This paper first appeared in Proceedings of EVT 2008, Usenix/ACCURATE, 2008.

You Go to Elections with the Voting System You Have: Stop-Gap Mitigations for Deployed Voting Systems

J. Alex Halderman
Princeton University

Eric Rescorla
RTFM, Inc.

Hovav Shacham
University of California, San Diego

David Wagner
University of California, Berkeley

Abstract

In light of the systemic vulnerabilities uncovered by recent reviews of deployed e-voting systems, the surest way to secure the voting process would be to scrap the existing systems and design new ones. Unfortunately, engineering new systems will take years, and many jurisdictions are unlikely to be able to afford new equipment in the near future. In this paper we ask how jurisdictions can make the best use of the equipment they already own until they can replace it. Starting from current practice, we propose defenses that involve new but realistic procedures, modest changes to existing software, and no changes to existing hardware. Our techniques achieve greatly improved protection against outsider attacks: they provide containment of viral spread, improve the integrity of vote tabulation, and offer some detection of individual compromised devices. They do not provide security against insiders with access to election management systems, which appears to require significantly greater changes to the existing systems.

1 Introduction

The widespread deployment of electronic voting equipment has put voting officials in a difficult position. On the one hand, the equipment has been deployed at great expense and transitioning away from it is difficult. On the other hand, every serious review of these systems has discovered significant flaws.

For instance, in every electronic voting system that has been studied, researchers have been able to compromise polling place devices with access similar to what a voter or pollworker would have. In several of the systems it appears to be possible to design a virus that, delivered to a single polling place device, could propagate through the *Election Management System* (EMS) to every device in the county. Moreover, detecting attacks may be difficult, as no good mechanisms are available for determining whether devices have been compromised or for restoring them to a known-good state.

One common response is to look for mitigations: modest changes to the systems or procedures that reduce the likelihood or severity of attacks. For example, after California's Top-To-Bottom Review (TTBR), the California Secretary of State imposed an array of new conditions on the use of the three voting systems certified for use in California: Diebold (now Premier), Hart, and Sequoia. Similarly, after Ohio's EVEREST review, the Ohio Secretary of State's office recommended new restrictions and procedures. In both cases the mitigations were designed under time pressure and with limited input from security experts. This paper attempts to undertake the same task with more time and analysis: designing a set of mitigation strategies that would meaningfully improve security yet be practical for deployment with the type of equipment currently in use.

1.1 Problem Statement

Our objective is to design mitigations that are compatible with the current generation of electronic voting equipment. More precisely:

> *With new but realistic procedures; with no changes to existing hardware; and with few and modest changes to existing software, how can we best secure elections?*

Replacing the existing equipment and designing a new system from the ground up would undoubtedly provide better security, but will take time and require new purchases many jurisdictions can ill afford. Therefore, in this paper we investigate how to make more secure use of the equipment that jurisdictions already own.

We take as a given that we wish to preserve the existing voting experience. This means that voters should be able to use both *Direct Recording Electronic* (DRE) and

optical scan (opscan) ballots in both precinct-count and central-count modes.

The changes we propose will not render any of the systems unbreakable, but we believe they would provide stronger defenses against certain kinds of attacks — such as voting machine viruses — than do current systems as they are commonly used today. This represents a trade-off between security and ease of deployability. While we recognize the desirability of having measures that can be deployed before the November 2008 general election, and some of what we propose most certainly can be deployed rapidly, we also describe measures that may not be deployable in six months but are more practical than a complete redesign of the existing systems.

1.2 Threat Model

The scope of this work is limited almost exclusively to outsider attack. We assume that insiders (e.g., county employees) who have direct access to central election management systems or to polling place devices will be able to do real harm. The current systems are very hard to secure against this type of threat without significant modifications. Our focus is on trying to prevent outsiders from doing too much harm and on being able to detect and recover from any attacks they may mount. In addition, we focus primarily on large-scale fraud; defeating small-scale fraud seems to be much more difficult.

1.3 Basic Assumptions

We start from several basic assumptions, which reflect lessons learned from past electronic voting studies:

- *County headquarters is kept physically secure.* We assume that the EMS is maintained with high levels of access control (locked rooms, dual person rules, no connections to the Internet, etc.) sufficient to thwart attack by outsiders. We appreciate that this is a difficult bar to attain, but if the EMS is not kept secure we know of no practical method for ensuring the security of the polling place devices it manages.

- *Software will remain vulnerable.* Experience with all kinds of security software shows that it is difficult if not impossible to produce vulnerability-free programs, and all serious reviews of voting systems have found significant security weaknesses. Therefore, we must assume the system software cannot be trusted to process malicious data without itself being subverted. This is clearly undesirable — software *ought* to be able to handle malicious data — but there is ample evidence that existing software is not secure and none that vendors can soon secure it.

- *Hardware will remain only modestly resistant to physical attack.* The locks, tamper seals, and other physical protections in current polling-place devices have generally proved easy to bypass. Given the generally low level of tamper-resistance provided by commodity seals [22] and the high cost of constructing truly tamper-resistant systems, we expect this situation to continue.

- *Polling places have little physical security.* Devices are often left unsupervised overnight at polling locations not chosen for their physical security. It would not be difficult for even a modestly dedicated attacker to obtain physical access to the devices under these circumstances. The threat we are concerned with is not that an individual device will be compromised but rather that it will be used as an attack vector against the entire county.

- *Compromise is undetectable and irreversible.* With today's voting equipment, once a device is subverted and its software replaced by malicious software, there is effectively no realistic way to detect this compromise. Because malicious firmware can be designed to emulate the correct software when subjected to any external checks, the only safe way to detect compromise is to directly examine the internal memory. This often requires disassembly of the device, which is not practical on a regular basis. Additionally, even if compromise is suspected, there may be effectively no way to reset the device to a known-good state. Many existing voting devices store their firmware in flash memory, so malicious code can overwrite the firmware and render the device forever compromised.

One consequence of these assumptions is that any equipment that ever leaves country headquarters (e.g., for deployment to a polling site) must be treated as if it is compromised. Similarly, any electronic data that comes from a polling site or from a device that has ever left headquarters might potentially be malicious. Because software cannot be trusted to handle malicious data safely, any contact that the EMS machines have with suspect data is a potential vector for compromise.

This paper focuses primarily on preventing the viral spread of malicious code, as this is the most powerful type of outsider attack known against current voting systems. While viral attacks require a significant upfront cost in terms of finding vulnerabilities in the target system and then crafting the appropriate malware, they can be deployed with minimal election-day effort, thus dramatically lowering the number of informed participants [5]. The California and Ohio reviews found viral spread vectors via essentially every channel through

which electronic data is conveyed. Moreover, the architecture of current voting systems is such that data flows in a cycle, from the EMS at county headquarters out to polling places in the field and back again. These cycles in the dataflow graph are what allow viruses to spread, so one of our core contributions is a set of recommendations for breaking these cycles.

1.4 Current Workflow

We can think of the election process as proceeding in five phases:

1. *Device initialization.* Before the election, officials use the EMS to prepare the ballot definitions and other information (such as cryptographic keying material) needed by the polling place devices to run the election. This information is then programmed into the polling place devices to prepare them for use in the field.

