Developing an Analysis of Threats to Voting Systems: Preliminary Workshop Summary

Hilton Washington DC North Hotel
Gaithersburg, Maryland
October 7, 2005

THIS PAGE LEFT INTENTIONALLY BLANK

Developing an Analysis of Threats to Voting Systems: Workshop Summary

Hilton Washington DC North Hotel , Gaithersburg, Maryland
October 7, 2005

FOREWORD

The Help America Vote Act (HAVA) of 2002 has given NIST a key role in helping to realize nationwide improvements in voting systems by January 2006. NIST research activities authorized by HAVA include the security of computers, computer networks, and computer data storage used in voting systems, methods to detect and prevent fraud, and protection of voter privacy and the role of human factors in the design and application of voting systems. Complete details of NIST voting research are available at http://www.vote.nist.gov.

The National Institute of Standards and Technology (NIST) hosted a workshop to allow the U.S. election community to participate in developing an analysis of threats to voting systems. The workshop took place on October 7, 2005, at the Hilton Washington DC North in Gaithersburg, Maryland.

The goal of the workshop was to solicit and gather threat analysis material and critical analysis of the collected threats; assess the plausibility of various scenarios and assumptions made; and extract lessons learned as a result of the analysis.

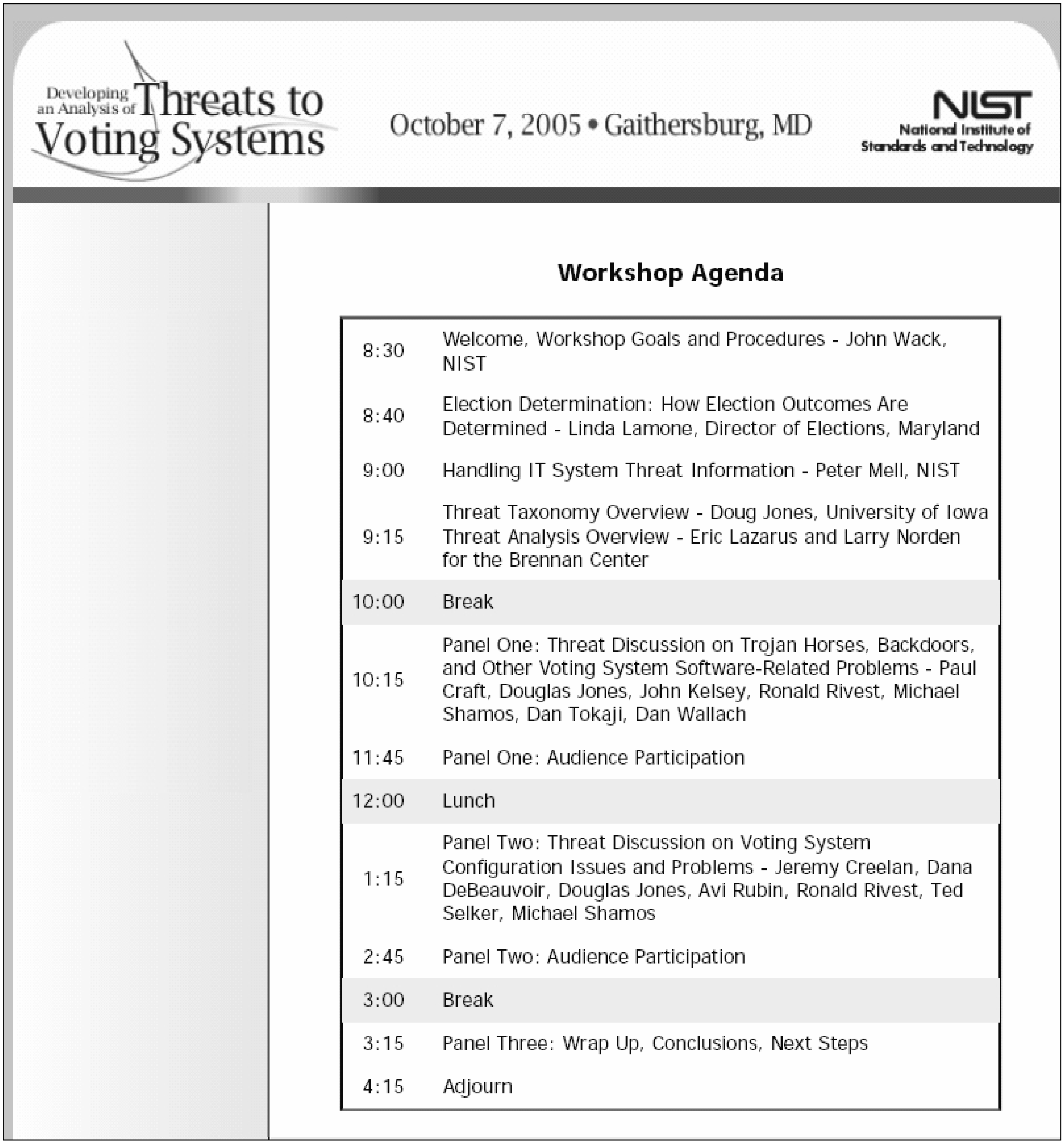
State and local election directors and officials, voting system security researchers, election lawyers, threat analysis experts, voting system vendors, and others from the public and private sectors submitted threat analyses of voting systems and participated in the workshop.

This workshop summary includes a synopsis of invited presentations and panel discussions as well as audience comments and questions. Audio recordings of the workshop proceedings served as the basis for panelist and presenter comments summarized herein. (Editor’s note: Best efforts have been made to paraphrase the remarks of all participants. The positions expressed are solely those of the presenter, panelist, or audience participant. Full audio transcriptions of the workshop are posted at: http://vote.nist.gov/threats/audio.htm.)

Threat Analyses papers referenced in the workshop are included as an appendix. NIST encourages the election community to continue the threat analyses dialog begun at the October workshop. Papers and comments will be posted on the workshop web page: http://vote.nist.gov/threats/submissions.htm. Submissions can be made directly to voting@nist.gov.

THIS PAGE LEFT INTENTIONALLY BLANK

TABLE OF CONTENTS



Developing an Analysis of Threats to Voting Systems

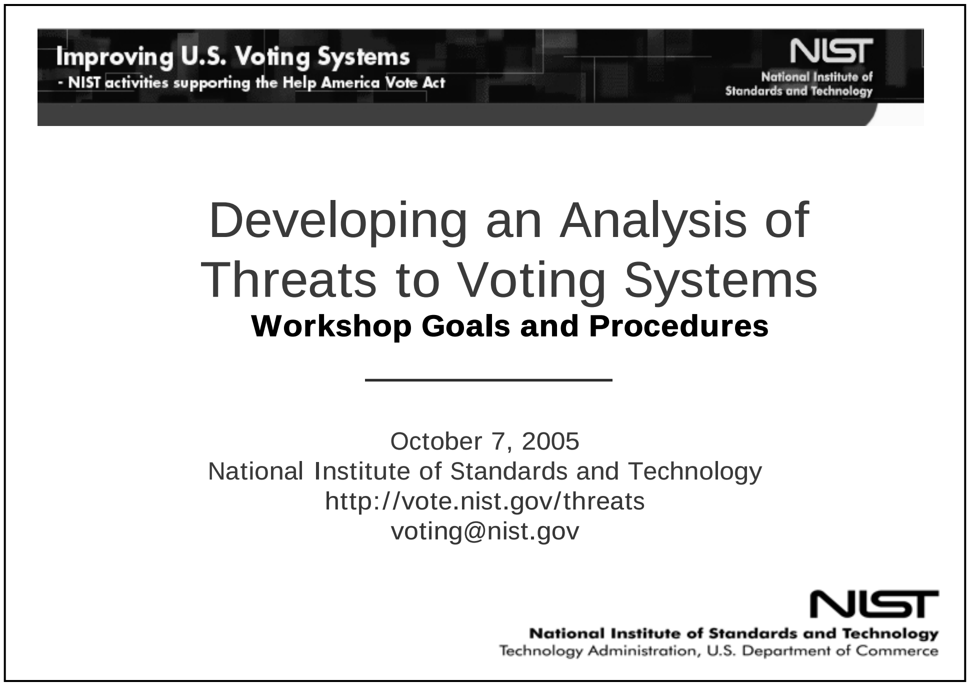
October 7, 2005 • Gaithersburg, MD

NIST National Institute of Standards and Technology

Workshop Agenda

8:30	Welcome, Workshop Goals and Procedures - John Wack, NIST
8:40	Election Determination: How Election Outcomes Are Determined - Linda Lamone, Director of Elections, Maryland
9:00	Handling IT System Threat Information - Peter Mell, NIST
9:15	Threat Taxonomy Overview - Doug Jones, University of Iowa Threat Analysis Overview - Eric Lazarus and Larry Norden for the Brennan Center
10:00	Break
10:15	Panel One: Threat Discussion on Trojan Horses, Backdoors, and Other Voting System Software-Related Problems - Paul Craft, Douglas Jones, John Kelsey, Ronald Rivest, Michael Shamos, Dan Tokaji, Dan Wallach
11:45	Panel One: Audience Participation
12:00	Lunch
1:15	Panel Two: Threat Discussion on Voting System Configuration Issues and Problems - Jeremy Creelan, Dana DeBeauvoir, Douglas Jones, Avi Rubin, Ronald Rivest, Ted Selker, Michael Shamos
2:45	Panel Two: Audience Participation
3:00	Break
3:15	Panel Three: Wrap Up, Conclusions, Next Steps
4:15	Adjourn

Figure 1



Improving U.S. Voting Systems
- NIST activities supporting the Help America Vote Act

NIST National Institute of Standards and Technology

Developing an Analysis of Threats to Voting Systems

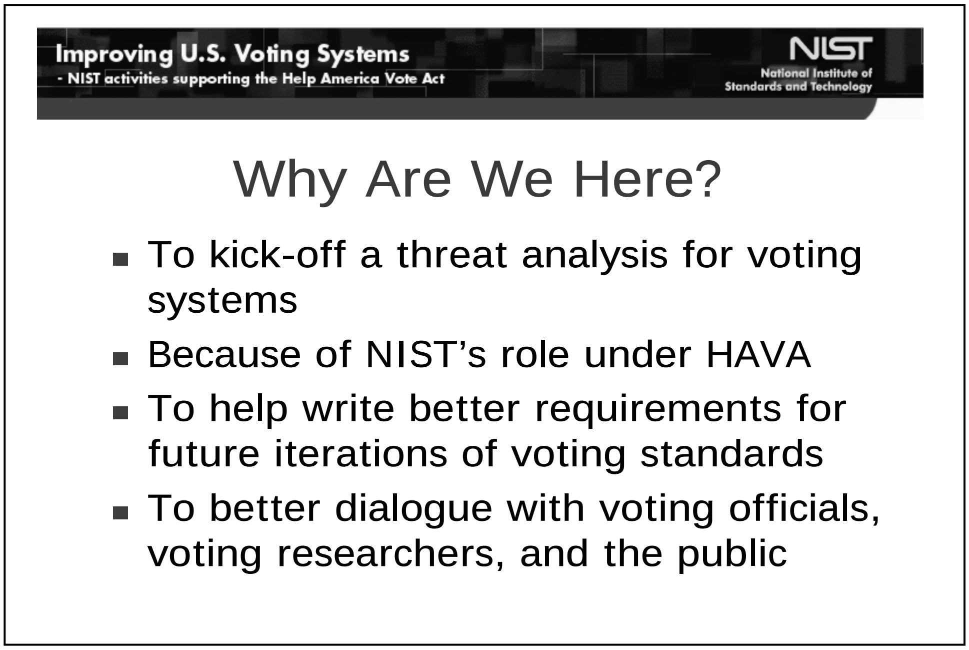
Workshop Goals and Procedures

October 7, 2005
National Institute of Standards and Technology
http://vote.nist.gov/threats
voting@nist.gov

NIST
National Institute of Standards and Technology
Technology Administration, U.S. Department of Commerce

John Wack and Mark Skall, Information Technology Laboratory, NIST

As an introduction, John Wack of NIST's Information Technology Laboratory (ITL) reviewed the agenda for the workshop (Figure 1). He noted that the audience would have thirty-minute opportunities for participation after each panel discussion. Attendees were encouraged to ask questions and submit statements to the NIST web site.



Improving U.S. Voting Systems
- NIST activities supporting the Help America Vote Act

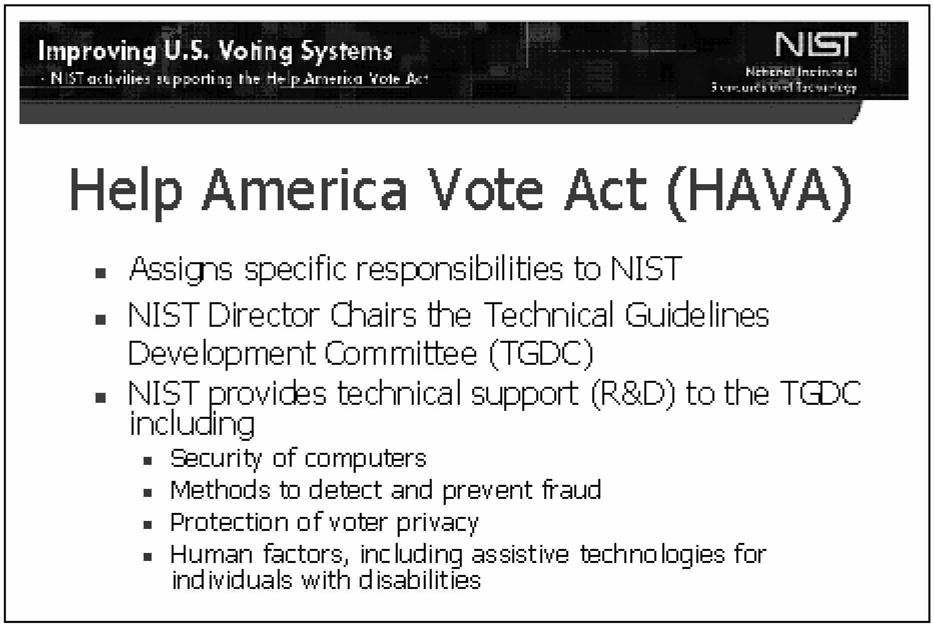
NIST National Institute of Standards and Technology

Why Are We Here?

- To kick-off a threat analysis for voting systems
- Because of NIST's role under HAVA
- To help write better requirements for future iterations of voting standards
- To better dialogue with voting officials, voting researchers, and the public

Figure 2

Mark Skall, Chief of ITL's Software Diagnostics and Conformance Testing Division, provided context for the workshop (Figure 2) and reviewed NIST's role under the Help America Vote Act (Figure 3). Specifically, NIST is engaged through the Technical Guidelines Development Committee (TGDC) in assisting the U.S. Election Assistance Commission (EAC) with technical guidance to write better requirements for future updates to the voluntary voting system guidelines (VVSG). A major element in the next iteration of the standards will be security requirements. It makes sense to define the problem before you engineer the solution. A threat analysis is a critical step towards defining the problem (Figure 4). Mr. Skall also noted that NIST and TGDC members viewed the workshop as a means to maintain a dialogue with the election community on threats to voting systems and to reach consensus where possible.



Improving U.S. Voting Systems

- NIST activities supporting the Help America Vote Act

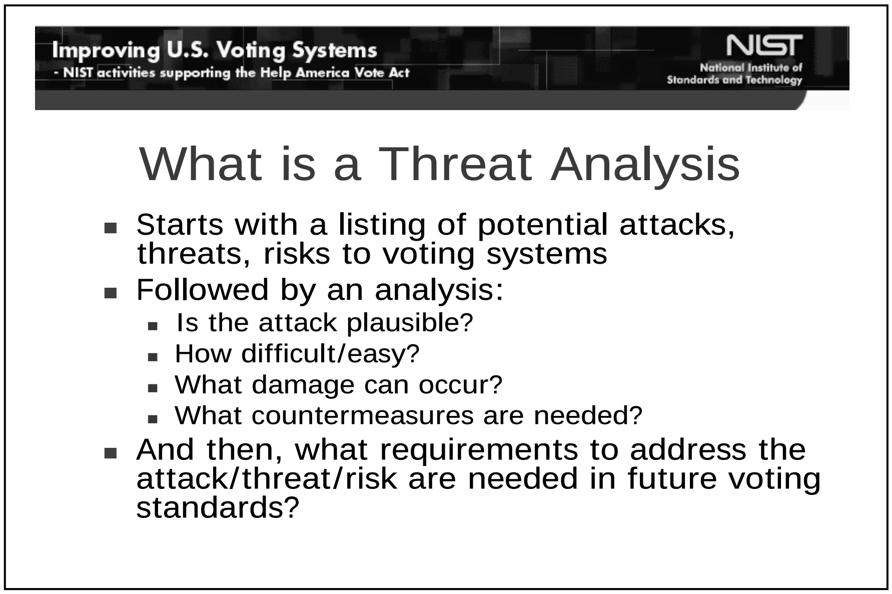
NIST

National Institute of Standards and Technology

Help America Vote Act (HAVA)

- Assigns specific responsibilities to NIST
- NIST Director Chairs the Technical Guidelines Development Committee (TGDC)
- NIST provides technical support (R&D) to the TGDC including
  - Security of computers
  - Methods to detect and prevent fraud
  - Protection of voter privacy
  - Human factors, including assistive technologies for individuals with disabilities

Figure 3



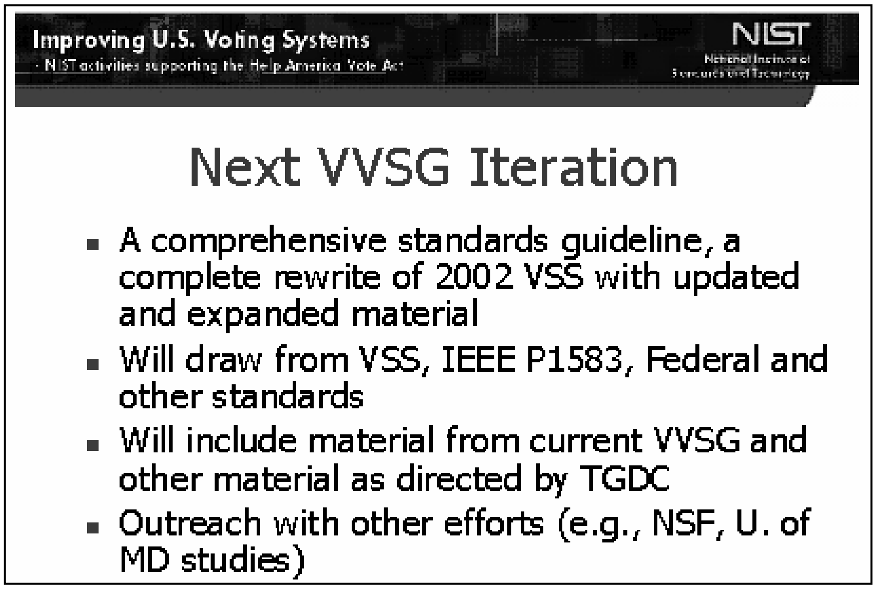
Improving U.S. Voting Systems
- NIST activities supporting the Help America Vote Act

NIST
National Institute of Standards and Technology

What is a Threat Analysis

- Starts with a listing of potential attacks, threats, risks to voting systems
- Followed by an analysis:
  - Is the attack plausible?
  - How difficult/easy?
  - What damage can occur?
  - What countermeasures are needed?
- And then, what requirements to address the attack/threat/risk are needed in future voting standards?

Figure 4



Improving U.S. Voting Systems
- NIST activities supporting the Help America Vote Act

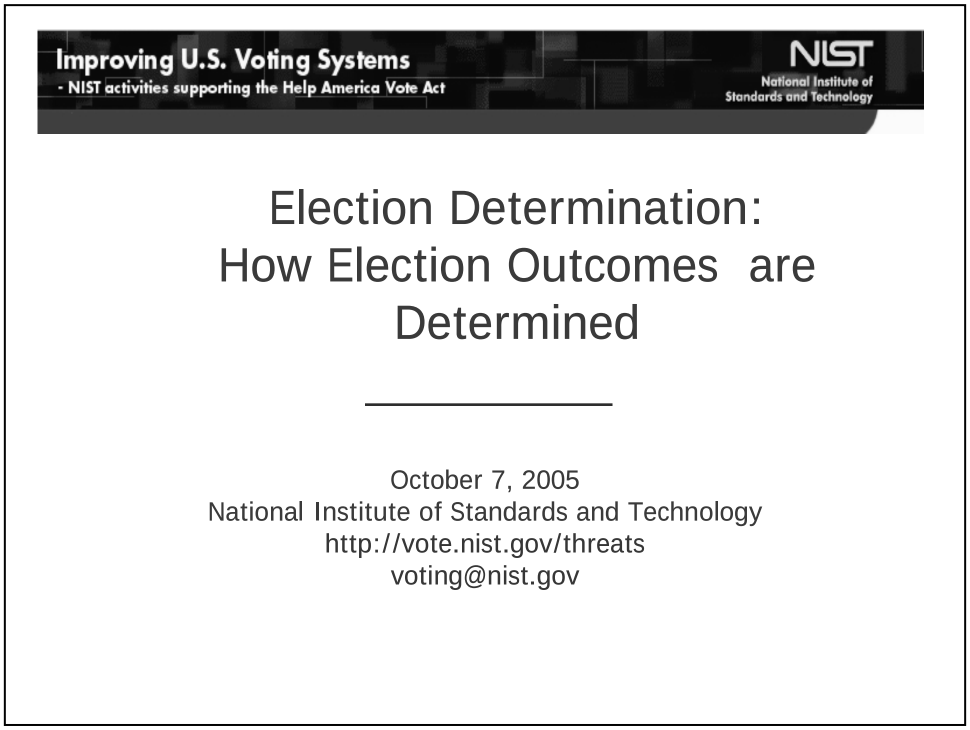
NIST
National Institute of Standards and Technology

Next VVSG Iteration

- A comprehensive standards guideline, a complete rewrite of 2002 VSS with updated and expanded material
- Will draw from VSS, IEEE P1583, Federal and other standards
- Will include material from current VVSG and other material as directed by TGDC
- Outreach with other efforts (e.g., NSF, U. of MD studies)

Figure 5

Mr. Skall noted that there was little time to engage in outreach to the election community during the development of the first iteration of the VVSG due to the strict time frame imposed under HAVA. NIST plans to engage in more comprehensive outreach with the development of the next iteration of the VVSG (Figure 5). Security requirements developed in the deliberative process of the TGDC may be onerous in terms of cost and time to implement. It will be important to inform the TGDC of the plausibility of threats. This threat analysis workshop begins that effort.



Improving U.S. Voting Systems
- NIST activities supporting the Help America Vote Act

NIST
National Institute of Standards and Technology

Election Determination:
How Election Outcomes are
Determined

October 7, 2005
National Institute of Standards and Technology
http://vote.nist.gov/threats
voting@nist.gov

Linda Lamone, Director of Elections, State of Maryland; President, National Association of State Election Directors (NASED)

Ms. Lamone noted that the State of Maryland has entered into a contract with the University of Maryland Baltimore County (UMBC) Institute for Policy Analysis and Research to conduct a study of the Diebold voting systems in Maryland to determine whether additional security measures are required and whether additional verification methods are needed. The reason for the study is to provide state legislators with facts on which to base security decisions and not assumptions. Many security papers and analyses are based on assumptions and not facts. The goal of the academic study is a scientific result as well as policy recommendation for the state. Another outcome could be an academic center for the study of voting systems at the UMBC campus.

The State of Maryland voting system has undergone two comprehensive security studies as a result of a Johns Hopkins University paper on the voting system source code security weaknesses. The studies resulted in the implementation of a large number of new security measures. All of the changes are meant to ensure the integrity of the state's voting system. Parallel testing is now part of Election Day procedures as well as county logic and accuracy testing.

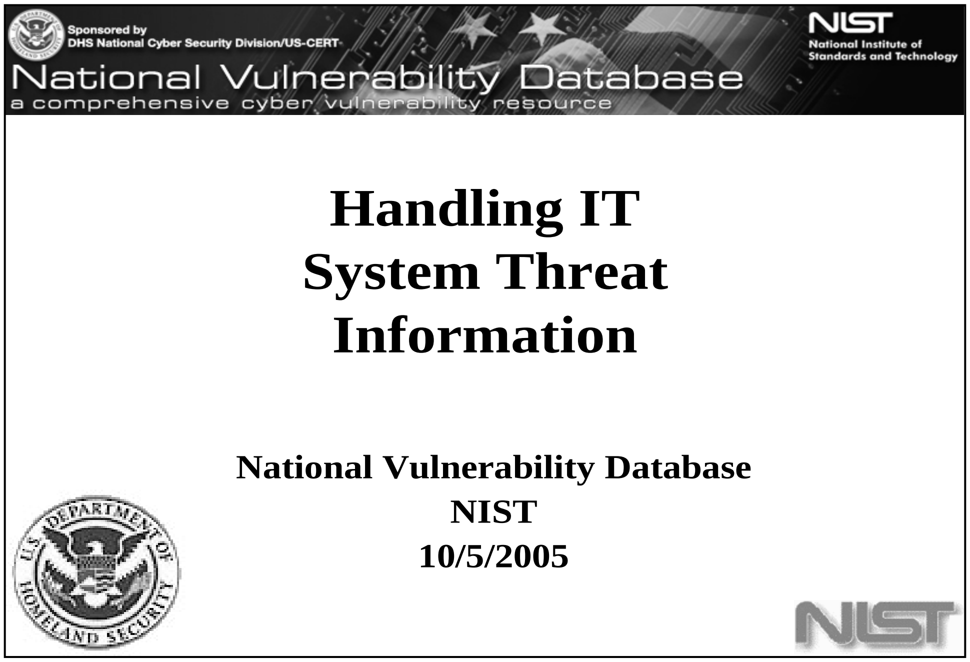
Maryland ensures that bipartisan election workers are part of the entire monitoring process with the voting systems. At the end of the day when the voting machines are closed down, the bipartisan workers sign printouts that indicate how many votes were cast on each machine. The tapes and the zero count tapes are publicly posted usually the next morning. Some jurisdictions in Maryland accumulate the PC memory cards from each voting unit onto one unit to aggregate

the results and send them to the county office by modem. Data encryption protocols are used. Other jurisdictions have bipartisan poll workers remove the PC cards and physically drive them to the county election office where the results are accumulated. In either case, these “election night” results are unofficial.

Unofficial election results are accumulated because the media and candidates want them and they are posted on the state's public web page. State and local election officials ideally would like to wait until official election results can be tabulated, a process that extends from a few days to a few weeks. In Maryland, policies and procedures exist to ensure that the integrity of the (official) election results is maintained.

The morning after the unofficial election returns have been disseminated, one hundred percent of the voting machine's memory cards are re-read into the GEM server. The GEM server is programmed to know whether a memory card is correct or not through an electronic handshake. In addition, the absentee and provisional ballot count begins. An audit of the election takes place at this time as well. Since 2002, using the Diebold Direct Recording Electronic (DRE) voting systems, each audit has resulted in perfect matches between machine counters, PC cards, and voting day polling place registration records.

In conclusion, Ms. Lamone requested that the ensuing threat analysis discussions be factually based and not rely on assumptions. She also noted that electronic voting has been in use for many years in the United States without any documented instance of voting equipment failure. The documented failures within the voting process have been human failures.



Sponsored by
DHS National Cyber Security Division/US-CERT

National Vulnerability Database
a comprehensive cyber vulnerability resource

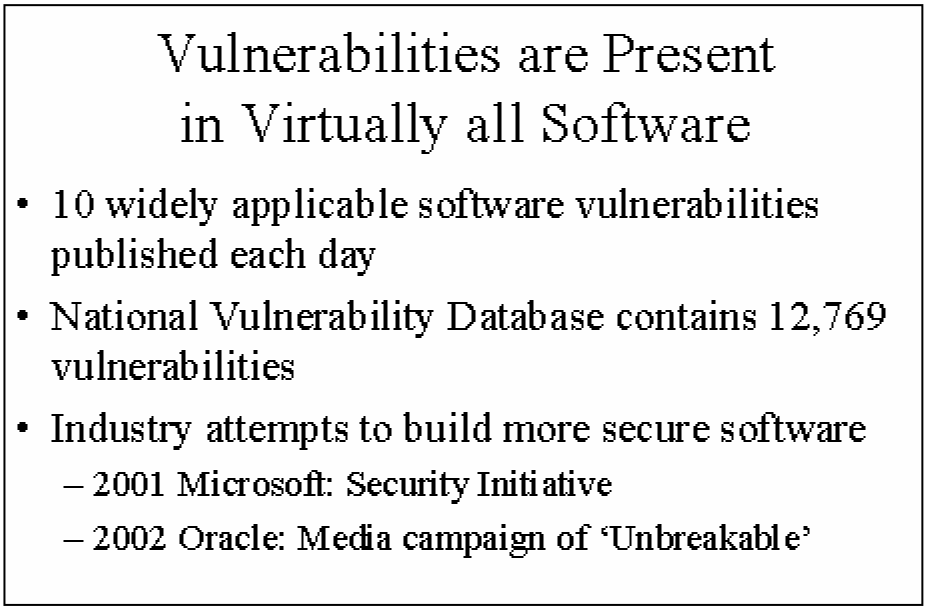
NIST
National Institute of Standards and Technology

Handling IT System Threat Information

National Vulnerability Database
NIST
10/5/2005

Peter Mell, Director, National Vulnerability Database, NIST Computer Security Division

Mr. Mell first described the National Vulnerability Database (NVD), which contains all known computer security vulnerabilities and is available at http://nvd.nist.gov.



Vulnerabilities are Present in Virtually all Software

- 10 widely applicable software vulnerabilities published each day
- National Vulnerability Database contains 12,769 vulnerabilities
- Industry attempts to build more secure software
  – 2001 Microsoft: Security Initiative
  – 2002 Oracle: Media campaign of 'Unbreakable'

Figure 6

Mr. Mell noted that if you use software, vulnerabilities are likely to exist in that particular computer program (Figure 6). Vulnerabilities also exist in open source software products.

Software companies including Microsoft and Oracle have been working to deal with this difficult software security vulnerability problem. For example, Oracle has reduced their software vulnerabilities to twenty per year, which is noteworthy for a major software company.

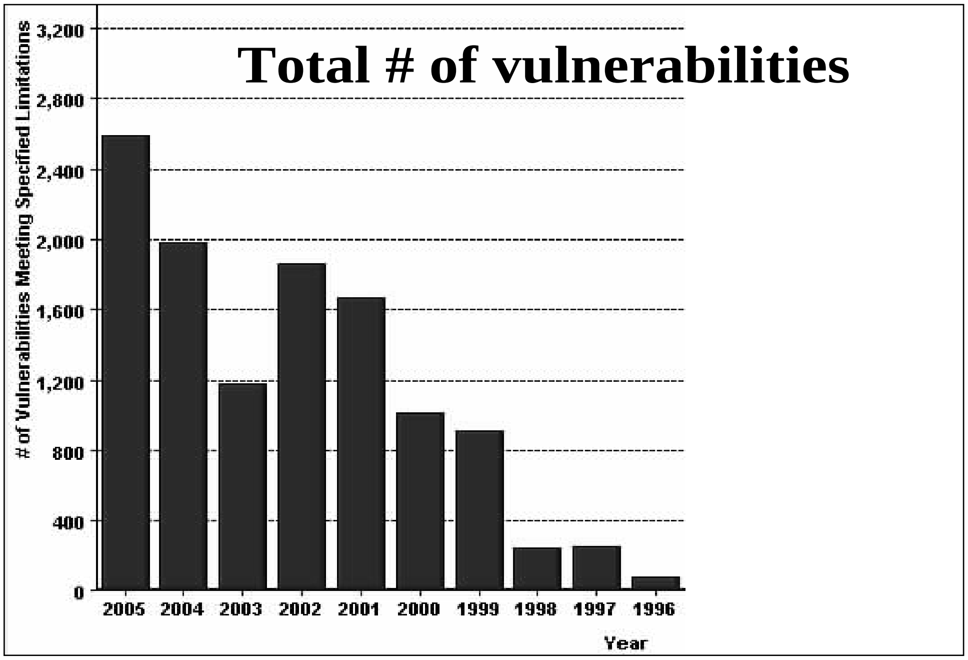


Figure 7

The number of vulnerabilities has grown nearly every year since 1996 (Figure 7). The reason for the anomaly in 2003 is not known at this time.

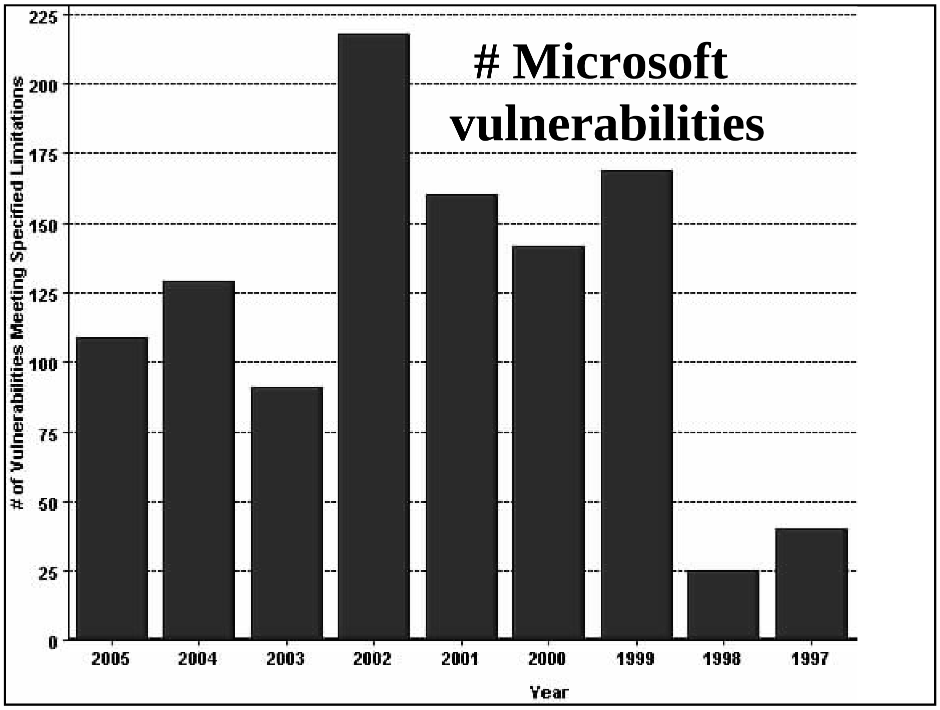
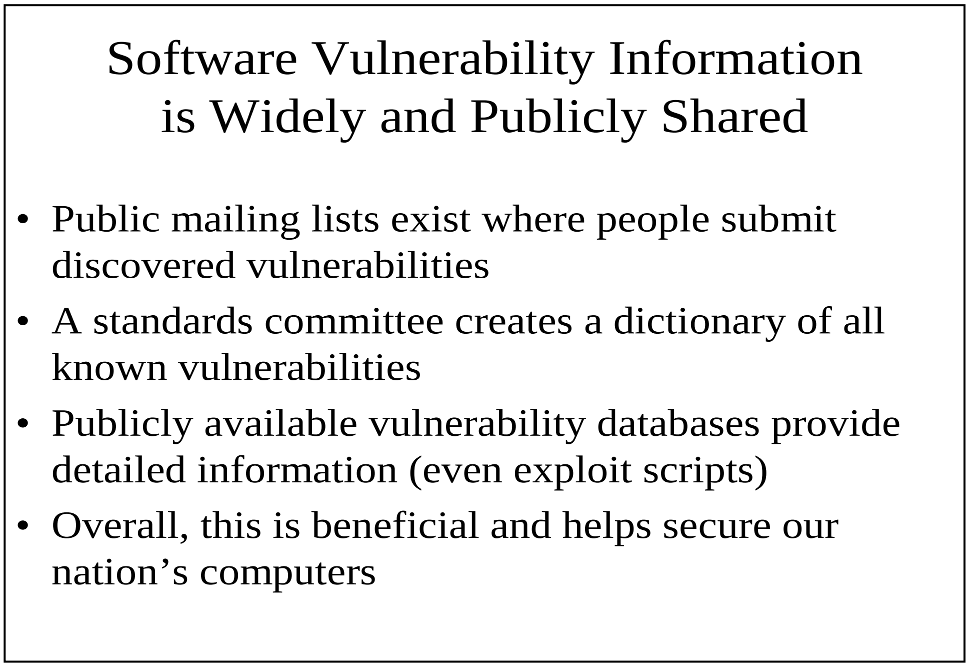


Figure 8

On the graph of vulnerabilities for Microsoft products, the last three years show the fruits of their security initiative to reduce vulnerabilities in their products (Figure 8). Still, the company's products are maintaining at around one hundred vulnerabilities per year.

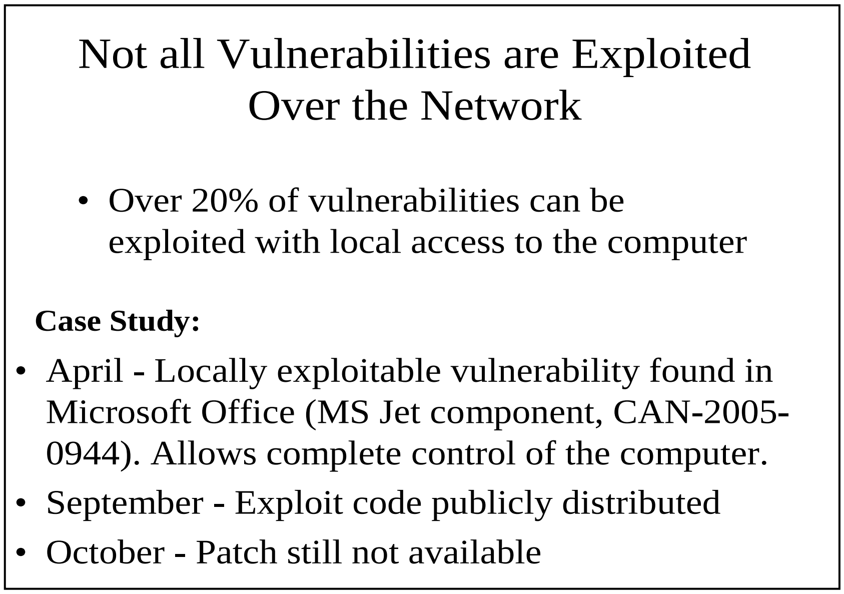


Software Vulnerability Information is Widely and Publicly Shared

- Public mailing lists exist where people submit discovered vulnerabilities
- A standards committee creates a dictionary of all known vulnerabilities
- Publicly available vulnerability databases provide detailed information (even exploit scripts)
- Overall, this is beneficial and helps secure our nation's computers

Figure 9

People in the computer industry debate when and what extent to disclose information on security vulnerabilities. However, the industry universally accepts the value of publicly disclosing the vulnerabilities as a way to improve the nation's computer security (Figure 9). Everyone benefits from public audits ensuring that vulnerability has been fixed.



Not all Vulnerabilities are Exploited Over the Network

- Over 20% of vulnerabilities can be exploited with local access to the computer

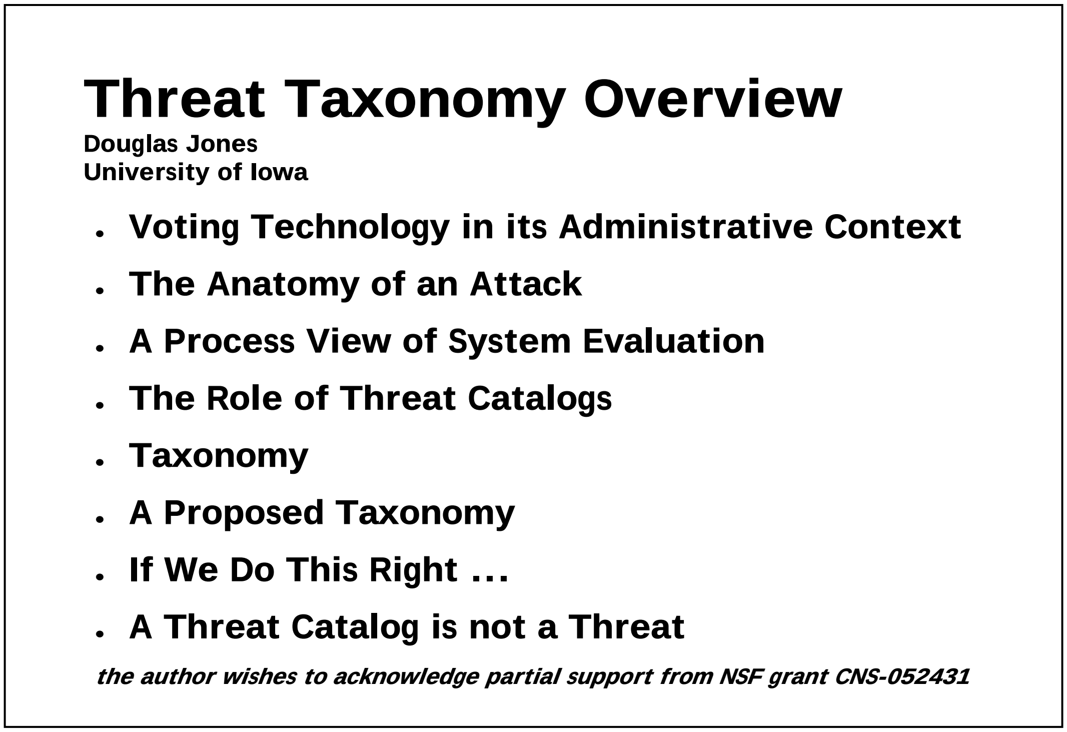
Case Study:

- April - Locally exploitable vulnerability found in Microsoft Office (MS Jet component, CAN-2005-0944). Allows complete control of the computer.
- September - Exploit code publicly distributed
- October - Patch still not available

Figure 10

Not all hacking is done over the Internet. Approximately 20% of security vulnerabilities occur locally on a computer (Figure 10). A case study in April 2005 revealed a local vulnerability that affected Microsoft Access and allowed complete control of the computer. As of October 2005, a patch for the vulnerability is still not available. (Vendors are not always able to release patches quickly.) Some voting systems utilize Microsoft Access.

During the question and answer session, Mr. Mell noted that categorization schemes for types of security vulnerabilities are not useful at this time. They tend to be either too large or not precise enough to input into a software scanner.



Threat Taxonomy Overview

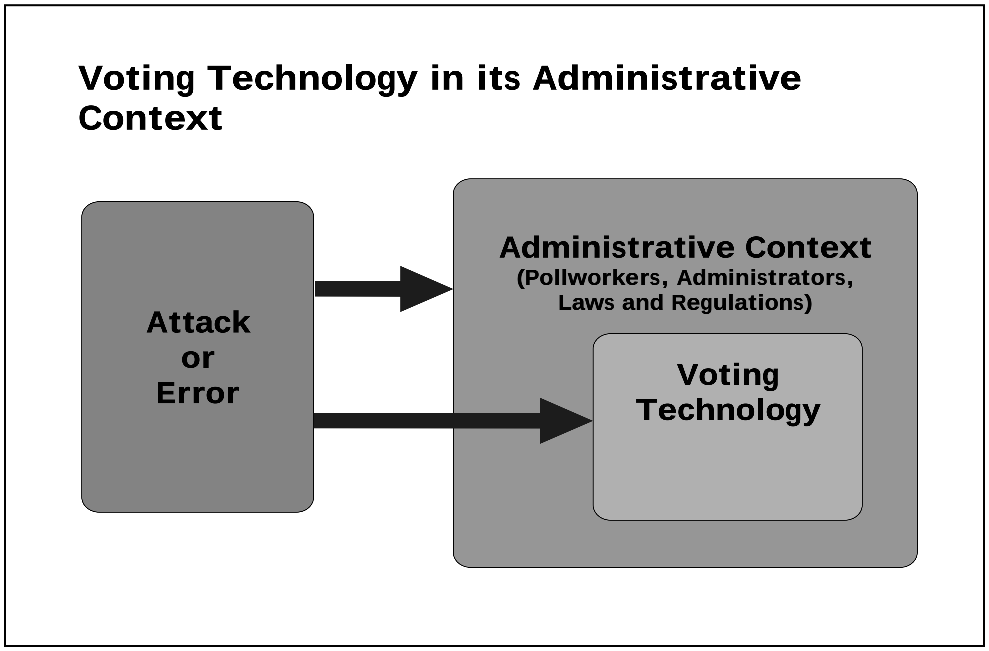
Douglas Jones
University of Iowa

- Voting Technology in its Administrative Context
- The Anatomy of an Attack
- A Process View of System Evaluation
- The Role of Threat Catalogs
- Taxonomy
- A Proposed Taxonomy
- If We Do This Right …
- A Threat Catalog is not a Threat

the author wishes to acknowledge partial support from NSF grant CNS-052431

Doug Jones, University of Iowa, Department of Computer Science

In his presentation, Dr. Jones discussed valid reasons for creating a threat taxonomy, why we need it, and how we would use it.



Voting Technology in its Administrative Context

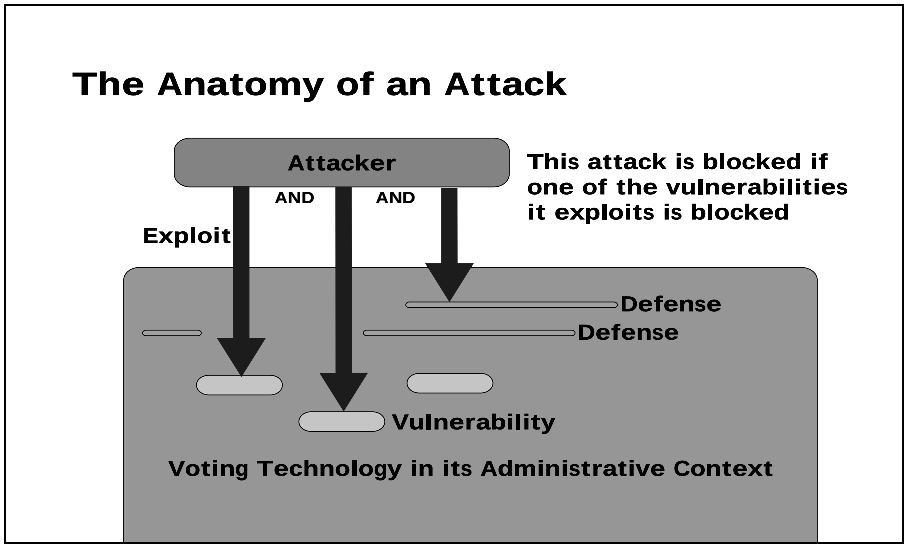
Attack or Error

Administrative Context (Pollworkers, Administrators, Laws and Regulations)

Voting Technology

Figure 11

Within the context of a threat analysis, election officials will point out that it is not just the technology, but also the administrative procedures that play an important role (Figure 11). The three “p’s” of elections are people, policies, and procedures. An understanding of the administrative context is crucial to analyzing threats of attacks. There is a tendency in the software world to look at the software components in isolation. Dr. Jones pointed out the need to enlarge the voting system perspective to look at the threats for attacks or errors within an administrative context. Attacks can occur inadvertently.

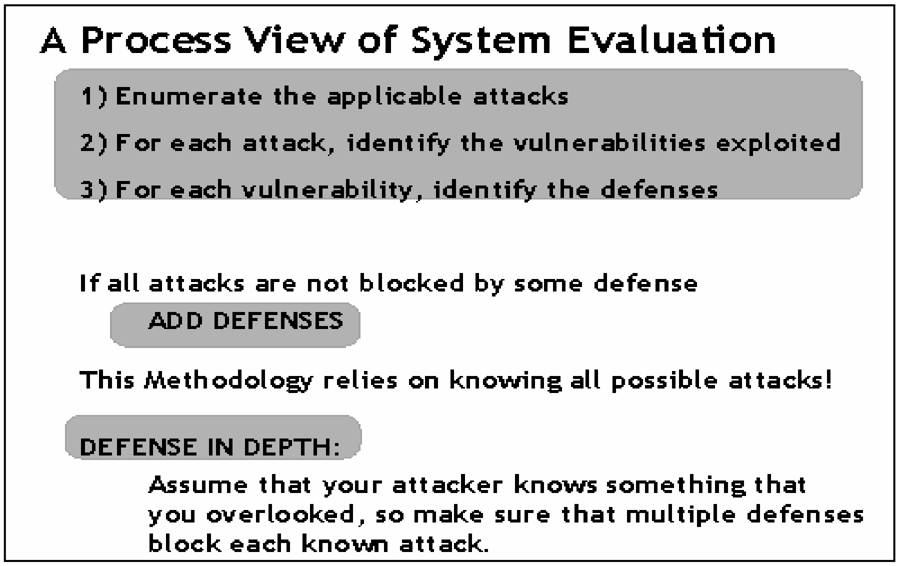


The Anatomy of an Attack

Attacker
AND
AND
Exploit
This attack is blocked if one of the vulnerabilities it exploits is blocked
Defense
Defense
Vulnerability
Voting Technology in its Administrative Context

Figure 12

Before discussing taxonomy, Dr. Jones discussed the need to agree on relevant terminology. Someone attacking a voting system needs to identify a set of vulnerabilities. An attacker usually exploits a subset of the existing vulnerabilities of a system (Figure 12). The vulnerabilities are often procedural and technological.



A Process View of System Evaluation

1) Enumerate the applicable attacks

2) For each attack, identify the vulnerabilities exploited

3) For each vulnerability, identify the defenses

If all attacks are not blocked by some defense

ADD DEFENSES

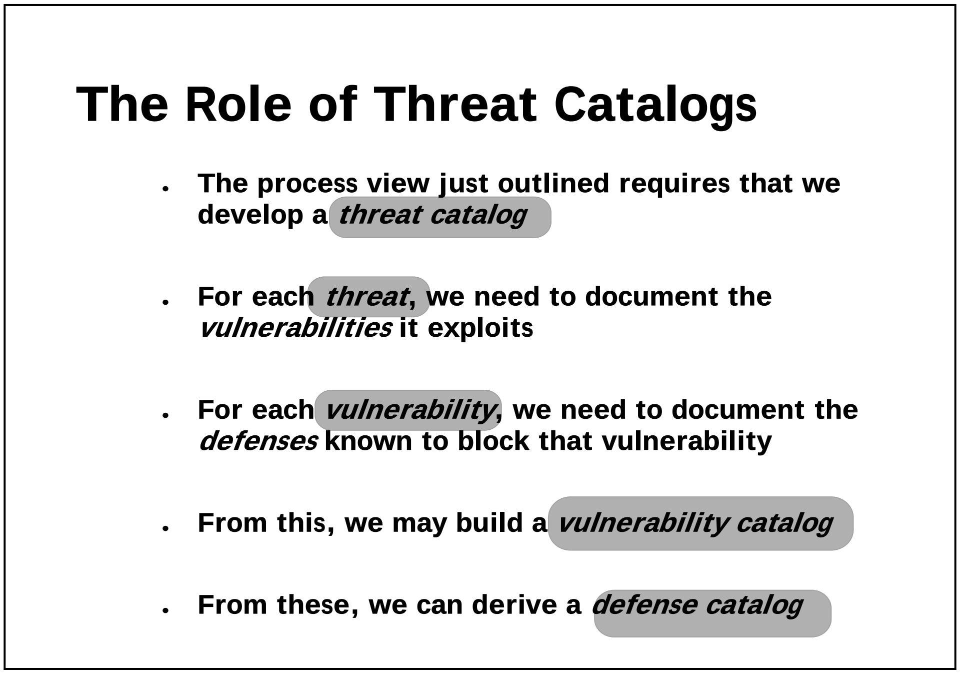
This Methodology relies on knowing all possible attacks!

DEFENSE IN DEPTH:

Assume that your attacker knows something that you overlooked, so make sure that multiple defenses block each known attack.

Figure 13

Designing defenses against particular hypothetical attacks does not always work. Frequently, the attacker will exploit a vulnerability against which the system has no protection. In the case of the anthrax attack against the U.S. Senate, the payroll office was protected because of defenses it had created to Y2K vulnerabilities- completely unrelated to the Anthrax attack. Figure 13 describes two methodologies: first, a standard planning process that assumes you know all attacks. A second methodology assumes you will overlook attack possibilities. So you create multiple defenses. This defense in depth methodology provides the system with the potential capability to defend against an unforeseen attack.

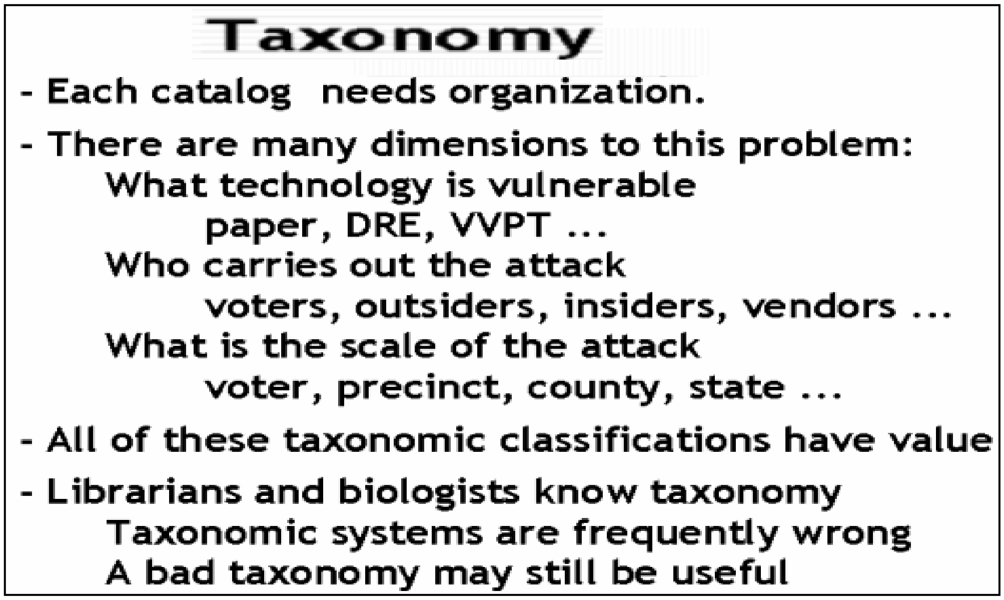


The Role of Threat Catalogs

- **The process view just outlined requires that we develop a *threat catalog***
- **For each *threat*, we need to document the *vulnerabilities* it exploits**
- **For each *vulnerability*, we need to document the *defenses* known to block that vulnerability**
- **From this, we may build a *vulnerability catalog***
- **From these, we can derive a *defense catalog***

Figure 14

A defense in depth process requires the creation of a threat catalog: a collection of all the attacks on a system thought possible. In addition, documentation of vulnerabilities and defenses for each threat allows a planner to build vulnerability and defense catalogs.

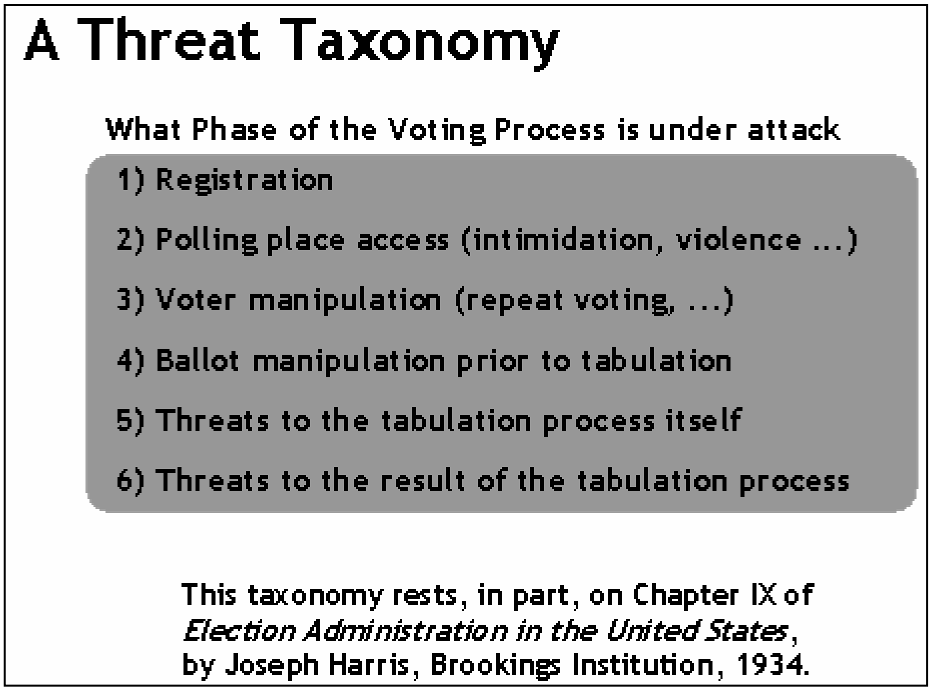


Taxonomy

- Each catalog needs organization.
- There are many dimensions to this problem:
  - What technology is vulnerable
    - paper, DRE, VVPT ...
  - Who carries out the attack
    - voters, outsiders, insiders, vendors ...
  - What is the scale of the attack
    - voter, precinct, county, state ...
- All of these taxonomic classifications have value
- Librarians and biologists know taxonomy
  - Taxonomic systems are frequently wrong
  - A bad taxonomy may still be useful

Figure 15

Threat catalogs need organizational structure to be useful. In areas such as biology and library science, taxonomic classifications work well as an organizational tool. Figure 15 offers three possible dimensions for threat catalog taxonomies. Each has potential value. Librarians and biologists are acutely aware that first drafts of taxonomies are often inaccurate. They can still offer useful guidance towards the development of an improved taxonomy.



A Threat Taxonomy

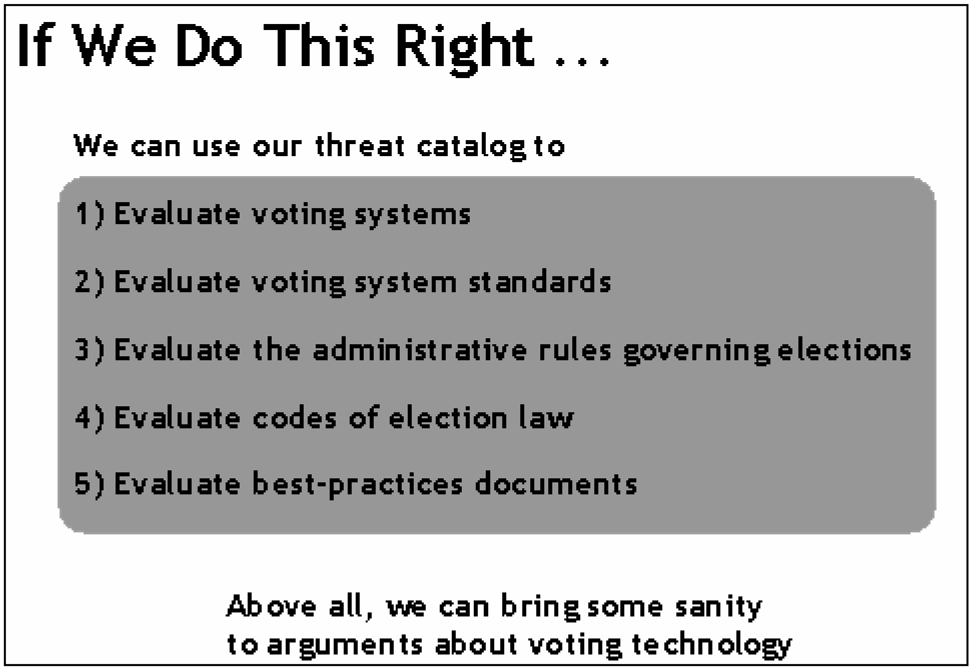
What Phase of the Voting Process is under attack

1) Registration
2) Polling place access (intimidation, violence ...)
3) Voter manipulation (repeat voting, ...)
4) Ballot manipulation prior to tabulation
5) Threats to the tabulation process itself
6) Threats to the result of the tabulation process

This taxonomy rests, in part, on Chapter IX of
Election Administration in the United States,
by Joseph Harris, Brookings Institution, 1934.

Figure 16

Based on the work of Joseph Harris in 1934, Dr. Jones offered a threat taxonomy classification scheme based on the phase of the voting process under attack in Figure 16. There are both procedural and technological attacks possible in all six phases.



If We Do This Right ...

