Towards Forensic-Ready Software Systems

Liliana Pasquale
University College Dublin, Ireland

Dalal Alrajeh
Imperial College London, UK

Claudia Peersman
University of Bristol, UK

Thein Tun
The Open University, UK

Bashar Nuseibeh
The Open University, UK & Lero, Ireland

Awais Rashid
University of Bristol, UK

ABSTRACT

As software becomes more ubiquitous, and the risk of cyber-crimes increases, ensuring that software systems are forensic-ready (i.e., capable of supporting potential digital investigations) is critical. However, little or no attention has been given to how well-suited existing software engineering methodologies and practices are for the systematic development of such systems. In this paper, we consider the meaning of forensic readiness of software, define forensic readiness requirements, and highlight some of the open software engineering challenges in the face of forensic readiness. We use a real software system developed to investigate online sharing of child abuse media to illustrate the presented concepts.

ACM Reference Format:
Liliana Pasquale, Dalal Alrajeh, Claudia Peersman, Thein Tun, Bashar Nuseibeh, and Awais Rashid. 2018. Towards Forensic-Ready Software Systems. In *ICSE-NIER'18: 40th International Conference on Software Engineering: New Ideas and Emerging Results Track, May 27-June 3, 2018, Gothenburg, Sweden.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3183399.3183426

1 INTRODUCTION

Forensic readiness represents the capability of an organization to support digital investigations proactively, i.e., before an incident occurs [24]. It is realized through the production of evidence that (*i*) facilitates the investigation and demonstration of compliance to organizational and regulatory policies, and (*ii*) can support legal proceedings [9]. To date, however, researchers' attention has been geared towards the provision of general guidelines that could potentially enhance organizations' operational and infrastructural capabilities to achieve forensic readiness. Little or no attention has been given to how the software systems deployed within these organizations can be designed to be themselves forensic-ready, i.e. forensic-by-design [**?**]. With a rapid rise in cybercrime and cyber-enabled crime—number of identity theft incidents increased by 222% in 2016 [7] whilst online child sexual exploitation increased by 135% in 2016 [21]—there is an urgent need to consider what forensic readiness means for software systems and how such readiness can be incorporated as part of software development processes. Work in this area is either very preliminary or has been limited to specific aspects of forensic readiness, such as ensuring that evidence relevant to potential incidents is preserved [3, 17, 19] or that evidence integrity is maintained [16]. No work to date has considered the wider potential or implication of rigorous software engineering on the development of forensic-ready systems.

Our vision is to investigate the notion of forensic readiness in software systems and understand how forensic-ready software systems can be developed systematically. To achieve this vision we investigate forensic readiness requirements over software systems and assumptions over their encompassing environment. Requirements and assumptions can be used to derive implementable software specifications that achieve forensic readiness. Some of these requirements are data-centred, aimed to ensure availability, relevance, minimality, non-repudiation, completeness, and linkability of data. Others are process-centred, aimed to ensure that the process through which the software system performs digital forensic activities is sound, e.g., supports data provenance and legal compliance. We elicit forensic readiness requirements by reviewing existing literature and examining a real world investigative toolkit, iCOP [20], which was designed with the purpose of facilitating investigations of online child abuse media shared throguh P2P networks. Finally, we present open research challenges that relate to different aspects of engineering forensic-ready software systems and that consider how such systems can operate within emerging cyber-physical environments.

