Rebecca T. Mercuri

On Auditing Audit Trails

It is incumbent upon us to examine our own auditing practices for their intrinsic vulnerabilities.

The current auditing crisis of big business can provide useful lessons and suggestions for improving similar practices within the computer industry.

Audit trails, whether computer-based or manually produced, typically form a significant part of the front-line defense for fraud detection and prevention within systems. Many of our security practices revolve around the generation and preservation of authenticated data streams that are to be perused routinely or periodically, as well as in the event of system attack, failure, or other investigations. But these audit trail systems are not necessarily robust, since components can be subverted or ignored. Furthermore, it is the surrounding controls, or overriding design-and-use philosophies, that are often discovered to be inadequate or circumvented.

Take the recent bogus accounting practices reported in 2002. Center stage was the Enron scandal, followed in short order by WorldCom, whose $7 billion in fictitious profit reporting dwarfed Enron's concealment of a mere $1.8 billion in debt. By year's end, the list of companies with questionable reporting included ImClone, Tyco, Xerox, Qwest, Merck, and AOL. Accounting firm Arthur Andersen was hardest hit, with bookkeeping practices for nine corporations questioned, while giants PricewaterhouseCoopers, KPMG, Deloitte & Touche, and Ernst & Young each also had clients under investigation. The resulting ripple effect of these challenges, and the loss of credibility in auditing, worsened an already sluggish stock market, and the economy was hard hit, with employment downturns in many related sectors.

Fraudulent information reporting has had adverse impacts in other areas as well. At Bell Labs, superconductor research by physicist Jan Hendrik Schön was revealed to have included falsified data at least 16 times in two years. The sheer number of papers co-authored by the young scientist—80 between 1998 and 2001—should have raised skepticism, but it was the lack of reproducibility that ultimately blew the whistle on the sham.

Schön was fired in September 2002, and six patent applications were dropped by the corporation. This was vaguely reminiscent of the 1998 downfall of freelance writer Stephen Glass, after revelations that he had fabricated a *New Republic* story about a computer hacker convention that turned out to have never existed. A subsequent investigation revealed that all or part of 27 of 41 stories Glass had written for *New Republic*

PETER HOEY

included phony materials. Glass even went so far to cover his trail by faking Web sites, letterheads, faxes, and voice mails from supposed sources.

How could all of this have happened? Accounting firms are expected to provide independent certification of financial reports. Scientific co-authors are supposed to witness the experiments on which they are reporting (Schön's never did). Publications and news agencies should earn the respect of their readers through extensive fact-checking and peer review. Did the auditing systems fail? Or did they succeed, because detection ultimately occurred—albeit somewhat belatedly. Perhaps the questions to ponder are: How much fraud has been left unnoticed?; and What are our roles and responsibilities as computer scientists in the construction of auditing processes?

Some direction may be found in a March 2002 report from Queensland, Australia that examined the courtroom use of audit trail data from law enforcement agencies to corroborate evidence [1]. The 11 cases in the study, performed by Caroline Allinson, highlighted internal corruption and misuse of government information, and included the conviction of a police officer who ran a stolen car ring, sexual harassment charges against a constable, and aggravated misappropriation of property. In the study, numerous criteria were identified for the significance of audit information:

- Proof of user activity
- Technical security for audit trails
- Expertise of information technology staff
- Relevance of security officer certification
- Proof of a business process
- Audit trail content
- Rules of evidence
- Recording of details by police officers
- Time and relevance of routine checks
- Recording of all activity
- Functionality of application systems
- Positive identification of users
- Documentation for process and procedure

These criteria can serve as an initial data construction and evaluation checklist, helpful since Allinson's conclusions revealed a growing trend for audit trail requests by the prosecution, a lack of challenge of such trails by the defense, and a troubling decrease in security of the audit trail material because of the migration from mainframe systems to networked personal computers.

The use of audits and audit trails also plays a major role in the information technology security evaluation model of the Common Criteria (CC) [3]. Security auditing is defined as involving the recognition, recording, storage, and analysis of "information related to security-relevant activities." Audit records are intended to "be examined to determine which security-relevant activities took place and whom (which user) is responsible for them." Security auditing is provided in CC Part 2, Chapter 3. The following are its six components:

- *Automatic response*. Defines reactions taken following detection of events that are indicative of a potential security violation.
- *Data generation*. Identifies the level of auditing, enumerates the types of auditable events, and identifies the minimum set of audit-related information provided.
- *Audit analysis*. Provided via automated mechanisms to analyze system activity and audit data in search of security violations.
- *Audit tools*. As available to authorized users to assist in audit data review.

Audit trails, whether computer-based or manually produced, typically form a significant part of the front-line defense for fraud detection and prevention within systems.

- *Event selection*. Inclusion or exclusion of events from the auditable set.
- *Storage*. Creation and maintenance of the secure audit trail.

Although these activities are frequently invoked as an adjunct to enforcement of many other security criteria within the trusted system model, there is little presented in the CC by way of guidelines as to the construction, maintenance, and auditing of such audit information.

Carnegie Mellon's Jon Peha states: "An auditor should be able to retrieve a set of records associated with a given entity and determine that those records contain the truth, the whole truth, and nothing but the truth. There should be a reasonable probability that any attempt to record incorrect, incomplete, or extra information will be detected. Thus, even though many transactions will never be scrutinized, the falsification of records is deterred."[7]

But how is this to be done? For e-commerce, Peha suggests a structure involving customer registration, transaction recording and notarization, protection of notarized records, and transaction auditing.