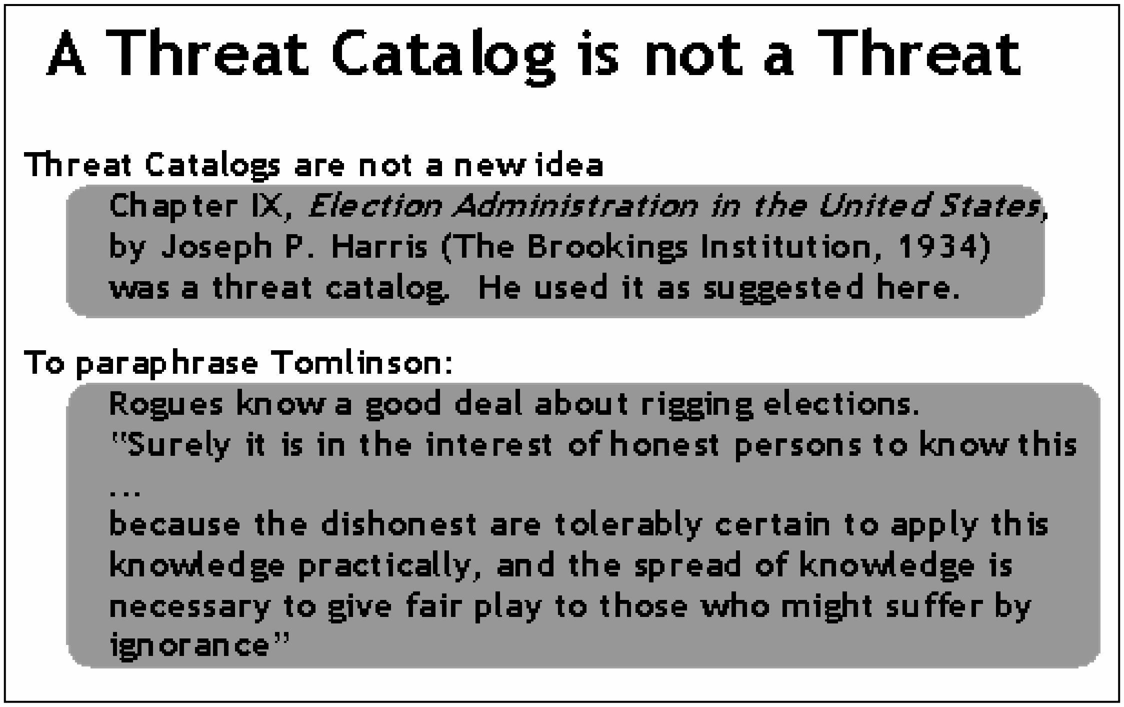
We can use our threat catalog to

1) Evaluate voting systems
2) Evaluate voting system standards
3) Evaluate the administrative rules governing elections
4) Evaluate codes of election law
5) Evaluate best-practices documents

Above all, we can bring some sanity
to arguments about voting technology

Figure 17

A threat classification based on a well-constructed taxonomy allows you to evaluate both voting system standards and best-practices documents. Figure 17 shows five areas where a taxonomy-based threat catalog will allow you to base evaluations on facts and not assumptions.



A Threat Catalog is not a Threat

Threat Catalogs are not a new idea

Chapter IX, *Election Administration in the United States*, by Joseph P. Harris (The Brookings Institution, 1934) was a threat catalog. He used it as suggested here.

To paraphrase Tomlinson:

Rogues know a good deal about rigging elections.
"Surely it is in the interest of honest persons to know this
...
because the dishonest are tolerably certain to apply this knowledge practically, and the spread of knowledge is necessary to give fair play to those who might suffer by ignorance"

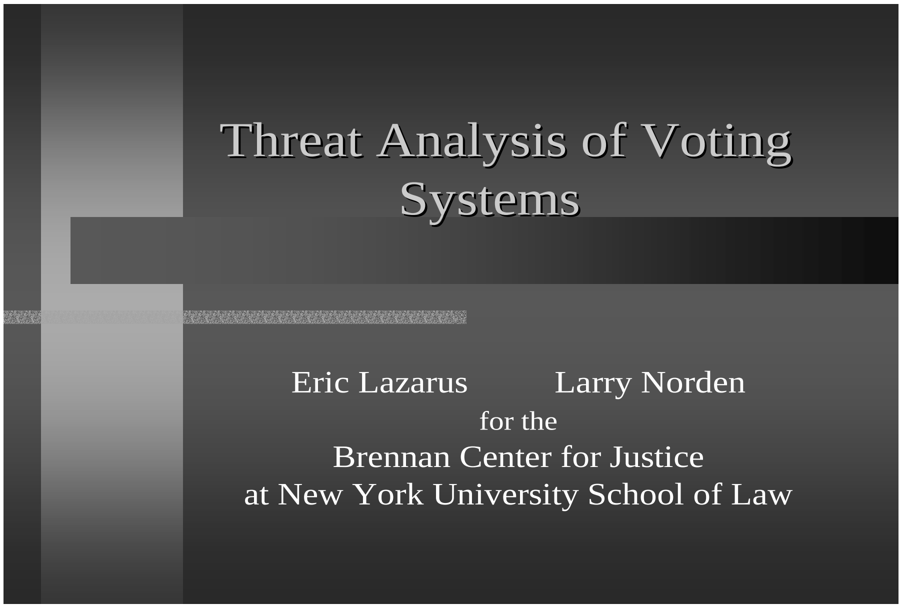
Figure 18

In 1934, Joseph Harris recognized the utility of a voting threat catalog. In the 1800s in a book on lock picking, Tomlinson discussed the value of spreading knowledge concerning election rigging mechanisms (Figure 18).

During the question and answer period, Dr. Jones reiterated the value of starting what may turn out to be a "bad taxonomy" simply to motivate the collection of threats. With the addition of examples over time, the taxonomy could be fixed incrementally by a representative small group of editors.

Referring to the lack of a voting threat analysis since the work of Joseph Harris in 1934, he noted that the catalog process needed to be institutionalized. In a contemporary catalog, we need to better understand the interaction of humans and voting technology. The election process is harder to manage than other computer security problems due to the human- technology interface.

Narrative threat descriptions will only be useful in a threat analysis after they are systematically classified into a formal taxonomy that forms the basis of a computer database.

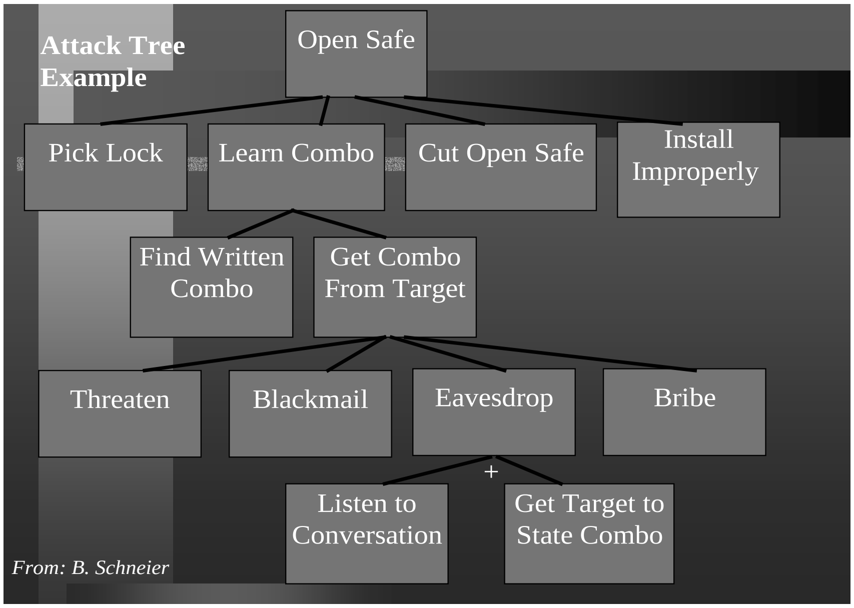


Threat Analysis of Voting Systems

Eric Lazarus Larry Norden
for the
Brennan Center for Justice
at New York University School of Law

Larry Norden indicated that the Brennan Center has spent the last several months cataloging close to one hundred potential attacks to voting systems. Making the threat catalog useful to the election community, especially decision and policy makers, has been challenging. It is useful to review the limitations of a threat catalog. Accuracy of voting systems, usability of voting systems and cost of voting systems are as important as security of a voting system. In their review, the Brennan Center did not examine technology-neutral threats to voting systems such as voter intimidation or voter roll manipulation.

The Brennan Center has concentrated on security threats to and counter-measures for voting systems themselves, especially vulnerabilities that will affect a large number of votes and thus the outcome of an election.



Attack Tree Example

Open Safe

Pick Lock

Learn Combo

Cut Open Safe

Install Improperly

Find Written Combo

Get Combo From Target

Threaten

Blackmail

Eavesdrop

Bribe

+

Listen to Conversation

Get Target to State Combo

From: B. Schneier

Figure 19

How seriously do you take each potential threat on a voting system? How do you balance priorities on both security threats and countermeasures? To examine these questions more fully, Eric Lazarus defined a threat tree approach not only as shorthand for organizing a collection of potential attacks but also as a repeatable and objective approach to analysis of threats to voting systems. Figures 19, 20, and 21 illustrate an example of an attack tree for opening a safe. On the first row, the tree structure defines the possible choices for opening the safe (the high-level goal). The example follows down methods to learn the combination, then methods to get the combination from the target, and finally the two required methods for eavesdropping to be successful.

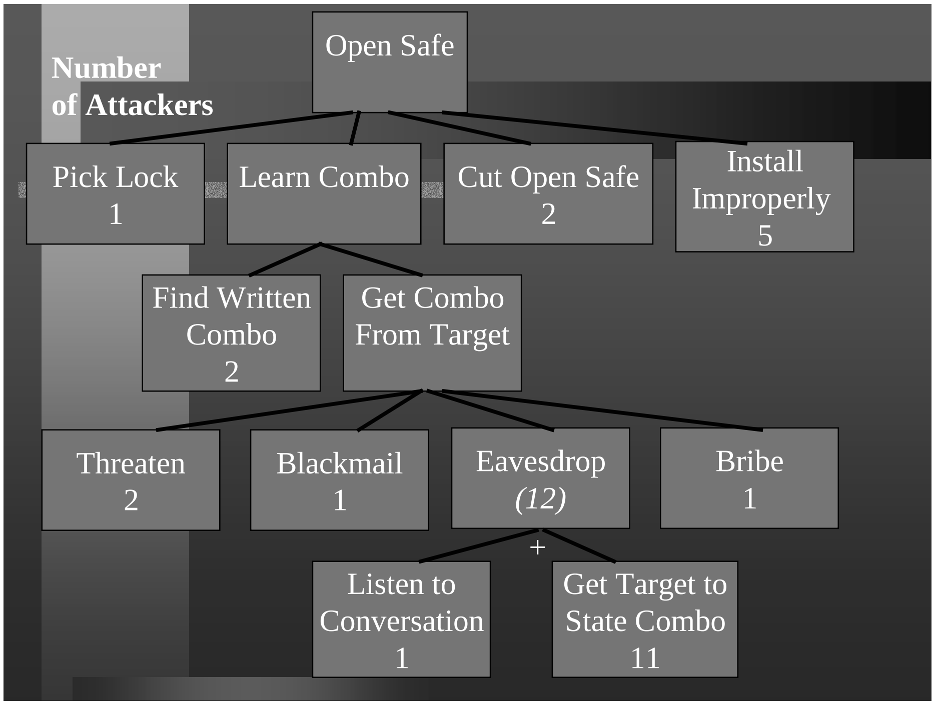
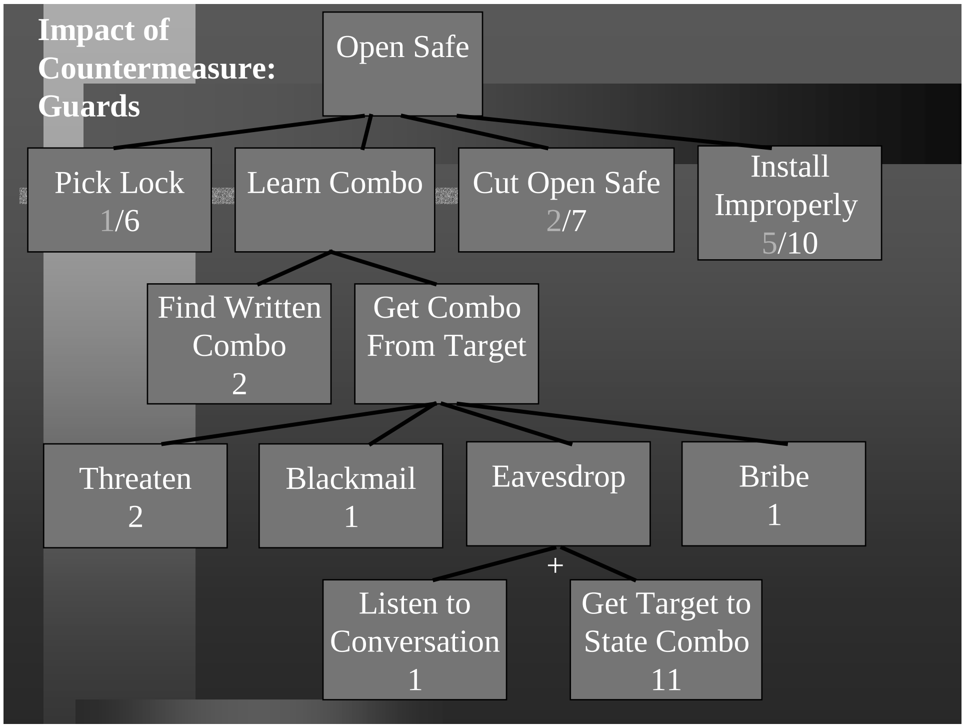


Figure 20

Figure 20 annotates attack plan steps in the tree with degree of difficulty information. In the illustrative example, the boxes contain the number of attackers hypothetically required to accomplish each task. The fewer the number of attackers required, the less the degree of difficulty. In the case of voting system attacks, we should be concerned about high-impact attacks: ones that can steal sufficient numbers of votes to overturn the outcome of a close election without being too difficult (i.e., too detectable for the attacker).



Impact of Countermeasure: Guards

Open Safe

Pick Lock
1/6

Learn Combo

Cut Open Safe
2/7

Install Improperly
5/10

Find Written Combo
2

Get Combo From Target

Threaten
2

Blackmail
1

Eavesdrop

Bribe
1

+

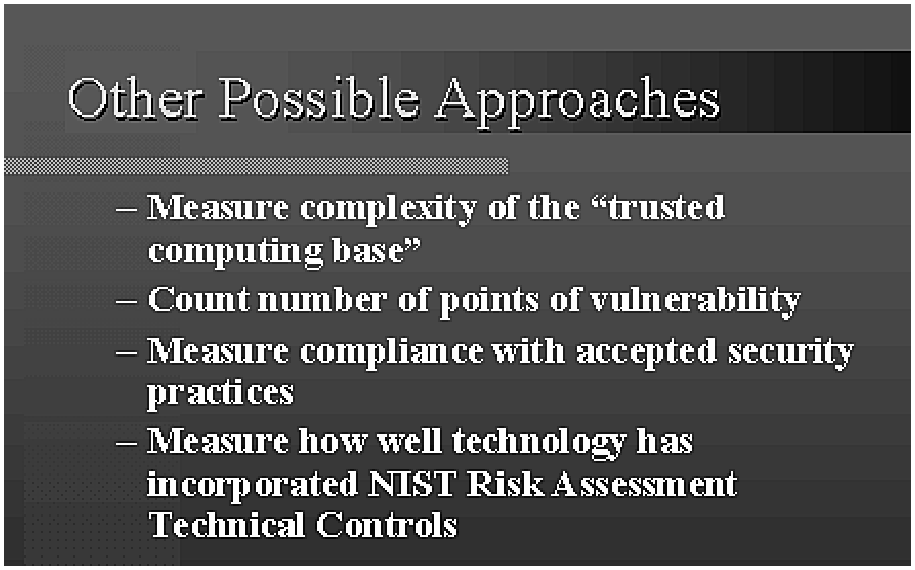
Listen to Conversation
1

Get Target to State Combo
11

Figure 21

Figure 21 illustrates the potential value of countermeasures on safe attacks. Assume that this business puts guards out front of the building housing the safe. For the plans that require breaking in at night (assume that those are: Pick Lock, Cut Open Safe, or Install Improperly) the teams require five people with guns so each of these plans now go up in difficulty, as measured by likely team size. The value of a countermeasure is quantified in terms of the difficulty of the plan for the attackers. The key is to determine which countermeasures will make the easiest attacks on a particular technology the most difficult for attackers.

Summarizing how to use the threat tree model with a voting system, Mr. Lazarus pointed out that you first need to determine a model jurisdiction for potential attack. This determination provides data such as number of poll workers and voters for building the threat tree. The modelers then need to agree on a level of attack difficulty. This data is annotated on the attack tree steps along with the impacts of countermeasure data. The modelers can then examine the effects of adding or removing specific countermeasures that increase the difficulty of a high-value attack.

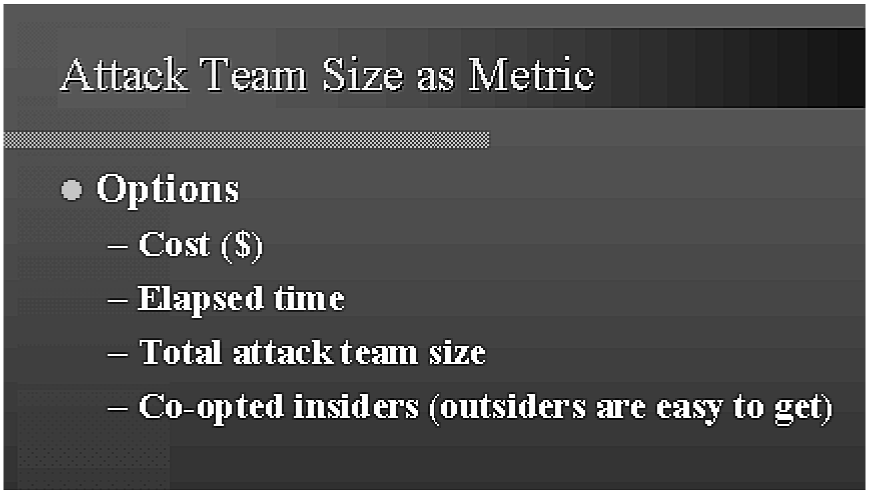


Other Possible Approaches

- – Measure complexity of the "trusted computing base"
- – Count number of points of vulnerability
- – Measure compliance with accepted security practices
- – Measure how well technology has incorporated NIST Risk Assessment Technical Controls

Figure 22

The Brennan Center looked at other approaches to threat classification (Figure 22). Measuring complexity of computer programs (lines of code) can lead to the erroneous conclusion that voting systems with less technology are more invulnerable to attacks. Counting the points of vulnerability in a system can also lead to a similar erroneous conclusion regarding security. Voting has unique security issues. Thus measuring a voting system's compliance with accepted security practices in other venues does not address many of the vulnerabilities that are unique to voting systems.



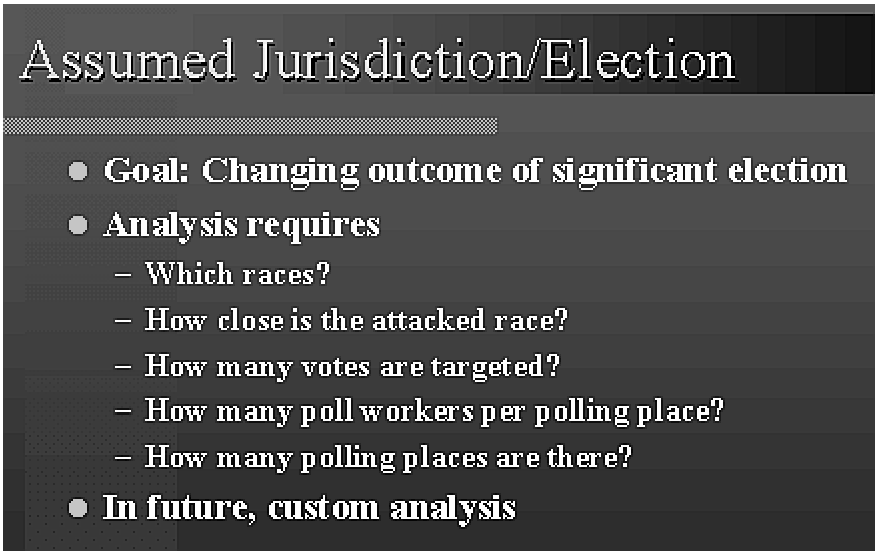
Attack Team Size as Metric

- Options
  - – Cost ($)
  - – Elapsed time
  - – Total attack team size
  - – Co-opted insiders (outsiders are easy to get)

Figure 23

Applying the attack tree to the voting security problem also requires that you initially examine costs involved for the attackers, attack team size (difficulty), and elapsed time necessary for an attack to take place (Figure 23). Cost is a relative measure and is not useful as a way of distinguishing the attacks from one another.

It may be rational to use time to measure strength of a safe. For the strength of election systems, it is not so relevant. When would you even start the clock? While it is clear why the attacker must act swiftly in a bank robbery; it is not clear why speed is important to the attacker in the case of election fraud. It is plausible to examine whether an insider can attack from within an election office (co-opted insiders).

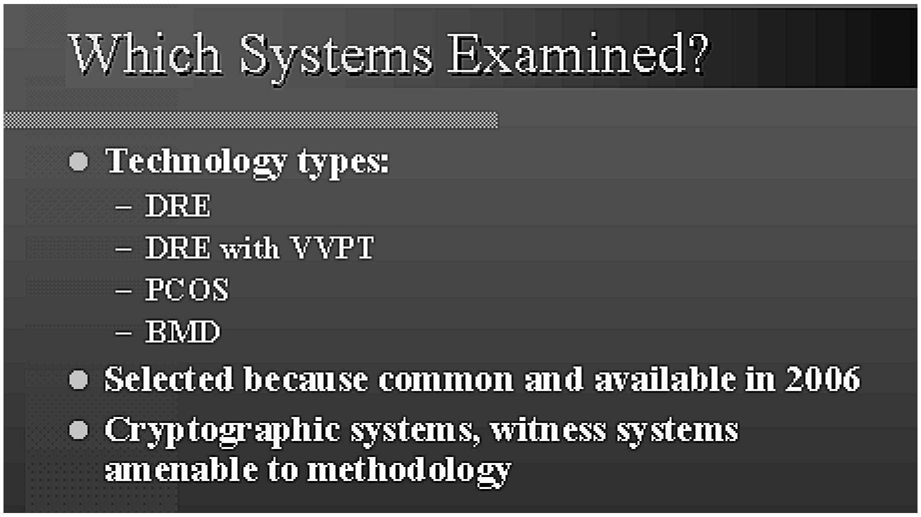


Assumed Jurisdiction/Election

- **Goal: Changing outcome of significant election**
- **Analysis requires**
  - Which races?
  - How close is the attacked race?
  - How many votes are targeted?
  - How many poll workers per polling place?
  - How many polling places are there?
- **In future, custom analysis**

Figure 24

Mr. Lazarus noted that the Brennan Center wanted to find jurisdictions that are typical of the elections we most need to protect (Figure 24). Having secure election systems that give us confidence even when the election is “freakishly” close may not be practical yet. On the other hand, it does not make sense to focus too much attention on attempted fraud against elections in which the outcome is a foregone conclusion. In the end, we may want to perform an analysis of attack difficulty based on the most plausible assumptions. For example, picking a highly populated county for analysis would make sense since an attacker could steal enough votes in that single location to influence the outcome.



Which Systems Examined?

- **Technology types:**
  - DRE
  - DRE with VVPT
  - PCOS
  - BMD
- **Selected because common and available in 2006**
- **Cryptographic systems, witness systems amenable to methodology**

Figure 25

The Brennan Center's attack tree analysis will focus on four types of voting systems (Figure 25). Those systems are Direct Recording Electronic (touch screen) Systems with and without paper printers; precinct count optical scan systems, and ballot marking devices. This same sort of analysis could be applied to cryptographic and witness voting systems.

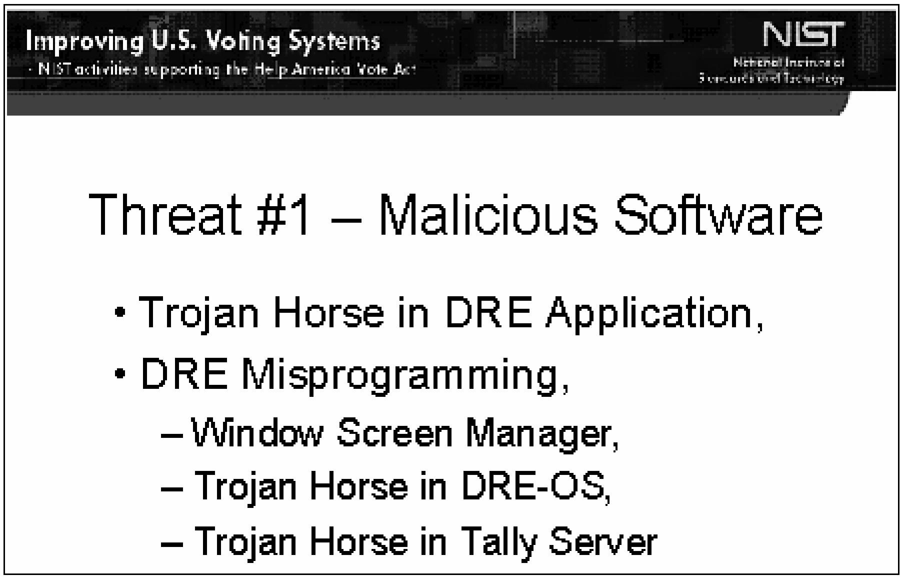
<u>**Panel 1**</u>**- *Threat Discussion on Trojan Horses, Backdoors, and Other Voting System Software-Related Problems***
Paul Craft, Douglas Jones, John Kelsey, Ronald Rivest, Michael Shamos, Dan Tokaji, Dan Wallach

Moderator: Barbara Guttman, NIST Information Technology Laboratory

The panelists introduced themselves.

John Kelsey, NIST Information Technology Laboratory, Computer Security Division
Michael Shamos, Carnegie Mellon University
Dan Wallach, Rice University
Dan Tokaji, Ohio State University, Moritz College of Law
Ron Rivest, Massachusetts Institute of Technology
Doug Jones, University of Iowa
Paul Craft, Voting Systems Certification, State of Florida

Guttman introduced the threats to be discussed by the panel (Figures 26, 27) and the questions to be answered (Figure 28).



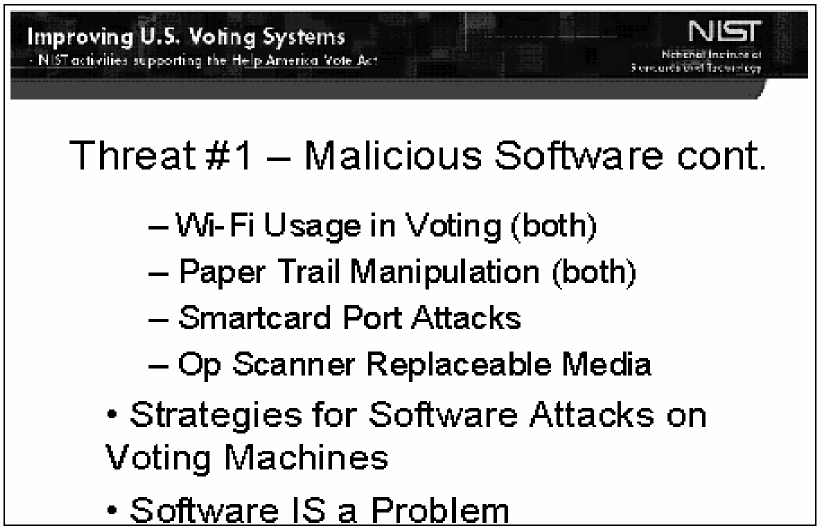
Improving U.S. Voting Systems
- NIST activities supporting the Help America Vote Act

NIST
National Institute of Standards and Technology

Threat #1 – Malicious Software

- Trojan Horse in DRE Application,
- DRE Misprogramming,
  - Window Screen Manager,
  - Trojan Horse in DRE-OS,
  - Trojan Horse in Tally Server

Figure 26



Improving U.S. Voting Systems
- NIST activities supporting the Help America Vote Act

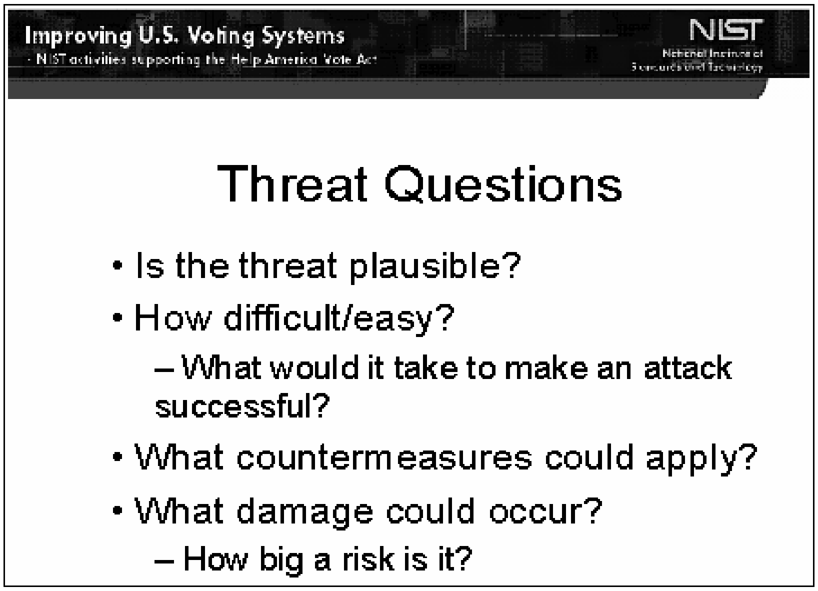
NIST
National Institute of Standards and Technology

Threat #1 – Malicious Software cont.

- – Wi-Fi Usage in Voting (both)
- – Paper Trail Manipulation (both)
- – Smartcard Port Attacks
- – Op Scanner Replaceable Media

- • Strategies for Software Attacks on Voting Machines
- • Software IS a Problem

Figure 27



Improving U.S. Voting Systems
- NIST activities supporting the Help America Vote Act

NIST
National Institute of Standards and Technology

Threat Questions

- • Is the threat plausible?
- • How difficult/easy?
  - – What would it take to make an attack successful?
- • What countermeasures could apply?
- • What damage could occur?
  - – How big a risk is it?

Figure 28

Discussion of malicious software threats:

Craft indicated he thought the threat was plausible, especially with software where the point of origin cannot be determined. There are probably jurisdictions in the United States where you cannot account for the origin and whether the installed version is the certified software.

Jones noted that in Iowa, California, and other states, uncertified software has been discovered by officials. The problem goes a level deeper. There is no way to determine whether a closed voting system is running the version of the software that it displays on self-check.
Jones described a non-malicious attack in Iowa that resulted in the introduction of an unintended “Trojan horse” bug by installing a Microsoft Windows 95 operating system maintenance software upgrade that was not certified.
Craft noted that, in one instance, the threat can be alleviated by validating the firmware for DRE equipment before installing it on the machine. It requires that the election administrator maintain

strict custody of the DREs after loading the validated software. Still, we need better ways to validate firmware after installation. We also need to develop firmware/software system validation into a simple process for all election officials.

Rivest agreed that malicious software is a real threat. Certification of software is indeed a sanity check and provides a level of assurance. But the process by itself will not find all of the bugs and malicious Trojans inside software. Probably most of the bugs inside complex voting software are non-malicious. The software code development process offers another approach to increase quality assurance. A third controversial approach is the use of open source code. Set-up validation is critical to the process in the ways mentioned by Craft and Jones previously.

Tokaji highlighted pre-election, election, and post-election countermeasures as safeguards including certification and parallel testing.

Craft noted that instances of “prior art” countermeasures often are overlooked. State and federal election codes evolved as mitigations to threats that occurred in previous elections.

Wallace addressed his initial remarks to the size of the trusted computing base: the things that have to work to make sure the system is secure. You need to minimize the places where an attacker can attack the system. Smart cards are a potential entry point for an attacker. Substituted malicious cards have the computing capability to reprogram a voting system. You can mitigate the threat through strict procedures or simpler designs of voting systems.

Shamos remarked that it will be important to prioritize the threats by levels of risks and potential gains from addressing them. He also brought up the features of voting software that the voting system vendor discloses to the customer but not to the examiner. An example is a feature that allows election officials to change the election total after the election “if needed.” Examiners find software bugs of which the vendor was aware, and examiners also find bugs unknown to the vendor. We need to be concerned about software as distributed separate from malicious attack.

There are no mechanisms for source code control or object code distribution effectively in place anywhere. Georgia has the best mechanism where the vendor sends the software to Kennesaw State University where it is vetted before it is sent to the jurisdictions. This moves the locus of trust from the vendor to Kennesaw State. A single locus of trust can still be an issue.

Shamos indicated that his key software tampering issue is whether an election can be conducted and an intrusion not detected. From his viewpoint, an unrealistic scenario is one that assumes a hacker can change an election outcome in a way that no manner of pre-election, election, or post-election testing or code reading can reveal the intrusion. There are numerous realistic intrusion scenarios, and an outcome for the workshop could be the enumeration of effective countermeasures.

Kelsey pointed out that examiners will not catch all the bugs in a program even in a thorough review.

Craft agreed and noted that one policy in place is to review software after each election to find new bugs. Software testing is in fact sampling methodology and will never be perfect.

Kelsey commented on Shamos' unrealistic attack scenario description. The point here for Kelsey is whether the attack would be caught with the procedures currently in place. While it is likely that software attacks can be successful, it is also likely that the attack can be detected.

Craft noted the need for more research into the plausibility of the attack threats. Individuals theorizing some of the threats are not aware of the scale of effort required to conduct the intrusion. To intentionally change firmware requires numerous individuals and levels of effort that are beyond the capability of a single clandestine hacker.

Wallach noted that it is much easier to attack a latent flaw in the software than to craft a malicious variant of the software. He challenged Shamos' scenario in that it does not address the complexity of the problem. He posited that a sufficiently crafty adversary could hack into voting software and go unnoticed, as did Ken Thompson's hack of C code (see http://www.acm.org/classics/sep95/).

Shamos noted that proving or disproving the existence of the "omniscient hacker" is impossible. He then initiated a discussion of parallel testing. His recommendation to the Secretary of State of California involved empowering a team of people who could walk into any precinct on Election Day and pick any DRE which is then cordoned off from the other machines. Throughout the day, a stream of people operates the machines as if they were voters. However, in advance, the team's examiners know the outcome of the votes for that machine. The voting is videotaped to capture voter errors. At the end of the day when the polls are closed, the parallel testing machine is also closed, and the vote total is compared with the expected total. Similar parallel testing teams operate throughout the state. In theory, any organized attempt to influence the election would be captured if enough random precincts are targeted. The question for the panel is whether the testing effort is worth the cost as a countermeasure.

Craft was not sure that parallel testing was an effective countermeasure. However, it mitigates many of the conspiracy theories. The best defense against bad software code is controlling your system and managing procedures. Parallel testing provides an understandable proof and level of assurance that correct procedures have been implemented.

Shamos and Craft agreed that parallel testing was an effective attack detection measure.

Rivest indicated that parallel testing put up a steep fence for an adversary to scale. However, he raised the possibility of an adversary determining in advance which machine was to be used in parallel testing through a signal by a voter to the DRE. Also, while parallel testing adds value, it also adds expense.

Shamos indicated that you would need a fairly large conspiracy to carry out the signaling exploit for every voting machine. Local elections are most vulnerable to the signaling type of attack, especially in elections where every ballot is different in every precinct. Countermeasures need to be explored here.

Rivest raised a concern with wireless technology as an attack method to signal to multiple voting machines all at once. Shamos agreed and stated that wireless technology and voting do not mix.

Kelsey and Jones began a discussion of state recount laws and their applicability to unexplained and unexpected (surprise) election results. Tokaji recommended a review of state election recount laws available in his paper, *The Paperless Chase: Electronic Voting and Democratic Values*, September 2004 (see http://www.dos.state.pa.us/election_reform/lib/election_reform/Paperless_Chase.pdf).

Jones indicated his concern with an accepted definition of firmware as precedent by the Independent Testing Authorities (ITAs) for voting systems. The ITA-accepted definition of firmware is software that runs on the voting machine in the precinct. So software resident on a PCMCIA card was defined as firmware.

Panel 1 Audience Participation:

Question/Comment: Tencati addressed a question to Rivest and Kelsey: With the common criteria, digital signature capability, and FIPS 140 standards, are not some of these malicious threats mitigated? Rivest noted the issue of complexity of software and the voting process. He also noted the need for the proper use of cryptography and key management in the development of voting systems. Kelsey noted that the FIPS 140 standard does not address the insertion of malicious code by the vendor or a COTS software programmer.

Question/Comment: Saltman raised for discussion the reduction of software size so that it is manageable to test. He noted that what is essential is that the system software is correct. The number of bugs that can be found is inversely proportional to the size of the software program. The issue is the correctness of software that could eliminate the possibility of malicious software. Software with millions of lines of code is not required to run individual DREs. Single-function, process control software would seem more appropriate here. Large COTS software operating programs often cannot be tested for bugs.

Rivest agreed with Saltman's premise. He offered one possible solution that divides the voting process into two parts: composing the vote and (security critical) casting of the vote. The user interface in the vote composing section requires the advanced software code. The casting of the vote would be done at a separate secure station with a manageable software program. (See Cal. Tech-MIT Voting Project Report, http://www.vote.caltech.edu/reports/2001report).

Craft noted that simple, concise, and well-formed code is desirable. The voting process with HAVA has become more complex. Today's voting system has to talk in a variety of languages with variable audio and visual features. Provisional voting and early voting along with complex graphics also compound voting system requirements. The biggest problem with system security and software integrity results from changes in user demands over the last five years. Jones referenced the avionics industry as a model, where spending on software testing and certification is ten times the amount spent on software development. In voting systems, relatively small amounts of money are spent on testing versus the amount spent on software development. In the future, what the commercial voting industry needs is small, easily reusable COTS software modules developed to high standards. However, designing unique software for voting systems is financially burdensome. Shamos noted that an electronic election in India was successful for 360 million voters. The voting machines were hardwired and the election itself simple. Ballots are too complicated in the United States to use the Indian system.

Question/Comment: Hall asked the panel to address disclosing source code as well as the commercial pressures on voting system vendors with respect to trade secrets. Shamos noted that there is a difference between open source and disclosed source code. Disclosed source code is critically important. The public needs to be able to verify the integrity of software for themselves. Shamos stated that commercial trade secrets and voting software are inconsistent with one another in this instance, due to the high impact of secure voting on the democratic process. He indicated that there is no competition solely in voting system software. Wallach noted that publicly available voting system software would result in the development of more secure software in the long run.

Question/Comment: Klein raised the issue of differentiation between attacks and equipment malfunctions. The current voting system reliability standard for mean time between failures allows an Election Day failure rate of 10 percent (163 hours MTBF). Some failures are due to unacceptable electrostatic discharge rates. Antistatic procedures are currently insufficient. It is difficult to separate reliability from individual attack threats. Kelsey noted that masking your attack as an error is plausible. However, Shamos proposed that most instances of system unreliability are honestly that. The current MTBF is unacceptable, and it should be upwards of 1000 hours. Craft noted that most voting machines in production attain a high MTBF rate. However, we need to look at ongoing quality assurance issues in future voting system standards. Klein raised the point that in Maryland, there was data to indicate that voting systems did not meet the 163-hour standard. Jones raised the issue of complex policies and procedures for election workers. Failure to plug in voting machines resulted in a “low battery” failure rate of 10 percent in one instance in Florida. The failure was a procedural failure.

Question/Comment: Freeman noted that the 163-hour MTBF was set on central counting systems twenty years ago. The model back then represented five years of use. The use of voting equipment has increased exponentially since then. Looking at threat analyses, closed DRE systems do not allow for an external check for integrity at the point of execution. You need to consider less restrictive countermeasures as a trade-off so that you can perform adequate safeguards against an attack approach. In addition, validating software can result in discovery of unrelated files and programs outside of the context of the voting software.

Question/Comment: Weatherbee noted the use of the common criteria by the defense community to solve the problem of certifying software code. He asked the panel to comment on the possibility of requiring voting system vendors to meet a protection profile for voting machines that could be developed through peer review by the technical community. He also asked the panel to comment on the certification process required for gambling slot machines in Nevada. Shamos agreed that the technology exists to create highly trusted and secure computer systems. However, the funding available to the defense and the gambling industry to create these secure systems far exceeds the funding available to the election community and the voting systems industry. He noted that, at this time, not enough concern for increased security of voting systems has been voiced by the public to elected officials. A heightened awareness could eventually provide the funds to increase security requirements. Jones commented favorably on Nevada’s certification of gambling machines as a model to emulate for future voting systems. The system for verifying that the software versions are correct requires additional hardware on each slot machine. The module that does the version control is produced and owned by the state.

Panel 2- ***Threat Discussion on Voting System Configuration Issues and Problems***
Jeremy Creelan, Dana DeBeauvoir, Douglas Jones, Avi Rubin, Ronald Rivest, Ted Selker, Michael Shamos

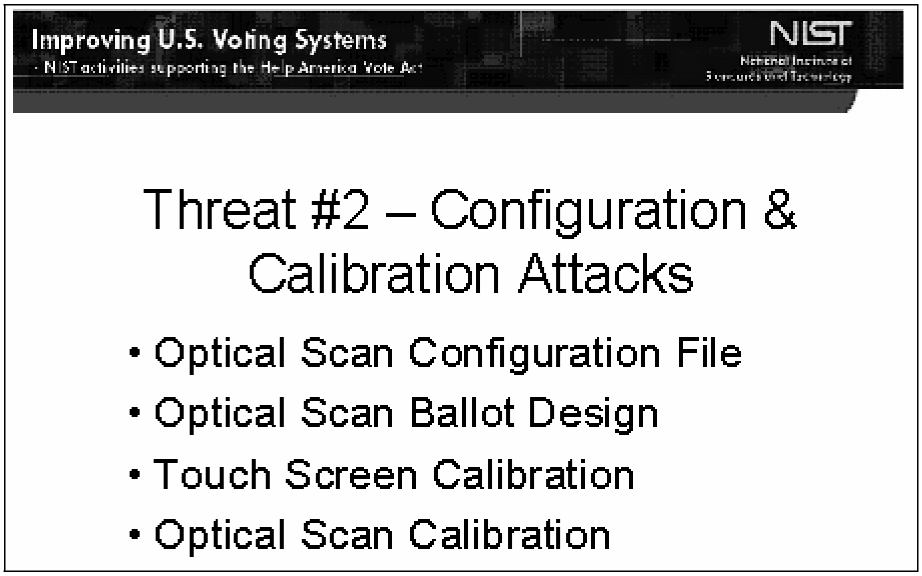
Moderator: Barbara Guttman, NIST Information Technology Laboratory

The panelists introduced themselves.

Michael Shamos, Carnegie Mellon University
Ron Rivest, Massachusetts Institute of Technology
Doug Jones, University of Iowa
Jeremy Creelan, NYU School of Law
Avi Rubin, Johns Hopkins University
Ted Selker, Massachusetts Institute of Technology
Dana DeBeavoir, Travis County, Texas, Clerk

Rubin first briefly described the NSF-funded ACCURATE project for studying security issues related to electronic voting. The NSF is the principal source of funding for university research into computer security issues. The funding is for basic research, education, and outreach. The purpose of the ACCURATE project is to create a platform of technology which others can use to make future voting systems more secure, accessible, usable, reliable, auditable, and transparent.

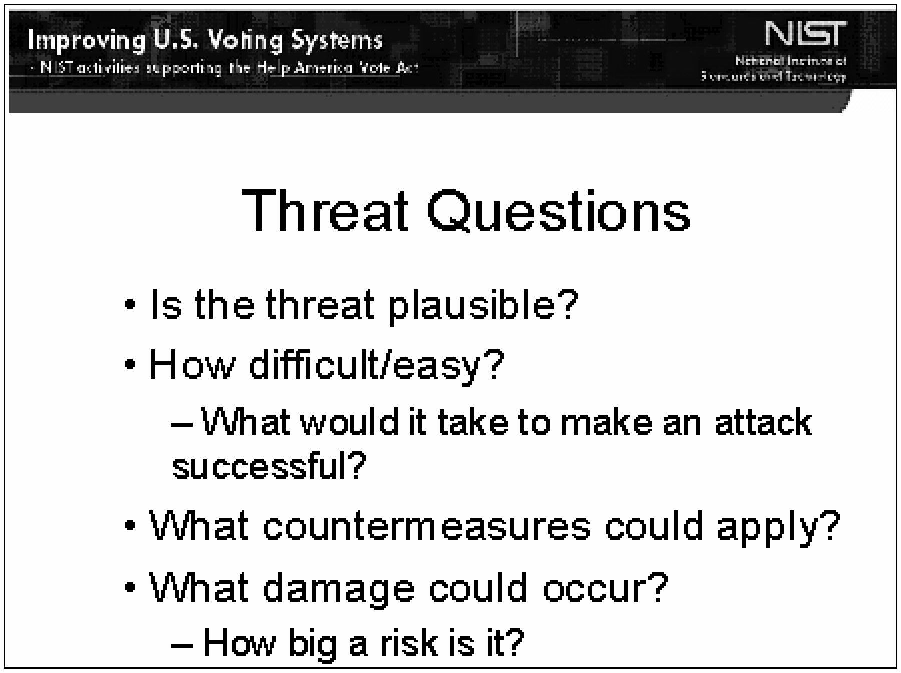
Guttman introduced the configuration and calibration attacks for the panel to discuss (Figure 29) and the questions to be addresses (Figure 30).


Improving U.S. Voting Systems
- NIST activities supporting the Help America Vote Act
NIST
National Institute of Standards and Technology

Threat #2 – Configuration & Calibration Attacks

- Optical Scan Configuration File
- Optical Scan Ballot Design
- Touch Screen Calibration
- Optical Scan Calibration

Figure 29



Improving U.S. Voting Systems

- NIST activities supporting the Help America Vote Act

NIST

National Institute of Standards and Technology

Threat Questions

- Is the threat plausible?
- How difficult/easy?
  - What would it take to make an attack successful?
- What countermeasures could apply?
- What damage could occur?
  - How big a risk is it?

Figure 30

Discussion of configuration and calibration threats to voting systems submitted to the workshop in advance:

Jones noted that the optical scan configuration file attack and the optical scan ballot file attack are two sides of the same coin. In op scan voting systems, there is no direct linkage between the candidate and the counter that is used to count that candidate’s votes. The ballot scanner only knows that there was a mark in a certain column and row. Inside the voting system is a configuration file that relates a position on the marked ballot and relates it to the candidate. The crucial configuration files have two sides to them: one configures the voting machine to count votes for candidates and the other is the file of information that goes to the printer to have the candidate’s names printed in the correct location. These represent two distinct attacks: one attack against the information going to the printer and the other, an attack on the file that goes to the tabulating machine. In one way, the touch screen calibration problem has similarities to the op scan problem. The touch screen device on a DRE is not the display screen but rather a thin transparent device that sits on top of the display screen. There is no direct connection between the coordinates that are sensed and the coordinates of a particular object on a display screen. Calibration is required. The mapping between the two represents an attack vulnerability. In the past, routine mistakes also have been made in the printing of ballots and in ballot configuration file generation.

Jones pointed out that optical scan calibration is a different issue. Here, it is a question of how dark a mark is required to be counted as a vote. If you have a ballot where calibrations can vary from precinct to precinct, you have the opportunity to make votes more likely to be correctly (or incorrectly) recorded in some locations than in others. The old ES&S central count scanners have separate photo sensor LED pairs looking at each column of the ballot. Those sensors are separately calibrated. Unless the sensors are calibrated correctly, the standard for what counts as a vote could differ by column. Counties do not always check this calibration. The 2002 VSS does not address the calibration issue because it is a human factors issue. There is a potential countermeasure here with new software from Hursti that would make tif file images of the ballots publicly available as an independent check on the counts.

Shamos noted that voters are intuitively aware of op scan voting technology from standardized testing. The ballot choices can be erased and re-marked, and the voter interface is user-friendly. The voter believes that the machine will view the vote as they recorded and viewed it. However, the op scan recognition technology is relatively unsophisticated and marks are not always picked up as the voter intended. States have different regulations defining the acceptable mark that constitutes a vote on an op scan ballot. In Hawaii, if any portion of the mark covers any portion of the oval, the mark counts as a vote. However, any mark outside the oval does not count as a vote. So circling the oval will not count as a vote. Shamos then addressed the calibration issue as it applies to the printer attack on optical scan voting. There are varying levels of friction in the rubber rollers that pick up the paper ballot in precinct op scan machines. Large black rectangles on the side of the ballot called timing blocks tell the op scan machine where to look for marked ovals. The blocks are made by the printer and tell the machine precisely where to look for the voter's mark. If the printer offsets the timing mark from the ovals, then the area over which the op scan will recognize a vote is reduced. Thus slightly variant marks in the ovals may not count as recognized votes by the scanner.

Craft recognized the attack threats described by Jones and Shamos as real and serious. However, he pointed out that simple mitigation techniques already exist in the form of effective management of the process by election officials. Checking configuration, proofing ballots, as well as testing machine components well before the logic and accuracy tests, will mitigate each of these threats.

Rivest noted that with calibration threats, there is plausible deniability for the attacker. With respect to the scanning attacks, a feedback mechanism would be useful as a mitigation tool. In converting from the analog (paper ballot) to the digital (op scan electrical record of vote), feedback would provide the voter with assurance that the vote was recorded as intended. Such feedback would be in line with the concept of equivalency in independent dual verification.

Selker commented on the critical need for backup of the configuration files as an essential election management technique. Redundancy is a key mitigation tool. With regards to op scan recognition technology, he noted that China employs a more reliable approach where an image of the ballot is recorded.

Craft emphasized the need for concrete guidance from this threat analysis effort for local election officials by the 2006 election. There needs to be a determination for each type of voting system of realistic risks and mitigations that election officials need to take.

Rubin agreed with Craft's call to action. Listing the threats is important because it begins a process of determining mitigation efforts. In a way, the calibration problems can be viewed as similar to the malicious software problems in that you can mitigate both with independent dual verification. If you are concerned that the op scan machine is not counting correctly, you can mitigate with random manual counts and pairing the results. Another op scan ballot marking device is referred to as the "$5000 pencil." Using a touch screen, the voter makes their selections on the ballot and prints it out. The marking machine makes the selections correctly with the mark locations hard coded into the voting system. The ballot is then run through the op scan counter. If the attacker is in collusion with the op scan equipment manufacturer, the scanner can be set to offset the marks and read them incorrectly.

Craft noted that neither the Automark system described above nor the various voter verifiable paper trail schemes are mitigations to threats. They are new types of voting systems, each of which need to be cataloged for their own lists of potential threats and attacks.

DeBeauvoir addressed the need to identify real threats and the role of officials in the field conducting elections in both identifying risks and establishing mitigations. Election officials believe that threats and mitigations need to be determined on the basis of a formal risk assessment. Many of the current threats have been proposed by people who are unfamiliar with elections and formal risk assessment procedures. Threats need to be analyzed in terms of the specific equipment used, the specific elections being run, and the control procedures in place. Travis County, Texas, has devised an end-to-end process model that identifies the areas of risk. This allows election officials to logically organize and deal with each individual threat. DeBeauvoir proposed that NIST and the TGDC come up with a list of minimum “common place” technical and procedural controls under which election officials can operate. In addition to the controls, hash code testing offers a basic setup validation tool. Additionally, parallel monitoring has become a politically manageable and common sense mitigation tool that the public can understand in terms of addressing potential attacks in a DRE environment. Chain of custody procedures need to be quantified on the basis of rules of criminal evidence. These include audit logs and tracking checklists, knowledge of the person that created the lists, and methods of securing the evidence. All of these procedures have to be determined for individuals unfamiliar with both risk assessment and assuming little or no funding for the efforts.

Rubin emphasized the need to think in terms of defense in depth covered previously by Jones. Identification of threats and mitigation efforts do not work if minimally trained poll workers do not read the procedural manuals. What is the fallback defense in this case?

Jones noted that management controls work only if the managers implement them effectively. Massive lists of threats and procedures will not work. There needs to be simplification for election officials as well. Also no-fault absentee voting and voting by mail require specific threat analyses and mitigation procedures. Finally, the disconnection between state laws and voting mechanisms is important. Op scan markers in the field actually count circles drawn around ovals in violation of Hawaii’s state law. The vendor documentation does not provide the acceptable mark criteria. Instead, you need to test for the counting capabilities of the system to see if it conforms to state law. This includes testing with marking devices other than a number-two pencil.

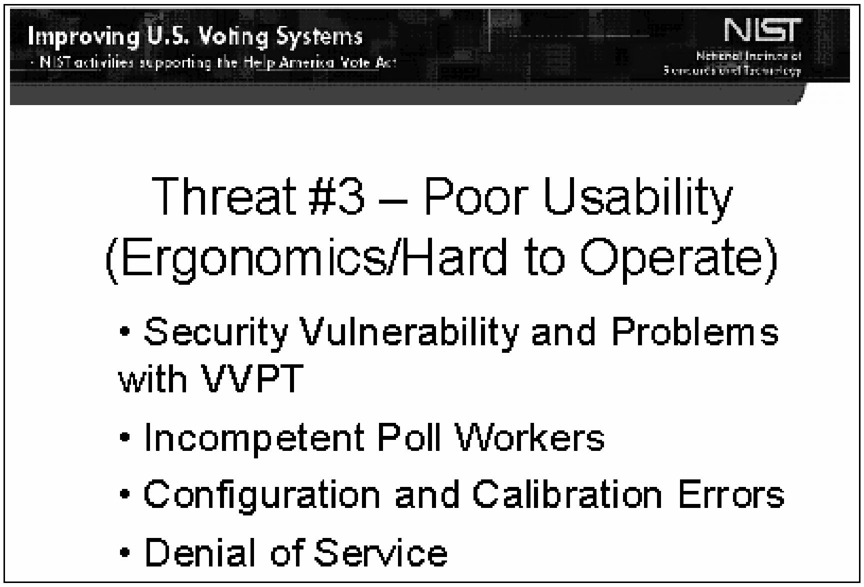
Creelan brought up the legal concept of “burden of proof.” When dealing with the plausibility of threats, you are implicitly dealing with assumptions about the burden of proof. Shamos’ previous discussion of whether we believe the “omniscient hacker” is relevant here. To rephrase this in a religious analogy, if the person is an atheist with respect to belief in the “omniscient hacker” on one side of the spectrum, the true believer in the plausibility of every attack is on the other side of the spectrum. Creelan takes the position of the agnostic, in the middle. We do not know in many instances whether a specific threat is plausible. The question then is do we err on the side of placing the burden of proof on the true believer or the atheist? Perhaps it makes sense to be agnostic, but religious at certain moments, protecting against threats when we are not sure whether they are going to happen. In other areas of law and regulation, the concept of technology forcing is very much part of the debate. If you want to get to an objective, such as reducing real threats to elections, you need to do more than assess current technology. There is value in standards that force technology to develop in new ways to address the threats in the long term.

Craft noted that this discussion brought up the point that while these are threat questions we are asking now, all election officials need to ask these questions with every election cycle. As a manager, you have a finite amount of resources with which to address security threats. You have to evaluate the plausibility of threats in your circumstances. You have to assign priorities and make decisions. At the end, you have to take your experience and go back through the threat model again. It is an ongoing process.

Rivest addressed the specific threat to configuration files and the use of hash functions to compute the digital signatures for various pieces of executable code. It is important to understand that you have different classes of objects that are not fixed and static but change with each election. The objects come from the national, state, and local levels. If you want to authenticate those components of the configuration files, you will need digital signatures on each of those objects which will check them dynamically with each election.

In summary, Guttman noted consensus that the threats discussed by this panel are plausible. She noted that there appeared to be agreement that the attacker would need to have some technical knowledge. Craft noted it would take technical knowledge to keep these threats from occurring. Guttman noted the panel agreed that countermeasures would be classified as managerial but if the procedures are too onerous, they would not be carried out. Craft noted that the countermeasures need to be as simple as possible. Damage that would occur if the attack was successful could include electing the wrong person.

Guttman then asked the panel to discuss the next group of usability threats (Figure 31) and address the requisite questions (Figure 32).



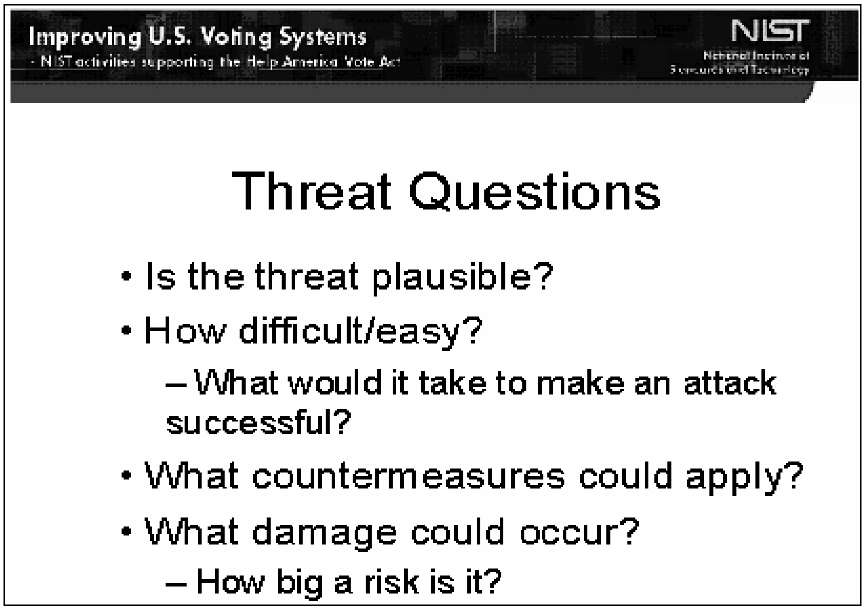
Improving U.S. Voting Systems
- NIST activities supporting the Help America Vote Act

NIST
National Institute of Standards and Technology

Threat #3 – Poor Usability (Ergonomics/Hard to Operate)

- Security Vulnerability and Problems with VVPT
- Incompetent Poll Workers
- Configuration and Calibration Errors
- Denial of Service

Figure 31



Improving U.S. Voting Systems

- NIST activities supporting the Help America Vote Act

NIST

National Institute of Standards and Technology

Threat Questions

- Is the threat plausible?
- How difficult/easy?
  - What would it take to make an attack successful?
- What countermeasures could apply?
- What damage could occur?
  - How big a risk is it?

Figure 32

Selker noted that the most plausible instances of hacking exist when people individually have access to the systems. Registration is an example where people put hurdles in front of the voter. Public sources of voter information can result in attacks on Internet sites that provide incorrect information to the voter. Jurisdictions need to take precautions with printed material in much the same way that workers at the U.S. mint handle money. Transmission of election data at the end of the day requires strict procedures. Too many election procedures are carried out by a single person.

Selker went on to state that you need multiple non-colluding “hands and eyes” to prevent and to detect attacks. In the voting process, proper setup of materials for election officials is critical to an effective process where someone checks another person’s work. We also need to work on cryptographic solutions that make for simpler voter usability. Audio verification is an example where this second record is used as a redundant perceptual feedback at the time the voter makes a decision. By making the voting process simpler, you increase accuracy.

DeBeauvoir emphasized that you cannot confine security to just the voting system itself. Election administrators must consider the whole election process. You cannot separate out just the voting equipment and hold a secure election.

Shamos highlighted the difficulties of confusing user interfaces with DRE voting systems. These systems vary tremendously in their ease of use. He illustrated the difficulty with the concept of the unexplained undervote. The theoretical minimum estimate for undervoting is .5 percent. This means that approximately one voter in two hundred is unable to cast a vote when they enter the voting booth. An undervote of 2.5 percent means that 2 percent of the undervote is unexplained. Some attribute the 2 percent to deliberate malicious manipulation of the voting machine. Others attribute the unexplained undervote to machine error. Shamos suggested that inadequate user interface in combination with machine error is the source of the unexplained undervote. Inadequate user interfaces result in the voter leaving the voting booth believing they voted one way, but because of the interface, the machine did not record the vote that way. As an example, in a multi-page ballot, a voter who makes and then cancels a straight party vote will not know the effect on the unseen pages unless they verify each page.

Rubin noted that poor user interface can result in a loss of privacy when the voter has to ask a poll worker for assistance.

Jones referred to the incompetent poll worker attack (see: http://vote.nist.gov/threats/papers/incompetent_pollworkers.pdf). Spoiled ballot processing always involves poll workers. The likelihood of a voter being offered the chance to spoil their ballot and the subsequent likelihood of the process being carried out properly is dependent on the competency of the poll worker. Jones characterized poll workers as a weak link in the security chain. He noted that they are hard to recruit and retain from election to election. Deliberately tampering with the poll worker pool by assigning competent poll workers to precincts that are demographically likely to support a particular candidate and incompetent ones to precincts where voters are not likely to vote for that candidate becomes a plausible attack. Given the competency level of many poll workers, this attack would be difficult to distinguish from an accidental event.

Craft took exception to the characterization of poll workers as a weak link in the chain. They may be a point of risk to which election administrators need to give attention. However, the voting process is extremely complex and depends on many dedicated volunteers. The main reason for the success of the process as a whole is the importance volunteers give to their public service.

Jones replied that he may have incorrectly stated the issue and that he had respect for the people who volunteer unselfishly as poll workers. However, handing a two-inch binder of hard-to-follow procedures to poll workers “borders on the inhumane,” and you cannot expect them to read it.

Rubin asked Craft if thought that poll workers were one of the least deterministic aspects of the election process. Craft agreed that they are an area of very high risk. They are an area that deserves a tremendous amount of election management’s attention. To simply hand them a two-inch binder without sufficient training is negligent management.

DeBeauvoir stated that in many jurisdictions, poll workers are recognized as a group for their bipartisanship and independence in that they watch over each other. Today, poll worker training is not given the minimalist approach of the past. In fact, they become mitigators for threats to the process.

Craft noted that the recruiting and training issue comes back to effective election management. There are election administrators that require poll workers to pass tests before they are allowed to participate in the process. Election administrators that do not follow this procedure due to a lack of poll workers need to go to their county administrators to obtain funding to hire more competent poll workers.

Selker described a spectrum of poll worker training experiences. Poll workers trained conceptually in Chicago with complex materials tended to become confused and make mistakes on Election Day. Poll workers trained procedurally in California with well-crafted and easy-to-use manuals appeared to carry out their assigned tasks effectively.

Shamos offered perspective on the number of poll workers in the United States, 1.4 million, which is larger than the size of the U.S. Army. There are estimates that two million poll workers are needed to provide adequate support on Election Day. Ten levels of officer grades manage the Army out of the world's largest office building. It is unrealistic to think that the funding exists for a similar management structure to effectively train and manage poll workers to operate at a high level of efficiency. However, the current deficiencies of poll workers are probably due to inadequate election management.

