Achieving Learning Objectives through E-Voting Case Studies

Matt Bishop
University of California, Davis

Deborah A. Frincke
Pacific Northwest National Laboratory

The rapidly increasing use of electronic voting machines in US elections provides a wonderful opportunity to teach students about computer security. The complexity of transitioning from traditional voting to an electronic environment allows educators to highlight threat models, requirements, and trade-offs involving e-voting in the context of ongoing international discussions and current events. Few issues include such a wide range of considerations—from the competing demands of accessibility and confidentiality to threat models incorporating coercibility and vote selling and even the business element of whether funding high assurance is a good way to increase voter confidence.

In this article, we present an informal e-voting case study to achieve five learning outcomes for students in a typical college (or even high school) classroom. Our intent is to motivate a set of lessons specifically involving e-voting, as well as illustrate the usefulness of mapping outcomes to simplified case studies:

- understanding how to write a "security specification,"
- learning about different forms of security policies,
- understanding confidentiality, privacy, and information flow,
- recognizing the importance of considering usability from a security perspective, and
- identifying assurance's role in establishing confidence in results.

This dovetails well with the guidelines established by the Accreditation Board for Engineering and Technology (ABET; www.abet.org) identifying desired student education outcomes for accredited colleges and universities. Table 1 shows how our case study can support the "ABET a–k" requirements.

We don't assume access to special-purpose hardware or high-maintenance security labs: if a voting system isn't available, instructors can build on published descriptions of the systems and algorithms.

Identifying security-relevant requirements

First, we ask what constitutes a "secure voting system." Discussing requirements is a useful place to start identifying what is being protected, from what, and why. A live exercise helps students understand the importance and complexity of requirements identification. Divide the class into four or five informal teams and have each identify a set of requirements and a threat model and then present their findings to the class.

Table 1. How the e-voting exercises satisfy the Accreditation Board for Engineering and Technology's requirements.

OUTCOME	ABET BEHAVIORAL OUTCOME CRITERIA	APPLICABLE E-VOTING SECTION
A	"An ability to apply knowledge of mathematics, science, and engineering"	Throughout, but especially in the sections on establishing confidentiality and understanding the human element
B	"An ability to design and conduct experiments … analyze and interpret"	Requirements and specification sections
C	"An ability to design … to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability"	Requirements, human element, and confidence sections
D	"An ability to function on multidisciplinary teams"	Throughout, but especially group discussions and confidence section
E	"An ability to identify, formulate, and solve engineering problems"	Requirements, specifications, and human element sections
F	"An understanding of professional and ethical responsibility"	Requirements, human-element, and confidence sections
G	"An ability to communicate effectively"	Throughout, but especially in group discussions
H	"The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context"	Throughout, but especially in requirements, human element, and confidence sections
I	"Life-long learning"	Potentially throughout
J	"A knowledge of contemporary issues"	Throughout
K	"An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice"	Requirements, specification, and confidence sections

This approach is particularly effective if the class includes students with different backgrounds, technological skills, and countries of origin. The discussion provides both an opportunity to talk about what people expect from e-voting systems and a way to assess and enhance critical thinking and public speaking skills within the class. If used early in a semester, the exercise will provide a good foundation for later exercises and lead to a livelier classroom.

The threat model's role is another useful concept to explore. E-voting presents various relatively unexplored threats; coercion is just one of them. Having individual teams compare their models in the context of the overall security goals they propose can be instructive—all too frequently, developers construct system defenses without precisely articulating the threats considered. These student-generated threat models will also prove useful when discussing assurance and associated costs.

For our case study, assume that the class chose four requirements: election fairness, ballot secrecy, election auditability, and system usability. Further, assume that the teams selected at least two threat models. At this stage, the instructor might have them compare methods of defining security requirements, particularly those with differing degrees of formality. She could go into more detail on threat modeling and introduce categorization approaches such as STRIDE (spoofing, tampering, repudiation, information disclosure, denial of service, and elevation of privilege). Another option would be to discuss the various standards that influence e-voting machine requirements.

Understanding specification

Identifying e-voting machine requirements and a threat model naturally feeds into discussion of how to specify an appropriate implementation. How would we formally specify a *fairness policy* for an election? Central issues include who can vote, how they vote, and how the votes are tallied and confirmed. Trust and accountability come into play: how can all parties assure themselves that the election was fair and the results were accurately reported? These issues present an excellent lesson on balancing competing requirements and illustrate how a purely technological solution is unlikely to be satisfactory.

