Modeling the Forensics Process

Sabah Al-Fedaghi and Bashayer Al-Babtain

Computer Engineering Department, Kuwait University
sabah@alfedaghi.com, Eng.bashayer@gmail.com

Abstract

Most forensic models focus on the investigative process and its different phases and are characterized by a rather informal and intuitive approach. This paper proposes an abstract model of the digital forensic model based on a new flow-based specification methodology. It is shown in examples that the method can uniformly specify the forensic process in various phases and across roles. It also provides more exact description where "things" (e.g., information, evidence) are separated into different streams of flow.

Keywords: *Forensics, digital investigation, conceptual modeling*

1. Introduction

Digital (computer) forensics is defined as "analytical and investigative techniques used for the preservation, identification, extraction, documentation, analysis and interpretation of computer media (digital data) which is stored or encoded for evidentiary and/or root cause analysis" [1]. Investigation in this area seeks to support court processing of criminal cases related to computers and networks as well as internal corporate investigations and disciplinary hearings [2]. It may involve the acquisition and analysis of digital evidence, authentication of documents, identification of sources and suspects, and so forth.

Achieving the objectives of a forensic investigation requires following several phases. Developing models for digital forensics aims to understand the scientific basis of the field [3]. According to [4], research in this area has a direct impact on

- prevention of further malicious events;
- improvement of current prevention mechanisms;
- improved standards for corporate security professionals;
- increased awareness of current vulnerabilities and prevention measures.

According to Tanner and Dampier [5], "Digital forensic investigations are becoming more complex due to the increasing size of digital storage… new approaches for managing the case details of a digital forensics investigation must be developed". Specifically, what is required is "to develop examination standards and to provide structure to computer forensic examinations ... and [to capture] the mechanism involved in digital forensics though diagrammatical specification of the involved process in such an operation" [4].

According to Köhn, Eloff, and Olivier [6], "Most of the modelling representations for forensic investigations found in the current literature [2003] are made in a rather informal and intuitive way". Also, "It can be safely said that [current] models are mainly ad-hoc and much needs to be accomplished in this particular domain" [4]. This paper proposes an abstract

model of the digital forensic procedure based on a new flow-based specification methodology that has been used in several research areas [7-11].

2. Brief Review

It is claimed that digital forensics is a process that can be modeled with some reasonably established phases [12]. Most proposed forensic models have focused on "the investigative process and the different phases, they addressed the complexity of an investigation and the features and functionality of devices, and the concrete principles of an investigation" [5].

To highlight works in this direction, we mention here a few examples as surveyed in [3] that are directly related to our methodology of modeling. Pollitt [3] identified four steps in digital forensics, as follows: acquisition, identification, evaluation, and admission of evidence, and further described admissibility of evidence. Carrier [13] outlined layers of abstraction of forensic examination where each layer has two inputs and two outputs. The National Institute of Standards and Technology (NIST) [14] defines the basic forensic process as follows:

> **Collection**. The first phase in the process is to identify, label, record, and acquire data from the possible sources of relevant data...
>
> **Examination**. Examinations involve forensically processing large amounts of collected data using a combination of automated and manual methods to assess and extract data of particular interest, ...
>
> **Analysis**. The next phase of the process is to analyze the results of the examination, using legally justifiable methods and techniques, to derive useful information that addresses the questions that were the impetus for performing the collection and examination.
>
> **Reporting**. The final phase is reporting the results of the analysis, which may include describing the actions used, explaining how tools and procedures were selected, determining what other actions need to be performed …, and providing recommendations for improvement to policies, guidelines, procedures, tools, and other aspects of the forensic process.

Other proposed models include those described by Reith, Carr, and Gunsch [15], Carrier and Spafford [16], Ademu, Imafidon, and Prestonn [22], and Ciardhuain [17].

To keep this paper self-contained, the next section reviews basic features of the model on which our methodology for describing forensics investigation is built.