2. *Voting.* During voting, voters register their choices for contests, either on paper ballots, which may be either locally or centrally scanned, or on DRE consoles. At the end of the election the polling place devices, memory cards, and paper ballots are returned to election headquarters for tabulation.

3. *Early reporting.* When votes are electronically counted at the precinct (either via DRE or precinct-count opscan), the memory cards containing the results can be quickly read by the EMS to yield early but unaudited and unofficial results. In some jurisdictions, being able to produce such results for public consumption soon after the election may be an important political imperative for voting officials.

4. *Tabulation.* In the days and weeks following the election, the election officials prepare a complete official tally of the results. This involves aggregating the electronic results from the polling places, scanning any centrally counted paper or absentee ballots, handling write-ins and provisional ballots, and determining the winner of each contest.

5. *Auditing.* Finally, the election officials audit the election results. The auditing process is intended to provide confidence that the various election systems (both procedural and technical) are functioning correctly and are delivering accurate results. In most jurisdictions, the auditing procedure involves manually recounting some subset of the ballots and comparing the totals to the reported totals.

Each of these phases represents some risk to the election process and therefore is a candidate for mitigation. The remainder of this paper discusses mitigations which can be applied to each phase, with the exception of the voting phase, which we assume must remain unchanged.

2 Device Initialization

Before the election, officials must program the polling place equipment with election definition files. Typically this involves resetting each device and transferring election-specific configuration information from the EMS to the device.

On existing voting systems, device initialization is a dangerous operation, as it may create opportunities for malicious code to spread. For instance, in many systems, election definitions are written by the EMS onto memory cards which are then distributed to the polling place devices. If there are vulnerabilities in the EMS code that processes the memory cards and cards from the field are reused and inserted into the EMS, then an attacker can leverage a single malicious memory card into control of the EMS and, through the EMS, attack all the polling-place devices. Calandrino et al. [6] describe just such an attack on the Premier system.

Calandrino et al. recommend mitigating this threat by having a specialized device which erases the memory card before it enters the EMS, thus protecting the EMS from attack [6]. However, this is insufficient because the memory card is potentially an active device, not merely a passive storage medium. For example, PCMCIA "flash drives" are typically flash memory chips with an attached ATA chipset. A malicious version of such a device could pretend to be zeroed but restore the malicious data for subsequent reads.

An attacker might construct such a malicious card in two ways. First, an attacker could construct a device which appeared to be a standard card but actually contained malicious hardware of his own construction. Second, some memory cards apparently contain software-upgradable firmware [15]. Thus, an attacker with access to voting equipment at one polling place might be able to overwrite the firmware on the memory cards in that polling place, or introduce illegitimate memory cards. Although election procedures contain safeguards (e.g., tamper seals, two-person rules) designed to prevent card replacement, because even a single compromised card can infect the entire county, these procedures likely do not reduce the risk to an acceptable level.

Our goals for device initialization are necessarily limited. First, when initializing a machine or memory card that *has not* been infected or tampered with, the initialization step must successfully reset the device or card to a known-good state. We do not require that the initialization process successfully restore an infected machine or memory card to a known-good state; as described above, this is difficult to guarantee. Second, when initializing

a machine or memory card that *has* been compromised or physically tampered with, the initialization process must not enable this infection to spread any further. In particular, the EMS or initialization device must be protected throughout this process from malicious memory cards and other devices.

Our basic approach for accomplishing the second goal is to ensure that data can flow only one way: from the EMS to the device or memory card being initialized. We assume in this section that the EMS is trustworthy and has not been subverted; that will, in turn, impose constraints on other election operations to ensure that this invariant is preserved.

Single-use memory cards. For the reasons discussed above, it is not safe to insert any memory card that has ever left county headquarters into any trusted central election management PC. In the best case, we could simply treat memory cards as disposable. Before an election, fresh new cards are bought from a trusted source, inserted into the EMS to be burned with election definition files, and then inserted into the voting devices. On election night, when a memory card is received at county headquarters it is immediately sealed into a secure tamper-evident bag (e.g., a see-through, tamper-evident evidence bag) and archived permanently. The crucial security property is that a memory card, once used in an election, is never re-used and never inserted into any other machine — so if a memory card does become compromised, it cannot become a vector for infection. Moreover, because only fresh unused memory cards are ever inserted into the EMS, we can be confident that those cards are not malicious and have not been subject to physical tampering by outsiders.

Cost could be an issue. PCMCIA memory cards are old technology and as a result are expensive ($\sim$ \$20–100 per card), so buying new PCMCIA cards for each election might strain county budgets. CF cards are cheaper ($\sim$ \$8–10 for a 1 GB card). Purely passive CF-to-PCMCIA adapters are readily available ($\sim$ \$10 apiece), so one could buy one adapter per voting machine (these never need be discarded) and buy new CF cards for each election. Note that because an attacker might replace an adapter with a malicious component, the adapters must be treated as part of the polling place device to which they are fitted. If each voting machine receives 80–100 votes per election, then the cost of single-use CF cards is circa \$0.10 per vote cast, which may be affordable.

Non-standard memory cards. Some voting machines (e.g., the Sequoia Insight and Premier AV-OS precinct-count optical scanners) rely upon non-standard memory cards that have limited availability or are proprietary and can be acquired only from the vendor. As a result, disposing of these after each election is not economically feasible, so we need a safe way to reuse them from election to election.

We propose to use a stateless, single-purpose, custom-built trusted *initialization gadget* to erase and re-initialize these cards. Such a device should:

- *have no persistent state*: It should boot from PROM and should have a reset button that can be used to hardware power-cycle it.
- *implement one function only*: It should perform the sole task of erasing the card's contents and then initializing it with new data for the election, and include only enough code to support this task.
- *use an independent implementation*: It should be implemented from written specifications of the protocol to be carried out, so that there is no reuse of source code from the vendor systems.

The first requirement is intended to ensure that if there is some way for a malicious memory card or card containing malicious data to compromise the initialization gadget, this does not provide the attacker with a viral propagation path. (In particular, even if the initialization gadget is compromised by some card, it will be reset before any other card is inserted into it, so the compromise cannot spread.) The remaining requirements are intended to ensure that the initialization gadget has a trusted computing base that is small, independent of potential vendor bugs, and, ideally, verifiably correct.

This is the first instance of a concept that we will see throughout this paper — a single purpose, stateless management gadget used to replace some function that would otherwise be performed by the EMS. We use these gadgets to shift trust from one place to another where better assurance can be provided. For instance, a legacy EMS cannot be trusted to read malicious memory cards without becoming infected; in contrast, the single-function, stateless nature of our initialization gadgets gives us better assurance that a malicious memory card cannot trigger a lasting compromise of the equipment used to initialize memory cards.

We envision that, after booting, the initialization gadget would allow insertion of a single card. The gadget would then work in two phases:

- *zeroization:* The gadget first zeroes the contents of the card byte by byte. To minimize the risk of subversion, this should preferably be done without reading any data from the card.
- *initialization:* Then, the gadget copies the election-specific data onto the card, using a simple byte-for-byte copy. Once the copy succeeds, the gadget would signal to the operator (e.g., via a green light)

Device	Initialization Techniques
ES&S iVotronic	Single-use CF cards; PEBs zeroed with initialization gadget
Automark	Single-use CF cards
Hart eSlate, eScan	Single-use PCMCIA memory cards for election definitions; machines zeroed with initialization gadget
Premier AV-TSX, Sequoia AVC Edge	Single-use PCMCIA memory cards
Premier AV-OS, Sequoia Insight	Non-standard memory cards zeroed with initialization gadget

Table 1: Applicable initialization techniques for major commercial voting machines.

that the initialization cycle is complete, and the gadget should then halt so that the operator must power-cycle the gadget before initializing any more cards.

