Good-Enough Security

Toward a Pragmatic Business-Driven Discipline

Ravi Sandhu
George Mason University and SingleSignOn.net

This issue of *IEEE Internet Computing* launches a new security "track" in the magazine. By means of this track, readers can look forward to at least one article on security in each issue. As one of the old-timers in the security business, this is an exciting development for me, as well as for the magazine and the security community at large. It is one of many signs that security is finally coming of age.

When I started in the profession with a fresh PhD in 1983, security was at best a peripheral niche concern. Today, it is a major concern at the top corporate, government, and academic levels, and security problems in cyberspace are unlikely to disappear or be solved any time soon. Indeed, new problems and requirements are likely to emerge, and we can anticipate continued interest in the field. I hope this track will contribute to that growth.

User-Friendly Security

I am bullish on the security discipline's future, but I am equally convinced that security practice and research must change in fundamental ways. We all recognize that security is intrusive and impedes our ability to get work done. Some of this can be blamed on poor design: we often make the problem worse than it needs to be. One of my favorite examples of this tendency is the pressure on users to frequently change their passwords. Many organizations require them to do so monthly, although I am not aware of any empirical study that proves the practice to be beneficial. In fact, I do not even know any anecdotal evidence to that end. On the contrary, many anecdotes indicate that users cope with such regimes in a way that actually degrades security.

Some users append the month name or a numeral to the password, for example, making the password xyzJan, xyzFeb, and so on, instead of xyz. Several characters of the password are thus devoted to the rotation algorithm with a loss of overall entropy. Other users toggle back and forth between two or three passwords. I am amazed when organizations devote resources to preventing these simple rotation algorithms, making it a battle between security administrators and users, rather than addressing the fundamental problem in the first place.

It would be friendlier to enforce password complexity rules and deploy password technology that is resistant to various kinds of online and offline guessing attacks, instead of battling users with an unproven approach. It would be less intrusive and more effective to deploy fraud-management and intrusion-detection techniques to recognize possible password theft and misuse.

 Published by the IEEE Computer Society 1089-7801/03/$17.00©2003 IEEE

Business-Driven Security

Intrusiveness is bad, but intrusiveness with unproven benefit is even worse. However, even with very good and careful design, security will inevitably have an element of intrusiveness. Practitioners must balance this inherent feature of security against other equally important goals, such as ease of use and total cost of ownership. Security products must be designed as consumer products. In the past, businesses could impose intrusive measures on their employees, but security must be increasingly user friendly as organizations reach out to engage partners and consumers in cyberspace.

In addition to end users, we must also consider the operators of the security infrastructure. Cumbersome technology will be deployed and operated incorrectly and insecurely, or perhaps not at all. Like many other IT products, operational rather than technological costs often dominate the total cost of ownership for security products. Many organizations have found that the cost of password resets — when users forget their passwords — is a major component of the authentication system's total cost. Hence the recent popularity of products that automate password resets instead of requiring a costly manual help desk intervention.

This brings me back to the words in this article's title: "good enough" and "business driven." I believe three golden principles guide information security:

- Good enough is good enough.
- Good enough always beats perfect.
- The really hard part is determining what is good enough.

The first principle is a vacuous tautology, but one that the technical security community (myself included) forgets too easily. The second principle is amply supported by strong empirical evidence in all aspects of information technology. Its application to our field is further amplified because there is no such thing as "perfect" in security. We might thus restate it as the nearly tautological, "Good enough always beats 'better but imperfect'."

The third principle tells us where the difficulty resides: We are completely clueless about what is good enough. That is the rub. Business people cannot tell us because they don't understand security, and security people cannot tell us because they don't understand business. We must close this divide to further our profession and make it "business driven."

The Quest

Albert Einstein once said, "Everything should be made as simple as possible, but not simpler." I suggest the following adaptation for the information security business: "Everything should be made as secure as necessary, but not securer." This is the essence of good-enough information security.

Some might question whether this is a reasonable quest, but I submit that it is. We already have at least one network with good-enough security in the system of automatic teller machines (ATMs). The scale is worldwide. The fraud is nonzero but tolerable, and there is a constant battle to contain it. As consumers we are quite comfortable using ATMs, and we rarely know first-hand of electronic fraud with them. Most of us would say that the ATM system is pretty secure while providing a very useful service.

To see what we can learn from this success story, let's try a thought experiment: How many security engineers would it take to design a system for ATM security today? I don't think it could be done. We would be debating biometric-enabled smart cards, assurance, protection profiles, denial of service, nonrepudiation, viruses, and buffer-overflow attacks until we were blue in the face. There is no way that such a system with good-enough security could be designed and built today on the basis of conventional security wisdom. Yet it happened. And it works.