2 SOFTWARE FORENSIC READINESS

Although forensic readiness is a notion that is not new in the context of digital forensics, what it means and how it is conceptualised, differs amongst researchers, e.g., [11, 22, 24, 27]. In this paper, we are interested in what forensic readiness means for software engineering practices. At the heart of digital forensic readiness is the digital data, including the media and the activity logs available within an organization's information system network, or on users' devices. These data could hold valuable information about how a particular incident occurred and by whom, potentially resulting in a successful prosecution of the perpetrator [5, 13]. In this context, forensic readiness of system network or devices implies maximal usefulness of the data held as potential digital evidence admissible in court. Such usefulness can only be attained if the data and the process through which they are acquired, analysed and stored is *forensically sound*. (A forensically sound process is one that maximizes the evidentiary weight of digital evidence [18], whilst forensically sound evidence is one that can endure legal scrutiny in a court of law [10].) We take the view that forensic readiness in the context of software engineering is a *property that encapsulates the capabilities of software to:* (1) *conduct digital forensic processes in a*

ICSE-NIER'18, May 27-June 3, 2018, Gothenburg, Sweden

ACM ISBN 978-1-4503-5662-6/18/05…$15.00
https://doi.org/10.1145/3183399.3183426

forensically sound way; and (2) *produce forensically sound evidence.* As we are concerned with developing software that ensures forensic soundness of data and processes, *prior* to the occurrence of an incident or attack, such capabilities must be *proactive*. Furthermore, we consider forensic readiness to be a property that is achievable either partially or fully, depending on the software capabilities, and is with respect to a set of speculated incidents that an organization has identified and assessed as critical.

3 MOTIVATING EXAMPLE

To provide a sense of our envisaged research direction and challenges to overcome, we discuss the iCOP toolkit [20]. This has the main purpose of identifying, preserving and analysing new or previously unknown child sexual abuse (CSA) media shared by suspects on peer-to-peer (P2P) networks. As shown in Figure 1, iCOP has two major components: the *P2P Engine* and the *Analysis Engine*. The P2P engine monitors information (e.g., IP addresses, filenames and hash values of files) together with metadata (e.g., when a user was last seen sharing a file) from public traffic on P2P networks. This information is passed on to the Analysis Engine, which compares the monitored hash values to a list of known hashes of CSA media seized by law enforcement. This allows the system to disregard CSA media already known to law enforcement. The file names that do not occur in the known hash lists are then analysed to assess their likelihood of containing CSA media. Filenames flagged as suspicious are passed back to the P2P Engine for downloading. The content of the downloaded files is subsequently analysed automatically by a *Media Analysis* module to determine whether the files contain child abuse material. Finally, the resulting list of suspicious new or previously unknown files is examined by investigators to confirm whether they contain child abuse images and videos. Once confirmed, these items are fed back into the hash database as known CSA media in future searches.

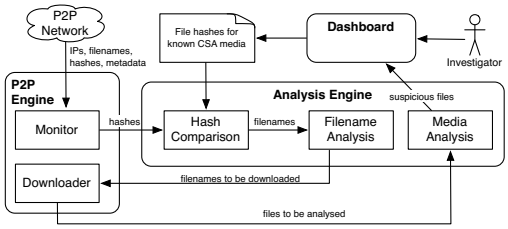

Figure 1: Overview of the iCOP toolkit

We aim to explore new software engineering methodologies and techniques for the design of software systems capable of supporting digital investigations proactively. In what follows, we consider the key requirements that software systems must satisfy and illustrate such requirements using the iCOP toolkit.

4 FORENSIC READINESS REQUIREMENTS

In this section, we describe a preliminary set of requirements for forensic-ready systems that were elicited by reviewing existing literature on forensic readiness. We distinguish between requirements that are data-centred and others that are process-centred.

Availability. Data that may be useful for investigating potential incidents should be available [24, 27]. To achieve availability, data that may provide investigative clues must be *preserved* and *retrievable* by law enforcement agencies or individuals who are in charge of conducting an investigation. As data may not be kept in non-volatile memories (e.g., network traffic) and physical devices can have limitations (e.g., damaged hard drives), the capability to preserve data proactively must be in place. Preservation can be triggered by changes in the data to be collected [25], can be performed periodically [30] or for a limited amount of time [14], in order not to consume resources (e.g., battery power in a mobile device). To facilitate retrieval of preserved data, metadata should also be stored. In the context of the iCOP toolkit, data that can be useful to investigate incidents can be video and image files indicating a new child abuse and information about the users sharing CSA media (e.g., IP address, client ID). However, not all files can be preserved successfully because downloads are often slow and they stall if the computers sharing the file go off-line. To facilitate retrieval of such data, CSA related material, P2P network users and victims should be identified unambiguously. This is challenging because often the same CSA material is shared under different filenames and victims can appear in different files. Moreover P2P network users cannot be uniquely identified because they can share a file from different locations and devices.

