APPENDIX B: SECURITY STATEMENTS FROM THE RUBIN REPORT & STATE OF MARYLAND CONTROLS

The following table is a brief analysis of statements made by Professor Rubin, et al, in their report on the Diebold source code entitled "Analysis of an Electronic Voting System", July 23, 2003. In general, SAIC made many of the same observations, *when considering only the source code.* While many of the statements made by Mr. Rubin were technically correct, it is clear that Mr. Rubin did not have a complete understanding of the State of Maryland's implementation of the AccuVote-TS voting system, and the election process controls or environment. During this assessment, SAIC had access to system and election documentation, personnel and equipment. Applying the NIST Risk Assessment methodology to the evaluation of the equipment in its operational environment and the totality of the management, operational, and technical controls, SAIC reached many different conclusions. Indeed, Professor Rubin states repeatedly in his paper that he does not know how the system operates in an election and he further identifies the assumptions that he used to reach his conclusions. In those cases where these assumptions concerning operational or management controls were incorrect, the resultant conclusions were, unsurprisingly, also incorrect.

Page #	Statement from Rubin Report	State of Maryland Controls
2	*"The anonymity of a voter's ballot must be preserved, both to guarantee the voter's safety when voting against a malevolent candidate, and to guarantee that voters have no evidence that proves which candidates received their votes."*	The anonymity of a voter's ballot is preserved because the AccuVote-TS voting system does not use or store personal information and does not provide an individual paper record for each voter, therefore leaving no evidence of a single voter's selections. **[Redacted]**
2	*"The voting system must also be tamper-resistant to thwart a wide range of attacks, including ballot stuffing by voters and incorrect tallying by insiders."*	The AccuVote-TS voting system only allows a voter to cast their vote one time. After the individual votes, the Voter Access Card is deactivated. In addition, there are physical, and procedural controls at the polling stations to ensure that voters are only given access to the DRE one time and to make sure that they do not vote multiple times. In addition, when the vote is cast by the voter, the Voter Access Card automatically ejects making a loud noise and the DRE is

Page #	Statement from Rubin Report	State of Maryland Controls
		disabled until another valid Voter Access Card is inserted.
2	*"A voting system must be comprehensible and usable by the entire voting population, regardless of age, infirmity, or disability."*	This is not a security requirement.
2	*"The only known solution to this problem is to introduce a "voter-verifiable audit trail." [DMNW03]. Most commonly, this is achieved by adding a printer to the voting terminal. When the voter finishes selecting candidates, a ballot is printed on paper and presented to the voter. If the printed ballot reflects the voter's intent, the ballot is saved for future reference. If not, the ballot is mechanically destroyed. Using this "Mercuri method," [Mer00] the tally of the paper ballots takes precedence over any electronic tallies. As a result, the correctness of the voting terminal software no longer matters; either a voting terminal prints correct ballots or it is taken out of service."*	The AccuVote-TS voting system requires that the voter verify their selections prior to the actual casting of the vote. This is done via a review screen on the DRE. The AccuVote-TS voting system does not provide a paper "voter-verifiable audit trail" specific to individual voters. Note: A printed paper ballot would still be subject to fraud. A compromised machine could be programmed to record votes incorrectly, but provide a correct paper ballot to the voter. Only in the event of a total recount would this be discovered. Additionally, the process of hand counting the millions of votes is time consuming and is prone to error.
4	*"Most notably, voters can easily program their own smartcards to simulate the behavior of valid smartcards used in the election."*	Although it is possible for someone to buy and to program their own smartcard, the attacker would be limited to changing their party affiliation in the case of a primary (i.e., they could see a ballot meant for another party) because the smartcard only contains party affiliation and access to vote on the DRE. The combination of logic controls in the DRE software, the physical controls and the openness of the voting booths minimize the likelihood of the voter being able to cast multiple votes without being detected.
4	*"With such homebrew cards, a voter can cast multiple ballots without leaving any trace."*	Although it is possible for someone to buy and to program their own smartcard, the attacker would be limited to changing their party affiliation in the case of a primary (i.e., they could see a ballot meant for another party) because the smartcard only contains party affiliation and access to vote on the DRE. The combination of logic controls in the DRE software, the physical controls and the openness of the voting booths minimize the likelihood of the voter being able

Page #	Statement from Rubin Report	State of Maryland Controls
		to cast multiple votes without being detected.
4	*"A voter can also perform actions that normally require administrative privileges, including viewing partial results and terminating the election early."*	A voter would need to manufacture a smartcard with administrator rights to obtain these privileges. Assuming someone could manufacture the card and obtained access to the DRE, the specific DRE device could be disabled (i.e., close election). Such an attack would be detected due to the physical controls and the openness of the voting booths. The Disaster Recovery and Incident Management Plan guide provides procedures for handling a disabled DRE.
4	*"Similar undesirable modifications could be made by malevolent poll workers (or even maintenance staff) with access to the voting terminals before the start of an election."*	The physical controls prevent any single individual from having access to the DRE devices prior to the election. The DRE devices are tested at the LBE warehouse, then sealed with tamper-proof tape prior to shipment to the polling site. The Election Judges remove the tamper-proof tape the morning of the election.
4	*"Furthermore, the protocols used when the voting terminals communicate with their home base, both to fetch election configuration information and to report final election results, do not use cryptographic techniques to authenticate the remote end of the connection nor do they check the integrity of the data in transit."*	The AccuVote-TS voting system is not using a modem to fetch election information. The results of the election however are transmitted. These transmissions are not encrypted. SAIC has recommended that these transmissions be encrypted and that a 100% verification of the transmissions and the PCMCIA cards occur.
4	*"Given that these voting terminals could communicate over insecure phone lines or even wireless Internet connections, even unsophisticated attackers can perform untraceable "man-in-the-middle" attacks."*	The DRE devices are not connected to a network. The DRE Accumulator is connected via modem after the election to transmit vote totals to the LBE. These transmissions are not encrypted and could be intercepted or modified. SAIC has recommended that these transmissions be encrypted and that a 100% verification of the transmissions and the PCMCIA cards occur.
4	*"Cryptography, when used at all, is used incorrectly."*	Currently, **[Redacted]** encryption is only used for the resident memory on the DRE in accordance with Federal requirements. Once the DRE is powered down, the resident memory is erased. SAIC has recommended that encryption

