Chapter 22

APPLYING THE BIBA INTEGRITY MODEL TO EVIDENCE MANAGEMENT

Kweku Arthur, Martin Olivier and Hein Venter

Abstract This paper describes the design of an integrity-aware Forensic Evidence Management System (FEMS). The well-known Biba integrity model is employed to preserve and reason about the integrity of stored evidence. Casey's certainty scale provides the integrity classification scheme needed to apply the Biba model. The paper also discusses the benefits of using an integrity-aware system for managing digital evidence.

Keywords: Evidence management, Biba integrity model, Casey's certainty scale

1. Introduction

The Internet has enabled the sharing of resources between millions of computer systems around the world. Applications such as email, retail banking and general information interchange (through file transfer and peer-to-peer networks) have positively influenced the lives of Internet users. However, vulnerabilities in applications, operating systems and networks connected to the Internet are constantly emerging—with cyber criminals making it their mission to identify and exploit these vulnerabilities. Their exploits can be categorized as "true cyber crime" and "e-enabled cyber crime" [3]. True cyber crime is a dishonest or malicious act that would not exist outside an online environment, e.g., virus attacks, suspicious probes (hacking) and denial-of-service attacks. E-enabled crime is a criminal act that was encountered before the advent of the Internet, but which is increasingly perpetrated using the Internet, e.g., fraud and identity theft [3].

Investigative methodologies, forensic tools, and host- and network-based forensic techniques have rapidly evolved to combat the unrelenting increase in cyber crime. However, a major challenge is the reliability and integrity associated with digital evidence from disparate sources. Given the nature of forensic investigations and the ease with which digital evidence (especially metadata) can be created, altered and destroyed [10], it is extremely important to protect evidence and preserve its integrity.

This paper describes the design of a Forensic Evidence Management System (FEMS) that determines and protects the integrity of digital evidence stored in the system, regardless of whether the evidence was entered as a basic fact or was inferred from pre-existing evidence by the system. The well-known Biba integrity model [14] is employed. The Biba model labels data within integrity classes. In the model, an application may only read data with an integrity classification equal to or higher than its own integrity classification. Also, the application may only write data to containers with integrity classifications that are equal to or lower than its own classification. Since data can only flow from higher to lower integrity classes, low integrity data cannot contaminate high integrity data.

An integrity classification scheme is required to apply the Biba model in the context of digital evidence. Casey's certainty scale [5] is useful for expressing the certainty of facts and inferences in a networked environment. Since certainty equates to integrity, Casey's scale is an ideal starting point for expressing the integrity of digital evidence in FEMS. In fact, we use the terms certainty and integrity interchangeably in this paper.

The remainder of this paper is organized as follows. Section 2 provides background information about data classification, information flow, the Biba integrity model and Casey's certainty scale. Section 3 describes the FEMS architecture. Section 4 discusses information flow within FEMS and illustrates the benefits of using an integrity-aware system. The final section, Section 5, presents our conclusions.

2. Background

Information has great value, and practically every enterprise gathers and uses information to gain a competitive advantage. At the same time, malicious agents actively attempt to exploit this information for financial gain. The recent CSI/FBI Computer Crime and Security Survey [8] reveals the massive costs associated with the loss or disclosure of private or proprietary information. As a result, enterprises must implement security mechanisms aimed at mitigating the risks of information loss and disclosure.

2.1 Data Classification and Information Flow

The risk of financial loss is one of the principal motivations for implementing data classification and access control and, in general, controlling information flow in an organization. Data classification involves the assignment of sensitivity levels to specific data assets [18]. A data classification exercise has to performed within the organization before an appropriate access control model can be implemented. Consequently, a risk analysis is undertaken, all organization critical assets are identified, the impact of the loss of these assets is determined, and appropriate controls are adopted to protect the assets [6]. A technical access control implementation incorporates preventative, detective and/or corrective controls, e.g., access control lists, intrusion detection systems and patch management. The access control model ultimately controls information flow, often implementing security principles such as least privilege and need-to-know [9].

