H. Ab Rahman and K.-K. R. Choo, "A Survey of Information Security Incident Handling in the Cloud," *Comput. Secur.*, vol. 49, no. C, pp. 45–69, Mar. 2015.

[14] A. Dehghantanha and K. Franke, "Privacy-Respecting Digital Investigation," in *Proceedings of 2014 Twelfth Annual International Conference on Privacy, Security and Trust (PST)*, 2014, pp. 129–138.

[15] F. N. Dezfouli, A. Dehghantanha, B. Eterovic-Soric, and K.-K. R. Choo, "Investigating Social Networking applications on smartphones detecting Facebook, Twitter, LinkedIn and Google+ artefacts on Android and iOS platforms," *Aust. J. Forensic Sci.*, pp. 1–20, Aug. 2015.

[16] T. Y. Yang, A. Dehghantanha, K.-K. R. Choo, and Z. Muda, "Windows Instant Messaging App Forensics: Facebook and Skype as Case Studies," *PLOS ONE*, vol. 11, no. 3, p. e0150300, Mar. 2016.

[17] D. Quick and K.-K. R. Choo, "Big Forensic Data Reduction: Digital Forensic Images and Electronic Evidence," *Clust. Comput.*, vol. 19, no. 2, pp. 1–18, Mar. 2016.

[18] N. Thethi and A. Keane, "Digital Forensics Investigations in the Cloud," in *Proceedings of 2014 IEEE International Advance Computing Conference (IACC)*, 2014, pp. 1475–1480.

[19] J. Dykstra and A. T. Sherman, "Acquiring forensic evidence from infrastructure-as-a-service cloud computing: Exploring and evaluating tools, trust, and techniques," *Digit. Investig.*, vol. 9, Supplement, pp. S90–S98, Aug. 2012.

[20] B. Martini, Q. Do, and K.-K. R. Choo, "Chapter 15 - Mobile Cloud Forensics: An Analysis of Seven Popular Android Apps," in *The Cloud Security Ecosystem*, Boston: Syngress, 2015, pp. 309–345.

[21] Y. Najvadi and A. Dehghantanha, "Forensic investigation of social media and instant messaging services in Firefox OS: Facebook, Twitter, Google+, Telegram, OpenWapp, and Line as case studies," in *Contemporary Digital Forensic Investigations Of Cloud And Mobile Applications*, Elsevier, 2016.

[22] Y. Mohd Najwadi and A. Dehghantanha, "Network Traffic Forensics on Firefox Mobile Os: Facebook, Twitter and Telegram as Case Studies," in *Contemporary Digital Forensic Investigations of Cloud And Mobile Applications*, Elsevier, 2016.

[23] D. Quick, B. Martini, and R. Choo, *Cloud Storage Forensics*, 1st ed. Syngress, 2013.

[24] C. Hooper, B. Martini, and K.-K. R. Choo, "Cloud Computing and Its Implications for Cybercrime Investigations in Australia," *Comput. Law Secur. Rev.*, vol. 29, no. 2, pp. 152–163, Apr. 2013.

[25] National Institute of Standards and Technology (NIST), "NIST Cloud Computing Forensic Science Challenges." National Institute of Standards and Technology, 2014.

[26] T. Yee Yang, A. Dehghantanha, and M. Zaiton, "Investigating America Online instant messaging application : data remnants on Windows 8.1 client machine," in *Contemporary Digital Forensic Investigations Of Cloud And Mobile Applications*, K.-K. R. Choo and A. Dehghantanha, Eds. Elsevier, 2016.

[27] Y.-Y. Teing, A. Dehghantanha, K.-K. R. Choo, and L. T. Yang, "Forensic Investigation of P2P Cloud Storage Services and Backbone for IoT Networks: Bittorrent Sync as a Case Study," *Comput. Electr. Eng.*, vol. 22, no. 6, pp. 1–14, 2016.

[28] Y.-Y. Teing, A. Dehghantanha, K.-K. R. Choo, Z. Muda, M. T. Abdullah, and W.-C. Chai, "A Closer Look at Syncany Windows and Ubuntu Clients' Residual Artefacts," in *Security, Privacy and Anonymity in Computation, Communication and Storage*, G. Wang, I. Ray, J. M. A. Calero, and S. M. Thampi, Eds. Springer International Publishing, 2016, pp. 342–357.

[29] Y.-Y. Teing, A. Dehghantanha, K.-K. R. Choo, T. Dargahi, and M. Conti, "Forensic Investigation of Cooperative Storage Cloud Service: Symform as a Case Study," *J. Forensic Sci.*, p. n/a-n/a, Nov. 2016.