Requiring the operator to press the hardware reset button after each card is removed and before the next card is inserted ensures that the initialization gadget is restored to a known-good state before each card is initialized, thus preventing viral spread through the gadget.

The major difficulty with such devices is that they require a new line of engineering: new hardware and software must be constructed to meet the requirements and the entire device must then be certified. Aside from the cost issues, this would significantly delay deployment due to the need to certify the devices.

As a cost trade-off, it might be possible to approximate such gadgets with properly configured general-purpose PCs. This would provide considerably weaker security guarantees. A PC, even with hard disk removed and booting from CD-ROM, is not necessarily stateless, since infection can persist, for example in updatable BIOS firmware [17]. Furthermore, the additional, unneeded functionality included in PCs vastly increases the attack surface of the gadget. It may be possible to obtain a modest degree of additional insulation by running the initialization software in a virtual machine, however as there have been published exploits [30] for escaping from virtual machines, it is probably insufficient to run on a virtual machine without the host PC also being stateless.

To prevent viral spread of malicious code between polling place devices, any memory card that is re-used should be permanently married to a single device. The card should never be used in another voting machine. To ensure that the association between memory cards and machines is not inadvertently broken, we recommend that cards be initialized by bringing the initialization gadget to the voting machine, removing the memory card from the machine, initializing it, and immediately replacing it into the voting machine.

We emphasize that re-using memory cards (even with a trusted initialization gadget) is fundamentally less safe than the single-use approach, and should be used only where the single-use approach is not feasible.

Network-based initialization. Some machines are initialized not with a memory card but by a network connection (Ethernet, serial, parallel) to the EMS. Reengineering these systems to be initialized in some other fashion seems impractical. Rather, we propose developing another initialization gadget that is able to speak just enough of this network protocol to instruct the machine to reset itself and to transfer any needed configuration information. Such a device should be connected to only one voting machine at a time. As before, we require the operator to power-cycle the initialization device after disconnecting it from one voting machine and before connecting it to the next. The security that can be obtained in this way is fundamentally limited: if the voting machine is compromised, it can refuse to reset itself, so the best that can be done is to try to limit the spread of infection.

The voting system produced by Hart uses a hybrid initialization system that combines a network connection and memory cards [18]. To initialize a Hart eSlate, eScan, or JBC, one must first connect the machine by Ethernet or parallel cable to SERVO, which then sends a command asking the machine to reset its vote counters and other state.[1] Also, one must initialize a removable PCMCIA memory card with the election definition. We recommend initializing Hart machines using (a) a trusted device that emulates SERVO (to send the reset command), and (b) single-use memory cards for election definitions, one per machine per election.

Even if secure initialization procedures are followed, the mere presence of network initialization is a threat that must be dealt with. For instance, in the Hart voting system, voting machines are networked in the polling place, with the same network ports used for both initialization and for device control during elections. Because any one compromised machine might compromise all other Hart machines it is networked to, to limit viral spread we also recommend that all of the Hart voting machines within a polling place be married to each other: they should remain together throughout their lifetime. Some other DREs (e.g., the ES&S iVotronic) use sneaker-net to net-

[1] SERVO is connected to eSlates indirectly, via a JBC that relays messages from SERVO to the eSlate.

work all the machines in a single polling place, which creates a similar risk; we recommend the same policy be applied to those systems as well.

Firmware upgrades. The problem of firmware upgrades is distinct from, but related to, pre-election initialization. Even if the correct firmware distribution is verifiably available to election officials, the firmware loading process presents its own risks.

Today, one common way to upgrade the firmware on voting machines is to create a memory card containing the firmware upgrade and insert it one by one into each of the voting machines. This creates a dangerous opportunity for rapid viral spread of malicious code: a compromised machine could overwrite the memory card with malicious data that will infect each machine the card is then inserted into.

In principle, if we had a memory card with a hardware-enforced write-protect switch, we could initialize that memory card, set the switch, and then use the card to upgrade every voting machine one by one. But this requires absolute confidence that the write-protect functionality is enforced via a hardware interlock (not in software) and that the memory card's firmware cannot be compromised or overwritten while the write-protect switch is set. These mechanisms are technically possible with both Flash and EEPROM, but it is not clear whether there is any commercially available memory card that meets these requirements, nor is it clear how to tell whether any particular card can be used safely in this way.

One can defend against this threat using the same procedures outlined above. The most secure approach involves disposable, single-use memory cards: for each voting machine, we burn a separate memory card with the upgrade, insert that card into that machine, and then securely dispose of that memory card. Note that this procedure still does not guarantee that compromised machines will get the new firmware — malicious firmware can simply ignore the update — it is intended solely to prevent viral spread. Also, this procedure does not guarantee that the upgrade is legitimate or prevent viral spread from the EMS to the voting machines; a malicious EMS could simply burn malicious firmware onto the memory card. The intent is solely to prevent the firmware upgrade process itself from becoming a vector for viruses to spread from voting machine to voting machine.

If disposing of the cards is not possible, the firmware upgrade can be performed using whatever existing memory card is married to the machine (as described above). The card could be removed from the machine, initialized with new firmware with our custom initialization gadget, and then reinserted into the machine for reinstall. This approach requires extremely careful procedures: if a card from infected machine A ends up in uninfected machine B, then machine B will become infected. Because of the chance of this kind of mishandling, the disposable approach is safer, though more expensive.

3 Early Reporting

Precinct opscan devices and DREs output records of cast votes on memory cards. In the procedures typically employed by counties, these cards are loaded one after another onto the EMS, which tabulates the votes from the cards and outputs the election results. This procedure is unsafe: DREs and other precinct devices can be compromised; the compromised devices can be instructed to write arbitrary data to the memory card; malicious data on a memory card can compromise software in the EMS used for tabulation; and if this happens the entire county's results would be cast into doubt.

To obtain vote counts that are correct, one must process votes only in forms that cannot allow compromise of the EMS: the optical scan ballots themselves; DRE VVPATs; and summary tapes from any precinct devices. We consider this trustworthy count in Section 4. Unfortunately, this process could take several days to complete. However, many jurisdictions currently conduct an unofficial count on election night, to provide early reporting for candidates and the press. For example, election results may need to be available before midnight if they are to be included in the next day's newspapers.

Unfortunately, the best procedures we are able to describe for early reporting are extremely brittle. By far the safest approach is to avoid any kind of early reporting, and perform only a single trustworthy count — but given the large number of jurisdictions which do early reporting, we consider in this section how an early count can be obtained most securely.

Early reporting is applicable only when precinct devices create vote records in electronic form. For vote-by-mail or other central-opscan voting setups, there are no such electronic records; the paper ballots must all be scanned to determine the results of the election. Before the scan, no total is available; once the scan has completed the available total is accurate and trustworthy, provided it is audited as provided for in Section 5.

