Creating Super Timelines in Windows Investigations

Stephen Esposito, Gilbert Peterson

Chapter 9

CREATING SUPER TIMELINES IN WINDOWS INVESTIGATIONS

Stephen Esposito and Gilbert Peterson

Abstract As the applications and adoption of networked electronic devices grow, their use in conjunction with crimes also increases. Extracting probative evidence from these devices requires experienced digital forensic practitioners to use specialized tools that help interpret the raw binary data present in digital media. After the evidentiary artifacts are collected, an important goal of the practitioner is to assemble a narrative that describes when the events of interest occurred based on the timestamps of the artifacts. Unfortunately, generating and evaluating super timelines is a manual and labor-intensive process. This paper describes a technique that aids the practitioner in this process by generating queries that extract and connect the temporal artifacts, and produce concise timelines. Application of the queries to a simulated incident demonstrates their ability to reduce the number of artifacts from hundreds of thousands artifacts to a few hundred or less, and to facilitate the understanding of the activities surrounding the incident.

Keywords: Windows forensics, events, artifacts, super timelines

1. Introduction

Computers and digital devices are ubiquitous, as a result these devices are often present at crime scenes or are used to commit crimes. This makes the devices valuable sources of evidence in criminal investigations. One of the key tasks when examining a device is to understand how the temporal nature of the relevant data items correlate with each other. For example, while it is important to identify when a virus first appeared in a network, it is also important to understand how other temporal data items correlate with the arrival of the virus, such as the users that were logged in and their activities at the time of the infection. Connecting these temporal items together creates a timeline of events

that can explain and confirm the details of an incident to include who, what, where and when.

This paper discusses how a timeline of events can be developed and processed into a concise listing, and subsequently analyzed to correlate temporal items and produce knowledge about how an incident occurred. Four queries are presented that help convert raw file and event log timestamps to a general timeline of what the user and executing processes were doing on a Windows system during the period of interest. The results of testing the queries on a simulated incident demonstrate a significant reduction in the number of data items to be evaluated, which greatly reduces the evidence analysis and correlation time.

2. Related Work

Creating a timeline of the various events that occurred during an incident is one of the key tasks performed by a digital forensic practitioner [4]. Several tools have been developed for extracting temporal artifacts from specific data sources. For example, `fls` [1] extracts filesystem metadata while RegRipper [2] and `yaru` [9] extract Windows registry details that can identify the ordering of events. Pasco [5] and Web Historian [6] are web history extraction tools that parse Internet histories. EnCase and FTK also provide search functionality to find timestamp information, but neither tool has a super timeline capability [7]. The two tools produce a narrow view of user activities, requiring the practitioner to manually standardize, connect and correlate the data to produce a quality super timeline.

Instead of running each tool individually, the `log2timeline` tool can be used to extract all the artifacts and more in order to produce a single super timeline for analysis. The tool engages a Perl script to parse a file structure and artifacts. The tool has several input and output modules that enable it to be used in a variety of operating system environments. The practitioner identifies the operating system when executing `log2timeline` and the tool automatically uses all the modules available for the operating system.

The SIMILE output module [8] is a key `log2timeline` module. The module provides output in a standardized format that enables a SIMILE widget to read and display a moving timeline of information. While this may seem like an ideal way to view `log2timeline` output, it is difficult to use the tool to view more than 200 items because the SIMILE widget loads and operates very slowly. Because of this limitation, a better option is to produce the output as a comma separated value (CSV) file. This format simplifies the importation of data to a database.

3. Timestamp Anti-Forensics

An important issue to discuss at this stage is the possibility of anti-forensic tools being used to alter timestamp data. For example, the Timestomp [3] tool can alter the modification, access and creation (MAC) timestamps in the $STANDARD_INFORMATION attribute of a master file table (MFT) entry, albeit not the four timestamps present in the $FILE_NAME attribute. An attacker intending to cover his tracks can use Timestomp to alter the timestamps of incriminating files. This makes the creation of an accurate timeline much more difficult.