[30] M. Shariati, A. Dehghantanha, B. Martini, and K.-K. R. Choo, "Chapter 19 - Ubuntu One Investigation: Detecting Evidences on Client Machines," in *The Cloud Security Ecosystem*, Boston: Syngress, 2015, pp. 429–446.

[31] D. Quick and K.-K. R. Choo, "Google Drive: Forensic Analysis of Data Remnants," *J Netw Comput Appl*, vol. 40, pp. 179–193, Apr. 2014.

[32] D. Quick and K.-K. R. Choo, "Digital Droplets: Microsoft SkyDrive Forensic Data Remnants," *Future Gener. Comput. Syst.*, vol. 29, no. 6, pp. 1378–1394, Aug. 2013.

[33] D. Quick and K.-K. R. Choo, "Dropbox Analysis: Data Remnants on User Machines," *Digit. Investig.*, vol. 10, no. 1, pp. 3–18, Jun. 2013.

[34] CloudMe AB, "Interview with founder and CEO," 2016. [Online]. Available: https://www.cloudme.com/newsletter/2015-01_en.html. [Accessed: 26-May-2016].

[35] CloudMe AB, "Pricing," 2016. [Online]. Available: https://www.cloudme.com/en/pricing. [Accessed: 26-May-2016].

[36] CloudMe AB, "Tutorials on CloudMe," 2016. [Online]. Available: https://www.cloudme.com/en/tutorials. [Accessed: 26-May-2016].

[37] P. Mell and T. Grance, "The NIST definition of cloud computing." 2011.

[38] P. Mell and T. Grance, *The NIST definition of cloud computing*. 2011.

[39] K.-K. R. Choo and A. Dehghantanha, *Contemporary Digital Forensic Investigations of Cloud and Mobile Applications*. Syngress, 2016.

[40] K. Ruan, J. Carthy, T. Kechadi, and I. Baggili, "Cloud forensics definitions and critical criteria for cloud forensic capability: An overview of survey results," *Digit. Investig.*, vol. 10, no. 1, pp. 34–43, Jun. 2013.

[41] K. Ruan, I. Baggili, J. Carthy, and T. Kechadi, "Survey on Cloud Forensics and Critical Criteria for Cloud Forensic Capability: A Preliminary Analysis," *Electr. Comput. Eng. Comput. Sci. Fac. Publ.*, Jan. 2011.

[42] S. Simou, C. Kalloniatis, E. Kavakli, and S. Gritzalis, "Cloud Forensics: Identifying the Major Issues and Challenges," in *Advanced Information Systems Engineering*, Springer International Publishing, 2014, pp. 271–284.

[43] A. Pichan, M. Lazarescu, and S. T. Soh, "Cloud forensics: Technical challenges, solutions and comparative analysis," *Digit. Investig.*, vol. 13, pp. 38–57, Jun. 2015.

[44] D. Birk and C. Wegener, "Technical Issues of Forensic Investigations in Cloud Computing Environments," in *2011 IEEE Sixth International Workshop on Systematic Approaches to Digital Forensic Engineering (SADFE)*, 2011, pp. 1–10.

[45] G. Sibiya, H. S. Venter, and T. Fogwill, "Digital forensics in the Cloud: The state of the art," in *Proceedings of IST-Africa Conference, 2015*, 2015, pp. 1–9.

[46] M. A. Fahdi, N. L. Clarke, and S. M. Furnell, "Challenges to digital forensics: A survey of researchers amp; practitioners attitudes and opinions," in *2013 Information Security for South Africa*, 2013, pp. 1–8.

[47] M. Taylor, J. Haggerty, D. Gresty, P. Almond, and T. Berry, "Forensic investigation of social networking applications," *Netw. Secur.*, vol. 2014, no. 11, pp. 9–16, Nov. 2014.

[48] S. Wilkinson, "ACPO Good Practice Guide for Digital Evidence." Association of Chief Police Officers (ACPO), 2012.

[49] K. Kent, S. Chevalier, and T. Grance, "Guide to Integrating Forensic Techniques into Incident." 2006.

[50] J. Farina, M. Scanlon, N. A. Le-Khac, and M. T. Kechadi, "Overview of the Forensic Investigation of Cloud Services," in *2015 10th International Conference on Availability, Reliability and Security (ARES)*, 2015, pp. 556–565.

[51] B. Martini and K.-K. R. Choo, "An Integrated Conceptual Digital Forensic Framework for Cloud Computing," *Digit. Investig.*, vol. 9, no. 2, pp. 71–80, Nov. 2012.