In defense of Jones, Rubin indicated that of the three areas susceptible to security vulnerabilities-procedures, equipment, and poll workers- it is the poll workers that are possibly the least predictable.

Craft reemphasized that poll workers are often the mitigation against many security attacks. For example, competent and well-trained poll workers will handle denial of service strategies efficiently.

Selker indicated that activist poll watchers can intimidate poll workers and represent a security problem as well.

Shamos mentioned that poll workers will be faced with supervising the use of new voting equipment as a result of HAVA. A poll worker checklist of tricks that people may try to subvert the election with the new equipment would be useful.

<u>Panel 2</u> Audience participation:

*Question/Comment:*Epstein referenced the IT industry's reaction to the Morris worm as the first large-scale attack against what became the Internet. The industry changed how it checked products for security in the aftermath and constantly became aware of and reacted to new types of security attacks. How does the panel propose to do retrospective testing of voting equipment that becomes certified for threats established today?

Shamos described the procedure in Pennsylvania. For a fee of $450, any ten voters can compel the commonwealth to reexamine any voting system in use should a new threat appear. Systems have been decertified in the past when determined to be unsafe.

Craft agreed that the threat review is a constant process initiated after every election cycle in preparation for the next.

*Question/Comment:*Browning addressed the poll worker problem from the perspective of an election administrator. The voting process cannot operate without poll workers. It is a people-driven process. Election management problems that came to light in 2000 existed in previous elections. People policies and procedures always have been and always will be the key to a successful election.

Jones added that there can be an overemphasis on management and an under recognition that technology can, in certain instances, help reduce the need for new procedures. He referred to a hardware design of a memory card that would make it impossible to connect it to a modem (to send election results) without removing the memory card.

Browning replied that while we are trying to simplify policies and procedures, the voting process has, over time, become increasingly more complex.

*Question/Comment:*Fisher asked the panel to address the issue of the insider threat with respect to chain of custody. Does it make sense to have a national certification and accreditation process for election administration at the state and local level? Also, is there a threat with absentee voting or with voter misidentification?

Craft responded in the affirmative to all three questions. The EAC is working on a federal certification program which will assume the role of the NASED voluntary accreditation program. Absentee ballots pose a high security threat even with stringent laws. Insider threats are an issue. You need to have effective management, separation of duties, and screening of workers to mitigate the threat.

Creelan addressed the voter misidentification issue. As distinguished from documented insider election fraud, voter fraud has not been shown to exist in great numbers. When you consider risk analyses, assumptions of individual voter fraud are somewhat baseless.

*Question/Comment:*An audience participant raised the issue to the panel of all-inclusive certification of voting systems to include the support materials for poll workers.

Jones indicated that he advocated that all voting systems include the support material needed to administer them. Vendors tend to be reluctant to tell the purchaser or examiner to guard against certain threats. He illustrated this point with an inadequate explanation for poll workers on calibrating the touch screen.

Selker agreed that system support information for poll workers needs to be at an understandable level for poll workers with limited education. He also indicated that the right qualification test for poll worker competency was not necessarily a written IQ type test but rather a performance test where the poll worker demonstrates their capability to carry out required tasks.

DeBeauvoir discussed the management issues surrounding those individuals who believe they are entitled to be poll workers but are not judged competent in dealing with the operation of new computerized voting equipment and election procedures.

Question/Comment: Coney posed several questions for the panel related to chain of custody of voting equipment, including the voter activation card. Coney commended the panel. She referred the panel to the state of Maryland’s poll worker “debriefing” as a useful post-election feedback procedure for identifying new threats. Also, poll workers were sent a survey to fill out on their experiences both in training and on Election Day. This is a model worth emulating in other jurisdictions. An initial question for the panel concerned the role of privacy and transparency in making elections more secure. Are the processes intertwined within security issues? A second question dealt with security related to voter access and activation cards.

DeBeauvoir indicated that procedures need to be in place to account for all of the voter activation cards at the end of Election Day. You document the number of cards used and compare that figure with the number of voters. To the extent that you discover missing voter access cards, you so document that in writing and install procedures to prevent this from happening in future elections. Again, this is part of the continuous improvement cycle for security procedures.

Coney asked if these cards, when used in future elections, pose a security threat.

Selker noted that these "smart cards" are programmed uniquely for each election.

Rubin indicated that if an adversary obtains these preprogrammed voter access cards, it provides only minimal assistance in subverting a future election.

Craft emphasized that a prudent election administrator will re-key the security information on the smart cards and the voting machines between elections as standard operating procedure.

Shamos recognized this case as a simple example of an instance where there is a simple managerial defense already available. The threat here is that a voter will save an access card and correctly reprogram it as validated for the next election. The card would provide a means for the attacker to vote twice in the next election- once with this card and a second time with the new card provided by the clerk on Election Day. However, with proper election management procedures in place, a poll worker would routinely check the public counter on the DRE between voters. The poll worker could then easily determine if a voter has voted twice.

DeBeauvoir brought up the issue of plausibility of this attack. It would seem more plausible for a voter to register twice with two different addresses and then vote in two different locations rather than to spend the considerable effort to correctly revalidate a voter access card to allow them to vote twice.

Question/Comment: Klein requested the panel's reaction to the issue of attacker profiles including the amount of money available for attacks by the entire attacker community. He provided an estimate of .25   billion dollars in a four-year election cycle in his position paper (see: http://vote.nist.gov/threats/papers/threat-modeling.pdf). The asset being protected is "governmental power." These are issues that should be part of a threat analysis. Secondly, the current discussion is one of matching the technology to the capabilities of the poll workers and the election administration officials. Non-technologically oriented individuals are being asked to watch for sophisticated attacks on complex voting systems. He posed the solution of paper backed up by evidentiary quality chain of custody procedures.

Craft addressed the issue of "dumbing down" the technology to meet the poll worker's capability. This is not an option in an environment where the operating requirements for the voting equipment have increased every year. An election administrator has to find poll workers with the capability to carry out the required procedures for a secure election. This may require a petition to state legislators for increased pay and benefits to recruit suitable individuals (retirees, government workers, etc.) to address the supply and demand challenges.

DeBeauvoir added that you can also have job descriptions for poll workers that match their level of skill. An election administrator also can create a smaller group of trained trouble shooters who travel around supporting election judges by answering questions and repairing equipment.

Rivest agreed with Klein that it is a fair assumption that the attack community has a fair amount of financial resources. As a nation, we need to address the underfunded effort to deal with the attack threats with improved technology and quality management procedures.

Selker referred to the quarter billion dollar estimate as hypothetical. He questioned whether paper was any less fallible than other technologies to security threats.

Rubin addressed the adversary issue from the standpoint of the substantial incentive to do harm. That incentive is control of the free world.

Craft noted that the debate over voter verified paper trail versus DREs and other voting technologies is moot. Congress has left the decision to the states regarding appointment of delegates, which translates down to state control on how they will conduct elections. The arguments over the methods of elections will never be resolved because individual jurisdictions will make their own decisions. The task at hand is to determine best practice to mitigate risk for each of the voting technologies in use.

Creelan asked a series of related questions of the other panelists that required a broader view of the purpose of the "Threat Analysis for Voting Systems" workshop. What do we mean by threats? Are we limited to deliberate attacks or are we including other areas where things can go wrong? Are we privileging security at the expense of other values including accessibility, equality, and usability? Are we limiting our concerns to voting systems and not the entire voting process, including registration?

Craft believed that great pains have been taken in the workshop to apply the broadest definition of threats to the entire process. The threats we are addressing are those events- accidental and intentional- that can cause an election to come to a wrong result.

Question/Comment: An audience participant asked the panel to address changing the dates of elections to weekends or holidays. She also inquired about a cost analysis of the election process. There seems to be no analysis of the aggregate cost of an election including registration and hiring poll workers as well as equipment costs, etc., is there a cost analysis of high-tech voting methods (higher costs) versus low-tech methods (lower costs)?

Creelan indicated that the Brennan Center is working on determining those costs. The ultimate goal is to create a cost calculator where an election official can enter in variables particular to a jurisdiction. The output would be a range of costs for various systems. It is a complicated exercise. "Apples and oranges must be reconciled." Currently there is inadequate information to make available to the election administrator, and only incomplete conclusions can be drawn on election purchases.

Panel 3- Wrap Up, Conclusions, Next Steps

Donetta Davidson, Ray Martinez, Mark Skall, John Wack, Linda Lamone,
Panel 1 members and Panel 2 members

Moderator: Barbara Guttman, NIST Information Technology Laboratory

The new panelists introduced themselves.

Donetta Davidson, EAC Commissioner
Ray Martinez, EAC Commissioner
Linda Lamone, Maryland State Director of Elections
Mark Skall, NIST, Information Technology Laboratory
John Wack, NIST, Information Technology Laboratory

Guttman introduced the goals of the final panel - to summarize where we have been and where we go from here.

Lamone expressed four summary points relating to consensus issues expressed at this workshop and an editorial comment. She thoroughly endorsed minimum quality assurance standards and guidelines for the manufacturers of voting equipment that include documentation standards. Secondly, election administrators in the field need guidelines, standards, and best practices for logic and accuracy tests, chain of custody, and parallel testing for all of the different types of voting equipment. Thirdly, the scientific and academic community needs to work closely with the elections administration community. Security-related problems will get solved only if we work together. Lastly, and most importantly, security discussions such as those at future workshops need to focus on existing voting systems. In order to be HAVA-compliant, election jurisdictions are making or have made purchase decisions. The 2005 voting standards proposed by the TGDC and adopted by the EAC will deal primarily with current technology. Election administrators need help making sure the voting systems they purchase are manageable and secure. Lamone offered an editorial comment that the failure rate in Maryland in the 2004 election was less than 1 percent, contrary to what some advocates say.

Skall summarized what he hoped NIST and the TGDC would take away from this workshop. Looking at our goals for this workshop, we identified many of the threats as plausible. There is clearly still much work to be accomplished in the area of threat analyses of election systems. A successful end result would be future security requirements proposed by the TGDC and delivered to the EAC that are traceable back to specific threats. Cost to the states to adhere to these requirements could be substantial. Quantification of threats represents a difficult task. We might look at this as “expected value” or “expected damage,” which would be the probability of a threat times the actual dollar value of the damage. If we could determine this value, we could give more guidance to the TGDC as to how much time to spend on requirements that address specific threats.

Commissioner Davidson thanked NIST for starting the process of addressing the issues of voting system security in terms of threat analyses. Activities we are undertaking at the EAC will assist election administrators with management issues raised at this workshop. We have a number of studies underway to determine best practices for election administration at the state and local levels. EAC will be working with NASED on management guidelines. As we have heard at this workshop, training is part of the risk assessment process. When considering resources to deal with threats, it is important to include small and medium-sized counties and jurisdictions in the analysis. As well as money, technology resources represent a challenge to smaller municipalities. This workshop represents a good start and provides the voters with trust and confidence that we are addressing the security issues critical to fair and safe elections.

Commissioner Martinez expressed his gratitude for the large turnout for this productive workshop, especially the discussion of the threats to voting systems in general. The EAC is striving to make progress in the area of election administration. HAVA was meant to improve the three-legged stool of election administration- the technology we use, the processes in place to ensure fairness at the polling place on Election Day, and the people involved in running the election. With HAVA, Congress appropriated $3.1 billion to improve all three aspects of election administration- the technology, the processes, and the people. An example is that jurisdictions are using the money to switch from lever machines and punch cards to op scan or DRE technologies. With respect to processes, HAVA also requires states to look at their election codes and to better define what "the intent of the voter" means. Finally, HAVA dollars are intended to help ensure that election officials and poll workers have the training to do the job correctly. In a December 2004 Wall Street Journal poll with a sample size of one thousand, 24 percent of those polled indicated they had little to no confidence that the vote they had cast had been correctly recorded. While the Commissioner is certain that election administrators are working diligently to ensure the integrity of the elections, we cannot deny the issue of lack of voter trust, and we must deal with it. Jones pointed out in his paper (see : http://vote.nist.gov/threats/papers/threats_to_voting_systems.pdf) that we need to solve this problem voluntarily before Congress or state regulatory bodies decide to solve it for us (i.e., a regulatory scheme is mandated). The Commissioner stated that he believed a voluntary cooperative effort to improve voter confidence is preferable to a regulatory scheme. This requires frank and earnest discussions in areas such as security threats to voting systems. It also requires pulling together best practice tools for state and local election administrators so that they can improve the election management process. Innovation is critical to improving trust and confidence of the voters.

Rivest addressed the issue of determining the probability of various attacks. If you think about any system from an attacker's point of view, you will try to go through "the open door" and not "the closed window." Cryptography succeeds when it is no longer the weakest link. If you have a voting system with many components and resultant vulnerabilities, there will be many different attacks you can employ. You cannot determine the probability of an individual attack any more than you can determine the probability of someone choosing a particular window or door, regardless of whether it is open or closed. It is a large-scale contextual problem. The right questions to ask are, 'What is the difficulty of achieving a particular attack?' and 'Where is the weakest link' in the entire system?" That is where the probability will be highest. Voting is interdisciplinary and requires input from people involved in every part of the election process to make it work better. We should have conferences like this devoted to analyses of voting threats.

Jones returned to the subject of a cost-benefit analysis of voting threats. Economic analysis of security is inherently difficult because you are spending money to avoid risk. Your most successful purchase with security is one where you never notice the benefit because you do not see an attack. You have spent the money well because you have deterred the threat. Therefore, economic analysis of risk produces almost bogus numbers. Sometimes money spent by management produces defenses against threats that were not even anticipated. The Y2K spending is an example of one such threat which put into place necessary redundancies to survive the anthrax attack. Jones also commented on the previous discussion of the "omniscient hacker." Looking back at Harris' 1934 analysis of threats to elections (see: http://vote.nist.gov/election_admin.htm), the hot topic for threats then was mechanical lever machines. There was a push to replace paper voting machines with paperless mechanical lever systems because they were allegedly more resistant to fraud. By the 1950s, the fraud possibilities were quite clear; mechanical voting machines could be and were in fact rigged long before it came to public notice. Thus, we cannot take the risk of claiming that we have the exhaustive threat catalog. While the vast majority of election jurisdictions have lived up to high standards, a minority have not done so.

Shamos agreed that no threat should be dismissed out of hand. Every threat deserves serious consideration even if the response is that we consider it unlikely to occur. The way to do this is to continue assembly of the threat catalog. Then it makes sense to develop standards (requirements) and cross-references from the standards to the threat catalog. When we find a threat that is not addressed by a corresponding standard, we have a wake-up call to address a risk that has not been covered.

Wack addressed the job of NIST and the TGDC to produce recommendations for future iterations of the VVSG. Several points discussed at this workshop will assist in this effort. Participants expressed a need for more documentation with respect to voting systems and procedures for poll workers. Voting systems can be much more secure if they are simpler. Voting system design is critical, especially with respect to usability by the poll worker as well as the voter. With respect to independent dual verification, more developmental research is necessary.

Skall agreed that many of the large economic studies are not useful and fail to provide the needed insight. On the other hand, even if we have specific requirements to address each threat in a threat catalog, there is no guarantee that the requirement will be precise or testable. You need to drill down these requirements until you have completely addressed the issues of testability and precision. NIST and the TGDC need some sense of priority of each of these threats to see which are the most important and where we should dedicate most of our time and resources. There has to be some quantitative analyses, rough as it may be, to arrive at a prioritization that gives direction to NIST and the TGDC.

Rivest referenced the Brennan Center approach described earlier by Lazarus as a useful first-cut metric. This threat analysis looks at the number of people required to carry out an attack. If you have a threat where one person can take away 1 percent of the vote, you have a serious attack that requires mitigation.

Wack noted that a number of speakers pointed out that a number of security problems are in fact usability-related problems. From a prioritization standpoint, we may want to determine what "user error" problems we can fix now.

Commissioner Davidson agreed here that we need to split off the technically challenging problems from the user-error problems that can be worked on up front. In some instances, different groups can deal with the different issues.

Jones expressed concern here about “too much dividing,” because it really is the case that every usability problem seems to be something that can be exploited in order to tinker with the vote. For example, if you can make things more usable in precinct 15 than they are in precinct 5, then you can effectively discriminate against the voters in precinct 5. That kind of strategy makes it possible to exploit almost every usability problem as a way to manipulate the election. That is why there needs to be a real cross connection between the voting systems standards and the best practices guidelines. Jones believes that, in many cases, the technology standards we have make assumptions about the ways the users are expected to use that technology. We need procedural documentation here. The voting system standards assume poll worker procedure standards.

Panel 3 Audience Participation:

*Question/Comment:*An audience participant asked the EAC Commissioners to address the roll of the EAC in updating the certification process when new vulnerabilities are discovered. Will the independent testing authorities (ITAs) at the state and federal levels receive feedback to update the voting system tests when new threats or vulnerabilities are discovered? Will the EAC oversee this certification review process?

Commissioner Martinez answered yes. There is some debate here concerning the role of the EAC. However, Congress has given the EAC the responsibility of becoming the certifiers of voting systems at the national level. The EAC is obligated under HAVA to transition from the NASED certification program to one administered by the EAC. We are still in the process of putting together the transition program and hope to have it in place in the next six months. We have a responsibility in the new certification program to keep track of patterns that would signal that a particular voting system has a particular vulnerability and to transmit that vulnerability to jurisdictions across the country, the ITAs, NIST, and the TGDC to address these kinds of problems.

*Question/Comment:*Lewis thanked NIST for holding this workshop on the security of voting systems. He made the point that perspective is important when assessing surveys of the confidence of voters in the perception of whether their vote was accurately recorded. In fact, surveys of voter trust have never been higher than 88 percent. He asked the panel whether he thought that Congress was committed to give NIST and the EAC what they need to improve voting systems.

Commissioner Davidson answered that, at least for FY 2006, Congress has funded the EAC at the same level as FY 2005. Beyond that, it is difficult to say.

*Question/Comment:*McClure asked the panel to address state statutes and their differing requirements with respect to electronic voting systems as well as the sometimes-conflicting relationship of state standards to federal standards.

Shamos noted that states began holding independent hearings in the mid 1980s to address these issues. Today, the electronic voting statutes in almost every state are quite detailed. They make careful distinctions between paper-based, lever systems and electronic systems. There are conflicts that states such as Pennsylvania have had to address with respect to interpretation by the vendors.

Commissioner Martinez addressed the issue of determining an effective date for the new voluntary voting system guidelines. In trying to determine this date, the EAC looked at state election codes and found that there was problematic wording that would play into any effective date the EAC chose. The EAC is taking into consideration how state laws and administrative procedures deal with decisions made at the federal level.

Question/Comment: Klein gave credit to the state of Maryland for conducting the first penetration tests of voting systems. However, that test was not comprehensive and did not rise to the level of a systematic search for vulnerability of critical systems that is included in documents such as the common criteria. In his comments on the VVSG, Klein notes that the lack of such penetration testing basically negates most of the security improvements. Serious security testing needs to be part of any program that goes forward.

Rivest agreed that a test of open-ended vulnerabilities is important in any security review. The TGDC has passed a resolution authorizing NIST to develop standards in that direction. Those are not part of the current VVSG because we had to prioritize NIST's work in a limited time frame to produce the current version. Rivest hopes to see future development of these standards.

THIS PAGE LEFT INTENTIONALLY BLANK

APPENDIX

THIS PAGE LEFT INTENTIONALLY BLANK

Chain Voting

Douglas W. Jones

Aug 26, 2005

Taxonomy

Retail, vote buying, or voter intimidation.

Applicability

Paper ballot systems (hand counted or mark-sense).

Method

The perpetrator must begin by obtaining a valid blank ballot for each precinct under attack. The perpetrator may counterfeit a ballot, steal a ballot before the election, smuggle a legitimately issued ballot out of the polling place instead of voting on it, or use an absentee ballot.

The perpetrator then repeats the following cycle: Mark the ballot for the desired candidates, find a subverted voter, and require that the subverted voter take the ballot to a polling place, exchange the pre-marked ballot for the blank ballot issued to that voter at the polling place, and return the blank ballot to the perpetrator to enable the next cycle.

Chains can also be run among subverted voters who have requested absentee ballots (or have been induced to request them for the purpose of participating in a chain). In this case, the initiator of the chain marks his or her absentee ballot and then gives it to a subverted voter in exchange for a blank absentee ballot, continuing to build the chain until the deadline for returning an absentee ballot, at which point the initiator marks and votes the last ballot in the chain.

Voters expecting payment receive their payment after returning the new blank ballot to the perpetrator. Voters are typically subject to punishment if they do not return the blank ballot.

Resource Requirements

Each perpetrator must have access to a pool of subvertable voters willing to vote in return for payment or unable to complain if threatened. Employees, tenants, and those with similar dependency relationships are particularly vulnerable.

Potential Gain

One vote per subverted voter.

Likelihood of Detection

The likelihood of detection depends on the degree of dependency linking the perpetrator to the subverted voters. Chain voting is fairly safe for the perpetrator where he is in a position to offer protection to voters in desperate circumstances. Examples include: protecting their jobs in times of high unemployment, or their leases in times of housing shortage, or their access to essential government services.

Countermeasures

Preventative Measures

Ballot Distribution Security:

> Strictly account for all ballots printed, with the requirement that all ballots not packaged for delivery to the polling place be destroyed. Multiple witnesses must be present at every stage of ballot processing to assure that no ballots escape. When the polls open, election workers must verify that the inventory of ballots delivered matches the manifest for the polling place.

Absentee Ballots:

> Mark absentee ballots distinctly to distinguish them from ballots voted at the polling place.

Prevent Ballot Counterfeiting:

> Use special inks and papers to deter counterfeiters.

Serial Number Ballots:

> Each ballot should have a unique serial number printed on a tear-off stub. When the voter signs in to vote, this serial number should be recorded. When the voter returns his or her ballot to be deposited in the ballot box, this number should be checked to verify that the voter is voting the same ballot they were issued. If the stub is already torn from the ballot or if the stub number is wrong, the voter should be subject to investigation and possible arrest. To protect voter privacy, the ballot should be contained in a privacy folder that exposes only the ballot stub and serial number, and the stub should be removed before the ballot is slid from the privacy folder into the ballot box. Alternatively, using serial numbered ballots: Note the time of issue of each serial numbered ballot, without noting the identity of the voter to whom that ballot was issued, and use this to enforce time limits on how long a voter may take to vote a ballot.

Detection Measures

Detection is difficult if markings on the ballot are made with pedantic attention to the ballot marking instructions, for example, by exactly darkening the ovals or making perfect X-marks with exactly the recommended type of pen or pencil. However, if someone has been marking many ballots, they are likely to develop a fast marking technique that may be visibly distinctive enough to be recognized from ballot to ballot. This has led, in the past, to detection of a "single hand" that marked many ballots.

Citations

Joseph P. Harris, Election Administration in the United States, The Brookings Institution, 1934. Chain voting is described on page 373. The use of serial numbered tear-off stubs is described on page 40. The potential use of absentee ballots to start a chain is described on pages 298 and 299. The risks of postal voting discussed on pages 301 to 303 do not include the applications of chains in this context, but they are fairly obvious.

Harris considers chain voting to be worthy of defending against, but he notes that it was secondary to other types of fraud that were, at the time, easier.

Retrospective

Despite the fact that the defense against chain voting was well understood and published in 1934, many states have not adopted these defenses and rely on inferior defenses. In several cases, states are still using methods that Harris explicitly criticized as being weak and ineffective such as having poll-workers initial or sign each ballot.

Prevention of ballot counterfeiting is far more difficult today than it was in 1934! Computer typography and the widespread availability of photocopy shops with a good supply of paper make most classical ballot security measures pointless. Many vendors of mark-sense voting systems claim that their ballots must be printed on special paper with special ink, but in the late 1990's, I disproved one vendor's claim by manufacturing counterfeit ballots at a neighborhood copy shop that neither the vendor's representatives nor their machine could distinguish from authentic ballots.

Some counties have apparently posted, to the web, the actual PDF files from which the official ballots were printed. This is easy, but it makes things very easy for a counterfeiter. I collected one such ballot from the web soon after Election 2000.

Threat to voter privacy with voter verified paper audit trail voting systems using spooled paper rolls

Taxonomy: Retail, vote buying or voter intimidation
Applicability: DRE voting systems with voter verified paper audit trail capability using spooled paper rolls that remain intact (uncut) post-election

Method:
This is an attack on voter privacy that is possible when using a DRE with a voter verified paper audit trail capability that uses a spooled paper tape to record the voter's choices. The spooled paper tape records each voter's choices in the same order as voters using the DRE.

This attack is relatively simple: The perpetrator watches the order in which people use a particular voting system and notes the order of each particular vote he is interested in. At some point after the election, the perpetrator or a counterpart obtains the paper tape and compares the order of ballot records with the order of individuals who used the voting system on Election Day.

This attack could be used to enforce vote selling, or simply to invade the privacy of voters and determine how particular individuals voted.

Resource requirements:
If the purpose of the attack is to sell votes, the perpetrator must have access to a pool of subvertable voters willing to vote in return for payment or unable to complain if threatened. The perpetrator must also watch the order in which people use the voting systems, which could be done rather easily by using a hidden camera. To get access to the voting system’s paper tape, the perpetrator must have access to the voting system post-election. This could occur in a number of ways, including subverting the physical security of the voting systems or by cooperation with a dishonest election official.

Potential gain:
One vote per subverted voter.

Likelihood of detection:

If the purpose of the attack is to sell or coerce votes, it depends on the degree of dependency linking the perpetrator to the subverted voters. It also depends on the ability of the perpetrator to take the paper tape, examine it, and then replace it without detection. Some paper tape units are sealed and provide some physical tampering indications; however a skilled and determined perpetrator could likely overcome these obstacles. Election officials may not be in a position to detect evidence of tampering or may attribute it to accident.

Countermeasures:
Appropriately-strengthened physical security on the election systems post-election will reduce the risk of this attack succeeding, unless the perpetrator is working with a co-conspirator who has physical access to the voting system. Use of tamper-resistant paper tape units that offer a very reliable physical indication of tampering would help. Also, cutting each ballot record from the paper spool will help to randomize the order of ballot records, thereby making the attack extremely unlikely to succeed.

Citations:
The risk of this attack has been cited frequently in newspaper articles, testimony on voting system security, and in many voting system research publications.

Retrospective:
Several voting system vendors use spooled paper rolls to record voter's ballot choices. The use of spooled paper tape units presents a dilemma, since the units if intact may be significantly easier to handle than separate sheets of paper or pieces cut from a paper spool, and therefore may have greater integrity associated with them. On the other hand, they represent a threat to voter privacy that can only be mitigated by tamper-resistant units and strong election procedures.

EXAMPLE ATTACK DOCUMENTATION

Optical Scan Configuration File

Douglas W. Jones
Sept 15, 2005

Taxonomy: Administrative, wholesale
Applicability: All voting systems

Method:

Typical mark-sense ballot scanners have a single mark sensing mechanism positioned over each column of the ballot, plus a sensor that scans down a column of index marks to sense what row of the ballot is passing under the scanning head. Thus, the scanner does not sense a vote for a particular candidate, by name, but rather, it senses a mark at the intersection of a particular row and column. The ballot text sent to the printer specifies the candidate name to be printed next to each voting target, and it specifies the positions of the voting targets. The vote tabulator does not read the text of the ballot, but rather, it must be configured, using a configuration file, so that it can relate the coordinates of marks it finds on the ballot to the names of the candidates. This mapping is sometimes a two-level mapping from ballot coordinate to candidate number, and then from number to name.

If the perpetrator can edit the ballot configuration file for a precinct, the perpetrator can do such things as making the scanner credit one candidate with votes intended for another.

Resource requirements: The perpetrator must gain access to the configuration files. These files are typically exposed in the computer system used to prepare the election, so they are available to the technicians setting up the election. Typically, these files are transferred to the mark-sense tabulator using removable media such as disks or PCMCIA cards. Anyone with access to these media could potentially attack the system.

For precinct-count mark-sense systems, attacks on one precinct could be done by someone who has access to these media before the polls open.

Potential gain:

All votes cast on the machines that have been may be corrupted. A serious thief must consider how to avoid being noticed. Adjusting the configuration files so that votes for one or more minor party candidates will be added to the total for a major party candidate is probably the safest attack. Another moderately safe attack is to exchange the totals for two candidates who are expected to attract comparable totals.

Likelihood of detection:

So long as the tinkering is done carefully, the likelihood of detection is small.

Countermeasures:

Preventative measures:

Authentication of the configuration files can protect against outsiders attempting this attack. This does not protect against insiders with access to the configuration files prior to their being authenticated, so voting system designs that prevent access to these files should be preferred.

Secure transmission of configuration media can help. Configuration files should not be loaded into voting machines if those machines are left in insecure locations for extended periods before the polls open.

Optical scan systems that actually read the ballot instead of just looking for marks at designated locations would be possible. It is conceivable that such scanners could be designed so that there was no need for a configuration file.

Detection measures:

Report vote totals by ballot position as well as by candidate name. This would expose the contents of the configuration file in the canvass, so that anyone could compare the positions reported in the canvass with the actual positions on the ballot.

Pre-election tests can help, but only if the test is performed with the same configuration file as is used in the real election, and only if the test includes different numbers of votes for each candidate, in order to assure that the vote totals for candidates are not exchanged.

Post-election auditing can help, for example, following the California law where one percent of all precincts, selected at random, are recounted after each election.

Recount laws that allow a hand recount of the actual ballots are an important defense. Recount laws that require use of the same tabulating equipment and the same configuration files as used in the first count serve to actively prevent detection of this category of error.

Citations:

Configuration file errors have been noticed on DRE and optical scan equipment. Franklin County Indiana had such a problem in 2004, in which straight party Democratic votes were credited to the Libertarians.

Inadequate pre-election tests that could not detect this type of tinkering are widespread. See
http://www.cs.uiowa.edu/~jones/voting/miamitest.pdf
(section 1, pages 1 to 3).

Retrospective:

There is no widespread understanding of the number of levels of indirection in the linkage between ballot marking location and candidate name. This comment applies equally to all electronic voting technologies from the Votomatic to the newest touch-screen voting systems.

The Potential Gain section needs modification.

The potential gain is not every ballot. Only the ballots whose positions were switched and where one of the switched positions were voted would be affected. Unless the attack was more of an attention grabbing type attack intended to be obvious, the gain may not even be effective. It is at this point that the author introduced the issue of the type of implementation because method used can have a big impact on the potential gain.

1. Pure switch. The candidate whose gain is desired (Cand A) is switched with a candidate who is expected to make a better showing (Cand B). The difference may be very small because most of the vote changes will cancel each other out. The difference will be the spread between the higher vote count and the lower vote count (typically, the absolute |Cand B Cand A| < Cand A). If the initial assumption that Cand B will be higher then Cand A is false, the change will result in a loss. Note that the change does not even need to be made in every precinct, especially if there are different positions due to rotation or placement in different precincts. (In many systems, a particular candidate s position can change between ballots due to rotation rules, trying to place the maximum number of candidates on one ballot, or just plain error. This can be particularly difficult to catch in ballot proofing because a Cand A position may be correct in, for an example, 9 out of 10 ballots and displaced in the 10th.)
2. Minor candidate add. A lesser candidate who is expected to get a few votes (Cand A and Cand B are both greater than Cand C) position is redefined to be the same as Cand A. Then Cand A = Cand A + Cand C and Cand C = 0 where A and C is the final count result. But in this case the gain is only Cand C (Cand B Cand A) This will tend to be subtle but may be so subtle it is ineffective (when Cand B > Cand A + Cand C). It is vulnerable to detection, especially when it is applied widely enough to effect every precinct because Cand C will have no votes in every case it is applied and observers are more likely to know and be able to prove that at least X>0 votes should have occurred..

Dr Jones claims that the likelihood of detection is slight if carefully done. This assertion is true only if good Logic and Accuracy (L&A) test procedures are not performed and/or physical security of the election program installation is weak. His paper does well in highlighting some common bad practices and issues in this regard. Unfortunately many voting jurisdictions are guilty of those bad practices and some are even encouraged in this by the vendors for other reasons. Best practices with the L&A and basic physical security of the election program installed will also be effective in many other threats to the ballot definition integrity and most ballot logic attacks that are not based on a time bomb or a swap out of control code after the L&A is completed.

This is a good example of a problem that is more likely to occur as an election programming error than a deliberate attack, especially where local procedures are so poor as to not detect it in L&A. Its more serious effect is that it can give a very graphic appearance of deliberate subversion of the election when it is only human error or incompetence.

Steven V. Freeman

Optical Scan Ballot Design

Douglas W. Jones

Sept 15, 2005

Taxonomy: Administrative, wholesale, probabilistic
Applicability: All voting systems

Method:

The perpetrator arranges the layout of the mark-sense ballots in such a manner that voters in favored jurisdictions are more likely to have their votes properly counted than voters in non-favored jurisdictions.

For example, where there are discretionary elements to the ballot layout, taking advantage of this discretion to create easy-to-read ballots in favored jurisdictions and hard-to-read ballots in non-favored jurisdictions. At the county level, for example, creating problematic instructions in some counties and clear instructions in others can be very effective.

Another effective design element is the false voting target, something that looks like the place where voters should mark their ballots but is in fact something else. A particularly popular version of this is a column of three-letter party abbreviations on the opposite side of the candidate names from the official voting targets and aligned exactly like them.

Ballot rotation can be used to make it difficult to hide popular candidates, in those states where rotation is mandated. Rotation is the listing of candidate names in different orders from precinct to precinct, nominally to avoid giving any one candidate the advantage of being listed first. Rotating an opponent's name into an obscure position in jurisdictions likely to favor that candidate can reduce the vote, particularly when candidate lists are long.

Finally, deliberate alignment errors on voting targets can be used, printing the target (or the index mark used by the scanner to locate the target) in such a way that marks in the printed target for the opposing candidate are less likely to be counted than marks for the favored candidate.

Resource requirements: The perpetrator must be in a position to control the design and printing of the ballots. For attacks targeted at the precinct level, this means that the perpetrator must either work for the ballot printer or the county. The printer can introduce alignment errors, while the county controls all of the textual content.

For attacks that exploit different ballot designs from county to county, the perpetrator must either control many county election offices or must work in a supervisory role at the state level. The state officer who approves ballot content can do quite a bit if he simply gives a free rein to incompetent county election administrators in counties controlled by the opposition while extending help primarily to election administrators in counties favoring the ruling party.

Potential gain:

Rates of voter error have exceeded 10% in some jurisdictions during some elections. If this error can be controlled so that these high rates occur primarily in communities where opposition voters are likely to vote, the net benefit, in terms of the final election total, could easily be on the order of 1% or more.

Likelihood of detection:

Anything involving ballot design is public record, and the ballots themselves remain to be examined for 22 months after the election. Should a candidate suspect that there has been deliberate misprinting of index marks or voting targets, this can easily be detected if the ballots are available for examination. There is a common catch-22 here: In many jurisdictions, attempts to examine the actual ballots have been blocked because the person wanting to make the examination had no proof that there was anything wrong. The proof, of course, rested in the ballots themselves.

Bad human factors in ballot design is so widespread that any deliberate manipulation of the design can be easily hidden or blamed on incompetent underlings or local officials.

Countermeasures:

Preventative measures:

Discretionary elements of ballot design should be minimized in order to avoid misuse of this discretion.

Pre-election tests of tabulating equipment should include hand-marked ballots as well as machine-printed test ballots. Ideally, the hand-marked ballots should include ballots marked by representatives of the public as well as by employees of the election department, although these latter should be screened, in public, for mismarkings that might be intended to deliberately foil the test and bring the election into disrepute.

Detection measures:

Sample ballots should be published that accurately demonstrate all relevant elements of the ballot, allowing anyone to compare ballots from multiple jurisdictions and identify potential sources of confusion. Unfortunately, this publication can also provide the information necessary to create the counterfeit ballots needed for chain voting or ballot box stuffing.

Unused actual ballots from the election could be made available for inspection as soon as this is possible without creating the possibility of fraud. Such ballots should be accounted for scrupulously, they should be defaced (for example, by being marked "sample ballot" in indelible ink), and released sufficiently long after the election that they could not be used as the basis for counterfeit ballots that could be used to manipulate the election.

These measures are of no value unless someone takes the time to critically examine the ballots disclosed by the government.

Citations:

In election 2000 in Florida, 5 counties had spurious voting targets such as DEM or (REP) to the right of the candidate name when the voting target (an oval) was to the left. In contrast, 27 counties had no obvious spurious target.
The remaining 7 optical-scan counties had intermediate designs. The statistical impact of this is difficult to assess because of other factors, but the rate of mismarking on ballots, as reported in the Miami Herald data, was almost 3 times the rate when there was an obvious false target than when there was not.

In election 2000 in Florida, 23 counties spelled out "For President" and "For Vice President" under each candidate's name, more than doubling the total amount of text on the presidential portion of the ballot compared to then 9 counties that listed the office names only once, at the head of the candidate list.
Again, the impact of this is difficult to assess because of other factors, but the rate of abstention (casting blank ballots) was about 2.5 times higher where office names were spelled out.

Again, from election 2000 in Florida, 24 counties split the list of presidential candidates across two columns of the ballot, while 12 managed to fit this list in one column. According to *Some had 1 from 'column A', 1 from 'Column B'*, Orlando Sentinel, January 28, 2001, the two column format was actually used in the sample ballot sent out by the state election office to those counties using ES&S central-count tabulating equipment.

Retrospective:

This form of election failure is clearly a violation of the voter's right to be weighed equally, but it is difficult to prove malice when so many ballots are routinely designed so badly. It is highly unlikely that most of the failures in this category are the result of deliberate fraud. Rather "this is the way we have always done things," or "this is the way the vendor told us to print the ballot," is probably the dominant explanation.

Incompetent Pollworkers

Douglas W. Jones

Sept 15, 2005

Taxonomy:

Wholsale, probabilistic, administrative
Applicability: All voting systems

Method:

The perpetrator ensures that the pollworkers in the favored party than the pollworkers in precincts supporting the opposing party. This may be done by deliberate assignment of pollworkers based on competence. It may be done by providing different quality of training depending on where the pollworkers are assigned, or by other means. A subtle way to do this is to provide only limited pollworker training through official channels and then offer supplementary training outside the system for selected pollworkers. An even more subtle way to do this is to assign pollworkers to their home precincts, relying on the educational demographics of the precincts to assure that, on the average, well educated voters have well educated pollworkers while poorly educated voters have poorly educated pollworkers.

In precincts with well-prepared pollworkers, fewer errors will be made, and the voters will therefore have a higher likelihood of having their votes counted. Typical pollworker errors include improperly turning away legitimate voters, improperly admitting illegitimate voters, failure to properly administer provisional ballots, failure to give proper instruction to voters requiring such instruction, failure to handle spoiled ballots properly, and many others.   Resource requirements: The perpetrator must be in a position to control the assignment of pollworkers or their training. Direct control makes this form of fraud easy, but indirect control, for example, through legislation, can be quite effective in cases where poorly educated voters are more likely to support the opposition party.

Potential gain:

Rates of voter and pollworker error have exceeded 10% in some jurisdictions during some elections. If this error can be controlled so that these high rates occur primarily in communities where opposition voters are likely to vote, the net benefit, in terms of the final election total, could easily be on the order of 1% or more.

Likelihood of detection:

Proof of deliberate discrimination based on this model is extremely difficult. Carelessness in election administration is so common at all levels that distinguishing between manipulated carelessness and random carelessness can be close to impossible. Similarly, educationally disadvantaged voters have a natural tendency to make errors, and this can easily mask the effects of this attack.

While difficult to prove, this attack is likely to be widely understood by the voters. Voters in precincts with poorly trained pollworkers generally notice their overall incompetence, and voters in precincts with good pollworkers generally notice that. Thus, this approach to election manipulation falls into the classic category of fraud classes where everyone knows about it but nobody can pin it on anyone.

Countermeasures:

Preventative measures:

Random assignment of pollworkers to precincts can equalize the training across the jurisdiction. On the other hand, this reduces the likelihood that the pollworkers at the precinct will personally know the voters. That makes the precinct more friendly and welcoming, and it can deter classic retail forms of vote fraud such as repeat voting.

Standardized education for all pollworkers can help immensely. Competently designed training courses, instructional materials and official pollworker manuals are very important.

Genuine objective tests of pollworker competence would be desirable, so that pollworkers could be selected based not only on residence and partisan criteria, but also on the basis of ability.

Detection measures:

Post election audits that count pollworker errors by precinct would be incredibly valuable. If the precinct-by-precinct or county-by-county error rate has a strong correlation with the electoral demographics of the precincts, this should be taken as strong evidence that this attack is taking place, although it does not pin the blame on anyone.

Election observers can also record pollworker errors for similar audits. Where observation is sufficiently widespread to get a statistical picture of the error rate across an entire jurisdiction to allow an examination of the correlation with the electoral demographics, this attack can be identified.

Citations:

Joseph P. Harris, Election Administration in the United States, The Brookings Institution, 1934. Widespread pollworker incompetence is discussed on page 35. Inadequate pollworker training is discussed on page 96.

Edmund F. Kallina, Jr. Courthouse over White House -- Chicago and the Presidential Election of 1960, University Presses of Florida, 1988. Widespread pollworker incompetence is discussed on pages 81 and 82, with considerable documentation based on election observers and press reports.

Joel Engelhardt and Scott McCabe, Poll workers ignored flaws in pre-vote machine tests, Palm Beach Post, December 9, 2001, demonstrates in one narrow area, pre-election equipment testing of Votomatic machines, that pollworker errors continue today at an astonishingly high rate. Over 11% of the test ballots used, at the precinct, on election morning, showed unpunched positions that should have been punched, yet not a single machine was reported as being non-functional. This is evidence that the test results were being ignored!

The two unofficial recounts conducted by the press after election 2000 clearly show correlations between voter error rates and both political and racial demographics. Unfortunately, there has not been a similarly intense examination of pollworker error rates.

Retrospective:

This form of election failure is clearly a violation of the voter's right to be weighed equally, but it is so difficult to prove and the institutions that lead to it are so entrenched that it is probably among the most difficult election failures to deal with. It is highly unlikely that most of the failures in this category are the result of deliberate fraud. Rather "this is the way we have always done things" is probably the dominant explanation.

Security Risks associated with pre-election delivery of Electronic Voting Machines

Barbara Simons

Attack names: hacking voting software and disenfranchising voters.

Applicability: security risks deriving from early delivery of voting machines.

Attack method: see below.

Resource requirements and costs: a successful hacking attack would require tamper proof tape and a device to place numbers on that tape similar to that which is used by the county. The disenfranchisement attack requires only the ability to access the machines the night before the election.

Consequences and potential gain: in a close race, the outcome of the race could be affected. If some of the techniques, eg disenfranchisement, were widely used, the impact could be more significant.

Likelihood of detection: see below.

Countermeasures: It's not clear how to avoid early delivery of voting machines, given the large number of machines that need to be delivered, combined with the need to charge the batteries prior to the election. Stronger security might reduce the risk of pre-election hacking of the software. But the disenfranchisement attack seems very hard to protect against, unless the machines are kept under lock and key until Election Day or there is an alternative method for voting. In my opinion, the obvious response to the disenfranchisement attack would be to provide adequate paper ballots to every polling station to allow the election to proceed in the absence of voting machines. A key countermeasure would be the passage of legislation that would mandate that an election be rerun in the event that tampering has been detected.

Citations and References: NA

Retrospective and Historical Notes: NA

Background.

I served as a polling station inspector in Santa Clara County, California, in the November 2004 election. My polling station was a commons room in a dorm on the Stanford campus. The set of attacks I describe range from small scale (hacking individual machines) to medium scale (disenfranchising voters in selected precincts).

Unfortunately, the general problem of delivering DREs prior to Election Day and storing them securely until Election Day is widespread.

How the machines were delivered and set up.

Santa Clara County delivered five Sequoia paperless DREs to the commons room a week before Election Day. When the woman who made the space available for the election arrived at work that morning, she was horrified to find that the machines already had been delivered. She had asked the county to deliver the machines after she had arrived at work, so that they could be placed in a secure room. Since her request had been ignored, she arranged for the machines to be moved into her office, where she kept them under lock and key until the night before the election. Obviously, the janitor had a key to her office. I don't know who else had a key. Even if her office were completely secure, she or potential co-conspirators would have had plenty of time to access the voting software. (I don't for a minute think that any of this happened. I'm simply pointing out the risks).

We poll workers met at the dorm the evening before the election. We were tasked with organizing the room for the election and with setting up the voting machines in a preliminary state so that the batteries could be fully charged. Because most polling stations do not have a large number of electric outlets, the machines are designed to be daisy chained. In other words, one machine is plugged into an electric outlet, the second is plugged into the first, the third into the second, and so on.

When initially delivered, the machines were "protected" by two levels of tamper proof tape, each piece of which had a unique number. The first level was to be removed the night before the election, when we did the initial set-up. The second level was to be removed on Election Day when we initialized the machines.

Prior to daisy chaining the voting machines, we had to remove the first level of tamper proof tape. The individual pieces of tape were stored in a plastic bag that had been provided by the county. Once the set-up work had been done, we went home. The machines were left unattended in the unlocked commons room.

We returned early the next morning to initialize the machines for Election Day. Prior to the initialization, the second level of tamper proof tape was removed and retained in a plastic bag. All of the removed tamper proof tapes were included in the material that we returned to the county election officials on election night.

Security risks of the procedures deployed by Santa Clara County.

There are multiple security risks, depending on the goal of the attacker. They require differing assumptions about the tamper proof tape and include:

1. Hacking the voting machine software without being detected. This could have been done either by someone who had access to the machines when they were in the commons room, or by someone who had access to the office where they were stored a few hours after delivery. It would be necessary to acquire identical tamper proof tape and a device to mark the tape. However, tamper proof tape is commercially available. It might even be possible for a "mole" working for the county to smuggle out some of the tape.
2. Hacking the voting machine software and risking detection. Since we poll workers had never seen the tamper proof tape and had no idea of what the numbers on the pieces of tape should be, we would not have been able to

determine that someone had hacked the software and replaced the original tapes with different tamper proof tapes. This attack might be detected by election officials if they review the tapes that we returned. Of course if the attacker happened to acquire identical or nearly identical tape and if the attacker used the same number on the counterfeit tapes as had been on the original tapes, it's likely that even diligent election officials would not detect the fraud.

3. Targeting specific precincts in a denial of service attack. This would have been a very easy attack, since the machines were left in a publicly accessible location the night before the election. All that would be required would be for the attacker to remove the second level of tamper proof tape. Poll workers had been instructed to request new voting machines if the tamper proof tapes had been removed. Had we requested new machines, we certainly would not have had the machines up and running by the time the polls were scheduled to open. Indeed, we were barely ready by opening time, even though we had all arrived at the dorm an hour early. I don't know how many machines the county had in reserve, but if there were a widespread attack that removed the tamper proof tape from machines in many voting stations, it is highly likely that the county would have been incapable of replacing the suspect machines.

A related issue is what would happen if hacking or tampering had been detected after the election. As we saw with the butterfly ballots in Florida and in the lost votes in Carteret County, N.C., we do not have adequate legislation for dealing with situations in which election problems are detected after an election. Had tampering or hacking been detected in the presidential race, it is unlikely that the election would have been rerun. The result would have been to raise questions about the validity of reported results and to increase the cynicism of the voting population.

EXAMPLE ATTACK DOCUMENTATION

Touch Screen Calibration

Douglas W. Jones
Sept 25, 2005

Taxonomy: precinct-level
Applicability: touch-screen user interfaces

Method:

Touch-screen input devices are actually entirely separate devices from the display screens that they overlay. As a result, there is no built-in relationship between the coordinates of a spot on the display screen and the coordinates sensed when someone touches directly over that spot. Instead, the software for the touch-screen interface must learn which spots on the touch sensor overlay which spots on the screen. This is called touch-screen calibration. In the case of the transparent plastic touch sensors that are in common use today for both voting machines and personal digital assistants (PDAs or PalmPilots), the calibration drifts slightly, so recalibration can be required on a fairly frequent basis.

Accidental miscalibration of touch-screen voting systems is probably more common than any deliberate efforts, however, it is possible to deliberately miscalibrate a touch-screen voting system to discriminate against certain candidates. In general, candidates who are assigned to voting targets near one edge or corner of the screen are easier to attack this way than those with centrally located targets.

To calibrate the machine normally, you typically go through a ritual where you are asked to touch a target at at least three locations on the screen, frequently two opposite corners and one central spot (on PDAs, this is usually part of the welcome or set-up sequence for new users). Deliberately touching the wrong location during calibration can make it very difficult to touch the voting target for a candidate whose target is on the same side of the screen as that miscalibrated location.

Resource requirements: The perpetrator must control the calibration of the touch screens. Since re-calibration is sometimes required after temperature or humidity changes, or after the machine is subject to vibration or shock, it is always possible to recalibrate voting machines at the precinct. Once miscalibration is discovered, competent precinct-level workers will typically remove the machine from service or recalibrate it. Therefore, this attack can only be effective if it is done with the cooperation of the precinct workers or if the precinct workers are so badly trained that they do not respond to calibration problems.

Potential gain:

Small and difficult to assess because every voter whose vote is changed is extremely likely to notice.

Likelihood of detection:

Each voter who notices that when they try to vote for one candidate, another candidate lights up or nothing happens is likely to complain. Polling place workers who use the touch screen are likely to notice.

There are voter errors that can lead to very similar symptoms (most notably, accidentally resting an idle finger on the touch screen while attempting to vote with a different finger). This can lead polling place workers to blame the voter when the machine is actually miscalibrated, lowering their response time to miscalibrated machines.

Countermeasures:

Preventative measures:

Forcing the pollworkers to use the touch screen is important. If the pollworkers are required to touch the screen with some precision as frequently as the voters vote, will be forced to notice the extent of any miscalibration. In contrast, if the pollworker interface does not involve touching the screen, they will have a far easier time blaming voters for any complaints about calibration (usually phrased "I tried to vote for X and it didn't work").

Voter interfaces with very broad voting targets make the system less sensitive to calibration, for example, where the voter is allowed to touch anywhere on the candidate's name instead of being required to touch a small target.

Physical design that discourages the voter from resting idle fingers on the screen will reduce the likelihood of voter error being confused with calibration problems. Raised ridges around the edge of the screen, for example, can help.

Elimination of the touch screen clearly eliminates this problem, and there are touch-screen technologies that sense the actual shape of the touch instead of sensing the "center of gravity" of the touched area. These latter technologies can sense the physical shape of the display screen itself or the shape of the edge of the opening over the display screen, and they can calibrate themselves against this shape, eliminating the opportunity to miscalibrate the touch sensor.

Detection measures:

Observing the frequency of voter complaint should be a very useful measure, as should observation of the frequency of recalibration.

Citations:

For a discussion of pre-election testing of touch-screen calibration, see http://www.cs.uiowa.edu/~jones/voting/miamitest.pdf (section 11, pages 20-23).

The St. Petersburg Times, Broward Official Apologizes for Voting Mess, Sept 20, 2002, contains a reference to touch-screen calibration problems. There have been many other reports of such problems, but little hard evidence.

Retrospective:

The common assumption that DRE voting systems must use touch-screen technology is unfortunate. The Hart Intercivic dial interface and the push button interfaces used by several of the older DRE systems such as that made by Microvote are clear evidence that there are other possibilities.

EXAMPLE ATTACK DOCUMENTATION

Optical Scan Calibration

Douglas W. Jones
Sept 25, 2005

Taxonomy: Administrative, wholesale
Applicability: optical-scan voting systems (precinct-count and central-count)

Method:

Optical scan voting systems have a mark-sensing threshold. Marks that appear darker than this threshold, to the scanner, will be counted as votes. Marks that appear lighter than this threshold will not be counted as votes. (Some scanners can be configured to detect marks within an intermediate range as questionable.)

The threshold (or thresholds, for those scanners with an intermediate range) is generally variable. It can be set to reject all but very dark marks or it can be set to accept even the faintest of marks. Scanner calibration involves setting the thresholds of the various scanners being used so that they will, as nearly as possible, count ballots in conformance with the applicable law. Ideally, all scanners should be set so that they will apply the same standards, as nearly as possible, and so that these standards are comparable to the standards a person examining the ballot would apply in determining whether or not a mark is a vote. Overly sensitive scanners will sometimes detect overvotes as a result of counting dots, dust specks or printing defects as votes, while overly insensitive scanners will frequently fail to notice legitimate marks in the voting target, leading to undervotes.

Depending on the scanner, setting the threshold can be a matter of physical adjustment, for example, of trimmer potentiometers, or it can be a matter of setting the contents of configuration memory (possibly through a configuration file). In some cases, calibration changes can be made by substitution of different photosensors, for example, replacement of infra-red sensors with visible-light sensors or visa-versa. Scanner calibration is frequently done by vendor's representatives.

Errors in scanner calibration are probably more frequent than any deliberate manipulation of calibration. Manipulation of election results by deliberately improper scanner calibration is possible. For example, if the scanners used in precincts (or counties) that are favored by the perpetrator are calibrated reasonably, while scanners used in precincts that the perpetrator wishes to attack are set unreasonably (overly sensitive or overly insensitive), leading to a higher likelihood that ballots scanned on those machines will be scanned as containing overvotes or undervotes.

Resource requirements: The perpetrator must control the calibration of the scanners. Since calibration is typically done by the vendor's technicians, they will typically be involved.

Potential gain:

The reviews of optical mark-sense ballots cast in Florida in 2000, done by the Miami Herald group, include data showing that widely variable numbers of voters made such errors as marking an X or checkmark in the voting target instead of blacking it in. Reported percentages were as high 1 percent (Washington county) and as low as zero. The average rate, statewide, for circled voting targets, improper marks or use of the wrong type of marking implement came to about 1/2 percent. These figures, based on eyeball examination, should not be taken as better than a rough lower bound on the mismarking rate, since the methodology varied from county to county and did not necessarily involve inspecting all ballots for potential problems.

Nonetheless, it is reasonable to guess that deliberate moderate manipulation of the calibration depending on the precinct or depending on the county could lead to swings of on the order of 1/4 percent. Larger manipulations of the thresholds leading to larger swings in the election output may be feasible.

Likelihood of detection:

In the absence of countermeasures, such small tinkering is very likely to go undetected.

Countermeasures:

Preventative measures:

The standard for pre-election logic and accuracy testing of optical mark-sense scanners involves scanning a stack of perfectly marked ballots. This test does not check the scanner thresholds, but only checks whether the scanner can count accurately. Augmenting this basic test with a test of scanner calibration is not hard. Ideally, the test ballots used for this purpose should be marked using not only the recommended ballot markers (number 2 soft lead pencil and black felt-tipped marker are the two most common), but also with a variety of pens and pencils representative marking implements of the kinds of markers people actually use (at the very least, several makes of black and blue ballpoint pen should be included in these tests). Ideally, the calibration test ballots should include ink and pencil specks (hesitation marks) that should not be counted as well as X and checkmarks that should be counted.

If all ballots that scan as blank or overvoted are kicked back for inspection by the voter (at the precinct) or by the canvassing board (for centrally counted absentee ballots), then this attack will quickly become visible and most of the ballots that would otherwise have been mis-evaluated will either be re-marked or correctly evaluated by people. This measure will be least effective if just one sensor of a multi-sensor scanner is miscalibrated to be underly sensitive, so that only votes read by that sensor are likely to be misread as blank; this makes totally blank ballots unlikely.

Elimination of human involvement in scanner calibration is possible. Self calibrating scanners calibrate themselves by observing the brightness variations on each ballot.

Detection measures:

Hand recounts of randomly selected precincts do not check the scanner calibration with any precision, but they will quickly find scanners that have been calibrated in an unreasonable way. Of course, the probability of detection depends on the fraction of the precincts subject to a hand recount and the fraction of the scanners that are miscalibrated.

In a machine recount, scanning on a different scanner than the one used for the first count will expose differences in scanner calibration, while scanning twice on the same scanner (without recalibration between runs) will expose the uncertainty of the machine count -- such uncertainty can arise if some ballots are marked very close to the detection thresholds.

Citations:

For a tutorial on mark-sense ballot technology, see http://www.cs.uiowa.edu/~jones/voting/optical/

(particularly Figures 8 and 9).

For a discussion of pre-election testing of mark-sense scanner calibration, see http://www.cs.uiowa.edu/~jones/voting/miamitest.pdf (section 8, pages 15 and 16).

Retrospective:

The complete lack of discussion of this issue in the 2002 voting system standards is strange. Yes, it is a matter of human factors, and the 2002 standards did not discuss human factors, but without discussion of this issue, most of the accuracy requirements of the standards as applied to optical mark-sense ballots are trivial and meaningless. What matters is how well the system captures the intent of real voters, not how well it counts perfectly marked test ballots.

The fact that very few jurisdictions properly test scanner calibration is also a serious problem. When I began doing these tests for the state of Iowa in the mid 1990s, we failed one vendor outright when their absentee ballot scanner could not come within ten percent from trial to trial on a stack of 100 ballots marked by real people. When I tried to perform these tests in Miami (cited above), I met with significant resistance.

EXAMPLE ATTACK DOCUMENTATION

Touch Screen Window Manager

Douglas W. Jones
Sept 26, 2005

Taxonomy: wholesale third-party firmware
Applicability: touch-screen voting systems using window managers

Method:

Many DRE voting systems use a window manager, frequently from Microsoft, but some open voting products will use the X window manager. On such systems, all display of text on the screen and interpretation of touches on the screen are generally done through window-manager routines. In many cases, the window manager is considered to be an industry-standard commercial off-the-shelf component, and is therefore subject to reduced scrutiny.

If the perpetrator can add code to the window manager, the behavior of the voting system can be modified in a way that alters the election outcome. For example, consider this attack that will favor candidates from the aaa party in states allowing straight party voting where the bbb party is the other major party and the ccc party is a strong third party:

Insert in the window manager code to detect that the current window includes the text "straight party", and that it includes the text "aaa", "bbb" and "ccc" in the same window. The window manager is programmed to misbehave whenever this combination is present in the window, but only on the first Tuesday after the first Monday of November, only when this window has been used at least 20 times, and only when the machine has been turned on for over 4 hours. The misbehavior is to misreport all touches in the vicinity of the text "ccc" as being in the vicinity of "aaa", thus stealing straight-party votes from the third party and giving them to the major party.

The code for this attack should of course be obfuscated, with misleading comments and carefully hidden function so that it evades the internal quality control checks of the software vendor. The art of obfuscated programming has been thoroughly explored.

There are, of course, many variations on this attack, some of which do not depend on the straight party option. For example, the attack can be limited to an office or it can apply broadly, throwing, say, 10% of the third party vote to the favored party in all races.

Resource requirements: The perpetrator must have access to the source code of the window manager.

Potential gain:

The target should be around 1/3 of the straight party votes for a major third party. In the past 50 years, third parties have rarely earned over 5% of the vote, but sometimes up to 15% (George Wallace in 1968). The fraction of straight-party voters is hard to determine, but it will be significant only for parties that put up candidates for many different offices. In recent years, only the Greens and the Libertarians have managed this, so these are the natural third parties to attack. As a naive guess, it is unlikely that this attack would win more than 1% of the vote.

Likelihood of detection:

Because the attack code is embedded in third-party off-the-shelf software, it is unlikely to be subject to the same scrutiny as purpose-written voting code. Because it only manifests itself under conditions typical of real elections, it is unlikely to be seen in any testing by the commercial off-the-shelf vendor. The checks sensitive to the number of votes cast and the length of time the machine has been running will evade many pre-election tests and possibly even many ITA tests. Small additions to the conditions suggested above can make it evade ITA testing.

If a voter does notice that their vote was cast for the wrong candidate (and there are variations of this attack that evade detection by the voter) the problem can easily be blamed on the voter (you simply touched the wrong point on the screen) or on touch screen alignment.

Because the attack code is modest, stealing only a small fraction of the votes cast, it is unlikely to show up in post election audits.

Countermeasures:

Preventative measures:

Eliminate testing and source code inspection exemptions for inspection of third-party commercial off-the-shelf software.

Eliminate testing and source code inspection exemptions for emergency patches and bug fixes.

Eliminate dependencies on window-manager functionality from the voting application. Typically, this will involve "flattening" the code to eliminate deep hierarchies of reusable software components. Instead, the voting application should directly manipulate the display screen.

Eliminate text from the voting application. Instead, display all ballot content on the screen as images, with extremely dumb image display software used to place all voting-related text on the screen. It would be helpful if there were a guarantee that the system contained no OCR software that could examine images to detect embedded text (such software is becoming increasingly widely available as a software component and may soon become a standard off-the-shelf component for other software systems).

Eliminate access to the real-time clock, or alternatively, strictly audit all use of the real-time clock so that no use of the date, the time of day or the time since power-up is permitted except for the purpose of logging events in the system event log.

Detection measures:

Take voter complaints of the form "I voted straight party ccc and it marked the aaa candidate" very seriously. Unfortunately, variations on this attack may be invisible to the voter.

Perform parallel testing on election day, with a test environment that the machine cannot possibly distinguish from real use. The machine should be turned on at and off at reasonable times for polling places to be opened and closed, the number of votes should be typical of a busy polling place,

Citations:

The Fidlar and Chambers EV 2000 was accidentally "attacked" by Microsoft following a distant relative of this scenario in January 1998. The "attack" was a cosmetic change that involved no change to the Windows applications programmer interface (API) and was therefore determined exempt from testing by the ITA. Unfortunately, this cosmetic change ended up revealing, to each voter, all votes cast by the previous voter to use that machine. I described this to the House Science Committee on May 22, 2001. See http://www.cs.uiowa.edu/~jones/voting/congress.html

That accidental attack led me to propose this attack in *E-Voting -- Prospects and Problems*, April 13, 2000. Available on-line at http://www.cs.uiowa.edu/~jones/voting/taubate.html

Retrospective:

The problem posed by emergency security patches from vendors is extremely serious. These come with a built-in urgency that is immense. We are training a generation of computer system administrators to install such patches immediately and without question. It is not clear that this is prudent except when we know, with a great degree of certainty, that the vendors software development procedures conform to the same standards as our application.

