A Critique of the 2002 FEC VSPT E-Voting Standards[*]

Mark Gondree Patrick Wheeler Dimitri DeFigueiredo
Department of Computer Science
University of California at Davis
{gondree,wheelerp,defigued}@cs.ucdavis.edu

September 20, 2005

1 Introduction

Throughout the world, electronic voting machines are recording and tallying votes during elections with increasing regularity. In the United States, most election jurisdictions use systems that conform to the 2002 standards promulgated by the Federal Election Commission (FEC) [5]. Although the federal government does not require conformance, most states have passed laws that do.

The goal of the standards is to "address what a voting system should reliably do, not how system components should be configured to meet these requirement" [5, Vol I §1.1]. The standards present a certification procedure involving testing by an Independent Testing Authority (ITA) and many jurisdictions cannot use systems that are uncertified. Virtually all vendors have this testing done. Nevertheless, there have been many instances of certified election systems being "broken" or performing unreliably. Thus the question arises: if systems that meet the standards can be induced to provide inaccurate or unreliable results in an election, is the problem that the standards are inadequate or is the problem that the testing is inadequate?

To answer this question, we throughly explore how the 2002 FEC standards address a variety of security concerns which have been raised by researchers while analyzing certified voting systems. These concerns include risks to the most basic assets of a voting system: threats to voter anonymity, to system integrity and availability, and to the integrity of the final vote tally.

2 System Integrity During Build and Deployment

There is much concern that, even with good standards in place, little requires vendors to deliver a system that is manufactured according to practices and using the materials that have met approval under the standards' review process. The SAIC report, for one, expresses these sentiments [12, §2.2.1]. Specifically, the standard does not require procedural mechanisms to enforce the following requirement:

[*] This paper was compiled to accompany "Toward Clarifying Election Systems Standards" (UC Davis Tech. Report CSE-2005-21), which provides background for voting systems research, updated criticisms relevant to the 2005 standards, and recommendations to ameliorate many of the issues outlined in this report.

[5, Vol I §9.3 b]: The software submitted for qualification testing shall be the exact software that will be used in production units;

The specific issue of software integrity *is* addressed in *Software and Firmware Installation* [5, Vol I §6.4.1]. This section, however, does not require any demonstration or proof that the installed software is the same as that approved by the ITA. Similarly, the standards do not comment on maintaining the integrity of the software or system throughout its lifetime. The standards do require that each ROM be validated before the run of each election [5, Vol I §6.4.1 a] but there is no requirement that the integrity be maintained throughout the election. As a symptom of this deficiency, the RABA Red Team uncovered flaws in the Diebold system which allowed an attacker to potentially change the DRE's software at any point in its lifetime [11, p. 19].

The standards could require the vendor to specify a procedure to verify the integrity of the software and system, at least throughout the manufacturing process. This procedure should be analyzed by the ITA for correctness. Such a procedure might be made part of the *Witness of System Build and Installation* [5, Vol II §9.6.2.4]. Relevant data (i.e. hardware serial numbers, cryptographic checksums, signed certificates) could be provided to the election officials, so they can be assured that the system received is the same as the system inspected by the ITA (e.g. the installed software is unmodified since the witnessed build). More generally, there should be a way to verify that the *complete* system from the witnessed build is the same as the system being used, at any point in its life cycle.

Evidence of the fact that the standards require insufficient procedures for assuring system (specifically, software) integrity is apparent from various elections. In 2002 at El Paso County, Colorado, uncertified DRE software was used in a primary election [7, p. 192]. In 2004, software on a DRE was modified using a patch *during* Indiana's primary election in LaPorte County [10]. Also in 2004, in Marion County, Indiana, ES&S installed uncertified software for use during the election, without election officials' knowledge, and later replaced it with certified software [4, 9, 3]; this became known when an ES&S employee blew the whistle on her employer.

3 Poor Auditing Policies

The standards' operational requirements for auditing are specified in *System Audit* [5, Vol I §2.2.5]. If a report is produced, the audit then falls subject to *Producing Reports* [5, Vol I §2.5.3], *Data Retention* [5, Vol I §2.2.11], and *Data and Document Retention* [5, Vol I §4.3]. Furthermore, audits are required [5, Vol I §2.2.6 i, §2.2.11, §2.5.3.1 f] to capture "the data indicated in Section 4.5," although these sections probably should reference *Audit Record Data* [5, Vol I §4.4], since *Vote Secrecy* [5, Vol I §4.5] applies only to DRE systems and gives no audit requirements other than to retain the voter's selections for vote counting purposes.

