Enhancing credibility of digital evidence through provenance-based incident response handling

Ludwig Englbrecht
Department of Information Systems, University of Regensburg
Regensburg, Germany
ludwig.englbrecht@ur.de

Gregor Langner
Faculty of Computer Science, Research Group Multimedia Information Systems, University of Vienna
Vienna, Austria
gregor.langner@univie.ac.at

Günther Pernul
Department of Information Systems, University of Regensburg
Regensburg, Germany
guenther.pernul@ur.de

Gerald Quirchmayr
Faculty of Computer Science, Research Group Multimedia Information Systems, University of Vienna
Vienna, Austria
gerald.quirchmayr@univie.ac.at

ABSTRACT

Digital forensics are becoming increasingly important for the investigation of computer-related crimes, white-collar crimes and massive hacker attacks. After an incident has been detected an appropriate incident response is usually initiated with the aim to mitigate the attack and ensure the recovery of the IT systems. Digital Forensics pursues the goal of acquiring evidence that will stand up in court for sentencing and sometimes opposes contradicting objectives of incident response approaches. The concept presented here provides a solution to strengthen the credibility of digital evidence during actions related to incident response. It adapts an approach for data provenance to accurately track the transformation of digital evidence. For this purpose, the affected system and the incident response systems are equipped with a whole system data provenance capturing mechanism and then data provenance is captured simultaneously during an incident response. Context information about the incident response is also documented. An adapted algorithm for sub-graph detection is used to identify similarities between two provenance graphs. By applying the proposed concept to a use case, the advantages are demonstrated and possibilities for further development are presented.

CCS CONCEPTS

• **Applied computing** → **Computer forensics**; *Evidence collection, storage and analysis*; • **Security and privacy** → *Intrusion/anomaly detection and malware mitigation.*

KEYWORDS

Incident Response, Digital Forensics, Digital evidence credibility, Data Provenance, Cyber Security, Evidence collection

ARES '19, August 26–29, 2019, Canterbury, United Kingdom

ACM ISBN 978-1-4503-7164-3/19/08…$15.00
https://doi.org/10.1145/3339252.3339275

1 INTRODUCTION

The detection of Advanced Persistent Threats (APT) can be very difficult since this type of attack are specifically tailored to certain companies or organisations. APTs are usually divided into three phases [8]. In the first phase, information about a penetration of the system is collected. Different techniques such as the evaluation of the target via social engineering or active vulnerability scanning are used. In the second phase, the actual infection of a system is prepared via waterholes or spear-phishing. In the third phase, data is extracted from the target system and traces are wiped out. Therefore information about known APTs allow the conclusion that the perpetrators have large financial resources as well as technical understanding. Securing traces in these situations is difficult, because the attack is usually only detected when damage has already occurred. The analysis of log files over a longer period of time in combination with Indicators of Compromise (IOC) is crucial to initiate and perform an effective incident response (IR).

Before a case can be investigated using Digital Forensics (DF), the process of an IR must be initiated. Different process models have already been defined for this purpose. In principle, the following seven components can be found in an IR approach: *pre-incident preparation, detection of incidents, initial response, formulate response strategy, investigate the incident, reporting and resolution* [15]. This ensures a goal-oriented progression, starting from a careful preparation to countering the attack until the incident is cleared up.

A DF investigation can also be initiated before an incident occurred. According to [7] the IR deals mainly with the immediate response and pursues a timely recovery of the systems. DF on the other hand, stretches from incident to normal operation and has its main objective in acquiring useful digital evidence. Since both aspects have different objectives, the two activities are inevitably in a conflict with each other. If evidence has been modified by IR actions, there is a significant risk that the evidence cannot be used for law enforcement purposes.