3. Flowthing Model

The Flowthing Model (FM) was inspired by the many types of flows that exist in diverse fields, such as, for example, supply chain flow, money flow, and data flow in communication models. This model is a diagrammatic schema that uses flowthings to represent a range of items that can be data, information, or signals. FM also provides the modeler the freedom to draw the system using flowsystems that include six stages, as follows:

- Arrive: a flowthing reaches a new flowsystem (e.g., a buffer in a router)

- Accepted: a flowthing is permitted to enter the system (e.g., no wrong address for a delivery); if arriving flowthings are also always accepted, Arrive and Accept can be combined as a Received stage.
- Processed (changed): the flowthing goes into some kind of transformation that changes its form but not its identity (e.g., compressed, colored, etc.)
- Released: a flowthing is marked as ready to be transferred (e.g., airline passengers waiting to board)
- Created: a new flowthing is born (created) in the system (a data mining program generates the conclusion *Application is rejected* for input data)
- Transferred: the flowthing is transported somewhere outside the flowsystem (e.g., packets reaching ports in a router, but still not in the arrival buffer).

These stages are mutually exclusive, i.e., a flowthing in the process stage cannot be in the created stage or the released stage at the same time. An additional stage of Storage can also be added to any FM model to represent the storage of flowthings; however, storage is not a generic stage because there can be stored processed flowthings, stored created flowthings, etc. Figure 1 shows the structure of a flowsystem. A *flowthing* is a thing that has the capability of being created, released, transferred, arrived, accepted, or processed while flowing within and between systems. A *flowsystem* depicts the internal flows of a system with the six stages and transactions among them. FM also uses the following notions:

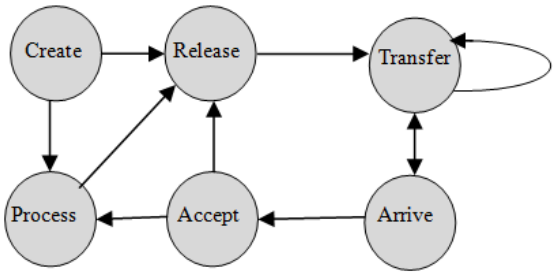


Figure 1. Flowsystem, Assuming that no Released Flowthing is Returned

Spheres and subspheres: These are the environments of the flowthing, such as a company and the departments within it, an instrument, a computer, an embedded system, a component, and so forth. A sphere can have multiple flowsystems in its construction if needed.

Triggering: Triggering is a transformation (denoted in FM diagrams by a dashed arrow) from one flow to another, e.g., flow of electricity triggers the flow of air.

A flowsystem may not need to include all the stages; for example, an archiving system might use only the stages Arrive, Accept, and Release. Multiple systems captured by FM can interact with each other by triggering events related to one another in their spheres and stages.

Example: In a tutorial about computer forensics, Figure 2 is introduced as a generic process that applies in computer forensics [18]. To illustrate the FM description, and because a more complete representation will be discussed in the next section, let us focus on the first two phases in that figure: collecting evidence, and analyzing it. Figure 3 depicts the FM representation of this example. There are two spheres (corresponding to roles): that of the Collector and that of the Analyst.

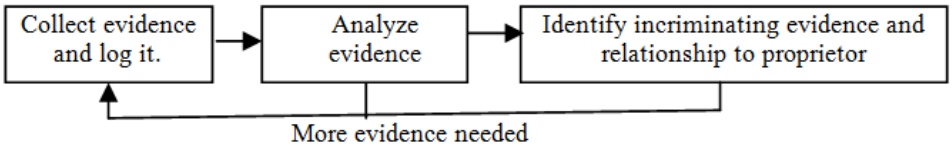


Figure 2. Forensics Process (From [18])