Page #	Statement from Rubin Report	State of Maryland Controls
		be employed for the modem transmission of the vote totals.
4	*"In many places where cryptography would seem obvious and necessary, none is used."*	Currently, **[Redacted]** encryption is only used for the resident memory on the DRE. Once the DRE is powered down, the memory is erased. SAIC has recommended that encryption be employed for the modem transmission of the vote totals.
4	*"More generally, we see no evidence of rigorous software engineering discipline. Comments in the code and the revision change logs indicate the engineers were aware of areas in the system that needed improvement, though these comments only address specific problems with the code and not with the design itself."*	The scope of the risk assessment did not include a review of Diebold's software engineering practices. SAIC's review of the source code also noted similar comments. It should be noted that since the publication of the Rubin report, Diebold has developed, documented, and implemented a change control process, which has been delivered to the SBE.
4	*"We also saw no evidence of any change control process that might restrict a developer's ability to insert arbitrary patches to the code."*	The scope of the risk assessment did not include a review of Diebold's software engineering practices. It should be noted that since the publication of the Rubin report, Diebold has developed, documented, and implemented a change control process, which has been delivered to the SBE. SBE and LBE's Logic & Accuracy tests verify that votes are recorded accurately prior to the use of the DRE for any election. SAIC has also recommended that SBE enhance the controls for certifying that the implemented source code is the same version as that certified by the ITA, and to expand their testing to include testing for time-oriented exploits (e.g., trojans). This may be accomplished by changing the machine date and time to correspond to that of the election during testing.
4	*"Absent such processes, a malevolent developer could easily make changes to the code that would create vulnerabilities to be later exploited on Election Day."*	The scope of the risk assessment did not include a review of Diebold's software engineering practices. It should be noted that since the publication of the Rubin report, Diebold has developed, documented, and implemented a change control process, which has been delivered to the SBE.

Page #	Statement from Rubin Report	State of Maryland Controls
		SBE and LBE's Logic & Accuracy tests verify that votes are recorded accurately prior to the use of the DRE for any election. We have also recommended that SBE enhance the controls for certifying that the implemented source code is the same version as that certified by the ITA and to expand their testing to include testing for time-oriented exploits (e.g., Trojans). This may be accomplished by changing the machine date and time to correspond to that of the election during testing.
4	*"We also note that the software is written entirely in C++. When programming in an unsafe language like C++, programmers must exercise tight discipline to prevent their programs from being vulnerable to buffer overflow attacks and other weaknesses."*	The scope of the risk assessment did not include a review of Diebold's software engineering practices or an evaluation of which software language may be more secure. Our review did note vulnerabilities that point to software inconsistencies and problems. SBE and LBE's Logic & Accuracy tests verify that votes are recorded accurately prior to the use of the DRE for any election. We have also recommended that SBE enhance the controls for certifying that the implemented source code is the same version as that certified by the ITA and to expand their testing to include testing for time-oriented exploits (e.g., trojans). This may be accomplished by changing the machine date and time to correspond to that of the election during testing.
4	*"Indeed, buffer overflows caused real problems for AccuVote-TS systems in real elections."* (Note: This reference has nothing to do with buffer overflows)	It is true that this system is not configured to defend against buffer overflow attacks. As the DRE has no network connections, an attacker is not provided a means to exploit this vulnerability.
4	*"Although the Diebold code is designed to run on a DRE device (an example of which is shown in Figure 1), one can run it on a regular Microsoft Windows computer (during our experiments we compiled and ran the code on a Windows 2000 PC)."*	This is not a security requirement.

Page #	Statement from Rubin Report	State of Maryland Controls
4	*"In the following we describe the process for setting up and running an election using the Diebold system. Although we know exactly how the code works from our analysis, we must still make some assumptions about the external processes at election sites. In all such cases, our assumptions are based on the way the Diebold code works, and we believe that our assumptions are reasonable. There may, however, be additional administrative procedures in place that are not indicated by the source code."*	This is not a security requirement, but it does give insight into the methodology used by the Rubin team in the drafting the report.
5	*"In common usage, we believe the voting terminals will be distributed without a ballot definition pre-installed."*	This assumption is invalid. The voting terminals are distributed with the state approved ballot information loaded.
5	*"We do not know exactly how the voter gets his voter card. It could be sent in the mail with information about where to vote, or it could be given out at the voting site on the day of the election. To understand the voting software itself, however, we do not need to know what process is used to distribute the cards to voters."*	This assumption is invalid. The Voter Access Cards are distributed at the polling site after the voter is vetted, and retrieved from the voter after the voter has cast their vote.
5	*"As we have only analyzed the code for the Diebold voting terminal, we do not know exactly how the back-end server tabulates the final results it gathers from the individual terminals. Obviously, it collects all the votes from the various voting terminals. We are unable to verify that there are checks to ensure, for example, that there are no more votes collected than people who are registered at or have entered any given polling location."*	SBE and LBEs have numerous checks and balances to ensure that the votes entered on the DRE devices are accurately reported. There are checks at the polling site, the LBE HQ and SBE. SAIC has recommended that the checks and balances be augmented to include a 100% verification of the vote transmissions to the PCMCIA cards.
9	*"Upon reviewing the Diebold code, we observed that the smartcards do not perform any cryptographic operations."*	That is correct, the smartcards perform no cryptographic functions. The smartcards also do not contain any sensitive or personal information. The smartcards contain party affiliation (in the case of a primary election) and access to vote on the DRE.

