IMPLEMENTING THE AUTOMATED PHASES OF THE PARTIALLY-AUTOMATED DIGITAL TRIAGE PROCESS MODEL

Gary Cantrell
Dixie State College of Utah
225 S. 700E
St. George, UT 84770
Phone: 435-879-4420
E-mail: cantrell@dixie.edu

Dr. David A Dampier
Mississippi State University
Mississippi State, MS 39762
Phone: 662-325-8923
E-mail: dampier@cse.msstate.edu

ABSTRACT

Digital triage is a pre-digital-forensic phase that sometimes takes place as a way of gathering quick intelligence. Although effort has been undertaken to model the digital forensics process, little has been done to-date to model digital triage. This work discusses the further development of a model that attempts to address digital triage, the Partially-automated Crime Specific Digital Triage Process model. The model itself will be presented along with a description of how its automated functionality was implemented to facilitate model testing.

Keywords: Computer Forensics, digital forensics, digital triage, evidence previewing, process modeling

1. DIGITAL TRIAGE

1.1 Digital Triage Defined

Digital triage, also known as digital evidence previewing, is the process of viewing digital evidence before the traditional digital forensic methodology as laid out by the first Digital Forensics Research Workshop (DFRWS) held in 2001 (Palmer, 2001). This traditional digital forensics process involves the imaging and authentication of all media before an examination begins, and digital triage occurs prior to this imaging process. Digital triage is therefore a pre-forensic examination process. The desired result of digital triage is quick

intelligence not necessarily court admissible evidence. Although, information obtained can be admitted in certain circumstances such as when only enough information is needed to seek a plea bargain or quick validation of evidence is required. This quick intelligence is used in the field for guiding search and seizure efforts, in the office for determining if a full examination is warranted, or in the lab for case prioritization decisions.

Other than with cell phones and other devices that must be examined "live," digital forensic tools seek to prevent any change to media under examination. Although digital triage tools are designed to protect the media under examination from alteration, this process is somewhat more volatile especially when being conducted on a live machine. This risk is acceptable if the tool is tested and the digital triage process used is well documented, explainable, and teachable. The triage documentation needs to be detailed enough to explain any changes made to the media during the examination within reason, and should follow the evidence throughout its life cycle to avoid any complications that may occur because it was subjected to digital triage.

1.2 Digital Triage Use

Digital triage is commonly performed on both "live" and "dead" evidence. In a "live" scenario the digital triage examiner is working on an active machine to extract data elements needed for the investigation. For example, in the case of a live server this would be performed to prevent down time of the server while it is imaged or to perform selective extraction of evidence due to storage constraints (Erin & Christopher, 2005). Another important use of "live" extraction with a digital triage tool is when encountering a volume with full disk encryption. Extracting data while the machine is still active allows the examiner to extract un-encrypted versions of all files. Once the machine is turned off, all files become inaccessible unless the password can be obtained. In this type of triage situation evidence alteration is unavoidable, but, as already discussed, with the proper documentation this should not be an issue to the courts.

"Dead" analysis is conducted on evidence that has already been powered off either because the computer housing it has been booted into a digital triage environment or it has been seized and powered down. In a situation where the computer has been seized and analysis software/hardware is available, the digital media can be removed from the suspect's machine and attached through a hardware write blocker to a another machine for analysis. Another option with both a "live" or "dead" machine would be to boot it into a digital triage environment with a live CD/DVD or bootable USB media. Once booted into this safe environment, the evidence is essentially in a "dead" state, and although not as secure as it would be with a hardware write blocker, it is protected from alteration. This technique can be used on site for intelligence or search and seizure guidance. It can also be used by offices that are not fully

equipped to perform a full digital examination and need quick intelligence or need to determine if the evidence is worth submitting to a lab for a full examination. Finally, it could also be used by the examining lab for evidence prioritization and case assignments.

Current digital triage tools include a host of live USB and live CD distributions (see http://www.livecdlist.com/ for examples of these distributions); law enforcement release only tools such as the FBI's Image Scan; and commercial tools such as IDEAL Corporation's STRIKE. These tools provide some automation and customization options allowing for "push button forensics", but the actual process of digital triage remains largely untested by the scientific community.

