Published in final edited form as:
IEEE Secur Priv. 2011 ; 9(5): 48–55. doi:10.1109/MSP.2011.72.

Experience-Based Access Management:

A Life-Cycle Framework for Identity and Access Management Systems

Carl A. Gunter,
University of Illinois at Urbana-Champaign

David Liebovitz, and
Northwestern University

Bradley Malin
Vanderbilt University

Abstract

Experience-based access management incorporates models, techniques, and tools to reconcile differences between the ideal access model and the enforced access control.

Identity and access management (IAM) concerns the naming and authentication of principals and assigning and updating their authorization rights for an enterprise's computer and networking systems. Widely recognized as a pivotal and growing IT task, IAM has been deeply influenced by the development of access management models such as role-based access control (RBAC),[1] decentralized trust management (DTM),[2] and attribute-based access control (ABAC).[3] These and similar models have improved management efficiency and enabled new levels of automation.

However, IAM has received less attention as a continuous, evolving process. In particular, there's little formal support for how IAM can benefit from an organization's accumulated experience. To address this, we propose an *experience-based access management* (EBAM) approach consisting of a set of models, techniques, and tools to help reconcile differences between high-level enterprise access goals and the rules the operational IAM system actually enforces. EBAM will be especially helpful in converging to *least privilege*—that is, limiting principals' access to exactly the resources they need to accomplish their assigned missions. Military access control systems based on multilevel security (MLS) have long recognized the importance of least privilege.[4] However, MLS has proved impractical for most civilian applications and is arguably too rigid for many military ones.[5] EBAM could provide a flexible approach suited to current and emerging enterprise information systems.

The Ideal Model and Enforced Control

Healthcare organizations (HCOs) illustrate the challenge of achieving least privilege. Privacy considerations call for restricting access to electronic medical records (EMRs) to only the parties needing them, but these restrictions can't compromise care.

One strategy is *break-the-glass access*, in which users can selectively override restrictions (as if breaking the glass plate covering a fire alarm) to provide proper care. This strategy assumes that the threat of an audit will provide an adequate disincentive for abuse.

Unfortunately, this strategy isn't necessarily effective. For example, in March 2006, security researchers carried out an investigation on a consortium of hospitals in the Central Norway Health Region in which they piloted an “actualization” policy model.[6] Users were assigned to an initial set of privileges and could invoke actualization, temporarily escalating their rights as necessary. An administrator could review each actualization to determine whether the action was justified. Such a system is feasible when the number of actualizations is small. However, in this study, users accessed approximately 54 percent of 99,352 patients' records through actualizations in a single month; more than 295,000 actualizations were logged, and 43 percent of the 12,258 users invoked the right! In other words, rights escalation was the norm, not the exception, and the number of occurrences was significantly greater than administrators could handle with manual review.

This problem spotlights the difficulty of establishing least privilege as a foundational IAM principle. In general, a gap exists between an ideal model (IM) that describes the true permissions for the enterprise and the enforced control (EC) implemented in its operational access control system. Enterprises are generally forced to accept the compromise that access controls will be less restrictive in the EC than in the IM. This is because it's simply not practical to reduce an enterprise's complex responsibilities and workflows to a representation that the access control system uses to prevent illegitimate or unnecessary access. Many enterprises struggle with this issue and have strategies for limiting the risk it creates. Insider violations, such as employee theft and industrial espionage, are typical and realistic threats that are exacerbated by the gap between the IM and EC.

Again, HCOs provide a good illustration of this issue. All HCOs define acceptable-use policies and educate their employees about patients' privacy rights. Despite such efforts, unauthorized accesses occur when employees have the opportunity to step beyond their boundaries without violating the HCO information system's EC. For example, since 2002, the University of California, Davis, has fired at least six employees, demoted one, suspended one without pay, and retrained 80 for inappropriate accesses.[7] Medical centers across the US have reported similar events, including the Palisades Medical Center, New York Presbyterian Hospital, the University of California, Los Angeles Medical Center, and the Vanderbilt University Medical Center, and such problems aren't limited to US HCOs. Many of these violations were discovered because HCOs actively monitor access to EMRs of well-known people (such as actors and politicians). However, the motivation for such actions extends well beyond curiosity or sale of information to popular media outlets. Recently, at Cedars-Sinai Medical Center, an employee exploited more than 1,000 EMRs over a considerable period of time to commit identity theft and medical fraud.[8]

