Computers and Elections

The Good, the Bad, and the Ugly

Matt Bishop
joint work with many students and colleagues

University of California at Davis

February 11, 2011

1 About Voting and Computers

2 Federal Voting Standards and Problems
- Standards
- Testing

3 Example: California Top-to-Bottom Review

4 Process Modeling
- Analyzing an Election Process
- Internet Voting

5 Conclusion

This Is Not About . . .

- Voting algorithms

This Is Not About . . .

- Voting algorithms
  - I recommend a good class on distributed algorithms or computing

This Is Not About . . .

- Voting algorithms
  - I recommend a good class on distributed algorithms or computing
- Different voting schemes like choice voting

This Is Not About . . .

- Voting algorithms
  - I recommend a good class on distributed algorithms or computing
- Different voting schemes like choice voting
  - There are lots of them, from the merely confusing to the downright mysterious

This Is Not About . . .

- Voting algorithms
  - I recommend a good class on distributed algorithms or computing
- Different voting schemes like choice voting
  - There are lots of them, from the merely confusing to the downright mysterious
- Who will win the next election?

This Is Not About . . .

- Voting algorithms
  - I recommend a good class on distributed algorithms or computing
- Different voting schemes like choice voting
  - There are lots of them, from the merely confusing to the downright mysterious
- Who will win the next election?
  - I'm a scientist, not a psychic!

Over- and Under-Votes

- Three seats open in Davis City Council election
- **Overvote**: voting too many times
  - Vote for 4 candidates
  - No votes in that race counted
- **Undervote**: voting too few times
  - Vote for 2 candidates
  - Both votes counted; no third vote counted

How an Election Works in Yolo County, CA

Voters:

- Go to polling station
- Give name, get ballot
- Enter booth, vote using marker to mark ballot
- Put ballot in protective sleeve (envelope)
- Leave booth, drop envelope into ballot box

End of the Day

- Election officials take ballot box to County seat
- Election officials remove ballots from envelopes
  - If provisional, handled differently
- Ballots counted, put into bags marked with precinct and count
- Ballots removed from bag, run through automatic counters (scanners)
  - Humans intervene when problems arise
  - Intermediate tallies written onto flash cards
  - Every so often, cards removed, walked to tally computer
- Tallies periodically updated, given to web folks

The Canvass

Required by California law:

- Ballots for 1% of precincts counted by hand
  - Must include all races!
- Compare to tallies from election
  - If different, check until problem found
- Certify final counts to Secretary of State
  - . . . within 28 days of the election

Actually, Yolo County also does more checking, including testing other proposed auditing methods with trusted researchers

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
  - Direct Recording Electronic (with or without VVPATs)
  - Ballot Marking Devices
  - Pens and paper
- Counting votes
  - Scanning at precinct (Precinct-Count Optical Scan)
  - Scanning at Election Central
  - Computer counting of electronic ballots

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

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)

Requirements for an Election

- 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

Assurance

- 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*

Standards

Each state sets its own; most based on 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)
  - Took effect Dec. 2007
- New ones under development (time frame uncertain)

Why Standards?

If systems are certified to meet standards, then people can have confidence they work!

- 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, including:
  - System integrity during build and deployment
  - Voter anonymity
  - Access control policies
  - Availability
  - Poor design and implementation
  - Data transmission
  - Language
  - Unclear bases

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

- In 2006, several California counties used uncertified software
  - Diebold downloaded last-minute fixes just before an election
- Also happened in other states such as Indiana and Colorado

Availability

- Required: $\frac{MTBF}{MTBF+MTTR} \geq 0.99$ "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

Unclear Bases

- 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

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
- Without it, vague requirements about processes, procedures, assumptions

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

Diebold AccuBasic

- Intent: add a scripting language to a report writing facility on the AccuVote-OS optical scan and AccuVote-TSx DREs
- CA required that it be "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 Findings

- 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 Findings

- "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

Led by David Wagner of UC Berkeley

- 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?
- Found code problems:
  - Buffer overflows (12 in AV-OS, 8 in TSx)
  - Other problems (4 in AV-OS, 2 in TSx)

VSTAAB Findings

- 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!

Summary

- ITA clearly missed many problems
- ITA report not very detailed ($\sim$ 5 pages); VSTAAB report very detailed ($\sim$ 33 pages)

CA Top-to-Bottom Review

Undertaken to "restore the public's confidence in the integrity of the electoral process and to ensure that California voters are being asked to cast their ballots on machines that are secure, accurate, reliable, and accessible."

