Elections and Computers: A Match Made in … Someplace?

joint work with Earl Barr, Dimitri DeFigueiredo, Mark Gondree, Patrick Wheeler

Matt Bishop
Dept. of Computer Science
University of California, Davis
1 Shields Ave.
Davis, CA 95616-8562

Outline

- Role of electronic voting systems in elections
- Federal voting standards and problems
- Assessing electronic voting systems
- Conclusion

How an Election Works (Yolo)

- Voters
  - Go to polling station
  - Give name, get ballot
  - Enter booth, vote using mechanical punch to perforate ballot or (lately) pen to mark ballot
  - Put ballot in protective sleeve
  - Leave booth, drop sleeve and ballot into ballot box

End of the Day

- Election officials take ballot box to Election Central (county seat, in Yolo County)
- Election officials remove ballots from envelopes
  - If provisional, handled differently
- Ballots run through automatic counters
- Ballots for 1% of precincts counted by hand
  - Compared to tallies from automatic counter

What's an "e-Voting System"?

- Intended to replace paper
  - Improve clarity of cast vote
  - Less error-prone to errors in counting
  - Easier to store
- Casting votes: DREs, BMDs
- Counting votes: opti-scan, computer vote-counting

What Should It Do?

- Summary: replace technology used in election process with better technology
  - "Better" means that the technology improves some aspect of the election process
- Examples
  - Easier to program ballots than print ballots
  - Can handle multiple languages easily
  - Easier to tally than hand counting

Goal

- Provide sufficient evidence of assurance to target audience that using e-voting systems makes elections at least as secure, accurate, *etc.* as current elections
- Who is "target audience"?
  - Computer scientists, election officials, politicians, *average person*

Thought

Thought

There's no sense in being precise when you don't even know what you're talking about.

— John von Neumann

Requirements for an Election

- Voter validation (authenticated, registered, has not yet voted)
- Ballot validation (voter uses right ballot, results of marking capture intent of voter)
- Voter privacy (no association between voter, ballot; includes voter showing others how he/she voted)
- Integrity of election (ballots not changed, vote tallied accurately)

More Requirements

- Voting availability (voter must be able to vote, materials must be available)
- Voting reliability (voting mechanisms must work)
- Election transparency (audit election process, verify everything done right)
- Election manageability (process must be usable by those involved, including poll workers)

Add In e-Voting

- System must meet state certification requirements
  - Usually these incorporate the FEC standards
- Systems used must be certified
- Systems must be available on Election Day
  - No re-runs allowed!
- Systems must be secure
  - Properties must hold in face of (limited) conspiracy to undermine them

Federal Standards

- Performance and Test Standards for Punchcard, Marksense, and Direct Recording Electronic Voting Systems (1990)
- Voting Systems Performance and Test Standards (2002)
- Voluntary Voting Systems Guidelines (2005)
  - Take effect Dec. 2007

Why Standards?

- If systems are certified to meet standards, then people can have confidence they work!
- Two questions here:
  - How good are the standards?
  - How good is the testing?

Current Standards

- Goal: "address what a voting system should reliably do, not how system components should be configured to meet these requirements"
- Security concerns that have been raised:
  - System integrity during build and deployment, voter anonymity, access control policies, availability, poor design and implementation, data transmission, language, basis unclear

System Integrity

- No procedural mechanisms required to ensure the software submitted for qualification is the exact software used in production units
- Integrity of ROMs must be validated before each election
- No requirement that integrity be maintained throughout election

Consequences

- Several California counties used uncertified software
  - Diebold downloaded last-minute fixes just before the election
- This happened elsewhere (Indiana, Colorado, *etc.*)
- Last-minute bugs *cannot* be fixed until patched software recertified

Voter Anonymity

- Audit trail records time of each vote
  - This allows you to reconstruct sequence of votes
  - Combine with observation and you may be able to tie voters to votes
- Potential problem with the way most ballots are recorded on VVPATs, which usually are reels

Access Control Policies

- Vendor recommends policies and describes mechanisms to enforce them
  - "permit authorized access to the system", "prevent unauthorized access", "provide effective voting system security"
- Example: access to a locked room
  - Vendor must list everyone with a key
  - Vendor need not describe how to handle duplication of keys, changing locks, or who or when those things can be done

Problems

- Locks on bays holding memory cards
  - AccuVote-TS: same lock on all systems; other keys work well (VAX panel keys?!?)
  - Diebold, ES&S: hard-coded passwords gave supervisor rights to anyone who knew keys

Availability

- Required: MTBF/(MTBF+MTTR) ≥ 0.99
  "during normal operation for the functions indicated above"
  - Reliability: measure MTBF over at least 163 hours
  - Mathematical model to predict availability (vendor); validate model (testing authority)

Problems

- Testing done under laboratory conditions
  - Actual conditions of use may be different
  - Physical attacks like yanking wires of jamming cards typically not tested
- Availability models are problematic
  - Method of validating model not specified; up to tester

Poor Design, Implementation

- Systems may feature unnecessary hardware, software, or software known to be vulnerable
- Examples
  - Wireless cards allowed (some states differ)
  - USB ports allowed (enabling booting alternate system)
  - Memory cards containing programs allowed

Data Transmission

- During transmission of vote data, DRE must be authenticated
  - But the server need not be … man in the middle
- Methods for handling external threats to telecommunications network must be documented
  - Here, encryption standards must be used to detect intrusive devices and/or processes

Vague Language

- "Prohibit the voter from accessing or viewing any information on the display screen that has not been authorized by election officials and preprogrammed into the voting system"
  - Is changing the order of authorized content an attack? Change order in ballot definition file on DRE but not on counting system …
  - What does "preprogrammed" mean?
  - How does system determine if content "authorized"?