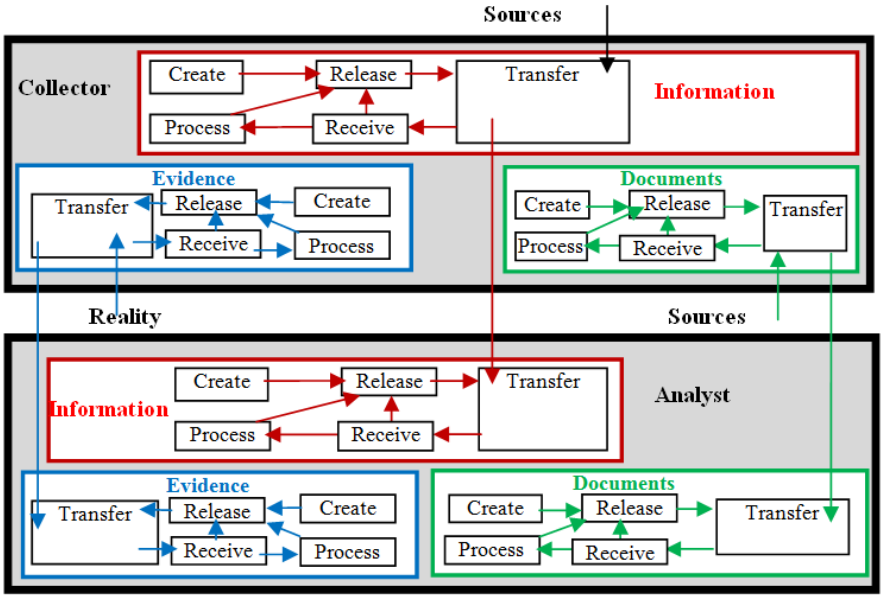


Figure 3. FM-based Description of the Collection and Analyzing Phases

In Figure 3, the Collector's sphere includes three flowsystems: information (e.g., from people), evidence, and documents (e.g., legal permissions). The Analyst's sphere includes similar flowsystems. For purposes of simplification, indicators of storage are not included in the figure. Note that the conceptual picture covers both the physical and the electronic environments. Suppose that the collector's sphere includes both physical evidence (hard copies) and electronic evidence (e.g., a computer file). Then the evidence flowsystem in the Collector sphere can represent an electronic flowsystem of e-evidence and records of physical evidence. As we will see later, when a collector triggers the system to create a new item of evidence (circle 1 in the figure), then he/she is given the choice of e-evidence (e.g., attaching a

file), or physical evidence, where in the latter case a record of information is kept about this evidence.

In Figure 3, information, evidence, and documents are also in the Analyst's sphere, who may or may not be the same person. Note that the flow of flowthings from the Collector to the Analyst is a logical flow. This means that the Collector and the Analyst can work in parallel, i.e., the analyst analyzes while the collector collects the flowthings. It is possible that processing information, evidence, and documents triggers creating "knowledge", as shown in Figure 4.

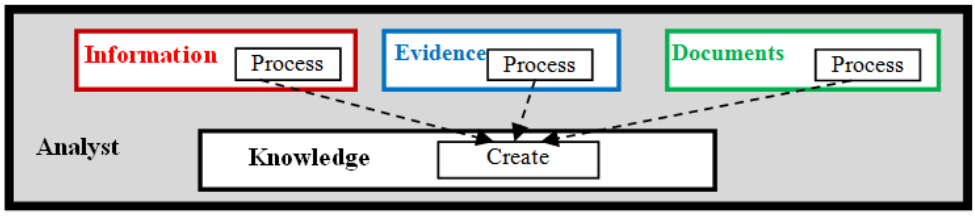


Figure 4. Triggering Mechanism in FM

4. FM-based representation of the forensics process

Köhn, Eloff, and Olivier [19] argue that Digital Forensic Process Models (DFPMs) in particular and the field of digital forensic investigation in general can benefit from the introduction of a formal modeling approach. They propose UML as a suitable vehicle for this purpose. In their paper they utilized UML activity with case diagrams and applied them to a digital forensic process model published by the US Department of Justice (USDOJ) [21].

The USDOJ model comprises four phases (Figure 5), namely collection, examination, analysis, and report. The collection phase involves searching for evidence. The examination phase aims to reveal any hidden or obscure data. The third phase involves analysis to determine probative value. The outcome of this phase produces evidence that can be used in court. The report phase results in a report presented in court about the process followed during the investigation. Figure 6 shows the Use Case and activity diagram of the US Department of Justice model.

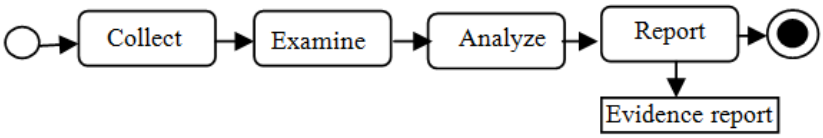


Figure 5. The US Department of Justice Model of Four Phases (from [19])