Relevance. Data preserved proactively should be relevant to potential incident cases [28]. Relevance of data means whether data is able to support or refute hypotheses explaining how incidents occurred [3]. Ensuring relevance of preserved data allows an organization to have the data preservation activities more targeted on the risks to the business [24]. Relevance can be subject to the judgement of an investigator [26] and is typically determined by the files and data types available [12, 23] (e.g., email addresses, message information, date and time information, cookies, social security and credit card numbers from a computer hard disk image). In our example, to satisfy the relevance requirement, iCOP should ensure that stored CSA media is of new material or previously unknown. This is challenging because analysing the content of any file shared on the P2P network is not computationally tractable. The iCOP toolkit uses textual features of the filenames and characteristics of the users sharing the files (e.g., users sharing the greatest number of suspected files or sharing the greatest number of files) to identify CSA related media. However, files that do not contain any textual clues to their illegal content -which otherwise is relevant data - or that are shared by new users may not be preserved.

Minimality. Data preserved proactively should be minimal and should not include any information that is unnecessary for the purpose of an investigation. Satisfaction of this goal can have the side-effect of reducing the amount of resources that are spent looking for digital evidence and, therefore, the costs of an investigation [27]. In our case study, to satisfy the minimality goal, the iCOP toolkit should not preserve files that do not refer to a child abuse case (false positives). A typical false positive error can be when webcam videos showing a child without any adult interaction are considered as CSA material. Satisfaction of the minimality goal highly depends on the type of data to be preserved; in our example, considering the difficulty of recognising CSA content, the minimality goal cannot be fully met.

Linkability. Preserved data should be linkable with other pieces of evidence, such as other evidence and witness statements. This is very important to reconstruct how an incident took place when heterogeneous data are preserved, as it allows creating cause-effect relations between incident activities indicated by different evidences [24]. In the iCOP toolkit, ensuring linkability between media sharing is important to identify an individual sharing specific content uniquely. A connection is assumed to be a single user sharing a given set of files from a specific location. Storing the IP and the geolocation information (GUID), an investigator can easily view which connections are related via a common IP address or GUID. Additionally, all files that are confirmed to contain CSA content can be used by police investigators to identify unknown victims.

Completeness. Preserved data should be sufficient to satisfy or refute an incident hypothesis. Satisfaction of this goal depends on the scope of an investigation, i.e. the portion of the environment in which an incident is assumed to have happened. For example, the scope of an investigation may be enlarged to include additional digital sources which can provide information about the location of new sources of evidence that may be relevant for the incident [15]. To satisfy the completeness goal, the iCOP toolkit should ensure preservation of any media related to child abuse that is shared on P2P networks. Currently the scope of the iCOP toolkit is limited to the Gnutella file sharing network and other P2P networks or social media are not considered. Achieving completeness requires making assumptions on the boundary of the investigation; this goal would be impossible to achieve if this boundary is not fixed.

Non-Repudiation. Preserved data should constitute an evidence that is admissible legally and should be accepted in a court of law [8]. To achieve this goal preserved data should satisfy the *integrity* requirement, i.e. they should not be tampered from the time of acquisition until its final disposition [24, 27]. Preserved data should provide high assurances about their *authenticity*; for example, only specific trusted parties should be authorised to access it [24]. The *chain of custody* of data should also be maintained [24]. This means that all changes in the control, handling, possession, ownership or custody of a piece of evidence should be documented. In our example, iCOP should ensure that preserved CSA content can only be accessed by police investigators in possession of login credentials. Moreover it should provide techniques to assess authenticity of media files and maintain their chain of custody. Additional tools and procedures to satisfy these requirements must be adopted by the individual law enforcement agencies.