Page #	Statement from Rubin Report	State of Maryland Controls
9	*"For example, authentication of the terminal to the smartcard is done "the old-fashioned way:" the terminal sends a clear text (i.e., unencrypted) 8-byte password to the card and, if the password is correct, the card believes that it is talking to a legitimate voting terminal. Unfortunately, this method of authentication is insecure: an attacker can easily learn the 8-byte password used to authenticate the terminal to the card (see Section 3.3), and thereby communicate with a legitimate smartcard using his own smartcard reader."*	The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to attach a card reader to the voting terminal would be easily visible to any of the many election officials. In addition, the vetting process limits access to DRE devices to eligible voters.
9	*"Furthermore, there is no authentication of the smartcard to the device. This means that nothing prevents an attacker from using his own homebrew smartcard in a voting terminal."*	Although it is possible for someone to buy and to program their own smartcard, the attacker would be limited to changing their party affiliation in the case of a primary (i.e., they could see a ballot meant for another party) because the smartcard only contains party affiliation and access to vote on the DRE. The combination of logic controls in the DRE software, the physical controls and the openness of the voting booths minimize the likelihood of the voter being able to cast multiple votes without being detected.
9	*"An attacker who knows the protocol spoken by the voting terminal to the legitimate smartcard could easily implement a homebrew card that speaks the same protocol."*	Although it is possible for someone to buy and to program their own smartcard, the attacker would be limited to changing their party affiliation in the case of a primary (i.e., they could see a ballot meant for another party) because the smartcard only contains party affiliation and access to vote on the DRE. The combination of logic controls in the DRE software, the physical controls and the openness of the voting booths minimize the likelihood of the voter being able to cast multiple votes without being detected.
9	*"Even if the attacker does not a priori know the protocol, an attacker could easily learn enough about the protocol to create new voter cards by attaching a "wiretap" device between the voting terminal and a legitimate smartcard and observing the communicated messages."*	The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to attach a card reader to the voting terminal would be easily visible to any of the many election officials. In addition, the vetting process limits access to DRE devices to eligible voters.

Page #	Statement from Rubin Report	State of Maryland Controls
		limits access to DRE devices to eligible voters.
9	*"The parts for building such a device are readily available and, given the privacy of voting booths, might be unlikely to be noticed by poll workers. An attacker might not even need to use a wiretap to see the protocol in use."*	The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to attach a card reader to the voting terminal would be easily visible to any of the many election officials.
9	*"Likewise, the important data on the legitimate voting card is stored as a file (named 0x3D40 — smartcard files have numbers instead of textual file name) that can be easily read by a portable smartcard reader. Again, given the privacy of voting booths, an attacker using such a card reader would be unlikely to be noticed. Given the ease with which an attacker can interact with legitimate smartcards, plus the weak password-based authentication scheme (see Section 3.3), an attacker could quickly gain enough insight to create homebrew voting cards, perhaps quickly enough to be able to use such homebrew cards during the same election day."*	The privacy of the voting booth is limited. If one pictures the old, curtained voting booths of the past, this could be possible. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to attach a card reader to the voting terminal would be easily visible to any of the many election officials.
9	*"The only impediment to the mass production of homebrew smartcards is that each voting terminal will make sure that the smartcard has encoded in it the correct m_ElectionKey, m_VCenter, and m_DLVersion (see DoVote() in BallotStation/Vote.cpp). The m_ElectionKey and m_DLVersion are likely the same for all locations and, furthermore, for backward-compatibility purposes it is possible to use a card with m_ElectionKey and m_DLVersion undefined. The m_VCenter value could be learned on a per-location-basis by interacting with legitimate smartcards, from an insider, or from inferences based on the m_VCenter values observed at other polling locations."*	Although it is possible for someone to buy and to program their own smartcard, the attacker would be limited to changing their party affiliation in the case of a primary (i.e., they could see a ballot meant for another party) because the smartcard only contains party affiliation and access to vote on the DRE. The combination of logic controls in the DRE software, the physical controls and the openness of the voting booths minimize the likelihood of the voter being able to cast multiple votes without being detected.
10	*"Since an adversary can make perfectly valid smartcards, the adversary could bring a stack of active cards to the*	The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and

Page #	Statement from Rubin Report	State of Maryland Controls
	voting booth. Doing so gives the adversary the ability to vote multiple times."	the voter's selections. The action of trying to run numerous smartcards through the voting terminal would be easily visible to any of the many election officials. In addition, the voting machine makes a loud noise and ejects the smartcard after each vote is cast.
10	*"More simply, instead of bringing multiple cards to the voting booth, the adversary could program a smartcard to ignore the voting terminal's deactivation command. Such an adversary could use one card to vote multiple times."*	The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to cast multiple votes would be easily visible to any of the many election officials. Additionally, there are procedures to ensure that only the correct number of votes have been cast on each DRE. Each polling site checks the number of Voter Authority Cards signed, to the register, then to the total votes cast on DREs.
10	*"Will the adversary's multiple-votes be detected by the voting system? To answer this question, we must first consider what information is encoded on the voter cards on a per-voter basis. The only per-voter information is a "voter serial number" (m_VoterSN in the CVoterInfo class). Because of the way the Diebold system works, m_VoterSN is only recorded by the voting terminal if the voter decides not to place a vote (as noted in the comments in TSElection/Results.cpp, this field is recorded for uncounted votes for backward compatibility reasons). It is important to note that if a voter decides to cancel his or her vote, the voter will have the opportunity to vote again using that same card (and, after the vote has been cast, m_VoterSN will not be recorded)."*	There are procedures to ensure that only the correct number of votes have been cast on each DRE. Each polling site checks the number of Voter Authority Cards signed, to the register, then to the total votes cast on DREs.
10	*"Can the back-end tabulation system detect multiple-vote casting? If we assume the number of collected votes becomes greater than the number of people who showed up to vote, and if the polling locations keep accurate counts of the number of people who show up to vote, then the back-end system, if designed properly, should be able*	As noted, Mr. Rubin did not look at the backend tabulating system. SBE and LBE have numerous checks and balances to ensure that the votes entered on the DRE devices are accurately reported. There are checks at the polling site, the LBE HQ and SBE. SAIC has recommended that the checks and balances be augmented to include a