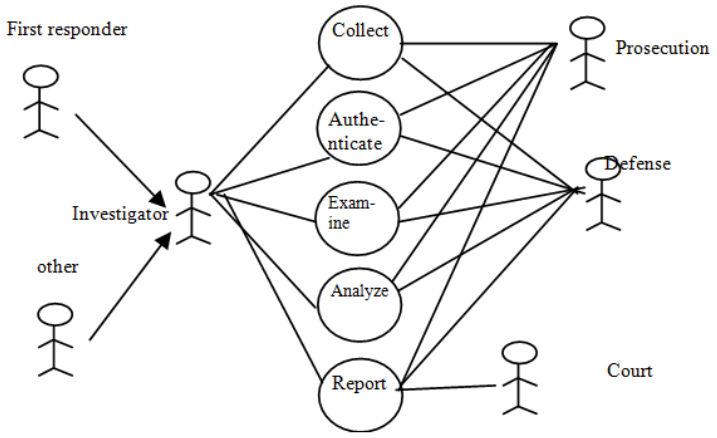


Figure 6. Use Case (from [19])

Köhn, Eloff, and Olivier [19] then propose an expanded model that includes elements from a model given by Kruse and Heiser [20] (Figure 6), superimposed on a framework proposed by Köhn, Eloff, and Olivier [6]. This framework has three phases: Preparation, Investigation, and Presentation. Figure 6 shows the resulting expanded activity diagram.

Still these types of diagrams do not focus on the central operation of investigation. Apparently, the arrows in the activity diagram denote a sequence of activities; nevertheless, these activities are intertwined and the issue is not just a "return statement" as in flowcharts. Collection of evidence overlaps with examination and analysis. The Activity diagram is a heterogeneous description that looks like a deformed flowchart. While Collect, Examine, and Analyze are processes, the diamond in Figure 7 is a decision notation from an ordinary flowchart. The dashed arrows seem to denote "compilation" of reports and evidence. The different roles of the use case have disappeared in the activity diagram; thus we have lost the semantics of who is doing what.

In FM, integration is accomplished in one diagram, as shown in Fig. 8, where the focus is on the investigation process. Four basic roles are included: those of Collector, Examiner, Analyst, and Presenter, which are modeled uniformly by their individual spheres and flowsystems.

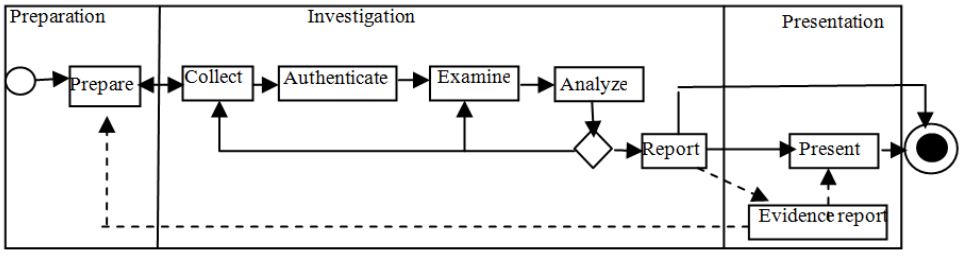


Figure 7. Activity Diagram (from [19])