Data provenance. The process adopted to preserve data should also record when, how and by whom such data is originated, moved and modified over time. The Transparent Computing[1] program encourages provenance of system components to identify relationships between system activities [6] that may be related to a cyber threat. As provenance information can grow over time it is also necessary to summarise such information meaningfully [2]. In our example, data provenance can refer to preservation of files metadata (e.g., creation date) that can provide more information about multiple abuses of the same victim over time. Data about the P2P users can also support identification of new users involved in sharing CSA material.

Legal compliance. The process adopted to preserve data should ensure compliance with existing regulations, which may vary depending on the jurisdiction(s) in which an incident may occur. Identification of what regulations apply to a specific system depends on the nationality and the physical location of the data subject, as well as the physical location of the organization collecting data. For example, the General Data Protection Regulation (GDPR)[2] in Europe and the Fourth Amendement in USA [1] regulate what data can be preserved and under which conditions (*privacy*). The EU Data Retention Directive, can prescribe for how long data should be retained (*retention*). The GDPR also prescribes for how long data be accessed and by whom (*access to retained data*). For the iCOP case study, media files can be preserved in UK because they are "voluntarily" shared to third parties but this would not apply in other European countries, such as Belgium. Any monitoring and downloading of CSA media can only take place at suitable law enforcement premises and access to CSA material is only given to police investigators, in order to ensure privacy of victims' identities.

5 SOFTWARE ENGINEERING CHALLENGES

We elicit a number of open software engineering challenges. These fall into four major categories: (*i*) Representation and reasoning, (*ii*) Methods, (*iii*) Verification and (*iv*) Technological developments.

1. Representing and reasoning about forensic-ready systems. We have presented a first conceptualization of forensic readiness requirements of software systems. There is a need to build a consensus around the key characteristics of forensic-ready software systems. We can divide the implications and challenges into three sub-categories: (*i*) concept (how to represent and reason about forensic-ready systems and their properties), (*ii*) method (how to design and implement forensic-ready systems), and (*iii*) tools (how to analyse and support the development of forensic-ready systems). Existing work on concepts and taxonomy of dependability [4] could be a useful reference model to extend to forensic readiness. Such characterization would facilitate a better understanding of the potential relationship between forensic-ready requirements and other types of requirements such as security, privacy and safety. Furthermore, there is a need to characterise forensic-ready systems formally. This requires identifying formal languages —if any— that are best suited to express forensic readiness requirements, and will allow us to understand the extent to which existing representation and reasoning techniques are applicable to forensic-ready systems.

2. Methods for engineering forensic-ready software systems. The notion of forensic readiness poses challenging questions for software engineering methods and particularly how should existing methods adapt to account for forensic readiness requirements. Research is needed to answer a number of fundamental questions related to how requirements for forensic-ready systems should be implemented and whether these requirements are solely about data preservation activities. Architectural patterns (similar to security patterns [29]) could also be investigated to design forensic-ready systems. Additional challenges relate to managing trade-offs between forensic readiness requirements such as privacy and availability given that some of these could manifest at runtime. For example, privacy may prevent the system from preserving relevant

[1] https://www.darpa.mil/program/transparent-computing

[2] Regulation (EU) 2016/679

evidence and depends on the country in which an investigation is conducted. Ensuring security of forensic-ready systems is also important. For example, analysis of media files in the iCOP toolkit should avoid malware execution and the criteria adopted by iCOP to decide when a file must be downloaded should be confidential.