[5, Vol I §2.2.5.1]: Because the actual implementation of specific characteristics may vary from system to system, it is the responsibility of the vendor to describe each system's characteristics in sufficient detail that ITAs and system users can evaluate the adequacy of the system's audit trail. This description shall be incorporated in the System Operating Manual, which is part of the Technical Data Package (TDP).

The auditing abilities must be described "in sufficient detail" to "evaluate the adequacy of the audit trail produced." Such a description, however, in no way sounds like material suitable for an operating manual, whose focus is on running (not critiquing) an audit system. As a manifestation of this inadequacy, the SAIC report has already commented on the inadequacy of documentation delineating how to configure the audit software and how to review the audits produced [12, §2.1.5].

The “System Operating Manual,” in which an audit capability should be described by the vendor, is referenced only once more [5, Vol II §A.1]. “System Operations Manual,” however, is referenced twice [5, Vol II §1.4 i, §2.8.7]. The fact that “System Operations Manual” [5, Vol II §2.8.7] appears as a subarticle of *System Operations Procedures* [5, Vol II §2.8] implies that it is the same thing as the System Operations Procedures documentation. If the “Systems Operations Procedures” documentation, “Systems Operating Manual” and “Systems Operations Manual” are indeed the same, then the audit description requirement should be covered by the following:

> [5, Vol II §2.8.5]: The vendor shall provide documentation of system operating procedures that meets the following requirements: . . . [deletia] . . .
>
> f. Provide administrative procedures and off-line operator duties (if any) if they relate to the initiation or termination of system operations, to the assessment of system status, or to the development of an audit trail.

In short, no section of the standards requires the vendor to document how to configure the audit system, when to invoke that audit system to develop an audit trail, how to protect that audit trail from being compromised, or how to use an audit trail to confirm the correct execution of the system or discover and fix errors. There is only the requirement that the vendor provide documentation showing the system can produce an adequate audit trail and explaining how to produce it. For example, in Florida’s Miami-Dade County in 2004, officials could produce a sufficient audit trail, but it overwhelmed the servers when passed to the central tabulation machines [6, p. 127]. It is not surprising that independent investigations have, among their discoveries, found databases where auditing was not configured properly [11, p. 21] and systems whose audit trail could be viewed and edited by unauthorized personnel, without leaving evidence [2, p. 47 §3.4].

4 Threats to Voter Anonymity

The standards require that a system “protect the secrecy of the vote throughout the voting process” [5, Vol I §2.4.3.3 q]. This is clearly undesirable. The content of the vote must become known in order to count the votes. What the system should protect is the *anonymity of the voter*. At times the secrecy of the vote *is* important, but only in so far as it protects the anonymity of the voter.

Voter anonymity, however, is potentially infringed upon by several of the standards’ requirements. For example, one system audit requirement demands the capability to reconstruct the exact timing and sequence of votes [5, Vol I §2.2.5.2.1]. By observing voters access the system, specific votes can be correlated to a specific voter by observing the audit trail and comparing with the observed sequence of voters. It is unclear why this potentially dangerous data its needed to perform a system audit. Similarly, voters must be able to cast a ballot in the face of exceptional scenarios, such as a power failure [5, Vol I §2.4.3.1] while, simultaneously, the system is required to log errors such as these in a permanent fashion for auditing [5, Vol I §2.2.4.1 i, §2.2.5.2.2]. If a voter is known to have voted, say, during a power failure, the content of their ballot can be determined from the audit trail (using time-stamped ballots and power-failure event logs). A voter might even prove to an outside entity how she voted for the purposes of selling her vote by unplugging the machine while casting.

5 Poor Access Control Policies

Access control is addressed by the standards in *Access Control* [5, Vol I §6.2] and *Telecommunications and Data Transmission — Access Control* [5, Vol I §6.5.1]. The functional specifications in *Access Control* require the vendor to recommended policies and describe the mechanisms used to enforce these policies. Specifically, the vendor's recommended policy and mechanisms should "permit authorized access to the system," "prevent unauthorized access," and "provide effective voting system security." Additionally, the vendor must provide a list associating all individuals with the functions and data to which they are granted access, including how and when they are granted access [5, Vol I §6.2.1.2]. As part of the "System Security Specification" documentation requirements, a vendor must "specify the features and capabilities" of its recommended access control policy and "provide a detailed description" of the access control measures and procedures used to enforce the recommended policy [5, Vol II §2.6.1, §2.6.2, §2.6.5, §2.6.6].