Appendix C

[of Stanley A. Klein, Position Paper on Voting System Threat Modeling, September 24, 2005]

Exploitation of Compromising Electromagnetic Emanations

Taxonomy

Retail, vote buying, or voter intimidation.

Applicability

DRE voting machines. Possible use against precinct-based optical scan tabulators.

Method

Perpetrator uses compromising electromagnetic emanations from voting machines to reproduce DRE screens in a vehicle near the polling place. Bought or intimidated voters are instructed to make certain combinations of selections and changes to enable the perpetrator to identify which voter is using which machine. Perpetrator watches the machine activity and ensures that voters vote as instructed.   This attack effectively returns voting activity to the conditions that existed prior to adoption in the late 1800's of the Australian Secret Ballot.

Exploitation of emanations from an optical scan tabulator would require either (a) the voter being instructed to vote in particular ways for offices/issues not of interest to the perpetrator, or (b) administrative records accessible to the perpetrator or an accomplice inside the polling place who can provide information on the sequence of voters whose ballots are being processed.

Resource Requirements

This attack requires development of software to monitor and process the compromising electromagnetic emanations. This development has economy of scope; once developed, the hardware and software can be reused in numerous elections. The cost of developing and producing the relevant equipment is likely to be in a multi-million-dollar range, but over time the relevant technology is likely to become ubiquitous.

The relevant technology may already exist and be in use within the intelligence community. The feasibility of exploiting compromising electromagnetic emanations from electronic equipment has been rumored since the 1970's. The Defense Department has long had a program called “Tempest” for minimizing compromising electromagnetic emanations from electronic equipment. Redacted Tempest documents were posted on the Internet a few years ago as a result of a FOIA request.

The technology requirements for accomplishing the attack are likely to include the following:

- High capacity software defined radio
- Digital signal processing and/or directive antenna technology (such as phased arrays) sufficient to separate individual voting machine emanations. For example, this might be done by using small differences in clock speeds or other processing hardware characteristics of the various machines.
- Digital signal processing to reconstruct the internal processing and screen displays from the voting machine emanations.

The software defined radio and high capacity digital signal processing technologies are currently available, although not necessarily at low cost and sufficiently small size to allow installation of the necessary facilities in a vehicle. These technologies at appropriate capacities, sizes, and costs are likely to become ubiquitous during the lifetime of voting machines in current service or currently being designed and purchased.

Perpetrators must also have access to a pool of subvertable voters willing to vote in return for payment or unable to complain if threatened. Employees, tenants, and those with similar dependency relationships are particularly vulnerable.

Potential Gain

One vote per subverted voter.

Likelihood of Detection

The likelihood of detection depends on the degree of dependency linking the perpetrator to the subverted voters.

Countermeasures

Preventive Measures

Apply to voting machines and polling places the Tempest technology and other measures used by the Defense Department for protecting against exploitation of compromising electromagnetic emanations.

Use only optical scan machines, and take measures to block the collection of information that could identify the sequence of voters whose ballots are being scanned.

Detection Measures

The attack can not be detected by technical or administrative means. The only possibility of discovering that it has occurred is if one of the voters reveals the existence of the vote buying or voter intimidation to authorities who are not themselves involved in the scheme.

Citations

None

Retrospective

None.

Appendix B

[of Stanley A. Klein, Position Paper on Voting System Threat Modeling,

September 24, 2005]

Smartcard Port Attack

Taxonomy

Retail if performed by a voter or polling place official in the polling place. Wholesale if performed by an insider during or subsequent to machine setup.

Applicability

DRE voting machines using smartcards for voter authorization and other functions.

Method

By creating an appropriate interface, an attack on a voting machine can be based on software resident on another device. Modern cell phones and personal digital assistant (PDA) devices contain computers suitable for such an attack. An example of this kind of attack would be to penetrate the voting machine electronically through a smartcard reader port, often used in DRE machines for voter authorization. The device interface software that would be the focus of this attack is likely exempt from inspection under the provisions of VVSG Volume 1 Section 1.6 because of status as unmodified "Commercial Off-The-Shelf" software. Plans for an electronic device that connects a computer to a smart card reader port can be downloaded from the Internet (at http://www.electronics-lab.com/projects/misc/003/). An attack can be pre-programmed by experts, making it necessary for the attacker only to place a device into the smart card reader and remove it. The relevant electronics can be made easy to hide in clothing and the connection to the device in the smartcard port can be made by thin cable or optical wireless, making it very difficult for polling place officials to see that the attack is taking place. The attack could be perpetrated for various malicious purposes either in the polling place or during pre-election setup.

The external computer subverts an exploitable smart card driver and gains access to the voting machine memory bus. Programs on the external computer are then run to accomplish the purposes of the attack. For the retail polling place attack, this would be to “edit” previously cast ballots. Examples of wholesale (post-setup attack) purposes could be to maliciously modify the voting machine setups or to load self-deleting malicious software onto the machines.

Resource Requirements

This attack requires development of the smartcard emulation hardware, the interface to the external computer, and the attack software resident on the external computer. This development has economy of scope; once developed, the hardware and software can be reused in numerous elections. The cost of developing and producing the relevant equipment can probably be performed by someone with electronics expertise for an amount ranging from under $100 to as much as $1 Million depending on the sophistication of the interface (e.g. ease of concealment) and number of devices produced.

Also required are perpetrators to execute the attacks. For retail attack, these can probably be recruited and trained at low cost. An insider executing an attack at setup time would probably have to be bribed or otherwise induced to perform the attack.

Potential Gain

For the retail attack, all the votes on each attacked machine can be modified. For the wholesale attack, all machines in a jurisdiction set up at the same facility could be loaded with malicious software.

Likelihood of Detection

Depending on the sophistication of the design and the training of the perpetrators executing the attack, this attack could be extremely difficult to detect.

Countermeasures

Preventive Measures

1. Eliminate use of smartcards.
2. Provide means to disrupt any connection between the smartcard emulator and the external computer. (This can create an escalating “arms race” of increased sophistication in prevention and attack technology. For example, in the 1990's

European telephones contained cable cutters to prevent a similar kind of attack. Attackers countered by using thinner cables.)

3. Ensure that the voting machine operating system and the smartcard driver are not exploitable. This will require removing any “COTS Exemption” from all relevant software and conducting penetration tests of attacks through the smartcard port.

Detection Measures

None, if attack has sophisticated design.

Citations

Smartcard emulation attacks on telephone systems were described in an article appearing in 2600 Magazine in 1996 or 1997.

Retrospective

None.

Misprogramming Threat

Jeremy Epstein

Sep 29, 2005

Taxonomy

Wholesale.

Applicability

Voting phases: Any type of system is vulnerable, but paper-less systems, such as DREs and lever systems are particularly vulnerable.

Method

When programming the voting machines, an insider could accidentally or intentionally misprogram the machines to count votes incorrectly. Examples of such misprogramming include:

- Counting votes for the wrong candidate/position. This has happened in several recent elections, including one where "yes" and "no" votes on a ballot initiative were reversed in some jurisdictions. [I believe this was on DREs in California, but have not located the details.]
- Counting party line votes incorrectly. In 2004, election results in North Carolina's 11th House results were reversed when it was discovered that party line votes were not initially counted. In that case, due to use of optical scan ballots, it was possible to recount and correct the results.
- In a demonstration (not a real election) by a vendor, votes cast in the Spanish ballot were not counted, but votes cast in English were counted correctly.
- In the June 2005 Republican Primary in Virginia, the home precinct of one of the candidates showed zero votes for the candidate. The machine in use was a lever system with no contemporaneous paper trail. No cause was established, but it is assumed to be misprogramming.

All of the above cases appear to be due to accidental misprogramming, and not due to any deliberate effort to change election results.

This case is not addressing problems of miscounting due to touchscreen alignment; it is focused exclusively on incorrect counting.

Detection is difficult if the modifications made in vote totals are relatively small, but a 5% change in vote totals could easily be made without detection.

Voting systems that provide a paper backup (e.g,. optical scan or DRE with VVPAT) can be recounted; a hand recount would detect any tampering.

Citations

None.

Retrospective

This is a variation on historical problems with miscounting ballots. The difference is the scope of miscounting – an accidental or deliberate error in the counting logic can impact a larger number of voters than a simple miscounting of paper ballots.

Resource Requirements

The perpetrator must have the ability to program voting systems. For the purposes of this threat paper, I assume that the perpetrator is an authorized insider. Methods used by an outsider to gain access for reprogramming is a separate threat.

Depending on the policies of the jurisdiction, misprogramming could impact a single precinct, a city or county, or a state.

Potential Gain

Ability to modify vote totals. The smaller the election (i.e., more local), the greater the likelihood of being able to change the election results, whether accidentally or intentionally.

Likelihood of Detection

As long as the vote totals are not too far outside the expected range, the likelihood of detection may be high if logic & accuracy tests are thorough, or low if they are not. The fact that the above listed counting flaws occurred (despite L&A testing) indicates that the L&A tests are insufficient for this purpose.

If the misprogramming is deliberate (vice accidental), the perpetrator can take steps to make miscounting arbitrarily difficult to locate through L&A tests.

Countermeasures

Preventative Measures

Review of the voting machine programming will make it harder to hide misprogramming code. However, review is only moderately effective even when flaws are accidental, and is reasonably ineffective against deliberately hidden flaws.

Policies and procedures to ensure that no single person can program a voting machine can ameliorate the risk. In order for this to be an effective countermeasure, both people (or at least two, if more than two are present) must be capable of understanding the programming process and detecting failures. A second person who watches but does not understand is not a countermeasure.

Staff authorized to program voting machines can be vetted to reduce the risk of their deliberately misprogramming voting machines. This will not address accidental misprogramming.

Detection Measures

Wi-Fi Usage in Voting (without inside assistance)

Jeremy Epstein

Sep 29, 2005

Taxonomy

Wholesale.

Applicability

Voting phases: Any system using Wi-Fi (typically but not exclusively DREs), whether or not the Wi-Fi is intentionally used. There is at least one model of DRE that uses Wi-Fi (the AVS WinVote); there may be others. Additionally, since many DREs are based on off-the-shelf laptop computers which have built-in Wi-Fi[1], there may be products that have Wi-Fi capabilities that are not advertised, and perhaps not even known by the DRE vendor.

Method

Many voting machines use Wi-Fi (wireless networking, typically following the IEEE 802.11b or 802.11g standards) for communication among machines in a precinct. In some cases, they are used for opening and closing the polls, while in other cases they might be used during the day.

This example assumes that the perpetrator has no ability to affect the software in the voting machine prior to election day (a “life cycle” attack), but rather is working strictly as an outsider. Many of the issues are the same as in a life cycle attack.

The initial goal of the attacker is to get access to the machine via the Wi-Fi connection, followed by any of the other typical types of attack (e.g., to modify the vote totals, modify the programming, or make the machine fail). For example, the programming could be modified to add every fourth vote for Jane Jones to the total for Sam Smith instead, while displaying the correct values on the screen.[2]

In some cases, vendors assert that the Wi-Fi capability is turned off at all times, or except during poll opening and closing. In that case, an additional attack method may require

[1] Nearly any laptop using the popular Pentium M chipset will have Wi-Fi.

[2] Such actions have happened by accident in voting system demonstrations. Whether they have happened in real elections is unproven, but is a matter of debate.

determining if the Wi-Fi hardware has remote "wake-up" capabilities, which allow enabling the device by sending a particular unpublished message.[3]

Resource Requirements

The perpetrator must have the ability to send Wi-Fi signals to the voting systems, which must have hardware to receive those signals. Further, the software in the voting system must have one or more vulnerabilities that allow using (or abusing) the Wi-Fi communications.

Potential Gain

- Ability to shut down as many precincts as can be visited on election day by the perpetrator and his/her co-conspirators (known as a "denial of service" attack).
- If vulnerabilities exist in the Wi-Fi capability, ability to make arbitrary modifications to the voting totals at as many precincts as can be visited on election day by the perpetrator and his/ her co-conspirators.

Likelihood of Detection

The likelihood of detection is very low, as the attacker need not be inside the polling place to launch attacks. A Pringles® potato chip can is a highly effective receiver for Wi-Fi traffic[4], allowing access from a substantial distance (e.g., a car driving within several hundred yards of the precinct). Further, it would only take a few seconds to modify the programming if the Wi-Fi implementation is vulnerable to attack, thus allowing the attacker to perform the reprogramming without even parking his/her car.

The difficulty is not access to the Wi-Fi signal, but rather the question of whether the Wi-Fi device is enabled (or can be remotely enabled) and whether the software using the Wi-Fi device has vulnerabilities. Assuming that the vulnerabilities exist, the chance of detection is very low.

Encrypting the Wi-Fi traffic (the most commonly described protection for Wi-Fi) is not a countermeasure to this type of attack.

Countermeasures

Preventative Measures

[3] Some network cards for wired networks have this capability. Whether Wi-Fi hardware has a similar capability is a supposition on the author's part.

[4] A report in 2001 gave the cost of building a Pringles® antenna at under $7 each, or less if built in bulk. See http://www.oreillynet.com/cs/weblog/view/wlg/448 for details.

Source code review may be able to find flaws that allow inappropriate use of Wi-Fi hardware. However, source code review is only moderately effective even when security flaws are accidental. Additionally, even the voting system vendors do not have the source code for much of their systems (e.g., the operating systems and device drivers which are a potential weak spot for Wi-Fi implementations).

Requiring hardware for voting machines that does not have any Wi-Fi features completely prevents this type of attack. As Wi-Fi is increasingly built into laptop computers (the basis for most DREs), this is increasingly infeasible.

Having all voting machines inside a Faraday cage, such as is used for processing classified information (where it is known as a SCIF). This would require that the attacker be inside the same facility, making a remote attack impossible. Equipping every precinct as a Faraday cage is impractical, and putting each voting machine inside a Faraday cage is equivalent to disabling the Wi-Fi, thus eliminating any benefit it might have.

Detection Measures

Detection is difficult if the modifications made in vote totals are relatively small, but a 5% change in vote totals could easily be made without detection.

Voting systems that provide a paper backup (e.g,. optical scan or DRE with VVPAT) can be recounted; a hand recount would detect any tampering.

If the attacker adds ballots rather than modifying those that have already been voted (or are yet to be voted), then a reconciliation of the number of votes vs. the number of voters will detect the attack.

Citations

None.

Retrospective

This is a variation on stuffing the ballot box. It does not require physical access to the voting machine, and operates by replacing ballots rather than adding new ones.

Wi-Fi Usage in Voting (with vendor complicity)

Jeremy Epstein

Sep 29 2005

Taxonomy

Wholesale at the precinct level.

Applicability

Voting phases: Any system using Wi-Fi (typically but not exclusively DREs), whether or not the Wi-Fi is intentionally used. There is at least one model of DRE that uses Wi-Fi (the AVS WinVote); there may be others. Additionally, since many DREs are based on off-the-shelf laptop computers which have built-in Wi-Fi[1], there may be products that have Wi-Fi capabilities that are not advertised, and perhaps not even known by the DRE vendor.

Method

Many voting machines use Wi-Fi (wireless networking, typically following the IEEE 802.11b or 802.11g standards) for communication among machines in a precinct. In some cases, they are used for opening and closing the polls, while in other cases they might be used during the day.

This example assumes that the perpetrator has the ability to modify the software used in the voting machine, either by being part of the development effort at the vendor, or by modifying the software during programming in the local jurisdiction. Other Wi-Fi examples submitted separately do not assume the ability to modify software.

The perpetrator causes the voting machine software to be enabled at an opportune time, and to accept commands once a "secret" enablement command has been provided[2]. This can be hidden from detection (see *Likelihood of Detection*, below). Once the Wi-Fi link is enabled, the attacker can retrieve vote totals and/or ballot programming, modify the settings, and download new totals and/or programming. For example, the programming

[1] Nearly any laptop using the popular Pentium M chipset will have Wi-Fi.

[2] The concept of using a secret enablement command is widely used by attackers on the internet, not specifically for voting machines, but for other forms of attacks involving "back doors".

could be modified to add every fourth vote for Jane Jones to the total for Sam Smith instead, while displaying the correct values on the screen.[3]

Another related alternative which could be used by the perpetrator is to cause the Wi-Fi communication to use a weak or predetermined encryption key. This is effectively impossible to detect without a careful cryptographic analysis, which is well beyond the scope of voting machine testing.

Resource Requirements

There are two roles who must be complicit in this example: the insider who introduces the flaw, and the person who exploit it on election day. These could be the same person or different people.

For the first role, the perpetrator must have the ability to modify the software used in the voting system, either as a member of the vendor's development team or during the local programming.

Stealing a local election would be fairly easy this way, since a person in the second role can go from precinct to precinct making the appropriate "zaps" to voting machines. On a broader base (e.g., a statewide election) would require more people to divide up the work, since it can only be done as fast as each machine can be remotely accessed.

Potential Gain

- Ability to shut down as many precincts as can be visited on election day by the perpetrator and his/her co-conspirators (known as a "denial of service" attack).
- Ability to make arbitrary modifications to the voting totals at as many precincts as can be visited on election day by the perpetrator and his/ her co-conspirators.
- Ability to make arbitrary modifications to the ballot setup at as many precincts as can be visited on election day by the perpetrator and his/ her co-conspirators.

Likelihood of Detection

The likelihood of detection can be made arbitrarily small. For example, software could enable the Wi-Fi device for a few seconds every ten minutes while the polls are open; if an enablement command is received during that window, the device is left enabled, and otherwise disabled. This would be almost impossible to detect as part of Logic & Accuracy tests, since continuous scanning for an open Wi-Fi link is unlikely. Even if the brief on period is detected during testing, without knowing the enablement command to keep the connection open permanently, it would likely be dismissed as a testing error.

[3] Such actions have happened by accident in voting system demonstrations. Whether they have happened in real elections is unproven, but is a matter of debate.

An attacker need not be inside the polling place to launch attacks. A Pringles® potato chip can is a highly effective receiver for Wi-Fi traffic[4], allowing access from a substantial distance (e.g., a car driving within several hundred yards of the precinct). Further, it would only take a few seconds to modify the programming once the Wi-Fi link is enabled, thus allowing the attacker to perform the reprogramming without even parking his/her car.

Encrypting the Wi-Fi traffic (the most commonly described protection for Wi-Fi) is not a countermeasure to this type of attack.

Countermeasures

Preventative Measures

Source code review will make it harder to hide code to enable the Wi-Fi enabling. However, source code review is only moderately effective even when security flaws are accidental, and is reasonably ineffective against deliberately hidden flaws.

Requiring hardware for voting machines that does not have any Wi-Fi features completely prevents this type of attack. As Wi-Fi is increasingly built into laptop computers (the basis for most DREs), this is increasingly infeasible.

Having all voting machines inside a Faraday cage, such as is used for processing classified information (where it is known as a SCIF). This would require that the attacker be inside the same facility, making a remote attack impossible (but local attacks would still be undetectable). Equipping every precinct as a Faraday cage is impractical, and putting each voting machine inside a Faraday cage is equivalent to disabling the Wi-Fi, thus eliminating any benefit it might have.

Detection Measures

Detection is difficult if the modifications made in vote totals are relatively small, but a 5% change in vote totals could easily be made without detection.

Voting systems that provide a paper backup (e.g,. optical scan or DRE with VVPAT) can be recounted; a hand recount would detect any tampering.

If the attacker adds ballots rather than modifying those that have already been voted (or are yet to be voted), then a reconciliation of the number of votes vs. the number of voters will detect the attack.

Citations

[4] A report in 2001 gave the cost of building a Pringles® antenna at under $7 each, or less if built in bulk. See http://www.oreillynet.com/cs/weblog/view/wlg/448 for details.

None.

Retrospective

This is a variation on stuffing the ballot box. It does not require physical access to the voting machine, and operates by replacing ballots rather than adding new ones.

I agree with Mr. Epstein concerning the threat WiFi equipped systems pose to electronic voting systems. I am somewhat concerned that this discussion is focused on WiFi and not Optical and RF in general. Any method of communicating with the voting system including optical or other RF communications systems presents an equivalent threat.

A number of existing voting systems are equipped with IrDA (Infrared Data Association) optical wireless ports. These systems support communications with the DRE voting systems at data rates up to 115 Kb/s.

In Diebold System's AccuVote TS systems these ports are supported using Microsoft's Windows CE with Winsock. This makes the application interface easy to program to, and all required drivers are already installed in the OS.

It is interesting that the VVSG currently under development, while mentioning this technology does nothing to restrict or prevent its use, not even on Election Day.

It is understandable that communications technology be used for pre election preparation, but is totally irresponsible and inexcusable to allow it to be used during an election. The presence of this technology makes it possible to upload to the voting system anything that is desired after the final "Logic and Accuracy" test have been performed, thus totally compromising the system. Even the ability to transmit as much as a single frame (even an error frame) of data could be sufficient to alter the approved behavior of the system.

I submitted a short paper discussing this issue to the TGDC entitled **Comment on Wireless Requirements,** **at http://vote.nist.gov/ECPosStat.htm. I believe this to be a highly likely line of attack on voting systems unless the technical community is vocal in exposing this threat before it becomes accepted practice to install optical ports in voting systems. I hope that you will use your influence to draw attention to these flaws as well as to the threat of WiFi.**

Thank You,
James C. Johnson

Voter "assistance"

Douglas W. Jones
Aug 26, 2005

Taxonomy: Retail, vote buying or voter intimidation
Applicability: All voting technologies

Method:

The perpetrator offers to "assist" a voter in casting a ballot. In fact, this assistance consists of either marking or casting the ballot for the voter or looking over the voter's shoulder to check that the voter is voting as instructed by the perpetrator.

Resource requirements: Each perpetrator must have access to a pool of subvertable voters willing to request "assistance" in return for payment or unable to complain if threatened. Employees, tenants, and those with similar dependency relationships are particularly vulnerable.

Potential gain:

One vote per subverted voter.

Likelihood of detection:

An election observer can easily note the frequency with which voters request assistance. Observations of inappropriate assistance are common, but prosecution is rare because voters have a legitimate right to request assistance and it is difficult to prove that the assistant acted improperly under the legal framework present in many states.

Improper assistance in the casting of postal absentee ballots is very unlikely to be detected. This applies to all "vote at home" schemes.

Countermeasures:

Preventative measures:

Restrict the right to assistance to those with a demonstrable need. This can be demeaning to the voter, since it requires the voter to prove that they have a disability or to prove that they do not understand the workings of the voting system.

Restrict who may assist a voter. Deny the voter the right to assistance from anyone but a close relative, guardian or pollworker, and require that if pollworkers offer assistance, they must do so in pairs representing opposing parties.

Develop voting systems requiring less assistance. Audio voting assistance devices, audio DRE machines, and tactile ballots can all reduce the need for assistance among illiterate or blind voters. It is impossible, however, to completely eliminate the need for assistance.

Restrict the right to postal absentee ballots or other "vote at home" systems. This is problematic, although if satellite polling places are provided for early voting, the need for postal absentee ballots decreases and with it, the number of votes that could be corrupted in this way decreases.

Detection measures:

Require documentation of every instance in which a voter requests the presence of an assistant in the voting booth. Routine audits of the frequency of assistance can lead to an understanding of what is normal, allowing the detection of unusual patterns of assistance.

Election observers should note the frequency of requests for assistance, and should make particular note of suspicious requests, for example, where the same person (not a pollworker) offers assistance to multiple voters, or where voters request assistance even though there is evidence that they have no need for assistance (as in the famous case of the voter who was reading a newspaper while waiting in line to vote, but who then requested assistance).

Citations:

Joseph P. Harris, *Election Administration in the United States*, The Brookings Institution, 1934. Improper assistance is discussed on pages 48 and 373. The legitimate need for assistance is discussed on page 184. Page 373 includes the qualitative judgement that assistance was, at the time, the dominant form of vote fraud.

Edmund F. Kallina, Jr. *Courthouse over White House*, University Presses of Florida, 1988. Assistance problems in Chicago in 1960 are described on pages 87-89; some of the incidents described are legitimate, while others are clearly coercive. Note, however, the philosophical issue raised on page 89 and the contrary opinion on page 91. The special need for legitimate assistance caused by the technological transition from paper ballots to voting machines are discussed on pages 82-85, with specific attention to the risks this poses.

Retrospective:

Some states tightened up their voter assistance laws long ago in response to the recommendations Harris made in 1934. Other states are still wide open to this scheme.

VVPR Attack with Misprinted VVPAT

David L. Dill

October 2, 2003

Taxonomy

Blank

Applicability

DRE voting terminals with voter-verifiable printers.

Method

> Malicious software misrecords voter intent consistently in its electronic records and on voter-verified printout.
>
> Software has sophisticated "cues" to detect whether it is being tested before the election, or tested in parallel with the actual election.
>
> This method relies on lack of voter diligence in checking the printout. The extent to which voters will be diligent is hotly disputed, but it is reasonable to assume that many will not check carefully.
>
> The software would attempt to minimize detection by voters by several methods (1) Steal only a small percentage of the votes; (2) steal votes for down-ballot races; (3) implement extensive "verification" on non-paper display, to make paper check seem redundant; (4) make the paper ballot inconvenient to verify.
>
> Minimize the ability of voters who detect errors to prove them. E.g., do not keep votes on display, or change displayed votes while they are being printed. Those (supposedly) few voters who notice a changed vote may have difficulty persuading poll workers that it happened. (Witness widespread reports of voting machines displaying wrong votes in 2004, with no investigation.)

Resource Requirements

> At least one individual with the necessary access to modify DRE software during development.

Complicity with other people designing the user interface and printer would make the attack more effective.

Potential Gain

Up to a 1% vote shift in an election jurisdiction. 1% is a rate that gives about 1 misprint per machine. With 5 machines per polling place and 20% of voters checking carefully, this would lead to an average one complaint per polling place, which could perhaps be dismissed as "voter error".

Likelihood of Detection

Medium

It is hard for me to quantify the risk if this is done on a nationwide scale. I believe that it is substantial, because consistent pattern of complaints will lead to widespread public suspicion, which might prompt a sufficiently serious investigation to catch a fraud of this nature, especially if the problems occur in repeated elections.

Countermeasures

Preventative Measures

Background checks on vendor employees
The goal is to reduce the probability that employees with past criminal histories, gambling and drug problems, etc. have access to software.

Cryptographic hashing of software, including COTS
The goal of this countermeasure is to make it difficult for outsiders to modify election software.

Detection Measures

Object Code Validation
This increases the skill required to insert an undetected Trojan for the first part of the attack (but not much!)

VVPT Paper has digital signature on it
If the digital signature contains an trustworthy time-stamp, this could make creating bogus VVPAT much more difficult, even with access to voting equipment. Trustworthy time-stamp technology is not used in current DREs, which now allow resetting of the

date/time by anyone with a password (or possibly even without a password in some models).

Realistic L&A (realistic numbers of votes cast, patterns of votes, in election mode).

This countermeasure detect incompetently designed Trojans, but is otherwise ineffective.

Parallel testing

Parallel testing might be more effective when there is a VVPAT. It is easier for a machine to decide whether to cheat safely if it can observe input for the entire election, then change votes. With VVPAT, it is difficult and expensive to change votes after the records are printed, so the decision to cheat would probably have to be made while there are still records to be printed. However, since only a small number of records need to be changed, machines could start cheating only after they have seen most of the votes.

Attack Economics

Cost is bribe price of a software developer.

Variations on attack theme

Variations on software corruption: Trojan inserted by someone other than a developer, election officials tricked into installing bogus software, bogus software intentionally installed by election office.

Conclusions

The most effective countermeasures are anti-counterfeiting, anti-tampering measures with paper records, plus physical security of special paper, physical security of paper records with votes, and prompt random auditing.

Citations

Ted Selker's unpublished(?) paper on voter detection of VVPT errors.

Retrospective

None

Trojan Horse in DRE
Application Software

Method

Malicious code hidden in DRE polling station that enables:

1. Someone to access special features that can siphon votes from one candidate (or option for questions) to another after ballot definitions have been determined.
2. Votes to be siphoned between candidates based on predetermined criteria such as moving votes between candidates associated with political parties. This would require the DRE software to read the ballot definition files. Votes could also be siphoned between candidates based on non-party-based attributes such as percentage of the vote received.
3. The attack could also result in disrupting an election, perhaps via a denial or service type method.

Access to each DRE, the host server(s) or machine(s) on which the master copy of the source code, or compiled binary image(s) of the application software are created and/or stored, or any intermediary system(s) that might be responsible for installing software onto the DRE's.

Taxonomy

Wholesale, Configuration-related, Change Management, programming, software.

Applicability

DRE, DRE with VVPT

Resource Requirements

For any of the three attack methods above, the perpetrator must be a skilled programmer and have access to the source code. This could occur at either the vendor (or a sub-contractor) site, or a test lab (assuming the vendor has provided the source code to the VSTL). It could happen within an elections office, but only in the case where the vendor made source code and installation procedures available at the local facility, which is not a vendor's typical method of operation. The perpetrator must be skilled at understanding how votes are actually created and tabulated, the methods used for internal auditing and crosschecks, and how the code is tested.

For a Type 1 attack, there must also be a perpetrator with access to the DREs at the states and counties. The perpetrator does not even need access to all counties, or even all precincts, but can could be effective by targeting DREs to be placed within critical demographic regions. The access can be either authorized (an election official, e.g., whose job it is to input ballot definitions, test DREs, or otherwise touch a large number of machines) or unauthorized (e.g., via a break-in to a storage warehouse).

Potential Gain

The potential gain is large. Since the attack impacts many separate DREs, a small number of votes can be stolen per DRE adding up to enough votes to change the results for a race.

Likelihood of Detection

The malicious code could be detected in several places: by the vendor, by the test lab, or by an election official noticing anomalous results during a test or in a real election. A skilled programmer will generally be able to hide a significant amount of dangerous code without being detected in testing. (See countermeasures.) Detection would depend on the individual skills and depth of the source code review (either at the vendor or the test lab), or the amount of attention being paid to each DRE's behavior during testing or in a real election.

Countermeasures

Source code review: User interface code, for example, tends to be extremely complicated calling multiple libraries. Source inspections and reviews that might catch this type of code typically cost over $500,000 and take over 6 months. In addition, any change to the source code must result in a similarly expensive re-review.

> Open-ended testing: This testing also is very expensive and requires significant security analysis expertise.
>
> Testing that fully simulates Election Day activities
>
> Independent Dual Verification with audit
>
> Parallel Testing

Citations

Ken Thompson, Turing Award Speech, 1984: http://cm.bell-labs.com/who/ken/trust.html

The moral is obvious. You can't trust code that you did not totally create yourself. (Especially code from companies that employ people like me.) No amount of source-level verification or scrutiny will protect you from using untrusted code. In demonstrating the possibility of this kind of attack, I picked on the C compiler. I could have picked on any program-handling program such as an assembler, a loader, or even hardware microcode. As the level of program gets lower, these bugs will be harder and harder to detect. A well installed microcode bug will be almost impossible to detect.

Retrospective

While there is no evidence that wholesale vote fraud has occurred using DREs, the issue is whether this is possible in the future. Given the large payoff possible, the relatively low likelihood of detection if a very skilled programmer in involved, the large number of very skilled programmers available, and the small number of perpetrators necessary, this threat is a serious threat to consider for the future.

See: Harri Hursti work on this here. Thanks to Black Box Voting for permission to use.

Replaceable Media on Optical Scan

Harri Hursti with Eric Lazarus

Taxonomy

Modification of basic functionality by replacing unprotected executable on replaceable media

Method

Generally, memory cards are thought of as containing data, including primarily the ballot definition data files (files that allow the OpScan to read the ballots) and, secondarily, the vote totals.

However, can memorycards also contain executable program which is started by firmware. Due programming programming langages cababilities, modified programs can falsify reports produced, hide pre-set counters, etc. Due capabilities of the interpreter are not know, what are the extends to use this exploit for trojan horses and other software and not be fully understood.

At least one major vendor has replaceable media (specifically, its memory cards) carrying software. This easily modified software is responsible for printing out the vote totals. It prints the "zero" tally report at the start of polling, and vote totals after the polls close. However, it is not template or macro script, instead it is modified BASIC language variant, making pool of programmers able to write these programs very large

> *The perpetrator must (a) acquire access to the PCOS memory cards, or (b) be able to change files on the central tabulator before election definitions are loaded into memory cards or (c) connect the PCOS machine to telephone line for remote reprogramming of the card. There is no password or other methods preventing change of the card or remote reprogramming. Any of these methods (among others) can be used to replace the software responsible for report generation on them or replace the cards with new cards with modified software on them. Method a also enables pre-election manipulation of the vote counters, injection of extra data to be transmited to central tablulator Election Night and conceal this with modified pogramming.*
>
> *To avoid detection, the perpetrator must prevent or subvert any hand counting or replace the paper ballots with forged ballots.*

Applicability

This attack applies to Optical Scan systems where software resides on the memory cards or other forms of removable or rewritable media.

Given the confidentiality of voting technology in the US, it is not possible for us to know exactly how many vendors keep their “print drivers” or "report generators" (the software which tells the printer how to tally ballots) or other executable software on replaceable media. However, we are certain (via testing performed by Harri Hursti) that at least one major vendor has its report generation program on replaceable memory cards. Memory cards are not vetted by the Independent Testing Authorities before being used.

Resource Requirements

Perpetrator(s) will need some programming background and (1) access to the cards, (2) the ability to inject files directly or indirectly to central tabulator before election definitions (i.e., the "defined ballot" for the election) are copied to cards (tampering with the central tabulator might be done on-site, or via modem if locality using PCOS connects the central tabulator to a telephone line, or (3) reprogramming the memory card via modem if the PCOS is connected to the central tabulator via a telephone line.

Note: The central tabulator is most often employed to perform ballot definition (i.e., creating ballots for election), copying of ballot definitions to the memory cards (so that voter choice will be recorded accurately), as well as tabulation of voter choice. The central tabulator is a conventional PC with additional software added. Accordingly, it provides a convenient single point of attack from which one can modify all the printer drivers from all the PCOS scanners. If this machine were to be used to generate the list of the Automatic Routine Audit (ARA) random polling places to be hand-counted, the attackers could arrange to make sure that the attacked polling places were never audited. This would assist the perpetrator(s) in avoiding attack detection.

Potential Gain

The number of votes that could be stolen this way is only limited to the number that could plausibly be changed without raising suspicions due to differences with exit polling and other polling numbers, etc.

Likelihood of Detection

If no hand count is performed, detection is unlikely.

Countermeasures

- Automatic Routine Audit (ARA) were the polling places are not selected by the tally server but "out of a hat" from a list known to be complete.
- Avoiding interpreted programs (i.e. programs that are not "compiled" and therefore somewhat easier for attackers to read and/or modify.)
- Avoiding the use of software on replaceable media
- Avoiding the use of any software by making all programs into firmware (programs that are burned as read-only onto special memory chip) (see: Read Only Memory) and that is validated via a strong method (i.e., someone is authorized to periodically pull the memory chip to ensure that it has not been tampered with) as in the gaming industry.

  Use or 3rd party equiment and software to compare memory cards with known-to-be-good reference image. It is important to know that due the central tabulator can be infected, central tabulator itself can not verify authenticacy of the card.

Attack Economics

One person with programming experience and access to central tabulator and/or the PCOS units.

Variations on attack theme

Attacks where a marked ballot can change the tally total.

Conclusions

Automatic Routine Audit (ARA) is critical.

The ITA system appears to have failed to warn the potential buyers, the public at large and computer security experts that the architecture of this system left open a “backdoor” vulnerability.

Citations

- Original report

Retrospective

This "backdoor" to installing software means that the software inspected by the ITA is not even necessarily the software that will run on Election Day. In certain makes and models the Logic & Accurancy test software is completely separated from Election Day under all circumstances, rendering L&A test results always meaningless.

The fact that vendors have created a system which allows users to replace software via memory cards suggests that they are extremely concerned with creating a flexible, adaptable system. Unfortunately, this flexibility opens up risks we need to be aware of and to mitigate.

Comments:

From JohnKelsey - 2005-09-19 2:58 PM

It seems like the obvious countermeasures here involve not allowing executables to be tampered with. Any kind of open-ended evaluation ought to catch this, and any decent security standards should say that you're not allowed to leave executable code someplace where it can be accessed by the attacker without some kind of cryptographic protection.

Trojan Horse In Tally Server (Attack1)

Taxonomy:

Configuration, Change Management, Account Management

Applicability:

Tally Server

Method:

Assuming a well motivated and financed person or group seeks to alter the outcome of an election. They would then determine the target environment as outlined in the resource section, below. Once key positions one and two (see "Attack Team Core Personnel", below) were identified and brought into the attack team, enough sensitive information would be collected and/or readily available to reach “critical mass” and design, develop and implement the technical mechanisms (e.g. determine a trojan horse attack method, the payload to be carried, the interaction points with existing program structures, etc.).

Trojan horse code can generally be placed into one of two classes:
1) Closed-loop, or self-contained code that requires no additional input to execute its task.
2) Open-loop code that creates a communication channel that can be used at a later point in time to accept additional input (explicit commands, scripts, etc.) in order to alter operating system or tally server application operations, or accept additional, 2nd or 3rd generation closed-loop trojan horse code. Open-loop is often referred to as a “backdoor”, typically created by members of the original programming team to allow system access at some future time when their authorized access has been suspended. [ed: Matt, do you have reference material for this section?]

An open loop Trojan could be placed in the original source code of the tally server by the vendor, masked in such a way that it would be unlikely to be detected. A subsequent 2nd or 3rd generation Trojan, developed by the attack team to target an upcoming election, would be placed on the tally server as part of the preparation cycle of the server for the election. The Trojan would be designed to only activate during an election (e.g. only after midnight on the first Tuesday of November), and not function when the system was placed into an a testing mode.

Once triggered, the Trojan Horse code could modify the tally server's voting database directly, or by intercepting the communication path between the tally source [e.g. the memory card] "read" and the database "write" actions, and alter the totals. It would also need to perform some housekeeping tasks on tally server, such as altering audit logs, file timestamps, transaction records, etc., to mask the evidence of its own activities.

An additional or alternate function of the tally server Trojan could be to re-write the memory cards as they are presented, to reflect the falsified data. Since some jurisdictions may consider these cards as the only official record, they would be favored over the Direct Recording Electronic (DRE) voting system's paper receipts or any DRE on-board audit records. Even more disruptive is the possibility that an election official could help “lose” the paper records.

Resource Requirements:

Based on pre-election voting registration, published polling data, census data, and statistical analysis, the attack team would target counties or precincts by selection for where the anticipated voting margins are razor-thin or dead-even, and where there is a sufficient population to change an election outcome. The targets are further refined for 1) low-budget elections offices, where the resources for system security, such as staff, tools, or procedures, are most likely to be thinned or omitted; 2) busy population districts, where it is presumed that the compromise would have a better chance of escaping detection; 3) districts where voting equipment is known, and may lead to a preference in attempts to suborn technical personnel (i.e., where vendor X equipment is in use, and the attack planning team has existing relationships or contacts within the organization of vendor X).

Attack Team Core Personnel

1) Vendor personnel. Someone with technical duties, such as programming responsibilities, that provide access to the system architecture, source code, and possibly direct physical access. If they had support duties with one or more targeted county or state elections organizations, even better, as that reduces their distance from the live system to be compromised. Physical access is not strictly necessary, as the activation of a Trojan horse would fall to the elections insider (#2, below) who would have the last possible access to the system before the election. A substitute for Vendor personnel might be someone at the ITA or someone who had access to a stolen machine to reverse engineer the technology.

2) Technical Elections official: This would be a mid-level county elections official, either an original conspirator or someone who can be motivated to participate in the attack by outright bribery, self-advancement, or extreme political ideology. This official would posess knowledge and/or provide oversight of some or all of the elections systems, including the operating processes on the tally server, the design of the balloting, etc.

3) The financier: Someone has to pay the bills necessary to gather the intelligence on the first two positions above, fund the bribery, and pay for research and development. This could be an individual or small collection of people dedicated to achieving the goals of the attack, able to manage the project and willing to fund the efforts.

Additional Resource Roles: These roles will be expanded as necessary to overcome identified obstacles (below)

4) Polling place elections judge: In order to co-opt the paper receipts that the individual DRE’s

generate (Obstacle 1, Paper records)

5) Professional-caliber software developer: a “hired gun” capable of 1) writing Trojan code that was original, and not just a re-use of an existing exploit; 2) understanding the programming approach of the tally server application processes, and isolating vulnerabilities; 3) designing a coordinated attack that could dynamically react to a range of tallied data conditions and alter them in a way that creates a plausible false record. The more resources that are invested in technical expertise, the better crafted the attack would be, and thus the more likely to succeed. (Obstacle 2, Complexity)

Potential Gain:

To successfully alter the tabulated vote from some or all DRE’s in a county. A Trojan Horse attack would not have to explicitly reverse the recorded outcome of an election (e.g., create a republican victor in a predominantly democratic district) to be successful. It may be sufficient to simply alter a few undervotes, or reduce the margin of victory by a few votes. This would also be useful in voting events where a simple majority was not the determinate state, but where a ratio of votes (e.g., electing candidates across multiple possible positions, or as the basis for determining electoral representation) would be of enough interest to motivate the attackers.

Likelihood of Detection:

The Trojan Horse code could be detected in several places: by the vendor, by the test lab, or by an election official noticing anomalous results during a test or in a real election. A skilled programmer, however, will generally be able to hide a significant amount of dangerous code without being detected in testing. If hidden in the subroutine code of supporting Commercial Off-The-Shelf (COTS) libraries or software objects, the Trojan would not be discoverable by the test lab, as they are not expected to have access to the relevent COTS software. (See countermeasures.)

Obstacles to attack:

1. Paper Records. Each DRE generates a paper receipt with the tallies of the votes recorded. These paper records are forwarded to the tallying place as part of the official record, and should be cross-checked with the electronic tally figures.

Counteractant: See Additional Roles, #4, above.

2. Subject Matter Complexity. Elections processes are too specialized to be easily understood by an outsider.

Counteractant: See Additional Roles, #5, above.

3. Ballot Assembly Timeline. The dynamic nature of the ballot database might, at times, leave too small a window of opportunity for such an attack to be mounted.

Counteractant: See Additional Roles, #5, above.

4. Security Process. The tally server is an important piece of hardware, kept in a secure location at all times. All physical access is controlled by authorized personnel, and all logical access is carefully controlled and monitored. (Many counties are not organized to protect a tally server or detect intrusions against their servers.)

Counteractant: An effective, around-the-clock security process is 1) expensive, 2) only as effective as its weakest link, and 3) unlikely to be reviewed. Election cycles are awkward in that they do not fall frequently enough to make good security practice a habit, or to even assume that the same people will participate in sequential events. Security awareness in employees must be continually retrained and refreshed. And when the voting system has been owned by the state/county for several years, certain contempt for process (e.g." we've never had any security problems here") is inevitable.

Also, an elections official (even though corrupted) is likely to have enough credibility to overwhelm any question of a security risk in a parochial setting. This problem is exacerbated when the typically small community of elections professionals is taken into consideration. A recent annual study released by IDG and PricewaterhouseCoopers, surveying information security professionals, reported that current employees accounted for 33 percent of all security related threats, up from 28 percent reported in the same period last year.

Most importantly, a security process is only effective when it is being actively monitored and enforced. Too frequently, this process is foregone in the interests of cost and expediency. Mandatory post-election audits should examine all aspects of system security and elections process, to verify that procedures were followed, and to determine the procedure effectiveness.

5. Automated Security. A tally server employs a wide range of automated security controls, to prevent physical and logical tampering, and provide detection of unauthorized changes.

Counteractant: A tally server is most often a PC-based system, with all the known vulnerabilities that implies. Also, voting systems do not require even the basic review of security controls that traditional information systems employ.

It is also axiomatic among security and hacking professionals that once an attacker has physical access to an automated information system, any security control can be defeated.

6. The Voting System Testing Laboratory (VSTL) certification testing process. The certification process to be conducted by the VSTL could discover any Trojans included.

Counteractant: Any Source Code review is extremely expensive, and is not 100% effective. Code review is typically a two to five person-year effort, over several months, at a cost anywhere from $300,000 to $600,000 [N.B.: We need a verifiable source for the quantified cost]

Even where source code is provided to the VSTL, Trojan code can be placed in the underlying operating system, support libraries, and other objects that are not included in the certification review. [N.B.: Need Matt Bishop reference on program review and securing code]

In cases where the source code is not made available to the VSTL, review of compiled tally server application code is extremely difficult, and even more expensive, and vastly less effective.

Countermeasures:

- Visible inspection of the DRE receipts at the polling place
- Wide dispersal of DRE receipt data among elections judges, observers, and general public at the polling place, with instructions on how to verify these figures against officially posted election results.
- Pre-election simulations with rigorous statistical analysis run on tally servers.
- Use of dual tally servers during elections, preferably at different sites, to compare results.
- Use of tally server system security tools and least-privilege enforced access policy
- Installation of active configuration management software to monitor the tally server software and determine if changes have been made.
- Disabling communication channels for software and other inputs to the tally server systems, Removing network adapters and all external connectivity from voting machines, except for allowed connection to the tabulation center over a cryptographically secured link.

Citations:

http://www.redherring.com/Article.aspx?a=13555&hed=Security+Threats+Rise+22%25+&secto r= Industries&subsector=SecurityAndDefense

NIST Special Publication SP 800-30, Risk Management Guide for Information Technology Systems

Attack on Configuration Data

Eric Lazarus/Stephen Green

Taxonomy:

Administrative, locale, legal

Applicability:

DRE, DRE with VVPT, any configurable electronic voting machine

Method:

The perpetrator must configure some machines to either discard ballots or count ballots when abandoned by the voter and to do so in violation of election law. These malevolently configured machines could be, for example, distributed to polling places likely to be unfriendly to the candidate or proposition the attackers are attempting to benefit.

Voters do abandon their ballots on occasion, either out of ignorance, carelessness, or confusion. DRE, DRE with VVPT machines permit this confusion because some voters, on seeing the review screen, could believe that it was reporting the vote they had cast and not the vote that they could cast if they press the red "vote" button (or in some other way indicate that the vote should be cast.)

The election officials, on finding a voting machine with an un-cast ballot, generally have a procedure to follow: Two poll workers working together are to go to the machine, insert the supervisor PEB, enter the "poll worker override" password and force the system to perform the programmed action for abandoned ballots.

Resource Requirements:

The attack is subtle enough that a single insider might well carry it off, with overall effectiveness determined by where the configuration files are maintained and controlled. That insider could be a county election official, voting system vendor, or contactor.

Outsiders could carry off this attack via a break-in to a warehouse or via many small break-ins to actual polling places, the latter being far less efficient.

Potential Gain:

? How many ballot abandonment cases are we likely to see?

Likelihood of Detection:

In the case of the cited FL elections, misconfiguration was not detected during the entire operational period from the original purchase and installation date. Typically, misconfiguration would not be detected until a forensic audit is conducted to validate election results.

Countermeasures:

Two-level configuration files or other ways to detect or prevent incorrect settings.

(see also countermeasure page on wiki to ensure we are covering relevant countermeasures already discussed)

Preventative Measures:

Centralized control of configuration files and a secure means of distributing the files out to the precincts. If local configuration is a requirement, a two person control process could be implemented where one person enters the required information and the second person verifies the information has been entered correctly. The configuration files are then hashed and recorded to address any post-election concerns.

Detection Measures:

Detection of configuration file errors could be accomplished through the use of setup validation, either automatically by file hash validation or manually through procedure. Printouts of the configuration files, pre- and post-election could be used to detect tampering or misconfiguration.

Attack Economics:

Small number of attackers. Number of votes that can be stolen this way is ______________________________ (please fill in!) This would depend on where the configuration files are controlled and how many voters abandoned, by negligence or persuasion, their ballots.

(Please see BC attack catalog info for BC estimates of costs)

Variations on attack theme:

Attack on PCOS systems by turning off over vote protection. General concept is look to the handling of unusual cases because the human mind tends to focus on the normal case.

Other system configuration settings besides handling of abandoned ballots need to be evaluated, i.e. counter thresholds, randomization seed values, etc. The location where the configuration information is stored should also be evaluated, i.e. is it stored on removable media, in a flashable memory module, a protected directory on a local file system, etc.

This attack can be considered a form of Trojan Horse except that no computer skills or software modifications are need. A small percent of votes are never cast for the attacker’s opposition due to a plausibly deniable incorrect configuration setting.
(e.g., what variations of this attack are there -- see BC attack catalogs for preliminary thoughts)

Conclusions:

Processes, procedures and technology that are not observable seem to create risks of fraud and of error both.

So, in this example, unlike the situation with conventional paper ballots, where an election observer could easily tell whether the pollworkers were placing that ballot in the ballot box, on the one hand, or in the trash can, on the other hand, nothing an observer could see would indicate what had happened to the ballot. In fact, it is not clear that a very knowledgeable pollworker could see what had happened.

Configuration settings which impact vote totals need to be overtly obvious to pollworkers, especially for abandoned ballots and the corresponding procedures and laws. During normal “poll worker override” operations, extra care and effort by the system designers is needed so that the display properly conveys the actions being taken with the ballots.

(i.e., countermeasures a, b and c likely to be most effective, countermeasure d inneffective because technology too expensive or not advanced enough)

Citations:

Miami-Dade Elections Supervisor Constance Kaplan resigned in March of 2005 because apparently, for at least one year Miami was using DRE systems the option had been set wrong, presumably not malevolently but due simply to the large number of settings and the fact that setting them right requires a detailed reading of the law and the ability to deal with nonobvious user interfaces in preparing the options file. There is lots of room for oversight and clerical error, and each county is on their own to get it.

While the legal responsibility sits squarely on the county election supervisor. The basic system design makes it at once very difficult for the commissioner to be sure that the law is being carried out and, at the same time very easy for another individual to expolite the volnerablity.

Miami-Dade elections chief quits under fire, Associated Press, April 2, 2005

Ballot Marking Device Attack

Taxonomy

Configuration-related, Ballot Database

Applicability

DRE, DRE with VVPT

Method

Based on assumption of Trojan code placed on voting machine that can successfully manipulate ballot database, and/or voting machine presentation of ballot data.

Resource Requirements

Expertise in voting application, voting system auditing.

Potential Gain

Medium to Low. Each voting machine must be modified; moreover, the exploit is in operation only until the omission is detected and verified by the polling place judges, at which point all machines in polling place would be examined and the tampering discovered and the altered ballots discredited.

Likelihood of Detection

This attack is predicated upon the voter overlooking the omission, and thus is only effective for small, low-awareness proposition issues and races. Most polling places publicly post a sample ballot, as well as offering one in the voter registration process. Local Elections officials also conduct mailing campaigns to inform the voters of the ballot prior to election day. Finally, the exploit must go undetected for the entire voting day (typically 13-14 hours) if the affected votes are to be entered into the official tally.

Countermeasures

Two-level configuration files, Checksums, configuration checking tools, or other ways to detect or prevent alterations to approvied configuration profiles.

Preventative Measures

Creating a well informed voting public, by means of public posting of official ballot content and other awareness-raising efforts prior to the day of voting (e.g. activities by the League of Women Voters, and the various political parties).

Detection Measures

A well-trained staff of polling place elections judges. An effective Logic and Accuracy (L&A) Testing protocol would also disclose any ballot alterations, assuming the trojan code enabling this exploit could not mask itself and its changes when the machine was in testing mode.

Trojan Horse in DRE -- OS

Taxonomy

Configuration-related, Change Management, COTS, OS, Trojan.

Applicability

DRE, DRE with VVPT

Method

A third party supplies a well known, publicly available operating system used in a DRE. The attack team introduces a Trojan horse that is activated on a specific date (e.g., the first Tuesday after the first Monday in November). The Trojan horse detects when a ballot is displayed, and reverses the order of the first two entries on the screen (so if the order should be, for example, John Adams and Tom Jefferson, the displayed order is Tom Jefferson and John Adams). The Trojan horse also checks for the names on the review screen and if either name appears, the other is substituted.

If desired, the trigger can be some event other than a date -- for example, if a voter selects and then cancels a certain candidate four times in a row, or if three voters sequentially vote for the selected candidate, then those patterns can be a trigger for the Trojan horse.

The Trojan horse is inserted when a piece of the operating system is rewritten, either by the perpetrator or by someone whom the perpetrator has compromised (bribed, blackmailed, etc.) The driver is NOT written by the DRE software developer, and hence is COTS software. The Trojan horse code can be placed within any OS components that are known to be configurable for a running installation, such as the video driver, the user interface devices, the drivers for removable storage, etc., that would not be part of the standard COTS as delivered by the vendor, but would be expected to test positive for signs of change.

Resource Requirements

Access to each voting machine, the host machine(s) on which the master images of the OS or application software are created and/or stored, or any intermediary system(s) that might be responsible for installing software onto the voting machines. This would also include access to those systems responsible for delivering software patches or updates to the DRE over the course of the system's operating lifetime.

Potential Gain

Most operating system functions are executed within privileged space in the system architecture, which means that they have both the rights and the ability to make any and all changes they wish to any part of the system, including those routines (such as audit logs) that are supposed to detect inappropriate behavior. As such, a successful attack on the DRE OS would open the door to any tampering that the attack team could create.

Likelihood of Detection

Detection would depend upon the (TBD) rigor of the Voting System Testing Laboratory (VSTL) examination process, and/or the pre-election testing of the voting system. Testing would have to be conducted in such a fashion that the complete ballot input and output datasets would be validated, and that the system would not offer evidence to the trojan horse code, such as entering a defined "test mode", that would enable the code to mask its presence and remain dormant throughout the testing.

Countermeasures

This attack presents little in the way of a knowable profile, making countermeasures almost impossible. Fingerprinting of the OS in the form of hashing an approved version would capture a trojan horse in the core functions, but would not include those modules and drivers that are typically reconfigured upon installation of the OS on a given device (or set of devices).

Paper Trail Boycott

Michael I. Shamos
Oct. 5, 2005

Taxonomy: retail, paper-trail subversion
Applicability: all paper trail systems

Method:

Assume that a given precinct is known to heavily support Party A and therefore Party B wishes to reduce the turnout in that precinct. Party B enlists legitimate registered voters in that precinct to appear at the polls early in the day to vote. Each of the voters complains to the poll workers that no matter how many times they try, the paper ballot never corresponds correctly to their choices. The election officials will have no choice but to remove the offending machines(s) from service. This will reduce the number of available machines, possibly to zero, and will cause long lines to be created and a large number of voters to leave without having voted, accomplishing the desired goal.

Resource requirements: Cooperative voters willing to sacrifice their votes.

Potential gain:

Small, and on a precinct by precinct basis only.

Likelihood of detection:

Unknown. With DRE systems exhibiting a failure rate of around 10% just on Election Day, a report of a malfunctioning machine is quite normal. Widespread implementation requires a conspiracy involving a significant number of people, is difficult to manage and creates a high risk that a traitor will reveal the fraud.

When the machines are examined after being removed form service, maintenance workers will be unable to reproduce the fault, but they will also not be in a position to know the sequence of touches that allegedly led to the problem.

Countermeasures:

Preventative measures:

The paper trail statutes do not deal with the question what to do when machine faults are reported. In some precincts, officials may remove a machine from service quickly. In others, they may allow the problem to continue all day.

Making a false report of a voting machine failure should be criminalized, but detection will be nearly impossible and prosecution consequently rare.

Detection measures:

Abnormal (?) frequency of reported paper trail problems.

Paper Trail Manipulation I

Michael I. Shamos
Oct. 5, 2005

Taxonomy: wholesale, paper-trail subversion
Applicability: cut–sheet paper trails that print non-human-readable indicia

Method:

To prevent ballot-box stuffing with forged ballots, most "voter-verified" paper-trail systems print one-or two-dimensional barcodes or cryptographic indicia, on the verified ballot. The indicium is usually a computed function of the content of the ballot, e.g. a hash. The indicium may also contain a pointer to the electronic record that is supposed to correspond to that specific ballot. In the event of a recount, legitimate ballots will possess the correct indicia, while a forged ballot will not.

Assume that the code in the voting machine has been subverted as follows: the system always produces accurate voter-verified ballots, but when a voter votes for candidate A, then with probability p the barcode or indicium is printed incorrectly and no electronic record is made of the ballot. The voter believes the ballot is correct, and therefore indicates that the vote should be recorded. The ballot is automatically dropped into the ballot box. After the voter leaves the machine, a new ballot is printed with a vote for candidate B with a correct indicium and an electronic record of this ballot is made. The second ballot is also deposited automatically in the ballot box. This effectively switches a vote from A to B.

When the polls are closed, the software removes all trace of the manipulating code so an inspection of the software after the election will not reveal anything amiss.

The method will not be successful with continuous-roll paper trails. Because of the physical integrity of the paper roll, there will be no rational explanation how ballots with incorrect indicia became interspersed.

Resource requirements: The perpetrator must be intimately familiar with the voting machine code and be in a position to substitute what amounts to a Trojan horse for the legitimate software.

Potential gain:

Massive, depending on the extent to which the manipulation is deployed. Care is required in selecting which races to manipulate, and by how much (i.e., the choice of A, B and p). If the swing is too lop-sided, great suspicion will be raised, but it is not clear what can be done about it.

Likelihood of detection:

If there is no recount, the manipulation will not be detected. If the ballot box is opened and the ballots are counted, a discrepancy will be observed between the number of voters who voted and the number of ballots in the box. Unless the ballots are individually examined, it will not be possible to distinguish the extras from ordinary spoiled ballots.

If a recount is performed, invalid ballots will be present. The number of valid ballots, however, will match the number of voters and the electronic count will match the valid ballots exactly. It is possible that the correct conclusion will be drawn that software tampering has occurred, but since the software has erased any trace of the intrusion, it will not be possible to prove. With the electronic count and the physical count being equal, the intrusion will have succeeded.

Countermeasures:

Preventative measures:

Careful code evaluation at qualification testing and chain of custody of executables that actually get installed in voting machines. Wholesale fraud can occur at the vendor, the distribution point or the county warehouse. Successful manipulation of individual machines after delivery to the precinct is difficult because of physical interlocks and results in retail fraud even if it occurs.

Detection measures:

The printing of the second ballot when the first has been invalidated can be detected aurally.

Parallel testing will also reveal this exploit.

Retrospective:

So-called “voter-verified” paper trails are not actually voter-verified. The paper record should not contain any information that cannot be read or understood by the voter yet can be used to invalidate the ballot when a recount is performed.

Paper Trail Manipulation II

Michael I. Shamos
Oct. 5, 2005

Taxonomy: wholesale, paper-trail subversion
Applicability: all paper trails that the voter cannot touch, whether cut-sheet or continuous roll

Method:

Under HAVA, a voter must have the opportunity to spoil a ballot and vote again. With paper trails, this is implemented by having the system void the paper ballot if the voter does not agree with its contents.

Assume that the code in the voting machine has been subverted as follows: the system always produces accurate voter-verified ballots, but when a voter votes for candidate A, then with probability p the ballot is voided by the machine even though the voter indicates assent, and no electronic record is made. After the voter leaves the machine, a new and non-voided ballot is printed with a vote for candidate B and an electronic record of this ballot is properly made. The second ballot is also deposited automatically in the ballot box. This effectively switches a vote from A to B.

When the polls are closed, the software removes all trace of the manipulating code so an inspection of the software after the election will not reveal anything amiss.

Resource requirements: The perpetrator must be intimately familiar with the voting machine code and be in a position to substitute what amounts to a Trojan horse for the legitimate software.

Potential gain:

Massive, depending on the extent to which the manipulation is deployed. Care is required in selecting which races to manipulate, and by how much (i.e., the choice of A, B and p). If the swing is too lop-sided, great suspicion will be raised, but it is not clear what can be done about it.

Likelihood of detection:

This manipulation will not be detected other than through parallel testing. The voided ballots will appear simply as normal spoiled ballots. The electronic count will match the physical count and nothing will appear extraordinary.

The method will not work with cut-sheet systems in which the voter physically deposits the ballot in a box herself. In such systems the machine has no opportunity to void the original ballot or print another.

Countermeasures:

Preventative measures:

Careful code evaluation at qualification testing and chain of custody of executables that actually get installed in voting machines. Wholesale fraud can occur at the vendor, the distribution point or the county warehouse. Successful manipulation of individual machines after delivery to the precinct is difficult because of physical interlocks and results in retail fraud even if it occurs.

Paper trail systems must be designed physically (not just in software) to prevent this exploit, that is, in such a way that a voter's ballot cannot be marked void without the voter knowing about it

Detection measures:

The printing of the second ballot when the first has been voided can be detected aurally.

Parallel testing will also reveal this exploit.

Retrospective:

So-called "voter-verified" paper trails are not actually voter-verified unless the voter is able to satisfy herself that the ballot she verifies is not later manipulated or replaced.

Cellphone Vote-Buying

Michael I. Shamos
Oct. 5, 2005

Taxonomy: retail, vote-buying
Applicability: all DRE systems

Method:

Many cellphone have cameras that can transmit real-time video. This allows a voter to record and/or transmit what transpires in the voting booth to a confederate who will pay him if he votes a certain way.

Resource requirements: A cellphone and cash to reward the compliant voter.

Potential gain:

As large as with any vote-buying scheme.

Likelihood of detection:

Dependent on the extent of privacy curtains. In the old lever machine booths the method would work perfectly because the voter was fully enclosed. With some DREs there is very little privacy structure surrounding the booth, and detection would be easy.