3. Verification of forensic readiness requirements. A key challenge is verifying that existing software systems satisfy forensic readiness requirements. Research questions relate to whether these requirements require development of different verification techniques compared to those adopted to verify safety and security properties and whether satisfaction of forensic readiness requirements can be guaranteed at design time. Our analysis of the iCOP system demonstrates that satisfaction of forensic readiness requirements cannot be blanket and trade-offs arise from the interaction of forensic readiness requirements with properties of the environment or various (human or software) agents and investigative processes that interface with a software system. There are also interesting challenges with regards to impact on other functionality of the system, for instance, ensuring that runtime forensic processes are not intrusive and disruptive of normal system functions.

4. Technological developments. Perhaps the wicked problem posed for forensic readiness is the one arising from the increasing deployment of Internet of Things (IoT) devices—and the software embedded within these devices—in everyday settings. In such smart cyber-physical environments, the system design cannot be anticipated a-priori and is only emergent a-posteriori when various IoT devices dynamically compose to deliver various services. Even more critically this emergent design is volatile in that the system configuration—and the devices engaged—may change on a regular basis. An example of this is a user with wearables walking through a smart city environment with various devices coming in and out of range and interfacing with each other. Such a dynamically aggregated environment poses major challenges for forensic readiness of software systems – how are the goals of availability, relevance, non-repudiation, legal compliance, completeness, minimality, and linkability impacted in such a setting is a non-trivial question to be addressed by software engineering research.

6 CONCLUSION

In this paper we investigated the notion of forensic readiness in software systems and the requirements that support its attainment, highlighting some of the open software engineering challenges. For future work we plan to provide a formal characterization of forensic readiness requirements. We will also explore techniques for analyzing tradeoffs between conflicting requirements quantitatively. Finally we will investigate aspects related to the implementation of a forensic-ready system, such as the generation of specification for such systems or assessment of relevance of preserved data.

ACKNOWLEDGEMENTS

This work is supported by EPSRC Grant: EP/N028112/1, EU Safer Internet Programme project SI 2601002, and SFI Grants 10/CE/I1855, 13/RC/2094 and 15/SIRG/3501.

REFERENCES

[1] 1967. Katz v. United States. (1967), 347 pages.

[2] Rui Abreu, Dave Archer, Erin Chapman, James Cheney, Hoda Eldardiry, and Adrià Gascón. 2016. Provenance Segmentation. In *Proc. of the 8th USENIX Workshop on the Theory and Practice of Provenance*.

[3] D. Alrajeh, L. Pasquale, and B. Nuseibeh. 2017. On Evidence Preservation Requirements for Forensic-ready Systems. In *Proc. of the 11th Joint Meeting on Foundations of Software Engineering*. 559–569.

[4] A. Avizienis, J. C. Laprie, B. Randell, and C. Landwehr. 2004. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing* 1, 1 (2004), 11–33.

[5] D. Barske, A. Stander, and J. Jordaan. 2010. A Digital Forensic Readiness framework for South African SME's. In *Information Security for South Africa*. 1–6.

[6] Sheung Chi Chan, Ashish Gehani, James Cheney, Ripduman Sohan, and Hassaan Irshad. 2017. Expressiveness Benchmarking for System-Level Provenance. In *Proc. of the 9th USENIX Workshop on the Theory and Practice of Provenance*.

[7] Symantec Corporation. 2017. Internet Security Threat Report. https://www.symantec.com/security-center/threat-report. (2017).

[8] J. Cosic and M. Baca. 2010. Do We Have Full Control Over Integrity in Digital Evidence Life Cycle?. In *Proc. of the 32nd International Conference on Information Technology Interfaces*. 429–434.

[9] M. Elyas, S. B. Maynard, A. Ahmad, and A. Lonie. 2014. Towards A Systemic Framework for Digital Forensic Readiness. *Journal of Computer Information Systems* 54, 3 (2014), 97–105.

[10] B. Endicott-Popovsky, N. Kuntze, and C. Rudolph. 2015. Forensic readiness: Emerging discipline for creating reliable and secure digital evidence. *Journal of Harbin Institute of Technology (New Series)* 22 (2015), 1–8. Issue 1.