Suppose we established three requirements: ballot secrecy, tally accuracy, and accountability. If we didn't care about secrecy, one way to achieve accuracy would be to watch each voter mark each vote, ask their intent if they made a mark that was unclear or questionable in any way, have them re-mark the ballot properly, and then count the votes. But this approach violates the confidentiality requirement; the re-marking process could potentially reduce accountability, as well. Similarly, ballot secrecy, in isolation, has numerous

solutions: once the marked ballots were in the box, officials could transfer the box to a second location, count the ballots, and then destroy the originals without record. Yet, this process would make establishing accuracy impossible, again eliminating accountability. Balancing the requirements creates complexity.

An examination of the US Election Assistance Commission's Voluntary Voting System Guidelines is appropriate here (www.eac.gov). Most states incorporate these guidelines in their requirements for voting systems. What drives the guidelines? What threats lead to the requirements specified in them, and how do the requirements prevent against those threats? E-voting presents an informative exercise in how to develop standards—and how not to.

Confidentiality, privacy, and information flow

Establishing ballot secrecy seems simple to many: record the vote without storing anything identifying the voter. The direct approach analogous to the paper case includes banning unique serial numbers and other marks from each electronic ballot. Yet, we must consider a more demanding notion of information flow in both the physical and electronic worlds to avoid indirectly encoded associations. Assume that the order in which ballots are cast is known—they might be printed on a spool of paper such as those used with cash-register receipts. This situation arises when a voting system prints a paper copy that the voter must approve to officially cast a ballot. It could also arise if the ballots were time stamped or otherwise uniquely marked. The ballot's position relative to the others would thus encode information unique to the voter, even if the ballot had no unique mark on it. This situation doesn't arise with paper ballots shaken or otherwise mixed up inside the ballot box.

Ask the class to determine what implicit information flow channels exist in an e-voting system. This leads to a discussion of how to counter those channels. Students typically begin by suggesting alterations to implementation details that conflict with policy requirements, have adverse effects on efficiency, or introduce additional information flows, which presents a good opportunity to discuss the intricate interplay between security requirements and security implementation, and why seemingly simple requirements can be hard to get right.

Now, ask students whether the threat models identified earlier encompass the circumstances that give rise to indirect information flow. If they realize at this point that their threat model is incomplete, what should they do?

For classes with greater sophistication in algorithm analysis, introduce ordering-attack threat mitigation. One method arises from noting that nonelectronic ballots are naturally ordered because of the physical medium on which they're printed (the spool of paper). In the physical realm, cutting the paper and mixing the cuttings helps solve the problem, but the mechanisms are cumbersome. Another method arises from noting that analyzing the information flow requires access to two components: the ordered ballots and the order in which voters used the machine.

If we can't prevent knowledge of the former, we must prevent attackers from learning the latter by ensuring that no one sees the order in which people vote. Yet, this conflicts with the auditability requirement, which says that every aspect of the election except the voters' marking of the ballots must be observable. Asking students to balance auditability with the mechanism that blocks the implicit information flow's violation of secrecy, or to devise another mechanism to block that flow, illustrates how mechanisms can affect policies, and vice versa.

Understanding the human element

Another aspect of the implementation is the e-voting system's usability. These systems must function properly during testing and elections but are otherwise unused. Furthermore, few of the volunteer poll workers who set up the systems will be familiar enough with the special-purpose e-voting systems to fix problems. The polling station setup and closing must therefore be simple. The situation with casting votes is similar: few voters will be e-voting experts, and some might even be fearful of general computing technology. Yet, all must cast ballots using the provided interface, so the way to use that interface must be obvious. The ballot shown on the screen can be lengthy, depending on the particular races—the 2003 California gubernatorial election saw more than 130 candidates run for governor, for example. As such, the ballot layout and the interface require careful design and testing to ensure that the average voter can find, and vote for, any candidate. The user interface must also be suitable for voters with special needs.

A system's security depends on its being correctly configured and used. This aspect of e-voting systems provides fertile ground for the class to see how usability requirements affect mechanisms and, in some cases, policy.