Countermeasures:

Preventative measures:

Statutory. Bringing camera phones into polling places is illegal in Singapore and a statute is under consideration in Italy. I have no found any U.S. state that forbids camera phones in polling places, although vote-buying is universally illegal in the U.S.

Jam cellphone frequencies in polling places. Currently illegal – would require federal regulations.

Detection measures:

Watch the voters carefully, where possible. Set up detectors to detect cellphone use in the polling place.

Citations:

The problem is not theoretical. Actual incidents of cellphone vote-buying have been reported in at least Hong Kong, Singapore and

Thailand. However, several writers claim that bringing cellphones into polling places can help reduce the incidence of corrupt practices, e.g. http://www.chathamhouse.org.uk/pdf/briefing_papers/Africa%20BP02.pdf, http://www.wirelessmoment.com/legal_issues_camera_phones/

Ted Selker, PhD Computer Science
Jon Goler
Caltech/MIT Voting Technology Project
April 2004

Attack name:
Security Vulnerabilities and Problems with VVPT

Applicability
This is a method of defrauding a paper trail record

Attack method

Designers use statistical assurances that voters don't verify paper trails as a premise. They design the paper trail software to misprint a few enough paper trails that it will go unnoticed but disrupt the election results.

.

Resource Requirements and cost

The cost is the same as any other electronic fraud in electronic voting system cost

Consequences and potential gain

The opportunity to defraud election results

Likelihood of detection

Must be verified by a camera or other enough parallel testing.

Countermeasures

Video verification of paper verification. Parallel testing, Nversion system,

Retrspective and historical notes

This approach came to us after watching how difficult it is for people to verify paper trails

Abstract

A proposed Voter Verifiable Paper Trail (VVPT) includes a printed ballot as a receipt that a voter can view to verify their vote before leaving an electronic voting machine. This method is also supposed to insure the accuracy of the recorded vote by allowing the tally to be checked later by counting the collected receipts.

This paper considers problems with ergonomics, logistics, security, fraud, and mechanical fragility with using VVPT. Ergonomic problems are introduced by the receipt having a different layout than the ballot, difficulty remembering previous selections to make the verification, by the extra step it introduces after making selections and by it not working well for sightless people. Logistics problems include difficulties in collecting and organizing the receipts, transporting them, and reading and reconciling them with electronic tallies. Security issues include the possibility that receipts can be systematically misprinted in a way that cannot be detected and that hand counting will not easily detect fraud. Mechanical problems include printer breakdowns and supplies running out. VVPTs could add problems by being questioned in various ways or though the development of computer programs that defraud the VVPT systematically. VVPTs do not address existing sources of disenfranchisement such as registration problems, equipment and ballot problems, and polling place problems.

Experiments and elections have yet to establish that people can in fact verify their ballots using a paper receipt. Effective approaches for accurately counting the paper receipts for auditing purposes have not been established either.

Proving that an election correctly records and transmits the intention of the voter is worthwhile. Computers are the first technology that can easily report voting results in multiple formats. Simple systems-verification solutions are possible. Parallel voting and time shifted testing require no extra equipment. Voter Verified

Audio Transcripts would simplify voting and improve audit security by presenting verification as feedback during the selection process rather than post hoc auditing. .

Introduction

Choosing a government is contentious and the mechanisms for collecting and counting votes have always been on the minds of the people involved. In ancient Greece, Egypt, and Rome people used physical objects, like shards of pottery, to document their choices. Over the last century, developing voting technology has continued to improve the way votes are marked and collected. In 1868 Thomas Edison invented an electronic voting machine. In the 1890s the so-called "Australian secret ballot" was adopted in United States. Hand transcription of marks on paper has given way to automated optical sensors reading the marks. Automated counting reduces the problems of people overlooking, adding, or removing a mark. Writing down columns of local tallies to be added together by hand has given way to spreadsheets and automated calculations. These methods further eliminate human errors. New computer voting machines will not let voters make the mistake of leaving extra marks on alternative selections or making too many selections for a race. Automated processes are eliminating some errors, as well. Prospects are good for using technology to simplify the voting user experience and increasing its accuracy.

However, all technological improvements raise questions and must be implemented in a controlled way. In the case of voting technology, improvements have required experiments, slow rollouts and adjustments. Brazil introduced electronic voting in stages. In 1996, Brazil put electronic voting into place for 40,000 voters with 7% not being able to succeed at recording their votes electronically. Improvements from that experiment allowed this rate to fall to 2% for the 150,000-person experiment 1998. Improvements from that experiment resulted in only an estimated .2% of 106 million voters who were unable to electronically deposit in Brazil in 2000.

User experience problems plagued the early electronic voting machines introduced in this country; in some cases the number of votes that were left unmarked on the new machines was greater than for the equipment they replaced. For example, some electronic ballots placed the selection to scroll to the next race too close to the selection for depositing the ballot, causing some voters to inadvertently cast their ballots before completing it.

In accordance with law, the paper punch cards from the 2000 Florida election have been destroyed. Many people believe that we will never know the intentions of the voters in United States 2000 presidential election. Forensics [1] shows that 2 to 3 percent of the votes were lost due to problems with registration, ballot design and polling place operations. These problems are not new or unusual but are dramatized by the closeness of the 2000 presidential race, coupled with the desire to properly vet its outcome in an information-sophisticated world. These simple-to-solve problems are not being addressed systematically. Instead, the public conversation has shifted to more vague issues of technology in elections and fraud.

The call has gone out for approaches that will produce accurate, secure recording of votes with complete integrity [6]. Unlike paper ballots, voting machines give feedback to voters as they vote. Voting machines that disallow voting for too many candidates have reduced disenfranchisement of voters [7]. The common belief is that electronic voting machines will simplify the vote collection and counting process for all. Historically, the fragmented voting industry consisted of several companies that compete for the occasional upgrade. In the wake of the 2000 election, the Help America Vote Legislative Act of 2002 changed this in that it made available $1.2 billion in 2003 to upgrade the country's voting machines quickly [3]. Are these monies being released to buy machines when it could be better spent researching how to improve them and the processes in which they are used?

Concerns about security of the collection and counting process have always been important. Computers offer the first technology that can easily make copies of information in different forms for archival preservation. Electronic voting machines of today keep records of the votes on disk, removable physical media in memories and, as a final count, on a paper scroll. These multiple records can improve voting machines' immunity to problems. For example, if a floppy disk from the Brazilian Procom voting machine is unreadable, the election administrator records another one from the internal flash memory in the voting machine.

However, the big question is how can we prove that the selections made on a computer interface by a voter are reflected correctly in the digital voting machine records? Critics of using computers to perform secure operations are speaking up. Broad media coverage has been given to the issue of how we can know that a vote is collected without the computer program tampering with it.

Many approaches to ensure the secure transfer of a voter's selections into the computer are possible [2]. Adequate and provable electronic security could make certain that the vote tallies reflect the voter's intention. A separate Votemeter machine can check the voting machine while it is running. Modular architectures can segment the process so that any changes in the votes would take multiple changes to code written by different organizations. Some call for the code being open for anyone to view in a so-called *open source* way. Many believe that separate records that are human readable will be most helpful. Open viewability of a second ballot has seemed attractive to many.

The most popular of these in the public's eye have included Voter Verified Paper Trails (VVPT). The various schemes for this all include a display on which a voter makes selections and a way of viewing a paper receipt that is printed to reflect these selections. The voter cannot take this voting receipt away with them because if they did, it could be used to show how they voted and would compromise the secret ballot and security of elections. Nonetheless, such approaches have captured media and governmental attention as a solution. This paper describes some of the difficulties with VVPTs. A forthcoming paper will describe several alternative verifiable approaches to security.

Ergonomics issues

The VVPT is in a different format than the ballot, in a different place, is verified at a different time, and has a different graphical layout with different contrast and lighting parameters. Handling VVPTs causes other ergonomic problems for the ballot workers. During the first use of VVPT in an election, in November 2003 in Wilton, CT, virtually all voters had to be prompted to find and verify their receipt. This turned into extra effort for poll workers and extra time for voting. Anything that takes a voters attention away from the act of casting a ballot or causes a voter to invalidate their vote will reduce the chances of them voting for the candidate they intended. Many voters are frightened of going to balloting places because they fear intimidations that actually can transpire. They fear the voting process, the technology, and their registration not being there. The complexity of the voting process is already a deterrent from voting; VVPT adds complexity, which could drive away more voters.

People are extremely good at remembering hundreds of precise images and comparing them against the same image [7]. But the format of the paper receipt will be different than that of the voting machine and because of these differences it is difficult for people to compare them after the fact. Most people have had the experience of taking two columns of numbers and finding it difficult to verify that they have not missed a number. Comparing dozens of selections on a voter-verified paper receipt will take such special care. Complications of comparing a separate paper trail in a different ballot format might add extra difficulty for people with learning and reading disabilities. The Wilton, CT experiment found people not noticing the VVPT because it was in a different place in the booth.

Time limits on voting (3 minutes in New York City) are designed to keep balloting running smoothly. This time will likely need to be extended to allow for checking of the voter-verified paper trail. When people are focusing on a ballot it will be extra work to remember that they have to look at another place to verify their ballot.

When a voter deposits his or her punch card ballot into the ESS PBC 2100, an electronic display shows that the voter has not voted for every race correctly, a paper trail is printed showing exactly the races in which a voter did not vote correctly. This system only shows problems that should be attended to and should be much easier to understand than a paper trail. In watching 500 voters casting ballots, I saw less than one in

10 people who, when they were told they had a problem with their ballot, were actually willing to take a new ballot and vote again. There appeared to be four reasons for this: many said they "knew" they had done the right thing and it must be all right, many felt pressed for time and wanted to leave, some were embarrassed, and some seemed overwhelmed. The task of reviewing the ballot after a person believes they have completed the task can be anticlimactic. One thinks they are done with voting but must go through it again.

The biggest difficulty in verifying a paper trail might be that some jurisdictions have over 100 races on which a voter makes selections. Remembering how one voted on each is difficult. Without a reference guide, it is likely that people who make decisions while marking their vote will forget how they marked the ballot that they are checking. Incorrectly calling fraud on a ballot machine will slow or stop others from getting to vote. In any case the difficulty of the cognitive task of checking a ballot afterwards will be much higher than any perceptual task that is required of the voter while they are marking their ballots [4].

The most popular description of VVPT places it behind glass to avoid losing the integrity of the secondary ballots. To the extent that the paper trail is not directly against the glass or the glass is not thick, offset parallax can make it hard to view. The apparent position of a finger against the glass changes with the viewing angle, making it difficult to accurately see which selection is being verified on a ballot with dozens of races.

Additional ergonomic considerations include lighting and readability issues that probably can be dealt with. For some vision-impaired people magnifying glasses and lighting will not make this process more accessible. A different verification mechanism such as audio verification will be required for them not to be disenfranchised.

The step of reviewing the voting machine after using it has been difficult for voters. In Cook County, IL there are videotapes or machines to train people in using the ESS PBC2100. But, in visiting some 60 precincts, I never saw anyone watch the video. Maybe people believe that they can figure it out once they are in the voting machine.

Ballot worker ergonomic problems exist in the logistics of keeping the receipts secure, counting them, verifying that they are the same number as the number in the DRE, sealing the receipts in a transport box, checking that these are prepared correctly for transport (hopefully under scrutiny of more than one person), and transferring them. Ergonomic problems complicating the process turn into logistical problems.

Logistics problems

Collecting and counting the ballots can be difficult. In Wilton, CT the ballot boxes had a gap through which ballots could have fallen. While watching a precinct close down in Cook County, IL in March 2002, we noticed a ballot on the floor. Transporting ballots has posed problems. Even in LA County, in the last use of punch cards in October 2003, a ballot box was lost for several hours. At 2:00 a.m. somebody had to go look for the hopefully-untampered-with missing box; finally it was found behind a door in the polling place. Ballots have been known to fall off the top of cars and have been left in trunks of cars during transportation. There were allegations in the 2000 election of replacing one set of punch cards in a balloting place with another. Typically a ballot worker transports ballots in a personal car to a collection station. In the fall of 2003 San Francisco election, some ballot workers transported paper ballots in shopping carts down the street. These methods of transportation raise serious concerns on the security of votes.

By the time election workers shut down a polling place, many of them have worked a 13-hour day. In LA County we recently saw a poll worker bully others into saying that they had completed checks that only one person actually did. We saw people closing a ballot box and covering the bar code "for security" which would make it unreadable by the machine as it traveled to the paper ballot collection center. These kinds of mistakes with physical things are always an issue for any system that a person is not familiar with or does not do on a regular basis. When people are doing something that is very important, nervousness as well as fatigue can make them less reliable.

Arranging to store and read the ballots later presents formidable problems. Punch card holes are designed to be the simplest of all possible separate paper records to read in an automated way. While it is easy to read one or ten cards, no one has made a reader that can read a million reliably. Being human readable will make it harder to accurately read the ballots with machines. Even when multiple people read ballots together the tally can change with multiple readings. How many hand counts are required to certify correctness? When the number is different between the paper and the electronic, which one should be trusted? Reading scraps of paper or receipts automatically has not been established as reliable. Machine reading Optical Character Reader (OCR) scan ballots, and punch cards, are more reliable than people reading paper [1]. The suggestion that some human -unreadable indicator, such as a barcode, be included on each receipt compromises the VVPT proponent's goal of the humans as the final judge.

The fact that the VVPT is not the primary election count will be known by the ballot workers likely leading them to be less careful with them than with primary ballots. Since receipts are curled thin paper, the process of counting them at the end of the day is harder than counting paper ballots. Not counting them at poll closing will make it harder to validate later.

Receipts printed with paper tape are hard to stack or organize. In Broward County, FL, for example, the ballots are counted in a warehouse where a loading dock door is commonly left open, letting wind blow in that could shift the paper. VVPTs will require workers to handle scraps of paper curled by the roll in the machine. The mechanical problems of handling the thin paper will be worse than with customary ballots. Interpreting the human readable words on them will be more complex than registering a hole or a filled-in oval.

All election machines today allow an administrator to change the time. Changing the time on the voting machine, ballot, or OCR could allow someone to maliciously revote a precinct. Knowing how many people voted for the day, a dishonest poll worker could fraudulently revote the election. The worker could produce a new fraudulent VVPT, putting into question which VVPT is correct. Luckily this would be a labor-intensive way to defraud an election.

Counting the paper trail presents other problems. Ballot workers arranging and moving cards around always seems precarious. Ballot workers who are running a punch card machine have procedures for dealing with misread cards. Even when everyone is watching in an organized punch card reading operation, people worry about cards getting disorganized, out of order, and being removed or changed.

People are inured to paperwork. People who work with computers constantly have to approve long contracts in order to install software. Computer users are used to approving contracts without reading them completely; most just press the approve button. Conversely, for the non-computer users, the very idea of checking a computer might be confusing; how would they know what to trust? Now consider people who go through checkout lines in the grocery store. When I was a teenager I bought food for my family and had to be frugal. The cashier hand transcribed the prices into the cash register; I would check my receipt and often find an error; when in my favor, I was refunded. Today cash registers that scan prices have reduced the problems of transcription of the prices and are more reliable. It is not so common to find errors any more and many people do not look at them. ATMs also give receipts. These receipts often have the balance of a bank account and can even indicate the account on them. Even with important financial information on them, these receipts are dropped on the floor or put in the trash can right next to the ATM where anyone could see them. Being surrounded by receipts that we do not pay attention to is an impediment on taking the voter verifiable paper trail seriously. It is unclear that voters will be more careful with a VVPT than they are in caring for their receipts at an ATM or in a grocery store.

Illiteracy can also be a problem when trying to verify a ballot. Variation in formats between the ballot and a verifiable paper receipt can confuse the voter. Voter information often helps people to familiarize themselves with the ballot they will see on the voting machine or to create a crib sheet to allow them to recognize where to mark the ballot. Unfortunately, the paper receipt is in a different format and would require a separate verification sheet to be tested by an illiterate person.

Less than fifty percent of eligible voters in this country vote. The increased logistical problems introduced by VVPT will not make people think voting is easier.

Software Security and Fraud in Voter Verification systems

A natural question about voting concerns possible fraud. David Orr, the county clerk of Cook County, Illinois, said he believes that only 1/3 of voters who are told they have an overvote will take a new ballot. Others have described seeing only one in 10 to one in 30 voters willing to revote when they learned from the ESS PBC2100 receipt that they had spoiled their ballot. Consider that a person decides to commit fraud against a machine with a VVPT. Software could be designed to take advantage of the way voters seldom verify or, even less commonly, act on the information on paper receipts. If the software is designed to print the paper trail incorrectly, some will not notice that there is a problem. Additionally, a line of people will likely be waiting to use the voting machines, and the ballot workers are confronted all day long by people who consider themselves to be disenfranchised by the process so any genuine concern may not be addressed. In the first 10 minutes of watching people vote in LA County, I saw a person give up and decide not to vote because of the line and another person outraged by the procedure for voting when he was not found as a registered voter. Voters want to be helped inside the ballot booth. Voters want to take more time than allowed. Are poll workers able to distinguish these kinds of concerns and concerns stemming from a genuinely defrauded machine?

To defraud a VVPT machine a hacker might make the machine skip a race or appear to have a bad printer, perhaps by making the printer look like it's printing while it's not actually printing anything readable, or simply by making an unreadable section on the receipt. If this unreadable section is carefully printed it will be unreadable in a later recount. This could be used to cover up software defrauding of the electronic vote or it could hide changes in the vote inside the computer.

The vote inside the machine and the vote on the paper could be made to agree or disagree with the electronic vote. In making the VVPT and electronic ballot disagree, the defrauder could be calling into question the quality of technology to create a reason to call for a new election.

In a more likely scenario, the defrauder will change the electronic ballot and depend on the statistics for reading and contesting bad receipts. If a person calls their receipt into question and asks for another receipt to be printed, the hacked VVPT machine can print the "duplicate" receipt correctly, fixing the mistake. By printing the correct receipt when a person asks for it a second time it could literally eliminate the changed ballot, thus eliminating the possibility of detection. Although the program has to give up this one changed ballot it won't happen often. If this follows the experience described above, only one in three to one in 30 people that see a problem will be willing to do something about it. A hacker changing one percent of votes could count on between one in 300 and one in 3,000 voters who see a problem wanting to do anything about it. Considering that up to 1/3 of the fraudulent receipts would be noticed, the hacker has to change one in 75 votes to get a one percent change in the outcome.

If everyone reads their paper receipt carefully, one out of 225 people might notice that their paper receipt is different from their vote. The natural thing is to have the printer reprint it. In a precinct voting 500 people, this will be noticed twice during the day. When a voter complains and it comes to the attention of one of the several ballot workers that are running the election in a balloting area, it is likely to be caused by the ergonomic problems described above.

If it is because of the fraudulent VVPT, it will likely be the first time the ballot worker encounters this problem, which will make it harder to handle correctly than if they encountered it often. They are likely to encourage the voter to reprint the receipt that would, as outlined above, allow the voting machine to fix the internal count and print the correct receipt to cover up the fraud. If the ballot worker does enter the balloting area where the voter is, in order to verify the legitimacy of a problem with a VVPT, then they would have compromised the secrecy of that ballot. Even if they did enter the voters balloting booth to observe the strangely printed receipt, the natural reaction to an unreadable receipt would be to print a duplicate receipt themselves. Exchanging printers would also reprint the ballot, thereby eliminating the evidence. Shutting down the machine is the only thing that would preserve the fraud to view later, but this would disenfranchise other voters.

As described above, a printer can fake printing problems to cover up changes to the electronic and physical records. By doing this, it can introduce fraudulent tallies. Another way for software to defraud the paper trail is to print more receipts than voters. This could easily be seen as a mechanical problem at the time.

Mechanical problems with VVPT

Voting experts have been concerned about VVPT printers having problems. For instance, the connection between the printer and the machine can be broken, which would stop the printer functioning, and would keep people from being able to vote. If the printer were in the same unit as the voting machine, this problem might be lessened. Unfortunately, that would mean that the voting machine itself would have to be serviced to service the printer. Still it is a separate subsystem and would reduce voting machine reliability.

A printer can break mechanically— the motor, the levers or the solenoids can stop working, for instance. A plug replacement printer could be available, but the problem with the plug replacement printer is whether or not it can pick up where the other one left off. Has one ballot been lost in the meantime? Are we inserting a ballot accidentally when installing a printer? The person replacing a part can read the receipt because it is voter-verifiable. If they do change the paper, do they have access to the printout?

Additionally, the ink can be dried up or run out. If all printers are given new supplies preceding the election and tested, this should not be a problem. However, ensuring that such procedures include signoffs and checks of ink expiration dates is crucial to eliminating ink problems. If the printer is thermal (as many voting equipment printers are), the ink can't dry out. The problem with thermal devices is that heat applied to the paper before or after the election can destroy the printing. Thermal printing also fades with time and the paper tends to deteriorate more quickly.

These issues of printer failure might seem to be minor, but when considering LA County in which 2.2 million people vote in one day, the implications of mechanic problems that can occur are gigantic. In order to add any system that will not increase spoiled ballots, it must not add errors to the system. For the additional paper receipt to complicate the voter experience it must not misprint, jam, run out of paper or ink, malfunction, break, or loose its connection in a way that compromises the secrecy, integrity or accuracy of the vote.

To not lose votes, the printers must be shown to be able to print without failure during a voting election. Each printer must be able to print a typical precinct ballot every election for its planned lifetime. The number of voters in a precinct would not likely be more 200 voters per machine per election. General and special elections typically occur not more than 5 times a year. If the printer it to be used for 10 years a calculation of 15 years of life gives that it should be able to print 15,000 ballots without breaking.

The chance of breaking as opposed to wearing out is different; no machine should break down the day of election in a way that could lose a vote. For LA County, printers would have to have a reliability test that would ensure that they have a mean time between failures that is much larger than 2.2 million.

Alternatives to VVPT

The possible means of improving the authenticity and reliability of software are many. First, better methods for better software development can easily be applied to voting. Modular architecture that separates the different parts of the machine and makes it possible for them to be tracked separately is a good approach. Encrypted votes could improve the validity of the system. Allowing everyone to view the computer program as "open source" is a fashionable approach to ensuring that simple problems in it are not evident.

The “votemeter,” is a separate system that allows the voter to observe the vote without changing the software. To the extent that a votemeter is written by a separate set of people that have no communication with each other, they cannot be in conspiracy to defraud votes. This separate verifying computer can also present the data in exactly the same format as the voting machine. This allows people to compare their votes with a record of those votes in the same format. It can be enhanced by special optics that overlay the two images of the two different displays. Such a votemeter system can easily be verified and work across disabilities. The most exciting improvement of votemeter over verified paper trails is that reading it is easy, doing it is easy, and establishing its separateness is easy. By solving all of these problems the votemeter can literally eliminate the problems of setup and teardown. It can recognize the problems of voting, and establish authentic and separate verifications of the ballot.

Another verification approach is Voter Verified Audio Transcripts (VVAT), which speaks the names of the selections into earphones as selections are made. One advantage of this system is that receiving feedback while a person is making selections is easier to verify than a ballot later. Also, the tape that it produces is easy to count and has better integrity than receipts in a ballot box. Such a system can be implemented with the audio hardware available in today’s DRE voting machines.

In the future, many other approaches for establishing verification and audit of votes are possible. Systems could have multiple pieces of software checking each other or multiple computers could verify each other’s results. The most exciting of these is a voter's ability to compare his or her vote with the vote stored in the database of the government before they leave the voting booth. This will, in fact, some day be possible. When this is possible not only will we have a qualified belief that the vote this person cast is the vote that is stored in the computer, but we will also have deep security and the knowledge that what occurs at the very front end of the computer in establishing voter intentions is carried through, not only from the registration and authentication, marking the ballot, recording the ballot, storing the ballot, but also to recording the ballot in the election as it is being counted.

We can begin by verifying the votes on parallel machines. Parallel voting consists of pulling a voting machine out of service at random and assigning it to a phantom precinct. By controlling the votes that are cast and checking the results it collects, the machine can show that it recorded them as they were cast, ruling out an extra computer program, a “Trojan horse”, “Easter eggs” or other fraud. The voting machine is then used in a real election as a test of its ability to count votes correctly on the day of election thereby establishing the quality of the machines.

Conclusions

This paper shows there are many different ways of disenfranchising a person using a voter-verified paper trail. First, people can be disenfranchised in all the normal ways. They can have registration problems; they can have valid design problems, polling place problems, etc. Second, the paper trail can be lost, stolen, or added to. Third, the equipment can be designed or accidentally set up so it doesn’t work, or it slowly changes itself. Finally, intentional fraud can be widespread and created in software in such a way that it can be hidden from the voter and from the ballot worker on the day of election and not be remedied later. The final problem is that counting paper cannot be done at the accuracy level that electronic counting can be done. In this way, even if everything is performed correctly, the difficulty of counting the paper electronically will make it impossible to compare electronic outputs with the paper outputs in a way that can determine whether an accurate count has been achieved.

The Voter-Verified Paper Trail discussion has diverted attention from the main sources of lost votes in past elections. The majority of votes are lost because of problems of registration databases, ballot design, and polling place operations. The force of this discussion is even diverting voting technology development away from improving voting computer architecture. The Voter-Verified Paper Trail has blocked us from establishing standards for improving voting equipment.

Furthermore, VVPT complicates two of the top three problems that have compromised more than one percent of American votes in 2000: equipment problems and polling place operations. It complicates the setup, teardown, and operations of the ballot place. It complicates polling place procedures during the vote. It gives extra and difficult tasks for a person to do and increases the problems with the user experience and the user interface. It also increases the length of time of voting, which makes it, with more steps, easier to make mistakes.

The goal of Voter-Verified Paper Trail—that of establishing a second set of eyes to look at the intentions of a voter—is a worthy one. In fact, ballot design and voting have always been improved by more people looking at the process. In every case improvements in voting have occurred when one person cannot make a decision that changes the vote of another. The idea of establishing a way of doing that is valuable.

We call for improved research in voting technology and for heightened concern over spending large amounts of money on a short-term solution to software hacking problems that have not yet surfaced in elections. Instead, let us focus on verifying the votes in many ways and improving the quality of the whole system.

Citations and References

1. Michael R. Alvarez, Steve Ansolebehere, Erik Antonsson, Jejoshua Bruck, Steve Graves, Thomas Palfrey, Ron Rivest, Ted Selker Alex Slocum Charles Stewart III, Voting - What is, what could be., Caltech/MIT voting project, July 2001

2. Eric Fischer, Election Reform and Electronic Voting Systems (DREs); Analysis of Security issues: CRS Report for Congress, Order Code RL32139 Congressional Research Services Library of Congress, November 4, 2003. http://www.epic.org/privacy/voting/crsreport.pdf

3. http://fecweb1.fec.gov/hava/hava.htm

4. Roberta Klatsky, Human Memory Structures and processes, Second Edition, W. H. Freeman, San Francisco 1980,

5. Rebecca Mercuri, “A Better Ballot Box?” IEEE Spectrum, Volume 39, Number 10, October 2002.

6. Roy G. Saltman, Accuracy, Integrity and Security in Computerized Vote-Tallying, National Bureau of Standards Special Publications 500-158, August 1988.

7. Roger N. Shepard, Recognition memory for words, sentences and pictures. Journal of Verbal Learning and Verbal Behavior, 1967, 6, 156-163. (a)

8. Michael Tomz , Robert VanHouweling,“How Does Voting Equipment Affect the Racial Gap in Voided Ballots? *American Journal of Political Science* 47, no. 1 (January 2003): 46-60.

Denial of Service

Robert J. Fleischer

September 30, 2005

Taxonomy

Selective disenfranchisement; denial of service

Applicability

Any polling place organization that has a software-controlled "bottleneck".

Method

The perpetrator causes the "bottleneck" system (it could be a DRE or an optical scan machine) to slow down, stall, or crash. This could be achieved through any software intrusion or "hacking" technique. Long lines will form of waiting voters. If the waiting time gets too long (determined for each individual voter by their obligations, such as job or family, or in some cases the voter's physical condition) some percentage of voters will leave the queue without voting.

This bottleneck slowdown may be implemented selectively in precincts whose known demographics or politics favor the opponent of the perpetrator. Alternatively, the attacking software might simply observe the actual voting pattern and implement the slowdown only when the count so far favors the opponent.

Resource Requirements

The perpetrator must have the opportunity to introduce a software intrusion into the bottleneck system. Other than possibly observe the count to identify an opponent-favoring location, all the software need to be able to do is slow down or crash the machine.

The software intrusion can be introduced through communication lines, memory devices, or it can be embedded in the software as an act of sabotage any time before election day.

Potential Gain

It would take some simulation, based on a lot of assumptions, to quantify this. Waiting times of several to 10 hours were observed in the last general election.

Likelihood of Detection

People usually accept computer crashes or slowdowns. To most people, nothing will seem amiss when a computer slows or crashes. The intruding software can easily cover its tracks, and reloading the software will usually clear any traces. There were a lot of reports of crashes in the 2004 election -- nobody seems to think much about it. The count is not tampered with in this attack. Voters are merely deterred from voting. The counts are "correct".

Countermeasures

Preventative Measures

Physical and communication must be so good that software intrusions are impossible. Since software tampering can be introduced even at the factory, this level of security may be impossible.

Detection Measures

Detection is difficult -- when would crashes or slowdowns be obvious? In the 2004 election, there were precincts whose waiting times varied widely, from under an hour to many hours. Did anybody take that as proof of anything?

Techniques for detecting software changes, such as checksumming, can help detect changes introduced after the checksum was calculated.

Citations

(I need to find some references to the many reports of slow downs and crashes in 2004 and other elections.)

Retrospective

The "dark side" of "turning out the vote" activities is action taken to discourage votes for the opponent. One problem with electronic voting systems is that they can introduce new bottlenecks into the voting process at the polling place. DREs are especially bad in this respect. A single voter occupies DREs for an extended period of time while the voter reads the ballot and marks their choices. However, due to cost, DREs are typically in shorter supply than, say, booths for marking paper ballots. The occasional failure of a DRE, and the need to restart it (or simply take it out of service) aggravates this bottleneck and causes longer lines.

CALTECH-MIT/VOTING TECHNOLOGY PROJECT

VTP

Precinct Voting Denial of Service

Prepared for NIST "Threats to Voting Systems" Workshop

R. Michael Alvarez
Caltech/MIT Voting Technology Project
October 5, 2005

This is a type of threat that has a long history in electoral politics, and can take many forms.[1] The basic approach is that a perpetrator attacks precinct voting, regardless of voting system, on election day in an effort to disrupt the process sufficiently to produce an effective "denial of service" attack. The perpetrator, based on an analysis of past elections returns, would target selected precincts that are highly likely to cast votes in a certain direction. For example, if the perpetrator wished to sway the election for party X, he or she would target precincts that have very heavy concentrations of party Y supporters. In a close election, especially in lower-level races, such an attack could either sway the outcome of an election to party X or could throw considerable doubt and distrust into the announced election outcome.

Such an attack could be mounted in a wide variety of ways. The perpetrator could attempt to mount some sort of disturbance at certain critical times on election day in selected precincts. For example, in a high-turnout election where there are long lines of citizens waiting to vote just before polls close, the perpetrator could stage a protest at the entrance of the poll site; while such a demonstration is likely to be illegal if sufficiently close to the polling place, again if it either led some number of citizens to turn away and

[1] Bensel writes about political clubs in Baltimore during the 1850's, and he described one example of this threat: "The usual tactics used by these clubs on election day entailed the occupation of the area in front of the voting window by dozens of their members. Would-be voters were then forced to make their way through the crowd in order to hand their tickets to the election judges. As they moved through the crowd, club members would insist on seeing the ticket they wished to vote. If it was the American ticket, the crowd would part ranks, making an open path to the window. If it was the Democratic or "reform" ticket (a euphemism for the Democratic ticket), a cry would go out, alerting other club members that a member of the opposition was attempting to vote … This was the signal prompting a general movement of the members in mass, outward to the street. The would-be voter was thus physically moved away from the window by the sheer bulk of the crowd" (Richard Franklin Bensel, The American Ballot Box in the Mid-Nineteenth Century, Cambridge University Press, 2000, pages 171-172).

not vote, or led to the perception that some number of citizens were not allowed to vote, many might question the integrity of the election.[2] Or, the perpetrator could coordinate sending confederates to certain polling places to intimidate potential voters, similar to the historical example cited above.[3]

Other ways such an attack could be mounted might be more difficult to monitor and prevent. For example, the perpetrator could threaten the operation of the polling place by sending operatives to somehow disable or cripple the voting devices (or a sufficiently large subset of the voting devices to generate confusion and long lines) early on election day, thus leading to long lines and potential disenfranchisement of voters later in the day. Or they could disable or cripple the voting devices near the close of voting on election day, producing long lines and potentially disenfranchising voters who might be told they were not allowed to vote after the close of the polls (or who grew frustrated and leave). The method of such an attack would depend on the voting system in place, and the perpetrator's ability to coordinate a number of confederates to assist in the attack.

Such attacks could be mounted as insider attacks. For example, in 2002 the U.S. Department of Justice resolved voting rights complaints in two Florida counties; the complaints alleged that one county "had not translated all of its election documents and information into Spanish, failed to assign a sufficient number of bilingual poll officials to polling sites with significant numbers of Spanish-speaking voters, and denied some voters assistance from persons of their own choosing."[4] If the perpetrator were able to recruit conspirators to assist in such denial-of-service attacks in such ways, such attacks could be mounted and could be difficult to prevent on election day, or even in the immediate aftermath of the election. They could result in considerable voter disenfranchisement, and could again cast doubt on the integrity of the election outcome.

Of course, such an attack could be mounted in a much more coordinated way, by a highly motivated and well-resourced perpetrator. For example, a highly-motivated and well-resourced perpetrator could attack infrastructure on election day, in a number of ways that either could directly disrupt the voting process or which could indirectly serve to distract or disenfranchise voters in certain areas of a jurisdiction. A perpetrator in such a scenario could disrupt utility service to some targeted part of an election jurisdiction (again the perpetrator could attack utility service in a part of the jurisdiction that has a high concentration of party Y supporters, thereby distracting or disenfranchising such

[2] While laws exist to help prevent direct voter intimidation or electioneering close to the entrance to polling places, one could imagine that a sufficiently large public demonstration near a polling place could serve as a distraction or effectively block access to the polling place, disrupting service for otherwise eligible voters at that polling place, for example, by blocking access to parking lots or by making access difficult from local surface streets.

[3] For modern allegations of voter intimidation tactics, see the report from the People For The American Way Foundation, "The Long Shadow of Jim Crow", http://www.pfaw.org/pfaw/dfiles/file_462.pdf (last touched October 5, 2005).

[4] See http://www.usdoj.gov/opa/pr/2002/February/02_crt_380.htm for details of the allegations and the consent decree.

voters if the attack prevents or distracts them from voting), or across a series of election jurisdictions.[5]

Also, perpetrators could mount highly effective denial-of-service attacks on election day in precinct voting if they could mount successful pre-election attacks (either insider or by other means) on election administration systems. An example here would be a pre-election attack on a voter registration file (either at the local or state level). The perpetrator, with access to an electronic voter registration file with associated voter history data could effectively disenfranchise certain types of voters (say again those highly likely to cast ballots for their opponent, determined from either their partisan registration or voting history information) by altering their registration status, changing registration information, or perhaps altering records like early or absentee voting status in the current election. Eligible voters showing up to vote in the affected precincts would find their names not on the voter registration list, or be told they had already voted, either directly disenfranchising those voters or causing significant disruptions and long lines.

The resources needed to mount these attacks vary with their planned scope. In closely contested local elections, the perpetrator might need to effectively disrupt polling place operations in a single precinct, if their opponent's supporters are highly concentrated in that precinct, to potentially keep even a handful of the opposition's supporters from having the opportunity to vote. Effective denial of service attacks, mounted in different elections (say legislative races) would require more resources, primarily requiring that the perpetrator recruit and coordinate the activities of a greater number of confederates. Or, a well-resourced perpetrator could attempt a denial-of-service attack, as noted above, without many confederates by targeting infrastructure.

As noted a number of times, these attacks can have two consequences. One direct consequence is the disenfranchisement of a selected set of targeted voters, who have been prevented or discouraged from voting. An indirect consequence is doubts raised about the integrity of the election outcome. There are some mitigation strategies for the direct effect, including extending polling hours, allowing impacted voters the right to quickly and easily cast provisional ballots in another polling place, or in the case of a broad attack, holding another election.[6] The most problematic aspect of any denial-of-service attack, however, is the threat to the integrity of an election. Thus, even low-level denial-of-service attacks, occurring in a hotly contested election, might pose a substantial risk.

[5] Some of these scenarios have been explored by John C. Fortier and Norman J. Ornstein, in their Election Law Journal, "If Terrorists Attacked Our Presidential Elections" (Volume 3, Number 4, 2004, pages 597-612.). See especially the section "The Disruption of Election Day", pages 601-604.

[6] These mitigation strategies might be ones that some election officials may have planned for, but in the context of natural disasters. However, if some type of denial of service attack were undertaken that affected a number of election jurisdictions, it might be difficult to quickly coordinate a response on election day that might alleviate potential voter disenfranchisement.

CALTECH-MIT/VOTING TECHNOLOGY PROJECT

VTP

Potential Threats to Statewide Voter Registration Systems

Prepared for NIST "Threats to Voting Systems" Workshop

R. Michael Alvarez[1]
Caltech/MIT Voting Technology Project
October 6, 2005

The Help America Vote Act (HAVA), passed in 2002, requires that states implement "… a single, uniform, official, centralized, interactive computerized statewide voter registration list defined, maintained, and administered at the State level that contains the name and registration information of every legally registered voter in the State and assigns a unique identifier to each legally registered voter in the State." Many states are now rushing to meet these requirements by January 1, 2006, and by the time of the November 2006 federal elections it is likely that virtually all states will have their statewide voter registration system operational.

These new statewide voter registration systems pose new risks for election administration, for a number of reasons. In most states voter registration processes and data prior to HAVA were primarily a local activity, controlled by a local jurisdiction, typically a county election official. Such decentralization meant that effectively in most states there multiple voter registration processes and systems, and that mounting a systematic attack on the voter registration process in most states implied attacking a variety of different voter registration systems, operating in many different locations, using different types of hardware and software, and so on. The post-HAVA reality in most states will be a single centralized system, and thus, a single place where attackers can focus their energies.

One critical problem regarding threats to statewide voter registration systems is that there are no existing standards for these databases, nor is there a corresponding testing and certification process to insure that the databases comply with these standards. Here I offer some analysis of potential threats to statewide voter registration systems, which might help fuel further analysis and discussion of the development of standards, testing and certification for HAVA-compliant statewide voter registration systems. I

[1] Thanks to Ben Adida, Doug Chapin, Jr., and Ron Rivest for very helpful comments.

conclude that one important way to mitigate some of these risks is through the development of standards, and that we clearly need close study of statewide voter registration systems as they are implemented in 2006.

The threats to statewide voter registration systems fall into four categories: authenticity of the registration file, secrecy of the registration file, integrity of the registration file, and potential voter registration system failures.[2] I discuss each in turn briefly below.

Authenticity of the registration file

A first threat to authenticity of the statewide voter registration file arises due to the centralization of the voter registration list. The new centralized statewide voter registration systems required by HAVA will involve some form of data transfer between the local election officials, who in many states will retain some responsibility for the voter registration data and who will need the voter registration data for a wide range election administration tasks. This means that these statewide systems will involve voter registration data being passed from state to localities, which implies new points of vulnerability --- during the data transmission process and in the local election office. So while there is a centralized statewide list, it is possible that attackers could isolate points of vulnerability in the transmission path, or in one of many local election offices and possibly access the state list via local vulnerabilities that might be outside the direct control of state election officials.

Second, the statewide voter lists will be interactive with other databases, as required by HAVA, in particular state Department of Motor Vehicle and Social Security Administration databases. Again, the statewide voter data will be transmitted for comparison to those lists, and thus again be potentially vulnerable in transmission and when in places potentially outside the state election official's control. There has also been much talk recently about potential interoperability of statewide voter registration lists between states, which depending upon how implemented again may open the door for new vulnerabilities not experienced in the former decentralized voter registration systems in place throughout most of the nation before the passage of HAVA.[3] Thus, these potential vulnerabilities imply that attackers could have access to voter registration information and the ability to alter that information or add entries to the file.

Secrecy of the registration file

There are potential privacy concerns with the new statewide voter registration lists. There will be a great deal of information in statewide voter lists, including voter addresses, birthdays, and contact information; voter history data; and other identifying

[2] Ben Adida suggested this useful framework.

[3] See, for example, the recent report by the Commission on Federal Election Reform, "Building Confidence in U.S. Elections" (http://www.american.edu/ia/cfer, last touched October 6, 2005).

information including either partial or complete social security numbers, drivers license numbers, or other state identification numbers. This data could be of great use for commercial purposes, or for other more nefarious purposes (identify theft, stalking, or other illegal purposes). On the other hand, we clearly desire that voter registration information, at least at some level, be available for use by academics, political organizations, and other observers of elections to allow for external scrutiny of these data to insure the databases have high integrity. Thus a balance must be struck, between the need to insure the privacy of the centralized statewide voter registration list (especially elements in that file that might be attractive for identity theft), and the need to allow public access to voter registration data for external analysis and review.

Integrity of the registration file

Prior to the development of statewide voter registration lists, responsibility for the voter registration files typically resided at the local level. With the state-centralized voter registration files under HAVA, it is unclear how responsibility for the integrity of the information in the files will be distributed between state and local election officials. If much of the responsibility for the information rests at the state level, which might make the job of verifying local registration status more difficult than if local officials controlled the information. If the responsibility is somehow shared between the state and local levels, the possibility arises that the voter registration data could be corrupted if file updating is not done correctly. These threats to the integrity of the new statewide voter registration lists need further examination, especially as new state systems are implemented.

Potential voter registration system failures

These threats run from unintentional system malfunctions to malicious attacks. For example, we have all experienced computer failures of various sorts in our experience; centralized statewide voter registration files should be implemented using systems that seek to minimize these failures and which will prevent data loss or corruption when system failures occur (this is an example where standards would be very helpful). At the other end of the spectrum would be a general "denial of service" attack on a statewide voter registration system, where the attacker would attempt to make it difficult or impossible for local election officials to access the statewide voter registration list immediately before, during, or after the election. There is thus need to study these risks and vulnerabilities and to insure that voter registration systems are robust and hardened.

Need for understanding the threats to statewide voter registration files

Unfortunately, unlike the technologies that are used for ballot casting and tabulation, the technologies that are being put in place to satisfy the HAVA requirements have not been developed necessarily consistently with any national or state standards, nor with any necessary state or federal testing and certification process in place. While the current state and federal testing and certification process for ballot casting and tabulation technology is not perfect (in fact the federal Voluntary Voting System Guidelines are now under revision), the state and federal processes now in place do provide some level of assurance that certain standards have been met. We simply do not necessarily have that level of assurance for the new statewide voter registration systems that will be in place throughout the nation after January 1, 2006.

While this analysis of potential threats is by definition somewhat vague, because either the statewide files are not operational yet or they have not been operational long enough to determine in more precise detail their vulnerabilities, there is reason for analysis and study of attacks on statewide voter registration systems. The incentives to attack a statewide voter registration list are great:

- An attacker could, with access to the statewide list, engage in various types of election fraud. The attacker could register fictitious voters, and could attempt to cast ballots using the fictitious via by-mail absentee voting. This could be very difficult to detect, if done as part of a careful and sustained attack on the voter registration system.

- The attack could instead focus on disenfranchising registered voters, effectively mounting a "denial-of-service" attack on precinct voting. With access to the statewide voter list, the attacker could potentially remove voters from the list, move them to inactive status, alter their address information --- or do any number of things with the file to make it difficult or impossible for the voter to be allowed to cast a ballot when he or she tried to vote.

- The attack could be a "denial-of-service" attack on the voter registration system itself; if local election officials try to access voter registration data in the days immediately before or after an election, the attacker could mount a "denial-of-service" attack on the local officials computer system --- or the system where the statewide list is controlled. This could lead to significant disruption of early or absentee voting, election day activities, or pre- and post-election administration tasks. This risk could be mitigated somewhat by providing the voter registration data to the local officials before election day.

- A similar attack could focus on "electronic pollbooks", especially those that are used in precincts on election day. An attacker could mount a "denial-of-service" attack on a server that distributes voter registration data before the election to "electronic pollbooks", and thereby possibly cause a serious disruption in the election if voter registration data is not easily available in polling places.

- As noted earlier, the attack could focus on obtaining voter registration data for other purposes, either commercial data mining or identity theft (for two possible examples), especially if the attacker could access the database at levels where important data like drivers license or social security numbers are stored. But

> voter registration data, even without that sort of identifying information associated with it, could still be vulnerable to theft and inappropriate use, as there still are many purposes that voter registration data with names, addresses, birth dates, and other contact information could be used for.

These are just some of the potential threats to statewide voter registration lists. No doubt, as these files become operative and are used, other potential or real threats to these systems will arise. We clearly need more analysis of the security vulnerabilities of these systems as they are implemented in 2006 and beyond. We also need development of standards for these systems, and processes for testing and certification to those standards.

Malware Loaders

Ronald E. Crane, J.D., B.S.C.S.

December 11th, 2005

Taxonomy

Vendor or a vendor’s rogue employee.

Applicability

Any computer-based voting equipment, including DREs, DREs with VVPAT, ballot printers (“VVPB”), computer-based tabulators. etc. For brevity, this note concentrates on malware loaders in casting stations (DREs and the like).

Method

This attack allows the manipulation of vote totals, the alternation of ballots, or any other desired manipulation by facilitating the injection of malicious code into the voting application.

A malicious vendor (or a well-placed malicious employee of a vendor lacking sufficient internal controls and external supervision) places a small piece of code in its DREs’ video BIOS[1], such that it will be invoked regularly during ordinary machine operation. This “malware loader” polls a communications device (such as a WiFi or WiMax port, broadband-over-powerline (BPL) port, IrDA port, Ethernet port, proprietary radio receiver, etc.) within the DRE for a signal to begin cheating. The vendor or malicious employee arranges to broadcast this signal during elections in which it/she wishes to cheat.

When the malware loader receives the cheating signal, it disables CPU interrupts to prevent interruptions from the operating system or any other applications. Then it locates a small area of unused memory and copies the “malware bootstrap” into it.[2] After doing

[1] The code could be placed in the mainboard BIOS, in an FPGA or ASIC, or in the operating system. I have chosen the video BIOS because it illustrates the technique most clearly, and is more difficult to accomplish than placing the code in the operating system.

[2] This exact arrangement assumes a single-tasking operating system. On a multitasking OS, the malware bootstrap would first have to find the process housing the voting application, probably by parsing the OS’s process table. Then it would have to find an unused area within that process’s address space.

so, it locates the address of a function called periodically by the voting application.[3] This function could do anything or nothing, just so long as the voting application calls it relatively frequently. For example, it could be a function that flushes audit records to persistent media, animates a logo, updates the time on the display, etc.

The malware loader modifies the first few instructions of the periodic function to contain a jump to the malware bootstrap, clears the CPU’s instruction pipeline, re-enables interrupts, and returns control to whatever invoked it.

Eventually the voting application receives control, and eventually it calls the periodic function. When it does so, the periodic function jumps to the malware bootstrap, which knows all about the voting application and the communications device. It uses the communications device to load data and/or code (“cheating information”) to make the voting application do anything the malicious vendor or employee desires. Once the application is compromised, it can even subject the machine to realtime remote monitoring and control, enabling the cheater to detect and evade parallel testing.

The malware bootstrap remains in memory to supervise further cheating, possibly including replacing the compromised voting application with the original application after the polls close, erasing itself, or copying the cheating information to “unused” areas of persistent storage for later use.

Note that this cheat works even if the entire voting application and operating system are publicly reviewed, found completely honest, and are properly and honestly loaded into the voting machines.

Resource Requirements

The cheater can be a vendor, a well-placed employee of a vendor lacking sufficient internal controls and/or external supervision, an integrator, or any other actor who can control the contents of the machines’ firmware. Since firmware often is stored in dynamically-rewritable persistent memory (e.g., flash), a virus-writer, hacker, or anyone who can cause a program to be run on the machine to be compromised might also be able to emplace a malware loader.[4]

[3] The easiest approach is to use a function at a fixed address. By matching versioning information from the voting application (e.g., the fact that it printfs “Voting App. v.1.4.5” to the display on startup) with similar information from the communications device, a more advanced malware loader easily could look up the appropriate address for any voting application version. Of course this address has to reside in the voting application’s address space; see note 2.

[4] The Chernobyl virus caused its victims’ systems to “melt down” by erasing their flash-based BIOS firmware. http://vil.nai.com/vil/content/v_10300.htm. A more advanced virus could modify the firmware instead, perhaps emplacing a malware loader or other malicious software.

The cheater must be able to broadcast the cheating signal to the machines containing the malware loader, and must be able to follow it with the data and/or code that the malware loader expects.

Potential Gain

As many votes as the cheater wishes.

Likelihood of Detection

Very low. The VVSG do not require firmware inspections, and, even if they did, a malicious vendor simply would provide "honest" firmware to the inspectors, then ship machines containing malicious firmware. Since it is difficult, time-consuming, and expensive to inspect deployed machines' firmware, no one is likely to do so. Further, a crafty vendor could hide a malware loader in programmable logic, such as an FPGA or ASIC, that is also used to perform legitimate functions, such as controlling a video display. Such a loader is far more difficult to find than one hidden in an ordinary video or mainboard BIOS.[5]

Finally, since most elections are decided by small margins, and since exit polls have been subject to an extensive campaign aimed at discrediting them, it is unlikely that this cheat would be detected by monitoring election results.

Countermeasures

Preventative Measures

1. Prohibit all communications devices in voting machines. This approach, if well-enforced (it is difficult to enforce against a determined vendor[6]), makes it much more difficult remotely to monitor and control compromised machines. It does

[5] A crafty vendor might also consider using the capabilities provided by standard system-management and security firmware, such as that that supports Intel's Active Management Technology ("AMT"), http://www.intel.com/technology/manage/iamt/, to inject malicious code into voting machines. This approach cannot be detected by hardware inspections, since it does not modify off-the-shelf firmware. Instead it uses the very off-the-shelf firmware intended to help improve enterprise computer security to instead inject malware.

[6] Enforcement requires rigorous hardware inspections (i.e. rip to shreds) of a statistically-significant set of machines randomly chosen from the deployed base. See note 7. Further, communications devices are becoming smaller (and thus easier to hide) every day, and the trend will continue. Intel, for example, recently announced the development of single-chip WiFi. http://66.102.7.104/search?q=cache:PBxlV18sh3gJ:news.morningstar.com/news/DJ/M06/D17/200506170315DOWJONESDJONLINE000458.html+%22said+it+has+developed+prototype+chip+technology+that+can+handle+all+popular+forms+of+wireless+networking%22&hl=en.

not, however, prevent their compromise, since the cheating signal and following information can be loaded from ostensibly-unused areas of in-machine storage or from data cards used to record tabulated votes. The cheating information can even be loaded from image files and other data into which it has been steganographically encoded..

2. Require rigorous hardware inspections. This approach is a superset of (1). It involves regular random sampling of a statistically-significant set of deployed machines,[7] to inspect not only for hidden communications devices, but for malware loaders themselves.
3. Never run code from RAM. This approach makes it more difficult for the cheater to load the malware bootstrap and succeeding information into memory. It can, however, be worked around by loading code into RAM intended for data, or by housing more of the malware in the device housing the firmware.
4. Don't use electronic voting machines. This is the most secure approach. Hand-filled, hand-counted paper ballots are immune to this attack, and to many others affecting electronic voting machines.

Detection Measures

1. Parallel testing. Rigorously-conducted parallel testing of a statistically-significant set of randomly-selected machines should be able to detect the effects of some kinds of malware loaders.[8] Such testing probably will not be able to detect malware loaders that enable realtime monitoring and control of machines, since the cheaters may learn which machines are being tested, either directly (by knowing or observing the testing teams' schedules) or by observing the voting patterns on the entire machine base.
2. Voter verified paper ballots or paper trails. These measures enable the voter to detect the operation of malware loaders that transmute votes during casting: a significant advantage over unaided DREs. Note, however, that the proportion of voters who will verify their ballots or trails is unknown and is likely to decline over time, and the accuracy of their verification is unknown. Further, verification is not always meaningful. For example, a malware loader could generate marks on

[7] It is insufficient to sample machines that the vendor provides specifically for this purpose, since a determinedly malicious vendor will provide "honest" machines for this purpose, while deploying dishonest ones in the field. Similarly, it is insufficient to conduct the exam once, since an existing base of machines can be replaced, supplemented with new machines, or modified by firmware and/or hardware updates. And the set of machines sampled must be randomly chosen to prevent manipulation by the vendor or by others, and statistically-significant to ensure that it adequately represents the entire population of deployed machines. These procedures are lengthy, complex, expensive, and prone to shortcutting by vendors and elections officials. In consequence, they are likely to be ineffective unless experts from the general public have legally-enforceable rights and reasonable practical opportunities to supervise them. Even then, effective supervision may be spotty.

[8] As with all inspections involving voting machines, rigor is essential here. Any differences between the voting patterns that obtain during parallel testing and those during actual voting can be detected by sufficiently advanced malware.

voter-verified paper ballots that, while invisible to the voter, direct a cooperating tabulator to count her ballot differently from her intent. Or elections officials simply could fail properly to use a voter-verified paper trail. Finally, frauds that alter the presentation of the ballot to the voter (e.g., moving a disfavored candidate to the bottom of the ballot), or the manner in which her selections are accepted (e.g., making it more difficult to select a disfavored candidate), can influence the voter’s actual selections, particularly if she is among the many voters who decide how to vote in the voting booth. Since these techniques create no mismatch between the vote the voter casts and the vote the machine records, their operation cannot be detected by voter verification.

Election Official Training Improvements

Ann Hypes

March 15, 2006

Taxonomy

The threat of maintaining the Status Quo election official training will continue the current election process that:

a. Is unable to identify Election Day errors.
b. Opens the door for election fraud.

This threat will decrease Voter Confidence and Voter Turnout.

And in turn, will cause the American Election process to produce un-representative selections and make the American Election process less credible.

Applicability

The applicability is applied to all voting systems.

Attack Method

I. Current Election Official training has been adequate for many years. It is evident that there have been numerous hours, resources, and efforts put forth in training, reviewing, and hiring election officials. However, election processes are changing and it is time to update the core of our election process; our election officials training.

Currently Voters view Election Officers as ‘part-time’ non-authoritative poll workers. This causes high risk of opportunity for errors and voter fraud.

An Election Official is an Election Law Enforcement Officer. Not all election officials know the election law.

1) The election officer should enforce the election law by:
   a. Understanding the law.
   b. Having Rules and Responsibilities identified and assigned for each Election Officer.
   c. Enforcing the law as a TEAM in a professional manner.
   d. Having an election day GOAL Identified.

Election Day Examples of Issues:

a) With the many types of ballots; Provisional Ballots, Absentee Ballots, spoiled Ballots; over-votes and under-votes may be correctly or incorrectly counted or eliminated by confusion of the voter or election official.

b) Persons may be coming in to vote twice (name changes), or at multiple voting locations, or issued multiple ballots mistakenly given to the voter by an under-trained Election Officer.

c) Voting Booths may have propaganda left inside which may influence the next voter.

d) Election Officials put in long 5am through 9pm, or later, hours with little to no breaks. Fatigue will cause a lack of ability to catch voter discrepancies or potential voter fraud. There are provisions in the law to allow part-time election-day officers, not to just have officers that work the whole day.

Well-intentioned Election Officers sworn to uphold the election law, cannot be blamed for election-day errors or fraud if they are not given the training, authority, direction, and responsibility to act as a <u>law enforcement officer</u>.

<u>II.</u> Use of 3rd party professionals to track, audit, and report Election Day events. There is no one at this time looking at the <u>whole</u> election process from the outside (or inside) and coordinating with the Election Board.
If there are no errors found and no fraud currently reported, does this mean there is no errors or fraud? No, just that there is no coordinated effort to collect and identify anything at this time.

Resource Requirements and Costs

Election Officers properly trained with knowledge of the election laws, election-day tools, and defined responsibilities can be alert to many Election Day fraud attempts.
- Cost can be the <u>same as current training costs</u>.

A key resource to this concern is 3rd party professional audits. These professionals can perform spot checks on several precincts to identify continuous improvements that can be shared across many locations.
- Cost can be <u>determined by efficiency increases</u>; determining the best fit equipment for a specific precinct, using less number of persons to run an election, and by producing more accurate vote counts which eliminate the need for re-counts.

Consequences and Potential Gain

Destroying voter confidence and decreasing voter turnout will make the American election process appear less representative of the American Citizen, which in turn will give the American Election Process little credibility.

Just the appearance of positive change will change the Voter perception, increasing voter confidence and voter turnout. This will give tremendous credibility to the American election process.

Likelihood of Detection

The likelihood of detection depends on the degree of independent professional 3rd party audits performed and reports, generated through the Election Board, distributed to the public. Detection increases as more audits and continuous improvements are made. These audits will review **<u>all</u>** aspects of the election process...not just the equipment.

Countermeasures

Preventative Measures

Election Official Training Improvements will;

- Prevent Election Official uncertainty about the Election Law, which will give the election process credibility.
- Prevent unreliable vote counts, which will decrease the need for re-counts in close elections. The margin for error will decrease.
- Prevent voter uncertainty about ballots markings, which will make the votes reflect the intent of all voters. Election Day voter instruction will be clear and voters will become more confident about the voting process.
- Prevent poll-book errors, which will make the election process more efficient and voters lines move along more quickly. Poll-book officers will have a better understanding of the election law and reasons for the poll-book markings.

Professional 3rd party auditors will;

- Prevent errors and election fraud from the lessons-learned in audits taken from one precinct and shared with other precincts in the following training sessions.
- Prevent low voter confidence and turnout, because of concise communication by the Election Board from the positive audit results performed by independent professionals.

Detection Measures

Detection must be based on professional 3rd party audits to identify issues, report them, and coordinate them with the Election Board. Detection is most revealing when seen from the outside looking in, done in a professional manner, without judgment or accusation. Sticking with the facts, Parato charts can highlight areas of most opportunities and biggest impact for improvements. All reporting to the citizens would be handled through the Election Boards.

Citations and References

There are no current papers on modifying the election officer training program or including professional 3rd party audits as proposed.

There is a movement beginning to penetrate through the election community that will identify the need for this work. Good information is being accumulated in coordination with the Usability Professional Association (UPA) at www.usabilityprofessionals.org and the Election Assistant Committee (EAC). This is a positive step toward useful changes in the election process.

Retrospective and Historical Notes

The American Election Process is the best in the world.

As other countries' election processes are studied and assessed, there is similar equipment being used world-wide as in the US and not one voting system stands alone as the best. Standardizing the election process and election systems (equipment) does not appear to be a good solution. Some equipment works best in some precincts for specific reasons. Some ballots can be argued for similar reasons.

Voter Education, Election Officer Training, internal and outside audits will allow the Election PROCESS as a whole to be evaluated and continuously improved at the same time as cutting costs and improving efficiency of the equipment and the voters.

Stanley A. Klein

7 Lorre Court

Rockville, Maryland 20852

sklein@cpcug.org

301-881-4087

September 24, 2005

Position Paper on Voting System Threat Modeling

Voting is the most critical and fundamental process of a democratic society, a process from which "the consent of the governed" and thereby all governmental authority is derived. Voting systems must be required, designed, tested, operated, monitored, and certified to be reliable, accurate, secure, usable, and auditable. Systems should be so designed that errors and malfunctions are recoverable, that any malicious tampering is both detectable and recoverable, and that ordinary citizens are fully capable of understanding, observing, and knowledgeably participating in all processes and procedures necessary to ensure these attributes.

Voting systems have been clearly and repeatedly demonstrated to be seriously insecure and vulnerable to malicious tampering. We need to view the fraudulent takeover of government power by cybercriminal stealth just as seriously as we view the wrongful takeover of government power by force and violence. In a 4-year election cycle, roughly $2 Billion to $3 Billion is spent to influence the outcome of elections. If ruthless, unscrupulous interests diverted just a single digit percentage of that money to developing and executing technically sophisticated attacks on voting systems, the aggregate expenditure could exceed a quarter billion dollars. This is not just an abstract possibility -- some individuals, reporters, and researchers have alleged that attacks at various levels of sophistication have already affected results in recent Federal and state elections. Valid or not, the allegations are within the envelope of technical and operational feasibility.

Threat modeling is part of the technology developed over the past 30 years for properly protecting computer systems that includes the Defense Department "Orange Book" and the Common Criteria (International Standards Organization standard ISO-15408). NIST is a primary center of expertise in this technology. A threat model that could be used as a basis was provided in Section 5.1.2.3 of IEEE P1583 draft 5.3.2 that was provided to the TGDC. The text of that threat model is reproduced as Appendix A to these comments.

The threat model clearly states that governmental power is the asset requiring protection in voting machine security and that those attempting to compromise election integrity are likely to be highly motivated, technically expert, and well-financed. The potential pool of threat agents is identified as including personnel of voting machine manufacturers and their suppliers, election administrators, political operatives, and polling place personnel. Based on allegations of malicious tampering in recent elections, the threat agent pool potentially attempting to influence elections by cybercriminal stealth should be expanded to include law enforcement officials, former operatives of US and foreign intelligence communities, and organized crime.

The development of attack technology has economy-of-scope. Once a few million dollars are spent developing an attack on a particular voting system, that attack technology can be reused for the lifetime of the system in every jurisdiction where that system is used.