[52] R. McKemmish, "What is Forensic Computing." Australian Institute of Criminology, Jun-1999.

[53] B. Martini and K.-K. R. Choo, "Cloud Storage Forensics: Owncloud as a Case Study," *Digit. Investig.*, vol. 10, no. 4, pp. 287–299, Dec. 2013.

[54] B. Martini and K.-K. R. Choo, "Remote Programmatic vCloud Forensics: A Six-Step Collection Process and a Proof of Concept," in *Proceedings of 13th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom 2014)*, 2014, pp. 935–942.

[55] B. Martini and K.-K. R. Choo, "Distributed Filesystem Forensics: Xtreemfs as a Case Study," *Digit. Investig.*, vol. 11, no. 4, pp. 295–313, Dec. 2014.

[56] H. Chung, J. Park, S. Lee, and C. Kang, "Digital Forensic Investigation of Cloud Storage Services," *Digit. Investig.*, vol. 9, no. 2, pp. 81–95, Nov. 2012.

[57] M. Scanlon, J. Farina, and M.-T. Kechadi, "BitTorrent Sync: Network Investigation Methodology," in *Proceedings of 2014 9th International Conference on Availability, Reliability and Security*, 2014, pp. 21–29.

[58] M. Scanlon, J. Farina, N. A. L. Khac, and T. Kechadi, "Leveraging Decentralization to Extend the Digital Evidence Acquisition Window: Case Study on BitTorrent Sync," *ArXiv14098486 Cs*, pp. 1–14, Sep. 2014.

[59] Q. Do, B. Martini, and K.-K. R. Choo, "A Forensically Sound Adversary Model for Mobile Devices," *PLoS ONE*, vol. 10, no. 9, p. e0138449, Sep. 2015.

[60] Q. Do, B. Martini, and K.-K. R. Choo, "Is the Data on Your Wearable Device Secure? An Android Wear Smartwatch Case Study," *Softw. Pract. Exp.*, Jan. 2016.

[61] N. H. Ab Rahman, N. D. W. Cahyani, and K.-K. R. Choo, "Cloud Incident Handling and Forensic-By-Design: Cloud Storage as a Case Study," *Concurr. Comput. Pract. Exp.*, Jan. 2016.

[62] R. Marty, "Cloud Application Logging for Forensics," in *Proceedings of the 2011 ACM Symposium on Applied Computing*, New York, NY, USA, 2011, pp. 178–184.
[63] C. Shields, O. Frieder, and M. Maloof, "A System for the Proactive, Continuous, and Efficient Collection of Digital Forensic Evidence," *Digit. Investig.*, vol. 8, Supplement, pp. S3–S13, Aug. 2011.
[64] S. Zawoad and R. Hasan, "Cloud Forensics: A Meta-Study of Challenges, Approaches, and Open Problems," *ArXiv13026312 Cs*, Feb. 2013.
[65] J. Dykstra and A. T. Sherman, "Design and Implementation of Frost: Digital Forensic Tools for the Openstack Cloud Computing Platform," *Digit. Investig.*, vol. 10, pp. S87–S95, Aug. 2013.
[66] T. Gebhardt and H. P. Reiser, "Network Forensics for Cloud Computing," in *Distributed Applications and Interoperable Systems*, J. Dowling and F. Taïani, Eds. Springer Berlin Heidelberg, 2013, pp. 29–42.
[67] D. Quick and K.-K. R. Choo, "Forensic Collection of Cloud Storage Data: Does the Act of Collection Result in Changes to the Data or Its Metadata?," *Digit. Investig.*, vol. 10, no. 3, pp. 266–277, Oct. 2013.
[68] J. S. Hale, "Amazon Cloud Drive Forensic Analysis," *Digit. Investig.*, vol. 10, no. 3, pp. 259–265, Oct. 2013.
[69] J. Farina, M. Scanlon, and M.-T. Kechadi, "BitTorrent Sync: First Impressions and Digital Forensic Implications," *Digit. Investig.*, vol. 11, Supplement 1, pp. S77–S86, May 2014.
[70] M. Shariati, A. Dehghantanha, and K.-K. R. Choo, "SugarSync Forensic Analysis," *Aust. J. Forensic Sci.*, vol. 48, no. 1, pp. 1–23, Apr. 2015.
[71] B. Blakeley, C. Cooney, A. Dehghantanha, and R. Aspin, "Cloud Storage Forensic: hubiC as a Case-Study," in *2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom)*, 2015, pp. 536–541.
[72] F. Daryabar, A. Dehghantanha, and K.-K. R. Choo, "Cloud storage forensics: MEGA as a case study," *Aust. J. Forensic Sci.*, pp. 1–14, Apr. 2016.
[73] H. Mohtasebi, A. Dehghantanha, and R. Choo, "Cloud storage forensics : analysis of data remnants on SpiderOak, JustCloud, and pCloud," in *Contemporary Digital Forensic Investigations Of Cloud And Mobile Applications*, Elsevier, 2016.
[74] F. Daryabar, A. Dehghantanha, B. Eterovic-Soric, and K.-K. R. Choo, "Forensic Investigation of Onedrive, Box, Google Drive and Dropbox Applications on Android and iOS Devices," *Aust. J. Forensic Sci.*, vol. 48, no. 6, pp. 1–28, Mar. 2016.
[75] N. D. W. Cahyani, B. Martini, K.-K. R. Choo, and A. M. N. Al-Azhar, "Forensic Data Acquisition from Cloud-of-Things Devices: Windows Smartphones as a Case Study," *Concurr. Comput. Pract. Exp.*, Jan. 2016.
[76] A9.com Inc., "OpenSearch 1.1 Draft 5," 2016. [Online]. Available: http://www.opensearch.org/Specifications/OpenSearch/1.1#The_.22OpenSearchDescription.22_element. [Accessed: 26-May-2016].
[77] CloudMe AB, "CloudMe Web Services," 2016. [Online]. Available: https://www.cloudme.com/en/api/webservices. [Accessed: 26-May-2016].
[78] SQLite, "Database File Format," 2016. [Online]. Available: https://www.sqlite.org/fileformat.html. [Accessed: 10-Nov-2016].
[79] D. Quick and K.-K. R. Choo, "Impacts of Increasing Volume of Digital Forensic Data: A Survey and Future Research Challenges," *Digit. Investig.*, vol. 11, no. 4, pp. 273–294, Dec. 2014.
[80] A. Dehghantanha, "Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, Github, and More," *J. Inf. Priv. Secur.*, vol. 11, no. 2, pp. 137–138, Apr. 2015.