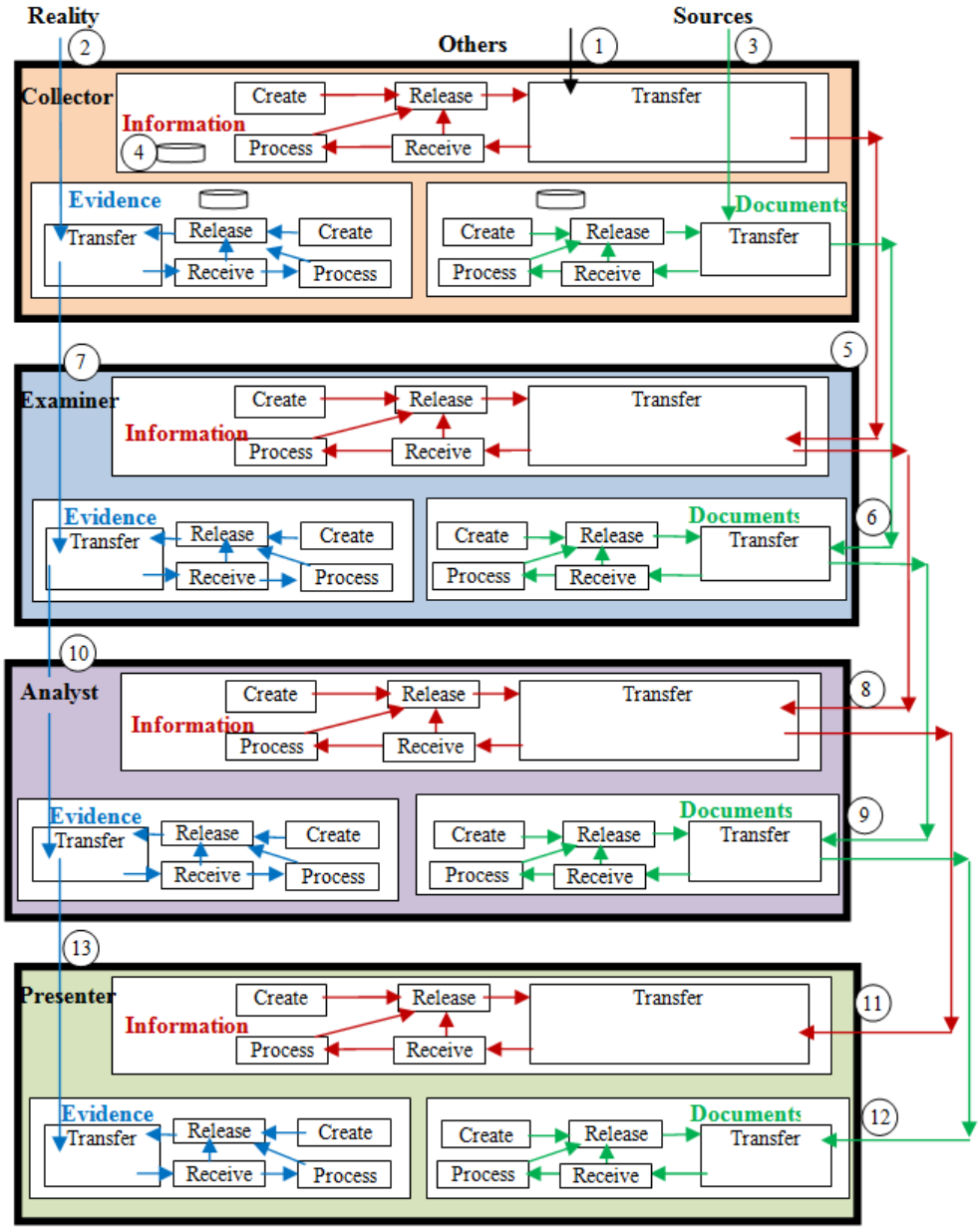


Figure 8. FM Integrated Model of Investigation

Without loss of generality, three flowthings are included: information, evidence, and documents. These flowthings flow to the Examiner's sphere from various sources (e.g., people, crime scene, other agencies – circles 1, 2, and 3 in Figure 8). The storage figure (4) indicates the possibility of storage of flowthings. For simplicity's sake, these storage notations are not included in the other spheres.

The Examiner's sphere receives information (5), documents (7), and evidence (6) from the Collector's sphere. This same type of relationship is applied to the relationships between the Examiner and Analyst spheres (circles 8, 9, and 10), and the Analyst and Presenter spheres (circles 11, 12, and 13). As indicated in the example in the previous section, the flows here

are logical flows among spheres. All roles may be performed in parallel, which implies the notion of "return to previous stage", as illustrated in Figure 9.

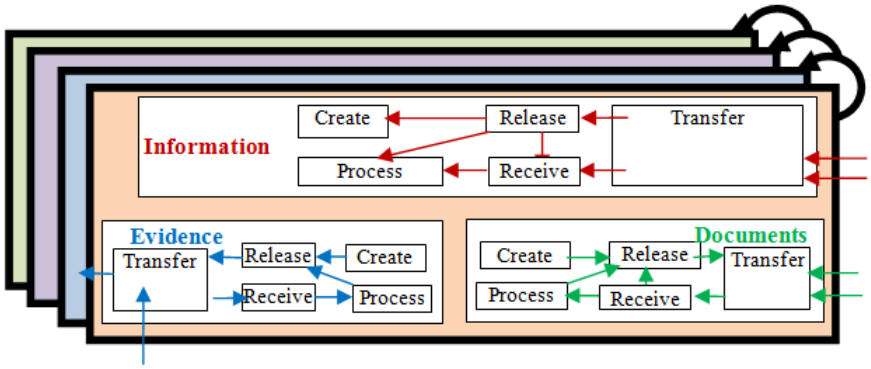


Figure 9. The FM Representation is Applied Uniformly with Possible Parallelism

Suppose that four different people actually take the roles of collector, examiner, analyst, and presenter. As soon as the collector acquires some information, the examiner can process it (e.g., authenticate it), and as soon as the information is examined, the analyst can analyze this partial information, and it is possible the presenter can work on ways to present it. The collector then continues adding information, giving the examiner more information to process, which provides more examined information for the analyst to analyze and more information for the presenter to work on. Such a picture contrasts with the serial consecutive stages of previous models discussed previously.