A proper threat model should address conditions over the intended lifetime of voting equipment. The lifetime expected by current purchasers of voting systems is likely in the range of 20-30 years. Accordingly a threat model should look forward to identify attack technologies likely to exist 15 to 25 years in the future. Examples of these threat considerations include:

- Reduced cost, increased capability, and increasing ubiquity of technologies for processing and exploiting compromising electromagnetic emanations. This issue is reflected in the threat model of Appendix A at item (d)(3).
- Reduced size, reduced power consumption, and increased capability of digital electronics. This will make it much easier to conceal attack equipment on the person of a voter or insider.
- Increased capability for attacking wireless systems, including optical wireless, at greater distances and with greater sophistication.

The threat model of Appendix A could probably serve as the basis of a wide variety of individual attacks. Appendices B and C provide outlines of two attacks: an attack through the smart card port (illustrating item d(5) of Appendix A), and exploitation of compromising electromagnetic emanations (illustrating item d(3) of Appendix A). The outlines are provided in the format requested on the NIST web page.

Appendix A

Threat Summary from IEEE P1583 Draft 5.3.2 (provided to EAC/TGDC)

5.1.2.3 Threat Summary

This section lists generic threats to which a voting system may be subject. It is, of course, not possible to enumerate all threats, but this establishes a lower bound on the threats that must be defended against.

Assumptions:

a. The persons who may be attempting to compromise the election process, and thereby the voting equipment, may be well-financed.

b. Given adequate unmonitored access there are motivated people who have the training and ability to compromise the election equipment.

c. The need for anonymity (where required by cognizant authority) of voter ballot reduces or entirely removes many traditional forms of auditing commonly used for other electronic systems (such as ATMs in banks).

d. Strong physical security is required to prevent unauthorized or unmonitored access during unattended storage periods.

e. For elections, the principal asset is governmental power. That power is transferred by the results of counting voted secret ballots. Hence, integrity of the voted ballot is critical through the entire process from capturing the voter's intent, casting it into the ballot box, counting it to produce the election results, and finally retaining it to resolve disputes.

f. The persons attempting to compromise the election process could be insiders with full knowledge of the election system including, but not limited to, political operatives, vendor personnel, polling place workers, or election administrators.

Threats: The principal vulnerabilities to the voted secret ballot are (1) undetected compromise of election integrity, (2) compromise of ballot secrecy, and (3) denial of voting service. All threats to voting systems can be classified under one or more of these vulnerabilities.

a) Software Development, Testing, and Distribution phase

1) A programmer embeds a backdoor or other software in a COTS product known to be potentially used in voting systems that enables malicious code to be later inserted into the voting system.

2) A programmer embeds code into the voting system software that directly or indirectly (such as by allowing later introduction of malicious code) allows one or more of the following to be done at a later time:

    i) Recording a ballot different from the ballot displayed and entered by the voter, either consistently or with intentional pseudo-randomness.

    ii) Modification of previously recorded votes or of vote totals

    iii) Causing a machine to become inoperable for further voting

    iv) Casting ballots that did not come from legitimate, authorized voters

    v) Observing recorded votes or vote totals prior to the time authorized.

    vi) Modifying audit trails

    vii) Identifying ballots cast by specific voters, with or without collusion of the voters involved.

    viii) Causing a machine to fail completely or to incorrectly record votes either generally or according to some logic.

    ix) Disabling features required for enforcing legal requirements of the ballot style or enabling features not permitted under legal requirements of the ballot style.

    x) Calculating vote totals inconsistent with legal requirements of a ballot style.

3) A vulnerability or other non-deliberate error in the development of a COTS product potentially used in voting systems enables malicious code or erroneous data to be later inserted into the voting system.

4) A vulnerability or other non-deliberate error in the development of a voting system has an effect similar to that identified in a-2.

5) Some systems are manufactured as to be subtly different from others such that malicious modifications can be made or deployed more easily.

b) Inter-Election Maintenance phase

1) An insider (election official or voting system technician) inserts malicious code into the software having an effect similar to that identified in a-2.

2) Someone who has illegally gained access to the voting systems (who is not an insider) modifies the devices. (This could also be true at any of the other points, but is most likely to happen during the months between elections where controlled access to the systems may be lax.)

c) Election Setup phase

1) An insider inserts code into the software and/or data into the election setup that causes item (a) of a-2 to be part of the election setup or to be introduced later and allows the remaining items of a-2 to be performed later.

d) Voting phase

1) A voter is able to insert malicious code or otherwise tamper with the voting device to cause or perform any of the items listed in a-2.

2) An insider is able to insert code or otherwise tamper with, e.g., adjust, the voting device or with any stored data that causes, performs, or allows any of the items listed in a-2 whether deliberately or inadvertently.

3) An eavesdropper is able to use compromising electromagnetic emissions to identify or modify ballots cast by voters, with or without collusion of the voters involved.

4) A voter, technician, poll worker or election official may be able to activate a Trojan horse or other malicious code that has been previously installed, in order to affect or manipulate ballot contents or vote.

5) An external device may be connected to the voting system through smart card or other external interface and allow unintended actions to occur.

e) Post election phase

1) Tampering having occurred with the voting system during the election, an insider is able to remove the tampering so it will not be detected.

2) Tampering is designed as to be self-removable such that it deletes any evidence of itself following its triggering or at the end of the election.

f) Data can be selectively activated and run as alternative code at any point in the election process.

Appendix B

Smartcard Port Attack

Taxonomy

Retail if performed by a voter or polling place official in the polling place. Wholesale if performed by an insider during or subsequent to machine setup.

Applicability

DRE voting machines using smartcards for voter authorization and other functions.

Method

By creating an appropriate interface, an attack on a voting machine can be based on software resident on another device. Modern cell phones and personal digital assistant (PDA) devices contain computers suitable for such an attack. An example of this kind of attack would be to penetrate the voting machine electronically through a smartcard reader port, often used in DRE machines for voter authorization. The device interface software that would be the focus of this attack is likely exempt from inspection under the provisions of VVSG Volume 1 Section 1.6 because of status as unmodified "Commercial Off-The-Shelf" software. Plans for an electronic device that connects a computer to a smart card reader port can be downloaded from the Internet (at http://www.electronics-lab.com/projects/misc/003/). An attack can be pre-programmed by experts, making it necessary for the attacker only to place a device into the smart card reader and remove it. The relevant electronics can be made easy to hide in clothing and the connection to the device in the smartcard port can be made by thin cable or optical wireless, making it very difficult for polling place officials to see that the attack is taking place. The attack could be perpetrated for various malicious purposes either in the polling place or during pre-election setup.

The external computer subverts an exploitable smart card driver and gains access to the voting machine memory bus. Programs on the external computer are then run to accomplish the purposes of the attack. For the retail polling place attack, this would be to “edit” previously cast ballots. Examples of wholesale (post-setup attack) purposes could be to maliciously modify the voting machine setups or to load self-deleting malicious software onto the machines.

Resource Requirements

This attack requires development of the smartcard emulation hardware, the interface to the external computer, and the attack software resident on the external computer. This development has economy of scope; once developed, the hardware and software can be reused in numerous elections. The cost of developing and producing the relevant equipment can probably be performed by someone with electronics expertise for an amount ranging from under $100 to as much as $1 Million depending on the sophistication of the interface (e.g. ease of concealment) and number of devices produced.

Also required are perpetrators to execute the attacks. For retail attack, these can probably be recruited and trained at low cost. An insider executing an attack at setup time would probably have to be bribed or otherwise induced to perform the attack.

Potential Gain

For the retail attack, all the votes on each attacked machine can be modified. For the wholesale attack, all machines in a jurisdiction set up at the same facility could be loaded with malicious software.

Likelihood of Detection

Depending on the sophistication of the design and the training of the perpetrators executing the attack, this attack could be extremely difficult to detect.

Countermeasures

Preventive Measures

1. Eliminate use of smartcards.
2. Provide means to disrupt any connection between the smartcard emulator and the external computer. (This can create an escalating “arms race” of increased sophistication in prevention and attack technology. For example, in the 1990's

European telephones contained cable cutters to prevent a similar kind of attack. Attackers countered by using thinner cables.)

3. Ensure that the voting machine operating system and the smartcard driver are not exploitable. This will require removing any “COTS Exemption” from all relevant software and conducting penetration tests of attacks through the smartcard port.

Detection Measures

None, if attack has sophisticated design.

Citations

Smartcard emulation attacks on telephone systems were described in an article appearing in 2600 Magazine in 1996 or 1997.

Retrospective

None.

Appendix C

Exploitation of Compromising Electromagnetic Emanations

Taxonomy

Retail, vote buying, or voter intimidation.

Applicability

DRE voting machines. Possible use against precinct-based optical scan tabulators.

Method

Perpetrator uses compromising electromagnetic emanations from voting machines to reproduce DRE screens in a vehicle near the polling place. Bought or intimidated voters are instructed to make certain combinations of selections and changes to enable the perpetrator to identify which voter is using which machine. Perpetrator watches the machine activity and ensures that voters vote as instructed.   This attack effectively returns voting activity to the conditions that existed prior to adoption in the late 1800's of the Australian Secret Ballot.

Exploitation of emanations from an optical scan tabulator would require either (a) the voter being instructed to vote in particular ways for offices/issues not of interest to the perpetrator, or (b) administrative records accessible to the perpetrator or an accomplice inside the polling place who can provide information on the sequence of voters whose ballots are being processed.

Resource Requirements

This attack requires development of software to monitor and process the compromising electromagnetic emanations. This development has economy of scope; once developed, the hardware and software can be reused in numerous elections. The cost of developing and producing the relevant equipment is likely to be in a multi-million-dollar range, but over time the relevant technology is likely to become ubiquitous.

The relevant technology may already exist and be in use within the intelligence community. The feasibility of exploiting compromising electromagnetic emanations from electronic equipment has been rumored since the 1970's. The Defense Department has long had a program called "Tempest" for minimizing compromising electromagnetic emanations from electronic equipment. Redacted Tempest documents were posted on the Internet a few years ago as a result of a FOIA request.

The technology requirements for accomplishing the attack are likely to include the following:

- High capacity software defined radio
- Digital signal processing and/or directive antenna technology (such as phased arrays) sufficient to separate individual voting machine emanations. For example, this might be done by using small differences in clock speeds or other processing hardware characteristics of the various machines.
- Digital signal processing to reconstruct the internal processing and screen displays from the voting machine emanations.

The software defined radio and high capacity digital signal processing technologies are currently available, although not necessarily at low cost and sufficiently small size to allow installation of the necessary facilities in a vehicle. These technologies at appropriate capacities, sizes, and costs are likely to become ubiquitous during the lifetime of voting machines in current service or currently being designed and purchased.

Perpetrators must also have access to a pool of subvertable voters willing to vote in return for payment or unable to complain if threatened. Employees, tenants, and those with similar dependency relationships are particularly vulnerable.

Potential Gain

One vote per subverted voter.

Likelihood of Detection

The likelihood of detection depends on the degree of dependency linking the perpetrator to the subverted voters.

Countermeasures

Preventive Measures

Apply to voting machines and polling places the Tempest technology and other measures used by the Defense Department for protecting against exploitation of compromising electromagnetic emanations.

Use only optical scan machines, and take measures to block the collection of information that could identify the sequence of voters whose ballots are being scanned.

Detection Measures

The attack can not be detected by technical or administrative means. The only possibility of discovering that it has occurred is if one of the voters reveals the existence of the vote buying or voter intimidation to authorities who are not themselves involved in the scheme.

Citations

None

Retrospective

None.

Threats to Voting Systems

Douglas W. Jones
University of Iowa

A position paper for the NIST workshop on Threats to Voting Systems
October 7, 2005, Gaithersburg, MD

Abstract

A public catalog of threats to voting systems should be created. While such a catalog may help educate attackers, it is essential to a reasoned public debate about the adequacy of our voting system standards, the adequacy of our recommendations for best practices and the adequacy of state laws and administrative rules. If we can quantify the costs of threats and defensive measures we will be able to rank order threats in order of their likelihood and defensive measures in the order of their importance, but such quantification will be difficult. We must be careful to avoid giving the impression that our threat catalog is complete, or that addressing all of the threats in the catalog is sufficient to absolve vendors or election officials from responsibility for the failures of their systems.

A Catalog of Voting System Threats is not a Threat

When asked about the vulnerabilities of their voting systems, many election officials will simply deny that their voting system has vulnerabilities. Others will refuse to answer, saying that discussions of this topic are inappropriate. The most frequently cited reasons for a refusal to discuss this subject are:

1) Public discussion of this subject could enable election fraud.

2) Voter confidence is essential to the legitimacy of elections, and public discussion of this subject is a threat to voter confidence; therefore such discussion is a threat to the legitimacy of elections.

3) After having spent millions of dollars on this voting system, a public admission that the system is less than perfect would invite questions about the propriety of this expenditure.

Curiously, the answer to the first objections was stated well over a century ago, in a book edited by Charles Tomlinson.[1] There, of course, the question was "whether or not it is right to discuss openly the security or insecurity of locks." The book offers the following answer:

> Rogues knew a good deal about lock-picking long before locksmiths discussed it among themselves, as they have lately done. If a lock, let it have been made in whatever country, or by whatever maker, is not so inviolable as it has hitherto been deemed to be, surely it is to the interest of honest persons to know this fact, because the dishonest are tolerably certain to apply the knowledge practically; and the spread of the knowledge is necessary to give fair play to those who might suffer by ignorance.

There is no doubt that rogues have been corrupting scattered elections across the

United States for two centuries. Joseph Harris devoted Chapter IX of his landmark 1934 book to this topic, clearly documenting numerous cases of fraud and providing a useful list of types of voting fraud.[2] Edmund Kallina's study of the 1960 election in Chicago shows that the kinds of irregularities documented by Harris continued with little change 30 years later.[3] While the technology has changed, and while we may be doing somewhat better today, there is no reason to believe that the rogues have lost interest.

Questions 2 and 3 rest on the same questionable ethical premise: That it is better for the public to remain ignorant of the shortcomings of their government or their voting system than it is to encourage open public discussion of such issues. While there may be some short term benefit of suppressing debate, in the long run, such suppression can only lead to an uninformed electorate making uninformed decisions. That is certainly a threat to our democracy.

Organizing a Threat Catalog

In any discussion of threats to voting systems, the list of possible threats can grow quite unwieldy. Even the short list of threats identified by Harris shows evidence of this (See Chapter IX of [2]):

1) Registration frauds.
2) Repeating (individual voters voting more than once).
3) Ballot box stuffing.
4) Chain ballots.
5) Voter assistance.
6) Intimidation and Violence.
7) Altering Ballots.
8) Substitution of Ballots.
9) False Count and False Returns.
10) Altering Returns.

Here, we find chain voting, a very specific and somewhat technical vote buying attack, listed on a par with voter assistance, a broad general category of attack. We can clearly sort the different approaches to election fraud according to several different criteria.

Before continuing with an enumeration of these criteria, it is worth noting the distinction between threats to a voting system and attacks against that system. In general, attacks are deliberate malicious acts, while the term threat is broader, encompassing accidents and mistakes.[4] An old maxim in the area of computer security is clearly applicable here: Almost everything that a malicious attacker could attempt can also happen by accident; for every malicious attacker, there may be thousands of ordinary people making ordinary careless errors. We are equally concerned by errors and by attacks, so we will use the term threat except where deliberate malice is necessarily involved.

What phase of the voting process is being manipulated. Most of Harris's taxonomy addresses this. Generally, an adversary can attack the system in one or more of the following phases:

1) Registration
2) Polling place access (intimidation, violence, destruction and vandalism).
3) Voter manipulation (repeat voting, chain voting, voter assistance).
4) Ballot manipulation prior to tabulation (substitution, stuffing, counterfiting).
5) Threats to the ballot tabulation process itself.
6) Threats to the results of the tabulation process.

All of the threats identified by Harris can be fit into this scheme, and if we set out to produce a master catalog of voting system threats, this appears to be a reasonable top-level organization for a threat taxonomy. An expanded version of this taxonomy is given in the appendix. There is good reason, however, to provide secondary indices into the threat catalog that support alternative taxonomies.

What technology is vulnerable. Certain threats are technologically neutral, while others target specific technologies. Configuration file manipulation can only be used to attack voting systems that have configuration files, while chain voting is only possible when voters are allowed to handle physical ballots.

Who carries out the attack. Everyone involved in the election, whatever their role, has an interest in the outcome, and everyone can make mistakes. While many people are involved, they can be classified into a few basic roles, and it is not difficult to identify, for each attack, the role of the initiator(s) and the roles from which participants must be recruited.

1) Individual voters.
2) Outside attackers, including hackers, precinct captains and others.
3) Polling place workers and other temporary election staff.
4) Permanent employees at the election office.
5) Election officials.
6) Equipment vendors.
7) Policy makers.

Matters of scale. Retail fraud involves small-scale tinkering, where a separate act is required for each illegally obtained vote. Most fraud committed by individual voters is in this category. Wholesale fraud is at the other extreme, where a single act can change the outcome of an entire election or even of all elections from then on. Adoption of discriminatory policies by the government represents the most extreme form of wholsale election manipulation, although the very word fraud is problematic in the context of immoral legislative acts.

Possible Refinements to the Threat Catalog

In its simplest form, a threat catalog consists of an enumeration of the threats to the voting system, with clear documentation of each threat. The description should be complete enough to allow evaluation of whether a particular voting system is adequately defended against that threat. In many cases, this level of completeness will not be sufficient to allow a potential attacker to carry out the threat, while in other cases, particularly for the nontechnical attacks, it will be difficult to avoid complete disclosure of the necessary details.

Many users of the catalog will need documentation, for each attack, of the defensive measures that can block or deter that attack. Some defensive measures are preventative, entirely blocking the attack if they are properly in place. Other defensive measures, such as post-election auditing, only allow detection of the attack. Some measures do not even guarantee detection, but merely create a risk of detection, and others merely raise the cost of an attack.

Some users of the threat catalog will prefer this simple presentation, where all information about a specific threat is consolidated in a single narrative description. On the other hand, some users of the catalog will notice that each individual attack or each error in the conduct of an election has structure. Each attack, or each error, involves the intentional or accidental exploitation of some set of vulnerabilities in the voting system. Many different attacks may exploit the same vulnerability.

Our threat catalog can be refined by identifying, for each attack, the set of vulnerabilities on which it rests, and then documenting the vulnerabilities.[4] Some attacks will rest on a single vulnerability, but others are more complex. Chain voting, for example requires obtaining a blank ballot, which may be done by exploiting any of a number of vulnerabilities, and then finding voters vulnerable to subversion, and then finding procedural vulnerabilities that allow those voters vote a different ballot than the one they were issued at the polling place.

By splitting attack descriptions from vulnerability descriptions, we can produce attack descriptions that are far more compact, but they will also be far less readable and they may be harder to produce. This suggests that the refined catalog should be a secondary document, but it is worth noting that the exercise of extracting vulnerabilities from attack descriptions can itself lead to the discovery of other attacks.

If we include defensive measures in our catalog, these can form a third section, since some defensive measures, such as various forms of auditing, defend against multiple vulnerabilities, while other defensive measures apply only to one. As with vulnerabilities, consolidation of the discussion of a defensive measure in one place will allow more complete discussion of that defense, but it also makes it more difficult for a reader to quickly determine which combinations of defenses will guard a particular voting system against a particular attack.

Using the Catalog

Threat catalogs can be used in a variety of ways. If we classify attacks according to the voting technology to which they apply, we can easily extract from our catalog, for any voting system, the set of attacks an adversary might exploit in corrupting that system. This, of course, could be used by an adversary to design their attack, but it is also the list of attacks an election administrator must be prepared to defend against. If the threat catalog includes defensive measures for each threat or vulnerability, we can use it to assess election administration at several levels.

Evaluating the defenses of a particular voting system. We can evaluate a voting system, as used in a particular administrative context, against the threats listed in our catalog. To do this, we take the set of all defensive measures that surround that voting system and ask if that set includes at least one defense that will block each applicable threat. If we are serious about defense in depth, we should ask that each applicable threat be blocked by more than one defensive measure.

It is important to note, here, that each defensive measure can be classified as having technical and administrative components. One defense against chain voting, for example, uses numbered tear-off ballot stubs (See Chapter II of [2]). These are a technical component. These stubs, however, are of no value unless the polling place workers use them, and that use is the administrative part of the defense. Thus, we can say that a particular voting system is adequately defended if the following conditions hold:

1) The voting system mechanism must incorporate all of the technical components of the identified set of defensive measures. This should be insured by some combination of the voting system standards, state certification, pre-purchase product evaluation and post purchase retrofits.

2) The voting system must be administered in a way that incorporates all of the administrative components for the same set of defensive measures.

Evaluating the voting system standards. Given a threat catalog and a set of voting system standards, we can ask, for each class of voting systems governed by the standards, do those standards require the technical components of the defenses necessary to adequately block the applicable threats.

If the standards do not address some threat, this strongly suggests a weakness in the standards. If the standards require mechanisms that do not address some threat, then it is possible that some threat has not been identified that belongs in the threat catalog, but it is also possible that the the requirement itself is wrong.

It is worth recalling that our voting system standards have been developed with strong vendor input. Sometimes, this works to everyone's benefit, since the vendors are in contact with many potential customers and are sensitive to the real needs of those customers, but at times, vendors may attempt to manipulate the standards to their own advantage, inserting requirements for the purpose of limiting the competition. A well managed attack catalog can help us ferret out these spurious requirements, defending the standards against regulatory capture.

Evaluating the laws and administrative rules governing the conduct of elections. Given a threat catalog and the laws and administrative rules of a jurisdiction, we can ask, for each class of voting system permitted in that jurisdiction, whether those laws and rules require the administrative components of the defenses necessary to adequately block the applicable threats.

This is perhaps the single most valuable use for the threat catalog. In 1934, Harris pointed out that the laws governing the use of voting machines were, to a significant extent, being written by the vendors (See Chapter VII of [2]). In many cases today, it is difficult to ascertain what these laws mean or why some feature is required. Given a threat catalog as proposed here, we have some hope of answering these questions and arriving at a rational basis for evaluating these laws and evading regulatory capture.

It is, of course, essential that the defenses selected have the necessary technical support! Currently, there is an almost complete disconnect between the technical voting system standards and the drafting of law and administrative rules to govern the use of voting systems, and this leads to some very odd results where mechanisms are required to be present that are not permitted to be used or where procedures are required that are ineffective because the necessary mechanisms are not fully implemented.

While the NIST, TGDC and EAC have no direct authority in the setting of the state laws and administrative rules that govern the conduct of elections, they do have the charge to examine and promulgate codes of best practices in this area. Such a code could take the form of a model code of election law, following the path that Harris took in 1934 (See Chapter II of [2]). The problem with this is that there is huge variation from state to state in the way voting systems are governed. In some states, statutes are general and all specific details are relegated to administrative rules, while in other states, almost everything is spelled out in statute.

It would be very useful if each edition of the voting system standards were accompanied by a checklist of the administrative measures that are assumed to be present to complete the implementation for each defense incorporated into the technical standards. This checklist could be used in any jurisdiction to determine if the local voting system laws and administrative rules meet the assumptions made by the voting system standards.

The Possibility of Quantitative Evaluation

In the above discussion, the basic measure of adequacy was completeness of coverage. Either the defenses in place for a particular voting system covered the set of threats listed in the catalog, or some threats were not covered. Defense in depth was discussed only in terms of counting the number of defenses that were in place to cover each threat. No basis was given for assessing the likelihood of different attacks, nor was a basis given for assessing which defenses should be used when more than one attack is possible.

Assessing the likelihood of an attack. If we can determine the cost of overcoming the defenses that are in place to guard against each threat, we can assess which attack to expect from a rational and well-informed attacker. For any particular voting system in any particular administrative context, we should expect the least-cost attack while we may be able to largely ignore the more expensive attacks.

The problem we face in doing this is arriving at an estimate of the cost of overcoming each defense that is in place. Cost can be dollarized, it can be estimated in man-hours of effort, or it can be estimated in terms of the number of people required. Some of these costs will be easy to estimate, for example, the cost of cracking a well-chosen password by trial and error, while others are extremely difficult to estimate, for example, how much it would take to bribe a key person. To determine the cost of a particular attack, we must determine the cost of overcoming each defense, and then navigate a least-cost path through the set of defenses to mount the attack.

The fact that so many of the costs are fuzzy poses a serious problem. We can confront this problem by using perturbation analysis. To do this, we vary the cost estimates for each component of the attack over the reasonable range of values for that cost, and then examine how this influences the overall result. Having done this, we can now describe the cost of each attack with a range of values, and as a result, we may have not just one minimum-cost attack, but a set of attacks that are each potentially the minimum cost attack. This is basically a Monte Carlo method, but we can also accomplish much the same thing analytically using fuzzy math.

Assessing the cost effectiveness of various defenses. There are many threats that can be blocked by several different defensive measures, and many defensive measures are effective against several different attacks. It is natural to ask, in this context, which defenses we should implement.

Consider, for example, the problem of improving an inadequate set of voting system standards. The resistance to any broadening of the standards will typically depend on the cost, to the election administrators, of the new defensive measures required by that broadening. In order to defend a proposed broadened standard, it would be nice to be able to demonstrate that, among the defenses that could have been required, the new defenses that were required are the most effective, in the sense that no other set of defenses with comparable costs raises the cost of an attack as much.

Demonstrating this will require not only reasonable estimates of the costs of each attack in our attack catalog, but also reasonable estimates of the costs of each of the applicable defensive measures. These estimates will likely be as imprecise as the estimates of attack cost because there are few good studies of the actual economics of elections. The cost of voting system software is extraordinarily difficult to assess, and accurate measurement of the costs of defensive measures taken at the polling place has only rarely been attempted.

Conclusion

The development of a voting system threat catalog offers some immediate benefits. If we can document the known defenses against each threat, we can use it as a tool for evaluating the laws and regulations governing both voting equipment and the conduct of elections to see if these threats are adequately addressed. This can help us in the evaluation of voting system standards, best practices documents, and much more.

If we can produce reasonable estimates of the cost of each attack in the catalog, we may be able to produce a useful rank-ordering of the threats we ought to be wary of. If, in addition, we can produce reasonable estimates of the implementation costs for each defensive measure, we should be able to conduct cost-benefit analysis of the different defensive measures. The value of these quantitative assessments will depend on the precision of our cost estimates. It seems likely that the best estimates we will be able to make will be imprecise, which means that we will be able to offer only rough rankings of the various attacks and defenses.

There is one very serious risk in publishing a threat catalog that has not been considered here: That the catalog might be considered complete, and as a result, vendors and government officials might be absolved of responsibility for defending against any threats not documented in the catalog. If our threat catalog ever grows to the point that it appears to be exhaustive, this will become a very real risk. Any published version of the threat catalog must therefore begin with a disclaimer and a warning that someone, somewhere, may be hard at work devising new attacks on the machinery of democracy.

Acknowledgement

I would like to acknowledge the helpful suggestions of those who have read early drafts of this paper, notably Ronald Rivest and Eric Lazarus. NIST and NSF grant CNS-052431 provided partial support for the author's participation in the NIST workshop.

Notes

[1] A. C. Hobbs, *Locks and Safes*, Charles Tomlinson, ed., Virtue & Co., London, 1853.

[2] Joseph P. Harris, *Election Administration in the United States*, The Brookings Institution, 1934.

[3] Edmund F. Kallina, Jr. *Courthouse over White House -- Chicago and the Presidential Election of 1960*, University Presses of Florida, 1988.

[4] This distinction is commonly made in computer security texts; see, for example, Section 1.2 of Charles P. Pfleeger and Shari L. Pfleeger, *Security in Computing*, Prentice Hall. 2003. IEEE Standard 729 introduces similar distinctions in the terminology for discussions of quality control.

Appendix: An Expanded Threat Taxonomy

The following threat taxonomy is an expansion of the taxonomy given in the body of this paper based on phases of the election process. It is, at best, a preliminary work, and will almost certainly need revision as a result of finding threats that do not fit cleanly into it. On the other hand, the exercise of building this taxonomic tree has itself suggested a number of threats which might have been difficult to identify without this effort.

- 1) Registration
  - 11) One person registering in multiple places
  - 12) Registration of non-voters (such as dead people)
- 2) Polling place access
  - 21) Intimidation to prevent voting
    - 211) Intimidation outside the polling place
    - 212) Selective challenges to "undesirable" voters
  - 22) Violence to prevent voting
  - 23) Vandalism to prevent voting
    - 231) Physical destruction of voting equipment
    - 232) Tampering with equipment
      - 2321) Tampering with hardware
        - 23211) Substitution of improper mechanisms
      - 2322) Tampering with firmware
        - 23221) Substitution of improper code
        - 23222) Easter-eggs inserted by corrupt programmers
        - 23223) Trojans inserted into third-party components
        - 23224) Code injection attacks
      - 2323) Tampering with election configuration files
        - 23231) Substitution of media prior to installation
        - 23232) Alteration of contents of proper media
- 3) Voter manipulation

31) repreat voting (note connection to category 1)
311) voting under an assumed identity
312) voting using illegal registration
32) chain voting
33) improper assistance to voters
331) improper instruction given outside of voting booth
332) improper advantage taken of voters with legitimate need for assistance
333) voter requests assistance in order to earn reward from assistant
4) Ballot manipulation prior to tabulation
41) ballot box stuffing
411) stuffing before the polls open
412) stuffing during voting
413) stuffing after the polls close
42) ballot alteration
421) alteration of individual ballots
4211) alteration prior to tabulation
4212) alteration during tabulation ("short pencil" methods)
422) substitution of counterfeit ballot box for authentic box
43) challenging the authenticity of legitimate ballots
5) Threats to the ballot tabulation process itself
51) announcement of tabulation result ignoring actual ballots
52) uneven criteria for accepting votes depending on who is voted for
521) threshold of acceptability depends on candidate
522) threshold of acceptability depends on polling place
53) incorrect counting
531) counter overflow errors
532) carry propagation errors
6) Threats to the results of the tabulation process
61) substitution of counterfeit data
611) substitution of counterfeit ballot box
612) substitution of counterfeit tabulation results
62) alteration of data
63) rejection of legitimate data

Developing an Analysis of Threats to Voting Systems
A Response by the Florida State Association of Supervisors of Elections

Introduction

The 2000 General Election has served as the catalyst for election reform throughout the United States. State and local governments have purchased and implemented a number of different types of voting systems, many choosing either precinct count optical scan or direct record electronic (DRE) touch screen voting systems. The implementation of these systems has caused state and local governments to reassess the threats to the security of these voting systems.

Security of voting systems has always been a concern of local elections officials but has taken on additional importance by the general public and the computer technology communities as these systems have been used in more elections throughout our country. It is important to note that security, as a technical term, means something is not only secure but that it has been secured.

Since more jurisdictions have begun to use new voting systems, there has been a huge effort to require these systems to provide a paper receipt, commonly referred to as a voter verifiable paper audit trail (VVPAT.) This plea has been primarily associated with the touch screen voting systems.

Florida elections officials have worked tirelessly to ensure that voting systems are accurate, reliable and secure. Florida officials believe the voting equipment, "the box," to be only part of the voting system. We firmly believe that people, policies and procedures are critical to the efficient operation of these systems. The "Three P's," as we refer to them, are often overlooked as being an integral part of the system. Because of this oversight, these systems have taken a great deal of the blame for "failure" when the truth is the "failure" was a result of inadequate policies and procedures and human error, not voting systems. These three elements are critical to the successful administration of elections and should be included in the discussion of threats to voting systems.

Thoughts on Assessing and Minimizing Threats to Voting Systems

The Election Reform Act of 2001, passed by the Florida Legislature and signed into law by Governor Jeb Bush, required all counties to use either a precinct count optical scan system or a touch screen voting system. Fifty-two counties are currently using an optical scan voting system. These counties, as required by the Help America Vote Act (HAVA), will be implementing either a touch screen voting system in each polling place or the AUTOMARK product, pending State certification, to accommodate voters with disabilities. There are 15 Florida counties that have chosen to implement touch screen systems throughout their counties. These counties, accounting for over half of all voters in Florida, are currently in compliance with the requirements of HAVA to accommodate voters with disabilities. This point is made to illustrate the fact that although a jurisdiction uses an optical scan system, many will be faced with implementing touch screen technology into their overall voting methodology.

Optical scan systems are "perceived" to be more secure and less of a security risk than touch screen systems because of the use of an actual paper ballot. But bear in mind, even optical scan systems use software for tabulation, and in most recount laws those ballots will not be manually recounted. Florida officials believe that all systems security should be analyzed using the same standards. However, one cannot intelligently compare or analyze voting systems without carefully examining how the technology is being implemented. Even with a VVPAT, testing, training and adhering to procedures is essential.

Florida elections officials believe that the discussion of the "Three P's" is even more important when discussing touch screen voting systems because of the perceived threats to their accuracy, reliability and security.

Local elections officials need to ensure that policies and procedures are in place that detail all steps and activities necessary to conduct an election. These policies and procedures need to go beyond being a document to meet some state requirement mandating elections officials to have "security procedures." These security procedures need to detail security of optical scan ballots used for absentee voting, the programming of the election parameters, the proofing and correction, if necessary, of ballot tabulation and collection parameters, the chain of custody of all election records and documents required to make the system election ready, among many others.

A copy of Florida Administrative Code, 1S-2.015, Minimum Security Procedures for Voting Systems, is attached as an example of steps taken to address security within the total process of elections administration.

The manufacturers of these systems are held to a much higher standard today than when these systems were first introduced into the market. This is a good thing. Local elections officials, generally through users' groups, have worked with the manufacturers to improve these systems. Additionally, we have worked with our State Division of Elections to address issues and improvements to these systems that the manufacturers have incorporated into their systems designs.

As previously stated, much more attention needs to be devoted to the training of local elections officials and their staffs. Local officials need to become "vendor independent," where possible. Local officials need to assume the responsibility of implementing, operating and maintaining these systems. You would no more want an untrained, untested elections official, and/or their staff, conducting your elections, regardless of the system being used, any more than you would want to fly with an untrained and untested pilot. Although this illustration may appear to be inappropriate, it demonstrates the importance of the need for training in a very specialized field and in a very politically volatile environment.

The need for competent and qualified people does not start or stop with the local elections official. The Florida Legislature has adopted more stringent standards for the recruitment and training of poll workers in an effort to minimize human error. The trend is that Florida is "professionalizing" their election day workforce.

Florida elections officials recommend that voting systems be designed to be independent systems, eliminating any networking of systems. Additionally, voting systems should not be configured in such a manner as to access the internet. This action would eliminate unauthorized access to the system from the outside.

Finally, as threats to voting systems are examined and addressed, this issue does not need to be confused with issues outside the realm of voting systems. It is a fact that these systems run on off the shelf computers with commercial operating systems. People all around the world use these computers and operating systems to transact business every day without incident. It is not logical to expect or require these operating systems to be included as part of a "voting system," making them subject to the voting systems standards. This argument only perpetuates the belief that these voting systems are incredibly complex and are unable to have malicious code detected.

Conclusion

Election reform and all its associated issues will and should continue to be "hot topics" for many years to come. Security of voting systems and the overall elections process needs to be continually reassessed and tested to ensure that our elections process is reliable, accurate and secure. Technology continues to change the manner and method of casting and counting votes. For this reason, local elections officials need to change the manner and method by which their entire system, "the box," people, policies and procedures, fit together to provide a seamless system that is not subject to outside or inside influences without detection.

There is a need to assess "real world" threats. It is important to note that just because something is possible it is not the same as saying it is probable. There has been no evidence of insertion of malicious code, attacks on individual machines at precincts or tampering with election results. Parallel testing in California and other jurisdictions has revealed that touch screen voting systems tested recorded votes with 100 percent accuracy.

As threats to voting systems are examined, it is also important to realize that voting systems technology has changed dramatically thereby minimizing, if not eliminating, many of the concerns that existed when these systems were first introduced. Many states have adopted precinct count optical scan system over central count to provide the voter the opportunity to correct deficiencies on their ballot prior to it being cast. This requirement, although seemingly minor, has eliminated many voters' choices from going uncounted. Additionally, touch screen systems have evolved into a much more sophisticated, secure and reliable system. Unlike the first generation full face DRE's, the second generation provides many more safeguards to prohibit errors from occurring or prompting the voter of an action that needs to be taken. The differences in these types of voting technology is important, especially as noted in the results of the new CalTech/MIT Voting Technology Project and Florida's Analysis and Report of Overvotes and Undervotes for the 2004 Election. Studies have shown a reduction in residuals and the elimination of the racial gap with DRE's.

The different, and sometimes competing, communities need to work together to ensure the security of these systems. The current perception of voting systems being unreliable, unsecure and inaccurate will never change as long as misinformation continues to be offered as fact.

Florida elections officials recommend, at a minimum the following:

- more emphasis on training local elections officials and/or their staffs
- a more comprehensive set of standards for election security procedures that extends beyond the actual voting system
- that voting systems not be a "networked" system
- that voting systems not be dependent upon the internet and prohibit voting systems from accessing the internet.

On a local level, elections officials should be held accountable for providing the safeguards necessary to ensure their electorate that their voting systems are secure. On the national front, organizations such as the National Institute of Standards and Technology (NIST) should not only look at threats coming from the actual voting unit or system but through the people managing these systems and whether policies and procedures are in place to minimize the threat of abuse.

1S-2.015 Minimum Security Procedures for Voting Systems.

(1) PURPOSE. To establish minimum security standards for voting systems pursuant to Section 101.015(4), F.S.

(2) DEFINITIONS. The following words and phrases shall be construed as follows when used in this rule:

(a) A "Ballot" when used in reference to:

1. "Paper ballot" means that printed sheet of paper, used in conjunction with an electronic or electromechanical vote tabulation voting system, containing the names of candidates, or a statement of proposed constitutional amendments or other questions or propositions submitted to the electorate at any election, on which sheet of paper an elector casts his or her vote.

2. "Electronic or electromechanical device" means a ballot that is voted by the process of electronically designating, including by touchscreen, or marking with a marking device for tabulation by automatic tabulating equipment or data processing equipment.

(b) A "Voted Ballot" means a ballot as defined above, which has been cast by an elector.

(c) "Voting System" means a method of casting and processing votes that functions wholly or partly by use of electromechanical or electronic apparatus or by use of paper ballots and includes, but is not limited to, the procedures for casting and processing votes and the programs, operating manuals, tabulating cards, printouts, and other software necessary for the system's operation.

(d) "Voting Device" means any apparatus by which votes are registered electronically.

(e) "Election Materials" means those materials provided to poll workers to properly conduct the election and shall include, but not be limited to: legally required affidavits and forms, provisional ballots, voter authority slips, precinct registers, and any electronic devices necessary to activate ballot styles in the voting system.

(3) SECURITY PROCEDURES. Requirements for filing security procedures with the Division of Elections. Each supervisor of elections shall place on file with the Division of Elections security procedures which meet the minimum standards set forth in this rule. Revisions to procedures on file with the Division of Elections shall be submitted at least 45 days prior to the commencement of early voting for the first election in which they are to take effect and shall be accompanied by a statement describing which part of the procedures previously filed have been revised. Each supervisor of elections has the authority to make changes to the security procedures within 45 days prior to the commencement of early voting for an election as a result of an emergency situation or other unforeseen circumstance. The supervisor shall document any changes to include the reasons why such changes were necessary. A copy of any changed document authorized by the supervisor shall be submitted to the Division of Elections within 5 days of the change.

(4) REVIEW OF SECURITY PROCEDURES.

(a) The Division of Elections shall conduct a review of the submitted security procedures to determine if they meet the minimum requirements set forth in this rule. The Division of Elections will notify the supervisor of elections as to the results of the review within 30 days of the date revisions to the security procedures are received in the office of the Division of Elections. If the Division is unable to complete its review within the time frame established in this rule, the procedures or revisions shall be temporarily approved until such time as the review is completed and the supervisor of elections will be notified accordingly. The notification of the results of the review will include an enumeration of specific provisions which were found to be incomplete or otherwise do not meet the provisions of this rule.

(b) Security procedures on file with the Division of Elections shall be reviewed by the Division of Elections in each odd numbered year, pursuant to Section 101.015(4)(b), F.S.

(5) STANDARDS FOR SECURITY PROCEDURES.

(a) Security procedures shall include copies of each referenced form, schedule, log or checklist or descriptions of the contents of forms, schedules, logs or checklists that vary from election to election.

(b) Election Schedule. The security procedures shall require the establishment of an election schedule at least 90 days prior to each regularly scheduled election and within 20 days of the date a special election is scheduled. The election schedule shall contain the following:

1. A list of all tasks necessary to conduct the election;

2. The legal deadline, where applicable, or tentative date each task is to be completed; and

3. The individual (position title), group or organization responsible for completing each task.

(c) Ballot Preparation. The security procedures shall describe the steps necessary to insure that the ballot contains the proper races, candidates and issues for each ballot variation and that the ballots can be successfully tabulated. The ballot preparation procedures shall, at a minimum, contain the following:

1. Method and materials required to determine each type of ballot or ballot variations;

2. Assignment of unique marks or other coding necessary for identifying ballot variations or precincts;

3. Verification that unique marks or other coding necessary for tabulation are correct;

4. Description of system used to facilitate ballot preparation, if applicable; and

5. Description of method to verify that all ballots and ballot variations are accurately prepared and printed.

(d) Preparation and Configuration of Tabulation System.

1. The procedures relating to the preparation and configuration of the tabulation system shall, at a minimum, include the following:

a. Description of the ballot definition and verification process;

b. Description of the steps necessary to program the system; and

c. Description of the process to install the program and the procedures for verification of correctness.

2. The security procedures shall describe the test materials utilized and the voting system tests performed prior to the conduct of the public logic and accuracy tests.

(e) Logic and Accuracy Test. The security procedures for use with electronic and electromechanical voting systems shall, at a minimum, describe the following aspects of logic and accuracy testing as required by Section 101.5612, F.S.:

1. Description of each test performed including the test materials utilized.

2. Description of how the programs, ballots, and other test materials are sealed, secured and retained.

(f) Filing election parameters. The security procedures shall include filing with the Division of Elections a copy of the software and parameters used within the voting system to define the tabulation and reporting instructions for each election regardless of filings for previous elections. The filing shall, at a minimum, include the following:

1. Copy of the voting system software;

2. Copy of the administrative database used to define the election;

3. Copy of all election-specific files generated and used by the system;

4. Documentation stating the release level of the precinct tabulation equipment and firmware; and

5. If the election definition is created by an individual who is not an employee of the supervisor of elections, then the parameters shall include a statement signed by the person who created the election definition. The statement shall be in substantially the following form:

ELECTION PARAMETER STATEMENT

Pursuant to Section 837.06, F.S., whoever knowingly makes false statement in writing with the intent to mislead a public servant in the performance of his or her official duty, shall be guilty of a misdemeanor of the second degree, punishable as provided in Section 775.082 or 775.083, F.S.

The election coding for ________ County was assembled according to specified procedures using (name of system and Florida certification number). Furthermore, included with the election materials is a duplicate copy of the administrative database used to define the election, a copy of the voting system software, a copy of all election-specific files generated and used by the system and a document stating the release level of the precinct tabulation equipment and firmware. To the best of my knowledge and belief, the foregoing statement is truthful.

Signature of the Person Coding the Election.

(g) Pre-election Steps for Voting Systems. The security procedures for use with voting devices shall, at a minimum, include the following:

1. Description of how the number of voting devices for each precinct is determined;

2. Description of each component of the public test, including any test materials utilized;

3. Description of the process to seal and secure the voting devices. It shall also provide for a record to be kept on which the identification numbers, seal numbers and protective counter numbers for voting devices shall be noted; and

4. Description of the procedures for retaining the test results and any records of the proceedings.

(h) Ballot Distribution. Where paper ballots (as defined in subparagraph (2)(a)1. of this rule) are used, the security procedures shall, at a minimum, include the following:

1. Description of how the number and variations of ballots required by each precinct is determined;

2. Description of the method for securing the ballots; and

3. Description of the process for distributing the ballots to precincts, to include an accounting of who distributed and who received the ballots, the date, and how they were checked.

(i) Distribution of Precinct Equipment. The security procedures shall describe the steps necessary for distributing voting system equipment to the precincts.

(j) Election Board Duties.

1. The security procedures when paper ballots, including provisional ballots are used shall, at a minimum, include the following Election Board duties:

a. Verification that the correct number of ballots were received, and that they are the proper ballots for that precinct;

b. Checking the operability or readiness of the voting devices;

c. Checking and sealing the ballot box;

d. Description of how spoiled ballots are handled;

e. Description of how write-in and provisional ballots are handled; and

f. Accounting for all ballots after the polls close.

2. The security procedures for use with voting devices shall, at a minimum, include the following Election Board duties:

a. Verification of the identification numbers, seal numbers, and protective counter numbers of precinct tabulation and/or voting devices;

b. Checking the operability or readiness of the voting device;

c. Verification that all counters except protective counters are set at zero on each voting device;

d. Securing a printed record from each voting device, if applicable;

e. Checking the correctness of the ballot;

f. Preparing voting devices for voting;

g. Verification that the correct number of voter authorization slips were received;

h. Checking and sealing the voter authorization slips container(s);

i. Handling write-in ballots;

j. Handling voting system malfunctions;

k. Securing voting machines at the close of the polls to prevent further voting;

l. Accounting for all voter authorization slips received; and

m. Recording and verifying the votes cast.

(k) Transport of Ballots and/or Election Materials. The security procedures shall describe the steps necessary to ensure a complete record of the chain of custody of ballots and/or election materials and shall include:

1. A description of the method and equipment used to transport all ballots and/or election materials;

2. A method of recording the names of the individuals who transport the ballots and/or election materials from one site to another and the time they left the sending site; and

3. A method of recording the time the individuals who transport the ballots and/or election materials arrived at the receiving site and the name of the individual at the receiving site who accepted the ballots and/or election materials.

(l) Receiving and Preparing the Ballots for Central and Regional Counting. The security procedures shall describe the process of receiving and preparing voted ballots, election data and/or memory devices for counting to include, at a minimum, the following:

1. Verification that all of the ballot containers are properly secured and accounted for and that the seal numbers are correct;

2. Verification that the ballot container(s) for each precinct contain voted ballots including provisional ballots, unused ballots, spoiled ballots and write-in ballots as shown to exist on the forms completed by each election board for that purpose;

3. Inspection of the paper ballots to identify those that must be duplicated or upon which voter intent is unclear, thus requiring a determination by the Canvassing Board. A record shall be kept of which paper ballots are submitted to the Canvassing Board and the disposition of those paper ballots; and

4. Description of the process for duplicating and recording the voted paper ballots which are damaged or defective.

(m) Tabulation of Vote.

1. The security procedures for use with central and regional processing sites shall describe each step of a ballot tabulation to include, at a minimum, the following:

a. Counting and reconciliation of voted paper ballots;

b. Processing, tabulation and accumulation of voted ballots and election data;

c. Processing and recording of all write-in and provisional ballots;

d. The process for handling unreadable ballots and returning any duplicates to tabulation;

e. Backup and recovery of tabulated results and voting system programs for electronic or electromechanical voting systems; and

f. Describe the procedure for public viewing of the tabulation process and access to results.

2. Security procedures shall describe the steps necessary for vote tabulation in the precincts.

3. The security procedures for use in the precincts shall include procedures that describe each step of ballot tabulation to include, at a minimum, the following:

a. Printing of precinct results and results from individual tabulating devices;

b. Processing and recording of write-in votes;

c. Endorsing a copy of the precinct results by the Election Board;

d. Posting of precinct results;

e. Transport of precinct results to central or regional site;

f. Consolidation of precinct and provisional ballot results; and

g. Describe the process for public viewing of the tabulation process and access to results.

4. The procedures for resolving discrepancies between the counted ballots and voted ballots and any other discrepancies found during the tabulation process shall be described.

(n) Electronic Access to Voting Systems. Security procedures shall identify all methods of electronic access to the vote tabulation system, including procedures for authorizing electronic access and specific functions, and specifying methods for detecting, controlling and reporting access to the vote tabulation system.

(o) Absentee Ballot Handling. The security procedures shall include procedures that describe absentee ballot handling to include, at a minimum, the following:

1. Description of process for determining and verifying absentee ballot variations;

2. Description for process to assure voters are issued the proper absentee ballot;

3. Process for precluding voters from voting at the polls and casting an absentee ballot;

4. Process for opening valid absentee ballots in preparation for tabulation;

5. Process for recording the receipt of advance absentee ballots, regular absentee ballots, State write-in ballots and Federal write-in ballots and determining which ones should be counted if more than one per voter is received; and

6. Security measures for storing absentee ballots and related materials prior to and after an election.

(p) Ballot Security. The security procedures shall describe ballot accountability and security beginning with their receipt from a printer or manufacturer until such time as they are destroyed. The procedures for each location shall describe physical security, identify who has authorized access and identify who has the authority to permit access.

(q) Voting System Maintenance and Storage. The security procedures shall describe the maintenance and testing performed on all components of the system to assure that it is in proper working order and is within manufacturer's operating specifications. Procedures shall also describe storage and nonoperational maintenance of all voting devices.

(6) ACCESS TO TABULATION PROGRAM SOURCE CODE.

(a) No supervisor shall have access to any vote tabulation program source code to be used in an election unless prior approval has been obtained from the Division of Elections. Approval shall be based on the supervisor establishing security procedures which provide for maintaining a secured control copy of the certified release of the tabulation program source code; protecting source code from unauthorized access; and verification that the tabulation program source code used for each election is identical to the certified release.

(b) Any modification to tabulation program source code must be certified by the Division under the provisions of Rule Chapter 1S-5, F.A.C., before use in any election.

Specific Authority 101.015 FS. Law Implemented 101.015(4) FS. History–New 5-27-85, Formerly 1C-7.15, 1C-7.015, Amended 8-28-93, 11-24-04.

Method for Developing Security Procedures in a DRE Environment

Dana DeBeauvoir, Travis County Clerk

As November 2004 approached, everyone seemed to have one issue on his or her mind. From newspapers to television comedy to conversations in coffee houses, the Presidential election was the hot topic. But, this election year was different from four years ago. The 2000 Florida controversy, the resulting large-scale implementation of electronic voting, the strong memories of the 9/11 tragedy, and the polarized opinions of the country had culminated into a general anxiety not only about who was going to win but whether our election process could be disrupted and the results trusted.

In Travis County, Texas, we not only fielded questions of concern from citizens, political parties, candidates, and media organizations; we had our own uneasy feelings, feelings that turned from worry to conviction. We were going to do whatever it took to make sure our election was protected and that the public could trust that it was safe, fair, and accurate, no matter what happened here or anywhere in the world. That was an admirable, lofty goal, but how do you implement stubborn determination?

Believe it or not, we laid an egg. Our first inspiration for the egg came from our association with the legal community and their use of the rules of evidence. According to Article I of the Federal Rules of Evidence, "these rules shall be construed to secure fairness in administration, elimination of unjustifiable expense and delay, and promotion of growth and development of the law of evidence to the end that the truth may be ascertained and proceedings justly determined."

Make no mistake, we are not attorneys, but when we saw their standards for rules of evidence, we thought they were on to something. To give support and integrity to evidence, you need to make sure you have: something physical (reports, audit logs, etc.), recorded details about persons who were involved in creating or collecting the evidence (times, dates, names, signatures, etc.), and secure storage so that evidence cannot be tampered with (areas with limited access). We decided to adapt these standards to our election processes.

The second part of this idea came from our computer staff and their obsession with developing risk analyses. So, we broke down the election process into categories and began to brainstorm about the possible minor or catastrophic events that could happen in each area. (Coming up with scenarios of horrible events is easier than you think thanks not only to real life news stories, but our exposure to the creative minds of television and movie scriptwriters.)

As ideas poured out, the rule quickly became that generalities had to be broken down to tangible events. For example, to say, "someone could tamper with the DRE system" had to be followed up with ideas of specifically how someone would go about doing such a deed. Therefore, what we ended up with was a tool that provided perspective, replaced emotion with facts, and guided us to a detailed plan of action.

If you look at the attachments, you will see the evolution of our egg and examples of how we combined all of our ideas into a method of mitigating risks and providing verifiable checks and audits that election procedures were properly followed.

The result of our egg analysis was not only a new way of thinking for us, but also a plan and checklist for what needed to be done for the 2004 election and for all future elections. The process led us to reinforce and fine-tune many of our existing practices and to develop new initiatives. Listed below are some examples of new, continued, or enhanced practices that increase a secure election environment and promote public trust. Examples of these items are provided in the attachments, and since we are particularly proud of the work we did to increase security by using hash code and parallel testing, we have included more detail on these practices.

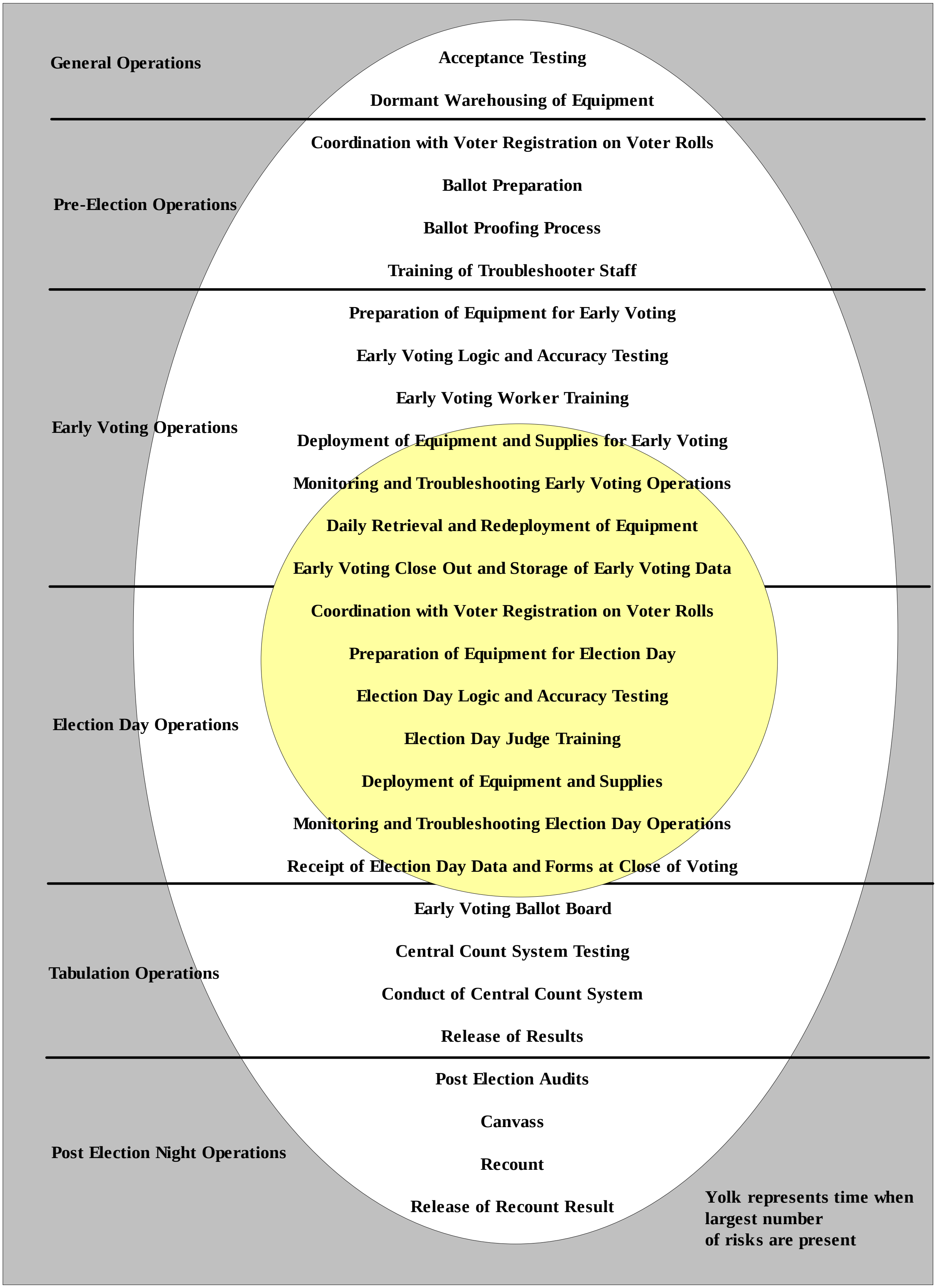
New, Enhanced, or Continued Security Practices

- Provide public invitation to attend all programming and testing activities
- Maintain written procedures and initialed tracking sheets
- Maintain independence from vendors
- Recruit, screen, and train skilled and trusted employees
- Coordinate emergency management plans with other relevant agencies
- Use Sheriff and Constable Officers to secure early voting electronic ballot boxes
- Improve security for the building where election activities occur
- Implement employee procedures that lower risk
- Conduct extensive pre-purchase testing of new equipment or software
- Provide continuous functionality testing of equipment
- Conduct Hash Code Testing on software
- Perform High Volume Testing of ballot programming
- Perform Parallel Testing
- Conduct Early Voting and Election Day audits by matching counts of voters by location as reported by the electronic voting system to the number of names on signature rosters
- Conduct post-election verification using the three redundant electronic sources, paper results printed from the electronic ballot boxes, and precinct-by-precinct election results

(When reviewing these practices, it may be helpful to understand that in Texas, a County cannot use a voting system unless the Texas Secretary of State has certified it. To date, no system allowing voter-verifiable paper ballots has been authorized, and therefore, could not be considered for use in the 2004 Presidential Election.)

Finally, about that egg concept... after you have read this, you may ask why we went with an egg shape instead of a rectangle or a circle. Truth be told, it started because the County Clerk's first drawing of an oval was less than perfect and resembled an egg. However, we capitalized on that idea. After all, we were birthing a new idea. Second, an egg has a hard shell wrapped around a permeable membrane. The shell ultimately served as a perfect metaphor and guide for determining the security levels needed for different groups (general public, candidates, law enforcement, etc.), and the membrane represented how information would flow back and forth through the process. Finally, the egg became a symbol for us. It is something with immeasurable value; something that must be given great love, care, and protection; and something that represents elections as the beginning and nucleus for a living democracy.

Egg Concept for Defining and Mitigating Security Risks in a DRE Environment



General Operations

Acceptance Testing

Dormant Warehousing of Equipment

Pre-Election Operations

Coordination with Voter Registration on Voter Rolls

Ballot Preparation

Ballot Proofing Process

Training of Troubleshooter Staff

Early Voting Operations

Preparation of Equipment for Early Voting

Early Voting Logic and Accuracy Testing

Early Voting Worker Training

Deployment of Equipment and Supplies for Early Voting

Monitoring and Troubleshooting Early Voting Operations

Daily Retrieval and Redeployment of Equipment

Early Voting Close Out and Storage of Early Voting Data

Election Day Operations

Coordination with Voter Registration on Voter Rolls

Preparation of Equipment for Election Day

Election Day Logic and Accuracy Testing

Election Day Judge Training

Deployment of Equipment and Supplies

Monitoring and Troubleshooting Election Day Operations

Receipt of Election Day Data and Forms at Close of Voting

Tabulation Operations

Early Voting Ballot Board

Central Count System Testing

Conduct of Central Count System

Release of Results

Post Election Night Operations

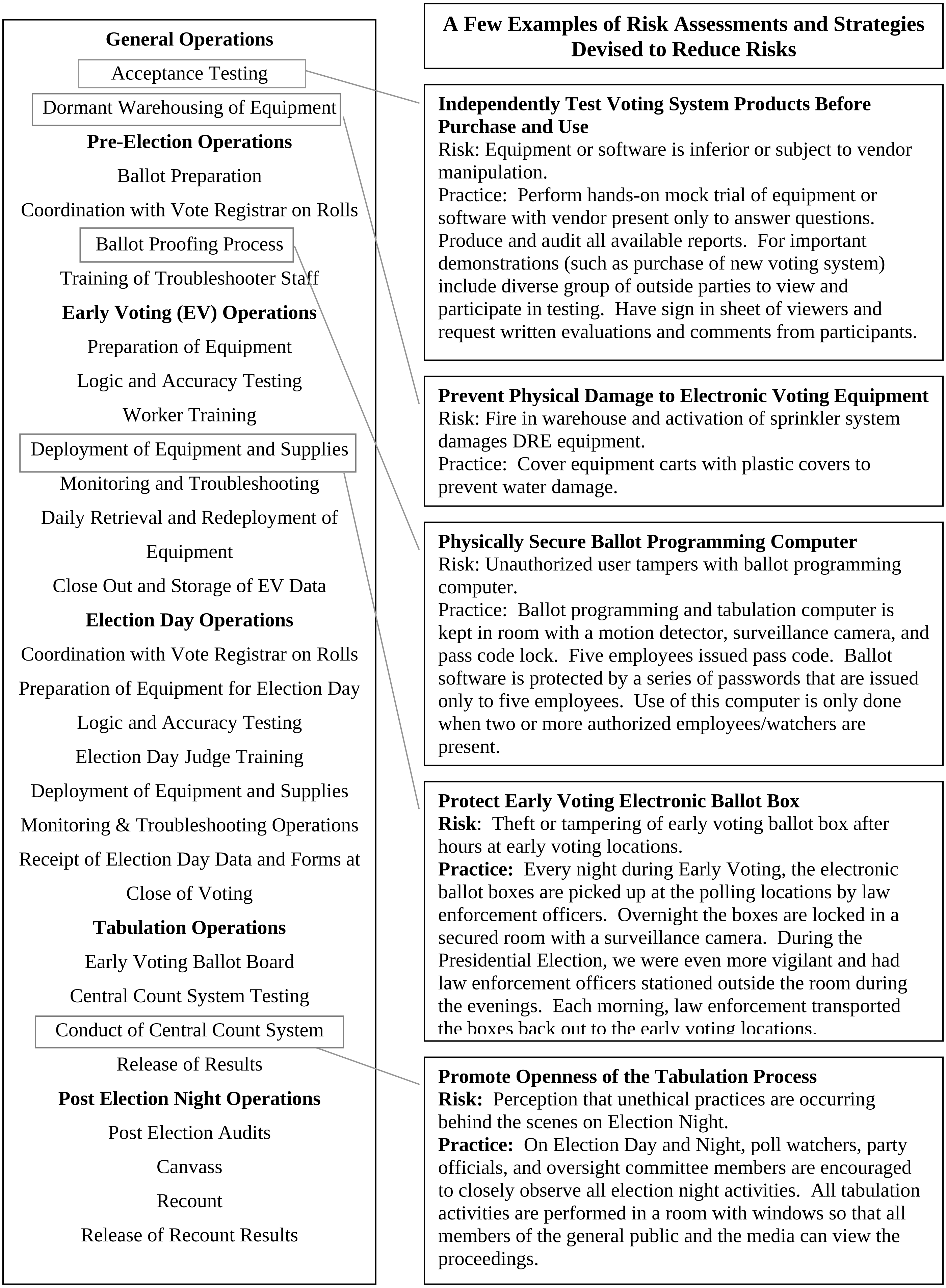
Post Election Audits

Canvass

Recount

Release of Recount Result

Yolk represents time when largest number of risks are present



General Operations

Acceptance Testing

Dormant Warehousing of Equipment

Pre-Election Operations

Ballot Preparation

Coordination with Vote Registrar on Rolls

Ballot Proofing Process

Training of Troubleshooter Staff

Early Voting (EV) Operations

Preparation of Equipment

Logic and Accuracy Testing

Worker Training

Deployment of Equipment and Supplies

Monitoring and Troubleshooting

Daily Retrieval and Redeployment of Equipment

Close Out and Storage of EV Data

Election Day Operations

Coordination with Vote Registrar on Rolls

Preparation of Equipment for Election Day

Logic and Accuracy Testing

Election Day Judge Training

Deployment of Equipment and Supplies

Monitoring & Troubleshooting Operations

Receipt of Election Day Data and Forms at Close of Voting

Tabulation Operations

Early Voting Ballot Board

Central Count System Testing

Conduct of Central Count System

Release of Results

Post Election Night Operations

Post Election Audits

Canvass

Recount

Release of Recount Results

A Few Examples of Risk Assessments and Strategies Devised to Reduce Risks

Independently Test Voting System Products Before Purchase and Use
Risk: Equipment or software is inferior or subject to vendor manipulation.
Practice: Perform hands-on mock trial of equipment or software with vendor present only to answer questions. Produce and audit all available reports. For important demonstrations (such as purchase of new voting system) include diverse group of outside parties to view and participate in testing. Have sign in sheet of viewers and request written evaluations and comments from participants.