Establishing confidence in the final tallies

A pervasive element of the earlier discussions is the notion of assurance—confidence that the e-voting machine meets the requirements for a fair and accurate election. In most environments, this requires enough evidence to convince an expert, or perhaps a group of them, but the election environment is different. One fundamental principle is that an election must be observable, so that anyone can watch the polls opening, the voting, the polls closing, the transportation of the ballots to the

For more information

- Electronic Frontier Foundation's e-voting Web site, www.eff.org/Activism/E-voting/.
- E. Barr, M. Bishop, and M. Gondree, "Fixing the 2006 Federal Voting Standards," to appear in *Comm. ACM*, Mar. 2007.
- A. Rubin, "Security Considerations for Remote Electronic Voting over the Internet," tech. report, 2006; www.avirubin.com/e-voting.security.html.
- US National Academies' Computer Science and Telecommunications Board Project: A Framework for Understanding E-voting, http://www7.nationalacademies.org/cstb/project_evoting.html.

Challenge question

In the US, states such as California require security-penetration studies as part of the periodic assessments of systems that record real estate documents over the Internet. Should electronic voting and tallying systems face a similar requirement?

counting sites, and the counting of the ballots to assure themselves that no chicanery has occurred. In practice, voters typically trust the election officials, but the fact that observers can enter at any time is a powerful inhibitor of electoral thievery.

Who should assurance evidence convince? Some say the election officials, who are the experts at running elections. Others add technical experts (such as computer scientists) who know the technology. Another view is to consider the voters to be the experts and to require that the evidence be as convincing to them as observing an election without e-voting machines would be. Yet, this raises some interesting, and difficult, issues.

First, it introduces an ethical issue. Western democracies and republics have shied away from Plato's ideal philosopher-kings, preferring instead to give the members of the body politic a voice in electing their governments. Holding that those members must rely on others' opinions that an election is indeed fair and accurate runs counter to this philosophy.

A second issue is practical: How can experts communicate the typically highly technical evidence of assurance to people who are unfamiliar with the intricacies of operating systems, device drivers, and software? An interesting exercise is to pick some simple system (such as a change machine driven by a program that exchanges coins for paper money), and ask the class how they could convince themselves that it always made the correct change. Then ask them to convince someone who doesn't regularly use computers. When they realize that evidence must account for the hardware, operating systems, and compilers used to generate executables, students quickly understand the problem's difficulty.

The third issue relates to system improvements. Assurance involves detailed specifications and an understanding of the operational environment. When verifying that a system meets a given set of requirements, an evaluator certifies the current system as it is—upgrades require recertification. If a system is modified in any way, the evidence for a particular degree of assurance must be reevaluated in light of the changes to determine if that evidence is still valid. Try asking the class to determine how patches affect the assurance needed for an e-voting system. High assurance development can also be expensive. How much are we willing to pay?

The fourth issue is the requirements themselves. The various laws governing elections come from different levels of government and can vary dramatically. In California, for example, the governor is elected by majority vote, but some cities use choice voting (generally, instant run-off) for local positions. Voters thus need assurance that the e-voting system correctly implements the correct set of laws.

Finally, consider the operational environment. Assurance techniques make assumptions about this environment—for example, that the system is not connected to a network or that the passwords used to protect accounts are hard to guess. An interesting exercise is to "reverse engineer" what assumptions the vendor, election officials, and voters must make about the e-voting system's environment and the way the system is used. This exercise brings out the policies and procedures required to support the system in a way that lets the students compare and contrast those policies and procedures with the ones they developed earlier in the class. The effects of increasing voter confidence through recount is another aspect of assurance and environment that the instructor can consider at this stage.

Electronic voting is a complex, difficult subject. It presents a cornucopia of challenges that can illuminate some aspect of security, and it supports the learning outcomes established by ABET. Students have responded enthusiastically to this real-world example of how important assurance and security is not only to them but also to their families, friends, and the community at large—and at a time when they themselves can contribute to decisions about e-voting. What more could we ask from a real-world example to use in a classroom? □

Acknowledgments

We appreciate the assistance of Alec Yasinsac and others who reviewed earlier versions of this column.

Matt Bishop *is a professor of computer science at the University of California, Davis. His interests include elections and election systems. Bishop is author of* Computer Security: Art and Science *(Addison-Wesley, 2002). Contact him at bishop@cs.ucdavis.edu.*

Deborah A. Frincke *is chief scientist of the CyberSecurity group at the Pacific Northwest National Laboratory. Her research interests include security of high-speed systems and system defense, especially intrusion detection. Contact her at deborah.frincke@pnl.gov.*