Different flowthings and their flowsystems can be inserted into the basic model of Figure 8. For example, Figure 10 shows the inclusion of a reports flowsystem in the presenter's sphere.

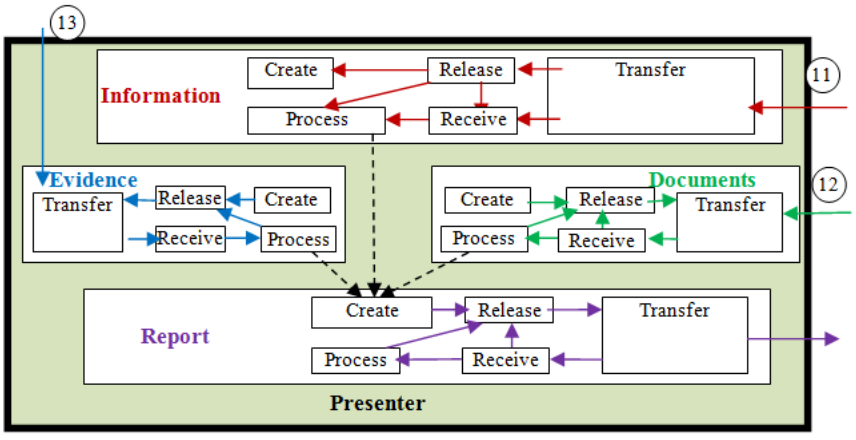


Figure 10. Reports Flowsystem in the Presenter Sphere

5. Sample Management System

The FM model can be a foundation in development of an automated system that supports the forensics process. The interesting aspect of such a system is that it can be built on the same elements and operations as the theoretical framework.

In the Forensic Information Management System of West Virginia [23], the software streamlines information gathering and integrates external and internal data collection processes. It integrates information on different platforms, shares information, tracks cases, and uploads data directly into a database. Figure 11 shows its overall architecture and a sample system screen associated with case reporting. It reflects a typical system where the description of real workflows disappears in the automated system's description. System builders draw the architecture according to automated system modules, and screens are designed by agreement between builders and users. The forensics processes that were modeled in the requirements phase have no resemblance to the screens, and the operation has no theoretical foundation.

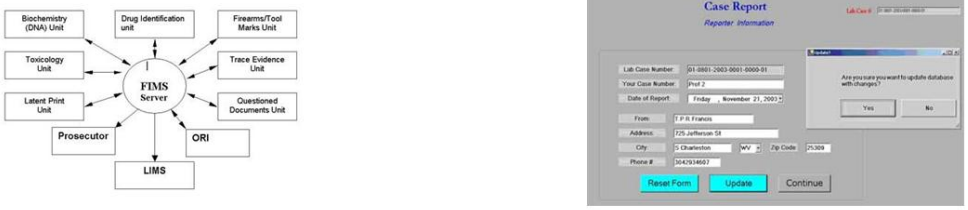


Figure 11. FBI Forensic Information Management System of West Virginia: Overall Architecture and a Sample System Screen [23]

By contrast, an FM-based system is applied uniformly from the requirements phase to its implementation. Its overall architecture is an FM-integrated model (e.g., Figure 7), built upon flowsystems.

Its main menu screen is the flowsystem of Figure 1 used to describe the flows in the requirement stages, as shown in Figure 12. This main screen for evidence is essentially the flowsystem shown in Figure 1. It has the six main stages for all users. The investigator, analyst, manager, and so forth have the same main screen that reflects the theoretical foundation used in drawing the conceptual map.

Figure 12. The Main Screen in an FM-based system

Suppose that an investigator opens her account. She has the choice of selecting cases, evidence, and documents with menus that include created, processed, released, transferred,

arrived, or accepted flowthings. Suppose that the investigator works on evidence. She can list all current evidence on her "virtual" desk (or within a certain case, if desired), search for certain evidence (e.g., when not sure at what stage the evidence is), or list evidence in storage, as will be discussed later.

