Requirements-driven adaptive digital forensics

Conference Item

Requirements-Driven Adaptive Digital Forensics

Liliana Pasquale*, Yijun Yu†, Mazeiar Salehie*, Luca Cavallaro*, Thein Than Tun†, and Bashar Nuseibeh*†
* Lero - the Irish Software Engineering Research Centre, Limerick, Ireland
{liliana.pasquale | mazeiar.salehie | luca.cavallaro}@lero.ie
† Department of computing, Open University, Milton Keynes, UK
{y.yu | thein.tun | b.nuseibeh}@open.ac.uk

***Abstract*—A digital forensics process aims to collect and analyze the evidence essential to demonstrate a potential hypothesis of a crime. We propose the use of forensic requirements to automate a digital forensics process. We augment traditional reactive digital forensics processes - used to perform an investigation - with proactive evidence collection and analysis activities, which provide immediate investigative suggestions before an investigation starts. These activities dynamically adapt depending on suspicious events, which in turn might require the collection and analysis of additional evidence. The reactive activities of a traditional digital forensics process are also adapted depending on the current investigation findings.**

I. Introduction

Digital forensics [1] aims to collect and analyze necessary digital evidence to demonstrate how a computer crime was committed, what harm was done, and who was responsible. A digital forensics investigation thus collects and analyzes the evidence essential to demonstrate a potential hypothesis of crime. It also includes a presentation activity to illustrate the investigation findings (proved/refuted hypothesis). While several research and commercial tools, such as EnCase [2], are available to automate evidence collection, investigations are still highly human-intensive. Investigators usually approach each crime case from scratch, by postulating potential hypotheses. Existing tools do not provide any investigative direction to suggest what are the possible hypotheses, including the evidence they require to be demonstrated, and their likelihood of being true. Since the findings of a digital investigation should be based on objective evidence, a digital forensics process should use well founded and systematic techniques to help investigators assess the likelihood of each hypothesis and provide sound evidence in court.

To address these shortcomings, we augment digital forensics processes with proactive analysis and collection activities. These activities preserve and analyze important evidence before an investigation starts. The outcome of such activities is then used to provide immediate suggestions regarding what hypotheses should be investigated (because they are more likely to be true) and what evidence should be collected to completely prove/refute them. To preserve important evidence, proactive analysis must identify suspicious events that require to adapt the proactive collection activities in order to gather additional evidence. The reactive activities of a traditional digital forensics process may also adapt depending on the current investigation findings.

This paper applies requirements engineering techniques to configure the behavior of an adaptive digital forensics process. We propose to model the *forensics requirements* to capture the crime scene and the potential hypotheses of a crime. We use structured arguments, *forensics arguments*, to represent the hypotheses of crime. Each hypothesis is a claim that is related to a set of facts necessary to prove or refute it. The facts represent the evidence to be collected from the digital devices (evidence sources) available at the crime scene. We formally express forensics arguments in the Event Calculus [3] and support the formal verification of hypotheses depending on the percentage of facts that have been demonstrated (i.e., ampliative probability [4]). Forensics arguments are also used to express conditions that may start/stop the full evidence collection performed proactively. Thus, forensics requirements are used to instrument the proactive and reactive activities of a digital forensics process.

II. Adaptive Digital Forensics Process

As shown in Figure 1, our approach comprises eight steps. **1) Requirements Modeling:** a security administrator designs the forensics requirements. These include a domain model of a crime scene, which represents the assets that can be harmed, the topology of the physical space where a crime can be committed, the configuration of the digital devices available, users roles and permissions. For example, the crime scene may indicate that *a valuable document (Doc) is stored on a machine (M1) located in an office (T225), and only authorized employees (Alice and Bob) can access this office, by swiping their badge on a NFC reader (NFC). A CCTV monitors the entrance and exit to/from T225*. The security administrator also designs the forensics arguments. These may represent suspicious events conditions that must hold to start and stop the full evidence collection performed proactively (starts/stop arguments). These also represent the hypothesis of the potential crimes that can be committed in the crime scene (reactive arguments). These arguments are initially expressed in a generic form and are subsequently customized depending on the potential offenders and the devices modeled in the crime scene. For example, the potential hypothesis of a crime can state that *at least one user is in T225, one of the employee accesses the Doc in M1 while his/her USB pen is mounted.* The start and stop argument can respectively express the conditions that signal that *an employee is logged on M1 and accesses the Doc* and *an employee is not logged anymore on M1*.

2) Configuration: a Requirements Manager uses the forensics requirements to configure the proactive and reactive activities of the digital forensics process. It uses the start and stop arguments to configure the Proactive Analysis. It also leverages the data necessary to check the start and stop arguments to configure the Proactive Collection, such as *accesses to T225 (from the log of the NFC), logins on M1 and accesses to Doc*

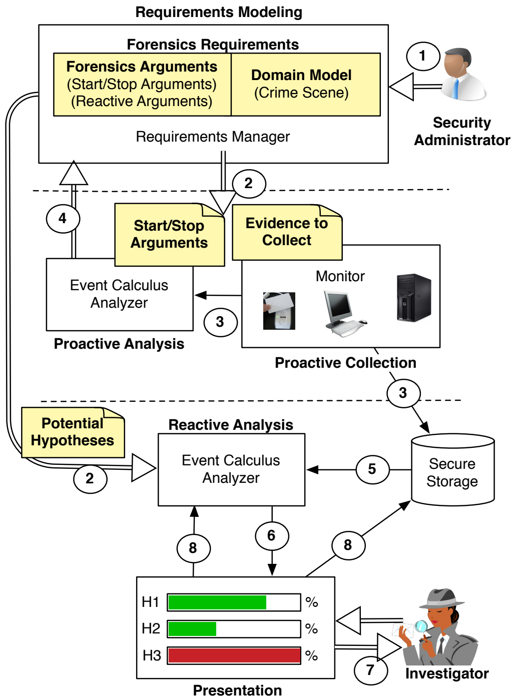

Fig. 1. Adaptive Digital Forensics Process.