1.3 Digital Triage Modeling

Digital triage has been represented many times in court procedures and law enforcement training conferences, and has become an accepted digital forensics process. However although the scientific community has produced works related or useful to digital triage, there have only been a few works that have presented research in modeling the digital triage process. The Computer Forensics Field Process Model introduced by Rogers, Goldman, and Wedge (2006) is one of the few works that has presented a digital triage model, and it provides a good foundation for what can be gathered on a Windows machine. This foundation served as inspiration for the Computer Profiling stage of the model discussed in section 2 The Foundation Model. However, the Field Process Model is most useful to advanced users as it depends on a user's expertise to be successful. It also does not call for any automation limiting its use by the novice. These are both issues that were addressed with the implementation of the Partially-automated Crime Specific Digital Triage Model.

Some research has also been presented in evidence prioritization that can be used during or serve as a function of digital triage. One such work is the cross-drive analysis performed by Garfinkel (2006). In this work a technique to perform cross-drive analysis using pseudo-unique identifiers like social security numbers and credit card numbers to determine relationships between different sets of drives was demonstrated. This work was not intended to be a digital triage process, and thus, it provides too much information and is too time intensive for triage use. However, the work toward automating digital evidence classification could be very useful to digital triage analysis. Another purposed prioritization technique is the Five Minute Forensic technique created by Grillo, Lentini, Me, and Ottoni (2009). In this work specific information is extracted from evidence sets in an effort to prioritize them by user expertise. This proposed model was also not meant for digital triage as it requires training of the system with manually pre-classified hard drives. Although not specifically about digital triage, both of these works are very

closely related and provide foundation research.

The utility discussed in this work focuses on the recently proposed digital triage model presented as the Partially-automated Crime Specific Digital Triage Process Model (Cantrell, Dampier, Dandass, Niu, & Bogen, 2012). This model was introduced as automated enough to be easily learned and fast enough to be useful in a digital triage situation in the field. It was also important for the model to be adjustable for less time critical situations. Finally, with technology constantly changing it was vital for the process model to be expandable to serve future needs. The following section will describe this model and discuss how it was implemented to accomplish these goals.

2. FOUNDATION MODEL

2.1 Model Overview

The Partially-automated Crime Specific Digital Triage Process Model is a model currently being developed at Mississippi State University (Cantrell et al., 2012). The model is a semi-linear framework with a short series of automated phases. Figure 1 provides a full graphical representation of this model.

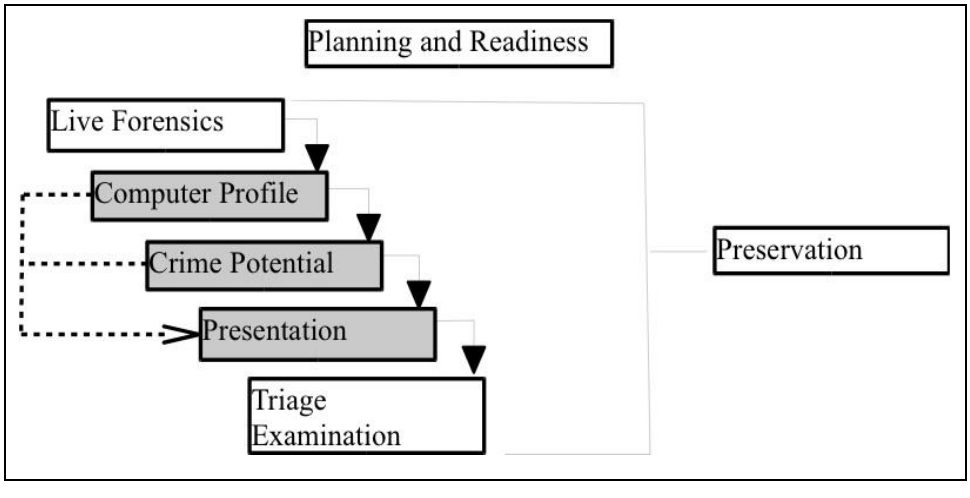

Figure 1 Partially-automated Digital Triage Process Model (Cantrell et al., 2012)

Planning and Readiness is the ongoing duty of being prepared and staying abreast of technology; Live Forensics is an optional phase involving the gathering of live memory before beginning any other data extraction; Preservation is an overarching principle throughout all phases involving the avoidance of any changes to the evidence where achievable; and the last phase

of Triage Examination is the ad-hoc process of examining the computer for intelligence guided by the results of the automated phases. The three phases in the middle are the main contribution of the model and are intended to be an automated process. Those three phases will now be described in detail. A complete description of the entire model can be found in the published manuscript (Cantrell et al., 2012).