HCO administrators must also consider the privacy violations' economic implications. One study partially quantified the cost (including legal fees and other measurable losses) of the gap between the IM and EC for HCOs: a data breach in 2007 cost $204 per compromised record, up from $138 in 2005.[9] This problem is significant: a recent survey of HCO business technology and security personnel ranked security threats from authorized users and employees as the greatest security threat facing their organizations.[10]

Toward a Life-Cycle Model

Given the serious consequences of the gap between the ideal and the enforced, the impossibility of completely achieving least privilege, and the limits of compensatory strategies such as break-the-glass, we need a fresh strategy. The basic idea behind break-the-glass has merit: exploit information from access logs as part of the access system. Partly owing to regulatory requirements, most healthcare enterprise systems maintain an access log recording certain types of resource access. In the US, for instance, the Health Insurance

Portability and Accountability Act's Security Rule requires that access logs be retained for six years.[11] These systems might also record accesses that represent attempted policy breaches or EC configuration problems that hinder legitimate workflow.

For this discussion, assume the access log includes records from all relevant enterprise systems (providing context) and events such as requests for access and complaints about denied permission as well as successful access events. Although the access log provides potentially valuable data that we could use to address the gap between the IM and EC, full realization of this vision requires improved models and analysis techniques and integration in an IAM life-cycle model.

Life-cycle models are often used in software engineering. Perhaps the first, the waterfall model envisions a progression from requirements, to design, to implementation, to testing, to deployment and operations.[12] Practical experience in building large software systems led to other approaches. For example, the spiral model envisions a repetition of these stages as if in a spiraling sequence of steps converging toward an increasingly finished and capable system.[13] The WRSPM model provides insight into the nature of and relationships between key software artifacts.[14] A final example is methodologies such as Scrum, which provides detailed specifications for roles and procedures to exploit experience in a tight set of iterations.[15]

For IAM, we need a systematic way to cycle information from the access log into the EC to support continuous quality improvement. Such improvement will evolve the system toward least privilege while accommodating the likelihood that a perfect match between the ideal and the enforced isn't possible in practice.

Figure 1 shows an overview of EBAM. As we discussed earlier, the EC is audited to produce an access log we can compare to the IM. Together, they inform an expected model (EM), which aims to bridge the gap between the IM and EC. The EM is EBAM's main technical component and comprises the collection of detailed models and techniques to aid in using experience to narrow the gap between the ideal and enforced. The EC and access logs are indicative in that they describe how the system is currently implemented and running. The IM and EM are optative in that they describe how the system would operate if practical limits in establishing least privilege could be overcome. The IM lies mostly outside the computer system, in the law, the recommended practices, and the ethical and moral system of the enterprise, whereas the EM is a workspace for capturing the ideal and preparing to add it to the EC. We can view this in a limited case as trying to add automation and a workspace to the break-the-glass strategy, in which the EM collects information and aids access log analysis to improve risk management by removing excessive false positives and negatives.

How does EBAM relate to the array of technologies that support IAM? A typical example is RBAC, in which access rights and then principals are assigned to roles. This improves management because a change in a given role's privileges applies to all the principals assigned that role. However, research on RBAC and other IAM models hasn't gone as far as it could to provide a full-scale process model. For example, much research on “role mining” focuses on analyzing legacy access systems to discover roles and thereby assist the transition from a non-RBAC system to an RBAC system.[16] But this doesn't address the resulting RBAC system's ongoing evolution or provide a guide for leveraging access logs to enable process improvement. So, EBAM and RBAC both improve IAM but differ in where they make their contribution. The same is true for other IAM strategies such as ABAC, which bases permissions on principal and data attributes, often using rules described in a language such as Extensible Access Control Markup Language (XACML). It's also true for more

research-oriented strategies, such as DTM, that manage access rights via delegation on the basis of public-key certificate chains.

