Grand Challenges in Information Security: Process and Output

S.W. Smith
Dartmouth College

Eugene H. Spafford
Purdue University

Even casual observers can see that our society's computing infrastructure has significant security problems. Technical sources such as CERT, BugTraq, and the *Risks Digest*—not to mention the popular media—regularly catalog critical vulnerabilities in deployed software. What's even more disturbing (but useful to university instructors) is seeing the same basic flaws repeatedly show up. Need an example of buffer overflow or time-of-check, time-of-use? Simply look at last week's news.

We also see flaws in our overall architectures, network protocols and deployment, and our regulation and law enforcement related to computing. Another disturbing trend is the way society turns to criminals and vandals as "experts," because we don't have enough well-trained professionals.

Deployed technology thus illustrates, by example, that we haven't learned how to build systems that we can reasonably trust to work correctly despite adversarial action. Nevertheless, we rush headlong to move even more important processes into computing systems, and to make commodity computing systems more ubiquitous. The increase in computing systems' performance, in net-bandwidth-per-dollar worldwide, and online storage all complicate the picture by providing more systems to protect, and greater temptation for those who could abuse those systems.

These all are disturbing trends. Unless we figure out how to build trustworthy systems in the real world, we're in trouble. Recognizing that fact, the Computing Research Association (CRA, www.cra.org), with support from the US National Science Foundation, recently drafted its Grand Research Challenges in security and assurance, intent on galvanizing the field by focusing attention and stimulating progress on these problems. This department generally focuses on "big picture" security issues, but in this issue, we'll focus on these grand research challenges—systems don't come much bigger than the real world.

History

The CRA represents nearly every North American college and university with a computing research program, as well as several industry and government research labs, and the major computing associations (including the IEEE Computer Society). The CRA also pursues efforts in other important areas, such as workforce development and support for the computing community. The CRA focuses on computing research both internally (by stimulating and promoting research and education) and externally (by representing the field for policymakers and others). When addressing the security crisis, both of these missions coalesce.

The CRA decided that it needed tangible goals (grand research challenges in information assurance) important enough to motivate and focus both specialists (educators, students, and scientists) and the general population, but revolutionary enough so that this pursuit and achievement might fundamentally change things. Just as former US President John F. Kennedy's exhortation to put a man on the moon helped usher in the Space Age, the right challenges might help usher in (finally) the Trustworthy Computing Age.

A first natural step is to hold a conference. During typical computer science conferences, researchers submit papers, and referees select a handful to be presented by the authors (and published in the proceedings); and anyone who registers can attend. This format thus focuses on finished results. However, a conference producing grand research challenges needs to work the other way, selecting participants not because of their finished work but because of the work they might finish collectively.

Here, the CRA drew on history. In 1931, Neil Gordon of Johns Hopkins University started a series of Gordon Research Conferences in the hard sciences that aimed for this goal. Attendees were selected based on what they could contribute to the discussion. Gordon structured the entire conference format to encourage discussion: everything was "off the record," and formal meetings were interspersed with free time—in a setting in which the attendees spent their free time in continued discussion.

In June 2002, the CRA held the Grand Research Challenges in Information Systems conference (www.cra.org/Activities/grand.challenges/). This conference used the Gordon

Research Conference format, but with a published output—a report outlining the resulting challenges.

In November 2003, the CRA repeated this process for the Grand Research Challenges in Information Security & Assurance conference. The call for papers invited prospective attendees to write a two-page statement proposing and discussing two or three grand challenges. The first author (Sean Smith) submitted his usual rants: How do we enable meaningful trust judgments in complex environments? How do we build large systems that do not have holes? How do we make security usable?

The second author (Gene Spafford, a member of the CRA board of directors) and Rich DeMillo of Georgia Institute of Technology, chaired the organizing committee that sifted through the 220 proposals and invited 50 attendees—selected not just for their leadership in the field, but also to represent a diversity of backgrounds and career positions.

The invitees then committed to attending the entire conference and to abiding by the spirit of a Gordon conference: that is, wildly open discussion, not for attribution (but in case you were wondering, this department was approved as not violating this commitment).

The process

And thus, one rainy November day at an isolated estate in rural Virginia, we gathered in a large meeting room. The CRA's Anita Jones and William Wulf explained the Gordon format, and challenged us to think "outside the box."

This opening charter sketched out two visions of our planet's computing future. One is a rosy world, in which all security problems have been solved. The other is a straightforward extension of the current state of things: a computing infrastructure plagued by bugs, vulnerabilities, outages, and unwanted activities such as spam—and largely useless for anything important. For Smith, this vision crystallized the need for revolutionary action in information assurance. Do we really think that 10 more years of business as usual will result in anything significantly better?

Even though the invitee list was small, we divided into smaller groups with smaller scopes to enable good discussions. The organizing committee distilled the original submissions into a set of five general subject areas. We moved into separate rooms to discuss those subjects, broke for a few hours, and then reconvened. During the breaks, the organizing committee would consider the groups' ideas and repartition the idea space again.

What was this process like for the participants? Initially, one of us (Smith) felt at a disadvantage because what he was passionate about touched on many of the topic areas. But this became a feature, not a bug—we could participate in one group long enough to see that things were going in a direction that seemed reasonable, and then switch to another. In true Gordon tradition, the isolated setting ensured that we'd have little to do but stroll the grounds and talk to each other. (More than one participant noted with amusement the warning signs that "Swans may be unfriendly—do not turn your back." A metaphor for security?)

It was an intense experience. We were struck by the mix of personality types, how people often moved out of the comfort zone of well-rehearsed ideas, and engaged in lively academic discussion, as well as by the number of interesting ideas that did not make the CRA's final cut. (These will be on at least a few personal research agendas, however.) It felt like our brains were wrung dry—after just one day.

