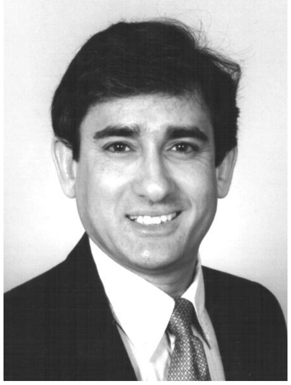

Anand Tripathi
Department of Computer Science
University of Minnesota
Minneapolis, MN 55455
tripathi@cs.umn.edu

Education in information security

Matt Bishop, University of California, Davis

The last four years have seen an explosion in the concern for information security. People are becoming aware of how much information is publicly available, as stories in the national news media discuss the ease with which hackers steal identities. On a less personal note, compromises of information involving authorized access show that organizations have information security problems. With this awareness has grown an understanding of our dependence on accurate, confidential information, and of the fragility of the infrastructure we use to secure that information. Of all the questions emerging, the fundamental one is this: How can we secure information? This essay discusses different forms of education relevant to the problem.

Public awareness

The most basic form of education is public awareness. Does the public understand that a problem exists? How can we communicate the depth of the problem effectively? The public does not want to know details or technologies; it wants to know how to keep private information private, from government entities as well as commercial and academic ones. So, education at this level is primarily procedural and should focus on making the public aware of the threats and what individuals can do to protect themselves.

For example, many people are connecting to the Internet using Digital Subscriber Line technology. The marketing literature touts the benefits of speed and Internet accessibility. The obvious conclusion, one that the members of the public do *not* make, is that DSL connects you to the Internet at all times, except when your modem (or system) is turned off. This approach broadens the interval in which attackers can probe your machine and increases your exposure to attack. You can ameliorate these risks somewhat simply by turning your modem off when you are not using your Internet connection. The public needs to learn this precaution; the reason for it is not important. However, if someone asks why, you should *always* explain it, by the principle of open design.[1] (For an explanation of open design, see the sidebar, "Saltzer's and Schroeder's principles of secure design.")

Academic education

Academic education addresses the problem more deeply than public-awareness measures do. The types of academic education, broadly stated, are training, undergraduate education, terminal master's education, and doctoral education. The differences among these types of education are illuminating.

Training

This type of academic education focuses on particular systems, situations, or both. How do you configure a Windows 2000 computer to be a WWW server in a demilitarized zone separating the protected internal network from the Internet? What happens if you don't use those specific settings, and what does each setting do? Whom do you call if you are a security guard who spots the director of the CIA carrying classified material out of the building? How do you send medical records to a doctor without compromising either the confidentiality or the integrity of those records? The answers to these questions are embodied in procedures and technologies. You don't need to understand why these procedures are in place or how the technologies work to use them effectively. Obviously, the more understanding a trainee has, the better he or she will handle the job, but in-depth understanding is not needed to perform the required tasks.

Trade organizations and professional and commercial groups provide this type of training in tutorials. These tutorials are typically intensive and hands-on. Attendees should be able to walk out of the tutorial and apply what they have learned immediately. One residual value of these tutorials is the possession of the book, slides, and other supplementary materials used in them. Often the tutorial covers more material than the students can retain perfectly—but if they later see a problem that looks familiar, they can review the tutorial material to refresh their memories.

Such training is also available on the job, provided you are willing to ask questions and are paired with a mentor who is willing to show you how the systems are configured and how to reconfigure them. Although this technique is usually slow, it is effective. The trainee learns specific information about handling a site's or system's problems, rather than the more general knowledge acquired in a tutorial attended by 150 or so people. When combined with a more general tutorial, this training is particularly effective.

Saltzer's and Schroeder's principles of secure design

Jerome Saltzer's and Michael Schroeder's principles of secure design[1] are fundamental to any security system or mechanism. They are as follows:

- *Least privilege*: A process should have only those rights necessary to complete the task. In government and industry, this is the "need to know" principle.
- *Fail-safe defaults*: When a security mechanism or system fails, the system should revert to a known, secure state. This principle essentially requires that the system deny access to sensitive information unless access is explicitly granted.
- *Economy of mechanism*: Security mechanisms and procedures should be as simple as possible, because as systems and mechanisms become more complex, more can go wrong. Furthermore, the more complicated a mechanism, the harder it is to convince people that the mechanism works as needed. This is a general rule, born of human nature and experience. Arthur C. Clarke's marvelous short story "Superiority"—in which the desire to develop complex, powerful weapons leads to defeat at the hands of simpler, less powerful, but functional weapons—casts this principle in terms of science fiction.
- *Complete mediation*: The mechanism cannot be evaded. Dorothy Denning made this principle's importance explicit in her talk at the National Information Systems Security Conference.[2] She pointed out that attackers often evade controls designed to stop them. The controls are never invoked, so they are completely ineffective.
- *Separation of privilege*: Multiple properties must hold for access to be granted. In financial circles, this is called "separation of duty." Two people must sign checks over $10,000. Two soldiers must insert keys to launch missiles. One person is easier to compromise than two who must work in concert. Mathematically, this is a fallacy, but humans are not mathematical.
- *Open design*: A system's security should not depend on hiding the details of how the system functions. Hiding specific information such as passwords does not violate this principle, but hiding the general design of a security policy or system does. Attackers can construct the details of systems in various ways. For example, for security procedures, dumpster diving is effective. In 1972, Bob Woodward and Carl Bernstein determined the lines of reporting in the highly secretive Committee to Re-Elect the President by examining telephone numbers and seeing who had phone numbers "close" to whom.
- *Psychological acceptability*: Security procedures and mechanisms must be as easy to use as to ignore. This principle is usually watered down to say that using the security mechanisms must not be too onerous. Passwords and badges are generally acceptable. In high-security institutions, fingerprints provide a high degree of authentication. But requiring fingerprints for authentication to enter a university laboratory would be unacceptable, at least at the University of California. The students, staff, and faculty would simply not tolerate it.

References

1. J. Saltzer and M. Schroeder, "The Protection of Information in Computer Systems," *Proc. IEEE*, Vol. 63, No. 9, Sept. 1975, pp. 1278–1308.
2. D. Denning, "The Limits of Formal Security Models," www.cs.georgetown.edu/~denning/infosec/ award.html (current Oct. 2000).

Undergraduate education

This type of academic education's goal is to learn broad principles and their application. It does not focus on any particular situation or system. In practice, the best instructors take case studies and generalize them to exhibit the underlying principles. This method helps students acquire a sense of what is principle and what is detail and how to differentiate them. Subsequent exercises emphasize these principles and have the students apply them in different ways.

The advantage of a good undergraduate education is the breadth of application of principles taught. For example, in computer science, classes in algorithms, databases, operating systems, programming languages, architecture, and information systems teach various principles of information security and how to apply them in the given realm. Political science and history classes teach principles of information security in studies of government and political movements, such as those discussed in Sun Tzu's *The Art of War* and Saul Alinsky's *Rules for Radicals*. Literature classes sometimes discuss those principles as they study stories such as "The Purloined Letter" and *Oliver Twist*. The fact that this knowledge comes from disciplines other than those naturally allied with information security testifies to its importance.

For example, consider Alinsky's third rule for organizers:

> The third rule is; Whenever possible go outside of the experience of the enemy. Here you want to cause confusion, fear, and retreat.
>
> General William T. Sherman, whose name still causes a frenzied reaction throughout the South, provided a classic example of going outside the enemy's experience. Until Sherman's march, military tactics and strategies were based on standard patterns. All armies had fronts, rears, flanks, lines of communication, and lines of supply. Military campaigns were aimed at such standard objectives as rolling up the flanks of the enemy army, or cutting the lines of supply or communication, or moving around to attack from the rear. When Sherman began his famous March to the Sea, he had no front or rear lines of supplies or any other lines. He was on the loose and living on the land. The South, confronted with this new form of military invasion, reacted with confusion, panic, and collapse. Sherman swept on to inevitable victory. It was the same tactic that, years later in the early days of World War II, the Nazi Panzer tank divisions emulated in their far-flung sweeps into enemy territory, as did our own General Patton with the American Third Armored Division.[2]

The relevance to information security is obvious. If you're an attacker, look for unexpected openings. Look at the models the defenders have used to secure their information. Find ways to sidestep the mechanisms that the model requires, or better, invalidate its assumptions. Dorothy Denning made this point eloquently in her National Computer Systems Security Award acceptance speech.[3] She described several incidents in which supposedly secure mechanisms were breached by people who went outside the conventional modes of analysis and found forms of attack that the models had not considered. Denning's talk, incidentally, emphasized the Alinsky passage's lesson for defenders: expect to be attacked in ways you cannot anticipate, and be prepared for it.