Upon review of these sections, it is unclear how the standards have defined an "Access Control Policy" or "Access Control Measures" — neither technical term appears in the provided glossary. One section [5, Vol II §2.6.1] implies that an Access Control Policy consists of "features" and "capabilities." A different section [5, Vol I §6.2] implies that an Access Control Policy is the system's access control matrix's initial state. Briefly, an access control matrix is a table which associates each individual with a list of rights [1, Ch. 2]. As an analogy, if the right is access to a locked room, the standards require the vendor to create a list of everyone who has a copy of the key to the lock. But the standard does *not* require that vendors disclose how the access control matrix can be modified, under what conditions it can be modified, and who can modify it. To continue our analogy, the standards do not require the vendor to describe how to handle duplication of keys or changing locks, or who can do those things, or when those things can be done.

The standards also do not require the vendor to train personnel in the administration and continued maintenance of the system's access controls. In fact, system administration training is only required if a network is used [5, Vol II §2.10.2 d]. The SAIC report's objection to the inadequacy of documentation pertaining to the process of maintaining access controls provided with the Diebold system [12, §2.1.9] is symptomatic of the inadequacy of this aspect of the standards.

Further, the standards choose language to avoid constraining the vendor's choice in access control policy and enforcement mechanisms. This potentially allows vendors to use practices that are considered poor. For example, one section [5, Vol I §6.2.1 g] requires vendors to "provide a description of recommended policies for . . . segregation of duties," while it would not have been much harder to require that the vendor's policy adhere to the principle of separation of duty wherever applicable. This best practice is clearly known to the standards' authors, but its incorporation in a security policy is left to the vendor's judgement.

Independent analyses by researchers have exposed the results of deferring to the vendor's judgement. For instance, various types of mischief are prevented by a lock on the bay on the Accuvote-TS terminal. Each terminal comes with two keys to the lock. Additionally, the lock is the same on all Accuvote-TS terminals [11, p. 18]. So, the access control policy for any precinct implicitly gives access to that lock to anyone with a key, including poll workers in a different precinct or a different state, technicians who work for the manufacturer or the testing authority, and individuals who had access at a tradeshow. As each machine had two keys, and approximately 16,000 machines were purchased by Maryland alone, a large number of individuals may have either retained a key or obtained a copy of a key. Similarly, hard-coded passwords were used to control supervisor functions in systems made by Diebold [11, p. 16] and ES&S [2, p. 96 §1.31]. Again, the implicit access control policy effectively gave supervisor access to anyone who had ever seen these passwords, including officials from past elections, poll workers in other precincts, the manufacturer's programmers, and anyone who has read the passwords online or in print (for example, in [8, §4.4] or [2, p. 57 §1.21(b)]). These policies, implicit in the vendor's design and that are not constrained by the standards, do not adhere to the best practices of systems for which security is a concern.

6 Threats to System Availability

One of the primary assets of an election system is that it is able to operate on election day. In the standards, *Availability* [5, Vol I §3.4.5] and *Reliability* [5, Vol I §3.4.3] define availability as a ratio using the Mean Time Between Failures (MTBF) during *typical system operations* and the Mean Time to Repair, i.e. the average time to perform a *corrective maintenance task* (CMT). For certification, the system must demonstrate at least 99% availability [5, Vol I §3.4.5] during (at least) 163 hours of performance testing [5, Vol II §4.7.3]. Since availability is bound, the time for a CMT is directly related to the MTBFs demonstrated in a laboratory performance-testing environment. In fact, it is unclear if the time for a CMT is also demonstrated in a laboratory scenario or if a vendor must simply argue a CMT takes less time than the bound determined by the laboratory-demonstrated MTBF.

This is obviously problematic: CMTs and MTBFs should both be demonstrated though system use in a deployment scenario, not laboratory performance testing. For example, accidental or malicious actions might cause a machine to require maintenance more often than demonstrated during the Availability testing. These problems might also require a CMT which takes longer than those CMTs demonstrated during testing. For instance, the RABA Red Team was able to disable a machine simply by repeatedly inserting a used voter card or by pulling on exposed monitor wires [11, p. 19]. A more complicated action allowed the investigators to cause the voting terminal to reject all voter and supervisor access cards, causing it to be unusable in an election until the passwords are reset [11, p. 19]. Attacks on availability are called denial of service (DoS) attacks. DoS attacks, however, are considered in the standard once, only with respect to an attack on a telecommunications network [5, Vol I §2.6.5 b 4]. It is very important to consider the effects of actions (intentional or unintentional) that a voter may take which may effect quality of service.

7 Hardware and Software Configurations with Unnecessary Vulnerabilities

The standards have no requirements to disallow systems from featuring unnecessary hardware, unnecessary software, or software that has known vulnerabilities. As a result, systems that are unnecessarily vulnerable to threats have been certified.

