

Requirements for Electronic and Internet Voting Systems in Public Elections

David Jefferson

Compaq Systems Research Center

Palo Alto, CA

david.jefferson@compaq.com

August 27, 2001

Voter authentication

- **Verification that voter is legally eligible to vote,**
 - S/he is the correct living human corresponding to an entry in the voter registration database
 - No proxy voting; the right to vote never transferable
- **... and votes using the proper ballot,**
 - i.e., is in the right precinct
- **... and has not voted yet in the current election.**
 - no multiple voting
- **A person who cannot be authenticated at poll site has a right to a provisional ballot**
 - Whether ballot is eventually accepted or not, full privacy guarantees must be preserved.

Accurate capture of voter intent

- Votes to be captured digitally
 - not in analog form, transduced to digital in a 2nd step
 - must be absolutely no ambiguity in what the ballot says
 - no need for human judgment (except for write-ins)
- Voter gets clear feedback: chosen candidates are highlighted on screen
- Voter gets opportunity to revise votes before committing.
- System must prevent overvotes
- Voter gets opportunity to confirm votes, especially undervotes, before committing

Privacy

- **What must be really private?**
 - Association between voter and his or her vote to within a precinct level of aggregation
 - here should exist no proof of how a person voted, even if the voter (or a court) wants proof (except in boundary cases)
- **What is less private?**
 - Who is registered to vote in what jurisdictions
 - Who votes and who does not in each election
 - What time a person votes (in real time)
 - What mode of voting a person chooses (e.g. absentee)
 - Actual full ballot images
 - allows correlations between races
 - allows reading of write-in field (which can be used to mark a ballot)

General Security Requirements

- No loss of votes already cast (reliability)
- No forging of votes (authentication)
- No modification of votes cast (integrity)
- No multiple voting
- No uncertified software in critical equipment (software authentication)
- No vote secrecy violation (privacy)
- No vulnerability to vote coercion
- No vulnerability to vote selling or trading protocols (receipt freedom, voter is an adversary)
- No loss of ability to cast and accept more votes (availability, no denial of service)

Potential adversaries

- **Voters**
 - may conspire in double voting or in vote selling schemes
- **Vendor and/or election official conspiracy**
 - including programmers!
- **Outside parties or organizations**
 - remote and/or programmed attacks via the Internet
- ***Probably should not include certifiers as adversaries***
 - hopefully open source and software simplicity can allow the assumption that certified software is trustworthy

Tools available to adversaries

- **The subset of crypto keys controlled by conspirators**
- **High computational and bandwidth resources**
- **Read access to voter registration database**
 - But not ability to modify it
- **Copies of all software, hardware, documentation, etc.**
 - Should be no secrets except crypto keys.
- **Some ability to modify or substitute for critical software**
 - The problem is to detect this before it does harm
- **Ability to monitor communications lines, inject traffic, reroute Internet communication, and do so arbitrarily and in real time**
- **Exploitation of security holes in any software for which there is no good argument why it might be considered secure**

Software Certification and Authentication

- Reasonable verification of correctness, robustness, security properties of the critical software be during certification
 - Implementation of algorithms with provable properties
 - Small amount of software
 - Use, insofar as possible, published, open protocols and security algorithms;
 - Open source
 - Ken Thompson's diabolical Trojan horse problem?
- Software authentication – verification that the software running during the election is the actual certified software
- Continuous software certification and decertification – when security standards change, or holes are recognized, voting systems must be able to be decertified

Auditability / verifiability

- Election systems must not only *be* correct and secure, but *be seen to be so* by skeptical (but educated and honest) outsiders.
- **Auditability:** There should be enough redundant information saved about the election that essentially any *random* failure or procedural error can be detected and corrected, especially the loss of votes.
- **Verifiability:** There should be enough redundant information saved about the election so that, barring large conspiracies, *proofs* of such statements as the following are possible:
 - “My vote was counted”
 - “All precincts were counted”
 - “The number of votes in each precinct is the same as the number of people who voted”
 - “No one I know who is ineligible to vote did so”
 - “No one voted twice”
 - “No votes in the canvass were forged or modified”
 - Etc.

Special hazards of “remote” Internet voting

- **Local attacks**

- SysAdmin / remote management attacks in institutional settings (employers, old-age facilities, universities)
- On-screen political advertising / electioneering
Pop-up ads perhaps sold by ISP appear during voting!

- **Automated or remote attacks, possible from anywhere, including foreign countries**

- Trojan horse attacks on vote clients
steals votes undetectably
Vast number of ways to accomplish it
- Internet infrastructure attacks, i.e. various kinds of spoofing
steals votes undetectably by acting as man-in-the-middle
- Server penetration attacks (exploiting security bugs, misconfigurations, etc.)
- Denial of service attacks against server
disenfranchises people

Special hazards of remote Internet voting

- **Automated or remote attacks, possible from anywhere, including foreign countries--
continued**

- Large-scale automated vote-selling and trading schemes

- Think “Vote-ster”

- Vote hijacking

- “Vote for me now, while you are thinking of it—just [click here](#)”

End