Another Example

- Standards imply roles (installer, troubleshooter, voter) without details
  - Access control policies controlling the interaction of roles and systems left to vendor's discretion

Unclear Basis

- Some numbers given but not explained
- Example: "achieve a target error rate of no more than one in 10,000,000 ballot positions"
  - Why this? Why not 1,000,000 or 100,000,000?
- Determine MTBF over 163 hours of testing
  - Again, why 163? Why not 14, or 48?

Lack of Threat Model

- Key question: against what threats should the systems be protected?
  - Standards silent on this model
- Without it, basis for many requirements unclear and requirements themselves vague

Lack of System Model

- Key question: in what environment, and under what processes, will the system be used?
  - Standards also silent on this model
- Problem leads to vague requirements about processes, procedures, assumptions

Conclusion: Standards

- Not very convincing because:
  - It's not clear how systems meeting the standards fir into the election process
  - It's not clear *exactly* what the conditions the systems meet because the standards are unclear
  - The standard doesn't present a threat model, nor was a threat model used in its development

Testing for Conformance

- Testing performed by independent testing authorities (ITAs)
  - Vendors pay for testing
  - Vendors can choose any ITA certified as such
  - Testing methodology up to ITA

ITAs Not Rigorous

- Example: Diebold AccuVote-TS certified (2003)
- RABA study used machines taken from State of Maryland; these would be used in election 6 weeks away
  - Lots of flaws found including …
    - Switch PCMCIA cards to load fake ballots or new programs (à la Hursti)
    - Default keys embedded in software (and available on the Internet)

More Evidence

- LaPorte County, IN: voting system software patched but not certified (2004)
- Miami-Dade County, FL: audit trail overwhelmed central servers (2004)
- San Diego County, CA: machines failed; no paper or provisional ballots available (2004)
- Alameda County, CA: machines failed; provisional ballots available and used (2004)

Diebold AccuBasic

- Programming language used to write scripts in a report writing facility on the AccuVote-OS optical scan and AccuVote-TSx DREs
- Required to verify that "not possible to compromise an election in any way through the (mis)use of AccuBasic, including an unintentional error or malicious AccuBasic script" (request for ITA review)

ITA

- Three violations allow manipulation, reading data in global space but can only be exploited by modified AccuBasic object file
- Bounds checking on stack, heap segments not detected, but bounds checking performed inside the code
- Interpreters lack proper degree of error checking to identify, recover from key failures in damaged environment

ITA Summary

- "Three security vulnerabilities and a small number of requirements violations that were not capable of being exploited by malicious code or operators"
- TSx ready for election
- AV-OS needs to have these problems corrected
- If memory cards not tampered with between AV-OS and GEMS, existing units ready for election

VSTAAB Independent Review

- Asked questions:
  - What kind of damage can malicious person do to undermine election if he can arbitrarily change contents of memory card?
  - How can such attacks be neutralized?
- Code problems:
  - Buffer overflows (12 in AV-OS, 8 in TSx)
  - Other problems (4 in AV-OS, 2 in TSx)

VSTAAB Summary

- 16 security problems in AV-OS, 10 in TSx
  - All code problems, easily fixed
- If you can tamper with memory cards, you can undetectably rig election
- TSx has memory cards digitally signed … using keys for which defaults are hard-coded
- Interpreters disallowed by FEC standards!

Result

- ITA clearly missed a lot
- Report of ITA is not particularly detailed; VSTAAB report is very detailed

Another Question

- How can we measure e-voting systems to see how secure they are?
  - Also allows us to compare systems from different vendors

Process vs. Machines

- Machine is component of process
  - Policies, procedures can be designed to mitigate/ eliminate threats from machines
- Do we measure qualities, properties of machine or process?
  - Most work focuses on machine
  - Some work focuses on process

Consistency

- Differing jurisdictions require different measures
  - Maryland can revote *precincts* if problems arise (court order only?)
  - California cannot; State Supreme Court can order *entire statewide election* rerun
- How does this affect the measurement of California's and Maryland's processes?

Certification

- Need to trust evaluators
  - ITAs don't seem to be doing as good a job as they should
- Need to certify to meaningful standards
  - Standards lack threat, system models; mix functional, testing requirements
  - Standards certify machines, not processes; processes can weaken secure systems

Usability

- *Critical to security*
  - Especially important here as *many operators will be computer-illiterate* or *non-technical* and employed only for one day(poll workers)
- Secure systems operated non-securely are non-secure (to put it mildly)

Transparency

- Must be as clear to voters as current system
- Anyone can observe *every* step of election except:
  - With DREs, cannot observe tallying of votes at per machine level
    - May be possible at per precinct level
  - With paperless DREs, cannot verify those tallies either

What's the Question?

- Not "how secure is this system"
- Right question will have several parts:
  - What properties do I care about?
  - What is the ideal for those properties (taken as a whole)?
  - How close to that ideal can we come?
  - How do we convince others that our measurements are good?

Conclusion

- We need to think in terms of *elections that use e-voting machines* and not about e-voting machines
  - Measures must take target environment into consideration
  - View the election process holistically

Conclusion

- We need to examine voting systems with respect to requirements of the jurisdiction using them
- We need to *design and build e-voting systems in such a way that we can analyze (and, if appropriate, measure) security properties*

Closing Thought

To those accustomed to the precise, structured methods of conventional system development, exploratory development techniques may seem messy, inelegant, and unsatisfying. But it's a question of congruence: precision and flexibility may be just as dysfunctional in novel, uncertain situations as sloppiness and vacillation are in familiar, well-defined ones. Those who admire the massive, rigid bone structures of dinosaurs should remember that jellyfish still enjoy their very secure ecological niche.

— Beau Sheil, "Power Tools for Programmers"