2.2 The Automated Phases

The three middle phases shown in grey in Figure 1 are intended to be an automated process. The line on the left represents the data flow from one phase to the next. Although shown in Figure 1 as a step-by-step process, the data flow was actually implemented in a more iterative fashion per data extraction module. During the development of the utility used for implementation, it was decided to code each profile element in a modular fashion with each module completing before handing control to the next. For example, Web history information is gathered, filtered by criminal profile, and then added to the report. All data is gathered in a similar fashion instead of all the information being gathered, filtered, and presented. A more detailed description of the profiling functionality is described in section 3 Fast Modular Profiling Utility.

2.2.1 Computer Profiling Phase: The Computer Profiling phase involves the gathering of quick information about the computer. This information is divided into the three categories of File System Information, File Classification Information, and Application Information. The concept of gathering file system information for investigative purposes was taken from Carrier (2005), which describes digital investigation from the viewpoint of analyzing the file system itself. The extraction of file system information allows the digital triage examiner to quickly view system overview information such as the complexity of the system, the amount of data there is to sort through, the location of any large hidden areas, and an indication of user expertise based on file system choice and layout.

File classification is an attempt to determine what the computer is used for by identifying the types of files that are being stored on the drive(s). If a computer is a media machine, it is likely to have a large percentage of images and movies. If a computer is a business machine, it is more likely to have a majority of emails and documents. Being digital triage, this information is presented to the user in a summarized fashion that allows the triage examiner to quickly assess the most common file types per volume and when possible by user directory.

Application information is the information recorded by applications (including the operating system) about user activity typically without the user's knowledge and in some cases without their consent. For example, the operating system allows each user to create their own profile. Knowing how

many profiles are on the computer and what each profile is called can be a significant clue as to how the computer is used. The two other types of application information this model calls for are the extraction of Web browser history and Windows registry information. Unless set to not retain history, most Web browsers store history information for every URL visited over a period of time. This Web browser history often provides good information during a forensic examination, and its extraction is a simple matter once the browser history file structure is determined. The Windows registry is a central database of software and user settings that Windows retains, and can be a gold mine of useful digital forensic information. The extraction of information from the registry is more complicated than Web browser history, but can still be quickly accomplished.

2.2.2 Crime Potential and Presentation Phase: Digital examination in all forms is dependent on the situation and the type of crime under investigation. For example, investigation into a child pornography case will concentrate on the search for images, movies, peer-to-peer sharing programs, and so on. Other classes of crime have evidence type concentrations of their own. Although there have been calls for research in crime class modeling, 42nd Hawaii International Conference on System Sciences for example (Nance, Hay, & Bishop, 2009), little work has yet been done. Also unfortunately, the scientific community is in the early stages of developing data sets for digital forensics research (Garfinkel, Farrell, Roussev, & Dinolt, 2009). Customizing each investigation by crime class, for now, is up to individual examiners. This by itself would be a very interesting area of new research.

With the lack of crime class modeling in mind, this model was designed so that crime profiling could be done by each digital triage examiner based on their experience using a simple template. This template is shown in Table 1. The three categories of information are keywords, file type alerts, and known file alerts. For any digital triage tool to be useful, it must be fast and efficient. Therefore, the search for keywords has to be limited to simple surface data scans and not in-depth file or raw sector searches. The data from the Computer Profile phase is filtered through the Crime Potential phase before going to the Presentation phase. In the Presentation phase the data is displayed in two different reports. The main report contains all data collected, and the alert report includes only those things filtered by the Crime Potential phase. Section 3 Fast Modular Profiling Utility provides more detail as to how these stages were actually implemented.