This problem has been recognised and guidelines for the integration of DF techniques into the IR process were developed [10]. Nevertheless, this integration has not yet been fully accomplished. It is still a field of ongoing research work. Even if an organisation invests a lot of effort in the area of IT security, its maturity level for a possible

DF investigation may not be sufficient [5]. In most cases, evidence is corrupted due to incorrect procedures within the IR procedures. Another challenge is the clear differentiation between the attack pattern and the IR measures.

The work presented in this paper aims to fill the gaps between protecting evidence, while recovering the system. A concept from the area of whole-system data provenance is used to collect IR data in order to distinguish the processes of the response team from the activities of an ongoing attack for a DF investigation. This concept can be used for a live forensic in order to increase the credibility and consequently the admissibility of acquired digital traces during an IR in court. The concept proposed here is presented as a preventive measure and can be assigned to the area of Digital Forensic Readiness (DFR). In particular, our concept ensures that traces of the attack have not been altered by IR measures.

The paper is structured as follows: in Section 2 we present the basics of DF in IR handling and whole-system data provenance. A conceptual presentation of the intended approach is presented in Section 3. In Section 4, a prototypical implementation of the concept is shown. The application of the model to a use case is described in Section 5 and the discussion of the results is provided in Section 6. Section 7 provides a summary of the proposed concept and an outlook on future work.

2 BACKGROUND AND RELATED WORK

Ensuring and increasing the admissibility of digital traces in court is a constant endeavour in science and practice. Especially the inappropriate handling of the data as well as the data storage medium during a DF investigation, can lead to the dismissal of decisive evidence from law enforcement proceedings. It is common for enterprises and organisations to be inadequately prepared for a DF investigation. This often leads to inappropriate actions being taken after an IT security incident has been detected. DFR deals with the adequate preparation of an organisation for a possible DF investigation through the development and implementation of specific DFR measures. That ensures that a DF investigation can be carried out in a targeted manner. For this, two objectives are pursued. The first goal is to conduct an DF investigation as efficiently as possible. The second goal is to find good traces and at the same time minimise the costs for an DF investigation [16].

Antwi-Boasiako and Venter [2] have developed a theoretical model to assess the admissibility of evidence. Their work proposes a harmonised framework for assessing digital evidence admissibility and to ensure the cross-jurisdictional acceptance and usability of digital evidence. Key technical and legal requirements are identified, combined and integrated within the framework for assessing digital evidence admissibility. Without the exact weighting of the impact of the individual criteria, the model shows that integrity verification in particular is an important factor for admissibility.

How data came to its current state can be described by using data provenance. It logs the origin of the file and all activities performed on it. This mechanism is already used in forensics or secure auditable logging. The idea of using data provenance is not new, but with the introduction of CamFlow an approach was found to integrate this mechanism with acceptable resource consumption into current systems [12]. CamFlow is the practical implementation of the whole-system provenance concept of Pohly et al. [14]. This whole-system provenance is a kernel-level provenance system which leverages the Linux Security Modules (LSM) framework guaranteeing completeness. CamFlow facilitates LSM which records how data has been transformed and information has been exchanged within a system through system calls. It is a flexible, efficient and easy to use provenance capture mechanism on operating system level. Beside other provenance systems operating on application or workflow level, this provides a provenance capturing between processes and enables a holistic view about what action occurred on the system [12].

CamFlow consequently implements the concept of a provenance monitor on kernel level in order to meet the requirements of a reference monitor concept [1]. A holistic mediation, tamper-proofness, and verifiability is given. The provenance history is also complete if an attacker is active on the system. This kernel module produces as an output, a provenance graph which is an extended version of the W3C PROV Data Model (PROV-DM) [3]. A provenance graph is a directed acyclic graph representing the execution within a system with vertices expressing states of kernel objects and the relations shows the information flow between those objects [12]. This concept provides the basis for the approach presented here. The objective of our study is to carry out detailed DF analysis in combination with IR using provenance data.

As stated by Pasquier et al. in [13] the mechanism for monitoring the information flow control with provenance-like data acquisition can meet requirements from the system audit. Thus, the transformations performed on the system to a file can be verified transparently while remaining trustworthy.