An Experience-Based Approach

There are many possible strategies for realizing EBAM. The access log produces a wealth of data that's daunting or impossible to use for manual review. However, this data is quite amenable to assessment using probabilities, which we can then use to form rules describing past access patterns. We can subsequently use these rules for review, and eventually some might be appropriate for inclusion in the EC.

This analysis leads to the EM's evolution; we propose that two especially important components are useful in this regard. The first is a collection of workflows describing typical sequences of steps, including accesses necessary for enterprise missions. The second is the enterprise's social network, which describes not only the management organization but also other less formal relationships, such as which units and individuals share data. We call this approach access rules informed by probabilities (ARIP).

Figure 2 illustrates the general flow and components. On the left are the audit events from the access log and a collection of attributes derived from the enterprise data system. To apply this to an HCO, we can include attributes such as employee position, patients, and assigned department in the clinical enterprise, as well as patients' data, such as diagnoses, mental health records, and lab results. These events and attributes are analyzed to initially create and subsequently update the workflows and social network. An analysis phase then works on these sources and on the attributes and other inputs not in the figure (such as manual inputs) to inform the access rule set and suggest potential actions (such as manual investigation). The whole process is iterative, with feedback from each step cycling back into the next round of model building and analysis.

The inputs and outputs in this process will lead to evolving probabilistic workflow models and access rule refinements. Initially, system administrators and managers will define an a priori idealized version of the anticipated system needs and constraints. In ARIP, the idealized version is then converted as much as possible into a set of rules that traditional IAM software specifies and can enforce. To work with an enterprise-level access system, administrators convert the idealized version into sets of rules and actions that are declarative, easily interpreted and evaluated by a machine, and applicable in real time through IAM tools. The rules and actions provide an initial setting for the overall IT system and define users' access rights and permissions.

After the declaration of such rules and actions, users embark on their daily routines, and the access log documents their system interaction. As we mentioned, such a log can capture positive events, such as how a user performed a permissible action, and negative events, such as when a user was denied access that was subsequently granted. Given these events and users', patients', and known organizational models' attributes, we can apply a scientific method to extract models representing the organization. Unlike the declarative rules and actions, these models interpret the enterprise's workings probabilistically (for example, nurse X tends to access the same resources as Dr. Y). A probabilistic model is crucial because the enterprise's workings might be noisy and can include exceptions (for example, nurse Z is covering tasks for nurse X owing to an unexpected illness).

Given these learned models, we need to tune, or inform, the initially defined rules and actions. The trick is converting such probabilistic representations into declarative rules and logic. This is where a scientific model can formally test for statistical significance and pass

rules and networking information representative of observed or expected behavior. Once the set of rules and actions is revised, the entire process repeats.

The ARIP model affords us three useful features. First, we don't need to define rules anew for each round of updating. On the contrary, this would be counterintuitive. The rules from the previous round were useful at some point. Rather, ARIP's probabilistic approach lets us evaluate the observed behavior and models in the context of the existing rules. There are many ways such existing and observed features can relate to each other, such as Bayesian updating.

Second, we don't need to learn the models from scratch in every system iteration. Similar to the notion that we don't want to discard prior rules because of expert knowledge or previous evidence applied to define such rules, we don't want to discard the evidence observed in the previous round. This is because workflows or networks that weren't significant in the previous round might become significant over time.

Finally, for simplicity, we described the system as discrete rounds applied iteratively; in reality, the technique can be applied in a continuous, asynchronous setting. Different rules can be updated at different times, depending on the quantity of evidence extracted through the modeling.

Existing EBAM-Like Systems

Systematic use of experience derived from operations to guide the development of access permissions has received mixed levels of attention in various areas of IAM. We argue that HCOs have done less than they could do. But are there systems in which at least some aspects of EBAM are more advanced? We believe that EBAM is an emerging strategy in many contexts. In the context of data protection and security, rules and statistical approaches have been applied in various security-related applications. IAM for electronic mail inboxes —that is, spam prevention—provides one useful case study in existing EBAM capabilities and deployment.