2.2 Biba Integrity Model

The Biba integrity model was the first to address integrity in computer systems [17]. It is based on a hierarchical lattice of integrity levels (similar to the Bell-LaPadula confidentiality model). The Biba model orders subjects (s) and objects (o) using an integrity classification scheme, denoted by $I(s)$ and $I(o)$, respectively. The integrity classification scheme controls object modification. The Biba model is based on two properties [14]:

- **Simple Integrity Property:** Subject s can read object o only if $I(s) \geq I(o)$.
- *** Integrity Property:** If subject s can read object o, then s can write to object p only if $I(o) \geq I(p)$.

These rules address the integrity of information in a natural way. Suppose a person A is known to be untruthful. If A creates or modifies documents, then others who access this document should distrust the truth of the statements in the document. Furthermore, if people are skeptical about a report based on flawed evidence, the low integrity of the source evidence (object) should imply low integrity for any evidence (object or inference) based on the source evidence (object).

2.3 Casey's Certainty Scale

If properly programmed and configured, all information technology and network objects are capable of producing log data reflecting their

Table 1. Proposed scale for classifying integrity levels (adapted from [5]).

Level	Description/Indicator	Qualification
C0	Evidence contradicts known facts	Erroneous/Incorrect
C1	Evidence is highly questionable	Highly Uncertain
C2	Only one source of evidence exists and it is not protected from tampering	Somewhat Uncertain
C3	The source(s) of evidence are more difficult to tamper with but there is not enough evidence to support a firm conclusion or there are unexplained inconsistencies in the evidence	Possible
C4	Evidence is protected from tampering or multiple, independent sources of evidence agree but the evidence is not protected from tampering	Probable
C5	Agreement of evidence from multiple, independent sources that are protected from tampering but uncertainties exist (e.g., temporal errors, data loss)	Almost Certain
C6	Evidence is tamper proof and unquestionable	Certain

activity [7, 15]. In the event of an incident, these audit logs could assist with reconstructing and understanding the activities that led to the incident [12]. Therefore, event auditing capabilities are typically enabled with respect to the classification of data, applications and systems to be protected. From this, one can understand why it is necessary for web servers, FTP servers, mail servers and access control mechanisms to generate logs. However, the logs are prone to data corruption, loss, tampering, incorrect interpretation and lead time in transmission that could render the evidence useless. Time-based differences such as time-zone bias and differing system clock speeds and settings also contribute to errors. Therefore, it stands to reason that levels of certainty must be associated with digital evidence as well as with evidence sources.

Casey's certainty scale [5] was developed to address the inherent uncertainties related to digital evidence in networked environments. In particular, it enables certainty (integrity) assessments to be associated with digital data. Casey's proposal is illustrated in Table 1. The first column of the table lists the seven certainty (integrity) levels. The second column indicates the preconditions that lead to the integrity conclusions in the third column. Note that the higher the certainty (integrity) level, the greater the integrity associated with the evidence source and, hence, the inferences based on the evidence.

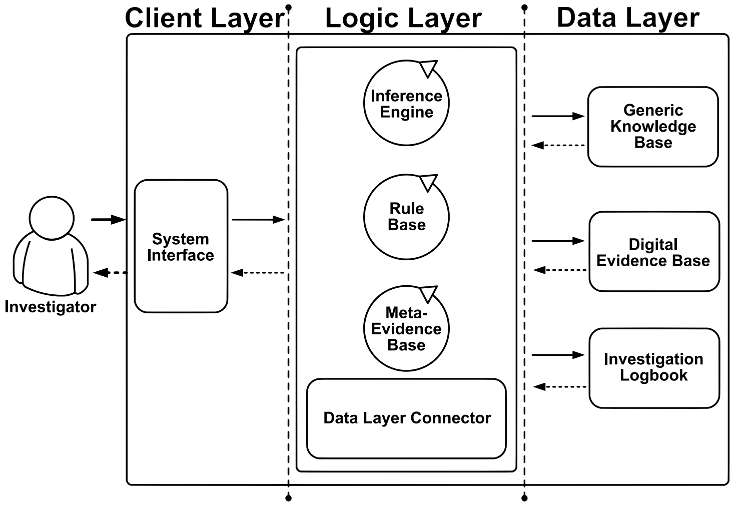

Figure 1. Forensic Evidence Management System (FEMS) architecture.