The application of CamFlow was evaluated by Han et al. [9] in the context of IT security. The application of data-provenance capturing was used for the detection of attack vectors in an intrusion detection system. The approach of Han et al. considers the use of a provenance graph to differentiate between normal and malicious activities for the detection of an attack. The opportunities and challenges of this approach provide the theoretical basis for the work presented in this paper.

Existing approaches also extend to the area of IR, but important requirements for the IR to ensure that the digital evidence can be used in a court of law, are not taken into account. We therefore focus on the usability of the traces, which may have occurred or been modified during an IR. A concept for the differentiation between compromised traces and untouched traces is the key contribution of our work.

3 DESCRIPTION OF THE MODEL

Technical measures for conducting a DF investigation include the provision of sufficient and appropriate capacity to store digital evidence. The digital evidence must be stored in such a way that it is secure from unauthorised access by third parties and retains its original condition. A common procedure for securing the data on a hard disk is, for example, the creation of an entire *image* of this data. This data can be stored as an exact copy of the original data on an external storage medium. Usually, a so-called *write-blocker* is used in such situations, to ensure that the data is not manipulated

by transmission or forensic analysis. [6]
In addition to the actual user data, this also applies to the metadata of a file, such as the time of access. The documentation of the *Chain of Custody* is also an important component. This is the complete traceability of the evidence, i.e. who had access to the data at a certain point in time [4].
An attacker can also be active during an IR. In this situation it is common that reciprocal actions occur. The attacker is repelled and tries several other ways to penetrate the system. In this process, the attacker can also make mistakes and, in the optimal case, valuable evidence for criminal prosecution can arise. The usability of such evidence should not be compromised by the fact that during the attack an organisation was actively defending their system with IR measures and has potentially corrupted or destroyed evidence [17].
The concept presented here is intended to supplement existing measures of DF investigations. After an incident has been detected, a 1:1 image of the systems can be created and used for further analysis. Our concept needs be implemented and prepared in advance as a precautionary DFR measure. In addition, a whole-system provenance capturing mechanism for the affected system and the systems for the execution of IR measures need to be installed before. The aim is to trace and understand IR related alterations on evidence and separate them from potentially emerging new evidence.

3.1 Application of data-provenance in incident response

Data provenance enables obtaining a more detailed and structured history of interactions of digital objects within a system. This was applied in areas such as system auditing and computational sciences. This concept can be used to acquire a holistic IR related view of system executions. To realise this, two provenance graphs from the attacked system and the system which performs IR activities needs to be acquired simultaneously. By an appropriate previous installation of CamFlow this just has to be activated on the attacked system in the event of an incident. A corresponding mechanism to extract the provenance data is already included in CamFlow. The client for the analysis must also be prepared accordingly. The graph of the attacked system is the super-set of all activities performed since the incident has been detected. The second graph of the IR system is a detailed record of all internal performed actions to mitigate the attack and to restore the system. As mentioned before, the proposed concept in this paper is an extension and not a replacement to common DF procedures where a forensically sound image of a system is created on a certain point of time.
Our concept pursues two objectives. The first goal is to track IR actions associated with the object under investigation. The second goal is to enrich the IR actions performed with context information about the activities performed. For the tracking of the actions it is assumed that IR actions are executed exclusively via a prepared IR client and the communication to the host is done via a TCP/IP connection. For the cross-host tracking of the actions using whole-system data provenance, payload data within the provenance graph is used. The recording of context information of the IR is done by a textual recording of the tasks and the capturing of the respective start and end time. As a result, a section of the provenance graph is given higher meaning. In order to store and reuse this information (text description and provenance graph) a flexible assignment is necessary. This is required since the IR tasks differ from case to case. Nevertheless, this requirement can be addressed as follows:
Given two provenance graphs, $G_1(V_1, E_1)$ of the IR related system and $G_2(V_2, E_2)$ of the infected system, a sub-graph detection can be performed. The comparative analysis of networks is a major topic in computational and integrative systems biology. Therefore, existing solutions from this discipline were used and adapted for forensic application. In [11] an algorithm and a tool for computing inexact solutions to the maximum common edge sub-graph problem for two or more graphs is presented. This algorithm is able to detect not just fully conserved edges but also partially conserved edges and is applicable to any set of directed or undirected, simple graphs. Given the possibility to relax the restrictions for the detection of common paths between two graphs and the existing practical implementation this approach has been chosen and adapted as described in the following paragraphs.
Since provenance graphs have vertexes and edges with a set of attributes (and not a single type attribute), combined with a varying number of such attributes, this has to be considered for detecting sub-graphs. Our approach therefore incorporates the *cosine similarity* into the comparison. This measure was used because it can handle missing attributes between two elements, irrespective to their magnitude. In addition, a threshold t has been introduced which determines the maximum divergence. The similarity of two vertexes, converted into two n-dimensional vectors (x and y), can be defined as:

$$cos(x, y) = \frac{x \cdot y}{||x|| \cdot ||y||} \tag{1}$$

The two graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$ are isomorph if there exists a bijective function $f : V_1 \rightarrow V_2$ such that there is an edge (u, v) in E_1 if and only if there is an edge $(f(u), F(v))$ in E_2 with the condition $cos(V_1, V_2) \geq 1 - t$. Given a set of two graphs $X = G_1(V_1, E_1), G_2(V_2, E_2)$, the determination of the number of edges that is isomorphic to the sub-graph of the first graph in X. Therefore, the alignment A as a set of injective functions is defined as follows:

$$A = f_i : V_1 \rightarrow V_i | 1 \leq i \leq 2 \tag{2}$$

where $f_1(v) = v$ for all $v \in V_1$. Given the alignment A, one can compute the number of edges conserved between G_1 and G_2.

$$f(A) = \sum_{u,v}^{n} \begin{cases} if(f_i(u), f_i(v)) \in E_i \forall i \in [1, n] \\ and\ cos(V_1, V_2) \geq 1 - t, \\ otherwise. \end{cases} \tag{3}$$

A provenance graph is a directed graph with edges indicating a one-way relationship. Therefore in this graph the sum have to run over all ordered pairs u, v where $u, v \in V_1$ and $u \neq v$. The algorithm of [11] is also capable to process un-directed graphs, but this will not be further discussed here due to the fact that this is not relevant for the use case of this paper. The given formulation of the (multi-) MCES problem builds the basis for finding an alignment A of two provenance-graphs by calculation $f(A)$.

3.2 The underlying conceptual model

The previously described approach provides an extension of existing DF approaches. Instead of creating a system image after the IT

security incident has become known and before initiating IR measures, a whole-system provenance capturing mechanism is initiated. This is done for the infected system (System 1) as well as the system that repels the attack (System 2). Provenance data is generated by both systems. A comparison engine can determine which paths of the provenance data interact with each other and detect similarities. The proposed concept also collects the context information, such as textual description of the applied IR measures, during IR and stores it together with the provenance data in a database.
This makes it possible to provide data about any transformation on data (respectsysively digital evidence) for use in a DF investigation. For example, the removal of malware or the closing of ports can be documented and justified in a transparent manner by the conformity of provenance paths. This finally leads to an enriched description of digital evidence. In addition to the initial 1:1 image, these can be used to gain further insights into the incident in relation to the differentiation to IR measures in court. The interaction of these components for the application of the concept is shown in the figure 1. The key element for the approach presented here