Sacrificial EMSs. We recommend the use of a sacrificial EMS for early reporting, as proposed originally by Calandrino et al. [6, Sect. 6.10]. The sacrificial EMS is an entirely separate copy of the EMS that runs on a system separated by an air gap from all other systems. The election definition database generated on the main EMS is replicated onto the sacrificial EMS before any memory card is inserted into the sacrificial EMS. A write-once medium, such as CD-R, is used to transfer the database,

rather than a network connection. Memory cards from the field are only ever inserted into the sacrificial EMS, and never into the main, trusted, EMS.

The sacrificial EMS must be considered potentially compromised once any memory card has been inserted into it. Thereafter, the sacrificial EMS must never be connected to any other system, directly or indirectly. The prohibition on indirect connection means that any memory card or other writable media inserted into the sacrificial EMS must not subsequently be inserted into another system. For this reason, early reporting can only be safely used when the memory cards are discarded rather than reused.

It is tempting to think that the memory card could be erased with a gadget and then reinserted into the system. However, this is unsafe. The security of this approach would require not only that the gadget not serve as a viral vector in the face of malicious cards, as described in Section 2, but that it be guaranteed to erase the cards successfully, to block the viral propagation path through the sacrificial EMS. If the gadget cannot be guaranteed to erase infected cards, then a compromised EMS can infect all the cards in the system. This goal is implausibly difficult to achieve. Even if the memory card is not running malicious firmware, it might contain malicious data that triggers a bug in the gadget, thwarting the erasure operation. If the memory card is running malicious firmware, even an ideal gadget cannot guarantee erasure.

In addition, the sacrificial EMS must be erased securely before being used again. At minimum one must erase the hard drives with an erasure tool booted from secure media, but it is not clear that this is sufficient, for the reasons discussed in Section 2. If the cost is not prohibitive, it may be better to retire the computer acting as sacrificial EMS after every election, retaining it as evidence. It may also be possible to remove only the hard disk, though again this might not prevent all infections. Another possibility is to run the sacrificial EMS inside a virtual machine and erase it after the election, though this only works if the VM can resist subversion by malicious guest software, which is contrary to our basic assumption that software cannot be trusted to handle malicious input. Moreover, even if the VM software itself is secure, it must be configured securely, kept up to date, etc. all which are likely to be challenging for election officials.

We have already observed that the sacrificial EMS, once compromised, can rewrite each memory card subsequently connected to it, and that this can lead to a viral infection mechanism if these cards are reused. There is an additional risk that remains even if memory cards are not reused and instead retained as evidence: An infection of the sacrificial EMS can rewrite memory cards arbitrarily; the rewritten cards will be useless as evidence in a later investigation. The attacker could misdirect investigators by leaving behind evidence suggesting that some other precinct device than the one he compromised was the source of the infection.

Accordingly, if the memory cards in use expose hardware write-protect switches, these switches should be engaged before the cards are inserted into the sacrificial EMS. As in Section 2, it is crucial that write protection be implemented via a hardware interlock rather than a software flag to be obeyed by the drive's firmware. A more general solution is to develop a memory-card archiving gadget that writes an image of each card to a CD-R. The archiving gadget must be applied to each card *before* it is read in the sacrificial EMS, which may introduce a slowdown that reduces the benefits of early reporting. As with all other gadgets, the archiving gadget would need to be stateless to avoid becoming an infection vector, so a new CD-R would be required for each memory card.

Warning: Unfortunately, electronically reading results is inherently risky. We cannot prevent a virus from infecting the sacrificial EMS and every memory card ever inserted into it. If any such memory card is ever inserted into trusted equipment (e.g., the trusted EMS), then the entire county can become irreversibly infected. Because a single seemingly minor procedural lapse can have such severe consequences, the safest approach is to avoid early reporting if at all possible.

4 Tabulation

As discussed in Section 3, while a sacrificial EMS can prevent viral spread *between* polling place devices, it does not prevent viral spread to the EMS during the tabulation phase. A single compromised polling place device from a precinct count optical scan or DRE device can potentially compromise the EMS. A compromised EMS can alter all of the election results or infect all of the polling place equipment with malicious code, so the potential for any one polling place machine to infect the EMS poses a serious problem.

The source of the problem is that the memory cards used to transfer precinct or device totals represent too rich a channel to be able to guarantee that the EMS can read them safely. One alternative is simply to abandon the vote tallying aspect of the EMS entirely and manually add the vote totals reported by the precincts. However, this is clumsy and error-prone and obviates much of the attraction of using electronic voting in the first place.

We observe that the tabulation function of the EMS actually consists of two functions: vote collection (reading the memory cards) and vote aggregation (computing the vote totals and determining who won). Only the first function represents a threat to the EMS. Separating these two functions allows us to contain the effect of

cards containing malicious data — their votes still cannot be trusted but we can have confidence that the non-malicious cards have been read and tabulated correctly.

We describe two strategies for performing this separation. The first strategy prevents the EMS from being compromised in the first place, but at the cost of more complicated workflow. The second strategy does nothing to prevent EMS compromise but allows compromise to be detected.

4.1 Preventing EMS Compromise

Because the EMS cannot be trusted to read the memory cards from the polling place devices correctly, this step needs to be replaced with something safer. As the central count optical scanner is assumed to be inside the election central security boundary, results from it can be directly electronically fed into the EMS — depending on the system the scanner may even be directly operated by the EMS. Similarly, if we are willing to rescan all precinct-counted ballots, we can do so safely, and then reconcile the count with the totals reported from each precinct. This is a simple and effective countermeasure that could be deployed in jurisdictions that use paper ballots.

However, if we wish to avoid rescanning and/or use DREs, then we need some way to sanitize the data read by the memory cards before it is fed into the EMS, as shown in Figure 1. The difficult part of this process is the sanitization stage, which must provide a high level of assurance that the sanitized data cannot represent a threat to the EMS. The simplest and safest approach is to have the per-device totals manually re-keyed from the summary tapes produced by each device. This eliminates all electronic communications between the compromised devices and the EMS; malicious code transmission in the remaining low bandwidth channel is unlikely.[2]

Although safe and simple, manual re-keying has significant drawbacks in terms of time and expense. Prices for commercial data entry services vary dramatically depending on the type of job and the accuracy level desired (number of fields per record; whether the paper must be directly handled or can be scanned; number of independent key-ins to detect entry errors), but we can take as a reasonable benchmark that the cost will be on the order of $1 to $10 per record, with each summary tape comprising a single record. These costs scale directly with the number of devices, so a county with 2,000 machines might incur an additional expense of $2,000 to $20,000 (less than $0.10 per vote) per election. Techniques such

[2]Even the shortest shellcodes are approximately 30 bytes long [29]. Results tapes will contain letters and digits only; alphanumeric shellcodes are several times as long [27]. Compromising vote counts requires a more intricate payload than spawning a shell. Conservatively, a channel capacity of many hundreds of bits seems required for this kind of compromise.