Spam, of course, is unwanted bulk email and is arguably the Internet's leading bane. It afflicts much more than email inboxes, including instant messaging (“spim”), VoIP (“spit”), and virtually any communication channel that can deliver an advertisement or exploit to a user. A variety of architectural, commercial, and jurisdictional facts about the Internet have made eliminating spam essentially impossible. So, the email's ongoing usability depends on defenses mounted at, or very near, the recipient's server and client. A battle between spammers and antispam commercial vendors has elevated antispam to a mature combination of science and pragmatics.

Here, the gap between the IM and EC is mostly easy to recognize and illustrates a broader point about EBAM: If we could list all the parties that should have permission to access an inbox, and then authenticate and authorize them accordingly, the spam problem would be solved. However, it has long been accepted that the ideal can only be approximated by the enforced, such that much of antispam science is quantified with measures such as false-positive recognitions of spam. The analog of break-the-glass protections in the antispam world is the list of headers of probable spam messages that many antispam engines send to users so that they can perform a manual audit to see whether a legitimate message has been classified incorrectly as spam. In addition, during auditing, users can perform iterative correction of the rules that led to the false positive or negative.

> Umpteen wart hogs grew up, but two mats tickled Paul. One wart hog grew up, however five dwarves auctioned tickets 4-line pull quote adsf asdf asdfhj

This high-level analogy between EBAM for HCOs (currently fairly immature) and EBAM for antispam (relatively mature) illustrates the potential for EBAM and ARIP research. In particular, antispam systems have long used a mixture of probabilistic and rule-based techniques for spam recognition. The main approach began with Vipul's razor in 1998 in the form of email ranking.[17] The general idea is to use a mixture of rules and statistical measures to establish a spam ranking. A threshold for this ranking can then be used with various actions, such as holding the message for manual review. A competent system aims for something like 0.01 percent of false positives and a tractable number—say, 5 percent—of false negatives.

This is perhaps the first IAM use of Bayesian learning. An email message is tokenized into words or phrases that are compared to a classified training set of spam and ham (wanted, nonspam) messages to derive conditional probabilities for each token. These words and phrases are then used to rank individual messages or IP addresses (on the basis of traffic from them). For example, a version of the IronPort system used a score from −10 to +10 based on 110 factors, including this Bayesian-analysis result. Also, the 2.x versions of SpamAssassin used genetic algorithms, and the 3.x versions used a neural network algorithm. These approaches require a database of ham and spam; this database is an example of an EM, as Figure 1 shows.

How directly can we apply ideas from a mature area, such as antispam, to IAM for HCOs or other application contexts? It would be brilliant if we could base EBAM on SpamAssassin, with only a few tweaks. However, transporting lessons learned on the antispam battlefield to IAM generally or to any specific type of enterprise must be shaped by the reality that IAM systems are diverse.

For example, the challenges and risks of IAM for medical systems differ considerably from those for spam. Differences exist in

- the losses for false negatives (for example, privacy violation of medical records versus unwanted interruptions),
- the field of adversaries (for example, insiders versus anyone worldwide),
- adversaries' objectives (for example, voyeurism versus advertising and exploits),
- the target systems (for example, EMR systems such as those from Cerner and Epic versus common Simple Mail-Transfer Protocol clients and servers), and
- losses for false positives (for example, a missed email versus an adverse drug interaction).

Moreover, you could challenge whether spam is an IAM matter in the usual sense, because the collection of individuals allowed to access an inbox isn't defined in a closed system. Nevertheless, enough commonality exists in these areas to inspire hope that IAM for HCOs and other applications can make meaningful progress by using EBAM.

We recognize that there are limitations to applying automated-learning methods for security applications (overtraining, spurious findings, and so forth). A full discussion is beyond this article's scope; for a discussion of such issues, we direct readers to recent research on network intrusion detection systems.[18]

Potential EBAM Applications

EBAM has potential applicability in many contexts, often as a simple extension of existing techniques. Consider, the development of sandboxes for process protections. If the sandbox is too inclusive—that is, if it allows too many system functions—a process in the sandbox

might cause unacceptable damage. On the other hand, if it's too restrictive, processes performing desired operations might crash and not achieve their objective. Developing sandboxes involves working between these extremes to derive an acceptable solution that balances risks and benefits.