Structure

- UC teams provided technical data for CA Secretary of State
  - UC Berkeley (Wagner): source code review, document review
  - UC Davis (Bishop): red team testing, accessibility testing
  - Both groups used people from around the country
- Secretary used this data and other data to make decision
  - Policies, procedures, and their implementation
  - Each county has its own

Goals of the Study

"to identify and document vulnerabilities, if any, to tampering or error that could cause incorrect recording, tabulation, tallying or reporting of votes or that could alter critical election data such as election definition or system audit data."
Assume attackers could be anyone (voters, poll workers, election officials, vendors, etc.)

Constraints

- Time
  - Exercise lasted 5 weeks for 3 vendors (ended July 20)
- Lack of information and vendor software
  - Some documents delivered on July 13
  - Some software delivered on July 18
- Secretary, staff *exceptionally* supportive throughout

Example Threats

Attacker modifies "firmware" to misrecord votes

Case 1: Paper trail modified to reflect misrecorded votes unless voter corrects it, so no discrepancies between paper and stored ballots

Case 2: Paper trail records correct vote, disagreeing with stored ballots, creating discrepancy

Results

"security mechanisms provided for all systems analyzed were inadequate to ensure accuracy and integrity of the election results and of the systems that provide those results"

Example: Diebold

- Election management server
  - Delivered unpatched
  - Not all security-related actions logged
  - Remotely accessible account that by default does not require password
  - GEMS users can conceal actions from GEMS logging
- Precinct count AccuVote-OS
  - Low-tech attacks to stop it from reading ballots

Example: Diebold

- AccuVote TSx
  - Physical security: bypass locks; disable printer
  - Firmware: overwritten; virus attack possible
  - Escalate privileges from voter to election official, and erase votes, close polls, etc.
  - Security keys: well-known key used as default
  - Malicious voter input: made machine act erratically (no time to craft working exploits)
  - Paper trail: can easily be put out of service; could destroy records before and after attack, in a way voters wont notice

What Secretary Bowen Did

- Diebold, Sequoia
  - Certification and approval for use withdrawn
  - 1 system per polling place (to comply with HAVA)
  - Vendors could fix problems and request recertification
- ES&S
  - Certification and approval for use withdrawn
  - ES&S could undergo testing
- Hart
  - Jurisdictions must reinstall all software and firmware on all systems before each election
  - Vendor must present procedures to prevent virus propagation and to harden system

Later Version: Diebold

- Diebold added cryptography in the version after the one California reviewed
  - Not examined in TTBR because it wasn't certified in California
- Florida did examine it as part of certification process
  - Led by Prof. Alec Yasinsac of Florida State University

The Crypto

Signature is a SHA-1 160-bit digest signed using RSA:

sign: write M, S_{2048}

where $S_{2048} = RSA(privkey, 0_{1888}|SHA1(M)_{160})$

The Crypto

Signature is a SHA-1 160-bit digest signed using RSA:

sign: write M, S_{2048}

where $S_{2048} = RSA(privkey, 0_{1888}|SHA1(M)_{160})$

verify: read M, S_{2048}

if $RSA(pubkey, S_{2048})_{160} = SHA1(M)_{160}$, accept M

The Crypto

Signature is a SHA-1 160-bit digest signed using RSA:

sign: write M, S_{2048}

where $S_{2048} = RSA(privkey, 0_{1888}|SHA1(M)_{160})$

verify: read M, S_{2048}

if $RSA(pubkey, S_{2048})_{160} = SHA1(M)_{160}$, accept M

But . . .

- *privkey* is 3
- Verify step above just checks the low-order 160 bits!