Prevent Physical Damage to Electronic Voting Equipment
Risk: Fire in warehouse and activation of sprinkler system damages DRE equipment.
Practice: Cover equipment carts with plastic covers to prevent water damage.

Physically Secure Ballot Programming Computer
Risk: Unauthorized user tampers with ballot programming computer.
Practice: Ballot programming and tabulation computer is kept in room with a motion detector, surveillance camera, and pass code lock. Five employees issued pass code. Ballot software is protected by a series of passwords that are issued only to five employees. Use of this computer is only done when two or more authorized employees/watchers are present.

Protect Early Voting Electronic Ballot Box
Risk: Theft or tampering of early voting ballot box after hours at early voting locations.
Practice: Every night during Early Voting, the electronic ballot boxes are picked up at the polling locations by law enforcement officers. Overnight the boxes are locked in a secured room with a surveillance camera. During the Presidential Election, we were even more vigilant and had law enforcement officers stationed outside the room during the evenings. Each morning, law enforcement transported the boxes back out to the early voting locations.

Promote Openness of the Tabulation Process
Risk: Perception that unethical practices are occurring behind the scenes on Election Night.
Practice: On Election Day and Night, poll watchers, party officials, and oversight committee members are encouraged to closely observe all election night activities. All tabulation activities are performed in a room with windows so that all members of the general public and the media can view the proceedings.

Use of Parallel Testing to Detect Presence of "Time Bomb" Software Codes

(Abbreviated version of our procedures as used with Hart Intercivic E-Slate System)

Risk: Introduction of malicious software program written so that it is activated during the actual election process and therefore goes undetected in pre-election testing.

Practice: Perform parallel testing during Early Voting and Election Day to ensure that no such program is being activated. Randomly pull out equipment slated for polling location just before it is to be sent out. Perform testing in ELECTION mode so that it mirrors the election cycle of opening polls, casting ballots, and closing polls. Conduct test in a controlled environment under video surveillance. Encourage public viewing of test.

A. Parallel Test Spreadsheet

1. Create a spreadsheet using the Logic and Accuracy spreadsheet as a template.
2. Randomly enter votes for each precinct in no particular pattern (so software will not identify if it as a test).
3. Include enough ballots to ensure at least two ballots are cast per hour per day.

B. Paper Ballots

1. Using the Parallel Test spreadsheet, mark all paper ballots according to spreadsheet.
2. Double check ballots where marked correctly to ensure 100% accuracy.
3. Make a stack of ballots for each day of Early Voting and one stack for Election Day.

C. Polling Location Equipment

1. Randomly select a polling location during the day of delivery of equipment.
2. Replace removed equipment with extra equipment.
3. Place equipment in secured area and clearly mark as PARALLEL TEST EQUIPMENT.

D. Ballot Box Preparation

1. Gather 2 Ballot boxes with red seals. (one for Early Voting and one for Election Day)
2. Lock and seal the boxes. Record the seal numbers. Seals are not broken until the end of each test period.

E. Secured Area

1. Setup all parallel test equipment where all actions are visibly recorded by video surveillance.
2. Tag area with PARALLEL TEST – AUTHORIZED PERSONNEL ONLY signs.

F. Casting Votes

1. Use ballots designated for the specified day and corresponding parallel test.
2. Retrieve an access code for the first ballot and begin voting ballot one e-Slate as marked on paper ballot.
3. Once ballot has been cast print your initials, date, and time on the top right hand corner of the paper ballot.
4. Then print your initials, date, and time on the parallel test spreadsheet.
5. Staple access code to paper ballot on top left hand corner.
6. Insert paper ballot into ballot box.
7. Two ballots per hour per day should be voted.

G. Tabulation of results

1. Once the parallel test is completed, all materials should be placed in the BOSS room.
2. Tabulation of results will occur after the Official Elections results have been finalized.
3. Create a database in TALLY named PARALLEL TEST - "Name of election".
4. Insert MBB cards from parallel test equipment.
5. Tabulate results.
6. Print Cumulative reports.

H. Backup equipment (SERVO)

1. Using SERVO, create an event using the same naming convention in TALLY.
2. Backup all parallel test equipment to this event.
3. Print out "Devices backed up report".
4. Compare totals between TALLY, SERVO, and the parallel test spreadsheet. Totals should match identically.

Use of Hash Code Testing to Detect Modification of Software
(Abbreviated version of our procedures as used with Hart Intercivic E-Slate System)

Risk: Modification of software by vendor, employee, or outsider.

Practice: Use Hash Code testing to verify that software files installed on computers are the same as the software files qualified by an Independent Testing Authority and certified by the Secretary of State. Hash Code is a digital algorithm signature of a variable-sized amount of text that is converted into a fixed-sized output that can be used to determine if two objects are equal. Testing must be performed before and after the software is used in an election.

A. Create Hash Code Spreadsheet
1. Access NIST website to obtain hash types and file names. (www.nsrl.nist.gov/votedata.html)
2. Download zip format file from website.
3. Open file CompleteNSRLfile.txt in Excel and follow steps in Excel wizard when opening the text document.
4. Sort by Product Code, then File Name. Delete rows NOT for Code 9031. (9031 is for our e-Slate system.)
5. Save file.

B. Install Hash Master Software
1. Verify that each station has the Hash Master software installed. If not, use the setup.exe file on installation CD.
2. Follow instructions in the software wizard to complete installation of Hash Master.

C. Execute the Hash Code function (from Readme.txt)
1. To calculate and display the hash of a file:
   a. From the File menu, select “Select Algorithm.” The "Configure Hash Options" window appears.
   b. Select the hash algorithm to be used (Travis County uses MD5 or SHA-1).
   c. Click “Save.” The hash algorithm selected displays in the Hash Master window.
   d. From the File menu, select Process Files. The "Select one or more files to process" window appears.
   e. In the Look In field, find the directory that contains the file(s) to be processed. Complete one group of files per software at a time. Refer to the Hash Code spreadsheet to determine file paths for each software type.
   f. Select the file(s) to be processed.
   g. Click the Open button. The "Select one or more files to process" window closes. The path to the last file selected and its hash value appear in the Hash Master window.
      1. To copy the hash to the Clipboard: From the Edit menu, select Copy Hash to Clipboard. —OR— While in the Hash Master window, hold down the Ctrl key and press C.
      2. To view the File Hash Report for the file(s) just processed: From the Report menu, select View Report. The "Hash Report" window appears showing the File Hash Report. The File Hash Report contains the path and hash value for each file processed with the Process Files command.
      3. To print the File Hash Report for the file(s) just processed: From the Report menu, select Print Report. —OR—View the report, then click the Print tool icon at the top of the Hash Report window.
      4. To save the File Hash Report as PDF for the file (s) just processed: From the Report menu, select Save report as PDF. The Save report as PDF window appears showing the file directory. Indicated the file name and location where you want to save. Click the save button.
      5. To run the Third Party Hash for the last file just processed: Do not change the hash algorithm that was in effect when you processed the file. From the File menu, select Third Party. A command prompt window appears. Wait until the third-party hash utility finishes.
2. After completing one group of files for a specific software and hash type, exist Hash Master and repeat the process for all files for each software and Hash Type from the beginning.

D. Compare Hash Code files
1. Generate a paper report from Hash Master for each computer, hash type, and group of files. Staple each report to the Hash Code spreadsheet that corresponds to each group of files.
2. Label each report to identify which computer it was generated from. (i.e. BOSS computer)
3. Compare Hash Code files generated from Hash Master to files located on the Hash Code Spreadsheet. All files should be accounted for and match identically.

BREVARD DEMOCRATIC EXECUTIVE COMMITTEE
Government Affairs Committee

RESOLUTION

WHEREAS, An essential element of an effective democracy is the ability of each eligible and qualified citizen to be able to vote in fair and open elections, and for that vote to be registered and counted honestly and accurately,

NOW, THEREFORE, BE IT RESOLVED, That the Brevard Democratic Executive Committee supports the following:

THAT THE RESPONSIBILITY OF THE SECRETARY OF STATE INCLUDE setting rigorous, mandatory policy and procedure for all of the Supervisors of Election in Florida, to provide a single point of scrutiny of policy, procedures, purge lists, etc., and

THAT POLL JUDGES/WORKERS be selected as follows: The chief judge at each polling place to be appointed by the Supervisor of Elections. All other judges/workers at each polling place to be appointed in equal numbers by the Executive Committees of the Democratic and Republican Party, eliminating the appointment of all judges/workers from only one party, as well as the appointment of judges/workers who are Democrats or Republicans in name only, and

THAT THE FOLLOWING CONSIDERATIONS APPLY TO THE PROCUREMENT AND MANAGEMENT OF VOTING MACHINES:

- No voting machine with proprietary software be purchased
- A voter verifiable paper trail be required, to include a printout read and approved by the voter, so that a recount may be possible
- Every machine be tested for hardware integrity
- Each machine in each precinct be tested for software integrity, with source code to be reviewed for extraneous instructions
- No voting machine be permitted to have any external input/output other than 60-Hertz power after hardware and software certification, with power inlets adequately isolated from any data processing hardware in the machine.
- Each voting machine, immediately subsequent to hardware and software certification, be secured until set up for operation in the polling place
- At the close of polls:
    - O An appropriate number of printouts from each voting machine be signed by all election judges/poll workers, the machine secured and submitted to the control of the courts, and all machine tallies for each precinct totaled manually before reporting the grand total to the Supervisor of Elections and the media
    - O Technicians not be allowed to repair a machine after certification
        - A machine that demonstrates a problem be immediately secured
        - Each polling place have a spare machine to activate as necessary
        - If a dysfunctional machine fails to print its report, poll workers shall utilize only the paper records to generate a manual report for that machine

THAT RECOUNTS be required where the tally varies by more than 3% from the exit polls, and if that identifies discrepancies, a full recount be required, and

THAT THE STATE OF FLORIDA ENACT LEGISLATION that provides the following:

- Sets a uniform standard for the number of voting machines in each precinct based on the voter registration total for that precinct
- Make acts of voter suppression and/or intimidation a felony
- Provide adequate and permanent funding for replacing, testing, and maintaining voting machines, for accessibility to all polling places, and for the training of poll workers.
- Develop uniform standards for the applicability and processing of provisional ballots
- Provide automatic re-enfranchisement for felons who have satisfied their sentence

AND THAT THE U.S. CONGRESS ADOPT THE FOLLOWING LEGISLATION: HR.550 and S.450, both entitled “A Bill to Amend the Help America Vote Act of 2002”.

Approved by a vote of the Brevard Democratic Executive Committee on September ___, 2005.

____
Xxxxxx Xxxxxxx Xxxxxxxx, Secretary,
Brevard DEC

Strategies for Software Attacks on Voting Machines

John Kelsey, NIST, September 2005

*Existing attackers **are** that sophisticated, and these attackers are probably not as smart as the ones that might be brought to bear against a voting system.*

1. Introduction

This white paper discusses strategies for changing the outcome of an election via software attacks on voting machines.   This discussion mainly focuses on DREs, but applies as well to DRE+VVPAT and PCOS voting systems. I am going to consider a number of the operational difficulties of software attacks and point out how these may be overcome. The goal of this white paper is to explain why I think software tampering in voting systems is a practical threat. There are operational difficulties, but I am convinced that these can be overcome by a skilled and intelligent attacker.

The nature of software-based attacks on electronic voting systems is that electronic records are changed. Depending on details of the attack, paper records may also be produced, and in some voting systems, the tampered software can alter the paper records in some way, albeit usually with the possibility of the voter noticing this.

1.1. Background and Environment

In the last several years, there have been increasingly sophisticated software based attacks on real-world systems. Among the targets have been:

- US government systems, including those containing classified data;
- Financial systems, including attacks that gained perpetrators large sums of money;
- Content protection systems intended to stand up to extensive external attack;
- Special-purpose cryptographic devices intended to be resistant to both software and physical attack; and
- Cryptographic and security software, again designed specifically to resist attack.

This is today's environment. It is important to understand that we probably hear of only a small fraction of attacks on real-world systems. For each high-profile case of someone eavesdropping on a congressman's cell phone or the pagers of secret service agents, there must be many other cases where the attackers don't disclose what they've learned. For every case where financial data is tampered with and the theft is discovered and reported, there must be many other cases where it is never detected, or is detected but never reported.

In addition, we have seen the rise of sophisticated attacks on widely-used computer systems (desktop PCs) for a variety of criminal purposes that allow the criminals to make money:

- Activities/methods such as phishing (spam intended to get users to disclose private data that allows an attacker to steal their money) and pharming (exploitation of DNS[1] to redirect legitimate web traffic to illegitimate sites to obtain private data) continue to grow.
- Extortion against some computer sites continues, with an attacker threatening to shut down the site via distributed denial of services (DDOS) attack unless he is paid off.
- Large networks of "bots"—random users' computers which have been taken over by an attacker for use in the above kinds of attacks, are bought, sold, and rented.

The sophistication of these attackers undermines the common responses to discussions of software attacks that "attackers wouldn't be that smart." Existing attackers *are* that smart, and these attackers are probably not as smart as the ones that might be brought to bear against a voting system—which might include national intelligence services of foreign countries. It is very hard to make an argument that the Russian or Chinese intelligence services can't find attackers who are more competent than the ones currently making money from spamming, phishing, pharming, DDOS, and related attacks. It is even harder to make the argument that these organizations wouldn't be interested in changing the outcome of a US national election.

1.2. Targeted Voting Systems

In this white paper, I am focusing on voting machines which are close to the voter, including DRE voting machines with or without paper audit trails, voting machines to fill out an optical scan ballot, and the machines which scan optical scan ballots. I do not generally work out full attacks—instead, focusing on the initial step of getting tampered software onto a voting system, controlling its actions so that it can change an election outcome in a reliable way, and keeping it from being discovered while doing so. Part of this may involve interfacing with other parts of a more complex attack.

2. How the Tampering Program Works

There are many ways for the tampering program to work. Without trying to get into fine details of the various voting systems, we can describe a number of broad methods for the tampering program to alter votes.

2.1. Changing System Settings or Configuration Files

[1] Domain Name System (DNS) is a distributed database that stores mappings of Internet Protocol addresses and hostnames to facilitate user-friendly web browsing.

The first method considered in to change the system setting or configuration files. There are many ways this can work. An attack program must tamper with the system settings or configuration files after L&A testing, but has a great deal of flexibility as to when to do so. The attack program can be buried in some driver or program that is only run when the voting is started, or some timed program that decides whether to trigger at a fixed time each day. Among the attacker's options within this class of attacks are:

- Swap contestants in the ballot definition or other files, so that a vote for John Smith is counted as one for Mary Jones, and vice versa, all the time. This only makes sense if the swapping can be applied selectively, and done only at voting machines which are likely to get a majority of the "wrong" vote. (This is an attack described in the RABA report[2], but we propose doing it wholesale instead of retail.)
- Alter configuration files or system settings for the touch screen or other user interface device, to cause differential error rates for one side vs. the other.
- Alter configuration files or system settings for the scanner to introduce differential error rates for one side vs. the other.
- Alter configuration files or system settings to make it easier to accidentally skip a contest or misrecord a vote, e.g., by increasing or decreasing touchscreen sensitivity or misaligning the touchscreen.
- Alter configuration files or system settings to change the behavior of the voting machine in special cases, such as detected undervotes or overvotes, or fled voters.

The main operational problems with these attacks include:

- Leaving incorrect configuration files at the end of voting, which may reveal the attack. The attack must thus trigger twice, once to change the configuration to an incorrect state for the attack, once to change it back.
- Deciding when to trigger—many of these attacks will cause voters' intentions to be misrecorded without regard for which way they're voting. Those attacks must trigger only for voting machines which are mostly used by people voting the "wrong" way. This implies either selectively installing the attack program, or selectively triggering it.
- Some of the attacks in this category may require fine knowledge of the format of the ballot definition files, though it is not clear that this must always be true.
- Changes in system settings or configuration files are likely to leave entries in the event logs. These entries must either be prevented or deleted by the attack program if the event logs are checked.

2.2. Active tampering with user interaction or recording of votes

In this class of attack, the attack program triggers during voting and interferes in the interaction between the voter and the voting system. For example, the attack program may:

[2] http://www.raba.com/press/TA_Report_AccuVote.pdf

- Tamper with the voter interaction to occasionally introduce an “error” in favor of one contestant.
- Tamper with the voter interaction both in vote entry and verification, so that the voter sees consistent feedback that indicates his vote was cast correctly but the rest of the voting machine software sees a changed vote.
- Tamper with the electronic record written after the verification screen is accepted by the voter, e.g., by intercepting the function call to write the results and altering those results before they are written.

This class of attack seems to raise few operational difficulties once the attack program is in place. One operational difficulty is of interest in dealing with systems with paper records:

- The attack which introduces biased errors into the voter’s interaction with the voting system is especially useful for attacking DRE+VVPAT and PCOS systems where the paper record is printed or filled in by the voting machine being attacked, since the attack behavior, if detected, is indistinguishable from user error. However, the attack program can improve its rate of successfully changed votes, and minimize its chances of detection, by choosing voters who are unlikely to carefully check their paper records. Thus, voters using assistive technology are likely targets, though there are probably not enough such voters to change the outcome of most elections.

2.3. Tampering with electronic memory after the fact

An alternative approach is to change votes in electronic memory at the end of voting, but before the totals are displayed locally or sent to the tabulation center.

In this case, the attacking program need only be activated at the end of voting. This also allows the attack program considerable flexibility, as it can decide whether to tamper with votes at all based on its local totals, which can include number of votes and elapsed time voting (to avoid being caught by parallel testing in many cases).

However, this raises a few interesting operational difficulties:

- This class of attack only works on voting machines that produce the electronic records, such as scanners in PCOS systems, and DREs. It is of no use against ballot marking devices. Attacks on systems that produce a paper record as well as an electronic record require an additional attack step to avoid detection.
- DREs typically store electronic records in many locations; the attack program must change them all.
- The attack program must avoid leaving entries in the event or audit logs of its accesses to the electronic totals which would indicate an attack. (If simple file access is logged, this raises no problems for the attack program; if each record

altered yields a log entry, this requires tampering with the event log to avoid detection.)

- Depending on details of the file accesses required, the attack program may face some time constraints on making the desired number of changes. However, note that a program that is to change 5% of votes for Smith into votes for Jones can simply hop around at random in the set of votes (they can't be stored sequentially for voter privacy reasons) and process about 10% of them to accomplish its goal. There also will very likely be a reasonable span of time between the closing of polls and the display and transmission of results.

3. Attack Program Control Strategies

3.1. Overview: Attack programs, remote control, and backdoors

One practical problem confronting any attacker is how to control his attack program. If we assume that getting a usable attack program or exploitable backdoor in a voting system is expensive, in terms of effort or risk, then a sensible attacker will want to reuse the attack program if possible. However, this must be balanced against the additional risk and effort needed to get a more flexible program into the voting system.

There are a number of broad strategies an attacker may have for controlling his attack program, including:

- One-step ("fire and forget") attacks: In a one-step attack, the attacker puts the attack program into the targeted machines, and has no further contact with the machine. This class of attack minimizes conspiracy size, since the attacker need not get anyone else involved in the attack to make it widespread. However, these attacks have little flexibility, and may require attack programs sophisticated enough to determine which candidate is to be favored in the election fraud from the ballot definition files.

    - One-shot attack programs—in this case, the attacker constructs an attack targeted at a single election, which will go dormant or delete itself if possible at the end of that election.
    - Persistent bias attack programs—in this case, the attacker creates a program which will attempt to bias future elections in a specific direction (probably toward a specific party, since individual candidates and questions will not be around for that many elections, in general).

- Two-step attacks:

    - Reusable attack programs—in this case, the attacker builds a program which allows some form of "remote control" to activate and/or control its tampering. This allows the attacker to reuse the same attack program in a flexible way many times, but requires a more complex and sophisticated

kind of program, and also requires some kind of additional control channel into the voting machines.
    - Reusable back doors—in this case, the attacker builds in (or leaves) an easily-exploited weakness in the voting machine software, which he knows how to use to install an attack program. The attack program may be any of the above kinds.

In terms of economics of the attack, the reusable choices are much better for the attacker. Both of the patterns of attack in these choices exist for programs used to attack real-world computers; some attackers install new backdoors to allow a later compromise of the machine as needed, while others install programs allowing them to simply send commands to the compromised machine.

3.2. One-Step Attacks

The most straightforward software attacks are one-step attacks: the attacker writes, tests, and inserts the attack program into the voting system, and has nothing more to do with the voting system.

3.2.1. One-Shot Attack Programs

A one-shot attack program is targeted at a single election. When the election is over, the attack program will either go dormant or (if possible) delete itself. Because this kind of program is targeted at a single election, it can be relatively simple. It may be developed before or after the ballot definitions are produced, but it is always targeted at a single election. It may thus use candidate names or other indications to decide how to change votes, and it may trigger on exactly one date at exactly one time.

The main limitations of this kind of attack are:

- The attacker spends all the resources to develop and insert the attack program into the voting system, and he gets to use it only once.
- There is no chance for the attacker to control the attack program's behavior. Thus, he cannot keep it from triggering in places or circumstances which may reveal its existence, except for whatever guidance he provides it in its design.

If the attack program is inserted into the voting system before the ballot definitions are specified, it must be able to determine, from the ballot definition file and other system information, which ballot question it must affect, and in which direction. We assume here that the program either can determine this from the ballot definition file or the onscreen display. It is sometimes disputed that an attacker could write an attack program of this flexibility. We find this claim unconvincing, given the remarkable sophistication of some real-world attacks. However, we will point out that:

- The attack program's required sophistication falls rapidly as the attack grows more targeted—it is going to be much easier to design an attack program to work in one county in Maryland than to work all over the US.
- The attack program is targeted at a single election, so that the name and party affiliation of the favored candidate is probably known when the program is written.
- If the attack program is finished and inserted after the ballot definitions are made available, then it need not be sophisticated at all.

This kind of attack is probably best inserted at a local level. If it is inserted into all voting machines of a certain make all over the country, it should detect an unfamiliar ballot definition file or format, or a state or county name on the ballot, and never trigger unless these look right.

I expect that this attack is a reasonably likely one to be used on a local or statewide level. The more broadly the attack is applied, the more likely it is to be detected, due to unforeseen software bugs, interactions with other options unfamiliar to the attack program's author, etc. The best way to control this kind of attack is to selectively install the attack program only on a small subset of voting machines, using either physical access, network access, or the ability to install invalid or tampered-with software patches.

3.2.2. Persistent Bias Attack Programs

A somewhat more efficient use of the attacker's resources may be to build an attack program which provides a systematic small bias for the party of the attacker's choice. The attack program must detect the political party for which someone is voting and introduce a slight bias. The two most obvious ways to do this are:

- Trigger only on straight-ticket votes; change some fraction of straight ticket votes from their original party to the different party.
- Trigger when the attacker's preferred party is losing on a specific voting machine, increasing the error rate (for example, by messing up the alignment of the touch screen or occasionally skipping a ballot question).

Again, this class of attack suffers from a lack of flexibility, though it is so broad in impact that that flexibility is not so essential. Another problem with this attack is that an attacker motivated by greed probably cannot get paid for it; while many people would broadly like to see one party or the other do better, a broad improvement for the whole party may be hard to get a single person to pay for. (By contrast, the one-shot attack program is probably relatively easy to get paid for, in the sense that there's a single beneficiary.)

I expect this kind of attack program is the least likely to be used in practice.

3.3. Two-Step Attacks

In this section, we describe attacks that require two stages—creating/planting the attack program or backdoor, and exploiting it. These are inherently more complex in operational terms, but they are also enormously more flexible.

In a two step attack, there are typically two different insiders involved. The first insider must insert the attack program or backdoor; the second must exploit it. To affect many polling places, counties, or states typically requires many insiders sending control information into the voting machines. On the other hand, the attacks become much more flexible, and an attacker who inserts the attack program into the system can in principle make money by selling access to a corrupt politician or campaign manager.

3.3.1. Attack Programs with Remote Control

Attack programs that provide the attacker some kind of remote control over the compromised machine are widespread. (This is basically how bot networks work; the attacker who has taken them over has some way of controlling their future actions.)

In an attack program for a voting system, the attacker wants to be able to exert control over:

- Which machines with the attack program installed trigger;
- What changes are made to the election outcome; and
- What additional conditions are checked for by the attack program prior to triggering.

Depending on when the remote control messages are sent, the attacker may already know things like the complete ballot definition file contents, making it much easier to tell the attack program how to change votes. The attacker is also likely to know any recently-announced countermeasures, such as parallel testing or hand-recounts of some paper-based machines. He can take this into account in his commands to his attack program.

There are really two broad categories of remote control messages to consider:

- Commands to the attack program. These can in principle be very low bandwidth messages, and can be hidden stenganographically in a variety of files and other messages.
- New programs to install. These are maximally flexible, but require that the attacker have a fair amount of bandwidth available to the machines being controlled. These are considered in the next subsection, on reusable back doors.

The attacker's major operational problem with remote control of his attack program is finding an available channel over which to communicate his commands to his attack program. Unlike compromised desktop PCs, voting machines are seldom directly on the internet, waiting for an inbound connection from the attacker. Instead, they are usually not powered on at all, and when they're on, they are likely in a somewhat restricted environment. Some broad classes of command channel include:

- Voter or poll worker interaction with the machine through its normal interface. This provides very limited bandwidth, but may be useful for a “secret knock” to activate attack behavior. Because a human must interact individually with each voting machine, attacks using this technique require reasonably large conspiracies. However, if the conspirators are simply voters, it doesn’t require especially highly-placed conspirators. Further, in some cases, the conspirators are simply told to vote a certain way; they don’t even need to know what they’re doing. A simple secret knock may not require any additional insider access after the attack program is inserted, but it has limited bandwidth; the most likely use for such a secret knock is simply to turn on the attack behavior.

  The secret knock can be almost anything that can be done through the normal user interface, so long as it is extremely unlikely to happen by chance. For example:

  - Touching several specific parts of the touchscreen simultaneously or in a specific order;
  - Voting for a specific pattern of candidates. (Note that this does not require that the attack program knows which pattern of candidates will be available ahead of time; see the discussion of steganographic techniques below!)
  - Voting for a specific write-in candidate; or
  - Mistyping a password or PIN a certain number of times when trying to log in.

- Configuration files for the election, including ballot definition files, audio ballot files, etc., can contain hidden instructions. This allows a very powerful attack, because the conspiracy size is potentially two people: one person to write and insert the attack program and one person to produce a file for the election which can be reviewed by anyone without detecting anything amiss. This also allows the attack program to simply read instructions about which ballot questions to tamper with, and in what directions, from someone who already knows the full contents of the ballot definition file.

How much information is needed to control an attack program? Let’s assume the attack program needs to know one ballot question to change and in which direction, and needs to trigger only when it’s told to.

- A checksum on the command of 20 bits leaves about a one in a million chance of incorrectly triggering.
- If there are no more than 128 ballot questions, and no more than 8 choices we might want to bias the machine towards, then an additional 10 bits are needed to specify them.
- With more bandwidth, we can embed further information. For example, a 16-bit additional command can specify an exact time to trigger, starting

at any hour, for the next seven and a half years.

How hard is it to embed such control information in a file? The fine details depend on the details of the file format, but a few parameters are easy to see:

- The creation time on a file will routinely include about 16 bits of choice for the attacker. If the attacker can choose the creation date on two such files, he can embed 32 bits, enough for our simplest attack control.
- Existing off-the-shelf steganographic programs for audio and picture files allow embedding of thousands of bits in ways that are not detectable by humans.
- If an attacker can choose many small variations in a ballot definition file, e.g., by adding an extra space character or not in each of 100 entries, he can produce a lot of different variations. For the example of the space character, he can embed about 100 bits in the ballot definition file. By using some checksum or hash function in his attack program, he can simply vary the ballot definition file in unobtrusive ways at random until he finds one with the right 32-bit CRC checksum, and use the CRC checksum as the command.

In general, blocking all possible covert channels into some program or device is extremely difficult. Countermeasures which would overcome any of the above techniques would still allow variations on them to be carried out. We thus come to the conclusion that an attacker who can provide one of more ballot definition files to the voting machine is very likely able to embed detailed commands to an already-present attack program on the voting machine, with almost no possibility of detection. Note that this requires a very specific kind of insider access—someone with the power to supply or alter at least one of the ballot definition files must be in on the attack. However, there will be nothing incriminating about the files; the insider embedding commands for the attack program will be able to give his files to uncorrupted observers without fear of discovery. The original author of the attack program will presumably give each conspirator a program to use in embedding commands to the attack program, so no great technical sophistication is assumed for the insider.

- Any network access for the voting machine during machine setup, testing, voting, or even at the end of voting, but before results have been reported, can be used to give the attack program detailed commands about how to tamper with the election results.

The most natural way for this to work is for the attack program to set up a program which is silently listening on some port for an inbound connection, and which accepts the connection and accepts commands. For this kind of command channel, the attack program might simply wait for a new uploaded attack program and then install it, or might accept a small sequence of commands as described above. This kind of command channel is widely used in bot networks today.

In this case, the second step of the attack can be done by someone with no insider access, simply by carrying a wireless-enabled PDA in his pocket while voting, or by establishing communications with the voting machines from outside the warehouse in which they are being configured. (Note that normal range limits on wireless access are for standard wireless networking hardware—larger antennas and better equipment can provide a substantial improvement in range—in some cases, 802.11 access has been achieved at a range of several miles!)

Even in the absence of such a powerful channel, however, the wireless network can be used as a signaling device. Among the obvious covert channels available are:

- Precise timings of ping or other inoffensive packets arriving on the network.
- Refused connection attempts.
- The names of networks broadcasting their presence.

Again, this doesn't require any insider access for the second step.

3.4. Reusable Back Doors

An alternative technique for attacking the voting system in software is for the attacker to insert some subtle bug into the program, which will allow a fairly easy takeover of the machine later. This has the enormous advantage that getting caught doesn't mean going to jail or getting fired, it merely means having to fix a subtle bug. A capable attacker who expects a competent code review will insert a dozen subtle bugs in the program, each allowing a silent takeover of the voting system software after the fact.

Instead of control channels, we must now consider future attack channels. The best of these are probably:

- Configuration and other files. The programs reading the files can include some kind of buffer overrun, or some unusual-looking escape sequence that gets part of an input string from a file out to a command shell, PERL interpreter, or some such thing.
- Software patches. The programs doing any verification of the software patches prior to installation can have a subtle bug by which the attacker can bypass that verification step.
- User interface. The program can have a subtle bug which makes it possible to get from a low-privilege user interface to a command shell running as root or administrator, and thus to change settings or install software.
- USB and device drivers. The COTS USB driver can have a bug which allows a tampered USB device to take the voting machine over.

Note that for all of the above, the second step of the attack requires insider access—either physical access to the voting machines to be compromised, or the ability to write configuration files or provide software patches to the machines being attacked.

- Wireless and wired networks. The programs that deal with the network access, either voting-specific programs or COTS programs, can have embedded attacks as described above.

In this case, the second step may be carried out by someone without any special access. For a known vulnerability with a known attack program to be inserted, the second step can be carried out by a conspirator carrying a wireless-enabled PDA running a program to take over as many voting machines as possible with this vulnerability. This is basically what is done in recent Bluetooth viruses, which seek out vulnerable devices via Bluetooth and attempt to infect them when they appear. Network worms also use known vulnerabilities to carry out automated attacks.

4. Attack Points

Modern voting machines typically have a lot of software and files on them, and provide complex interfaces for human users and other machines. An attacker needs to find a point at which he may insert his attacking program without detection. Depending on the attack point, we can determine a great deal about the nature of the attack; an attacker who tampers only with a few machines' software can write a very simple attack program, based on a thorough knowledge of election procedures, local ballot design, etc. On the other hand, an attacker who tampers with a whole line of voting system software by a major vendor must either make use of a two-stage attack, or must write a very sophisticated attack program to avoid detection and correctly tamper with the election in a huge variety of circumstances.

4.1. Original Voting System Software and Configuration

The original voting system software is developed by the vendor, with use of COTS software and tools. It is then provided to a testing lab for some level of checking. We may assume that the testing lab will not pass voting system software with obvious attack program behavior (e.g., a reachable menu screen asking the user how he wants the election results cooked).

However, there are a number of ways that an attack program might hide within the original voting system software: (This is by no means an exhaustive list!)

- The attack program could be part of COTS software which was purchased for use on the voting system.
- The attack program could be inserted into the executables and libraries after they have been built from reviewed code.
- The attack program could be hidden within the operating system using rootkit-like techniques, or perhaps a commercial rootkit for the underlying operating system.

- The attack program could be stored in some data file which is not reviewed, but which is read by a program with a subtle bug of some kind, allowing the program in the file take over the program reading it.

Further, we are deeply skeptical of the ability of the testing labs to review all the software in the voting system carefully enough to catch all possible attacks. Even if obvious attack behavior doesn't remain, either intentional, subtle bugs or subtle attack behavior (e.g., messing up the touch screen alignment after certain user interactions from voters) may still remain despite the testing lab review.

Finally, it's worth noting that tampering with the software in the initial voting system is not limited to programmers working for the voting system vendor. COTS software writers, who may themselves be contractors or subcontractors of the original company from whom the COTS software was purchased, are in an even better position than voting system programmers to insert an attack program. This is especially true for drivers written for devices that are mostly used for voting systems. Further, anyone able to get access to the voting system software either during design or after it has been reviewed and before it has been installed on the voting machines may install an attack program. This might include people with full access to the software during development, storage, or testing.

4.2. Software Patches and Updates

COTS software often has patches and updates which are required for security. Voting software can also require updates, either to fix bugs or to extend functionality in some way, e.g., by supporting more assistive technology or a larger set of screen characters for alternate-language voting. This is an obvious attack point if the updates are not secured. The attack program may be inserted by someone working for the COTS software vendor, or by someone working at the voting system vendor, or the election official handling the installation of patches and updates.

4.3. Configuration Files and Election Definitions

If the voting system software is vulnerable to attack, an attacker may be able to take over the machine by improperly formed files. (For reference, a very successful e-mail virus used a flaw in the WinZip engine on many people's PCs to mount an attack of this kind.)

4.4. Network Communication

Some voting systems use wireless or wired network connections. If there is a vulnerability in the configuration of the voting machines, then this can allow an attacker to insert an attack program.

4.5. Device I/O

Some voting systems involve the use of an external device such as a memory card, printer, or smart card. In some cases, access to these external devices has allowed attacks to be demonstrated in the laboratory. (The RABA report gives one example, and the Diebold optical scan attack gives another.)

This is not meant to be a complete listing, and it necessarily leaves out a lot of detail to cover so many different machines. However, it is important to recognize the large number of possible attack points. In general, we expect that two-step attacks make the most sense to apply at the top, either in the original voting system software or in patches sent around to many different voting machines. The remote control aspects of these attacks makes it possible to control the attack, so that it doesn't trigger in obviously unreasonable ways and get detected. On the other hand, the more local attacks fit nicely with a one-shot attack; the attack programs can then be very simple and focused on a single election in a single state or county.

5. Avoiding Discovery

One of the most basic problems for a software-based attack is how to avoid detection. The tampering program must avoid discovery to successfully alter the election. Also, the attacker will have a strong interest in avoiding detection, since an investigation may determine that he is responsible for the attack.

5.1. Insert, Delete, or Modify Votes?

In most cases, the most effective way to tamper with an election will be to change votes that have actually been cast; this avoids introducing a disagreement between the number of votes reported by the voting machine and the number of registered voters allowed to vote. In the case of a DRE voting system, changing votes electronically changes all the records of the voters' intent which are formally available to the voting system, and so this kind of attack cannot be directly detected by comparing the electronic totals with other records. In the case of other voting systems, such as DRE+VVPAT or PCOS, the attacker must also tamper with the paper records or prevent their being cross-checked against the electronic records.

By contrast, inserting or deleting votes introduces a disagreement between the number of registered voters allowed to vote, and the number of votes recorded electronically.

Some attack scenarios may require a predefined sequence of electronic votes to be produced by the tampered software. In this case, it's not feasible to selectively change votes. However, the attack software can start with a predefined sequence of electronic votes to be stored, and record the first *N* votes from the sequence, where *N* is the number of votes actually cast on the voting machine.

5.2. Deciding How Many Votes to Change

An attack may also be detected by too strong a disagreement between informal numbers (polling data, for example) and reported election results, though it isn't clear what procedure would be needed to invalidate an election based on this kind of evidence alone. We think one of the most likely scenarios for this kind of attack to be detected is for some event to happen which radically changes the expectations of the election, just before the election takes place. The death, indictment, or complete discrediting of one candidate a few days before the election offer an opportunity for an inflexible software attack on a voting system to be revealed.

This leads to a few natural approaches for an attack program to minimize its chances of discovery:

- Where possible, there should be some way for the attacker to control which voting machines alter votes and for which races. The attacker should use this to minimize his attack's "footprint" while still leaving the attack likely to succeed. (This is discussed in a different context below.)
- Where possible, the attack program on the voting machine should change a fixed portion of the votes, e.g., move 5% of the votes for John Smith to Mary Jones, rather than simply reporting a preordained result. This avoids the situation where a dead or recently indicted candidate mysteriously wins a few precincts, while losing badly in all others, revealing the attack.
- The attack program should notice when the tampering is hopeless (e.g., when the election appears so one-sided that the benefit of improving the favored candidate's outcome is outweighed by the cost of increased chance of detection from implausible results. In that case, it should refrain from any tampering at all, since this implies a risk of detection with no corresponding chance of success.

5.3. L&A and Parallel Testing

Tampered software must avoid detection during testing. There are a number of techniques to use to ensure that testing does not detect the attack program.

- The attack program can note the time and date, and only trigger when the time and date are consistent with an election. This prevents detection during L&A and acceptance testing, but not during parallel testing. Further, a tester may attempt to reset the machine's clock; however, a resettable machine clock may open up other vulnerabilities in the voting system
- The attack program can observe behavior which is consistent with a test vs. with real voting. For example, if L&A testing in a given place is known to never go on for more than four hours, the attack program can refuse to trigger until the 7th hour of voting. Note that this is strongly affected by the nature of the attack program's operation, the nature of testing that is ever done, and the nature of voting in a specific place.
- The attack program can activate only based on some communication with the attacker or his confederates. For example:

  - Some specific pattern of interaction between the voter or election official and the voting machine may be used to trigger the attack behavior. This is often called a “secret knock.”
  - Any of the control communications channels described later in this document may be used to turn the attack behavior on or off.
- The attack program can wait for a remote interaction with the attacker before deciding whether to tamper with stored electronic votes.

5.4. Avoiding Event and Audit Logs

Tampered software must not leave telltale signs of the attack in any event or audit logs. In principle, this could be pretty difficult. However, this depends on the nature of the attack program:

- Tampered user-interface software may simply display the wrong things to the voter, while not causing any other system events. In this case, there will be no trace of the attack in the event log.
- Tampered driver software for storage devices or tampered BIOS can alter what is written to the storage devices.
- Tampered operating system or other high-privilege-level software may be able to entirely bypass the event logging mechanisms of the operating system or tamper with the logs after entries are made.
- Tampered operating system or other software may simply provide a different log to the outside world than the one stored internally, if the log is not stored on removable or write-once media.

5.5. Coordinating with Paper Record Attacks

When the tampering program must support the attacker tampering with paper records as well, it is often going to be useful for the attacker to be able to prepare replacement paper records before the voting is completed.

There are two interesting variations on this problem: First, when a DRE+VVPAT system is using a paper roll, the electronic and paper records must be identical. Second, when the paper records or ballots are kept separate, an attacker need only produce paper records that agree, not necessarily do so in the right order.

This coordination task can be solved in a number of ways:

- The attacker may simply wait until the electronic results are ready, and then print the replacement paper records. This raises some logistical problems for the attacker.
- If the attacker is in contact with the voting machine during the voting process, for example over a wireless network or via an exposed infrared port, the attacker can print replacement paper records as the tampered records are produced on the voting machine.

- The attack program can have a predefined sequence of votes, which it produces electronically and which the attacker also prints.
    - The attack program can decide based on its observed conditions whether to alter its electronic records or not. Thus, if the attack program observes a very one-sided election, or a voting pattern more consistent with parallel testing than with normal voting, it can simply not carry out the attack. The attacker must then determine which thing has happened, and replace the paper records or not depending.
- The attacker can communicate with the voting machine after the voting has ended but before the votes have been displayed to poll workers or sent to the tabulation center. In this case, the attacker can tell the voting machine what totals to report and store.

5.6. Forensics and Postmortem Analysis

Perhaps the hardest test for an attack program is some kind of forensic analysis. In this case, I imagine that an attack is strongly suspected, and competent people are analyzing the machine at great depth to find evidence of the attack program. In the most extreme case, this would involve making bit-level copies of everything on the machine, and taking the machine apart in a lab to verify that everything looks exactly as it should.

Drawing from experience in other areas, I expect that while a reviewer who doesn't know an attack exists will probably not find a competently implemented attack program, a reviewer who knows or strongly suspects that an attack has occurred, and who is allowed to destroy the machine in the quest for evidence, will probably discover that evidence.

The attack program has a number of techniques at its disposal to avoid detection by less thorough reviews, however, including:

- Hiding from operating system utilities attempting to scan files and memory, in the manner of existing stealth viruses and rootkits.
- Hiding the active part of the attack program in obscure places, such as data or configuration files read by programs running at high privilege levels (exploiting a weakness in those programs to take them over when they read an improperly formed configuration file), or driver software stored in EEPROM or flash memory.
- For one-shot attacks (see below), simply deleting all attack programs from the system after the attack is carried out, if this is possible. (This requires the use of some kind of secure delete, and even the secure delete will likely leave evidence of something interesting happening, but it is clearly possible.)
- For attacks based on backdoors, the whole attack may take place entirely in RAM, with no altered programs.
- For attack programs embedded in original COTS software or voting system software, scanning memory contents, even in a way that resists attack programs' tampering with what is seen, will not detect anything wrong.

After all this, I return to my initial assessment: a thorough, destructive analysis of the voting machine is quite likely to find evidence of an attack, especially if the broad direction of the attack is known or suspected from other evidence.

6. Conclusions

In this white paper, I have discussed a number of operational difficulties an attacker might have in compromising an election using some kind of tampered software on the voting machine. While the difficulties are real, they are not impossible to overcome, and I've tried to show at a high level how this might be done.

Software attacks are a major potential threat to voting systems because it is possible for them to be so well hidden that no outside observer ever notices the attack behavior, and because these attacks, unlike so many other potential attacks on voting systems, may be mounted with a very small set of people.

All Threats

David Biddulph

September 26, 2005

Taxonomy

Any and all threats, both intentional and accidental, to the accuracy of the vote tabulation.

Applicability

All voting systems that can export a ballot image file incorporating a 19 digit random character password in XML format and include a voter verified paper ballot such as DRE's with a ballot printer and Optical Scan.

Method

An individual or group may accidentally or intentionally alter the accuracy of the vote tabulation.

Likelihood of Detection

The likelihood of detection is very high because every voter would be able to confirm the accuracy of the vote tabulation.

Countermeasures

Preventative Measures

> Separate the voting process from an open source, peer reviewed, unaltered copy of a generic vote tabulation software that allows each voter to confirm that their secret vote was accurately tabulated without allowing the voter to prove how they voted to an unauthorized third party. Election jurisdictions would obtain an unaltered copy of the vote tabulation software from the NIST software library.

Detection Measures

Detection is the primary responsibility of the voter. Because each voter could detect that their vote was missing or altered, the chances that a perpetrator could alter an election undetected would be greatly diminished.

Citations

WHITE PAPER: THE EROSION OF CONFIDENCE IN THE NATION'S VOTING PROCESSES AND A NEW SOLUTION TO ADDRESS THE PROBLEM

PREPARED BY DAVID BIDDULPH, DEVELOPER OF THE *PERFECT VOTING SYSTEM*, FOR THE U.S. ELECTION ASSISTANCE COMMISSION (EAC) MEETING IN DENVER, COLORADO, AUGUST 23, 2005

The Perfect Voting System (PVS) can be implemented on any electronic voting device that includes an accessible voter verified paper ballot and "stub." The printed ballot and stub could include a bar code, such as PDF 417, so that its contents could be viewed and "heard" via a bar code scanner and audio device.

DESCRIPTION OF "THE PERFECT VOTING SYSTEM"

The heart of the PVS is a voter-verified paper ballot and "stub" or receipt, which includes a 19 digit random alpha-numeric character string. The size of the random alpha-numeric character string has been mathematically designed to reduce the odds of two independent computers producing the same character string in a ten million vote election to less than one in a hundred billion.

The first step is for the voter to make his or her selections on an electronic voting device or on a preprinted ballot, such as an optical scan or mail-in ballot. If the voter finds a mistake or changes his mind, the 19 digit random character string could be used to amend the ballot. Once the voter is satisfied that his ballot is accurate, he would be required to deposit it in a locked container (the same as with paper ballots and punch card ballots in past elections), then sign and retain the "stub" as his proof of having voted and his connection to a specific voter verified paper ballot.

The ballot and "stub" include the 19 digit random alpha-numeric character string, precinct number and -- in any jurisdiction using a direct recording electronic voting device (DRE) such as the now familiar touch-screen machine -- a machine number, date and time stamp, To maintain the secrecy of the ballot, the stub does not indicate the voter's ballot selections.

At the conclusion of voting, each precinct would download from each voting or tabulation device all the ballot summaries in an electronic database or spreadsheet format. The files would be forwarded to the jurisdiction's vote tabulation headquarters. In addition, the summary data from the "poll book," which lists the registered voters and contains the signatures of those who voted, along with a detailed list of voters casting ballots, would be forwarded to headquarters. The ballot and "poll book" data would be imported into the PVS vote tabulation database. The PVS vote tabulation database should be hosted on a computer that is only accessible through the election jurisdiction's official secure intranet.

To ensure accuracy and voter confidence, the PVS would produce several critical error reports including:

- The number of votes cast in a precinct compared with the number of voters signed in on the "poll book."
- The number of votes cast in a precinct ranked from highest to lowest.
- The number of votes cast on a machine ranked from highest to lowest.

Once all the votes were entered into the PVS tabulation database, any voter would have the right to audit their vote. Simply by accessing a PVS enabled election system using 19-digit random character string printed on their ballot stub, they could view their ballot data and confirm it was entered as they voted and that it was counted exactly as intended. To eliminate the possibility of the PVS being used to verify a vote in "vote selling" scheme, the voter would be required to have their identity and ballot stub signature verified by an election official before privately viewing the secure PVS tabulation database.

The first screen a voter would see upon accessing the PVS vote tabulation database would be their own ballot summary. The voter can then select any contest and see how their vote was counted. They could also view all the other ballots cast but without the identify 19-digit random character string. A spread sheet of the entire jurisdiction, or any portion of it, could be viewed, but not printed. The names of all the voters who cast a ballot would also be accessible, but not their votes. Voters, for example, could see how their votes were counted by machine, precinct or for the entire election jurisdiction. Sub totals and totals could be checked. The names of neighbors who voted could be checked for accuracy. Media representatives, candidates, party official, campaign workers, and political scientists would be able to audit the PVS vote tabulation database.

If a voter believed their vote was missing or altered, they would have the right to view the paper ballot retrieved from the lock box of the precinct indicated on the stub and matching the 19-digit random character string. In the highly unlikely event that the PVS tabulation database is found to be in error, the voter-verified paper ballot would be used as the official record for any recount.

If the voter disputes that the ballot summary with the matching 19 digit random character string was actually theirs, forensic research could be used to prove or disprove that the ballot summary was only handled by the protesting voter.

Retrospective

Despite numerous congressional hearings, enactment of new laws, and the expenditure of billions of dollars of federal, state and local government money for new voting systems and equipment, confidence in the America's voting systems is declining, instead of increasing as hoped for by the reforms that followed the 2000 election. In a recent survey of Georgia voters, for example, fewer than half the voters (48 percent) were "very confident" that their vote was counted accurately. That is down from 56 percent of voters who were "very confident" their vote was accurately counted in 2001, and after the state abandoned the discredited punch card system for touch-screen electronic voting machines. Even more alarming is that the level of confidence was only 33 percent among minority voters in the most recent Peach State Poll.

This represents only one example of the erosion of voter confidence in the election systems in place across the United States. In the 2004 election, for example, 32 states used some type of computerized voting. But according to the Verified Voting Foundation, a nonprofit, nonpartisan group in San Francisco, 1,700 complaints were filed.

A lack of confidence in the bedrock of our democratic system, and a failure to resolve voting system problems despite mandates and money from congress does not bode well for democracy in this country. In the view of David Biddulph, political activist and inventor, what has been missing from all the electoral reforms is the element of transparency. "There's more transparency in the country's lottery systems than in its voting system," he observed. For this reason he has developed a vote tabulation system that will make elections, in any jurisdiction in the country, as transparent and trustworthy as the state lotteries that collect and distribute billions of dollars with nary a dispute or question as to the legitimacy of the winners and losers.

Granted that more is at stake in our elections than in any lottery, the fact that people are more willing to accept a decision that they did not win a multimillion prize than they are the outcome of a gubernatorial election says a great deal about how people view the relative trustworthiness of the two processes.

To address this lack, and continuing decline, of confidence in the nation's electoral process, Mr. Biddulph has developed a vote tabulation system designed to make elections completely transparent to the citizens, media, political parties and candidates. In fact, it would result in a voting system as transparent as the lottery. Dubbed the "Perfect Voting System" (PVS), a patent pending business process, it would enable voters to confirm that their individual votes were counted exactly as they intended them to be.

(The following statement was delivered to the Joint Committee on Election Laws of the Massachusetts General Court on 19 July 2005.)

I hold a Master's degree in Computer Science from MIT and have over 30 years experience in programming and computer systems consulting, most recently in wireless and network security. I am retired from Hewlett Packard and am now a principal in a software startup.

The word "machine" is used to refer to any computer-based election systems, including Direct Recording Electronic ("DRE" -- commonly also called "touch screens"), Optical Scan, and central tabulating systems.

Software IS a Problem

As a computer professional, I find myself in a bit of an awkward position arguing against the use of computers in elections. My position is this: computers are extremely helpful, even necessary, to solving many problems; but computers are not essential to elections, and the risks are just too great.

We would be unable to use our ATM cards to access our bank accounts from around the world without computer-based funds transfers. However, there are risks associated with all those computer systems and transmission links carrying all those funds transfers, and extraordinary measures are taken to avoid the risks, and to detect tampering or other breaches of security. For modern financial transactions, there is no alternative to the use of computer-based systems.

Computer based systems are not essential to the conduct of elections. Many large democracies in the developed and developing world conduct their elections without computer-based systems. There are basically only three real benefits to the use of computers in elections. One is that results are available a few hours earlier. The second is that certain accomodations can be made for voters with disabilities. The third is that certain errors made by voters in marking their ballots can be detected and the voter informed so that a correction may be made at the polling place. Getting early results is an extremely minor benefit that must be weighed against the dangers, which I will outline below. The other benefits, accessibility and checking for voter errors, can be provided without using computer systems in the *counting* and *tabulating* process.

Software is a powrful medium for solving problems. Software can be duplicated easily and instantly. Software is what makes a computer-based system perform its functions. Anything that a computer-based system can do is performed at the command of the software running invisibly inside, perhaps transmitted the instant before from somewhere else.

As a result, *software is a powerful medium for creating problems.* A software defect can cause any kind of malfunction. Both pranksters and saboteurs love to work with software. Malicious software can take advantage of phone lines and networks and memory cards and discs to transmit itself to other systems. Malicious software can lie in wait -- even for years -- before doing its evil deeds. Malicious software can cover its tracks and even erase itself after the deed is done.

In my work as a computer systems consultant, I must assume that attempts will be made to attack, compromise, and invade any software-based systems I design.   I must be humble enough to assume that a clever prankster or saboteur may overcome my best defenses. As a result I design systems to check both for innocent errors that WILL occur and deliberate tampering. I must always check for intrusions and failures, and the system must be designed so that *reliable independent and original records* are maintained so that a meaningful check can be made.

It Only Takes One Person

One person acting alone can cause many computer-based machines to malfunction.

One person can write a piece of software (a "virus" or a "Trojan horse" -- we'll call them generically an "intrusion"') that can corrupt any number of machines. A machine can be infected at any time before an election. Software can even be infected before it is put on the machine, even at the factory.

Intentional sabotage (by an authorized programmer) is also always a possibility. Consider that the software is held to be a "secret" by the vendors; this possibility cannot be dismissed.

Well-intentioned programmers sometimes make provisions in the software for "maintenance"; while not directly malicious, such provisions can subsequently be exploited to alter the software in malicious ways.

Any connection, permanent or temporary, can be exploited to transmit an intrusion. By "connection" I mean a computer network, a phone line, a memory card, a disk, or wireless communication to an internal device. (Note that the person actually establishing the connection, for example, inserting a card, may not know that a software intrusion is being transmitted -- as far as they know, it is an innocent maintenance or data retrieval operation.)

Given the attraction and high value of election tampering, we must assume that tampering will be attempted, and that it may sometimes succeed in spite of our best efforts. Thus if we were to use computer-based systems we would have to take measures to *detect* tampering with election machines, and we would have to implement procedures to *recover* from this tampering. However, as I will show, tampering is surprisingly hard to detect, even harder to prove, and after-the-fact recovery mechanisms may be ignored.

It Only Takes One Vote (per Machine)

In 2004, Ohio didn't appear to be that close, but a shift in just one vote in 87 would have changed the outcome of the presidential race in Ohio and, thus, in the US. This would need only a handful of changes <u>per machine</u>.

A change in many machines is as easy as a change in one machine. Thus the pattern we are more likely to experience -- but less likely to notice -- is one of many small discrepancies on many machines.

Who would do this? This tampering could be accomplished by a single individual or a small band -- a prankster, a disgruntled employee, an unscrupulous campaign worker, a vendor that is over-zealous in its support of a candidate, organized crime, a foreign power, or a terrorist group -- anyone with an interest in or desire to see a particular outcome in any US election, or perhaps just wanting to create chaos.

Tampering is hard to detect

Software in a machine is hard to see, and hard to fully understand, even for experts! Software intrusions can accomplish any effect; in particular they can mimic "glitches" and human error.

Since many machines can be infected, and since only a small change in result is needed per machine, the tampering is easy to miss or overlook. For example, someone can switch whom votes are for, but keep the total number of votes cast the same. This kind of insidious small change is easy to ignore, or easy to dismiss as "insignificant."

There were tens of thousands of reported small computer problems in 2004. But we don't know how many additional problems were never reported because they were not noticed or they were considered "insignificant."

Some kinds of tampering might look quite harmless -- for example, an occasional "default" vote (which has the side effect of a "higher quality" election -- fewer "undervotes"!)

Another kind of "innocent" tampering is one that doesn't alter votes and thus cannot be detected by any kind of auditing. The election can be biased against certain precincts by software tampering that causes the machines in those areas to slow down or crash. If these precincts are chosen to be precincts that favor one particular candidate or party, such tampering will cause that candidate or party to lose votes. *This is why I discourage any thought of "auditing" and "paper trails" as solutions to the threats against electronic voting.*

We cannot assume that fraud would be "obvious" if it were serious enough to change the outcome of an election. Software intrusions can cover their tracks, even erase themselves when done -- only the altered election result remains!

Can machines be made more secure?

The very nature of computer-based systems makes the above risks possible -- one person making very small changes in many places without leaving "tracks" is just not possible with paper!

Today's computerized voting systems are very poorly designed with regards to security -- passwords are widely known and are rarely changed, breakable forms of encryption are used, and systems are connected to networks, phone lines, and memory devices without "best practices" in security. Once they are delivered, election systems are rarely under tamper-proof seal from the point at which known certified software is loaded.

Regarding certification and testing: it is a maxim in computer science: *"Testing can only show the presence of errors, never the absence of errors."* Likewise, testing cannot prove the absence of malicious code or the absence of opportunities for intrusion. Testing the software is not a solution.

Some of the problems with computer-based systems have technological fixes, but only at the cost of increased complexity, rendering the systems beyond the knowledge of all but a handful of experts. All of us non-experts would simply have to trust that these systems had not been compromised.

A quote from computer science Professor David L. Dill of Stanford University, is sums up the problem quite well:

> *Why am I always being asked to prove these systems aren't secure? The burden of proof ought to be on the vendor. You ask about the hardware. 'Secret.' The software? 'Secret.' What's the cryptography? 'Can't tell you because that'll compromise the secrecy of the machines.'... Federal testing procedures? 'Secret'! Results of the tests? 'Secret'! Basically we are required to have blind faith.*

I can assure you, even if *nothing* were secret, it would still be a practical impossibility to *prove* the security and reliability of a state-of-the-art electronic voting machine.

On paper trails and auditing

Note that in systems that print a "paper trail", the paper trail itself is created by software that may be altered by tampering or error, and thus is unreliable as a record. Having the voter review the paper trail is an attempt to fix this additional problem, but it is an attempt that is likely to fail.

A "voter verified paper audit trail" is a problematic attempt to create the equivalent of an original document -- using, in part, the system being audited to create its audit document! That document itself must then be "audited" by the voter. Such an "audit trail" is certain to be an accurate reflection of what the voters selected only if 100% of voters check 100% of the votes 100% correctly -- an impossibility in real situations. Otherwise we start out with an audit record that itself cannot be assumed to be 100% correct, resulting in a less than useless sham of an audit.

With a printed paper trail we also have the problem of "what if we find a discrepancy"? If we only see one or two discrepancies per machine, would we do anything about it? Would it just be treated as a glitch, written down and forgotten? Would that one machine be taken out of service -- but what about the votes it already "counted"? What about the other, presumably similar machines on which no voter reported a discrepancy? Remember, all it takes to steal an election are a few discrepancies per machine.

The "Political Realism" Problem

The first "official" results create a strong presumption of the correct result. How compelling would any evidence of tampering have to be to work against that presumption? Our efforts must be directed towards limiting the opportunities for tampering in the first place. If possible, the counting process itself should produce the first crosscheck or audit of the result. One way to accomplish this is to immediately count the ballots twice by two independent teams. Detection of tampering is always necessary, but experience shows that evidence of tampering won't always change a tampered result if it is delayed.

Simplicity, Transparency, Openness

Our voting systems must be simple enough so that non-technical observers can see what is going on. They must be transparent and open enough so that, once the ballot is cast in secret, the rest of the process is observable by the public, and all intermediate results are open to checking by all. Our election systems must be designed so that the secret actions of a few cannot have an effect without raising suspicion. (It is unfortunate that in some jurisdictions you will be arrested if you try to observe the vote counting process.)

Paper systems can be made to meet these criteria easily; computer-based systems cannot. When this fact is combined with recent results that show a lower error rate for hand-counted paper ballots, is there any reason to consider machines?

Only a hand-marked paper ballot is an original documentation of the voter's intent. We must have that at a minimum, and if we have that, there is no reason against (and many reasons for) counting those ballots in an open, public process that is visible and understandable to all.

Software based systems introduce many opportunities for problems, including tampering. "One person, one vote" must be a principle of democracy, not a description of all it takes to steal an election!