If tampering is suspected, all eight timestamps in the MFT, the file metadata timestamps and the recently-used registry keys should be correlated to identify tampering. It is also important to search for traces of the execution of tools such as Timestomp and any artifacts that may reside in the filesystem.

4. Timeline Artifact Analysis

Temporal data analysis comprises the collection and extraction of artifacts, correlation of the artifacts, and the determination of when an activity occurred. Our collection and extraction process involves generating an image of the hard drive and executing the `log2timeline` tool on the image to extract timestamp data.

The `log2timeline` CSV output file was imported into a Microsoft Access database with the fields shown in Table 1. SQL queries were developed to produce timelines of the temporal artifacts that correlate the timestamp data, which were subsequently used to reconstruct the timeline of events that occurred just before, during and after the incident. SQL queries providing the following information were developed to aid a digital forensic practitioner in correlating artifacts:

- All user activities (items not associated with a user are excluded).
- Web history traffic with user login and logoff times.
  - Refined query with keyword search.
  - Additional refined query to add a user.
- Documents recently accessed by a user.
- All login, logoff and program execution times within a specific timeframe.

The SQL queries were designed for reusability and to reduce the time and effort required to investigate potentially hundreds of thousands of

Table 1. Data items in the `log2timeline` CSV file.

Field Name	Data Type
ID	AutoNumber
date	Date/Time
time	Date/Time
timezone	Text
MACB	Text
source	Text
sourcetype	Text
type	Text
user	Text
host	Text
short	Memo
file_description	Memo
version	Number
filename	Memo
inode	Number
notes	Memo
format	Text
extra	Text

artifacts. The keywords and time frames may change, but the queries would remain the same.

4.1 User Program Activity Query

If no information is available about an incident, then a listing of the users and what each user did on the system is a good way to begin the search for information. The user program activity query gives the practitioner an overall view of a user's general activity. The query displays artifacts like the user assist registry key, which is created when the user executes a program such as Notepad. But it does not display operating system artifacts such as MFT and prefetch artifacts for the same execution of Notepad. If there is no user associated with a timestamp, then `log2timeline` places a "–" in the user field, which the query uses to identify non-user events.

A user program activity query has the following format:

```
SELECT L2t.[date], L2t.[time], L2t.[source],
       L2t.[sourcetype], L2t.[type], L2t.[user],
       L2t.[short], L2t.[desc], L2t.[filename],
       L2t.[notes], L2t.[format], L2t.[extra]
FROM L2t
WHERE ((NOT (L2t.[user])="-"))
ORDER BY L2t.[date] DESC, L2t.[time] DESC;
```

4.2 User Web History Query

Digital forensic practitioners are often required to investigate incidents involving the intentional or unintentional access to an inappropriate or unauthorized website. In such a situation, a practitioner would be interested in extracting web history artifacts to identify the websites accessed by a user. The login and logoff events corresponding to the user can be found by querying for the audit log event IDs 528 and 538 in a Windows XP system, and the event IDs 4624 and 4647 in a Windows 7 system. Web history artifacts may be found by searching for artifacts for which the source is logged as `WEBHIST`. The following query uses the term `(filename) Like "*Security/528;*"` to search for `Security/528;` anywhere in an artifact.

```
SELECT date, time, source, user, filename
FROM log2timeline
WHERE (source = "WEBHIST") OR ((filename) Like "*Security/528;*")
                           OR ((filename) Like "*Security/538;*")
ORDER BY date DESC, time DESC;
```

This is different from only using `source = "WEBHIST"` because the resulting query will only find artifacts where the source is logged exactly as `WEBHIST`.