For example, RABA's investigators discovered remnants of test software on a production voting terminal that would allow an attacker to overwrite results on the terminal [11, p. 18]. Similarly, an unnecessary USB port accessible via an improperly secured back panel of the GEMS server opened an avenue for potential mischief [11, p. 22].

Requirements for systems that have network access are qualified as requirements for systems that "*use* a public telecommunications network" (emphasis ours) [5, Vol I §5.2.6, 6.5.4, 6.4, 9.4.1.4][5, Vol II §6.6]. These requirements, therefore, do not apply to systems that are connected to, but make no use of, a network. The GEMS server of the Diebold system analyzed by the SAIC report was unnecessarily connected to an administrative network, itself connected to the internet [12, §2.2.2]. A system that happens to be connected (directly or indirectly) to the internet is potentially vulnerable to all of the same threats as a system that uses the internet. The specific language of the standards, however, not only allows a system to be unnecessarily connected to the internet, but excludes it from fulfilling those requirements designed to pertain to systems connected to a network.

Lastly, the standards do not delineate a policy for handling known threats to software. For those systems which use a public network, the vendor must provide documentation to address "newly discovered external threats" [5, Vol II §6.4.2]. This, however, fails to address *old* threats, such as those for which patches are available. Potentially addressing this issue, the vendor must provide documentation to "resolve internally identified defects for items regardless of their origin" [5, Vol I §8.5 c]. It is unclear if this requirement applies to those defects not identified specifically by the vendor. In both cases, the vendor's policy delineated in the documentation is entirely unrestricted by the standards. As a symptom of these confusing guidelines, which offer no standard policy for patching, the RABA investigators discovered there were more than a dozen Microsoft patches which had not been applied to the central tabulation server; one patch would have fixed a vulnerability which the investigators exploited to gain remote access on the server with administrative privileges [11, p. 20].

8 Threats to Data Integrity during Transmission

The standard addresses data transmission in *Data Transmissions* [5, Vol I §5.1.3] and *Telecommunications and Data Transmission* [5, Vol I §6.5]. The following subsection tries to ensure vote-data integrity during transmission:

> [5, Vol I §6.5.2]: Voting systems that use electrical or optical transmission of data shall ensure the receipt of valid vote records is verified at the receiving stations.

While this is a necessary condition, it is not sufficient. If *mutual* authentication is not used then it is possible for someone to intercept the transmitted data before it reaches its destination, alter it, and then forward it to the central election management system (a "man-in-the-middle" attack). The system inspected by the RABA Red Team passed certification because the GEMS servers did verify the authenticity of the vote records. The DRE machines, however, did not verify the authenticity of the GEMS server, so the security of the system was compromised by a man-in-the-middle attack using only a laptop computer [11, p. 21].

The sections *Data Interception Prevention* [5, Vol I §6.5.3] and *Data Interception and Disruption* [5, Vol II §6.4.2] may have been relevant in prohibiting a man-in-the-middle attack as described above, since the attacker must intercept the transmitted signal and relay a modified signal to the target. *Data Interception Prevention*, however, only requires use of some encryption standard and appropriate measures to detect intrusive devices/processes. In a previous draft of these standards, this section required the use of AES specifically and appropriate measures to detect physical taps (electromagnetically-coupled pickups, wiretaps) that could leak data to an unauthorized recipient. It is odd that physical taps are only considered in the case of systems that use telecommunications; it is more odd that the current version of the standards do not consider such threats at all. *Data Interception and Disruption* requires that the ITA review and use judgement in deciding the acceptability of the manufacturer's documented solutions to handling new external threats to the system's use of a telecommunications network. Since man-in-the-middle attacks do not require intrusive actions that could be caught by an intrusion detection system, do not require physical tapping, and are not a "new threat," this attack is not prevented by any version of these requirements nor do they fall under the responsibility of the

ITA's review.

9 Vague Language

Requirements that can be interpreted in multiple ways will inevitably be interpreted differently by each ITA. As a result, systems certified using one ITA might meet a set of requirements that is entirely unlike those requirements met by a system using a different ITA. The result being that the certification process provides no assurance that one particular set of requirements is met. As standards should provide assurance that a particular set of requirements has been met, this is entirely unsatisfactory.