Suppose the investigator selects "create" in the main menu. The screen shown in Figure 13 then appears. The current pieces of evidence in the creation stage are classified in different categories (e.g., policies, networks), and the investigator can select to create a new case or list existing cases.

Similar to cases, we have a main menu for evidence. Suppose that the investigator selects to create a new case; the screen shown in Figure 14 appears. The investigator has the choice of either adding an evidence description (e-file), or creating a record about a piece of evidence (e.g., hair) that is actually in the laboratory. We can see here that FM can manage electronic evidence and also record information about physical evidence. Also, in Figure 14, the investigator can move a piece of evidence from the creation stage to the process stage (e.g., authentication, signing, etc.). If she moves the evidence to the process stage, the evidence is no longer in the creation stage, but in the processed stage. It can be found listed when the processing stage menu is accessed. Here the investigator can select from a set of standard processes, as shown in Figure 15.

While we are not introducing a complete description of the FM-based forensics management system, the set of screen shots we show demonstrate how the methodology is systematic and uniform from the phase of collecting data and operations to the phases of implementation and use of the system. Contrast this with a systems specification using, say, UML activity and sequence diagrams, and the users screens after implementing the system.

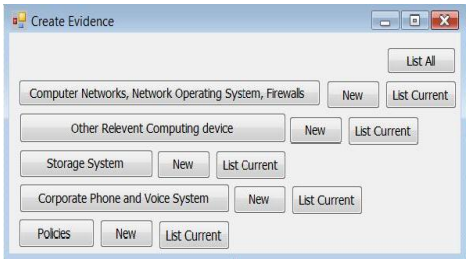


Figure 13. Create Stage

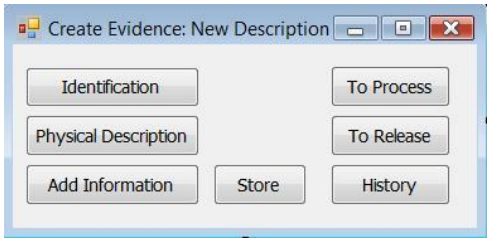


Figure 14. Create New Evidence

Figure 15. Process Stage

6. Conclusion

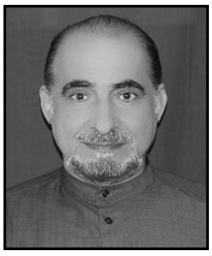
This paper proposes an abstract model of the digital forensic procedure based on a new flow-based specification methodology. It is shown through examples that the method can uniformly specify the forensic process in various phases and across roles. It also provides a more exact description where "things" (e.g., information, evidence) are separated into different streams of flow with six generic internal operations: create, release, transfer, arrive, accept, and process. Further research aims at experimenting with the method in real environments to build an information system to support stages of investigation.