Page #	Statement from Rubin Report	State of Maryland Controls
	to detect the existence of counterfeit votes. However, because m_VoterSN is only stored for those who did not vote, there will be no way for the tabulating system to count the true number of voters or distinguish the real votes from the counterfeit votes. This would cast serious doubt on the validity of the election results. We point out, however, that we only analyzed the voting terminal's code; we do not know whether such checks are performed in the actual back-end tabulating system."	100% verification of the vote transmissions to the PCMCIA cards.
10	*"Just as an adversary can manufacture his or her own voter cards, an adversary can manufacture his or her own administrator and ender cards (administrator cards have an easily-circumventable PIN, which we will discuss in Section 3.2). This attack is easiest if the attacker has knowledge of the Diebold code or can interact with a legitimate administrator or ender card."*	Assuming someone could manufacture the card and obtained access to the DRE, the specific DRE device could be disabled (i.e., close election). Such an attack would be detected due to the physical controls and the openness of the voting booths. The Disaster Recovery and Incident Management Plan guide provides procedures for handling a disabled DRE.
10	*"Using a homebrew administrator card, a poll worker, who might not otherwise have access to the administrator functions of the Diebold system but who does have access to the voting machines before and after the elections, could gain access to the administrator controls. If a malicious voter entered an administrator or ender card into the voting device instead of the normal voter card, then the voter would be 10 able to terminate the election and, if the card is an administrator card, gain access to additional administrative controls."*	Assuming someone could manufacture the card and obtained access to the DRE, the specific DRE device could be disabled (i.e., close election). Such an attack would be detected due to the physical controls and the openness of the voting booths. The Disaster Recovery and Incident Management Plan guide provides procedures for handling a disabled DRE.
11	*"The use of administrator or ender cards prior to the completion of the actual election represents an interesting denial-of-service attack. Once "ended," the voting terminal will no longer accept new voters (see CVoteDlg::OnCardIn()) until the terminal is somehow reset. Such an attack, if mounted simultaneously by multiple people, could shut down a polling place. If a polling place is in a precinct considered to favor one*	Assuming someone could manufacture the card and obtained access to the DRE, the specific DRE device could be disabled (i.e., close election). Such an attack would be detected due to the physical controls and the openness of the voting booths. The Disaster Recovery and Incident Management Plan guide provides procedures for handling a disabled DRE.

Page #	Statement from Rubin Report	State of Maryland Controls
	candidate over another, attacking that specific polling place could benefit the less-favored candidate. Even if the poll workers were later able to resurrect the systems, the attack might succeed in deterring a large number of potential voters from voting (e.g., if the attack was performed over the lunch hour). If such an attack was mounted, one might think the attackers would be identified and caught. We note that many governmental entities do not require identification to be presented by a voter, instead allowing for "provisional" ballots to be cast. By the time the poll workers realize that one of their voting terminals has been disabled, the perpetrator may have long-since left the scene."	If as suggested, multiple individuals mounted a simultaneous attack at a polling site, with forged administrator cards, and closed the DRE devices, and we assume that they all successfully got away, the Election Judges still could immediately reopen the DRE devices. The Disaster Recovery and Incident Management Plan guide provides procedures for handling a disabled DRE.
11	*"Upon looking more closely at this administrator authentication process, however, we see that there is a flaw with the way the PINs are verified. When the terminal and the smartcard first begin communicating, the PIN value stored on the card is sent in cleartext from the card to the voting terminal. Then, when the user enters the PIN into the terminal, it is compared with the PIN that the smartcard sent (CPinDlg::OnOK()). If these values are equal, the system accepts the PIN. Herein lies the flaw with this design: any person with a smartcard reader can easily extract the PIN from an administrator card. The adversary doesn't even need to fully understand the protocol between the terminal and the device: if the response from the card is n bytes long, the attacker who correctly guesses that the PIN is sent in the clear would only have to try n¡3 possible PINs, rather than 10,000. This means that the PINs are easily circumventable. Of course, if the adversary knows the protocol between the card and the device, an adversary could just make his own administrator card, using any desired PIN (Section 3.1.2)."*	Assuming someone could manufacture the card and obtained access to the DRE or obtained a valid administrator's card and PIN combinations, the specific DRE device could be disabled (i.e., close election). Such an attack would be detected due to the physical controls and the openness of the voting booths. The Disaster Recovery and Incident Management Plan guide provides procedures for handling a disabled DRE. Additionally the privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections.
12	*"There are several issues with the above code. First, hard-coding passwords in C++ files is generally a poor design*	Hard-coding of passwords is not consistent with best security practice. We have recommended that the hard-

Page #	Statement from Rubin Report	State of Maryland Controls
	choice. We will discuss coding practices in more detail in Section 6, but we summarize some issues here. Hard-coding passwords into C++ files suggests a lack of key and password management."	coded passwords be removed and changed.
12	*"Furthermore, even if the developers assumed that the passwords would be manually changed and the software recompiled on a per-election basis, it would be very easy for someone to forget to change the constants in VoterCard/CLXSmartCard.cpp. (Recompiling on a per-election basis may also be a concern, since good software engineering practices would dictate additional testing and certification if the code were to be recompiled for each election.)"*	This assumption is invalid assumption. The software is not recompiled on a per-election basis. In addition, only source code certified by the ITA is loaded on the devices. SBE and LBE's Logic & Accuracy tests verify that votes are recorded accurately. SAIC has recommended that SBE enhance the controls for certifying that the implemented source code is the same version as that certified by the ITA and to expand their testing to include testing for time-oriented exploits (e.g., trojans). This may be accomplished by changing the machine date and time to correspond to that of the election during testing.
12	*"The above issues would only be a concern if the authentication method were otherwise secure. Unfortunately, it is not. Since the password is sent in the clear from the terminal to the card, an attacker who puts a fake card into the terminal and records the command from the terminal will be able to learn the password (and file name) and then re-use that password with real cards. An adversary with knowledge of this password could then create counterfeit voting cards. As we have already discussed (see Section 3.1.1), this can allow the adversary to cast multiple votes, among other attacks. Hence, the authentication of the voting terminal to the smartcards is insecure."*	The smartcard allows the voter to enter a vote, but the user is authenticated during the vetting process, (i.e., the control over who gets to vote is not controlled by the smartcard, but by the vetting procedures). Once again the privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to cast multiple votes would be easily visible to any of the many election officials. In addition, the voting machine makes a loud noise and ejects the smartcard after each vote is cast.
12	*"Furthermore, note the control flow in the above code-snippet. If the password chosen by the designers of the system ("\xED\x0A\xED\x0A\xED\x0A\xED\x0A") does not work, then CCLXSmartCard::*	The smartcard allows the voter to enter vote, but the user is authenticated during the vetting process, (i.e., the control over who gets to vote is not controlled by the smartcard, but by the vetting procedures). In addition, once again the privacy of the voting booth is limited. The AccuVote voting