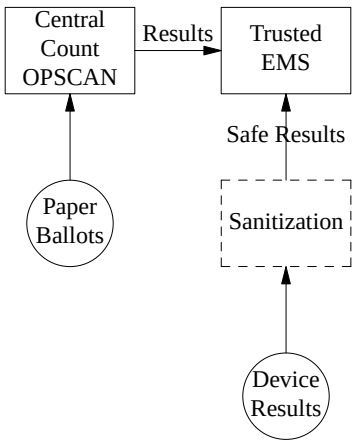

Figure 1: Tabulating with sanitization

as multiple independent entries can be used to achieve an arbitrarily high degree of accuracy (one commercial services quotes accuracy rates of "99.995% or better"), though of course these come at additional cost.

It is harder to estimate the effect on tabulation time. At minimum the summary tapes must be gathered and entered into the system, so it is reasonable to expect a somewhat higher level of latency than in current digitally read systems. If the data entry is outsourced, there will of course be additional transport latency and issues of the security of summary tapes themselves. For instance, are the results scanned and electronically transmitted or are the actual summary tapes sent to the processing center? If policies or practical realities forbid outsourcing, then the county will need to have staff on hand, which significantly increases logistical issues.

An alternative to manual re-keying is to machine-read the summary tapes. The text on these tapes is often difficult to read [14] and it is unlikely that an appropriate degree of accuracy can be achieved without assistance, so with existing devices it must be OCRed and then manually checked and corrected — there is not enough redundancy in the tapes to allow automatic error detection and correction. In fact, many data entry services use OCR followed by manual correction as an alternative to full manual entry. Alternatively, polling-place devices can be augmented to add a more machine-friendly representation (e.g., a 2-D bar code such as DataMatrix [19], which could be used to check the OCR). As an additional security measure, the machine-readable section could include a digital signature by the polling place device, allowing for detection of substituted summary tapes. We note that because both of these changes require modifying device software, it is unlikely they can be developed and certified prior to the November general election. In addition, because cryptographic practice in current systems generally makes use of a system-global or county-global sym-

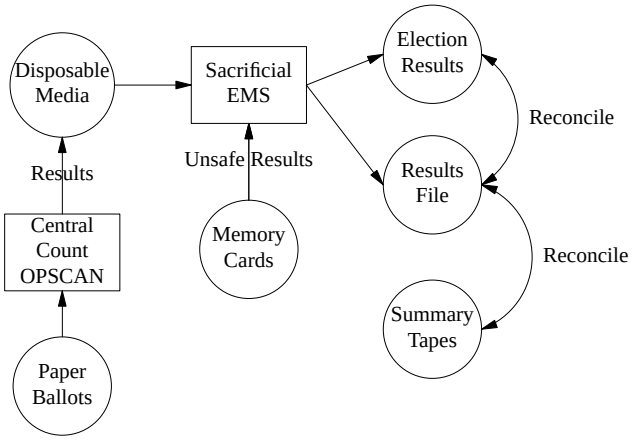

Figure 2: Tabulating with a sacrificial EMS

```
Device     Hoover    Roosevelt
 1             40           50
 2             45           56
...
Total        1010         1011
```

Figure 3: An example results file, in a simple machine-parseable format.

metric integrity key, providing a per-device signature key would likely require nontrivial software and procedural changes.

One limitation of machine scanning is that it provides a significantly higher bandwidth channel into the EMS — image processing libraries in particular are notorious for having security vulnerabilities (see, e.g., [3, 24]) — than does manual entry and thus represents a correspondingly higher risk of EMS compromise via malicious data. In addition, it is not clear that thermal-printed summary tapes are actually suitable for scanning and OCRing [14].

4.2 Detecting EMS Compromise

An alternative approach is to assume that the polling place devices are usually uncompromised and to use a procedure that allows error detection and investigate when discrepancies are found.

The workflow, shown in Figure 2, is similar to — and could even be integrated into — the early reporting workflow. As described in Section 3, we feed the memory cards into the sacrificial EMS and tabulate there, and then either discard or sanitize the cards. However, we must read the centrally counted ballots on a trusted scanner (perhaps attached to a trusted EMS depending on the system) and then carry those results to the sacrificial EMS on disposable media, ensuring an accurate, independent count of those ballots. As in Section 3, a single compromised memory card can compromise the sacrificial EMS and invalidate the results. Thus, we need a mechanism for checking the results; specifically, we need to check that (1) the memory cards were read correctly by the sacrificial EMS and (2) the results from the memory card were added correctly.

To enable these checks, we propose that the EMS output a "results file" in a machine readable format (tab-delimited, CSV, etc.) listing vote totals for each candidate in each contest for each device, such as the sample shown in Figure 3.

To check that the cards were read correctly, election officials randomly sample the devices during the official canvass and compare the totals in the results file to those printed on their summary tape. The usual statistics [4, 2, 23, 28] for the required number of samples for precinct-based auditing apply here as well. An additional check on the correctness of the device results can be provided by having each device digitally sign its results with a per-device key (as opposed to the system-wide keys used by most current systems).

We note that another source of information about the data that should have been fed into the EMS can sometimes be found on the devices themselves. The Hart system, for instance, stores a duplicate copy of each vote cast on the polling place device. However, downloading this data would require yet another gadget, which seems substantially more onerous than using the results tape.

Once the individual results are checked, the tabulation process then must be checked. This can be done by using generic spreadsheet tools (e.g., Excel) to independently read the file and compute the totals and compare them to the ordinary election reports provided by the EMS. The most significant limitation of this technique is that extreme care must be taken with the results file. Because it is prepared by the potentially compromised EMS, it may contain malicious data that could compromise whatever tool is used for checking. This risk can be mitigated in two ways. First, the data can be processed with tools specifically designed to handle malicious data (e.g., carefully written Perl scripts). Second, the data can be filtered to ensure that the file conforms to a restricted format prior to being processed with a more generic but also potentially more sensitive tool such as Excel. In addition, the data can be checked with multiple independent tools on multiple platforms, forcing the attacker into the more difficult task of devising a single malicious file that produces consistent results across all such platforms.

This procedure can be extended to allow public checks of the EMS operation. Once the reconciliation phase has successfully completed, election officials would publicly post the results file and scanned images of the results tapes to a Web site or other public repository. Any third

party can independently perform the appropriate checks that the EMS has added the votes correctly. In addition, if each machine includes digital signatures on its results and those signatures are propagated into the results file, a third party can quickly achieve some confidence that the per-machine results being reported have not been modified by a compromised sacrificial EMS without resorting to examining the results tapes, at the cost of requiring very careful key management by election officials.

Discrepancies in either of these processes, if they cannot be ascribed to human or procedural errors, indicate that at least one of the polling place devices and, potentially, the EMS has been compromised. Consequently, discrepancies must be investigated, and in some cases it may be necessary to recount all the ballots using a more secure method.

The major advantage of this technique vis-à-vis that presented in Section 4.1 is that it has minimal impact on the current workflow. The major impact is the burden of operating the sacrificial EMS required for any electronic results processing. If early reporting (Section 3) is used, the memory cards need only be read once.

This procedure is inherently more risky than that described in Section 4.1. As with early reporting, because untrusted cards are read by machine, procedural errors can lead to viral propagation. In addition, the post-election reconciliation stage is more complicated (albeit more efficient as only a small number of summary tapes are reviewed) than the manual-entry technique described in Section 4.1, and is dependent on the correctness of the software — which of course must be written and certified — that processes the results file. This technique may also require some modifications to EMS software to allow for exporting the results file. By contrast, all the systems we are aware of allow for manual data entry, so it is likely that the approach described in Section 4.1 can be executed with no change to the system software.