(from the system and registry log of M1). The Requirements Manager uses the reactive arguments to generate all possible hypotheses of a crime, which are given in input to the Reactive Analysis. In this case, 12 hypotheses must be checked. These are obtained by customizing the reactive argument depending on *who can be in T225 (Alice only, Bob only, or Alice and Bob together), who can log on M1 (Alice or Bob), and who can own a USB pen that is mounted on M1 (Alice or Bob)*.

3) Proactive Collection: during the normal system functioning, the Monitor collects the data identified during the previous step, sends them to the Event Calculus Analyzer (Analyzer) and stores them securely.

4) Proactive Analysis: every time new evidence is available, the Analyzer checks whether the conditions to start/stop the full evidence collection are satisfied and sends the results to the Requirements Manager. To perform this task the Analyzer uses the deductive reasoning functionality of event calculus. In case a start argument is satisfied for a set of specific elements of the crime scene, the Requirements Manager reconfigures the Proactive Analysis in order to check whether the corresponding stop argument is satisfied for the same elements of the crime scene. The Proactive Collection is reconfigured to gather all possible evidence. For example, *if Bob is logged on M1 and accesses the Doc, the stop argument that should be checked by the Proactive Analysis claims that Bob is not logged anymore on M1. While the Proactive Collection gathers additional events, such as when devices are mounted, unmounted, or installed on M1*. When the full evidence collection is switched on and the stop argument is satisfied, the proactive collection and analysis activities are (re-)configured as in step 2.

5) Investigation Set-up: when an investigation starts, the Analyzer retrieves the data collected by the Proactive Analysis from the secure storage.

6) Reactive Analysis: the Analyzer evaluates the satisfaction of each hypothesis and sends the results to the Presentation activity. The Analyzer uses the abductive reasoning functionality of the Event Calculus. For each hypothesis that can still be satisfied, the Analyzer generates a set of potential events that represent the missing evidence necessary to satisfy this hypothesis. For example, *when an investigation starts, we can assume that the sequence of events retrieved from the Secure Storage state that Bob was logged on M1 and accessed the Doc while a USB was mounted. In this case, the Analyzer will discover that only 6 of the original hypotheses are satisfiable (the ones that state that Bob logged on M1)*.

7) Presentation: this activity shows the probability of satisfaction of each hypothesis. The investigator selects the hypotheses s/he wants to focus on and receives indications regarding the remaining evidence to be collected. For example, *the investigator is suggested to collect additional evidence from a CCTV to confirm that Bob is in T225 when he logged on M1, and to verify whether Bob owns the USB pen*.

8) Reactive Collection: the investigator retrieves the remaining evidence, by using, for example, existing commercial tools, stores it securely and sends it to the Analyzer that updates the satisfaction of the hypothesis. The cycle (5-8) continues until the investigator identifies a set of hypotheses that can be presented in court.

III. CONCLUSIONS

Although several research approaches have been proposed to collect forensically sound evidence, only a few work use formal techniques to tackle the automatic analysis of the acquired evidence. Some existing approaches use finite state machines [5], and event calculus [6] for events reconstruction. However, none of them suggests how to automate and adapt the whole digital investigation. One approach [7] integrates proactive collection and analysis of digital evidence with reactive digital forensics processes. However, it does not provide detail on how a digital forensics process should be configured. Furthermore, it does not specify how proactive and reactive activities can be coordinated, for example, how proactive digital evidence can be used during an investigation. It is our belief that providing a requirements-driven approach can facilitate a digital investigation and shorten the cycles for the events reconstruction.

REFERENCES

[1] G. Palmer, “A Road Map for Digital Forensic Research,” Air Force Research Lab, Rome, DFRWS Technical Report DTR-T001-01, 2001.

[2] Guidance Software, “EnCase Forensics - Computer Forensics Data Collection for Digital Evidence Examiners,” http://www.guidancesoftware.com/encase-forensic.htm, Accessed on Feb 2013.

[3] E. T. Mueller, *Commonsense Reasoning*. Morgan Kaufmann, 2006.

[4] J. B. Freeman, “Argument Strength, the Toulmin Model, and Ampliative Probability,” *Informal Logic*, vol. 26, no. 1, pp. 25–40, 2008.

[5] P. Gladyshev and A. Patel, “Finite State Machine Approach to Digital Event Reconstruction,” *Digital Investigation*, vol. 1, no. 2, 2004.

[6] S. Y. Willassen, “Using Simplified Event Calculus in Digital Investigation,” in *Proc. of the Symp. on Applied Computing*, 2008, pp. 1438–1442.

[7] T. Grobler, C. P. Louwrens, and S. H. von Solms, “A Framework to Guide the Implementation of Proactive Digital Forensics in Organisations,” in *Proc. of the 5th Int. Conf. on Availability, Reliability and Security*, 2010, pp. 677–682.