Table 1 Example crime class template (Cantrell et al., 2012)

Keywords	*words of particular interest in a crime class or particular case*
File Type Alerts	*file types that would normally be found on a computer for a specific crime class*
Known File Alerts	*known files to be of interest in a particular case identified by file name*

3. FAST MODULAR PROFILING UTILITY

The authors of this work developed a series of scripts using the automated phases of the model described in section 2 as a foundation. Those automated phases are Computer Profiling, Crime Potential, and Presentation. These scripts gather quick useful information to create a profile of the computer. As the FMPU is creating this profile, the information is monitored for keywords to assist in crime potential determination. Lastly, it presents the information to the user in an HTML report format in both a main report and a red flag alert report.

This utility was titled the Fast Modular Profiling Utility FMPU for short. The FMPU is comprised of original and open source tools written in Perl. The following sections will describe its framework, present a few details about its development, describe its profiling capability, and finally discuss its initial evaluation.

3.1 FMPU Modular Process Framework

It was decided that the best methodology for creating a digital triage tool to implement this model would be one created in a modular fashion. This modular framework design allows for easy expansion, simple customization, and incorporates existing tools and commands where possible. Program execution can be summarized in the following steps:

Main module accepts as input: report name and location
Main module writes HTML header
Main module gives control over to module 1
Module 1 extracts information
Module 1 formats information as text, HTML table or separate HTML pages
Module 1 appends HTML table, link, or text to the appropriate report
Main module creates HTML footer to close the report

Steps 3 through 6 are executed for each data item extracted. A series of scripts

incorporating open source and original code were created to implement this design. For small amounts of information, the data is added directly to the report. Larger sub-reports are written as separate files and linked in to the main report or alert report.

3.2 FMPU Development Environment

The traditional digital forensics model requires the imaging and authentication of each piece of evidence prior to examination. Digital triage cannot wait for these time consuming steps. Digital triage must avoid changing the evidence when examining a "dead" system. Digital triage can be done in a lab or on site. However, some form of boot media must be used to incorporate full onsite capability. With these requirements in mind, the Linux distribution Caine installed to a USB drive was chosen as the development and testing environment. It is already designed to be a digital forensics distribution and can be used with USB drives allowing for the use of read-write media instead of write-once optical media. Using USB drives for the FMPU vehicle may restrict its use on older machines that do not allow for USB boot. However, transfer from USB boot media to optical would not be a difficult process. The Caine environment was used for development, testing, and for the final vehicle of the utility.

The FMPU uses original code, built in Linux commands, and calls other open source programs. It was also determined during development that it would have the need to quickly perform text parsing. With these requirements in mind the programming language Perl was chosen. Perl is also already included in Caine distribution requiring no modification of the Caine environment. Perl proved to be a wise selection as it easily facilitated the development of the FMPU.

3.3 FMPU Functionality

The FMPU gathers the following information:

File System Information
- Physical/logical disk layout
- Sector allocation
- File system types and locations

File Classification
- File type report for each user directory
- File type report per volume

Application Information
- Usernames on the system
- Web browser history
- Windows registry data

The file system information gathered includes the physical disks attached to the computer being examined, the logical volumes available for mounting, the

sector layout of the system, and the file system label of each volume. Viewing of the physical and logical layout of the system allows the digital triage examiner to quickly determine the amount of data on the system and the organization of each disk. Among the questions this will allow the examiner to answer are how many drives the user actually has connected, whether the user has a storage drive or drives, and what the basic disk layout of the system is.

The physical and logical disks attached to the system are derived by determining what the Caine operating system assigned mount points to during start up and presenting this information to the user. The sector layout of the disk and the file system label of each volume is determined using the Sleuth Kit's mmls command as shown in Table 2.

This quickly allows the digital triage examiner to determine the complexity of the disk layout; determine the potential for data recovery on the disk; theorize about the expertise of the user; and locate possible areas on the disk that could be used for data hiding. The amount of data recovery possible is dependent on the file system used to store the data. For example, Windows FAT file systems are notorious for leaving data remnants behind, but Linux based file systems are designed in such a way that data remnants are more quickly written over (Carrier, 2005). Also, a disk with multiple types of file systems or file systems that are less commonly used could indicate a more advanced user that is willing to experiment with different file systems instead of a user who sticks with the file system preinstalled on the machine.