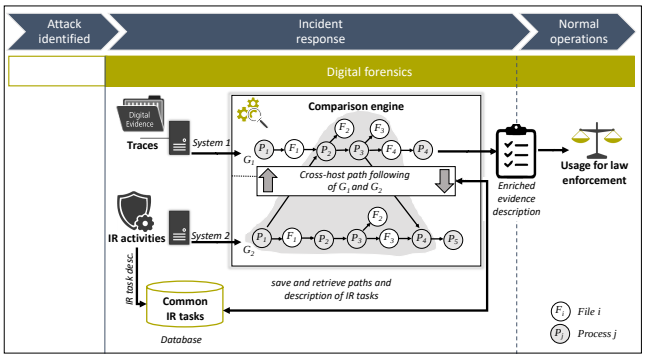

Figure 1: Schematic representation of the concept.

is the following of a provenance graph from the IR system to the effected system and the inclusion of context information from IR. To synthesise IR related tasks, the algorithm for determining two sub-graphs of [11] as a basis for the formal description of the project is adapted. The differentiation of IR measures and the activities of an attacker is a result of the path tracking and the consequential differentiation to remaining data transformations on the effected system.
As stated in [9] analysing dynamic, attributed graphs is not a trivial task. Especially the separation of normal and malicious activities is challenged to detect changes by attributed vertices and edges. These changes can occur both structurally and in labels. In particular, provenance graphs usually contain at every vertex and edge a set of attributes [9]. These sets can vary by attack vector and the application of IR measures.
By incorporating this algorithm it is possible to utilise the desirable aspect of similarity detection. The difficulty in comparing two provenance graphs can be overcome with the flexible approach which can tolerate minor perturbation. This means that the similarity of two graphs can be determined without accepting strong out-liners from a changed subset of attributes in the vertex and edge.

4 PROTOTYPICAL IMPLEMENTATION OF THE CONCEPT

A prototypical implementation of the approach has been performed. Therein a mechanism for the simultaneous acquisition of typical IR measures and the object under investigation using CamFlow was initially carried out. As described in [12] a cross-host provenance capturing is possible if the data is post-processed. This has been done and it was possible to demonstrate a possible application.
For storing the provenance data from the investigated object and the IR system, Neo4j[1] was chosen as a suitable database. Neo4j is an open-source, NoSQL database and is capable to store and process graphs provided in JSON format. Additionally, a HTML-based user-interface for recording IR activities (e.g.: *update firewall rules* or *reset admin rights*) has been added. Due to the usage of timestamps according the start and end of IR tasks an association between the textual IR task description and a sub-graph of the provenance data is possible.
Afterwards, the data was extracted from the database and imported into an existing prototype[2] to visualise the provenance graphs. This is shown in figure 2. The prototype uses a web-based implementation of Cytoscape[3]. This allows the extraction of the converted data of the provenance-date as a network for Cytoscape and to import it into the native implementation of Cytoscape for further analysis.

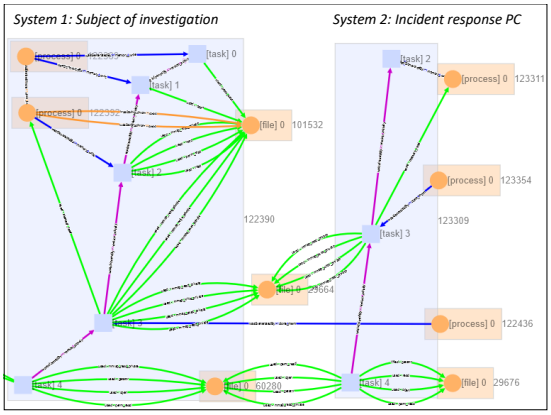

Figure 2: Provenance-based interaction between affected system and incident response computer.

To apply the concept, both graphs (from System 1 and System 2) were loaded separately into the prototype and exported as a Cytoscape network. Then, both graphs were loaded into the software Cytoscape and a sub-graph detection was performed.