TABLE I
TOOLS PREPARED FOR SYNCANY FORENSICS

Tools	Usage
FTK Imager Version 3.2.0.0	To create a forensic image of the .VMDK files.
dd version 1.3.4-1	To produce a bit-for-bit image of the .VMEM files.
Autopsy 3.1.1	To produce directory listings for the forensic images as well as extracting files and analysing the Windows Registry, swap file/partition, and unallocated space from the forensic images.
HxD Version 1.7.7.0	To conduct keyword searches in the forensic images.
Volatility 2.4	To extract the running processes and network information from the physical memory dumps, as well as dumping files from the memory space of the CloudMe client applications (i.e., using the 'pslist', 'netstat'/'netscan', and 'memdump' functions).
SQLite Browser Version 3.4.0 and SQLite Forensic Explorer version	To view the contents of SQLite database files.
Photorec 7.0	To data carve the forensic images.
File juicer 4.45	To extract files from files.
BrowsingHistoryView v.1.60	To analyse the web browsing history.
Nirsoft Web Browser Passview v1.58	To recover the credential details stored in web browsers.
Nirsoft cache viewer, ChromeCacheView 1.56, MozillaCacheView 1.62, IECacheView 1.53	To analyse the web browsing caches.
Thumbcacheviewer Version 1.0.2.7	To examine the Windows thumbnail cache.
Windows Event Viewer Version 1.0	To view the Windows event logs.
Console Version 10.10 (543)	To view log files.
Windows File Analyser 2.6.0.0	To analyse the Windows prefetch and link files.
NTFS Log Tracker	To parse and analyse the $LogFile, $MFT, and $UsnJrnl New Technology File System (NTFS) files.
Plist Explorer v1.0	To examine the contents of the Apple Property List (PLIST) files.