3. Forensic Evidence Management System

Digital forensic techniques are employed when crimes are commissioned through the use of computers and the evidence is electronic in nature. Digital evidence sources include digital cameras, log files, network traffic, local and removable storage devices, and metadata such as MAC times [16]. If a cyber crime is suspected, a digital forensics specialist is tasked to investigate the electronic crime scene to reveal the root cause or perpetrator. Proprietary tools (e.g., FTK [1]) are often used during the investigative process. Unix-based and open source tools such as `grep` and `dd` may also be used in digital forensic investigations [11].

The multitude of digital evidence sources presents a major challenge to investigators. An aggregated forensic evidence store could greatly assist investigators by providing them with holistic views of the forensic evidence pertaining to cases and insights into the quality of their inferences.

Our integrity-aware Forensic Evidence Management System (FEMS) was designed with these issues in mind (Figure 1). The FEMS architecture distributes its components within a client layer, logic layer and data layer. The client layer serves as the investigator's interface to the system. The logic layer houses the processing rules. The data layer stores the data used in an investigation and provides the necessary abstraction layer between the investigator and the raw evidence stored in the system.

3.1 System Components

FEMS has several components: system interface, rule base, meta-evidence base, inference engine, investigation logbook, digital evidence base and generic knowledge base (Figure 1).

The system interface is the channel through which a forensic investigator accesses facts within FEMS. The investigator uses queries to interrogate the system for evidence; these queries include hypotheses that the investigator tests against facts within the system. The system interface also enables the investigator to update data.

The rule base is a store for the action(s) to be taken given any particular fact in the system. The rule base also represents knowledge about facts that can be inferred by the system [4]. We assume that the rules do not necessarily exist *a priori*; they may be entered *a posteriori* based on the specific case at hand. In addition to supporting inferences, the rule base is incorporated due to its ease of implementation and the intuitive nature of rule generation [2]. For example, a rule may specify integrity labels to be associated with facts from a specific evidence source or facts derived from an evidence source.

The meta-evidence base, which is directly interfaced with the rule base, only houses inferred evidence. Queries to the system that have been confirmed or refuted by the system are routed to and noted in the meta-evidence base.

The inference engine implements Casey's certainty scale. It draws its inputs from the rule base and the meta-evidence base in ascribing certainty labels to evidence and inferences.

The investigation logbook records all investigative actions. The automated documentation of actions is certainly not new. Several forensic tool suites (e.g., Encase, FTK and ProDiscover) offer this feature; FTK, for example, even provides for customizable reports. In the FEMS context, system queries, hypotheses, rules, data and inferences are recorded. The logbook thus traces the inputs and the reasoning used to reach the investigative conclusions.

The digital evidence base interfaces to forensic tools, e.g., EnCase and FTK. The evidence base also incorporates inputs from log correlation sources, logical access records (including access control matrices), and physical access records such as work attendance registers.

The generic knowledge base houses static information such as software names, versions and descriptions. It also includes a database of known files: standard operating system files and hashes for generic file signatures such as GIF or JPG compressed files. The generic knowledge base has the same function and is referenced in the same way as

FTK's "Known File Filter" [1] and NIST's National Software Reference Library [13].

3.2 Evidence

Digital evidence in FEMS is often stored in the form of facts:

- Fact 1: web proxy 223.1.3.27 return code 200 = OK: Request succeeded
- Fact 2: web proxy 223.1.3.27 return code 400 = Bad request: Request could not be understood by the server

Fact 1 above is interpreted as "client request successful," while Fact 2 is interpreted as "client request not understood by the proxy server." Facts may be represented as predicates.

For a murder case in the physical world, the fact that gunshot residue was found on suspect S could be expressed by the predicate *GSR(S)*. Similarly, if S was placed at the scene of the crime, the corresponding predicate is *AtScene(S)*. Other predicates may be used to represent various facts about S, about the crime itself, and about other agents.

The physical world is a realm where intangible properties such as time, space and physical location cannot be controlled or amended by agents (humans). It is also characterized by its deterministic and finite nature, where an action has a definitive source and destination. On the other hand, actions in a digital environment may be virtually independent of time and physical location and, more often than not, the source of an action is obfuscated. Also, in the digital world, properties such as time and location can be amended by agents. Consequently, something that may be easy to prove in the physical world could be difficult, if not impossible, to prove in the digital world.