4.3 Recent Document Access Query

Other important information relates to documents "touched" by a user. In a Windows system, whenever a file or resource is touched by a user, a link file is created in the user's recently accessed folder. The following query for all recent documents can provide useful information about user activities in an investigation:

```
SELECT L2t.date, L2t.time, L2t.source, L2t.sourcetype,
       L2t.type, L2t.user, L2t.filename, L2t.short,
       L2t.desc, L2t.notes, L2t.extra
FROM L2t
WHERE (((L2t.desc) Like "*recent*"))
ORDER BY L2t.date DESC, L2t.time DESC;
```

4.4 Processes Executed During User Login Timeframe Query

After a user or timeframe has been identified, a query for artifacts of executable programs during a user's login timeframe allows a practitioner to focus on the main activity on the computer. Once the practitioner understands the user's activities, the query can be expanded to

include all the artifacts during the timeframe in order to add correlative temporal artifacts to the timeline. The basic query for executable artifacts in a specific timeframe is:

```
SELECT date, time, source, user, file_description
FROM log2timeline
WHERE (((date)=#4/27/2012#) AND ((time)>#17:50:52# AND
        (time)<#18:21:8#) AND ((source = "Event Log") AND
        ((file_description) Like "*528*" OR
        (file_description) Like "*538*")) OR
       (((date)=#4/27/2012#) AND ((time)>#17:50:52# AND
        (time)<#18:21:8#) AND ((file_description) Like "*.exe*"))
ORDER BY Date DESC, Time DESC;
```

This query displays the executable programs (`"*.exe*"`) and user login and logoff times (`"*528*"` or `"*538*"`) from 27 April 2012 between the times 17:50:52 and 18:21:08. To expand the artifact listing to display all the artifacts during a given timeframe, the digital forensic professional can use the same query, but remove the section of the statement that displays the executable programs (`(File_Description) Like "*.exe*"`).

5. Test Results

The test data involved a scripted Windows XP incident created with specific activities. In the incident, Miss Scarlet logged into a Windows XP system at 17:50. She opened the Notepad program and then the Calculator program. After closing both programs, she inserted a USB memory stick and ran the program `ProcessList.exe`, which spawned two programs: (i) `ProcessHacker.exe`, which she could see running on the desktop; and (ii) `Services2000.exe`, which she could not see running in the background. `Services2000.exe` corresponds to a Netcat program in listener mode that waits for a hacker to connect to the system. When the Netcat connection is established, a command shell is passed to the hacker allowing him access to Miss Scarlet's Windows XP system. Miss Scarlet left `ProcessHacker.exe` running on the desktop and ran the Solitaire program. She played one game of Solitaire then logged off the system at 18:21.

This incident centers around a file named *HateLetter.txt* containing text that appears to be generated by the user logged in as Miss Scarlet and addressed to Mr. Boddy. The text says that "[she has] had enough of his antics and is going to kill him." The question is whether or not the file was created by Miss Scarlet.

The `log2timeline` tool was executed on a `dd` image of the simulated incident to perform the data collection of temporal artifacts required