Page #	Statement from Rubin Report	State of Maryland Controls
	Open() uses the smartcard manufacturer's default password of "\x00\x01\x02\x03\x04\x05\x06\x07." *One issue with this is that it implies that sometimes the system is used with un-initialized smartcards. This means that an attacker might not even need to figure out the system's password in order to be able to authenticate to the cards."*	booth provides privacy only for the touch screen and the voter's selections. The action of trying to cast multiple votes would be easily visible to any of the many election officials. In addition, the voting machine makes a loud noise and ejects the smartcard after each vote is cast.
12	*"As we noted in Section 3.1, some smartcards allow a user to get a listing of all the files on a card. If the system uses such a card and also uses the manufacturer's default password of \x00\x01\x02\x03\x04\x05\x06\x07, then an attacker, even without any knowledge of the source code and without the ability to intercept the connection between a legitimate card and a voting terminal, but with access to a legitimate voter card, will still be able to learn enough about the smartcards to be able to create counterfeit voter cards."*	The smartcard allows the voter to enter vote, but the user is authenticated during the vetting process, (i.e., the control over who gets to vote is not controlled by the smartcard, but by the vetting procedures). Once again the privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to cast multiple votes would be easily visible to any of the many election officials. In addition, the voting machine makes a loud noise and ejects the smartcard after each vote is cast.
13	*"Unfortunately, under Windows CE, which we believe is used in commercial Diebold voting terminals, the existence of the removable storage device is not enforced properly."*	The PCMCIA cards are locked into the DRE device. The key is controlled by the Chief Judges. Additionally, we have recommended that the State further secure this locked compartment using tamper-proof tape during the actual election
13	*"Unlike other versions of Windows, removable storage cards are mounted as subdirectories under CE. When the voting software wants to know if a storage card is inserted, it simply checks to see if the Storage Card subdirectory exists in the file system's root directory. While this is the default name for a mounted storage device, it is also a perfectly legitimate directory name for a directory in the main storage area. Thus, if such a directory exists, the terminal can be fooled into using the same storage device for all of the data. This would reduce the amount of*	Pre-election Logic and Accuracy testing checks both the main storage area, and the removable memory.

Page #	Statement from Rubin Report	State of Maryland Controls
	redundancy in the voting system and would increase the chances that a hardware fault could cause recorded votes to be lost."	
13	*"The majority of the system configuration information for each terminal is stored in the Windows registry under HKEY_LOCAL_MACHINE\Software\GlobalElectionSystems\AccuVote-TS4 . This includes both identification information such as the terminal's serial number and more traditional configuration information such as the COM port that the smartcard reader is attached to. All of the configuration information is stored in the clear, without any form of integrity protection. Thus, all an adversary must do is modify the system registry to trick a given voting terminal into effectively impersonating any other voting terminal."*	Exploitation of this vulnerability requires access to the system registry. Since the DRE is not connected to a network, an attacker's access to the registry is limited by procedural and physical barriers.
13	*"It is unclear how the tallying authority would deal with results from two different voting terminals with the same voting ID — at the very least human intervention to resolve the conflict would probably be required."*	Prior to each election, the GEMS server assigns a unique number to each PCMCIA card as part of the ballot loading process. When the results are read from the PCMCIA cards at the conclusion of the election, the GEMS server uses this unique number to validate acceptance of the data. If two of these numbers are identical, the election officials would investigate using established procedures.
13	*"The Federal Election Commission draft standard requires each terminal to keep track of the total number of votes that have ever been cast on it — the "Protective Counter." This counter is used to provide yet another method for ensuring that the number of votes cast on each terminal is correct. However, as the following code from Utilities/machine.cpp shows, the counter is simply stored as an integer in the file system.bin in the terminal's system directory (error handling code has been removed for clarity):*	This exploit requires access to the system. Since the system is not connected to a network, physical access is required. The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to connect devices to the system would be easily visible to any of the many election officials. Other physical and procedural controls are effective in preventing access to the system prior to, or after an election.

Page #	Statement from Rubin Report	State of Maryland Controls
	long GetProtectedCounter() *{* *DWORD protectedCounter = 0;* *CString filename = ::GetSysDir();* *filename += _T("system.bin");* *CFile file;* *file.Open(filename, CFile::modeRead | CFile::modeCreate |* *CFile::modeNoTruncate);* *file.Read(&protectedCounter, sizeof(protectedCounter));* *file.Close();* *return protectedCounter;* *}* *By modifying this counter, an adversary could cast doubt on an election by creating a discrepancy between the number of votes cast on a given terminal and the number of votes that are tallied in the election. While the current method of implementing the counter is totally insecure, even a cryptographic checksum would not be enough to protect the counter; an adversary with the ability to modify and view the counter would still be able to roll it back to a previous state. In fact, the only solution that would work would be to implement the protective counter in a tamper-resistant hardware token, requiring modifications to the*	