Table 2 mmls sample output

DOS Partition Table
Offset Sector: 0
Units are in 512-byte sectors

	Slot	*Start*	*End*	*Length*	*Size*	*Description*
00:	*Meta*	*0000000000*	*0000000000*	*0000000001*	*0512B*	*Primary Table (#0)*
01:	*-----*	*0000000000*	*0000002047*	*0000002048*	*0001M*	*Unallocated*
02:	*00:00*	*0000002048*	*0001257471*	*0001255424*	*0613M*	*NTFS (0x07)*
03:	*00:01*	*0001257472*	*1953523119*	*1952265648*	*0930G*	*NTFS (0x07)*
04:	*-----*	*1953523120*	*1953525167*	*0000002048*	*0001M*	*Unallocated*

The Sleuth Kit suite of tools also includes a tool called sorter that will classify all files on a system. This tool proved too time intensive to utilize. The file classification report produced by the FMPU is instead compiled using native Linux commands. The determination of file type can be done in two different

ways. All files are in essence binary data and the examination of this binary data can often be used to identify the file. What makes a file useful is the program used to interpret that binary data into something useful to the user. Windows machines use the two or three letters following the last period in a file name, commonly called the file extension, to determine what program to use for this interpretation. These last three letters can also be used to identify the file type.

Using naming conventions for file type identification has a disadvantage. Windows operating systems typically do not use any verification of file type against the file name. This means nothing prohibits the user from renaming a file to any incorrect file extension. There is also the possibility of a glitch in the system resulting in files having an incorrect extension. For these reasons, digital forensics programs typically depend on the first few internal bytes of a file to determine the true file type.

The FMPU divides the output for file classification information by Windows user directory and by entire volumes. In addition, the user can select first byte signature identification or file extension identification for user directories and volumes independently. Table 3 shows a sample of data from a report created during testing. This table displays full volume file type classification as produced by the FMPU. As shown, this particular identification was done by extension instead of first byte signature analysis.

The choice of one technique over the other is dependent on the time critical nature of the situation. As explained, a first byte signature analysis is more trustworthy. However, each time a file is identified by signature, its first bytes have to be compared to a list of known byte signatures. This has to be an extensive list to be useful. In the initial tests this analysis performed on user directories added three minutes of processing time on the testing machine. Doing full byte signature analysis on complete volumes with Windows 7 installations increased this time to over twenty minutes due to the number of files that had to be examined. Full byte signature analysis on non-Windows operating system volumes varied by the amount of files stored on the drive. The compromise adopted was to set the tool by default to do full byte analysis when it encounters a user directory, but file extension analysis when it encounters a complete volume. During later testing this was changed to file extension analysis for both.

Table 3 Sample of full volume file type classification

File Extension	Occurrences
.mui	*7655*
.cat	*3710*
.png	*3301*
.DLL	*2629*
.exe	*2515*
.GPD	*4648*
.inf	*2130*
.xml	*2061*
.mum	*2010*
.sys	*1550*
.WMF	*1465*
.nib	*1444*
.xib	*1444*

Application information is information that is collected about the user by an application, including the operating system, potentially without user consent. Usernames for each user are collected by looking at the user directories as listed on the system. This can provide an indication of how many users are on the system and who those users might be. However, the digital triage examiner must remember that there is no easy way to tell who is actually using which account, and take this into consideration. It is also important to note, user directories can be placed in non-standard locations eliminating this benefit. However, for more advanced FMPU users the configuration file can be edited to restore this benefit by specifying the user directory location.

In order to facilitate future Web browsing, the default on most Web browsers is to keep a record of what sites a user has visited. Unless a user changes this setting, a Web history is maintained. In the first FMPU version, Web history analysis is performed only for Internet Explorer. Internet Explorer is arguably the most popular Web browser in use today (http://marketshare.hitslink.com/) and was thus chosen as a focus. Internet explorer stores its Web history in index.dat files. The structure of these files has been well researched and documented (Jones & Blani, 2010a, 2010b; Oh, Lee, & Lee, 2011). The URLs themselves are stored in plain text and extracted using Linux commands and basic Perl text parsing. Location of the index.dat files is contained in an easy to edit configuration file to allow for easy updating of the FMPU for older or newer versions of Internet Explorer. Future versions can easily incorporate other Web browser history files.