Is it an anomaly? I don't think so. The GSM phone network has also achieved good-enough security, and it too is worldwide (except for the US and a few other places). The subscriber identity module (SIM) cards deployed in more than 600 million mobile phones — composing a large majority of the mobile handset market — make the system pretty secure.

How about the credit-card network? It shares some characteristics with the ATM network, but its security is a grade or two below. I say this based on personal experience and first-hand anecdotes from friends and acquaintances regarding their experiences with credit-card fraud — an experience that is not replicated with ATM cards. The bottom line remains that good-enough security has already been engineered successfully on global-scale systems that run across many organizations.

Achieving Good-Enough Security

So how do we achieve good-enough security for the Internet? The short answer is: I wish I knew. This is the challenge for the security community in the coming years. For the moment, let me offer two design principles.

Contribute to the Security Track

It is wonderful to be in a hot field like security, but sometimes the excitement and high stakes generates more heat than light. As leading-edge practitioners and researchers, we must thus bring some clarity to fundamental and practical security issues — even as the discipline itself rapidly changes.

We invite contributions of two forms:

- short papers (maximum 3,000 words) that take an informed and focused look at current topics such as denial of service, digital rights management, privacy, and identity management
- longer papers (maximum 5,000 words) that describe new research results or experiences in developing and deploying Internet security technologies.

In either case, submissions should be technical, but lively enough to stimulate discussion. We seek articles that appeal to *IC*'s diverse audience, not just security specialists. See *IC*'s department author guidelines for more information (www.computer.org/internet/dept.htm).

To determine whether your submission is suitable, please send a brief overview of the proposed article to department editor Ravi Sandhu (sandhu@gmu.edu).

Design with the Application in Mind

This idea is so obvious that it must be true. Yet it contradicts conventional security doctrine. Much of the early security research came from the military sector and regarded applications as intrinsically insecure. Researchers thus focused on tightening information flow controls in the operating system kernel. The resulting systems were flawed in two respects. Unauthorized information flow was still possible via so-called covert channels. More significantly, it was impossible to build interesting applications on top of these platforms without granting many exceptions from the underlying OS controls.

The goal of pushing security out of the application was inherently doomed. We see a similar phenomenon in modern packet-filtering firewalls: Punching a hole in the firewall to enable Web browsing also lets other applications use that hole for their own purposes. Applications such as instant-messaging and peer-to-peer collaboration systems can ride in on these holes and leave the firewall rather porous. Simply said, our community cannot ignore applications. We must push security into them while providing a secure infrastructure of services and policies to build on. How else can we deploy security in a large distributed system?

Security is about Trade-offs, not Absolutes

Again, this principle is so obvious that it must be true. But can we make trade-offs without considering the application? Aren't they possible only within some context? Security goals have their own contradictions because confidentiality, integrity, privacy, accountability, and recovery often conflict fundamentally. For example, accountability requires a strong audit trail and end-user authentication, which conflicts with privacy needs for user anonymity. Just as maintaining tight control on a ship's overall weight is infeasible without some knowledge of secret cargo, maintaining information integrity is difficult when confidentiality makes some information unavailable. Intrinsic security trade-offs are hard enough to resolve, but we also need to factor in nonsecurity goals such as cost and ease of use.

Role Models

Many of my esteemed colleagues, particularly from the aerospace community, have recommended that security practitioners draw inspiration from the airline industry. The rigorous software development process for avionics software, maintenance regime, oversight by regulatory bodies, high level of training for pilots and air-traffic controllers, intense investigation of aircraft failures, and so on come together to produce a good-enough system. This is a great success of engineering and business. Yet even here, the events of 11 September 2001 demonstrated basic weaknesses that the community is now addressing. Nevertheless, the airline industry remains a successful role model for us.

As an alternative, I would like to suggest the automobile as a more appropriate role model for the security industry:

- The automobile is a consumer product that drivers operate with minimal, but nonzero training.
- Hundreds of millions of units are deployed all over the world.
- Safety, ease of use, and total cost are all very relevant in automobile design and deployment.

Of course, neither the airplane nor the automobile is a perfect role model for security engineers. We should learn from both industries, but we perhaps have more in common with the latter.

Ravi Sandhu is chief scientist and cofounder of SingleSignOn.Net and professor of information technology and engineering at George Mason University. His research interests include role-based access control, usage control, and identity management. Contact him at sandhu@gmu.edu.