Time	Source Type	Description
17:50:55	Event Log	2012 - C:/Program Files/VMware/VMware Tools/vmtoolsd.exe - 1928 - MissScarlet - (0x0-0x24DEF)
17:50:55	Event Log	2004 - C:/Program Files/VMware/VMware Tools/VMwareTray.exe - 1928 - MissScarlet - (0x0-0x24DEF)
17:50:55	Event Log	580 - C:/WINDOWS/system32/verclsid.exe - 1928 - MissScarlet - (0x0-0x24DEF)
17:50:55	**Event Log**	**1928 - C:/WINDOWS/explorer.exe - 332 - MissScarlet - (0x0-0x24DEF)**
17:50:55	Event Log	896 - C:/WINDOWS/system32/verclsid.exe - 1928 - MissScarlet - (0x0-0x24DEF)
17:52:25	Event Log	1436 - C:/WINDOWS/system32/notepad.exe - 1928 - MissScarlet - (0x0-0x24DEF)
17:53:08	Event Log	192 - C:/WINDOWS/system32/verclsid.exe - 1928 - MissScarlet - (0x0-0x24DEF)
17:53:08	Event Log	1568 - C:/WINDOWS/system32/verclsid.exe - 1928 - MissScarlet - (0x0-0x24DEF)
17:53:11	Event Log	1436 - C:/WINDOWS/system32/notepad.exe - MissScarlet - (0x0-0x24DEF)
17:54:01	Event Log	808 - C:/WINDOWS/system32/calc.exe - 1928 - MissScarlet - (0x0-0x24DEF)
17:54:22	Event Log	808 - C:/WINDOWS/system32/calc.exe - MissScarlet - (0x0-0x24DEF)
18:00:02	Event Log	772 - C:/WINDOWS/system32/verclsid.exe - 1928 - MissScarlet - (0x0-0x24DEF)
18:00:06	**Event Log**	**908 - /Device/Harddisk1/DP(1)0-0+5/ProcessList.exe - 1928 - MissScarlet - (0x0-0x24DEF)**
18:00:08	**Event Log**	**1612 - C:/DOCUME~1/MISSSC~1/LOCALS~1/Temp/eW_3.tmp/ProcessHacker.exe - 908 - MissScarlet - (0x0-0**
18:00:08	**Event Log**	**1824 - C:/DOCUME~1/MISSSC~1/LOCALS~1/Temp/eW_3.tmp/services2000.exe - 908 - MissScarlet - (0x0-0x2**
18:00:47	Event Log	1992 - C:/WINDOWS/system32/sol.exe - 1928 - MissScarlet - (0x0-0x24DEF)
18:05:54	Event Log	220 - C:/WINDOWS/system32/cmd.exe - 1824 - MissScarlet - (0x0-0x24DEF)
18:14:58	Event Log	1656 - C:/WINDOWS/system32/attrib.exe - 220 - MissScarlet - (0x0-0x24DEF)
18:16:01	Event Log	220 - C:/WINDOWS/system32/cmd.exe - MissScarlet - (0x0-0x24DEF)
18:16:01	Event Log	1824 - C:/DOCUME~1/MISSSC~1/LOCALS~1/Temp/eW 3.tmp/services2000.exe - MissScarlet - (0x0-0x24DEF)
18:20:18	Event Log	1992 - C:/WINDOWS/system32/sol.exe - MissScarlet - (0x0-0x24DEF)
18:20:32	Event Log	908 - /Device/Harddisk1/DP(1)0-0+5/ProcessList.exe - MissScarlet - (0x0-0x24DEF)
18:20:32	Event Log	1612 - C:/DOCUME~1/MISSSC~1/LOCALS~1/Temp/eW 3.tmp/ProcessHacker.exe - MissScarlet - (0x0-0x24DEF)
18:21:06	Event Log	1928 - C:/WINDOWS/explorer.exe - MissScarlet - (0x0-0x24DEF)
18:21:06	Event Log	2004 - C:/Program Files/VMware/VMware Tools/VMwareTray.exe - MissScarlet - (0x0-0x24DEF)

Figure 1. Subset of programs executed during Miss Scarlet's login time.

to construct a super timeline. The `dd` image was mounted with Mount Image Pro version 4. The `log2timeline` tool parsed each artifact to an output file. The `log2timeline` output was imported into a database with the fields shown in Table 1. Several queries were then executed.

The initial queries had a general scope, such as listing the login and logoff times of all users, and when executable programs began and ended execution. The results of these queries provide a digital forensic practitioner with a repeatable starting point for the incident. Because a specific timeframe based on the investigation was set, the super timeline could be narrowed down and the analysis could focus on artifacts that can be correlated in order to determine what happened. When multiple different artifacts, such as the user assist registry key, last accessed time of the file and audit log process begin time, all correlate to the same file and time, then the corresponding process can be confirmed to have executed during a specific timeframe.

5.1 Executing Processes and User Login Timeframe Activity

Based on the incident description, the executing processes and user login timeframe query focused on all programs that executed during Miss Scarlet's login time (i.e., like `.exe`).