References

[1] S. H. Van Solms and C. P. Lourens, "A Control Framework for Digital Forensics", IFIP 11.9, **(2006)**.
[2] ISO/IEC Information Technology, "Security techniques: codes of practice for information security management", International Organisation for Standardization and the International Electrotechnical Commission. ISO/IEC 17799, **(2005)**.
[3] M. Pollitt, "Computer forensics: an approach to evidence in cyberspace", in Proceedings of the National Information Systems Security Conference, Baltimore, MD, vol. 2, **(1995)**, pp. 487-491, http://www.digitalevidencepro.com/Resources/Approach.pdf.
[4] A. Agrawal, M. Gupta, S. Gupta and C. Gupta, "Systematic digital forensic investigation model", International Journal of Computer Science Security, vol. 5, no. 1, **(2011)**, http://www.cscjournals.org/csc/manuscript/Journals/IJCSS/volume5/Issue1/IJCSS-438.pdf .
[5] A. Tanner and D. Dampier, "An Approach for Managing Knowledge in Digital Forensics Examinations", Int. J. Comput. Sci. Secur., vol. 4, no. 5, **(2010**).
[6] M. Köhn, J. H. P. Eloff and M. S.Olivier, "Framework for a digital forensic investigation", in Proceedings of the ISSA 2006 From Insight to Foresight Conference, Sandton, South Africa, July 2006, Edited by H. S. Venter, J. H. P. Eloff, L. Labuschagne, and M. M. Eloff, **(2006)**.
[7] S. Al-Fedaghi, "Awareness Of Context and Privacy", Am. Soc. Information Sci. Tech. Bull., vol. 38, no. 2, **(2011)**.
[8] S. Al-Fedaghi, "Conceptual Foundation for Specifying Processes", Int. J. Adv. Comput. Tech., vol. 3, no. 4, **(2011)**.
[9] S. Al-Fedaghi, "Information security management systems", Fifth Annual IFIP WG 11.10 International Conference on Critical Infrastructure Protection, Dartmouth College, Hanover, New Hampshire, USA, **(2011)** March 23–25.
[10] S. Al-Fedaghi, "Software requirements as narratives", Third International Conference on Information, Process, and Knowledge Management, Gosier, Guadeloupe, France, **(2011)** February 23-28.
[11] S. Al-Fedaghi, "System-based approach to software vulnerability", IEEE Symposium on Privacy and Security Applications (PSA-10), Minneapolis, USA, **(2010)**.
[12] Digital Forensic Research Workshop (DFRWS), Research Road Map, Utica, NY, **(2001)**, http://www.dfrws.org/2001/dfrws-rm-final.pdf.
[13] B. Carrier, "Defining digital forensic examination and analysis tools using abstraction layers", Int. J. Digital Evidence, vol. 1, no. 4, **(2003)**, http://www.utica.edu/academic/institutes/ecii/publications/articles/A04C3F91-AFBB-FC13-4A2E0F13203BA980.pdf.
[14] K. Kent, S. Chevalier, T. Grance and H. Dang, "Guide to Integrating Forensics into Incident Response", Special Publication 800-86, Computer Security Division Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, MD, **(2006)**, http://csrc.nist.gov/publications/nistpubs/800-86/SP800-86.pdf.
[15] M. Reith, C. Carr and G. Gunsch, "An Examination of Digital Forensic Models", Int. J. Digital Evidence, vol. 1, no. 3, **(2002)**.
[16] B. Carrier and E. H Spafford, "Getting Physical with the Investigation Process", Int. J. Digital Evidence, vol. 2, no. 2, **(2003).**
[17] S. O. Ciardhuain, "An extended model of cybercrime investigations", Int. J. Digital Evidence, vol. 3, no. 1, **(2004)**.
[18] Information Security Site, Tutorial: Computer Forensics Process for Beginners, http://www.shortinfosec.net/2008/07/tutorial-computer-forensics-process-for.html.

[19] M. Kõhn, J. H. P. Eloff and M. S. Olivier, "UML modelling of digital forensic process models (DFPMs)", in Proceedings of the ISSA Innovative Minds Conference, Pretoria, South Africa, Edited by H. S. Venter, M. M. Eloff, J. H. P. Eloff, and L. Labuschagne, **(2008),** pp. 1-13, http://mo.co.za/open/umldfpms.pdf.
[20] W. Kruse and J. Heiser, "Computer Forensics: Incident Response Essentials", Addison Wesley, **(2002)**.
[21] USDOJ. Technical Working Group for Electronic Crime Scene Investigation, Electronic Crime Scene Investigation: A Guide for First Responders, United States Department of Justice **(2001)**, https://www.ncjrs.gov/pdffiles1/nij/219941.pdf.
[22] I. O. Ademu, C. O. Imafidon and D. S. Preston, "New Approach of Digital Forensic Model for Digital Forensic Investigation", Int. J. Adv. Comput. Sci. Appl., vol. 2, no. 12, **(2011)**.
[23] R. S. Ahluwalia and A. Srinivasan, "Forensic Information Management System", Forensic Science Foundation, vol. 6, no. 2, **(2004)**, http://www.fbi.gov/about-us/lab/forensic-science-communications/fsc/april2004/research/2004_02_research01.htm.

Authors



Sabah Al-Fedaghi holds an MS and a PhD in computer science from the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, Illinois, and a BS in Engineering Science from Arizona State University, Tempe. He has published two books and over 140 papers in journals and conferences on Software Engineering, Database Systems, Information Systems, Computer/information Privacy, Security and assurance, Information Warfare, and Conceptual Modeling. He is an associate professor in the Computer Engineering Department, Kuwait University. He previously worked as a programmer at the Kuwait Oil Company and headed the Electrical and Computer Engineering Department (1991–1994) and the Computer Engineering Department (2000–2007).

Bashayer Al-Babtain is a graduate student in the Computer Engineering Department at Kuwait University. Her research interest includes Access Control, data mining, SVM and Flow Model.