Master's education

This type builds on undergraduate education. The student must examine a particular area of the discipline in depth, either through additional course work and examinations or through course work and projects culminating in a thesis. The thesis typically develops an application of an information security principle to a specific situation or set of situations. This education teaches the student to weigh competing interests and determine how best to apply different technologies to reach the desired balance.

Some examples of typical master's-level work are

- analyzing a particular network security protocol to determine whether it has flaws and suggesting changes to ameliorate the flaws,
- designing and implementing a library of specifications for security properties that are to be used with a testing tool, and
- developing a policy model for an academic institution.

The first task applies analytic and experimental techniques to a protocol to determine whether it works correctly on the Internet. The second uncovers common flaws in programs and shows how to abstract from them a description sufficient to identify previously unknown instances of the problems. The third combines technology with an analysis of the needs of the differing organizations making up an academic community and presents mechanisms to enable the disparate groups to work together.

People with this kind of experience know how to negotiate the conflicting demands of policy requirements, technological capabilities, and human factors. They can analyze problems and look for solutions. Sometimes no solutions are possible and an approximation is necessary, and good analysts can determine the potential problems with approximations. In any case, they can bring together their experience in technology, principles, and analysis to formulate guidelines that describe the protection needed. They can then design mechanisms to provide that protection.

Doctoral education

This type also builds on undergraduate education. Unlike a master's education, doctoral-level work analyzes the principles of information security, extends them, changes them, and improves them—or derives new principles altogether. The goal is to deepen the student's understanding of systems in such a way as to enable him or her to add to the body of knowledge. From this perspective, the student gleans fundamental insights into improving the state of the art and science of information security—and indeed, into what is and is not possible.

The primary difference between doctoral and master's work is the nature of the concepts studied. Master's work typically emphasizes applications or applied research in some form. Doctoral work emphasizes fundamental results and research, often called "basic research." Doctoral work pushes the boundaries of knowledge. The results might not be applicable immediately, but they improve our understanding of the technology and its limits and uses—and for that reason they are critical.

Doctoral study also provides the necessary credentials for employment at a research university; it testifies to the student's ability to perform original, significant research. At research universities, doctoral-level teaching is not only a classroom exercise. Professors work with students in their research. Students learn how to conduct research, how to ask meaningful questions, and how to design experiments to demonstrate problems and solutions. In addition, students acquire an understanding of how to abstract problems into mathematical realms, where they can be analyzed formally. They also learn how to relate the formal analyses back to the problem to use whatever light their abstract analysis sheds on the problem.

Which is best?

These four forms of academic education have no hierarchy of importance; someone with a doctorate is not better educated to handle a particular problem than someone with training for that problem. But someone with a doctorate can analyze that problem, abstract it, work with the abstraction, and suggest potential lines of research to eliminate the problem and similar ones. People with PhDs tend to generalize and try to solve classes of problems; people with training tend to focus on the particular problem at hand.

Compare and contrast

Academics emphasize the principles underlying computer security. These range from the theoretical (such as the Harrison, Ruzzo, and Ullman result stating that, in the most general case, security is undecidable)[4] to the applied (such as Jerome Saltzer's and Michael Schroeder's design principles for security mechanisms—see the sidebar). The goal is to be able to apply those principles to situations—in other words, to practice the science and art of computer security.

Good instructors use exercises to drive the ubiquity of these principles home to the students. This type of teaching requires equipment and software either that reflects the principles being taught or to which the students can apply the principles and improve or visibly alter the system. The students then see that they understand the principles well enough to apply them appropriately. (See the "Exemplary Curricula" sidebar for

Exemplary curricula

The following are outlines of topics discussed in the general computer security classes at the University of California at Davis. Special-topics courses cover specific material in more depth.

The undergraduate class (ECS 153) focuses on applications of computer security principles. It emphasizes how to protect systems and discusses some broader principles and models. The main focus is on how to apply the models.