The highlighted row in Figure 1 corresponds to when Miss Scarlet's Windows environment began by creating the `Explorer.exe` process

(PID 1928) with parent PID 332. The figure shows that the program `ProcessList.exe` was executed by Miss Scarlet, and that two additional programs were also executed within two seconds (6:00:06 to 6:00:08). This is suspicious behavior because the three programs were executed during a very short period of time, and only `ProcessList.exe` ran from a user directory while the other two programs ran from a temporary directory.

Examination of the three processes in question (highlighted area) reveals that `ProcessList.exe` (PID 908) ran from Miss Scarlet's environment since its parent process is 1928. The two questionable programs, `ProcessHacker.exe` (PID 1612) and `Services2000.exe` (PID 1824) are child processes of `ProcessList.exe` because their parent PID is 908, the PID assigned to `ProcessList.exe`.

More suspicion is raised when the `cmd.exe file` is examined. As highlighted by the arrows, the command prompt began at 18:05:54 creating the event ID 592, and the process ended when event ID 593 was logged at 18:16:01. The command prompt has a parent PID of 1824, which means it was spawned by `Services2000.exe`. All the files and events associated with this ten minute duration that the command prompt ran are now suspect.

The user program activity query reduced the number of entries from 303,000 to 3,000. This saves the digital forensic practitioner from having to sift through a massive number of artifacts. Note that, because Miss Scarlet did not use a browser, web history information would not be present during this time.

5.2 Recent Document Activity

Figure 2 shows the user assist keys from the time `ProcessList.exe` began (18:00:06) through the time the command prompt process ended (18:16:01). The user assist keys are in the registry *HKCU"Software"Microsoft"Windows"CurrentVersion"Explorer"UserAssist*, which logs when a user accesses objects. The arrow points to where `ProcessList.exe` began executing. The bold lines show where the command prompt process began executing. The highlighted line identifies the time when the file `HateLetter.txt` was accessed. This file was created at 18:09 and then modified again at 18:11. Since the file in question, `HateLetter.txt`, was created and modified during the window when the command prompt process was active, and the file has no owner attributes, this file was likely created by the command prompt process and not by Miss Scarlet. Additionally, Miss Scarlet did not run a command prompt nor did she run