Extraction in this manner ignores other information that is included in the index.dat files. For example, along with the URL visited by the user the index.dat file also contains the time the URL was visited, whether it was intentional or a redirect, and the associated cached item if applicable (Jones & Blani, 2010a, 2010b). Further research will determine if it is useful to include this information in the final FMPU report as well as the URL. The information was intentionally excluded to streamline the output to the user.

The purpose of the FMPU is to selectively collect those items that are of the most interest to the digital examiner and provide that information in a useful fashion. The goal of the FMPU is not to present all the possible data. Therefore, a similar approach to what was done with the file classification analysis was also performed for the Web browser history analysis. In addition to listing all the URLs, each domain visited is counted. For example if a user visits www.website.com/link2 and then www.website.com/link1, the FMPU will report "www.website.com, 2." The final listing is then sorted by number and sent to the output. The raw output used to create this list is also included in the report in case the triage examiner needs more detail about a specific link. Table 4 provides sample output from a test report of domain summaries.

Table 4 Screen shot of domain name analysis

Domain	Occurrences
www.driveridentifier.com	*10*
mail.google.com	*9*
h2000.www2.hp.com	*7*
www.google.com	*6*
support.microsoft.com	*6*
www.tomshardware.com	*4*
driverboost.com	*3*
feeds.feedburner.com	*3*
www.getnotify.com	*2*
googleads.g.doubleclick.net	*2*
driver-id.info	*2*
www.bing.com	*2*
hotfixv4.microsoft.com	*2*
cdn.driverboost.com	*2*

Presenting the data in this manner allows the triage examiner to quickly determine what Websites have been visited and to what depth or frequency. A single visit to a Website could indicate a redirect or accident. A higher number will indicate multiple recorded visits or a much deeper exploration of the

Website. Number does not always signify importance however. A Website visited a single time might be an important clue or the same Website multiple times may not record each visit. Thus, lower on the list does not necessarily indicate less importance to the examiner.

The Windows registry is a database of settings often accessed by the digital examiner for information (Carvey, 2005; Dolan-Gavitt, 2008). The format of these settings are not intended to be user friendly, and are more often edited by individual programs not directly by the user. The registry creates a huge store of information that is often unknown and ignored by the typical user. This is the most complicated structure accessed by the FMPU. This access is accomplished through the open source tool RegRipper. RegRipper is a tool maintained and provided free of charge by Harlan Carvey (2012). The FMPU calls RegRipper to extract information and provides the results in the final report. The pieces of information, registry keys, that can be extracted are dependent on what modules have been written for RegRipper. As can be seen on the RegRipper Website (Carvey, 2012), modules are still being created and the public is encouraged to submit new modules if registry key information is found that would be useful to extract for which the tool does not currently have support.

The keys to be extracted are set by including or excluding their name from an input file to the FMPU. The following registry keys were selected as an initial set of information to extract for testing:

Per User Information Extracted:

- Logon name of the user used to verify the user list
- Websites typed directly into a Web browser
- Recently opened documents
- Recently run items from the command line box
- Media Player recently played files
- AOL instant messenger information
- Skype communication program settings
- Yahoo instant messenger settings
- MSN messenger settings

System Information:

- List of USB devices that have been attached to the system
- Shut down counts and times

Software Information:

- The default browser

Further analysis will determine the usefulness of this selection of information.

Adding or removing plugins does not noticeably affect execution time. So, it remains for each digital triage examiner to select the sets of information most useful to their situation. Figure 3 shows a small sample of program output.

```
TypedURLs
Software\Microsoft\Internet Explorer\TypedURLs
LastWrite Time Wed May 16 01:26:25 2012 (UTC)
  url1 -> http://www.google.com/
  url2 -> http://www.getnotify.com/
  url3 -> http://www.gmail.com/

**************************************************
Software\Microsoft\Windows\CurrentVersion\Explorer\TypedPaths
LastWrite Time Tue Sep 13 01:08:27 2011 (UTC)

Software\Microsoft\Windows\CurrentVersion\Explorer\TypedPaths has no values.

**************************************************
RecentDocs
**All values printed in MRUList\MRUListEx order.
Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs
LastWrite Time Thu Feb 16 18:27:17 2012 (UTC)
  1 = Downloads
  28 = Root.docx
  27 = super one click 1.7
  26 = instrucciones.txt
  25 = super one click 1.7.rar
  23 = New Text Document.txt
  24 = warning.vbs
  22 = inst_Funny Video.zip
  21 = UGACHAKA.AVI
```