Page #	Statement from Rubin Report	State of Maryland Controls
	physical voting terminal hardware."	
14	*"The "ballot definition" for each election contains everything from the background color of the screen to the PPP username and password to use when reporting the results. This data is not encrypted or check summed (cryptographically or otherwise) and so can be easily modified by any attacker with physical access to the file."*	As stated, this assumption requires access to the system. Since the system is not connected to a network, physical access is required. The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to connect devices to the DRE would be easily visible to any of the many election officials.
14	*"By simply changing the order of the candidates as they appear in the ballot definition, the results file will change accordingly. However, the candidate information itself is not stored in the results file. The file merely tracks that candidate 1 got so many votes and candidate 2 got so many other votes. If an attacker reordered the candidates on the ballot definition, voters would unwittingly cast their ballots for the wrong candidate. As with denial-of-service attacks (see Section 3.1.2), ballot reordering attacks would be particularly effective in polling locations known to be heavily partisan."*	This exploit requires access to the system. Since the system is not connected to a network, physical access is required. The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to connect devices to the system would be easily visible to any of the many election officials. In addition, the ballot is on the PCMCIA card, which is locked in the DRE device. Note: SBE uses a public FTE site to distribute ballot information. While there are many checks at the LBE of the ballot, SAIC has recommended that SBE implement a secure method to transfer the ballot.
14	*"Even without modifying the ballot definition, an attacker can gain almost enough information to impersonate the voting terminal to the back-end server. The terminal's voting center ID, PPP dial-in number, username, password and the IP address of the back-end server are all available in the clear (these are parsed into a CElectionHeaderItem in TSElection\TSElectionObj.cpp). Assuming an attacker is able to guess or create a voting terminal ID, he would be able to transmit fraudulent vote reports to the backend server by dialing in from his own computer. While both the paper trail and data stored on legitimate terminals could be used to compensate for this attack after the fact, it could,*	The LBE GEMS server (i.e., backend server) is not connected to a network. The LBE GEMS server checks for PCMCIA cards from the modem transmissions. This error checking accounts both for card validity (i.e. that the card was issued and is not a duplicate) and ensures that all issued cards are reported. SAIC has recommended that the modem transmissions be encrypted and that the LBE perform a 100% verification of the vote transmissions to PCMCIA cards.

Page #	Statement from Rubin Report	State of Maryland Controls
	at the very least, delay the election results."	
14	*"(The PPP number, username, password, and IP address of the back-end server are also stored in the registry HKEY_LOCAL_MACHINE\Software\GlobalElectionSystems\AccuVote-TS4\ TransferParams. Since the ballot definition may be transported on portable memory cards or floppy disks, the ballot definition may perhaps be easier to obtain from this distribution media rather than from the voting terminal's internal data storage.)"*	Ballots are public knowledge. After the ballot is created at SBE, the LBE performs the Logic and Accuracy tests to ensure validity and correctness.
14	*"We will return to some of these points in Section 5.1, where we show that modifying and viewing ballot definition files does not always require physical access to the terminals on which they are stored."*	Modification of the ballot requires access to the PCMCIA cards since the DRE devices are not connected to a network.
15	*"Unlike the other data stored on the voting terminal, both the vote records and the audit logs are encrypted and check summed before being written to the storage device. Unfortunately, neither the encrypting nor the check summing is done securely.* *All of the data on a storage device is encrypted using a single, hard-coded DES [NBS77] key:* *#define DESKEY ((des_key*)"F2654hD4")"*	Currently, **[Redacted]** encryption is only used for the resident DRE memory. Once the DRE is powered down, the memory is erased. Note, we have recommended that encryption be employed for the modem transmission of the vote totals. The DRE devices are not connected to a network and physical access would be required to get to the data. The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to connect devices to the system would be easily visible to any of the many election officials.
15	*"Note that this value is not a hex representation of a key. Instead, the bytes in the string "F2654hD4" are fed directly into the DES key scheduler. If the same binary is used on every voting terminal, an attacker with access to the source code, or even to a single binary image, could learn the key, and thus read and modify voting and auditing records."*	Currently, **[Redacted]** encryption is only used for the resident DRE memory. Once the DRE is powered down, the memory is erased. Note, we have recommended that encryption be employed for the modem transmission of the vote totals. The DRE devices are not connected to a network and

Page #	Statement from Rubin Report	State of Maryland Controls
	records."	physical access would be required to get to the data. The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to connect devices to the system would be easily visible to any of the many election officials.
15	*"Even if proper key management were to be implemented, many problems would still remain. First, DES keys can be recovered by brute force in a very short time period [Gil98]. DES should be replaced with either triple-DES [Sch96] or, preferably, AES [DJ02]."*	We found no evidence that data was encrypted. However, the devices are not connected to a network and physical access would be required to get to the data. The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to connect devices to the system would be easily visible to any of the many election officials.
15	*"Second, DES is being used in CBC mode which requires an initialization vector to ensure its security. The implementation here always uses zero for its IV. This is illustrated by the call to DesCBCEncrypt in TSElection/RecordFile.cpp;* *since the second to last argument is NULL, DesCBCEncrypt will use the all-zero IV.* *DesCBCEncrypt((des_c_block*)tmp, (des_c_block*)record.m_Data, totalSize,* *DESKEY, NULL, DES_ENCRYPT);* *This allows an attacker to mount a variety of cryptanalytic attacks on the data."*	Currently, **[Redacted]** encryption is only used for the resident DRE memory. Once the DRE is powered down, the memory is erased. Note, we have recommended that encryption be employed for the modem transmission of the vote totals. The DRE devices are not connected to a network and physical access would be required to get to the data. The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to connect devices to the system would be easily visible to any of the many election officials.
15	*"Before being encrypted, a 16-bit cyclic redundancy check (CRC) of the plaintext data is computed. This CRC is then stored along with the cipher text in the file and verified whenever the data is decrypted and read. This process in*	Currently, **[Redacted]** encryption is only used for the resident DRE memory. Once the DRE is powered down, the memory is erased. Note, we have recommended that encryption be employed for the modem transmission of the