Time	SRC	Source Type	Type	Description
18:00:06	**REG**	**UserAssist ke**	**Time of Launch**	**UEME_RUNPATH:E:/ProcessList.exe**
18:00:06	EVT	Event Log	Time generated/writt	Security/592;Success;908 - /Device/Harddisk1/DP(1)0-0+5/ProcessList.exe - 1928 - MissSca
18:00:07	FILE	NTFS $MFT	$SI [M..B] time	/Documents and Settings/MissScarlet/Local Settings/Temp/eW_3.tmp/ProcessHacker.exe
18:00:07	FILE	NTFS $MFT	$FN [MACB] time	/Documents and Settings/MissScarlet/Local Settings/Temp/eW_3.tmp/ProcessHacker.exe
18:00:07	FILE	NTFS $MFT	$SI [...B] time	/Documents and Settings/MissScarlet/Local Settings/Temp/eW_3.tmp
18:00:07	FILE	NTFS $MFT	$FN [MACB] time	/Documents and Settings/MissScarlet/Local Settings/Temp/eW_3.tmp
18:00:07	FILE	NTFS $MFT	$SI [.A..] time	/WINDOWS/system32/crtdll.dll
18:00:07	FILE	NTFS $MFT	$SI [M.C.] time	/Documents and Settings/MissScarlet/Local Settings/Temp
18:00:08	FILE	NTFS $MFT	$SI [.A..] time	/Documents and Settings/MissScarlet/Local Settings/Temp
18:00:08	FILE	NTFS $MFT	$SI [MACB] time	/Documents and Settings/MissScarlet/Local Settings/Temp/eW_3.tmp/services2000.exe
18:00:08	FILE	NTFS $MFT	$SI [MAC.] time	/Documents and Settings/MissScarlet/Local Settings/Temp/eW_3.tmp
18:00:08	FILE	NTFS $MFT	$SI [.AC.] time	/Documents and Settings/MissScarlet/Local Settings/Temp/eW_3.tmp/ProcessHacker.exe
18:00:08	FILE	NTFS $MFT	$FN [MACB] time	/Documents and Settings/MissScarlet/Local Settings/Temp/eW_3.tmp/services2000.exe
18:00:08	FILE	NTFS $MFT	$SI [.A..] time	/WINDOWS/system32/drivers/etc/services
18:00:08	EVT	Event Log	Time generated/writt	Security/592;Success;1612 - C:/DOCUME~1/MISSSC~1/LOCALS~1/Temp/eW_3.tmp/Proce
18:00:08	EVT	Event Log	Time generated/writt	Security/592;Success;1824 - C:/DOCUME~1/MISSSC~1/LOCALS~1/Temp/eW_3.tmp/servic
18:00:42	REG	NTUSER key	Last Written	Software/Microsoft/Windows/CurrentVersion/Explorer/MenuOrder/StartMenu2/Programs
18:00:47	FILE	NTFS $MFT	$SI [.A..] time	/WINDOWS/system32/sol.exe
18:00:47	REG	NTUSER key	Last Written	Software/Microsoft/Solitaire
18:00:47	EVT	Event Log	Time generated/writt	Security/592;Success;1992 - C:/WINDOWS/system32/sol.exe - 1928 - MissScarlet - VICTIM
18:00:47	REG	NTUSER key	Last Written	Software/Microsoft
18:00:47	REG	UserAssist key	Time of Launch	UEME_RUNPIDL:%csidl2%/Games/Solitaire.lnk
18:00:47	REG	NTUSER key	Last Written	Software/Microsoft/Windows/ShellNoRoam/MUICache
18:00:47	REG	UserAssist key	Time of Launch	UEME_RUNPATH
18:00:47	FILE	NTFS $MFT	$SI [.A..] time	/WINDOWS/system32/cards.dll
18:00:47	REG	UserAssist key	Time of Launch	UEME_RUNPIDL:%csidl2%/Games
18:00:47	REG	UserAssist key	Time of Launch	UEME_RUNPATH:C:/WINDOWS/system32/sol.exe
18:00:47	REG	UserAssist key	Time of Launch	UEME_RUNPIDL
18:05:54	**FILE**	**NTFS $MFT**	**$SI [.A..] time**	**/WINDOWS/system32/cmd.exe**
18:05:54	**PRE**	**XP Prefetch**	**Last run**	**CMD.EXE-087B4001.pf: CMD.EXE was executed**
18:05:54	**EVT**	**Event Log**	**Time generated/wri**	**Security/592;Success;220 - C:/WINDOWS/system32/cmd.exe - 1824 - MissScarlet - VIC**
18:06:04	**FILE**	**NTFS $MFT**	**$SI [MAC.] time**	**/WINDOWS/Prefetch/CMD.EXE-087B4001.pf**
18:09:00	**File**	**Dir**		**\Documents and Settings\MissScarlet\My Documents\HateLetter.txt**
18:11:00	File	Dir		\Documents and Settings\MissScarlet\My Documents\HateLetter.txt
18:12:02	FILE	NTFS $MFT	$SI [MAC.] time	/Documents and Settings/MissScarlet/My Documents
18:13:30	FILE	NTFS $MFT	$SI [.A..] time	/RECYCLER/S-1-5-21-1390067357-1078145449-839522115-1005
18:13:30	FILE	NTFS $MFT	$SI [.A..] time	/RECYCLER
18:14:58	FILE	NTFS $MFT	$SI [.A..] time	/WINDOWS/system32/ulib.dll
18:14:58	FILE	NTFS $MFT	$SI [.A..] time	/WINDOWS/system32/attrib.exe
18:14:58	EVT	Event Log	Time generated/writt	Security/592;Success;1656 - C:/WINDOWS/system32/attrib.exe - 220 - MissScarlet - VICTI
18:14:58	EVT	Event Log	Time generated/writt	Security/593;Success;1656 - C:/WINDOWS/system32/attrib.exe - MissScarlet - VICTIM - (0x
18:15:41	FILE	NTFS $MFT	$SI [.A..] time	/WINDOWS/system32/logon.scr
18:15:41	EVT	Event Log	Time generated/writt	Security/600;Success;628 - C:/WINDOWS/system32/winlogon.exe - VICTIM$ - WORKGRO
18:15:41	EVT	Event Log	Time generated/writt	Security/592;Success;1780 - C:/WINDOWS/system32/logon.scr - 628 - VICTIM$ - WORKGF
18:16:01	EVT	Event Log	Time generated/writt	Security/593;Success;220 - C:/WINDOWS/system32/cmd.exe - MissScarlet - VICTIM - (0x0-
18:16:01	EVT	Event Log	Time generated/writt	Security/593;Success;1824 - C:/DOCUME~1/MISSSC~1/LOCALS~1/Temp/eW_3.tmp/servic