The ECS 153 topics are

- Introduction and what computer security is, basic principles, and ethics;
- Models: confidentiality and integrity (Bell–LaPadula, Biba, Clark–Wilson, Chinese Wall);
- Assurance: robust programming, security in programming, specification, design, testing, and proving programs correct;
- Cryptography: basics, authentication, key management, and example protocols;
- Mechanisms: identity, access control lists, and capabilities; and
- Attacking and defending: models of vulnerabilities, penetration testing, and malicious logic.

The graduate class (ECS 253) covers many of the ECS 153 topics, but focuses on the theory of computer security principles. It covers theoretical foundations as well as much deeper analyses of models.

The ECS 253 topics are

- Introduction and what computer security is, basic principles, and ethics;
- Foundations: the access control matrix model, Harrison–Ruzzo–Ullman results, the take-grant protection model, and undecidability results;
- Cryptography: key management, cipher techniques, and example protocols;
- Models: confidentiality, integrity (Bell–LaPadula, Biba, Clark–Wilson, and Chinese Wall), noninterference and nondeducibility security, and information flow models;
- Assurance: building secure systems, specification, design, testing, and proving programs correct;
- Mechanisms: identity, access control lists, capability lists, ring-based protection, propagated access control lists, the confinement problem, and information flow models; and
- Attacking and defending: models of vulnerabilities, penetration testing, malicious logic, auditing, and intrusion detection.

examples of topics covered in actual computer security classes at the University of California at Davis.)

Industry needs to protect its investments in people, equipment, and intangibles (such as bank balances, availability of services, and proprietary information). Security mechanisms must do this effectively. The principles they embody are less important than their efficacy. In this realm, computer security is applied and practical. Industrial computer security education's goal is to analyze a site and balance internal and external threats to the company with the costs of implementing security measures—with a minimum cost in training to the company.

The government uses computer security as one of many tools to protect the national interest (I assume this is well defined). Threats arise from both external attackers and government employees who abuse their authority or act against the citizenry's best interests. The specific protections are legally mandated and not subject to the same cost-benefit analysis industry can afford. So, computer security education for government employees focuses more on developing policies and systems to implement laws and regulations, and less on cost balancing.

These differences point out the need for education at several levels. Each level has something to contribute, and people at each level help educate each other. For this reason, all levels deserve support and use in protecting information.

The state of information security education

The difficulty of hiring people educated in computer security has led to interest in and discussion about improving information system security education. The desired improvements include establishing core curricula and integrating computer security into more aspects of computer science education.

Specifically, the National Security Agency program establishing Centers of Academic Excellence in Information Assurance Education has as an evaluation criterion that the academic program treat information security not as a separate discipline but as a multidisciplinary science, incorporating information assurance knowledge into various disciplines.[5] This program recognizes institutions that are teaching students about information security even when a student's primary area of study is not information security. The NSA's recognition of these Centers of Excellence is a first step in increasing the visibility and quality of computer security education.

It is, however, only a first step. A designation as a Center of Excellence confers no support or benefits on the institution other than the distinction itself. To be fair, the NSA has always said that this would be the only reward, but it had hoped that the Centers for Academic Excellence would become focal points for recruiting and would create a climate to encourage independent research in Information Assurance. Perhaps that will happen soon.

Past security problems continue to recur. The ILoveYou worm is a perfect example of this pattern. In 1988, before the Internet virus appeared, the Christma Exec worm threaded its way through several IBM networks. Victims received a letter telling them to save the body of the letter as a file and then execute the file to get a pleasant Christmas greeting. When they did so, they saw a Christmas tree with blinking lights drawn on their screens. What they did not see was the rest of the program, which looked in their Names and Netlog files to get the names of other correspondents to whom it would forward itself. The resulting e-mail storm made several IBM networks unusable until the worm was cleaned out. The ILoveYou worm used almost exactly the same techniques. The only differences were that the recipient had to click on a button rather than save the file and execute it and that the ILoveYou worm downloaded a second program that harvested passwords from the Windows system's cache.

Software still suffers from buffer overflows. Privileges are not constrained properly. Race conditions let unscrupulous users gain control of systems. There is nothing new under the sun. What has happened before will happen again, but we are not learning from these mistakes.

Nor have we improved how we design systems and programs to account for security problems. Consider Windows 2000. Microsoft's security mechanisms are conceptually excellent, but their implementation and integration into the system lack coherency and cohesiveness. Furthermore, some subsystems have design and implementation problems. Microsoft has released several patches for both systems and application software and still has numerous security-related issues pending. Similar criticisms hold for all varieties of Unix or Unix-like systems. The underlying problem is that we still do not design with security as an integral part of the design. We patch. We add security above the kernel, or retrofit it.