5 APPLICATION OF THE MODEL TO A USE CASE

This section describes a possible application of the proposed concept of a data provenance-based IR handling to increase the credibility of digital evidence acquired during an IR.
The RUAG "Melani" case [8] was chosen as a baseline for the use

[1] https://neo4j.com/
[2] https://camflow.org/demo
[3] https://js.cytoscape.org/

case. The RUAG report was prepared by the Swiss government CERT. We used it because both architecture structures were described in detail. The attacker's approach, the evidence left behind and the IR approach were described in detail so that the major steps could be repeated. Four containers were created for the experimental setup. For two of these containers the conventional IR is carried out and for the remaining two, the IR is carried out according to our concept. All containers were configured identically. Each container was equipped with a Linux OS, a relational database, a firewall with default configurations and various files classified from unimportant to important. This was done to create an infrastructure for simulation regular business operations. Every login via SSH is documented in the firewall logs. For the databases three users were created, the admin-user with all rights, a user with read and write rights and a user with read rights only. No changes were made to the handling of log files of the database.

The attackers infrastructure includes a command and control server, three tier one server proxies, and three tier two server proxies. This is an identical structure as described in the RUAG case [8].

The experimental attack is divided into two phases. In the first phase the attacker obtains comprehensive rights and installs a malware. In the second phase the data is examined for relevance. The download of the data (from the infected system to the attacker) is initially started with a low download rate and increased continuously to prolong the time until detection.

First phase: After installing the malware, the target system was scanned for information about possible accounts and user rights. After the attacker identified a suitable account, the account was attacked using key loggers and fake mails. The goal was to identify username and password. After obtaining this data, the firewall settings of the system were changed. This included opening ports to guarantee permanent access as well as preparing for the download of data. Furthermore, the firewall logs were altered in order to conceal traces. The rootkits mentioned in the RUAG report [8] were not used to evaluate the basic concept presented in this paper, since no kernel information was manipulated. Log files that record data traffic were also altered, so that all suspicious data was deleted from the logs and the attacker additionally ensured future data to be redacted automatically by leaving a script in the system.

Second phase: During this phase, the attacker used the access rights obtained in phase one. The purpose of this step was to examine the target system and to find relevant files and database entries. These were tagged according to their level of importance, to ensure they would be included in the file download. The latter was designed to load all data tagged as highly relevant at a low download rate. After retrieving the important data, the remaining data is downloaded at a much higher rate, therefore increasing the risk of being caught. This strategy ensures that traffic analysis does not result in a rise of the IOC.

To make the data collected during the experiment more comparable, the test setup was regularly stopped at important points and appropriate response mechanisms initiated. When applying IR procedures on the first two containers, evidence was obtained without considering DF requirements. At the same time, a forensic image of the situation was created for future analysis. Among the actions initiated by the IR team was the closing of the firewall ports and resetting them to default values as well as deleting any suspicious software. The deletion was done without checking which software was installed and how it worked. Therefore an analysis of the log files for suspicious data points must be done without having a deep knowledge of the malware.

The analysis approach described in the above paragraph describes the current state-of-the-art IR procedure and does not include software or concepts of the solution of this paper.

In the first experimental setup wherein the malware is installed and user rights obtained, the process was stopped and the IR was carried out. All further traces may have been compromised by IR actions. Although there was evidence that something was altered in the system and there was no data in the log files that could be tracked or used to distinguish between IR and the attacker.

In the second trial setup, the attack was not detected until the download reached a critical value. As an immediate measure, all ports were closed and reset to default. Since the attacker still had access to the system, the ports were reopened and the download was restarted. After another peak in the download rate, another IR measure was performed. This resulted in the closure of all ports, without allowing them to be reopened. Also, the user administration was reset and all accounts deleted except admin access. The attacker had no more rights and no access to the system. Suspicious software was now removed without analysis of its area of use. The log files and firewall files were checked for suspicious entries without additional software support. As a last step the ports were released one after the other and DNS filters were implemented. In this case, the IR destroyed most of the data or prevented the occurrence of new evidence.