Figure 2. Query results for the `ProcessList.exe` and `cmd.exe` processes.

Notepad to create file `HateLetter.txt` during this timeframe, placing even more suspicion on the command prompt process.

5.3 Summary

A variety of simple and complex queries can be created based on the evidence collected for an incident. The queries discussed above demonstrate how evidence can be quickly discovered using simple queries and how a practitioner can progress through a more complex, yet granular, examination of the artifacts. These queries enable the practitioner to reduce hundreds of thousands of artifacts to roughly one hundred artifacts or less, helping focus the investigation on the artifacts of interest.

6. Conclusions

Queries of the output of a tool such as `log2timeline` can be used very effectively to produce super timelines of events in incidents involving Windows systems. The super timelines provide excellent overviews

of the events that occurred before, during and after the incidents of interest. Four reusable queries were presented for extracting concise timeline artifacts that allow a digital forensic professional to decisively determine the activities involved in incidents. The queries can dramatically decrease the number of artifacts that need to be examined by digital forensic professionals – in the simulated incident, the number of artifacts was reduced from 303,000 to 100.

The approach presented in this paper does not incorporate auto correlation methods that could be used to fuse multiple timestamps associated with a single event; this is an important area for future exploration because of the variety of timestamps with different temporal resolutions that must be accounted for when creating a super timeline. Another area for future research involves the creation of an interactive visualization of the `log2timeline` output that would provide a means to view details at multiple levels of abstraction.

The views expressed herein are those of the authors and do not reflect the official policy or position of the U.S. Air Force, U.S. Department of Defense or the U.S. Government.

References

[1] B. Carrier, The Sleuth Kit (`www.sleuthkit.org`).

[2] H. Carvey, RegRipper (`regripper.wordpress.com`).

[3] J. Foster and V. Liu, Timestomp (`www.forensicswiki.org/wiki/Timestomp`).

[4] K. Guojonsson, Mastering the Super Timeline with `log2timeline`, SANS Gold Paper, SANS Institute, Bethesda, Maryland, 2010.

[5] K. Jones, Pasco v.1.0, McAfee, Santa Clara, California (`www.mcafee.com/us/downloads/free-tools/pasco.aspx`), 2012.

[6] Mandiant, Web Historian, Alexandria, Virginia (`www.mandiant.com/resources/download/web-historian`).

[7] J. Olsson and M. Boldt, Computer forensic timeline visualization tool, *Digital Investigation*, vol. 6(S), pp. S78–S87, 2009.

[8] SIMILE Project, The JFK assassination timeline with Dutch timeline labels, Massachusetts Institute of Technology, Cambridge, Massachusetts (`www.simile-widgets.org/timeline/examples/jfk_i18n/jfk.html`), 2009.

[9] TZWorks, Yet Another Registry Utility (`yaru`), Herndon, Virginia (`www.tzworks.net/download_links.php`).