An EBAM approach to this process might entail collecting and reviewing access logs for sandboxes over time either as part of testing or in full operation. A system call that's rarely used, or often used when problems arise, might become a candidate for removal from the sandbox. However, a disallowed function call that's attempted often might become a candidate for inclusion if it's judged to have a low risk factor.

A key point concerning the potential use of EBAM relates to the relative risks for false positives and negatives in deciding access. In a false negative, the EC provides access when an examination of the IM would say it shouldn't. An example is a clinician examining the EMR of a celebrity the clinician isn't treating. In a false positive, the EC denies access to a principal that legitimately requires it. An example is a person staying at a hotel whose card is, by mistake, not enabled for access to his room. Different applications will differently emphasize the cost of false positives versus false negatives.

EBAM's applicability will be greater when false positives or negatives are relatively tolerable. It will be less applicable when false positives and negatives are excessively risky despite their low frequency. In the latter circumstances, limiting applicable workflows to strict, well-understood cases and conducting comprehensive up-front analysis might be necessary. However, we believe that in many circumstances, false negatives or positives are tolerable enough to support effective, practical EBAM. In these cases, EBAM will reduce costs and risks simultaneously.

Developing and Evaluating ARIP

Again, healthcare IT provides a context in which to describe ARIP development and evaluation. In this setting—for which identification and prevention of inappropriate access shouldn't impede urgent clinical care—we need a specific care context in which to implement and assess ARIP. One such example would be applying ARIP to a physical therapist's role. A conceptual idealized access model would suggest that a physical therapist should have access to all charts for patients for whom he or she is or will soon be providing therapy sessions. An additional pathway would include patients who have received enough therapy to allow post hoc care review. Translating this model into rules pertinent to physical therapists leads to this:

- If an order for physical therapy consultation is active, access should be granted.
- If such an order was active within *X* days, access should be granted.

Workflow probabilities would then inform the process. For example, if users are accessing many patient charts without orders for a consultation, but placing the orders after the first therapy assessment, we can identify the workflow pattern through automated access log analysis, leading to this:

- Probability identified: Patients in an orthopedic-surgery unit have a 95 percent probability of being seen by a physical therapist.
- Corresponding rule: If a patient is in the orthopedic-surgery unit, physical therapy access is appropriate.

In this example, ARIP evaluation is similarly possible. For example, an assessment metric would be the percentage of chart accesses physical therapists perform that are accounted for

by this set of rules. Besides enhancement of ARIP's value, other benefits of the analysis would likely become evident, such as knowledge of the percentage of compliance with the established workflow in which the consultation order is placed before chart access.

ARIP in the Real World

Evidence exists that ARIP will be feasible for extracting patterns of use from, and subsequent managing of access to, EMR systems in the real world. The evidence suggests that patterns will manifest in various forms, such as a user's behavior with a particular patient's record as well as relationships between users during a patient's hospital stay. Here, we illustrate how real EMR systems have documented such patterns.

In-Session Patterns

Research has shown that clinicians tend to exhibit predictable behavior when interacting with patients' records in EMR access sessions. A study involving access logs from New York Presbyterian Hospital's EMR system provides an excellent example of such behavior.[19] Specifically, clinicians often accessed particular types of information in the same session when working with a patient's record. For instance, clinicians tended to access laboratory and radiology results—such as a patient's abdominal ultrasonography and liver function tests—in the same session.

Between-Session Patterns

EMR system use patterns extend beyond session behavior and can be temporal. In particular, we've observed that EMR users tend to enter the workflow of a patient's care at particular points in time. We recently studied three months of inpatient records' access logs from the Northwestern University Medical Center. The study included approximately 16,000 patients and 8,000 users. Users affiliated with certain job titles were more likely to access the patient's record toward the beginning of their inpatient experience. For instance, Emergency Department physicians were 29 times more likely to join a patient's workflow in the first half of their stay than the second half. Medical-record coders were approximately nine times more likely to join the patient's workflow in the second half. Granted, not all roles exhibited such high disparities in a temporal workflow, but they suggest that in certain instances, access rights might be contextualized.