The differences between the two worlds are important because it is possible to automatically derive facts in the digital world from augmented forensic tools. Suppose, for example, that it is relevant whether or not a certain picture or email messages were on a suspect's computer. These facts could be added automatically to an evidence knowledge base when the hard drive of the suspect's computer is imaged and analyzed using a digital forensic tool. Note, however, that not all facts in the digital world can be derived automatically. Due to the subjective nature of pornography, the question of whether or not a picture found on a suspect's computer is a pornographic image may only be answered by a human investigator.

The integrity of evidence can expressed by associating integrity levels from Casey's certainty scale with facts. For example, if image I was found on a disk, the fact may be represented as *OnDisk(I, C6)*. Certain deduction rules are applicable regardless of the integrity of the facts on which they operate. Suppose that an image containing data that was hidden using steganography is found on a computer. An investigator would want to know whether or not there is evidence of the presence of a tool for decoding the hidden message. This rule would apply regardless of the certainty of the preconditions. Alternatively, a rule that categorizes the image based on an applied compression algorithm would also be applicable regardless of the image integrity label.

In line with the Biba model, the certainty of a fact derived by such a rule will depend on the certainty of its preconditions. In actuality, the new fact will, in general, have the lowest certainty of its preconditions. For example, suppose an image came from a removable disk that had been corrupted by a virus. Clearly the evidence source cannot be trusted completely. The Casey scale states that the certainty of a fact can increase if it is supported by independent sources. Consider the requirement for the C5 level in Table 1: "Agreement of evidence from multiple, independent sources that are protected from tampering but uncertainties exist (e.g., temporal errors, data loss)." Therefore, the converse to our earlier example is that, if the image was retrieved from a secure FTP application, which is subsequently corroborated by FTP session logs in the evidence management system, then the integrity of the image and other evidence based on the image is improved. In other cases, some facts that have been well established may lead one to form a rather tentative hypothesis about other facts, in which case the certainty of the new fact will be lower than its preconditions. Consequently, there is a need for trusted upgraders and downgraders. This issue is discussed in the next section.

4. Information Flow

The flow of information within FEMS is described using a computer intrusion scenario. Such an incident usually warrants a thorough forensic investigation.

Consider an intrusion that exploits a web browser vulnerability to compromise a computer. Assume that the intruder subsequently accesses a financial system by harvesting authentication information from the compromised computer. In this context, the forensic investigator would interrogate FEMS for configuration files, source code, executable programs (e.g., rootkits), Internet activity logs and password-protected text files. These queries would be submitted using the FEMS system interface and then brokered by the data layer connector, which parses information returned by the data layer.

Table 2. Upgrader matrix.

	C0	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**
C0	C0	C0	C0	C0	C0	C1	C1
C1		C1	C1	C1	C1	C2	C2
C2			C2	C2	C2	C3	C3
C3				C3	C3	C4	C4
C4					C4	C5	C5
C5						C5	C6
C6							C6

Suppose that the intruder has modified event logs during the attack. Therefore, the Internet activity logs may have been tampered with. However if these logs had been generated and sent directly to a secure log correlation server, then the rule base might infer: *LCorrelation(Internet log, C6)*, i.e., the log-related information is tamper proof and unquestionable.

At this point, the intruder's access must be verified in the audit logs of the financial system. However, assume that the audit logs are deemed to be unreliable because they are not explicitly protected from tampering. This situation could be expressed by the fact: *FinSyst(log, C2)*. Using this fact and the inference rule: *for (LCorrelation(log, C6)* $\geq$ *FinSyst(log, C2)) update_meta-evidence*, it would be deduced by the inference engine (and sent to the meta-evidence base) that, although the financial system logs verified that the victim's credentials were used at a specific time, conclusions based on this information should not be trusted.

We are now in a position to discuss the concepts of upgraders and downgraders. An upgrader is any evidence or evidence source with an integrity label $\geq$ C5 (and corroborated by two or more trusted evidence sources), which is used to improve the certainty associated with facts or inferences in FEMS. In contrast, a downgrader is any evidence or evidence source with an integrity label $\leq$ C1. Upgraders and downgraders are influential because they cause the inference engine to modify evidence integrity labels.