Page #	Statement from Rubin Report	State of Maryland Controls
	handled by the ReadRecord and WriteRecord functions in TSElection/ RecordFile.cpp. Since the CRC is an unkeyed, public function, it does not provide any real integrity for the data. In fact, by storing it in an unencrypted form, the purpose of encrypting the data in the first place (leaking no information about the contents of the plaintext) is undermined. A much more secure design would be to first encrypt the data to be stored and then to compute a keyed cryptographic checksum (such as HMAC-SHA1 [BCK96]) of the ciphertext. This cryptographic checksum could then be used to detect any tampering with the plaintexts. Note also that each entry has a timestamp, which will prevent the re-ordering, though not deletion, of records. Each entry in a plaintext audit log is simply a time stamped, informational text string. At the time that the logging occurs, the log can also be printed to an attached printer. If the printer is unplugged, off, or malfunctioning, however, no record will be stored elsewhere to indicate that the failure occurred. The following code from TSElection/Audit.cpp demonstrates that the designers failed to consider these issues: *if (m_Print && print) {* *CPrinter printer;* *// If failed to open printer then just return.* *CString name = ::GetPrinterPort();* *if (name.Find(_T("\\")) != -1)* *name = GetParentDir(name) + _T("audit.log");* *if (!printer.Open(name, ::GetPrintReverse(), FALSE))*	vote totals. The DRE devices are not connected to a network and physical access would be required to get to the data. The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to connect devices to the system would be easily visible to any of the many election officials.

Page #	Statement from Rubin Report	State of Maryland Controls
	::TSMessageBox(_T("Failed to open printer for logging")); *} else {* *15* *Do the printing: : :}* *If the cable attaching the printer to the terminal is exposed, an attacker could create discrepancies between the printed log and the log stored on the terminal by unplugging the printer (or, by simply cutting the cable)."*	
16	*"An attacker's most likely target will be the voting records, themselves. Each voter's votes are stored as a bit array based on the ordering in the ballot definition file along with other information such as the precinct the voter was in, although no information that can be linked to a voter's identity is included. If the voter has chosen a write-in candidate, this information is also included as an ASCII string. An attacker given access to this file would be able to generate as many fake votes as he or she pleased, and such votes would be indistinguishable from the true votes cast on the terminal."*	The devices are not connected to a network and physical access would be required to get to the data. The privacy of the voting booth is limited. The AccuVote voting booth provides privacy only for the touch screen and the voter's selections. The action of trying to connect devices to the system would be easily visible to any of the many election officials. Additionally, in the State of Maryland implementation, the total votes recorded on the DRE is reconciled with the number of votes cast on the DRE using the paper Voter Authority Card that is placed into the Voter Authority Card envelope, attached to the DRE voting terminal by the election official.
16	*"While the voter's identity is not stored with the votes, each vote is given a serial number. These serial numbers are generated by a linear congruential random number generator (LCG), seeded with static information about the election and voting terminal. No dynamic information, such as the current time, is used.* *// LCG - Linear Conguential Generator - used to generate ballot serial numbers*	The anonymity of a voter's ballot is preserved because the AccuVote-TS voting system does not use or store personal information and does not provide an individual paper record for each voter, therefore leaving no evidence of a single voter's selections. The individual ballots however, are stored sequentially. If someone kept track of all of the individuals who voted on a particular DRE and then was able to obtain the PCMCIA card, they would be able to tie votes back to individuals. However this would require collusion between multiple individuals.

Page #	Statement from Rubin Report	State of Maryland Controls
	// A psuedo-random-sequence generator *// (per Applied Cryptography, by Bruce Schneier, Wiley, 1996)* *#define LCG_MULTIPLIER 1366* *#define LCG_INCREMENTOR 150889* *#define LCG_PERIOD 714025* *static inline int lcgGenerator(int lastSN)* *{* *return ::mod(((lastSN * LCG_MULTIPLIER) + LCG_INCREMENTOR), LCG_PERIOD);* *}* *While the code's authors apparently decided to use an LCG because it appeared in Applied Cryptography[Sch96], LCG's are far from secure. However, attacking this random number generator is unnecessary for determining the order in which votes were cast: each vote is written to the file sequentially. Thus, if an attacker is able to determine the order in which voters cast their ballots, the results file has a nice list, in the order in which voters used the terminal. A malevolent poll worker, for example, could surreptitiously track the order in which voters use the voting terminals. Later, in collaboration with other attackers who might intercept the poorly encrypted voting records, the exact voting record of each voter could be reconstructed."*	
16	*"Physical access to the voting results may not even be necessary to acquire the voting records, if they are*	Voting records are not transmitted via the Internet in the State of Maryland implementation

Page #	Statement from Rubin Report	State of Maryland Controls
	necessary to acquire the voting records, if they are transmitted across the Internet."	State of Maryland implementation.
17	*"We first note that it is possible for an adversary to tamper with the voting terminals' ballot definition file (election.edb). If the voting terminals load the ballot definition from a floppy or removable storage card, then an adversary, such as a poll worker, could tamper with the contents of the floppy before inserting it into the voting terminal."*	LBEs do load ballots and a malicious worker could tamper with this process. Each LBE has policies and procedures in place, such as a two-person rule, to limit any single individuals access to voting terminals. The Logic and Accuracy testing preformed prior to the election, would uncover any falsified ballots.
17	*"On a potentially much larger scale, if the voting terminals download the ballot definition from the Internet, then an adversary could tamper with the ballot definition file en-route from the back-end server to the voting terminal. With respect to the latter, we point out that the adversary need not be an election insider; the adversary could, for example, be someone working at the local ISP."*	DRE devices are distributed with the approved ballots loaded and locked into the machine. The machines are sealed with tamper-proof tape prior to shipment to the polling site. The Election Judges remove the tamper-proof tape the morning of the election.
17	*"If a wireless network is used, anybody within radio range becomes a potential adversary. With high-gain antennas, the adversary can be sufficiently distant to have little risk of detection. If the adversary knows the structure of the ballot definition, then the adversary can intercept and modify the ballot definition while it is being transmitted. Even if the adversary does not know the precise structure of the ballot definition, many of the fields inside are easy to identify and change, including the candidates' names, which appear as plain ASCII text.10"*	Wireless networking is not used.
17	*"Let us now consider some example attacks that make use of modifying the ballot definition file. Because no cryptographic techniques are in place to guard the integrity of the ballot definition file, an attacker could add, remove, or change issues on the ballot, and thereby confuse the result of the election."*	DRE devices are distributed with the approved ballots loaded and locked into the machine. The machines are sealed with tamper-proof tape prior to shipment to the polling site. The Election Judges remove the tamper-proof tape the morning of the election.