All forms of education, from basic research to training, are critical to an effective response to the information security crisis we face. We must particularly focus on basic research and higher education. This will provide the teachers and researchers needed to train system administrators, business executives, and management in the intricacies of information security that affect them and their organizations. Furthermore, the emphasis on basic research will seed more universities and academic institutions with people who can teach and do research in that area.

Throughout this process, we should not forget the dreamers, the people with long-range vision. Most education focuses on short-range or medium-range planning, which should not preempt long-range planning. Technologies will change. Systems will become obsolete. The infrastructure will evolve in unanticipated ways. The dreamers will provide the vision. For example, Ted Nelson conceived of hypertext in the mid 1970s, as he studied how computers and books could work together. Can you imagine the World Wide Web without hypertext? Nelson was a dreamer, but he had a technologically sound vision. People such as Nelson lead the way.

A focus on the immediate and near future runs the risk of creating people like General Carpenter in Alfred Bester's story "Disappearing Act."[6] In that story, America is involved in a war and has become a nation of experts, in which "every man and woman must be a specific tool for a specific job, hardened and sharpened by … training and education to win the fight for the American Dream." Some injured soldiers in a hospital learn to vanish and reappear at will. An investigation convinces the general that the casualties are going back into time, so he asks a historian to determine how they do it. The historian quickly realizes that the casualties are traveling elsewhere, "back into a time of their own imagination." He continues:

> "The concept is almost beyond understanding. These people have discovered how to turn dreams into reality. They know how to enter their dream realities. They can stay there, live there, perhaps forever. My God, Carpenter, *this* is your American dream. It's miracle working, immortality, Godlike creation, mind over matter, ... It must be explored. It must be studied. It must be given to the world."
>
> "Can you do it, Scrim?"
>
> "…No, I cannot. I'm an historian. I'm non-creative, so it's beyond me. You need a poet ...." … Carpenter snapped up his intercom. "Send me a poet," he said.
>
> He waited and waited ... and waited ... while America sorted through its two hundred and ninety millions of hardened and sharpened experts, its specialized tools to defend the American Dream of Beauty and Poetry and the Better Things in Life. He waited for them to find a poet, not understanding the endless delay, the fruitless search; not understanding why Bradley Scrim laughed and laughed and laughed at this final, fatal disappearance.

The worst catastrophe would be to have a "cyberspace" of hardened, sharpened tools trained and educated for a specific job, and no one who knows how to ask whether another approach to the task exists or how to look for it.

References

1. J. Saltzer and M. Schroeder, "The Protection of Information in Computer Systems," *Proc. IEEE*, Vol. 63, No. 9, Sept. 1975, pp. 1278–1308.
2. S. Alinsky, *Rules for Radicals,* Random House, New York, 1972, pp.127–128.
3. D. Denning, "The Limits of Formal Security Models," www.cs.georgetown.edu/~denning/infosec/ award.html (current Oct. 2000).
4. M. Harrison, W. Ruzzo, and J. Ullman, "Protection in Operating Systems," *Comm. ACM*, Vol. 19, No. 8, Aug. 1976, pp. 461–471.
5. Centers of Academic Excellence in Information Assurance Education (Graduate and Undergraduate Levels): Criteria for Measurement," www.nsa.gov/isso/programs/coeiae/measure.htm (current Oct. 2000).
6. A. Bester, "Disappearing Act," *Virtual Unrealities: The Short Fiction of Alfred Bester*, Vintage Books, New York, 1997.

Matt Bishop is a professor in the Department of Computer Science at the University of California at Davis. He received his PhD in computer science from Purdue University, where he specialized in computer security. He was a research scientist at the Research Institute of Advanced Computer Science and was on the faculty at Dartmouth College before joining the Department of Computer Science at the University of California at Davis. He gave the academic keynote address for the first National Colloquium on Information Systems Security Education, revisited it at the fourth Colloquium, and has taught at conferences and workshops on computer security. Contact him at the Dept. of Computer Science, Univ. of California, Davis, One Shields Ave., Davis, CA 95616-8562; bishop@cs.ucdavis.edu; http://seclab.cs.ucdavis.edu/~bishop.