Bob Fleischer
119 Nashua Rd
Groton, MA 01450
(978) 448-6151
AOL IM: rjf7r
www.caef.us

September 30, 2005

National Institute of Standards and Technology
Gaithersburg, MD

Re: Developing an Analysis of Threats to Voting Systems

We are writing on behalf of the members of the Information Technology Association of America's Election Technology Council in response to your call to the elections community for submission of threat analysis papers and for participation in NIST's upcoming workshop: "Developing an Analysis of Threats to Voting Systems."

The Election Technology Council (ETC) is a group of companies that offer products and services which support the electoral process and have decided to work together to address common issues facing the industry. Members of the ETC currently operating in the voting systems business are: Advanced Voting Systems, Danaher Guardian Voting Systems Diebold Election Systems, Election Systems & Software, Hart InterCivic, Perfect Voting System, Sequoia Voting Systems, UniLect Corporation and VoteHere.

Our member companies applaud NIST's work in driving the development of the Voluntary Voting Systems Guidelines and its array of meetings, workshops, and papers designed to facilitate the evolution and improvement of voting systems standards, infrastructure, and processes serving America's voters. In fact these member companies have all been participants in this process of creation and development of new voting system guidelines.

While every member of the ETC has a strong interest in working to promote voting systems security, the ETC group serves as an advocate and representative for a wide array of companies with a huge assortment of products and services offered in the marketplace. We recognize that identifying, describing, and cataloguing every realistic threat to those systems and services will be an immense undertaking. ITAA and our ETC member companies will support this effort to the best of our abilities. You will see many of the member companies' representatives at this meeting, in which each may provide direct input or participate. However, at this early stage, it is impossible for us to provide meaningful comments as a group. We will offer an ETC group response after the meeting when we have a sense of the issues and breadth of scope that the security threat analysis project will include.

In developing a framework for this project, we would respectfully request that NIST and the voting community look to the guidance of the Help America Vote Act (HAVA) of 2002, which recognized **both** process and infrastructure shortcomings in the American election system.

Threats to voting system security are not confined solely to electronic or software-based systems. The physical act of casting a vote on a voting station is just one facet of a much larger and more complicated process. Also, the history of voting in America is replete with examples of attacks, fraud, and tampering committed on paper-based systems. We believe that a comprehensive view of voting systems and the processes in which they are used is a prudent approach.

Again, our members look forward to working with NIST as the process of identifying, describing, and cataloguing threats to voting systems goes forward. We ask only that this effort provide a comprehensive view of the entire array of systems and processes that make up this country's election system. In the end, we believe that such an approach will yield a more valuable and useful work product that should instill even greater levels of confidence in America's elections.

Sincerely,

Harris N. Miller
ITAA President

John S. Groh
Chair, ITAA Election Technology
Council

Paper v. Electronic Voting Records – An Assessment

Michael Ian Shamos[1]
School of Computer Science
Carnegie Mellon University
April 2004

Abstract

There has been much discussion in the popular press concerning the use of contemporaneous paper trails to plug various perceived security risks in electronic voting. This paper examines whether the proposed paper solutions in fact provide any greater security than properly maintained electronic records. We conclude that DRE machines pose a number of security risks but that paper records do not address them. A number of alternatives to paper trails are suggested to respond to DRE security concerns.

1. Introduction

Among the arguments that have been advanced against the use of direct-recording electronic (DRE) voting systems are the following:

1. Voting machines are "black boxes" whose workings are opaque to the public and whose feedback to the voter is generated by the black boxes themselves. Therefore, whether or not they are operating properly cannot be independently verified and the machines should not be used.
2. No amount of code auditing can ever detect malicious or even innocently erroneous software. Therefore the machines should not be used.
3. No feasible test plan can ever exercise every possible combination of inputs to the machine or exercise every one of its logic paths. Therefore the machines should not be used.
4. Hackers can break into the FBI's servers and deface its website. It ought to be child's play for them to throw an election. Therefore the machines should not be used.
5. DRE machines have been plagued by a host of failures all around the country. Therefore the machines should not be used.
6. The DRE industry is dominated by a small number of companies, some of whose executives are announced supporters of the Republican party. An executive could command his programmers to add code to each machine manufactured by that company to move votes to a favored candidate, thus determining the outcome of the election. Therefore the machines should not be used.
7. Many prominent computer scientists have said that DRE machines cannot be trusted.

Therefore they should not be used.[2]

8. If added to a DRE machine, a voter-verified paper trail allows the voter to satisfy herself that her voting preferences have been recognized correctly by the machine. Therefore, the voter-verified paper trail solves every one of the aforementioned problems and every DRE machine should be required to have one.

Each of these arguments will be examined in this paper and found fatally flawed, at least to the extent that it implies that machines cannot be relied upon to count votes in real elections. The numbered statements above all share the property that the first sentence of their premise is true, yet their consequent, that DRE machines should not be used, does not follow from the premise.

In 1993, I prepared a paper for the Computers, Freedom and Privacy '93 conference exploring the risks of electronic voting[3]. Since then, I have often been asked whether I still adhere to the opinions expressed in that paper in light of the incidence of widespread hacking, Internet worms and viruses, new cryptographic attacks and the increased used of DRE machines around the world. The answer is that I still hold those opinions but feel compelled to update the justification for them to respond to the arguments raised above.

Since the Industrial Revolution, man has chosen to rely on machines for tasks that are either impossible for humans to perform, or so expensive or repetitively boring that there is no justification for continuing to waste human labor on them. Many of these machines, such as cars, airplanes and therapeutic radiation equipment, among numerous others, have the capacity to take human life. They also commonly contain embedded computer systems. In the business world we rely on computers to execute financial transactions totaling at least $2 trillion per day. It is well-known that all of these systems present risks. There are approximately 40,000 deaths annually in the U.S. due to automobiles[4]; some number of the victims are killed by malfunctioning software rather than human error. People have also been killed by the computer programs that control radiation machines[5]. In light of such failures, why do we continue to drive cars, fly on planes and receive radiation treatments? Why hasn't the government outlawed these killing machines?

The reason is that testing and safety procedures are in place that reduce the risks to levels that are deemed acceptable. There is no basis for applying different reasoning to voting machines. Once we decide what a tolerable risk in such systems might be, we can require that the equipment meet that standard. Perfection is never required, expected or even possible in any real system, though it is a laudable aspiration, and perfection is not required, expected or possible in voting systems, either. Federal Election Commission Standard 3.2.1 allows a maximum error rate of 1 in 500,000 voting positions[6]. With a typical ballot size of 235 positions, this is an allowed error of almost one in every 2000 ballots, or 0.2% of the vote.

When the safety procedures are found to have flaws, the flaws are ultimately corrected because of public pressure, government mandate or the relentless law of the marketplace. We are now seeing immense public pressure being put on voting machine manufacturers, along with threats to legislate, both of which are appropriate.

A secondary reason that machines presenting some risk of injury are not outlawed is that people generally have the option not to use a particular machine. This choice is also available to a voter, who may eschew voting machines completely and cast a paper absentee ballot.

While the United States has been using direct-recording electronic voting equipment for well over 20 years without a single verified incident of successful tampering, within the last year a number of people knowledgeable about computer security have questioned whether certain DRE systems in current use are sufficiently secure to be employed safely in elections. Some criticism of these systems resulted from examination of their source code, perceived flaws in their handling and use or from consideration of purely hypothetical scenarios. A calm observer might take solace in the observation that if DREs are so dangerous, then surely at least one security hole would have manifested itself by this time. Realistically, however, hacking has been advancing at a alarming rate, and new attacks are constantly being discovered, so we are entitled only to a small bit of comfort from DRE history.

It is an error, though, to ascribe to DREs generally the bad attributes exhibited by some of them. The spectrum of available systems is broad. Some machines are excellent, some are terrible.

1.1. The “Black Box” Phenomenon

That a machine contains a computer and the computer contains object code not readily viewable or understandable by the public is by itself no reason not to use the machine. If it were, no one ought to own a personal computer. Neither passenger nor pilot can see or understand the software that operates the control surfaces of a jet plane. Such software could contain code, malicious or otherwise, that might send the plane into a dive at noon on a specific date from which the pilot could not recover. How do we know for a fact that such code is not present? We don’t. Yet pilots and passengers continue to board planes every day. Let’s look carefully at the reasons we allow jets to operate. All of them apply to voting systems as well.

1. It is beneficial to aircraft manufacturers to make safe planes. Planes that crash will not sell and will eventually be outlawed, not to speak of the legal liability associated with such incidents. This benefit induces the manufacturer to develop internal procedures designed, but not guaranteed, to produce safe products. It is beneficial to voting system vendors to make safe systems also. Whether they know how to do so, or have successfully implemented procedures for doing so, is somewhat questionable. In examining more than 100 different voting systems for certification purposes, I recommended that over 50% of them be denied certification. The quality and reliability of particular DREs is certainly a matter of concern, and later in this paper various solutions will be suggested.

I have heard it expressed that it might not be beneficial under certain circumstances for a voting system manufacturer to produce an honest machine, but that substantial gain could be achieved by distributing machines or software altered to cause the election of specific persons who may not actually be favored by the electorate. We will discuss below the practical difficulties with such a scheme, but if a manufacturer felt that its underhanded activities would not be discovered, such a fraud might be attempted despite the possibility of severe criminal penalties[7]. Therefore any plan for the administration and use of voting machines should contain safeguards against this type of manipulation.

2. Planes are built to high performance and engineering standards. Agreed. Voting machines, which are far simpler than airplanes, can be (but are not always) built to even higher performance and security standards.

3. Planes can be tested. So can voting machines. Neither needs to operate perfectly. Planes shouldn't crash much and neither should voting machines.

4. If a plane crashes, we'll know about it. The significance of this statement, made by DRE opponents, is that we would then at least be able to take remedial action to prevent a recurrence, a fact of little consolation to the victims' relatives. The argument is made that election can be stolen under our very noses and no one would be any the wiser. But that ignores the real political fact that elections are local and local party operatives have an extremely accurate sense of how the community is going to vote. The smell of irregularity is sufficient to set off alarms resulting in investigations and recounts. DRE opponents claim erroneously that in a disputed election there is nothing useful left to recount since all the records that remain were made by the malfunctioning machine. But this argument is wrong because the software that was used in the machine survives. (We can deal later with the assertion that the software might modify or delete itself to evade discovery.)

5. The people who fly airplanes have a vested interest in their safety. The people who run voting systems are likewise committed to clean elections. Pilots have been known to crash planes deliberately and election officials have been known to manipulate votes. Safeguards need to be built in to prevent both of these efforts from succeeding.

In short, I am unable to discern any engineering difference that allows us to entrust our lives to aircraft but would impel us to avoid voting machines. Not to endorse questionable voting systems or trivialize the possibility of chicanery, but I believe I and the republic will survive if a president is elected who was not entitled to the office, but I will not survive if a software error causes my plane to go down.

1.2. Computer Security

It is pointless to discuss the security of a computer system in the absence of a well-articulated list of threats. So let's enumerate and deal with them in order.

1. Isolated attacks on individual machines. There are any number of ways of interfering with the operation of any computer system, such as pounding on it with a sledge hammer or the

slightly more sophisticated technique of exposing it to several watts of radio-frequency emission. Such efforts fall into the class of mischief rather than tampering because they cannot be used to cause a predetermined result.

A different form of attack is to gain access the hardware or software of an individual machine or small number of such machines and alter them, either by connecting to ports and interfaces or by opening the machine by force or with the help of an insider who may have the keys, along with manuals, plans and source code listings for the machine. It should be obvious that no machines should be used that allows any voter to connect to it electrically to during an election and any device that permits this should be decertified immediately. The question is how to prevent people from modifying the machines offline or at least to be sure the tampering will be detected before the machines are used.

One solution is to ensure that all software needed to operate the machines, including the operating system, is not installed in the machine until election day. The authorized, certified software, distributed from a central authority (not the manufacturer), can be brought up at the time the polls are opened. In this way no advance modification of any software would be fruitful. If it is deemed undesirable to do a full machine boot, a portion of the code can be loaded on election day and verify through message digests and encrypted checksums that none of the prestored files has been altered.

2. Attacks by hackers or insiders at a polling place. The tendency to use networked voting machines at polling places for ease of administration also increases the risk that an insider could use a computer connected to the network to distribute malware to the voting machines after the election has begun. The miscreant would presumably remove the malicious code or restore the original at some time before the end of voting so that no trace would remain of the misdeed. This sort of attack presupposes that the insider is able to erase evidence of his deed during the election, for if the altered software is still present in the machine at the close of polls it can be detected. It also is a highly localized manipulation that affects the results at a single precinct only.

3. Attacks by hackers or insiders at a central count facility. Now the magnitude of the problem grows because the number of votes that are potentially affected can be extremely large. There are 35 counties (out of a total of 3170) in the United States with populations exceeding 1 million[8]. The total population of these counties is over 73 million, approximately 25% of the country's population. A successful attack on central count systems in these 35 counties, (representing just 1.1% of the total number) would certainly influence any election, so every step must be taken to prevent such an event. Fortunately, in most states the results produced at central count stations are informational only, and are not the official election returns. With DRE systems, the ballot images representing individual voters' choices are stored both in the machine on which they were cast in redundant memories and also in removable modules than can be transported. All of these memories are cryptographically linked so substitutions and cracking are not feasible. A manipulation of the central count computer would not be to any avail since the totals produced there would not correspond to the canvass of individual precincts.

4. Insertion of malicious code by the machine manufacturer. There are two subcases. In the first, the manufacturer delivers software to a jurisdiction with prior knowledge of the ballot layout, candidate names, etc. for each precinct in the jurisdiction. The machine is programmed to behave perfectly before and after the election but to switch votes to favored candidates during the election. This manipulation is possible if the manufacturer is able to distribute software directly to specific precincts prior to an election. Countermeasures are discussed in sections 3.5 and 3.6, below.

In the second subcase, the manufacturer has no foreknowledge of the details of any specific election but distributes master software that causes candidates of a particular party to win in all future elections. The practical possibility of such a scheme is nil. There are about 170,000 election precincts in the United States. It is not possible to move a constant fraction of votes from one party to another in each jurisdiction without it being obvious that manipulation is going on because the political demographics of the precincts are too individualistic and distinctive. Therefore the software would have to be distributed with a database telling it how to alter the vote for each relevant candidate in each precinct. The database would have to contain at least the names of political parties and possibly candidates and would have to know in advance the precise hours during which all future elections are to be conducted so the machine would know when to behave properly.

This nightmare scenario, in which a small number of programmers manipulate the politics of the United States by injecting undetectable malicious software into voting machines has more in common with spy novels than it does with reality. For example, in the movie *Goldfinger* (1964), a crazed collector of gold apparently uses nerve gas to kill the entire garrison of troops guarding Fort Knox, then enters the vault where U.S. gold is stored and almost sets off an atomic device that would render the U.S. bullion supply radioactive and useless, which would immensely increase the value of his own holdings. When the film appeared, did the Army close Fort Knox out of fear that the plot was realistic? No. The reason is that adults eventually develop the ability to distinguish fact from fiction, a critical intellectual facility that should not be abandoned simply because we are talking about voting. Did the Pentagon evaluate the plot to determine whether there were security weaknesses that ought to be remedied? Probably. Were some security procedures modified to reduce the probability that such a plot would succeed? Maybe. Is breaking into Fort Knox in such a manner absolutely impossible? No. Why, then, if there is some nonzero probability that a person could do it, do we allow our gold to remain stored there? It’s because we never require perfection in real systems. We balance the risks rationally against the cost and other detriments of preventing the risks and make a reasoned determination. Just because a novelist (or a computer scientist) can dream up an entertaining doomsday plot involving voting machines does not mean we should toss them on the junk heap.

The argument I have with DRE opponents is that they insist that any conceivable risk of any kind of manipulation is unacceptable. That standard is never applied anywhere in human affairs, and there is no reason it should apply to voting, despite appeals to patriotism and pious claims that our very constitutional system is in jeopardy.

I do not propose that machines or software ought to be trusted just because they use advanced technology. In his 1984 Turing award lecture, entitled "Reflections on Trusting Trust," Ken Thompson demonstrated a method of hiding malware so it absolutely cannot be detected by any amount of examination of the corresponding C source code[9]. The technique involves corrupting the C compiler so that it recognizes certain patterns in the source program and compiles them into object code that performs not as written but as the malicious intruder intends. Of course if one is able to modify the compiler in this fashion the compiler could just substitute an entire program of its own choosing upon reading a "signal" string in the source text. Efforts to test the compiler to reveal its misbehavior would be frustrated unless one knew the signal string, since if the string were missing the compiler would always perform properly. Theoretically this hack enables arbitrary amounts of code to be inserted into any program at the cost of introducing but a short sentinel string to tell the compiler to start its dirty business.

The Thompson Trojan horse is frequently cited by opponents of electronic voting[10] as a reason not to rely on voting machines. No one has ever suggested a remotely practical manner in which the world's compilers could become corrupted, but let's assume there is some way of sneaking a rogue compiler into a huge number of computers. This ignores the fact that jurisdictions themselves do not compile voting software, and that even though the source code may not be revealing, the object code contains all the evidence necessary to detect the intrusion. A decompiler can be used to verify that the malware is not present and/or that the object code being used corresponds to the original object code.

The argument has even been made that Turing's proof of the undecidability of the Halting Problem has some applicability to DRE machines[11]. The cited paper asks us to draw the conclusion that "Determining that software is free of bugs and security vulnerabilities is generally impossible." That statement is true only if the word "generally" is carefully defined. A correct version of the statement, but one unsuited to the opponents' purposes, is "There is no procedure that is always *guaranteed* to determine whether an arbitrary program is free of bugs and security vulnerabilities." The unsolvability of the halting problem does not imply that no program can be proven correct, nor does it imply that the halting problem for restricted programs is unsolvable. For example, FOR-loops that do not modify the index variable or its limits and contain only straight-line code do halt. These are precisely the type of loops that are used for iteration in vote tabulation.

Assuming that one believes it is necessary for voting system vendors to produce mathematical proofs that their software is correct (an unreasonable proposition), one can easily imagine structuring a program that reads a finite number of ballot images and produces vote totals to be amenable to such a proof. I therefore must brand references to undecidability in the context of electronic voting simply as sophistry.

1.2.1. The Omniscient Hacker

Combining the misleading Halting Problem argument with the Ken Thompson code-hiding method produces a fantasy that I refer to as the "omniscient hacker," which was explained to me by an opponent of DRE machines who will probably be grateful not to be named here. The hypothetical omniscient hacker is able to insert arbitrary amounts of malware into a voting system in such a way that it can never be detected by any amount of code reading (source or object) or testing (before, during or after the election), yet is able to alter the votes to achieve any predetermined result in any jurisdiction for an arbitrary numbers of years into the future. We need not yet go into the details of why such a thing is or is not possible, since a moment's reflection reveals such a hypothesis to be no more than a purely religious belief. By the very premise of the statement the malware cannot be detected, so no amount of evidence of its non-existence can disprove the statement. If the malware ever is detected, the hacker will explain that he just didn't do a good enough job hiding it, but he'll succeed the next time. In this way belief in the omniscient hacker is indistinguishable from belief in a Supreme Being. There is simply no argument one can give that will dissuade a true believer, yet when the believer is asked for a demonstration he is unable to produce one.

That said, here is an adversary argument that demonstrates that the omniscient hacker cannot exist, though for the reason just stated I do not expect true believers to accept it. If we test the machine during the election by feeding it votes in a manner indistinguishable from regular voting, the malware must decide whether it is going to tell the truth or lie about the vote count. If it tells the truth, it has disabled itself and we need not be concerned that it is present. If it decides to lie, we will catch it, since we are casting a set of ballots whose totals are known.

It is of course possible that there are ballot combinations we may not have tried that will cause the malware to enter lying mode, but there is little risk that ordinary voters will happen upon those combinations either and the malware is either effectively silenced or it will be caught. One can imagine a magic input to the machine that will cause to begin lying (such as writing in the name "Turing" for President). But then activating this feature on every voting machine, or even a substantial number of them, would require a conspiracy of huge proportions.

By its very definition there can be no defense against the omniscient hacker, since we would never be able to tell whether he has been thwarted. (We might as well postulate the existence of an omniscient tamperer who is able to substitute an arbitrary number of voter-verified paper trails without detection. There's no defense against him, either.) Belief in omniscience is a matter of faith. Those who really accept the possibility of an omniscient hacker will never be satisfied with DREs.

1.3. Voting Machine Standards

Since 1990, the Federal Election Commission has developed and promulgated Voting System Standards[12]. The current version of these standards is now several hundred pages long. They deal with hardware, software, telecommunications, security, qualification, testing and configuration management, among other issues. They are voluntary in that any state may, but is not required to, adopt the standards as part of its voting system certification process. As of this date, 36 states and the District of Columbia have done so. The standards are clearly a step in the right direction and obviously enjoy widespread state support, although one wonders whether the states have really evaluated the standards and found them to be meritorious or have adopted them for convenience. It is difficult, however, for a standards-making body to keep up with developments in computer security, develop countermeasures for newly-recognized threats and document them in the form of precise standards. Thus Volume I Standard 6.4.2, entitled "Protection Against Malicious Software" is just two sentences long: "Voting systems shall deploy protection against the many forms of threats to which they may be exposed such as file and macro viruses, worms, Trojan horses, and logic bombs. Vendors shall develop and document the procedures to be followed to ensure that such protection is maintained in a current status." An Independent Testing Authority (ITA) would be justified in claiming that the standard gives no operational guidance in testing a system to see whether it is secure against malicious code. It also appears to pass the burden to vendors, who are the very parties against whom we seek protection.

Independently of the FEC Standards, Section 301 of HAVA [13] purports to impose certain minimum standards on "each voting system used in an election for Federal Office." [14] The term "Federal Office" is not defined in the statute but the Department of Justice takes the position that it has the meaning defined for it in other Federal election statutes, namely, "the office of President or Vice President, or of Senator or Representative in, or Delegate or Resident Commissioner to, the Congress." Laying aside for a moment the question whether Federal control of Federal elections is a good or bad thing, Section 301 of HAVA is unconstitutional on its face. While the Congress may make rules concerning elections for senators and representatives [15], it has no power to specify standards for presidential elections. Article II, Sec. 1 of the U.S. Constitution reads in part: "Each State shall appoint, in such Manner as the Legislature thereof may direct, a Number of Electors, equal to the whole Number of Senators and Representatives to which the State may be entitled in the Congress … The Congress may determine the Time of chusing the Electors, and the Day on which they shall give their Votes; which Day shall be the same throughout the United States." Thus Congress has no power to determine the manner in which presidential electors are chosen other than to specify the time and date of their election.

No one seems to have noticed this unconstitutionality, but more probably the states simply do not care, since HAVA allocates billions of dollars to them for acquisition of voting machines – a case of not acknowledging that the gift horse even has a mouth. In any case, HAVA does not deal at all with the problem of malicious software.

1.4. Testing

DRE opponents argue that DRE software may contain up to 50,000 lines of poorly-written code that is impossible to read or test[16]. The argument is misleading – deliberately so in the author's opinion. It is true that complete voting software systems, including ballot setup and printing components, may reach that size, but the portions of code that accept input from the voter and record ballot images – the very portions suspicions about which have given rise to calls for paper trails – are tiny by comparison.

While it is surely true that not every logic path of a computer program of any size can be exercised, this is obviously not a reason not to use software. Otherwise no commercial software would ever be used, and surely not in any situation in which human life were at risk. The issue is whether any combination of code reading, program testing, open source code publication and other techniques can give us adequate assurance that the software does not contain malicious code or logic errors that will cause votes to be altered. The answer is certainly yes. If code is too obscure, or contains portions that are not readily understandable, it should not be used. Only if the relevant programming is transparent and available to the public should we be confident about using it.

One should realize that the basic loop that interrogates portions of a touchscreen and interprets them as votes is not very complex, although an entire election administration system might be. When the user touches the screen the processor is notified through an interrupt and receives the geographic coordinates of the point that has been touched. A search is made to determine which box on the screen has been touched. Any code that is present that treats candidates differently based on their ballot positions should not be there.

1.5. Machine Failure

By far the most justifiable criticism of DRE machines is that they fail during service or in some cases cannot even be brought into service on election day. There are numerous documented instances of such failures. These incidents are real. They are intolerable when they interfere with the act of voting.

It is important, however, to understand the nature of the machines' failure modes. They do not suddenly decide to move votes from Democrats to Republicans. They may "hang up," refusing to accept any more votes. The mechanical components, particularly the touchscreens, may develop dead spots or fail to register at all. Switches and buttons wear out. Circuits exhibit erratic behavior. These situations can result in severe voter inconvenience and loss of confidence in the process. Long lines can develop, causing voters to balk and go home. The sight of technicians opening machines and replacing components in full view of the voters does not promote trust in the integrity of elections.

While voter inconvenience is certainly detrimental, the critical question is whether any votes are actually lost or modified when the machines fail. In properly designed DRE, no vote once cast is ever lost because ballot images are stored in redundant memories, including write-once devices. It is possible, however, for a machine to fail in such a way that votes cast subsequent to the failure are misrecorded. When the failure is discovered later, it may be too late to reconstruct the lost votes. This situation is akin to mechanical failure of a lever machine – regrettable, but not fatal so long as the failure is not systematic or deliberately induced.

The matter of machine reliability is a question of design, engineering, testing and adherence to maintenance procedures. The responsibility of the vendor is not to be overlooked. A proper voting machine procurement will impose heavy penalties on vendors whose machines do not conform to warranty. If a jurisdiction is unwilling to rely on indemnification by a vendor, a solution is to acquire spare machines and stand ready to deploy them as needed during an election.

It is the author's opinion that many of the so-called failures of DREs in fact resulted from inadequate training of poll workers in using the equipment. HAVA has created an incentive for counties to rush to procure and begin using DREs. Some jurisdictions have done so without adequate preparation, and have seen failures occur during an election. When machines are tested at the warehouse immediately prior to an election and are found to be working, yet cannot be started on election day morning, it is much more likely that the problem results from unfamiliarity with startup procedures than a sudden and unexplained failure of the equipment.

Despite energetic efforts by opponents to slow their adoption, DRE machines continue to be adopted at a prodigious rate. India, the world's largest democracy with over 650 million voters recently adopted DRE machines nationwide. Just its 600,000 villages constitute more than four times as many election districts as there are in the entire United States.

2. Paper Trails

It has been asserted that adding paper trails to DREs allows prompt detection of all of the possible intrusions discussed above. It is based on the mistaken belief that paper records are in some way more secure or free from tampering than electronic ones, which is not the case.

On March 20, 2004, a presidential election was held in Taiwan. The winner by 29,518 votes (out of over 13 million cast) was the incumbent, Chen Shui-bian. To achieve this result, the Central Election Commission had to declare 337,297 ballots as invalid, more than 11 times the supposed margin of victory. The voting method was by paper ballot, and there weren't even any DRE machines to blame. Surely if the voters could rely on the paper ballots to be counted properly this result could not have occurred.

2.1. Paper Records

Humans have a profound affinity for that which they can see and touch. This results in a deep reverence for the printed word, whether it is true or false, and explains the comfort people derive from paper receipts. There are very few paper documents that have preclusive legal effect, meaning that the writing on the face of the document is not subject to challenge.

There are basically four types of paper records:

1. Bearer instruments. Examples: currency, bearer bonds, checks, movie tickets. Here the instrument itself entitles the bearer to rights with no further inquiry into his bona fides. Title to

the document passes with possession. These instruments are extremely convenient for transactions because they can convey rights and title instantaneously without resort to offline records and databases. They are also a frequent subject of theft.

2. Receipts. Instead of being a instrument used to effectuate a transaction, a receipt is merely evidence of the transaction. As such, a receipt takes its place among all of the other forms of evidence, including spoken words, videotapes, witness testimony, business records, computer databases, etc. The receipt confers no independent rights, but is given for several reasons. First, a party to the transaction usually insists on a receipt (a) as evidence of the transaction, as in an ATM withdrawal; (b) to verify the correctness of its details, as in a restaurant bill; (c) as an aide-memoire to recall the transaction. It is used in the event of a dispute to lend credence to the claim of one party or another. The contents of a receipt may be challenged or rebutted and the effect it has will be determined by the trier of fact.

3. Business records. These are notes kept by a business as part of its operations. Records kept in the ordinary course of business are admissible as evidence, but they are only evidence and may be challenged. They differ from receipts in that they are created by one party to a transaction and but are not normally reviewed for correctness by the other party. A dispute between a bank and its customer over a questioned ATM transaction usually turns on the question of which records are more credible, the customer’s paper receipt or the bank’s computerized business records.

4. Ballots. A ballot is an expression by a person indicating how she wishes to cast her vote. A ballot is a unique document defined by election law and is itself only evidence of how a voter wanted to vote. A ballot may be challenged on many grounds, including an allegation that the voter was not entitled to vote, the ballot was mismarked, the voter voted in the wrong precinct, the voter cast votes for candidates she was not entitled to vote for, the ballot was mangled, defaced or was otherwise unreadable. In many, but not all, states when the content of a ballot is disputed, a court is required to determine the intent of the voter in marking the ballot and is not bound by that the ballot actually says.

There are numerous other forms of paper records, such as documents of title, licenses, wills, diplomas, written offers, etc., that are not relevant to our discussion here. The question is what desirable properties, if any, do paper records have that would cause us to prefer them over electronic ones for voting.

The largest industry in the world in terms of daily cash flow is foreign currency trading, which often totals more than $2 trillion per day. The entire world securities industry rarely exceeds one-tenth of that amount, and no sector that deals in physical goods can even approach it. The vast majority of foreign currency trades are made without any use of paper whatsoever, either in the form of an original order or a generated receipt. If computers are unsafe and hackers and well-placed insiders lurk behind every door, one wonders why the traders don’t lose a billion dollars a day (or at least a million) as a result of malware. In December 2003, no less a figure than Senator Hilary Clinton stated while introducing her “Protecting American Democracy Act of 2003[17]”: “You go to an ATM, you get a receipt. You play the lottery, you get a ticket. Yet when you cast your vote, you get nothing. The systems used by the people of the United States to exercise their constitutional right to vote should be as reliable as the machines people depend on to get their money. What's required for money machines should be required for voting machines." Statements that play well to the electorate often fail when subjected to the cool light of logic.

Sen. Clinton is correct that Regulation E of the Federal Reserve Board[18] requires a financial institution to make a receipt available when a consumer initiates an electronic funds transfer at an ATM. She might be surprised to learn how limited the legal effect of the receipt turns out to be. If a financial institution fails to provide a receipt through "inadvertent error," it is not in violation of Regulation E[19]. Furthermore, the receipt itself is only prima facie proof (subject to rebuttal) that the consumer made a payment to a third party[20]. It is not proof of the amount of transfer and is of course of no effect at all in the case of an ATM deposit, since the data associated with the deposit is generated completely by the consumer, not the bank.

In the event of a later dispute between the consumer and the bank, the ATM receipt is evidence only and is not dispositive of the question what amount was transferred. The bank may challenge the data on the receipt based on its own records. Note that the receipt has been in the hands of the consumer and thus has been subject to alteration or forgery, which means that the document itself cannot be given absolute effect. Of course in electronic banking transactions initiated over the Internet there are no paper receipts at all, yet this fact has not dampened enthusiasm for online banking.

The law governing ordinary sales transactions, the Uniform Commercial Code, gives no legal effect to receipts and certainly does not require them[21]. In fact, neither party to a sale transaction has the legal right to demand a receipt, although it may be a customary business practice to comply with such a demand.

Sen. Clinton would be positively dismayed to learn that a lottery ticket has even less value to its holder than an ATM receipt. State lottery rules typically provide that if a dispute arises between the holder of a lottery ticket and the state lottery bureau, the computer records of the lottery bureau govern. This New Hampshire Lottery rule is illustrative: "To be a valid ticket and eligible to receive a prize … [t]he information appearing on the ticket shall correspond precisely with the Commission's computer record."[22] The lottery rules clearly provide that computer records govern over paper ones.

And so it must be. If presentation of a small piece of paper were sufficient to claim a prize of $363 million[23], the inducement to fraud and bribery to produce a counterfeit ticket would be extreme, and the nature of paper is that it would be essentially impossible to invalidate the ticket based on a physical examination because genuine ticket stock can easily be obtained. This raises the question what the purpose of a lottery ticket might be if not to ensure the buyer that he will get paid in the event of a win. Despite what the public might believe, the lottery ticket is simply a receipt, that is, an item of evidence that can be considered in the event of a dispute. It also provides the buyer with the opportunity, in the act of buying a ticket, to verify that the human operator typed in his numbers correctly. The issue is not that the lottery ticket machine may have malfunctioned, but that the human seller may have made a mistake. (As we have seen, if the lottery machine malfunctions, that is, communicates a different set of numbers to the lottery commission than those printed on the ticket, the buyer has no effective recourse.) Because the only human in the voting booth is the voter herself, and the voter has ample opportunity to review her ballot, the verification function of the lottery ticket is not relevant to elections.

The lottery ticket also serves to remind the buyer which numbers he chose so he can later compare his numbers with the winning ones. It is also necessary to claim the prize, since a lottery ticket is anonymous and transferable. The state must know whom to pay. None of these considerations is applicable to voting[24].

Of course Sen. Clinton’s Protecting American Democracy Act of 2003 is unconstitutional for exactly the same reason that Section 301 of HAVA is unconstitutional – it purports to allow Congress to legislate standards for presidential voting, a privilege reserved to the states.

When I raise the point to opponents of electronic voting that huge volumes of commerce are conducted based only on computer records, their answer is, “If anyone lost a billion dollars they would know. If someone steals votes, we’ll never know.” This explanation is appealing, but specious. If someone were able to manipulate a bank’s computer records to spirit away a huge sum of money, it is reasonable to believe that he could do so while at the same time not only deleting any computer records of the transaction but also modifying the bank’s records so it did *not* know there was any loss. But in any event it does not matter whether the bank knows that it has lost a billion dollars or not – the money is gone and the risk the bank tried to avert has occurred anyway.

2.2. Electronic records

The areas of human endeavor in which electronic records are used in place of paper ones are far too numerous to list. Among them are banking transactions, income tax filings, medical diagnosis, military orders (including nuclear launch instructions) and securities purchases.

The public and the legal system have come to recognize that electronic records can be reliable if properly maintained. The Electronic Signatures in Global and National Electronic Commerce Act (“E-Sign”)[25] raises electronic records to at least equal dignity with paper ones. It provides that in “any transaction in or affecting interstate or foreign commerce … a signature, contract, or other record relating to such transaction may not be denied legal effect, validity, or enforceability solely because it is in electronic form."[26] There are a small number of exceptions for such specialized documents as wills and testamentary trusts and notices of termination of insurance benefits, but otherwise electronic records do not have inferior status.

The regulations implementing the E-Sign statute generally provide that electronic records are equivalent to those on paper[27]. The Uniform Electronic Transactions Act (UETA) has been adopted in 45 states and is pending the three others. UETA specifies the legal effect of electronic records and has as one of its stated purposes "to promote public confidence in the validity, integrity and reliability of electronic commerce and governmental transactions."[28] If electronic records are questionable in some way, how has this fact escaped the vast majority of our state and federal legislators?

The Federal Rules of Evidence give equal weight to electronic records in court proceedings as they do to paper ones[29]. It is therefore a puzzle why electronic records should be acceptable for every other government purpose except voting. Neither E-Sign, UETA nor the Federal Rules of Evidence contain any receipt requirement.

2.3. Paper ballots

Paper ballots can be divided generally into those that are intended to be read and counted by humans, which we shall call Australian ballots to avoid ambiguity, and those intended to be counted by machine. The latter included punched-card and mark-sense (optical scan) ballots.

Every form of paper ballot that has ever been devised can and has been manipulated, in general with considerable ease. The reason is that humans are familiar with paper and its characteristics, how to mark it to look genuine and how to erase it. Likewise, the number of people in the U.S. capable of producing professional printed matter is huge. There are over 50,000 printing companies in the U.S, employing over 1.2 million people, of whom more than 100,000 are prepress operators[30]. This means that it is not difficult to locate people who can print or modify documents.

Other types of manipulation, such as destroying ballots or substituting other ones, require no skill at all. By contrast, altering redundant encrypted write-once computer records is impossible even for experts. So assuming that the electronic voting records are written correctly in the first place (a subject that indeed deserves discussion), the possibility of modifying them later is remote.

The simplest form of paper ballot manipulation is ballot-box stuffing, that is, inserting extra ballots, usually genuine ones that have been pre-marked, into the container meant to hold only those voted by registered voters. In any jurisdiction in which the voter can touch a physical ballot and personally insert it into a ballot box, she can conceal extra ballots on her person and insert them at the same time. This is true whether the ballots are Australian, punched-card or mark-sense. The practice is so widespread that many states have statutes specifically dealing with the situation in which more ballots are found in the ballot box at the close of voting than the number of voters who appeared at the polls that day. The Florida statute is both horrifying and amusing: "[I] if the number of ballots exceeds the number of persons who voted, as may appear by the poll list kept by the clerk and by the stubs detached by the inspectors, the ballots shall be placed back into the box, and one of the inspectors shall publicly draw out and destroy unopened as many ballots as are equal to such excess."[31] Yes, ballots are chosen at random and discarded until the totals come out right! The most appalling thing about the law is not how the procedure is to be conducted, but that the situation occurs frequently enough that the law had to be drafted in the first place.

Actually, the Florida process solves nothing except to avoid the unseemliness of having more votes cast than voters, which is always an embarrassment. If the ballot box has been stuffed, the random discard process will not alter the candidates' percentages on average. That is, whoever wins by the stuffed vote total will probably also win after the excess votes are tossed away, and the stuffers will have achieved their objective.

Another form of manipulation is to perform substitution of ballots on a large scale. In central-count jurisdictions, ballots are not counted at polling places but are transported by vehicle to a centralized counting station, usually at the county seat. The ballots are carried in transport cases outfitted with locks and seals, but the locks can easily be opened and the seals counterfeited. It can take several hours in large counties for the ballots to reach the counting station, giving ample opportunity for chicanery. Instances are known in which manipulators did not even bother to open and reseal the ballot cases, but merely substituted others that had been prepared once the total turnout in each precinct became known. This sort of manipulation is made easy by the fact that printed Australian ballots are insecure and transport cases and seals easily obtained from unauthorized sources.

One of the oldest and easiest forms of tampering is to invalidate an Australian ballot while touching it. When I was in middle school during the 1950s, our American history teacher explained that poll workers would break off a piece of pencil lead and insert it under their thumbnail. When they found a ballot voted for a candidate they didn't like, they would make a second mark for some other candidate in the same office, thus creating an overvote that had the effect of erasing the undesirable choice. Once this has been done, there is no effective way to reconstruct the original ballot.

Because Australian ballots have to be marked and read by hand, there is no real prospect for tampering to occur on a national scale. The same is not true of punched-card and mark-sense ballots. The only remaining use of punched cards in the United States is for voting, and only two manufacturers remain in the business. Without giving a catalog of possible tampering methods, there are many parameters in card manufacture than can be varied to the advantage of one candidate or another if the voting positions corresponding to the candidates are known at the time of manufacture.

The problem of hanging chads, long known in the election industry, came to public attention in Florida in 2000. But for years many states used "chad teams," groups of poll workers who function was to tear loose chads from ballots before they were fed into the card reader. Once we allow a person to alter a ballot that has been cast by a voter, anything is possible. A perfect tool for punching out chads by hand is the metal tongue from an ordinary waistbelt. Small and easily concealed in the hand, it can be used the same way the old pencil lead was employed to overvote Australian ballots.

With mark-sense ballots it is known that if the areas for marking the ballots are printed improperly or the timing marks at the side of the ballot are skewed, votes that are cast will not be read properly by the scanning machine. More tampering is possible through the selective application of inks that appear white but absorb the infrared light that is used in the reading process. An answer, one might think, is that we always have the original ballots around to recount by hand, but mark-sense ballots are just as susceptible to loss, substitution or augmentation as Australian ones.

In general, the rampant problems with paper ballots are neither acknowledged nor addressed by opponents of electronic voting, who seem oblivious to the fact that their opposition to new technology, if successful, will compel us to retain something that is much worse.

2.4. The "Voter-Verified" Paper Trail

It is alleged that adding a so-called "voter-verified paper trail" to a DRE machine will either permit tampering to be detected or at the very least will provide a reliable record of how each voter voted that can be used for a recount, even if the recount must be conducted by hand. This is incorrect. A paper trail accomplishes one thing, and one thing only – it provides assurance to the voter that her vote was initially captured correctly by the machine. This is no small accomplishment, but it can be achieved in numerous other ways, as explained below. That is the only voter-verified part. The paper trail provides no assurance at all that her vote will ever be counted or will be counted correctly. The reason simply is that the paper trail itself becomes insecure at the moment of its creation.

First, if the machine cannot be trusted, which is the working hypothesis of paper trail proponents, then it cannot be trusted to deal with the paper trail safely. After the voter leaves the voting booth, it can mark her ballot as void and print a different one. The voter will have left the booth believing not only that her vote was cast and counted properly, but that it will also be counted properly in any recount. None of these beliefs is correct.

One might argue that inspection and testing of the machine would reveal such abjectly bad behavior, but the claim of DRE opponents is that no amount of inspection and testing is ever sufficient. If testing is adequate to reveal paper trail flaws, then it is adequate to uncover other faults in the machines.

Here is a further, but only partial, catalog of problems with paper trails.

1. The paper trail cannot be on a continuous roll of paper, since that would permit reconstruction

of each voter's ballot based on the order in which votes were cast. Therefore, the paper trail must consist of separate pieces of paper. However, once the pieces of paper are separated, the integrity of the trail is lost. Looking at a piece of paper, we will not be able to tell for certain where it came from. Stuffing and all other paper ballot tampering methods then become possible. The addition of cryptographic indicia, which has been proposed as a method to prevent insertion of unauthorized ballots, cannot work since the voter will never know whether her real ballot contained the proper indicia when it was created. If it didn't, the ballot will not be tabulated during a recount.

2. Adding a paper printing device to a DRE machine naturally adds another component that can fail, run out of ink, jam or run out of paper. If DREs are alleged already to be prone to failure, adding a paper trail cannot improve that record. In Brazil in 2003, where a small number of precincts had installed paper trails, failure of the printers delayed voters by as much as 12 hours,[32] a figure that would be catastrophic in the U.S.

3. There is no voter-verified paper trail machine that has been tested on any large scale.

4. States that propose to implement the paper trail have promulgated regulations stating that the paper shall govern over the electronic record in the event of discrepancy[33]. This has the effect of making the insecure paper record paramount over the secure electronic one, a return to the early days of the Australian ballot.

5. With complex ballots, voters are prone to forget exactly whom they have voted for. When confronted with a paper record, they may erroneously claim that the machine made a mistake. This will call the machine's reliability into question, prompt calls for a recount and cast doubt even on machines that are functioning properly.

6. Paper trails do not address the problem of DRE failures. If the complaint is that a machine cannot be initialized for use on the morning of election day, then having a paper trail mechanism is of no help. In fact, the presence of the mechanism increases the load on the machine's power supply and processor and itself increases the probability of failure.

7. The paper trail requires a re-examination of meaning of the terms "ballot" and "official ballot." This is not a mere semantic exercise, but a question of great legal and, in some states, constitutional significance. Can a piece of paper be a ballot if it is neither marked nor touched by the voter? If so, significant statutory changes will be required. If the paper is the ballot, then what conceivable meaning can be ascribed to the computer count, which is not derived by counting the "ballots," but by processing the voters' original inputs that were separately used to generate the ballots? If the paper ballots are official, then we are put in the untenable position of having to certify an election without ever actually counting the ballots, unless an allegation of irregularity compels a "recount."

8. Each losing candidate will claim that the election was stolen from him by the machine and will insist that the only true indication of the voters' preferences reside on the paper, even if there is no evidence of irregularity or tampering. Thus paper recount will become the default method

of vote counting, mitigated only by the high cost of such recounts. If this is to be the case, why use voting machines in the first place?

9. Paper trails cannot readily be viewed by disabled voters, requiring them yet again to reveal their votes to strangers in order to have them verified. It is no answer to say that there are other mechanisms to review their votes. If paper trail proponents truly believe the paper trail is necessary for fair elections, then elections will not be fair for the disabled.

10. A report of the Caltech-MIT Voting project concluded that the presence of paper trails actually decreases public confidence in the voting system[34]. This can be understood as follows: would requiring airplane passengers to inspect the plane's engines before boarding enhance their belief in the safety of the aircraft?

My position on paper trails, despite their problems, is not an extreme one. If a manufacturer produced a reliable paper trail device and the remainder of his system were acceptable, I would see no problem in certifying such a machine. I am firmly opposed to any audit trail requirement, however, and even where audit trails are used, the paper record should never govern over the electronic one because it is vastly less secure. The proper use of audit trails is as evidence. If the paper trail totals differ from the electronic ones, that is the starting point for investigation, not the end of the issue..

3. Alternatives to Paper Trails

If paper trails are not the answer, are there practical alternatives that will not only render DREs safe but also persuade the public that they are safe? Let us assume that all of the security risks discussed above (except the omniscient hacker) are realistic. Are there measures other than paper trails that will prevent them? The author does not discount the importance of assuring the voter that the machine is working and that her preferences have been collected without error. This can be done in a multitude of ways that do not involve paper.

3.1. Audit devices

A prime motivation for audit trails is the possibility that the machine has been programmed improperly, either by accident or by design, or that rogue software has been substituted for the authorized version. Suppose we were to require voting machines to be architecturally separated into two distinct devices: a panel, possibly but not necessarily a touchscreen, whose only function is to display the ballot and capture voter choices, and a tabulation and recording device, which accepts input from the panel and performs computations. The panels and tabulation devices could be supplied by different manufacturers.

Now suppose we feed the output of the panel to two different devices simultaneously. One is the tabulation machine; the other is an audit device made by yet a third manufacturer and programming by an independent body, such as an accounting firm or public interest group not affiliated with the tabulation manufacturer. The audit device displays the voter's choices on a screen of its own for verification. The voter views the audit screen, and if it is correct, presses a

"VOTE" button. Both the tabulation device and the audit device make redundant read-only records of each ballot image. At the end of the election, all the records are compared. If they different in any respect whatsoever, the results from that machine are called into question and an investigation is launched. An examination of the software installed in the two devices should reveal whose records are the reliable ones.

So long as there is no collusion between the audit device manufacturer and the tabulation manufacturer, no amount of tampering with either machine will go unremedied. The prospect of tampering identically with both, since their software systems would be completely different, is too small to consider seriously. The audit device could easily be outfitted so disabled voters could verify their votes.

3.2. Open source

The manufacturers of voting equipment claim that their software is a trade secret and go to extraordinary lengths to preserve that myth. The author has been looking at the source codes of voting systems for over 20 years and has yet to find any significant differences in their design except possibly for the number of bugs they contain. They all do the same thing, albeit in somewhat different ways. No vendor's software is a significant selling point providing any competitive advantage over other systems – jurisdictions focus on the hardware. All the software has facilities for setting up elections, storing the candidate and party names in a database, presenting ballot choices to the voter, tabulating and storing the results and possibly transmitting them after the election. The systems vary in ease of use and capacity, but they do not contain trade secrets for the simple reason that every aspect of election setup and balloting is well-known to all.

One might speculate then on why they try to keep the source code confidential. The uncharitable view, which appears to have some justification, is that they don't want the public to see how bad their code is. A legitimate reason might be to avoid making matters easy for competitors, but that does not justify withholding information from the public that is necessary to promote confidence in the electoral process. Another reason is to hide security measures which, if disclosed, would provide a roadmap for hackers. I am somewhat sympathetic to that view, despite the meaningless but mocking phrase "security through obscurity," since I know a thief will have a much harder time stealing my car if he does not know where I have hidden the key than if he does, and a party who happens to find my hidden key will have no idea which car it fits.

On the other hand, there is no reason that the ballot setup, display, tabulation and reporting sections of voting system code should be kept secret, and manufacturers would be wise to accede to public demand in this regard.

3.3. Administrative procedures

The administrative procedures concerning the handling of DRE machines and materials are usually not spelled out at all, or, if spelled out, then not circulated and not followed. Many of the observed vulnerabilities in DRE systems stem not from problems of machine design, but from lax handling procedures. A thorough election administration manual should explain at least the following steps:

1. Custodianship of machines at all times, including transportation to and from polling places.
2. Receipt and registry of software to ensure that only authorized copies of everything, including operating system versions, are used in voting machines.
3. There should be no delivery of any software directly from vendors to jurisdictions; otherwise (2) will not be observed.
4. Deposit and security for ballot materials, including any election programming. Likewise, control of installation of election programming into voting machines.
5. Chain of custody for any removable media containing ballot images or vote totals.
6. In the event an audit trail is used, chain of custody for the paper ballot images.
7. Freezing of machines and their software at least until the election is certified and the time for any challenge has passed.
8. Exception procedures for handling irregularities during an election, including custody of partial totals on any machine that is removed from service.

3.4. Standards

It may not be fruitful to have all the states separately ponder and solve the myriad of problems in election administration posed by the sudden introduction of new voting technology. Knowledge and experience should be pooled and election officials ought to be able to rely on a full set of standards, including security and vote handling procedures, that they can follow. The FEC Standards were principally written for ITAs to follow, not for election jurisdictions, and do not specify processes that are responsive to numerous objections that have been raised to DRE voting.

The budget provided by HAVA is fully sufficient to fund development of a comprehensive set of standards and procedures which, if followed, would greatly diminish the number of problems observed at polling places.

3.5. Parallel testing

More than 15 years ago, in a Pennsylvania certification report, I wrote of the possibility that a DRE machine could contain an on-board clock and that an intruder could rig the machine so that it behaved perfectly in all pre- and post-election tests, but switched votes during an election. The prospect is even more real today than it was then, since computers now routinely possess such clocks. This attack presupposes that the software knows all dates and times for elections into the indefinite future, but let’s assume it has such knowledge[35].

One solution is to forbid on-board clocks altogether, but that would limit various other capabilities, such as making a time-stamped record of happenings during the election. It also raises the question how one can tell whether a clock is present in a machine or not. The second obvious solution is to reset the machine’s clock to a time on election day, run a test and then set the clock back to the correct time. This is ineffective since the machine could contain software that would detect such a change and know that it was being watched.

A better solution is to employ parallel testing, a plan originally suggested by this author that was used in 10 counties in California during the 2004 primaries. Under this method, a set of examiners is empowered to enter any polling place at the start of voting and commandeer any voting machine for test purposes. No actual voters cast votes on the selected machine. No change whatsoever is made to the test machine – it is not even moved from its position (to counter the argument that it might contain a motion sensor to warn that it was under test). The examiner votes a number of predetermined ballots comparable to the number that would be voted on a typical machine in that precinct. Of course, manual entry of votes by a human is an error-prone process, so a video camera is used to capture his actual vote entries. At the normal close of polls, the votes on the test machine are tabulated and compared with the expected totals. If any software is present that is switching or losing votes, it will be exposed.

The function of this test is limited. It of course does not ensure that even one other machine in the precinct is working properly. It is designed to detect the nightmare scenario in which some agent has tampered with every machine in the jurisdiction undetectably, a major risk cited by DRE opponents to justify the addition of paper trails.

The examiners would select precincts and machines at random on the morning of the election. It is an issue of statistical quality control exactly how many precincts should be chosen. This testing, while cumbersome, is much easier that statutorily mandated recounts in which a certain percentage of ballot images must be totaled manually.

3.6. Separation of candidate names

Perhaps the ultimate protection against malicious code is to keep candidate and party names segregated from the software so it cannot perform any meaningful manipulation. If the machine is programmed to move votes from one party to another, it will be stymied if it is unable to determine the party with which a candidate is affiliated or even which candidate is associated with a given ballot position. This can be done by presenting the candidate and party names and issue text in the form of graphic files that can only be read by a human being. The only thing the software can do us faithfully record the numbers of the ballot positions that were selected. Of course, since it also knows no candidate names, it can only report results by ballot position. To defeat such a countermeasure the software would have to contain a complete optical character recognition algorithm.

It is possible that in a conspiracy a tamperer's confederate could, while voting, provide information via touchscreen selections or the write-in panel that could inform the software of the particular voting positions to manipulate. However such an act would have local effect only, since it would take one confederate for each voting machine involved. It would not be feasible to perform manipulation on a large scale with such a scheme.

4. Answering the Objections

We are now equipped to respond to the objections to DRE voting raised in the Introduction.

Objection 1. DREs are black boxes. So are all other computer systems, on which we rely for our lives and our fortunes.

Objection 2. Code cannot be audited. Yes, it can. Not all code can be audited, and we can bar unauditable code from being used in elections. We can also make the code available for scrutiny by an arbitrarily large audience by making source code open.

Objection 3. Machines cannot be tested. Why not? Every other type of machine can be tested, and voting machines are not nearly as complicated as airplanes.

Objection 4. Hackers can do anything. Only in books and movies. The hacking stories we read in the papers concern attacks over the Internet against systems that are deliberately held open for access by the general public. Voting machines, by contrast, are highly controlled and cannot be accessed over the Internet. Hackers are not omniscient and even vendors have trouble programming tabulation software correctly. The prospect that a hacker could not only manipulate an election but do it without exhibiting a detectable bug is so far-fetched an idea that no one has come close to showing how it might be done[36].

Objection 5. DREs are failing all over the place. The answer here is simple: buy reliable ones. The FEC Standards specify numerous tests designed to weed out unreliable hardware.

Objection 6. The vendor can rig the machines. But we can expose him through any number of mechanisms, including audit devices and parallel testing. An we can render his manipulations fruitless by separating candidate and party names from the capture and recording logic.

Objection 7. Computer scientists say DREs are unsafe. Since when was this technological issue to be decided by popular vote rather than by analysis? There are over one million computer scientists and mathematicians in the United States[37]. About 100 of them have signed a resolution in favor of paper trails proposed by www.verifiedvoting.org[38]. No information is available on how many have any familiarity with the processes of voting or the actual architecture of DRE machines, but the total number represents about 1 in 10,000, a minuscule proportion. The good news seems to be that the other 9,999 out of 10,000 have remained open-minded on the subject.

Objection 8. Paper trails meet objections 1-7 and make DREs minimally acceptable. As we have seen, this is not true. The paper trail does no more than persuade the voter that her vote was initially captured properly, but at the risk of announcing to the voter that the whole process is so insecure that her own vigilance is necessary. If the voter has to be watching at the polling place, what sort of confidence will she have in the remaining procedures that are conducted outside her presence? We have shown a number of alternatives to paper trails that genuinely meet the objections raised.

DRE machines have been described, somewhat dramatically, as a threat to democracy [39]. A far greater threat to democracy is a return to any form of paper ballot, but both of these pale in comparison to the fact, not widely known, that in each presidential election more than 5 million Americans who are eligible to vote and want to vote are unable to cast a ballot because they happen to be outside their home districts on election day and cannot comply with their state's absentee procedures. Many of these people are overseas. The claim that tens of thousands of Floridians were disenfranchised in the 2000 election because of butterfly ballots, though probably true, is insignificant when measured against the millions who were unable to obtain any ballot at all. If computer scientists are truly concerned about threats to democracy, that's one they should work on.

---

[1] The author is Distinguished Career Professor in the School of Computer Science at Carnegie Mellon University and an attorney admitted to practice in the Commonwealth of Pennsylvania and before the United States Patent and Trademark Office. From 1980-2000 he was statutory examiner of electronic voting systems for the Commonwealth of Pennsylvania. From 1987-2000 he was the designee of the Attorney General of Texas for voting system certification. During those years he personally examined more than 100 different computerized voting systems for certification purposes. In the 2000 election, machines for which he participated in certification (which did not include Florida) were used to count more than 11% of the popular vote of the United States. This paper was prepared to accompany the author's appearance on an electronic voting panel at the ACM Computers, Freedom & Privacy Conference held in Berkeley, California in April 2004.

[2] The feminine pronoun is used to drive home the fact that a majority of U.S. voters are women.

[3] Shamos, Michael, "Computerized Voting – Evaluating the Threat." Proc. Third ACM Conf. on Computers, Freedom & Privacy. San Francisco, CA (Mar. 1993). Available at http://www.cpsr.org/conferences/cfp93/shamos.html.

[4] National Transportation Safety Board Publication NTSB/SR-02/02, "Safety Report: Transportation Safety Databases," September 11, 2002. Available at http://www.ntsb.gov.

[5] Leveson, Nancy et al., "An Investigation of the Therac-25 Accidents," *IEEE Computer 26*, 7, pp. 18-41 (July 1993).

[6] Available from the Federal Election Commission website at http://www.fec.gov/pages/vssfinal/vss.html.

[7] N.Mex. Stat. Ann. 1-20-5 provides, "Unlawful opening of a voting machine consists of, without lawful authority, opening, unlocking, inspecting, tampering, resetting or adjusting a voting machine owned by any county, or

conspiring with others to have the same done. Whoever commits unlawful opening of a voting machine is guilty of a fourth degree felony." In general, tampering is a felony but the penalties are probably not sufficiently high. *Quaere* whether under the New Mexico statute a manufacturer who ships rigged software would in fact be committing this crime, which seems to require modification of a machine after is has become owned by a county.

[8] U.S. Bureau of the Census, "Population Estimates for the 100 Largest U.S. Counties: April 1, 2000 to July 1, 2002," available at http://eire.census.gov/popest/data/counties/tables/CO-EST2002/CO-EST2002-09.php. Six of the 35 counties are in New York; another six are in California.

[9] Thompson, Ken, "Reflections on Trusting Trust," *CACM 27*, 8 pp. 761-763, August 1984.

[10] Neumann, Peter, "Risks in Computerized Elections," Inside Risks 5, *CACM 33*, 11, p.170, November 1990

[11] Jefferson, David et al., "A Security Analysis of the Secure Electronic Voting and Registration System (SERVE)," Jan. 21, 2004. Available at http://www.servesecurityreport.org/paper.pdf.

[12] Available from the Federal Election Commission website at http://www.fec.gov/pages/vssfinal/vss.html.

[13] There is one reference in HAVA to the FEC Standards, but it pertains to acceptable error rates in ballot counting. 42 U.S.C. §15481(a)(5).

[14] 42 U.S.C. §15481(a).

[15] Article I, Sec. 4 of the U.S. Constitution provides: "The Times, Places and Manner of holding Elections for Senators and Representatives, shall be prescribed in each State by the Legislature thereof; but the Congress may at any time by Law make or alter such Regulations, except as to the Places of chusing Senators."

[16] Bannet, John, "Hack-a-Vote: Security Issues with Electronic Voting Systems," *IEEE Security and Privacy Magazine*, Jan/Feb 2004.

[17] Bill S. 1986, 108th Congress, First Session.

[18] 12 C.F.R. §205.9.

[19] 12 C.F.R. §205.17.

[20] 12 C.F.R. §205.17.

[21] There is a type of document of title known as a "warehouse receipt," which is necessary for a buyer to secure possession of his goods in certain situations, that has special status under the Uniform Commercial Code. But this is not the sort of receipt one ordinarily receives from a merchant in a sale transaction.

[22] New Hampshire Lottery Rule 7(C).

[23]

The largest U.S. lottery payout in history, $363 million, resulted from the May 9, 2000 drawing in The Big Game, a multistate lottery now known as “Mega Millions.”

[24]

In his CFP ’93 paper the author endorsed the use of state lottery systems for voting (without giving receipts, of course) and still does because their security and reliability is proven daily all around the country and they are clearly trusted by the public.

[25]

15 U.S.C. §7001 ff.

[26]

15 U.S.C. §7001(a)(1).

[27]

The Food and Drug Administration regulations are typical: “Where electronic signatures and their associated electronic records meet the requirements of this part, the agency will consider the electronic signatures to be equivalent to full handwritten signatures, initials, and other general signings as required by agency regulations, unless specifically excepted by regulation(s) effective on or after forth the criteria under which the agency considers electronic records, electronic signatures, and handwritten signatures executed to electronic records to be trustworthy, reliable, and generally equivalent to paper records and handwritten signatures executed on paper.” 21 C.F.R. §11.1 (c).

[28]

UETA Comment 1(f).

[29]

F.R.E. 1001 reads, “For purposes of this article the following definitions are applicable: (1) Writings and recordings. ‘Writings’ and ‘recordings’ consist of letters, words, or numbers, or their equivalent, set down by handwriting, typewriting, printing, photostating, photographing, magnetic impulse, mechanical or electronic recording, or other form of data compilation.”

[30]

Press release of the Indiana Business Modernization and Technology Corporation, Dec. 21, 2001.

[31]

Fla. Stat. §102.061.

[32]

Mira, Leslie, “For Brazil Voters, Machines Rule,” *Wired News*, Jan. 24, 2004.

[33]

Standard 2.1.1.4, “State of California DRAFT STANDARDS For Use of Accessible Voter Verified Paper Audit Trail Systems in Direct Recording Electronic (DRE) Voting Machines,” Secretary of State of California, March 18, 2004.

[34]

Selker, Ted. et al, “The SAVE System: Secure Architecture for Voting Electronically: Existing Technology, with Built-in Redundancy, Enables Reliability,” CalTech/MIT Voting Project VTR Working Paper, Oct. 22, 2003, revised January 4, 2004.

[35]

It is actually not difficult to deduce this information from the ballot programming, which usually contains the date of the election in a predefined text field, the presence of which could be required by the system.

[36]

See note 16. Hack-a-Vote is a project in which students are asked to develop malicious vote-counting software and other students try to find the malicious portions. It's not easy when posed in that framework.

[37] According to the Bureau of Labor Statistics, in 1990 there were about 881,000 computer scientists and mathematicians in the U.S.

[38] Spannaus, Edward, "Electronic Voting is Threat to the Constitution," *Executive Intelligence Review*, Jan. 30, 2004.

[39] Zetter, Kim, "How E-Voting Threatens Democracy." Wired.com, Jan, 29, 2004.