For transaction auditing, relevant literature reveals that statistical techniques are typically used, accompanied by random sampling and identification of certain trigger events. One of the most noted researchers on this subject is Xerox PARC's Teresa Lunt (formerly of SRI), whose body of work includes extensive surveys of automated audit trail analysis programs [2]. She and her co-authors have methodically and extensively classified the types of events capable of being audited, as well as the mechanisms of audit analysis and risks related to audit avoidance. Lunt urges a combination approach for products, weighing likelihoods and threats against types of subversive activities.

Another useful resource on auditing can be found in the pharmaceutical industry. The enactment of the 1997 U.S. Food and Drug Administration (FDA) regulation 21 CFR Part 11 on the criteria for acceptance of electronic records and signatures, has had a major impact on the collection and processing of FDA-relevant information. Such information tracking is especially critical within the new drug approval cycle. Labcompliance [4], an independent contracting firm, summarizes the primary requirements of the FDA regulation on analytical laboratories as follows:

- Use of validated equipment and computer systems;
- Secure retention of records for instant analysis reconstruction;
- User-independent, computer-generated, time-stamped audit trails;
- System and data security, data integrity, and confidentiality through limited authorized system access; and
- Use of secure electronic signatures for closed and open systems, and digital signatures for open systems.

Such clear specifications are atypical within other domains. For example, consider voting. The section on audit capacity in the "Help America Vote Act of 2002" by the U.S. Congress reads:

"A) In General: The voting system shall produce a record with an audit capacity for such system. B) Manual Audit Capacity: i) The voting system shall produce a permanent paper record with a manual audit capacity for such system. ii) The voting system shall provide the voter with an opportunity to change the ballot or correct any error before the permanent paper record is produced. iii) The paper record produced under subparagraph (A) shall be available as an official record for any recount conducted with respect to any election in which the system is used."[8]

Those of us (including myself) who vigorously testified about the need for assurable voting systems in the election reform hearings were pleased that audit capacity was ultimately mentioned in the final version of the bill. Unfortunately, the wording used is troublingly similar to laws enacted earlier in California and Florida, where subsequently the states allowed systems to be purchased that deliberately precluded voters from participation in the manual

Neither the local election officials, the candidates, nor the voters are provided with any way of validating whether the equipment that recorded the ballots was operating properly throughout the election.

auditing process. Using a subtle interpretation, vendors have chosen to implement the law by producing the required audit trail on paper, from electronic records stored within the computerized voting system, after the closing of the polls.

Since these same vendors have also protected their products under restrictive trade-secret agreements with the counties that purchased them, the integrity of the audit trail generated by the computer cannot be ascertained [6]. Neither the local election officials, the candidates, nor the voters are provided with any way of validating whether the equipment that recorded the ballots was operating properly throughout the election. Nor are the voters given any opportunity to self-verify that the post hoc printed ballots accurately represent the votes they intended to cast. Any independent examination of the voting system for correctness would have to be performed under court order, and to date, the courts have refused to grant such investigations. The intention of the Federal Act is to provide a voter-verified, human-readable, physical recount mechanism for use in elections, but it will be left to be seen whether the subsequent enforcing regulations (not yet developed) will be stringent enough to carry out the spirit of the legislation.

Such flagrant and misguided behavior on the part of developers will continue unless and until the computer industry recognizes the need for external examination of its auditing systems. Currently, NIST performs some of these functions, but the secure computer products required to be certified under NIST's auspices are extremely few in comparison to those that employ auditing components in general use.

In the wake of the Enron debacle, Harvard Business School professor Jay Lorsch called for such oversight within the accounting community [5]. Lorsch believes corporate management should be held accountable for their companies' auditing practices by a body of enforcement auditors who must have no undisclosed ties; they should be rotated every few years, and would be precluded from hiring by an audited firm for a period of three years following any review. This model for the creation of an independent, self-regulatory agency, with rule-making, supervisory, and disciplinary powers similar to those of the stock exchanges, could perhaps be applied across the computer field as well.

The computer industry certainly suffered a significant setback by the backlash of the corporate accounting audit crisis. Similarly, deficiencies in electronic auditing systems may bring serious repercussions on a broad scale.

Therefore, it is incumbent upon us to examine our own auditing practices for their intrinsic vulnerabilities, and to deploy corrective controls lest we (and others) fall victim to our neglected auditing loopholes. C

References

1. Allinson, C. Audit trails in evidence—A Queensland case study. *J. Info. Law Tech.* (Mar. 22, 2002); elj.warwick.ac.uk/jilt/02-1/allinson.html.
2. Anderson, D., Lunt, T., Javitz, H., Tamaru, A., and Valdes, A. Detecting unusual program behavior using the statistical component of the next-generation intrusion detection expert system (NIDES). SRI Technical Report, SRI-CSL-95-06, 1995; www.sdl.sri.com/papers/352/.
3. Common Criteria Implementation Board. Common criteria for information security evaluation. Version 2.1, 1999; csrc.nist.gov/cc.
4. Labcompliance. Electronic records and signatures. FDA's 21 CFR Part 11; www.labcompliance.com/e-signatures/overview.htm.
5. Lorsch, J. A cure for Enron-style audit failures. *H. Bus. Sch. W.*, May 13, 2002 (reprint from *Financial Times*).
6. Mercuri, R. Electronic vote tabulation: Checks & balances. Ph.D. Dissertation, University of Pennsylvania, 2001.
7. Peha, J.M. Electronic commerce with verifiable audit trails. In *Proceedings of ISOC*, 1999; www.isoc.org/isoc/conferences/inet/99/proceedings/1h/1h_1.htm.
8. U.S. Congress. Help America Vote Act of 2002, conference version of draft bill, October, 2002; www.acm.org/usacm/Legislation/ElectionReformConference.pdf.

Rebecca T. Mercuri (mercuri@acm.org) is a professor of computer science at Bryn Mawr College and the president of Notable Software, Inc., Princeton, NJ.