Four Grand Challenges

After this long process—soliciting and selecting attendees, distilling areas, endless discussion, dissension, and consensus, shuffling, and starting again—we agreed on four grand challenges. (These are "Four Grand Research Challenges," not "The Grand Challenges"—because the idea space was rich.)

Epidemic-style attacks

We must stop epidemic-style attacks in 10 years. Computing is plagued by epidemic-style attacks. Spam makes it hard to read email; denial-of-service (DOS) attacks bring down critical sites at inopportune times; and viruses and worms continue to plague systems—and are starting to plague critical infrastructure (such as ATMs and emergency-response systems) that previously had resisted them.

Attacks are propagating increasingly faster, and humans (and automated systems) cannot respond. The problem is asymmetric—attackers can be local, and they require few resources and entry points, whereas defenders must be global and organized.

This problem is "grand" because it's important. Such attacks are high-cost and will grow as more critical infrastructure is affected. (*IEEE Security & Privacy*'s own Bruce Schneier has recently speculated that, in part, a worm might have caused the 2003 blackout.) Solving this problem requires overcoming many technical and logistical challenges, but success can be easily and tangibly demonstrated: DOS attacks or worms, for example, simply would no longer halt the infrastructure on a regular basis.

Trustworthy large-scale systems

We must build trustworthy large-scale systems for important societal applications. We use computers for important tasks, such as voting, health records, and law enforcement. However, something about the way we build large, networked software systems leads to vulnerabilities—again, witness the state of the CERT curve, or the fact that computer scientists were reputed to have said the same thing at the 1968 NATO Conference on Software Engineering.

As we move sensitive operations onto networked general-purpose machines, on what grounds can stakeholders trust that the networks can resist dedicated attackers? What's going to happen when remote-code-injection vulnerability is shown to exist—and has been used—in the commodity OS supporting a presidential election?

As before, this problem is grand because it's important, and because solving it will require solving software engineering, production, and composition problems whose solutions have eluded the field since its inception.

Quantitative information systems risk management

We must make quantitative information systems risk management as good as quantitative financial risk management. At first glance, the credit-card system baffles security students. The authentication is so weak, but somehow the credit-card industry remains afloat. Part of the answer here is the way this technology embeds in a larger financial system, where decisions about risks and defenses are supported by well-understood risk-management techniques.

In the financial realm, corporate officers can gauge what they are getting for their investment, and when they are spending too much or too little. However, in information security it's all black magic. Corporate chief information officers do not have well-founded techniques to evaluate whether they are spending too much or too little on security technology, or whether they are incurring more or less risk than they did a year earlier. To paraphrase Lord Kelvin, we cannot manage what we cannot measure.

We need a sound quantitative risk-management theory for information technology risk. Such a thing would enable government, industry, and consumers to make rational decisions about security investment and provide a basis for both the free market and public policy to seek out a stable, trustworthy state.

End users security and privacy

We must give end users security they can understand and privacy they can control. Two recent trends in computing seriously impact the human end user. First, technology is becoming complex to the point of incomprehensibility. Even an experienced user has trouble conceptualizing exactly what services his or her machine offers right now on the network and what pull-down menus and configuration files to change to steer those services into a more acceptable state.

This situation will only get worse as we continue to extend the analysis to less savvy users or to designers and integrators—or to the complex, pervasive computing environments looming just around the corner.

Second, you manage your privacy by making free choices about your actions. However, moving activity into a networked computing environment, with machines and software representing many stakeholders (potentially remote and invisible) makes it much harder to delineate exactly what's involved in these actions. Where is your private information going today? Did you know it was going there?

Reconciling these trends is a grand challenge. Human users must make rational choices about their computing actions, but cannot make such choices if they cannot understand the systems. All stakeholders should have thorough discussions of the range of potential privacy policies for computing services—but they can't do that if the default policy offered by the default technology is accepted as the only solution technology can offer. Technology dictates social values, when it should be the other way around. At the end of the day, if our computing systems are not serving the needs of human users, then what's the point of building them?

What's next

Some initial materials from the conference—a press release and some slides—are available at the CRA Web site (www.cra.org/Activities/grand.challenges/security/home.html). A more complete report is being prepared and should be available in early 2004.

Working on these four grand challenges is the next step, but the overriding motivation behind them is: how can we ensure that the next decade of concerted effort can make our information infrastructure trustworthy, reliable, and usable? By default. For everyone. □

__S.W. Smith__ is currently an assistant professor of computer science at Dartmouth College. Previously, he was a research staff member at IBM Watson, working on secure coprocessor design and validation, and a staff member at Los Alamos National Laboratory, doing security reviews and designs oratory for public-sector clients. He received a BA in mathematics from Princeton University, an MSc and a PhD in computer science from Carnegie Mellon University. He is a member of ACM, Usenix, the IEEE Computer Society, Phi Beta Kappa, and Sigma Xi. Contact him at sws@cs.dartmouth.edu or through his homepage, www.cs.dartmouth.edu/~sws/.

__Eugene Spafford__ is a professor of computer science, philosophy, communication, electrical and computer engineering at Purdue University. He is also the executive director of the Purdue University Center for Education and Research in Information Assurance and Security (CERIAS). His primary research is on issues relating to information security, with a secondary interest in the reliability of computer systems, and the consequences of computer failures. He received a BA in mathematics and computer sciences from the State University College at Brockport, New York, and his MS and PhD in information and computer sciences from Georgia Institute of Technology. He is a member of the ACM, Usenix, the IEEE Computer Society, American Association for the Advancement of Science, and Sigma Xi. Contact him at spaf@cerias.purdue.edu or through his homepage, www.cerias.purdue.edu/homes/spaf/narrate.html.