Figure 3 Screen shot of registry analysis output

The final report is provided in the form of easy to navigate HTML pages. HTML is a common format for digital examination reports, and is more or less universal on all computer systems. The final report is separated into two reports one for all data and an alert report containing data that is identified during the crime potential phase. Users have the option to populate an input file of red flag words prior to running the FMPU. During execution this list is used to identify any data that might be of particular interest. File system information is not changed by this phase. File classification is filtered. As part of the file classification process a list of all files is created. During this sorting, file names containing any keywords or known file names are identified and flagged for the alert report. Application information is also filtered. As application information is gathered it is monitored for keywords or known file names and anything identified is also included in the alert report.

3.4 FMPU Evaluation

Initial developmental testing was done on a series of 300 GB drives with 2 partitions each. On all drives 1 partition had system information and the other had an active Windows 7 installation with multiple user accounts. Tests were run with the FMPU installed on a USB drive and on a CD-ROM. No significant run time difference was discovered between the two. The use of the CD-ROM was simpler as it did not include the flash drive itself with the test results. A command line option is available to ignore specified drives, but having to do so might lead to error and could confuse a novice digital triage examiner. Tests were also conducted using 1 to 3 drives at once. Time was affected by the number of drives processed, but the number of drives had not effect on accuracy. The final report is divided up per data extracted and then per drive examined.

FMPU functionality testing was also conducted with the use of 2 validation subjects and 12 testing subjects. The validation group consisted to 2 digital forensic examiners actively performing digital forensic examinations. They were provided the FMPU and asked to evaluate its use on real evidence. Each subject listed 5 pieces of information that would have been useful to know prior to their original examination or in a digital triage situation. These facts were items that would have facilitated digital triage, helped them prioritize evidence, or help guide their examination. Subject 1 found 4 out of 5 items and subject 2 found 3 out of 5 items. In addition subject 1 found 6 additional items of interest that would have been useful, and subject 2 found 5 additional items that would have been useful. As a result of this validation testing the FMPU was further modified to include a more comprehensive configuration file that allows the digital triage examiner to easily choose between Windows versions.

Initial quantitative data has also been taken using law enforcement officers with digital forensic process knowledge, but not active examiners. Three test sets were created through the use of student volunteers. One test drive simulated a child pornography case utilizing kitten images and phrases instead of actual child pornography. Another drive held a fictitious murder scenario, and the final drive contained the base image used to create the other two representing a drive containing nothing of interest. Subjects were given short descriptions of each case and then asked to use the FMPU to classify each drive accordingly. In addition, they were also asked to rate the level of confidence of their response. The available ratings were totally confident, somewhat confident, or complete guess. The experimental group was provide the FMPU and the control group was not.

At the time of this work 6 experimental subject tests and 6 control subject tests were conducted. In the experimental group all but one subject was able to identify all three drives correctly. The control group had, 1 with all 3 correct,

2 subjects with 1 correct, and 3 with no correctly identified drives. After testing with the first 4 subjects (2 control and 2 experimental), the tool was set to file extension identification instead of byte signature identification for both full volumes and user directories. File extension identification was used during the remainder of the testing. Identifying files by file extension is more risky as a user or program may misname files. However, this allowed for an 85% decrease in the amount of time it takes the FMPU to complete its analysis. Allowing for the selection of one analysis over the other can be left up to the digital triage examiner. With this modification in place there was a 65% decrease in examination time in favor of the experimental group without a decrease in accuracy.

4. FUTURE WORK AND CONCLUSION

This first iteration of the FMPU was a prototype with presets chosen by the designer. Possible future upgrades for this utility could include:

- Default settings chosen in a more scientific manner
- More Web browser support
- Option for a deeper file or sector scan

The selection of which file type classification to do, first byte or naming convention, and which registry keys to extract was made based on the developer's experience and interviews with currently working digital forensics examiners. The current series of test are being carried out with these settings and have the goal of testing the usefulness of the tool and the methodology it helps to implement. Once the value of the tool is verified, further testing should be carried out to better determine what default settings are the most useful. With file type determination there is an element of processing time that has to be evaluated, and with the registry keys there is the consideration of how much information is too much as it contributes to user's evaluation time.