5 Post-Election Auditing

While the procedures that we recommend in Sections 2–4 can help slow the spread of malicious software among the components of a voting system, they cannot prevent all such attacks. For instance, they cannot defend against insider fraud, nor do they provide any way for observers to independently verify election results. Further safeguards are necessary: following every election, a post-election audit should be carried out to ensure that the totals from the tabulation phase agree with the voter-verifiable paper ballot records created during the voting process, and to ensure that election observers can verify that this is the case [21, 25].

While conducting a thorough audit may be time-consuming, it provides a higher level of confidence in the integrity of the result than any other mechanism we have been able to identify. Unlike the early reporting and tabulating phases, where software and hardware are trusted to behave correctly in the interests of speedy reporting, the auditing phase should provide a way to verify the correctness of the count, without requiring trust in any computer component. Election officials generally take several days or weeks to release final "certified" results, and they can use this time to conduct an audit that might detect evidence of fraud (even if they cannot necessarily correct the damage).

To ensure that audits meet their transparency goals, audits must be open to public observation, and it must be possible for observers monitoring the audit to verify that each contest was decided correctly. Conversely, audits must not endanger the secrecy of the ballot; for instance, they must not create new opportunities for vote-buying.

5.1 Auditing Paper Ballots

For paper ballots, whether marked by hand or via a ballot marker, most jurisdictions employ statistical auditing methods where only a fraction of ballots are manually reviewed. The goal of a statistical audit is to establish with a given level of confidence that if all ballots were to be hand counted, the election outcome would remain the same. If discrepancies are found between the paper and electronic records, neither set of records should be discarded out of hand. Instead, officials should launch an investigation to determine the cause of the errors and the extent to which either set of records can be trusted.

One standard method for post-election auditing is first to publish the election tallies broken down by precinct, then to manually recount all the ballots in a randomly selected set of precincts and compare the manual tallies to the previously published electronic tallies. If discrepancies between the two tallies are sufficiently rare, then this provides probabilistic evidence that a 100% manual recount would not change the outcome of an election.

How many precincts will be sampled is generally specified as part of a jurisdiction's auditing procedures. Some procedures call for a fixed percentage, while better procedures, like those that would be mandated by H.R. 811 [1], use a "tiered" approach, where thresholds for the margin of victory determine the auditing percentage. Unfortunately, these strategies will occasionally yield substandard levels of statistical confidence. Consider a race involving 500 precincts, roughly the average number for a U.S. congressional district. Under H.R. 811, if the margin of victory was slightly greater than 2%, auditors would sample 3% of precincts and obtain a 55% confidence level. With a margin of victory slightly greater than 1%, auditors would sample 5% of precincts and achieve a 48% confidence level. A bet-

ter strategy would be to target a high, fixed confidence level — perhaps 95% or 99%. Auditors could then calculate the necessary sample size using methods such as those proposed by Aslam et al. [4, 28, 23, 2].

Per-precinct audits can be inefficient: sometimes one must recount a substantial fraction of the precincts to gain sufficient confidence [4, 28]. For example, in a race with 500 precincts and a 1% margin of victory, auditors would need to sample around 28% of precincts to achieve 99% confidence. It would be more efficient to audit on a per-ballot basis, choosing a random sample of ballots (not precincts) to manually recount. Calandrino et al. have proposed one way to do this, based on optically scanning the paper ballots while simultaneously stamping each ballot with a serial number [7]. Unfortunately, existing central-count optical scan machines do not provide this capability. Also, their scheme does not meet the transparency goals of providing observers a way to verify the election outcome for themselves: for independent observers to be able to verify election results, they would need a copy of all of the electronic cast vote records, but disclosing this information to members of the public introduces opportunity for coercion and vote-buying due to pattern voting and other attacks [26]. Thus, while ballot-based auditing might be suitable as a means for election officials to augment the sensitivity of per-precinct auditing, it is not yet suitable as the sole audit mechanism. We consider it an important research problem to find a way to perform transparent, privacy-preserving per-ballot audits with existing voting equipment.

5.2 Auditing DRE Paper Records

Paper records produced by DREs present different auditing challenges. The thermal-tape printers used by most DREs to print VVPATs make poor ballot printers. Ideal ballot stock is heavy, so it can be handled by scanners without causing jams; comes in discrete units, so it can be reordered randomly; and is premarked to enable easy opscan registration. By contrast, thermal-printer tape is flimsy, continuous,[3] and initially empty. The continuous tape reveals the order in which voters cast their vote, which poses risks for ballot secrecy.

Assuming VVPAT records are available, one simple approach is to perform a 100% manual count of the VVPAT records. This may be reasonable in jurisdictions (such as Diebold- and Sequoia-using precincts in California) that use DREs only for accessibility, because then the number of records to be recounted is limited. However, this is awkward and impractical when many votes are to be counted [16]. The natural alternative in these cases is to apply per-precinct auditing.

One small improvement is to perform per-machine auditing, where a random sample of machines is selected rather than of precincts. This requires the ability to publish election tallies broken down by machine, rather than by precinct; current voting systems may or may not have this capability, but it does not seem conceptually difficult to extend existing EMSs with this capability. Per-machine auditing does introduce voter secrecy issues when one machine is used by only a small number of voters, so this model seems best suited for jurisdictions where all voters vote on a DRE (with VVPAT).

As with optical scan, per-ballot auditing for DREs would provide greater sensitivity, but providing transparent per-ballot auditing without violating ballot secrecy seems challenging. An additional challenge is finding the correct ballot without having to manually go through the entire tape. One approach we considered was to have the DRE number each entry on the VVPAT, but this of course makes the ballot secrecy problems worse. For these reasons, we believe more research would be needed before per-ballot auditing for DREs is suitable for use with public elections.

Spoiled VVPAT records. One of the primary shortcomings of any method of VVPAT auditing is the difficulty of detecting *presentation attacks* by a malicious DRE [20]. For instance, the DRE could try to misrecord the voter's vote both electronically and on the VVPAT record; if the voter notices, then the DRE could allow them to change their vote and could behave honestly from then on for that voter. In this way, the electronic and VVPAT record would always be consistent, and if only some voters check the VVPAT record carefully enough to notice errors, such an attack could be very effective. In one study, over 60% of experimental subjects did not notice such errors [13]. Of course, a malicious DRE might attack only a small fraction of voters, making it harder to detect such an attack.

We observe, however, that this attack leaves evidence on the VVPAT in the form of spoiled ballots. In a fair system, we would expect that the distribution of spoiled ballots would be unbiased: Smith voters are just as likely to inadvertently vote for Jones as Jones voters are to vote for Smith. However, a presentation attack creates a biased error distribution: an attacker favoring Smith moves votes from Jones to Smith; when voters correct the errors we see more spoiled ballots for Jones than we would otherwise expect. In a sufficiently large election, it should be possible to distinguish random and malicious effects. For instance, in a 100,000 vote election with two

[3] Cut-and-mix paper-trail printers have been considered as a means to creating a VVPAT; these would reveal less information about the order in which ballots were cast but are not in widespread use today. Another flawed approach is to cut the paper tape into individual ballot records after the voting but before the tallying. This produces small, flimsy, difficult-to-process records.

candidates and a small margin of victory, an attack that shifts 2,000 votes can be detected against a background error of 5% spoiled ballots even if only $1/3$ of voters actually check the VVPAT. We note that presentation attacks are not the only factor that could produce biased errors — for instance, there could be a UI error which tends to cause voters to vote preferentially for Jones versus Smith — however any such anomaly is evidence of a problem which needs investigation.