TABLE II
TABLES AND TABLE COLUMNS OF FORENSIC INTERESTS FROM CACHE.DB

Table	Table Column	Relevance
user_table	user_id	A unique numerical user ID for the user(s) logged in from the local device. This ID could assist a practitioner in correlating any user-specific data that might have been obtained from other sources of evidence.
	username	Username provided by the user during registration.
	devicename	Device name provided by the user during registration.
	created	Holds the addition time of the user account(s) in datetime format.
syncfolder_table	owner	Owner's ID which correlates with the 'user_id' table column of the 'user_table' table.
	name	Folder name.
	local_path	Local directory path.
	cloud_path	Server's directory path.
	folder_id	A unique numeric folder ID for the sync folder(s).
	created	Folder creation date in datetime format
	last_run	Last sync time in datetime format.
	inactivated	Folder has been inactivated; 'true' if yes, 'false' if no.
	encrypted	Folder has been encrypted; 'true' if yes, 'false' if no.
Syncfolder_folder_table	name	Folder name which correlates with the 'name' table column of the 'syncfolder_table' table.
	root_folder_id	Folder ID for the root sync folder, which correlates with the 'folder_id' table column of the 'syncfolder_table' table.
	folder_id	Folder ID for the sync folder(s), including the folder ID for the subfolder(s).
	child_folder_id	A unique numeric folder ID for the subfolder(s) associated with the sync folder(s). The root folder retains its original folder ID unchanged.
	creation_date	Folder creation time in datetime format.
	deleted	Folder has been deleted; *NULL* if not deleted.
	owner	Owner's ID for the sync folder(s), which correlates with the 'user_id' table column of ' user_table' table.
syncfolder_document_table	owner	Owner's ID for the sync folder(s), which correlates with the 'user_id' table column of ' user_table' table.
	name	Folder name.
	root_folder_id	Folder ID for the root sync folder.
	folder_id	Folder ID for the sync folder(s), including the folder ID for the subfolder(s), which correlates with the 'child_folder_id' table column of the 'syncfolder_folder_table' table.
	document_id	A unique numeric document ID for the sync file(s).
	size	File size.
	modified_date	Last modified date in datetime format.
	checksum	MD5 checksum for the modified document.
	main_checksum	MD5 checksum for the original document.

TABLE III

TABLE FIELDS OF FORENSIC INTEREST FROM THE DB.SDB DATABASE.

Table	Table column	Relevance
files	_id	A unique numerical user ID used to identify a CloudMe sync file.
	name	Filename for the sync file.
	folder_id	Folder ID for the folder housing the sync file.
	size	File size for the sync file.
	href	URL to the sync file.
	published	Sync file addition time in datetime format.
	updated	Last updated time of sync file in datetime format.
	owner	Owner's name of the sync file.
	Mime	Multipurpose Internet Mail Extensions (MIME) format of the sync file.
folders	Owner	Owner's name of the sync folder.
	Folder_id	A unique numerical user ID used to identify a CloudMe sync folder.
	Name	Folder's name.
	Parent	Folder's name for the parent folder.
	Is_root	Whether the sync folder is a root folder?
	Path	Original directory path for the sync folder.

TABLE IV
LOCATIONS OF FILES OF FORENSIC INTEREST FROM CLOUDME

Content	Directory Paths
Database	• Cache.db database in *%AppData%\Local\CloudMe*, */home/<User Profile>/.local/share/CloudMe/*, */Users/<User Profile>/Library/Application Support/CloudMe/*, and *%<Universally Unique Identifier (UUID) for the CloudMe iOS app>%/Library/Caches/com.xcerion.icloud.iphone/nsurlcache/*of the Windows, Ubuntu, Mac OS, and iOS clients. • */storage/sdcard0/Android/data/com.xcerion.android/cache/db.sdb* on the Android client.
Log files	• [Year-Month-Day].txt in *%AppData%\Local\CloudMe\logs*, */home/<User Profile>/.local/share/CloudMe/logs/*, */Users/<User Profile>/Library/Application Support/CloudMe/logs/*, and of the Windows, Ubuntu, and Mac OS clients.
Default download directory	• *%Users%\[User Profile]\Documents*, */home/[User Profile]/Documents/*, */User/[User Profile]/Documents/*, *%<Universally Unique Identifier (UUID) for the CloudMe iOS app>%/Documents\persistentCache*, and */storage/sdcard0/Android/data/com.xcerion.android/cache/files/Downloads/* on the Windows, Ubuntu, Mac OS, iOS, and Android clients.
Configuration files	• *HKEY_USERS\<SID>\Software\CloudMe registry key*, */home/<User Profile>/.config/CloudMe/Sync.conf*, */Users/<User Profile>/Library/Preferences/com.CloudMe.Sync.plist*, *%<Universally Unique Identifier (UUID) for the CloudMe iOS app>%/Library/Preferences/com.xcerion.icloud.iphone.plist*, and *%com.excerion.android%/shared_prefs/user_data.xml* files on the Windows, Ubuntu, Mac OS, iOS, and Android clients.
Web caches	• *www.cloudme.com/v1* directory of the web application.