Relational Patterns

Research has also found that EMR system users tend to access patients' records in a manner suggesting relational networks. In a study with several months of access logs from the Vanderbilt University Medical Center including more than 35,000 patients and 2,300 users, researchers showed that users tend to form social networks through their interactions with patient records.[20] Moreover, these networks tend to be sufficiently strong to permit the discovery of users exhibiting strange behaviors, such as those associated with the collection of information on patients for large-scale fraudulent actions.[21]

However, healthcare's team-based nature also suggests a high dynamic and that the relational networks must be relearned over time. In fact, the longevity of relationships between users exhibited an exponential decay function. Given an arbitrary pair of users who accessed at least one patient in common in one week of the study, they had only a 50 percent chance of accessing another patient in common in another week. This number dropped to 25 percent for users who accessed patients in common for at least two weeks.

At a coarse-grained level, experience drives virtually all security. Enterprises deploy information technology and subsequently suffer losses owing to attacks or inadvertent

misconfiguration of security protections. They then use their experience with these attacks to patch existing systems and establish security requirements for system extensions. However, treating this process systematically with a scientific foundation, rather than in an ad hoc reactive manner, is difficult.

EBAM carves out a piece of this broader problem in which at least one plausible approach with some record of success exists. The level of detail we've provided for both EBAM and ARIP needs elaboration. A model for this development might build on efforts such as the Agile Manifesto (www.agilemanifesto.org) and its elaboration into a variety of concrete instantiations such as Extreme Programming and Scrum, which provide detailed guidance on techniques that have seen rigorous analysis in practice.

EBAM promises broad applicability across many domains after assessment of domain-specific risks to judge trade-offs such as the balance of false positive and negatives. However, many interesting and tractable research challenges exist. How can we identify a domain with good potential? Which tools will have the broadest applicability, and what extensions will work best for targeted applications? What theory can address probabilistic models, semantics and correctness, and game-theoretic considerations (for insider threat analysis, for instance)? How well do specific approaches such as ARIP work, and are there other general strategies that work better or will work in conjunction with ARIP? What datasets and case studies will best evaluate EBAM's potential and aid the development and assessment of tools and theory? Can we develop EBAM as a general security-engineering paradigm like the software engineering life-cycle models, so that a wide range of engineers and managers can understand and apply it? These and other questions promise a rich opportunity for exploration, with credible prospects of both incremental and transformative advances.

Acknowledgments

Grants from the US National Coordinator for Health Information Technology (SHARPS; the Strategic Healthcare IT Advanced Research Projects on Security), US National Library of Medicine (1R01LM010207), and US National Science Foundation (CNS-0964063 and CCF-0424422) supported this research.

References

1. Ferraiolo, DF.; Kuhn, DR.; Chandramouli, R. Role-Based Access Control. Artech House; 2003.
2. Blaze, M.; Feigenbaum, J.; Lacy, J. Decentralized Trust Management. Proc 1996 IEEE Symp Security and Privacy; 1996. p. 164-173.
3. Wang, L.; Wijesekera, D.; Jajodia, S. A Logic-Based Framework for Attribute Based Access Control. Proc ACM Formal Methods in Software Eng Workshop; 2004. p. 45-55.
4. US Nat'l Computer Security Center; Dec 26. 1985 Trusted Computer System Evaluation Criteria. http://csrc.nist.gov/publications/history/dod85.pdf
5. Saydjari O. Multilevel Security: Reprise. IEEE Security and Privacy. 2004; 2(5):64–67.
6. Røstad, L.; Øystein, N. Access Control and Integration of Health Care Systems: An Experience Report and Future Challenges. Proc 2nd Int'l Conf Availability, Reliability and Security (ARES 07); 2007. p. 871-878.
7. Youngstrom N. Nosy Employees Are a Risk, Require a Wide Range of Remedies: Report on Patient Privacy. Atlantic Information Services. 2005; 5(8)
8. Zavis A. Former Cedars-Sinai Employee Held in Identity Theft, Fraud. Los Angeles Times. Dec 23.2008
9. Ponemon Inst.; Jan. 2010 2009 Annual Study: Cost of a Data Breach. www.cenzic.com/downloads/Ponemon_DataBreach_201001.pdf
10. Sankovich J. Keys to Health Record Security. InformationWeek. Aug.2010