Because this technique works only when large numbers of votes are analyzed, we need to measure the vote shift across the entire county. To obviate examining every VVPAT to count spoiled ballots, DREs could be modified to emit spoiled ballot and vote shift statistics on a per-precinct basis. These could be published and then confirmed as part of the precinct-based audit. This auditing method is itself subject to a more sophisticated ballot spoiling attack that eliminates the biased error distribution by introducing dummy cancellations [11], but this attack requires the VVPAT to perform extra vote printing, which might be noticed by voters or poll workers, so it is more difficult to mount than a simple presentation attack and might be fixable by future improvements of the VVPAT.

VVPAT-less DREs. Not all DREs are deployed with VVPATs, and some states, such as Pennsylvania, prohibit their use. Such machines typically produce only summary tapes. We are unable to offer any practical auditing measures for these devices. Although some systems (e.g., Hart) provide redundant electronic records which might be useful for forensic purposes, because those records are under the control of the same software that controls the devices, they are not suitable for an end-to-end audit.

6 Putting It Together

The previous sections describe a series of independent techniques intended to increase the security of individual phases of an election. In this section, we describe how to apply these techniques to the combinations of voting equipment in widespread use. We consider three common scenarios: optical scan with electronic ballot markers (EBMs) for accessibility, optical scan with DREs for accessibility, and pure DRE systems.

6.1 Opscan with EBMs for Accessibility

Scanning Twice. The most secure operational mode for opscan is to use both precinct count and central count: all ballots are scanned and counted on precinct count machines to detect overvotes, undervotes, and other kinds of voter error, and then the paper ballots are returned to election headquarters to be scanned again on central-count scanners. Because errors in the precinct count will be detected when the ballots are centrally counted, this approach dramatically reduces the impact of attacks on the precinct optical scanners, In addition, if the one-way dataflow procedures described in Section 2 are followed, externally introduced viral attacks on the precinct optical scanners are impractical, which means that an outsider would need to attack each device individually, thus dramatically increasing the cost of such an attack.

The primary attack vector on this type of system is to attack the paper ballots directly, for instance by having poll workers introduce bogus paper ballots into the system. However, because the precinct counts are compared against the central count, to avoid detection attackers would also have to feed these ballots into the precinct scanner. This blocks some attack vectors (e.g., stuffing the ballot box itself) and limits the damage to only those polling locations where poll workers have been subverted. A compromised precinct scanner can still mount limited attacks, for instance, "accepting" overvoted ballots, thus invalidating some ballots, or rejecting valid ballots, perhaps discouraging votes for one candidate. Insider attacks are not prevented, but can be detected during the audit if an appropriate level of statistical confidence is provided.

Precinct Count with Central Aggregation. In systems where only precinct count is used, compromise of the precinct count scanners becomes a more significant issue. Sufficiently aggressive auditing can detect tampering, but as described in Section 5, this may be costly. This approach cannot prevent compromise of the precinct scanners, but the measures described in Sections 2–4 should prevent viral spread, thus forcing an outside attacker to individually attack each precinct scanner one by one. As with double-scanning, non-technical ballot stuffing and insider attacks remain possible.

Electronic Ballot Markers. Electronic ballot markers can be used to support accessibility with minimal impact on the security properties of these optical scan systems. The ballots output by the EBM can be fed into the ordinary paper ballot processing path.

Unlike manually marked ballots, EBM ballots are subject to presentation attacks by a compromised EBM. This attack is to some extent alleviated by the increased resistance to viral spread provided by one-way dataflow measures. An outsider who wishes to mount a presentation attack on the EBM must attack each device individually (this may be somewhat easier than attacking optical scanners, because the voters have direct access to the EBM). If the EBMs are used only for accessibility, the

limited number of votes cast on the EBM significantly reduces the impact of this attack.

6.2 Opscan with DREs for Accessibility

Many counties have deployed a hybrid system consisting of manually marked opscan ballots, with one DRE available in every polling location for accessibility. The safest way to utilize these systems is by following the approaches outlined in Section 6.1, using the DRE purely for accessibility, and performing a 100% manual recount of the DRE VVPAT records — effectively using the DRE as an EBM. Because few votes will be cast on the DRE, the 100% manual recount may be operationally feasible. This approach mirrors the model required in California for the Premier and Sequoia systems [8, 9].

This approach has similar security properties to opscan with EBMs. As with EBMs, there is a risk of presentation attacks, though this risk is largely mitigated by the difficulty of compromising a large number of devices without insider access and by the limited number of votes cast on DREs in such a system.

6.3 Pure DRE systems

Finally, we consider DRE-only systems. In any county of reasonable size, it will be impractical to perform a 100% manual recount of the VVPAT records. Rather, the DRE results must be aggregated using the techniques described in Sections 3 and 4. As before, auditing can detect some kinds of attacks, but this may require a significant increase to the number of precincts audited compared to current practice. In principle, we might expect better specificity because the DRE records should normally contain no errors, and so any error is suspicious. However, experience indicates that DRE machines do produce discrepancies between tallies, even in cases where there is no evidence of attack [14, 12], and thus it is not clear that the specificity is in fact superior, or that it is possible to recover from even clear errors.

The security attainable with DREs is inferior to what is attainable with an opscan system (Section 6.1) in two significant respects. First, presentation attacks become a serious issue when all votes are cast on DREs. Auditing spoiled ballots (Section 5.2) may help ameliorate these attacks in elections of sufficient size, but probably won't in small local races. Second, false counting attacks are both easier to mount and harder to detect and compensate for: easier to mount because it is easier to obtain access to the DREs than the optical scanners; harder to detect and compensate for because a complete manual recount of the entire election is impractical except with strong evidence of fraud. By contrast, a complete central rescan of all paper ballots is practical, though expensive.

These risks are partially ameliorated by the one-way dataflow techniques described in Sections 2–4, which force an attacker without insider access to individually tamper with each machine that he wishes to compromise. While significant risks do remain, this is a significant improvement over current systems, where an outsider can attack a single device and compromise the entire county.

7 Conclusions

This paper has described a set of techniques for hardening the operation of existing voting systems. When used, these techniques provide the following security properties:

- *Containment of viral spread.* A single compromised polling place device cannot be used to compromise either the EMS or other polling place devices in another polling location.
- *Correct vote tabulation.* Although a single compromised polling place device can report false results, the results from uncompromised devices are reported and tabulated correctly.
- *Some detection of compromised single devices.* The auditing and reconciliation procedures will detect many instances of compromise in which ballots, VVPAT, or summary tapes remain correct.

These properties represent an improvement over the current deployed systems, which are extremely vulnerable to single-point failures difficult to detect and impossible to correct. However, our proposals are not intended to provide perfect assurance of the correctness of results. As far as we know, that is not possible given the current or likely future state of voting machine technology. The mechanisms we proposed reflect a compromise between the competing imperatives of retaining the current systems and providing confidence in the results of the election.