The FMPU was designed to find and extract Web browser history for Internet Explorer. There are, of course, other Web browser options available to each user. A more comprehensive scan would include the search for these Web browsers as well. Also, currently the FMPU only extracts the URLs listed in the history. Another question that needs to be explored is the usefulness of including the other information such as time stamps and direct connection versus redirect information. The goal of the FMPU is to stream line all information to facilitate quick digital triage decisions.

For keyword searches the thoroughness of scans can be divided into 3 levels. The FMPU looks only at the information already being gathered when building its alert report, for example file names and Web history. This could be considered a level 1 scan. A level 2 scan would also include the scan for words inside files. A level 2 scan would take considerable more time. How

much time would be dependent on the number of files present on the system. A level 3 scan would be a sector-by-sector search for key words. Level 3 scans would take the most time, and be time dependent on the drive size itself. Level 1 scanning was chosen based on the idea that digital triage has to be as quick as possible to be useful. This is certainly true when performed in the field. The other two situations mentioned in the introduction were, in the office for determining if a full examination is warranted, or in the lab for case prioritization decisions. These situations are not as time critical, and what level of scan would be the most useful would be an interesting area for future work as well.

In conclusion, the FMPU was created to facilitate the testing of the Partially-automated Crime Specific Digital Triage Process Model described in section 2. Both the validation subjects and the test subjects described in section 3.4 are part of a larger series of tests that are still being conducted for this research. Once completed, it is planned that these tests be released as a future work at which time the tests themselves will also be described in more detail. However, these initial trials provide support that the FMPU does have value and is worth further testing.

REFERENCES

Cantrell, G., Dampier, D., Dandass, Y., Niu, N., & Bogen, C. (2012). Research toward a partially-automated, and crime specific digital triage process model. Computer and Information Science, 5(2), 29-38.

Carrier, B. (2005). File system analysis. Upper Saddle New Jersey: Addison-Wesley Professional.

Carvey, H. (2005, September). The windows registry as a forensic resource. Digital Investigation, 2(3), 201-205.

Carvey, H. (2012). The regripper. Retrieved from http://regripper.wordpress.com

Dolan-Gavitt, B. (2008). Forensic analysis of the windows registry in memory. Digital Investigation, 5 (supplement), 26-32.

Erin, K., & Christopher, B. (2005). Risk sensitive digital evidence collection. Digital Investigation, 2(2), 101-119.

Garfinkel, S. (2006, September). Forensic feature extraction and cross-drive analysis. Presented at 6th Digital Forensic Research Workshop.

Garfinkel, S., Farrell, P., Roussev, V., & Dinolt, G. (2009). Bringing science to digital forensics with standardized forensic corpora. Digital Investigation, 6

(supplemental), 2-11.

Grillo, A., Lentini, A., Me, G., & Ottoni, M. (2009, September). Fast user classifying to establish forensic analysis priorities. Presented at Fifth International Conference on IT Security Incident Management and IT Forensics.

Jones, K., & Blani, R. (2010a, November 2). Web browser forensics, part 1. Retrieved from http://www.symantec.com/connect/articles/web-browser-forensics-part-1

Jones, K., & Blani, R. (2010b, November 2). Web browser forensics, part 2. Retrieved from http://www.symantec.com/connect/articles/web-browser-forensics-part-2

Nance, K., Hay, B., & Bishop, M. (2009, January). Digital forensics: Defining a research agenda. In Proceedings of the 42nd Hawaii International Conference on System Sciences.

Oh, J., Lee, S., & Lee, S. (2011). Advanced evidence collection and analysis of web browser activity. Digital Investigation, 8 (supplemental), 62-70.

Palmer, G. (2001, August). A road map for digital forensic research. Presented at Digital Forensic Research Workshop, Utica, New York.

Rogers, K., Goldman, J., & Wedge, T. (2006). Computer forensic field triage model. Journal of Digital Forensics, Security and Law, 1(2), 19-38.