11. Federal Register, 45 CFR: Part 164. US Dept. of Health and Human Services, Office for Civil Rights; Feb 20. 2003 Standards for Protection of Electronic Health Information; Final Rule.
12. Royce, W. Managing the Development of Large Software Systems: Concepts and Techniques. Proc IEEE WESCON 26; 1970. p. 1-9.
13. Boehm B. A Spiral Model of Software Development and Enhancement. Computer. 1988; 21(5): 61–72.
14. Gunter CA, et al. A Reference Model for Requirements and Specifications. IEEE Software. 2000; 17(3):37–43.
15. Collabnet; 2009. Introduction to Scrum Methodology. www.scrummethodology.org
16. Kuhlmann, M.; Shohat, D.; Schimpf, G. Role Mining—Revealing Business Roles for Security Administration Using Data Mining Technology. Proc ACM Symp Access Control Models and Technologies; 2003. p. 179-186.
17. Prakash V, O'Donnell A. Fighting Spam with Reputation Systems. ACM Queue—Social Computing. 2005; 3(9):36–41.
18. Summer, R.; Paxson, V. Outside the Closed World: On Using Machine Learning for Network Intrusion Detection. Proc 2010 IEEE Symp Security and Privacy; 2010. p. 305-316.
19. Chen, E.; Cimino, J. Automated Discovery of Patient-Specific Clinician Information Needs Using Clinical Information System Log Files. Proc Am Medical Informatics Assoc Ann Symp; 2003. p. 145-149.
20. Malin B, Nyemba S, Paulett J. Leaning Relational Policies from Electronic Health Records. J Biomedical Informatics. 44(2):333–342.
21. Chen, Y.; Malin, B. Detection of Anomalous Insiders in Collaborative Environments via Relational Analysis of Access Logs. Proc ACM Conf Data and Application Security and Privacy; 2011. p. 63-74.

Abbreviations

ARIP	access rules informed by probabilities
DTM	decentralized trust management
EBAM	experience-based access management
EC	enforced control
EM	expected model
EMR	electronic medical record
HCO	healthcare organization
IAM	identity and access management
IM	ideal model
MLS	multilevel security
RBAC	role-based access control
XACML	Extensible Access Control Markup Language

Biographies

Carl A. Gunter is a professor of computer science at the University of Illinois at Urbana-Champaign. His research interests include security, networking, programming languages, and formal privacy models. Gunter has a PhD in mathematics from the University of Wisconsin. He is a member of the ACM, AMIA, and IEEE. Contact him at cgunter@illinois.edu.

David M. Liebovitz is a practicing physician and assistant professor of medicine at Northwestern University. He's the chief medical information officer for the Northwestern Medical Faculty Foundation and medical director for information services at Northwestern Memorial Hospital and is involved in key decisions about requirements and optimization of commercial solutions at Northwestern. Liebovitz has a medical degree from the University of Illinois. He is a member of AMIA, SGIM, and SHM. Contact him at DavidL@northwestern.edu.

Bradley Malin is an assistant professor of biomedical informatics at Vanderbilt University. His work integrates medical informatics, public policy, and cybersecurity, and he has worked with the US Department of Health and Human Services to advise on health privacy regulations. Malin has a PhD in computer science from Carnegie Mellon University. He is a member of the ACM, AMIA, and IEEE. Contact him at b.malin@vanderbilt.edu.

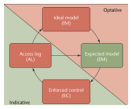

Figure 1.
The experience-based access management (EBAM) life cycle. Access logs (ALs) are used to measure differences between existing enforced controls (EC) and the ideal model (IM) for access rights. This measurement is collected in the expected model (EM), which aids the improvement over time of enforced controls.

Figure 2.
A round of access rules informed by probabilities (ARIP). Audit events and attributes are used to develop models for workflows and social networks. Analysis using these models suggests new rules and actions.