Acknowledgments

We thank Matt Blaze, Dan Wallach, and our anonymous EVT reviewers for many insightful comments and ideas that significantly improved this paper. This work would not have been possible without our colleagues on the California TTBR whose work forms the necessary background for this work. During the TTBR, we had many discussions with them that shaped our thinking on many of these topics.

For productive discussions on the operational parameters of the environment in which these systems are deployed, we thank the California Secretary of State's

office, especially Secretary Debra Bowen and Deputy Secretary Lowell Finley, as well as Clerk/Recorder Freddie Oakley and Director of Technology Tom Stanionis from the Yolo County Elections office.

We also thank CommerceNet for the kind use of their facilities during the preparation of this paper.

Of course, any errors or flaws in this paper are the fault of the authors alone.

References

[1] 110TH CONGRESS. H.R. 811: Voter confidence and increased accessibility act of 2007.

[2] APPEL, A. Effective audit policy for voter-verified paper ballots. Presented at 2007 Annual Meeting of the American Political Science Association, Sept. 2007. http://www.cs.princeton.edu/~appel/papers/appel-audits.pdf.

[3] APPLE COMPUTER. About the security content of iPhone v1.1.2 and iPod Touch v1.1.2 Updates. http://docs.info.apple.com/article.html?artnum=306993, Nov. 2007.

[4] ASLAM, J. A., POPA, R. A., AND RIVEST, R. L. On estimating the size and confidence of a statistical audit. In *Proc. 2007 USENIX/ACCURATE Electronic Voting Technology Workshop (EVT '07)* (2007).

[5] BRENNAN CENTER TASK FORCE ON VOTING SYSTEM SECURITY. The Machinery of Democracy: Protecting Elections in an Electronic World, June 2006.

[6] CALANDRINO, J. A., FELDMAN, A. J., HALDERMAN, J. A., WAGNER, D., YU, H., AND ZELLER, W. P. Source code review of the Diebold voting sysem. Part of [10], Aug. 2007.

[7] CALANDRINO, J. A., HALDERMAN, J. A., AND FELTEN, E. W. Machine-assisted election auditing. In *Proc. 2007 USENIX/ACCURATE Electronic Voting Technology Workshop (EVT 07)* (Aug. 2007).

[8] CALIFORNIA SECRETARY OF STATE. Diebold Election Systems, Inc.: Withdrawal of approval/conditional reapproval — October 25, 2007 revision. Part of [10], Oct. 2007.

[9] CALIFORNIA SECRETARY OF STATE. Sequoia Voting Systems: Withdrawal of approval/conditional reapproval — October 25, 2007 revision. Part of [10], Oct. 2007.

[10] CALIFORNIA SECRETARY OF STATE D. BOWEN. "Top-To-Bottom" Review of voting machines certified for use in California, 2007. Online: http://sos.ca.gov/elections/elections_vsr.htm.

[11] CRANE, R. E. Paper Trail Manipulation III. NIST Workshop on Developing an Analysis of Threats to Voting Systems, Nov. 2006. http://vote.nist.gov/threats/PaperTrailManipulationIII1.pdf.

[12] ELECTION SCIENCE INSTITUTE. DRE Analysis for May 2006 Primary: Cuyahoga County, Ohio, Aug. 2006. http://bocc.cuyahogacounty.us/GSC/pdf/esi_cuyahoga_final.pdf.

[13] EVERETT, S. P. *The Usability of Electronic Voting Machines and How Votes Can Be Changed Without Detection*. PhD thesis, Rice University, 2007.

[14] FELTEN, E. Evidence of New Jersey Election Discrepancies. http://www.freedom-to-tinker.com/?p=1266, Mar. 2008.

[15] GALBRAITH, R. Firmware update utility for Lexar CompactFlash nearing release, July 2001. http://www.robgalbraith.com/bins/content_page.asp?cid=7-4112-4114.

[16] GOGGIN, S. N., AND BYRNE, M. D. An Examination of the Auditability of Voter Verified Paper Audit Trail (VVPAT) Ballots. In *Proc. 2007 USENIX/ACCURATE Electronic Voting Technology Workshop (EVT 07)* (Aug. 2007).

[17] HEASMAN, J. Implementing and detecting a PCI rootkit, Nov. 2006. Preliminary version presented at BlackHat Europe 2006. http://www.ngssoftware.com/research/papers/Implementing_And_Detecting_A_PCI_Rootkit.pdf.

[18] INGUVA, S., RESCORLA, E., SHACHAM, H., AND WALLACH, D. Source code review of the Hart InterCivic voting system. Part of [10], Aug. 2007.

[19] ISO JTC 1/SC 31 COMMITTEE. ISO/IEC 16022:2006: Automatic identification and data capture techniques. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=44230.

[20] JEFFERSON, D. New concerns about electronic voting: What VVVPAT cannot fix, Apr. 2004. Personal communication.

[21] JEFFERSON, D., GINNOLD, E., MIDSTOKKE, K., ALEXANDER, K., STARK, P., AND LEHMKUHL, A. Evaluation of Audit Sampling Models and Options for Strengthening California's Manual Count, July 2007. Report of the Post-Election Audit Standards Working Group, http://www.sos.ca.gov/elections/peas/final_peaswg_report.pdf.

[22] JOHNSTON, R. G. Tamper-indicating seals. *American Scientist*, 94 (Nov.–Dec. 2006), 515–523. Reprint online: http://ephemer.al.cl.cam.ac.uk/~rja14/johnson/newpapers/AmericanScientist(2006).pdf.

[23] MCCARTHY, J., STANISLEVIC, H., LINDEMAN, M., ASH, A., ADDONA, V., AND BATCHER, M. Percentage-based versus SAFE Vote Tabulation Auditing: A Graphic Comparison, Nov. 2007. http://www.verifiedvoting.org/downloads/SAFE-Auditing-Nov-2-Final4.pdf.

[24] MICROSOFT. Buffer overrun in JPEG processing (GDI+) could allow code execution (833987). Microsoft Security Bulletin MS04-028, Dec. 2004.

[25] NORDEN, L., BURSTEIN, A., HALL, J. L., AND CHEN, M. Post-Election Audits: Restoring Trust in Elections, Aug. 2007. http://www.brennancenter.org/page/-/d/download_file_50227.pdf.

[26] POPOVENIUC, S., AND STANTON, J. Undervote and Pattern Voting: Vulnerability and a mitigation technique. In *Workshop on Trustworthy Elections* (June 2007). http://research.microsoft.com/conferences/WOTE2007/papers/10.pdf.

[27] RIX. Writing ia32 alphanumeric shellcodes. *Phrack Magazine 57*, 15 (Dec. 2001). http://www.phrack.org/archives/57/p57-0x18.

[28] STARK, P. B. Conservative Statistical Post Election Audits. *The Annals of Applied Statistics* (2008). In press.

[29] THE METASPLOIT PROJECT. Shellcode archive. http://www.metasploit.com/shellcode/.

[30] VMWARE. VMware Security Advisory VMSA-2008-0005.1. http://www.vmware.com/security/advisories/VMSA-2008-0005.html, Mar. 2008.