The remaining containers, on which the concept presented in Section 3 were applied. The data from the first test setup on the first two containers, is used here as a basis for comparison. The IR is carried out on the basis of the previously explained concept and the consideration of data-provenance. Log files created by the CamFlow were not altered. These log files cannot be adapted as easily, by changing the data since the signature would have been corrupted. In contrast to the previous containers, data-provenance capturing was implemented on the attacked target system and the IR computer. This enables the actions taken for the mitigation of the attack can be distinguished from the actions of the attacker.

After a forensically-sound image of the affected system has been created, the target system and an IR system were equipped with a whole-system data provenance capturing mechanism. In both systems, the provenance data was extracted into a Neo4j database. In addition, the actions of the IR team were recorded via a simple web interface. After the execution of the IR measures a second forensically sound image was created. The collected provenance data of both systems were loaded into Cytoscape and processed with the adapted CytoMCS algorithm aiming to match data-provenance sub-graphs. After applying the sub-graph detection algorithm, an aligned network was created. In the underlying data of this network a juxtaposition of the nodes is given so that a connection between the graphs can be obtained.

6 DISCUSSION OF RESULTS

In the test setups it was shown that the different approaches have advantages and disadvantages. In the first test run the aim was to

restore the system to a working state as quickly as possible and no further measures were taken to secure the evidence during the IR. The utilisation of evidence is not guaranteed due to this circumstance. The evidence is destroyed or no longer meaningful due to the alterations made during IR. Law enforcement based on this evidence is therefore not possible.

In the second experimental setting, the proposed concept with data-provenance was applied. Here it was shown that time has an important influence. Equipping an IR system with the additional software presented in this paper takes a significant amount of time. Creating an image of an entire system also takes a significant amount of time. The evaluation of the traces is not possible in a single application. Thus, in the first test setup according to the concept, the data was evaluated several times to identify the suspicious data points. After documenting the evidence, it had to be compared to the original evidence. The aim of the concept is to show that the evidence has been altered in a controlled and comprehensible manner by IR related analysis. If evidence have been modified by IR, this can be easily understood with the concept. Even with the sub-graph detection method an expert needs to check and verify these. Nevertheless, a complete analysis with the tool requires a complete knowledge of the systems used. This means that all affected systems of the attack as well as the IR systems used must be known and equipped with a data-provenance capturing method. Without this step, the overall view of the incident and the systems used could be missing and wrong conclusions could be drawn.

The evaluation of the graphs provides a valuable insight into the DF analysis of an incident and new evidence created during the IR procedure could be used for law enforcement.

The security of the presented concept is based on the assumption that the integrity of the system is guaranteed at run-time and features of using a Trusted Platform Module (TPM) are used by CamFlow. Checking for installed rootkits on the system is still necessary during the application of our concept in order to guarantee the conclusiveness of the traces.

Furthermore, the attacker may recognise that a provenance capturing mechanism has been installed on the infected system and therefore behave differently based on this information.

7 CONCLUSION AND FUTURE WORK

If an attack on information systems is detected, an IR is initiated. This pursues different objectives than a DF investigation. While IR aims for the quick restoration of systems, DF aims to acquire evidence that stands up in court. With data provenance the transformations on a file can be traced from its data source to its present form. CamFlow, a whole-system provenance capturing module which facilitates latest kernel features of Linux builds the baseline for our approach presented here.

The proposed concept shows how the activities of an IR can be distinguished from countermeasures of the attacker using data provenance capturing. The provenance graphs of two systems are generated simultaneously, compared and enriched with IR related context data. This has the advantage that not only a forensic image can be used at the time before the IR, but also evidence that has arisen during the mitigation of an incident.

We have shown in the use case that an incident can cause some problems and the initiated IR can compromise or even destroy evidence. An application of the proposed concept can overcome these issues and provides an enhanced credibility of the digital evidence. The present concept provides the basis for a profound integration of whole-system data-provenance capturing into the handling of incidents. Nevertheless, the concept presented needs to be further evaluated. The determination of the threshold for determining the similarity of two vertexes must be further refined and the application of the model must be discussed with law enforcement experts. A further research project is the application of the concept to systems which cannot be forensically acquired within a reasonable time. The use of whole-system provenance would enable a simultaneous IR with data acquisition for a DF investigation. This makes it possible to track all changes made during the period of evidence acquisition.