The log correlation evidence source in our example is considered to be an upgrader. This is because, all else being equal, the implementation of a correlation solution is typically fortified. Therefore, as a direct consequence of the Biba model, the log correlation evidence source is allowed to upgrade the integrity label of the financial system log. Table 2 presents a sample upgrader matrix. Using the available information, the inference engine upgrades the integrity label of the financial system log to C3.

Although the financial system logs may be included within the log correlation system, they may not positively influence the integrity of other evidence in the system until their own integrity is enhanced. The investigation logbook is programmatically instructed to record all logical steps and inferences throughout this process.

5. Conclusions

The Forensic Evidence Management System (FEMS) is an attractive solution to the problem of assessing, maintaining and reasoning about the integrity of digital evidence in networked environments. The solution uses Casey's certainty scale in conjunction with the well-known Biba integrity model. The FEMS architecture incorporates a rule base, meta-evidence base, inference engine, digital evidence base and generic knowledge base for reasoning about the integrity of evidence. It also offers a system interface for evidence input and queries, and an investigation logbook that records all investigative actions. The principal benefit of FEMS is that it provides investigators with holistic views of the forensic evidence pertaining to their cases and insights into the quality of their inferences.

References

[1] AccessData, Forensic Toolkit (FTK) (www.accessdata.com).

[2] Aprisma, Event correlation in Spectrum and other commercial products (www.aprisma.com/literature/white-papers/wp0551.pdf), 2000.

[3] K. Burden and C. Palmer, Cyber crime – A new breed of criminal? *Computer Law and Security Report*, vol. 19(3), pp. 222–227, 2003.

[4] L. Burns, J. Hellerstein, S. Ma, C. Perng, D. Rabenhorst and D. Taylor, Towards discovery of event correlation rules, *Proceedings of the IEEE/IFIP International Symposium on Integrated Network Management*, pp. 345–359, 2001.

[5] E. Casey, Error, uncertainty and loss in digital evidence, *International Journal of Digital Evidence*, vol. 1(2), 2002.

[6] H. Doernemann, Tool-based risk management made practical, *Proceedings of the IEEE Joint Conference on Requirements Engineering*, p. 192, 2002.

[7] D. Forte, The art of log correlation: Tools and techniques for correlating events and log files, *Computer Fraud and Security*, pp. 7–11, June 2004.

[8] L. Gordon, M. Loeb, W. Lucyshyn and R. Richardson, *2006 CSI/FBI Computer Crime and Security Survey*, Computer Security Institute (i.cmpnet.com/gocsi/db_area/pdfs/fbi/FBI2006.pdf), 2006.

[9] S. Harris, *CISSP Certification*, McGraw-Hill Osborne, Emeryville, California, 2005.

[10] C. Hosmer, Proving the integrity of digital evidence with time, *International Journal of Digital Evidence*, vol. 1(1), pp. 1–7, 2002.

[11] R. Morris, Options in computer forensic tools, *Computer Fraud and Security*, pp. 8–11, November 2002.

[12] A. Muscat, A log-analysis-based intrusion detection system for the creation of a specification-based intrusion prevention system, *Proceedings of the University of Malta Annual Computer Science Research Workshop*, 2003.

[13] National Institute of Standards and Technology (NIST), National Software Reference Library (www.nsrl.nist.gov).

[14] C. Pfleeger and S. Lawrence-Pfleeger, *Security in Computing*, Prentice Hall, Upper Saddle River, New Jersey, 2003.

[15] B. Smith, Thinking about security monitoring and event correlation (www.lurhq.com/confarticle.htm).

[16] P. Stephenson, The right tools for the job, *Digital Investigation*, vol. 1(1), pp. 24–27, 2004.

[17] H. Tipton, Integrity models (www.ccert.edu.cn/education/cissp/hism/023-026.html).

[18] J. Tudor, *Information Security Architecture: An Integrated Approach to Security in the Organization*, Auerbach/CRC Press, Boca Raton, Florida, 2001.