For example, the standards prohibit the voter from accessing "information on the display screen that has not been authorized by election officials and preprogrammed into the voting system" [5, Vol I §2.4.3.3]. It is unclear if changing the *order* of the authorized content is an exploit which is still allowed under these provisions. The RABA investigators were able to manipulate a systems' ballot definition file so a vote for candidate *A* would be registered as a vote for candidate *B* [11, p19]. Its unclear if the ballot definition file now contains unauthorized content. Even if entirely new content were introduced to the definition file, its unclear what maliciousness the above requirement precludes. "Preprogrammed" is undefined and, unless given context, ambiguous: as long as content is added to the machine before the screen displays it, it has been preprogrammed. In fact, it would be challenging for the screen to do the opposite. Lastly, it is unclear how content becomes "authorized by election officials." If the system identifies that content has become authorized by an election official because it is accompanied by a password or key, then content is only controlled through key management policies. Since there are no guidelines for key management (in fact, some systems have administrative passwords that can be guessed almost immediately [11, p. 16]), the notion of "authorized" is, from a practical point of view, meaningless.

Relatedly, the standards state that "Voting systems vendors shall . . . Permit the voter to cast a ballot expeditiously, but preclude voter access to all other aspects of the vote-counting processes" [5, Vol I §6.2.1.2 c]. It is unclear if the "vote-counting process" includes loading the definition file or not. The lack of a voting model causes these terms to be vague. If the "vote-counting" process is defined as the process of correctly storing all votes according to the ballot definitions in the machine, then a machine whose ballot definitions can be modified, such as the machine analyzed by the RABA Red Team, may meet this requirement.

10 Conclusion

We find that the 2002 FEC voting standards provide vague security criteria, and rely too often on the discretion of the system's vendor to provide effective, sufficient, and appropriate security policies. The standards acknowledge that "system security is achieved through a combination of technical capabilities and sound administrative practices" [5, Vol I §2.2.1] but insist "the standards are not intended to define appropriate election administrative practices" [5, Vol I §1.11]. Also, the standards give no guidelines with which the ITA is able to judge the sufficiency of the vendor's policies. As a result, the standards fail to provide any assurance that best-practices, or even marginally satisfactory practices, have been met regarding, for example, key management, system audit, software management, or access control policies. Although the standards constrain the vendor in specific implementation choices (e.g. source code line length and function recursion depth), there are no similar constraints for basic design choices (e.g. how parties interact to authenticate themselves and what access points will be available in the system). Returning to the question motivating this study, we find the standards fail to preclude many of the problems witnessed in existent voting systems. We feel these inadequacies of the standards are a major factor leading to insufficient or ad-hoc testing and, subsequently, to the certification of insecure and unsatisfactory voting systems.

References

[1] Matt Bishop. *Computer Security: Art and Science*. Addison-Wesley, 2003.

[2] Compuware Corporation. Direct Recording Electronic (DRE) Technical Security Assessment Report, November 2003. `http://www.sos.state.oh.us/sos/hava/compuware112103.pdf`.

[3] Rick Dawson and Loni Smith McKown. Marion county election board demands answers from ES&S. WISH-TV Indianapolis, Indiana, April 22 2004. `http://www.wishtv.com/Global/story.asp?S=1808590`.

[4] Rick Dawson and Loni Smith McKown. Voting machine company takes heat over illegal software. WISH-TV Indianapolis, Indiana, March 11 2004. `http://www.wishtv.com/Global/story.asp?S=1704709`.

[5] Federal Election Commission. Voting systems performance and test standards, 2002.

[6] John H. Fund. *Stealing Elections*. Encounter Books, 2004.

[7] Bev Harris. *Black Box Voting*. Plan Nine Publishing, 2004.

[8] Tadayoshi Kohno, Adam Stubblefield, Aviel D. Rubin, and Dan S. Wallach. Analysis of an electronic voting system. In *Proceedings of the 2004*, pages 27–40. IEEE Computer Society, May 2004. Appeared previously as Johns Hopkins University Information Security Institute Technical Report TR-2003-19, July 23, 2003.

[9] Mary McDermott and Loni Smith McKown. Marion county clerk accuses ES&S of lying. WISH-TV Indianapolis, Indiana, April 20 2004. `http://www.wishtv.com/Global/story.asp?S=1799902`.

[10] Kristin Miller. Computer glitch still baffles county clerk. Michigan City News Dispatch Online, November 2004. `http://www.michigancityin.com/articles/2004/11/04/news/news02.txt`.

[11] RABA Innovative Solution Cell. Trusted Agent Report: Diebold AccuVote-TS Voting System, January 2004. `http://www.raba.com/press/TA_Report_AccuVote.pdf`.

[12] Science Applications International Corporation. Risk Assessment Report: Diebold AccuVote-TS Voting System and Processes, September 2003. `http://www.dbm.maryland.gov/SBE`.