8 ACKNOWLEDGEMENT

The authors would like to thank the EU H2020 project CS-AWARE (grant number 740723) for supporting the research presented in this work.

REFERENCES

[1] James P. Anderson. 1972. Computer Security Technology Planning Study. (1972).
[2] Albert Antwi-Boasiako and Hein Venter. 2017. A Model for Digital Evidence Admissibility Assessment. In *IFIP International Conference on Digital Forensics*. Springer, 23–38.
[3] Khalid Belhajjame, James Cheney, Sam Coppens, Stephen Cresswell, Yolanda Gil, Paul Groth, Graham Klyne, Timothy Lebo, Jim McCusker, et al. 2013. PROV-DM: The PROV Data Model. (2013).
[4] Jasmin Cosic, Goran Cosic, J Ćosić, and Z Ćosić. 2012. Chain of custody and life cycle of digital evidence. *Computer Technology and Aplications* 3 (2012), 126–129.
[5] Ludwig Englbrecht, Stefan Meier, and Günther Pernul. 2019. Towards a capability maturity model for digital forensic readiness. *Wireless Networks* (01 January 2019).
[6] Anders O Flaglien, Aleksander Mallasvik, Magnus Mustorp, and André Årnes. 2011. Storage and exchange formats for digital evidence. *digital investigation* 8, 2 (2011), 122–128.
[7] Felix Freiling and Bastian Schwittay. 2007. A common process model for incident response and digital forensics. *Proceedings of the IMF2007* (2007).
[8] GovCERT.ch. 2016. *Technical Report about the Espionage Case at RUAG.* Technical Report.
[9] Xueyuan Han, Thomas Pasquier, and Margo Seltzer. 2018. Provenance-based Intrusion Detection: Opportunities and Challenges. In *10th {USENIX} Workshop on the Theory and Practice of Provenance (TaPP 2018)*.
[10] Karen Kent, Suzanne Chevalier, Tim Grance, and Hung Dang. 2006. Guide to integrating forensic techniques into incident response. *NIST Special Publication* 10, 14 (2006), 800–86.
[11] Simon J Larsen and Jan Baumbach. 2017. CytoMCS: a multiple maximum common subgraph detection tool for Cytoscape. *Journal of integrative bioinformatics* 14, 2 (2017).
[12] Thomas Pasquier, Xueyuan Han, Mark Goldstein, Thomas Moyer, David Eyers, Margo Seltzer, and Jean Bacon. 2017. Practical whole-system provenance capture. In *Proceedings of the 2017 Symposium on Cloud Computing*. ACM, 405–418.
[13] Thomas F. J.-M. Pasquier and David M. Eyers. 2016. Information Flow Audit for Transparency and Compliance in the Handling of Personal Data. In *2016 IEEE International Conference on Cloud Engineering Workshop, IC2E Workshops, Berlin, Germany, April 4-8, 2016*. IEEE Computer Society, 112–117.
[14] Devin J Pohly, Stephen McLaughlin, Patrick McDaniel, and Kevin Butler. 2012. Hi-Fi: collecting high-fidelity whole-system provenance. In *Proceedings of the 28th Annual Computer Security Applications Conference*. ACM, 259–268.
[15] Chris Prosise, Kevin Mandia, and Matt Pepe. 2003. Incident response & computer forensics. (2003).
[16] John Tan. 2001. Forensic Readiness. (2001).
[17] Shiuh-Jeng Wang and Cheng-Hsing Yang. 2005. Gathering digital evidence in response to information security incidents. In *International Conference on Intelligence and Security Informatics*. Springer, 644–645.