Summary

- Standards, testing are not enough
- You need to know what the systems are to do
- You need to know under what constraints they will need to function
  - Environment
  - Policies and procedures
- You need to know with what assurance you can trust the systems

Election Process

- Elections are a *process* composed of specific tasks
- Tasks related to one another
  - Temporal order (one must follow another)
  - Dependancy (output from one task used as input to another)
  - Exception handling (handling problems)
- Machines may perform these tasks

Continuous Process Improvement

1. Create a precise, accurate model of the real-world election process
2. Use formal analysis methods to automatically identify potential problems in the model
   - We focus on single points of failure
3. Modify process model to ameliorate problems
   - Verify the modification makes things better
4. Deploy improvements in real-world process
5. Repeat steps 2–4

Fault Tree Analysis

- Fault trees show how problems could arise
- Can automatically generate fault trees from process model and a hazard
  - Hazards are conditions under which undesired, possibly dangerous events may occur
- Analyze fault trees automatically to identify points of failure
  - Especially Single Points of Failure (SPFs)

Compute Cut Sets

- Combination of events such that, if all events in the cut set occur, the hazard occurs
  - Minimal if removal of any event causes the resulting set not to be a cut set
- Can be computed automatically from the fault tree

Three Effects

- Process
  - Change process to reduce number of SPFs
  - Gives changes to procedures to detect, handle failures
- Machine
  - Determine inputs to, outputs from particular tasks
  - Compare existing systems to existing process to find discrepancies

Assurance Issues

- Goal of e-voting system is to perform some task or set of tasks in the process
- How do you know it will correctly perform the task or tasks?
  - Take into account environment
  - Take into account how results are validated
  - Take into account the audience to be convinced, and to what degree of certainty

Internet Voting

- A generic term for many different possible ways to handle the casting and transmission of votes over the Internet
- First version: voter votes at home on a PC using a web browser connected to a server at Election Central
- Second version: voter votes at special kiosk that then transmits the votes to Election Central over the Internet
  - This is like the first, but the PC—the kiosk—is (essentially) trusted
  - So only talk about first

First Version: How to Do It

- PC transmits authentication information of voter to Election Central
- Election Central transmits ballot to PC
- PC displays ballot
- PC records vote
- PC transmits vote to Election Central server

First Version: How to Do It

- PC transmits authentication information of voter to Election Central
- Election Central transmits ballot to PC
- PC displays ballot
- PC records vote
- PC transmits vote to Election Central server

Every step can be compromised

First Version: How to Attack It

- PC transmits authentication information of voter to Election Central
  - PC contacts fake Election Central site
  - PC has a Trojan horse that constructs bogus data
  - User requests wrong ballot
- Election Central transmits ballot to PC
  - Ballot is a PDF with malicious content
  - Wrong ballot is sent
- PC displays ballot
  - Display does not match underlying ballot

First Version: How to Attack It

- PC records vote
  - User cannot cast vote for desired candidates, races
  - Displayed votes on ballot do not match votes stored in computer
- PC transmits vote to Election Central server
  - PC cannot contact Election Central
  - PC again contacts fake Election Central site
  - PC sends incorrect votes to EC
  - Attacker intercepts ballot in transit, either deletes it or changes it
- Software, hardware maybe compromised by vendors, third parties

Server at Election Central

- As is on the Internet, *anyone* can access it
- Standard server side technology riddled with holes
  - Need to write your own server *from scratch*
- Even if server carefully written, relies on flawed libraries, operating systems, and network infrastructure
- Small configuration errors may create gaping vulnerabilities
- Procedures and policies may also cause security problems
- Attacker only needs to find one problem

Bottom Line

- NASDAQ, Pentagon, government sites regularly penetrated
- If those experts cannot stop compromises, why should we assume election servers will be invulnerable?

Bottom Line

- NASDAQ, Pentagon, government sites regularly penetrated
- If those experts cannot stop compromises, why should we assume election servers will be invulnerable?

Key Question:
as a citizen and a voter, are you comfortable that your vote will not be altered or discarded undetectably?

Conclusion

- Security should be part of the design and implementation of the system and not added on "after the fact"
- Policies and procedures should be either designed with, or drive the design of, the system as it is being designed and implemented

Acknowledgements

- Process modeling work done with Prof. Lee Osterweil, Prof. Lori Clarke, and their graduate students at UMass Amherst, and our graduate students and post-doc at UC Davis
  - Funding provided by NSF grant CCF-0905530
- CA TTBR co-led by David Wagner
  - Team leaders Bob Abbott, Matt Blaze, Joseph Lorenzo Hall, Candice Hoke, Dick Kemmerer, Deirdre Mulligan, Eric Rescorla, Noel Runyan, Giovanni Vigna
- Special thanks to Yolo County, CA Clerk-Recorder Freddie Oakley and Chief Deputy Tom Stanionis

Contact Information

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

email: bishop@cs.ucdavis.edu
web: http://seclab.cs.ucdavis.edu/~bishop
phone: (530) 752-8060