[11] B. E. Endicott-Popovsky and D. A. Frincke. 2006. Embedding Forensic Capabilities into Networks: Addressing Inefficiencies in Digital Forensics Investigations. In *Proc. of the IEEE Workshop on Information Assurance*. 133–139.

[12] S. L. Garfinkel. 2006. Forensic Feature Extraction and Cross-Drive Analysis. *Digital Investigation* 3 (2006), 71–81.

[13] C. P. Grobler and C. P. Louwrens. [n. d.]. *Digital Forensic Readiness as a Component of Information Security Best Practice*. 13–24.

[14] J. Grover. 2013. Android forensics: Automated Data Collection and Reporting from a Mobile Device. *Digital Investigation* 10 (2013), S12–S20.

[15] I. Hong, H. Yu, S. Lee, and K. Lee. 2013. A New Triage Model Conforming to the Needs of Selective Search and Seizure of Electronic Evidence. *Digital Investigation* 10, 2 (2013), 175–192.

[16] V. R. Kebande and H. S. Venter. 2016. On Digital Forensic Readiness in the Cloud Using a Distributed Agent-Based Solution: Issues and Challenges. *Australian Journal of Forensic Sciences* (2016), 1–30.

[17] J. T. King, J. Stallings, M. Riaz, and L. Williams. 2017. To Log, or Not To Log: Using Heuristics to Identify Mandatory Log Events - A Controlled Experiment. *Empirical Software Engineering* 22, 5 (2017), 2684–2717.

[18] R. McKemmish. 2008. *When is Digital Evidence Forensically Sound?* 3–15.

[19] L. Pasquale, S. Hanvey, M. Mcgloin, and B. Nuseibeh. 2016. Adaptive Evidence Collection in the Cloud Using Attack Scenarios. *Computers & Security* 59 (2016), 236–254.

[20] C. Peersman, C. Schulze, A. Rashid, M. Brennan, and C. Fischer. 2016. iCOP: Live Forensics to Reveal Previously Unknown Criminal Media on P2P Networks. *Digital Investigation* 18 (2016), 50–64.

[21] MET Police. 2017. Figures released ahead of National Child Sexual Exploitation Awareness Day. (2017). Retrieved 23.10.2017 from http://news.met.police.uk/news/figures

[22] A. Pooe and L. Labuschagne. 2012. A conceptual model for digital forensic readiness. In *Information Security for South Africa*. 1–8.

[23] D. Quick and K. R. Choo. 2016. Big Forensic Data Reduction: Digital Forensic Images and Electronic Evidence. *Cluster Computing* 19, 2 (2016), 723–740.

[24] R. Rowlingson. 2004. A Ten Step Process for Forensic Readiness. *International Journal of Digital Evidence* 2, 3 (2004), 1–28.

[25] C. Shields, O. Frieder, and M. Maloof. 2011. A System for the Proactive, Continuous, and Efficient Collection of Digital Forensic Evidence. In *Digital Investigations*, Vol. 8. 3–13.

[26] J. Stüttgen. 2011. Selective Imaging: Creating Efficient Forensic Images by Selecting Content First. *Mannheim University* (2011).

[27] J. Tan. 2001. Forensic Readiness. *Cambridge, MA:@ Stake* (2001), 1–23.

[28] P. Turner. 2006. Selective and Intelligent Imaging Using Digital Evidence Bags. *Digital Investigation* 3 (2006), 59–64.

[29] M. Yoshizawa, H. Washizaki, Y. Fukazawa, T. Okubo, H. Kaiya, and N. Yoshioka. 2016. Implementation Support of Security Design Patterns Using Test Templates. *Information* 7, 2 (2016), 34.

[30] S. A. Zonouz, K. R. Joshi, and W. H. Sanders. 2011. Floguard: Cost-Aware Systemwide Intrusion Defense Via Online Forensics and On-Demand IDS Deployment. In *International conference on Computer safety, reliability, and security*. Springer, 338–354.