Page #	Statement from Rubin Report	State of Maryland Controls
17	*"Likewise, an attacker who can change the ballot definition could also change the ordering of the candidates running for a particular office. Since, at the end of the election, the results are uploaded to the server in the order that they appear in the ballot definition file, and since the server will believe that the results appear in their original order, this attack could also succeed in swapping the votes between parties in a predominantly partisan precinct. This ballot reordering attack is also discussed in more detail in Section 4.3."*	DRE devices are distributed with the approved ballots loaded and locked into the machine. The machines are sealed with tamper-proof tape prior to shipment to the polling site. The Election Judges remove the tamper-proof tape the morning of the election.
17	*"Suppose that the election officials are planning to download the configuration files over the Internet and that they are running late and do not have much time before the election starts to distribute ballot definitions manually (i.e., they might not have enough time to distribute physical media with the ballot definition files from central office to every voting precinct). In such a situation, an adversary could mount a traditional Internet denial-of-service attack against the election management's server and thereby prevent the voting terminals from acquiring their ballot definitions before the start of the election. To mount such an attack effectively, the adversary would ideally need to know the topology of the system's network, and the name of the server(s) supplying the ballot definition file.12 If a fair number of people from a certain demographic plan to vote early in the morning, then this could impact the results of the election."*	DRE devices are distributed with the approved ballots loaded and locked into the machine. The machines are sealed with tamper-proof tape prior to shipment to the polling site. The Election Judges remove the tamper-proof tape the morning of the election.
18	*"Unlike such traditional attacks, however, the network-based attack (1) is relatively easy for anyone with knowledge of the election system's network topology to accomplish; (2) this attack can be performed on a very large scale, as the central distribution point(s) for ballot definitions becomes an effective single point of failure; and (3) the attacker can be physically located anywhere in the*	The DRE devices are not connected to the Internet or to any other network. The DRE devices are distributed with the approved ballots loaded and locked into the machine. The machines are sealed with tamper-proof tape prior to shipment to the polling site. The Election Judges remove the tamper-proof tape the morning of the election.

Page #	Statement from Rubin Report	State of Maryland Controls
	Internet-connected world, complicating efforts to apprehend the attacker. Such attacks could prevent or delay the start of an election at all voting locations in a state. We note that this attack is not restricted to the system we analyzed; it is applicable to any system that downloads its ballot definition files using the Internet."	
18	*"Just as it is possible for an adversary to tamper with the downloading of the ballot definition file (Section 5.1), it is also possible for an adversary to tamper with the uploading of the election results. To make this task even easier for the adversary, we note that although the election results are stored "encrypted" on the voting devices (Section 4.4), the results are sent from the voting devices to the back-end server over an unauthenticated and unencrypted channel. In particular, CTransferResultsDlg::OnTransfer() writes ballot results to an instance of CDL2Archive, which then writes the votes in cleartext to a socket without any cryptographic checksum. Sending election results in this way over the Internet is a bad idea. Nothing prevents an attacker with access to the network traffic, such as workers at a local ISP, from modifying the data in transit."*	The Internet is not used for transmitting voting counts.
18	*"If the voting terminals use a modem connection directly to the tabulating authority's network, rather than the Internet, then the risk of such an attack is less, although still not inconsequential. A sophisticated adversary (or employee of the local phone company) could tap the phone line and intercept the communication."*	Modem communications are subject to intercept. SAIC has recommended: a) encryption for the transmissions; b) a 100% verification of PCMCIA cards to the vote transmissions.
18	*"All of these adversaries could be easily defeated by properly using standard encryption suites like SSL/TLS, used throughout the World Wide Web for e-commerce security. We are puzzled why such a widely accepted and studied technology is not used by the voting terminals to*	Modem communications are subject to intercept. SAIC has recommended: a) encryption for the transmissions; b) a 100% verification of PCMCIA cards to transmissions.

Page #	Statement from Rubin Report	State of Maryland Controls
	safely communicate across potentially hostile networks."	
18	*"In some configurations, where the voting terminals are directly connected to the Internet, it may be possible for an adversary to attack them directly, perhaps using an operating system exploit or buffer overflow attack of some kind. Ideally the voting devices and their associated firewalls would be configured to accept no incoming connections [CBR03]. This concern would apply to any voting terminal, from any vendor, with a direct Internet connection."*	The DRE device is not connected to the Internet or to any other network.
19	*"Of course, reading the source code to a product gives only an incomplete view into the actions and intentions of the developers who created that code. Regardless, we can see the overall software design, we can read the comments in the code, and thanks to the CVS repository, we can even look at earlier versions of the code and read the developers' commentary as they committed their changes to the archive."*	This is not a security requirement.
19	*"Inside cvs.tar we found multiple CVS archives. Two of the archives, AccuTouch and AVTSCE implement full voting terminals. The AccuTouch code dates to around 2000 and is copyrighted by "Global Election Systems, Inc." while the AVTSCE code dates to mid-2002 and is copyrighted by "Diebold Election Systems, Inc." (The CVS logs show that the copyright notice was updated on February 26, 2002.) Many files are nearly identical between the two systems and the overall design appears very similar. Indeed, Diebold acquired Global Election Systems in September, 2001.13 Some of the code, such as the functions to compute CRCs and DES, dates back to 1996, when Global Election Systems was called "I-Mark Systems."* *This legacy is apparent in the code itself as there are portions of the AVTSCE code, including entire classes,*	This is not a security requirement.

Page #	Statement from Rubin Report	State of Maryland Controls
	that are either simply not used or removed through the use of #ifdef statements. Many of these functions are either incomplete or, worse, do not perform the function that they imply as is the case with *CompareFiles in Utilities/FileUtil.cpp:* *BOOL CompareFiles(const CString& file1, const CString& file2)* *{* */* XXX use a CRC or something similar */* *BOOL exists1, exists2;* *HANDLE hFind;* *WIN32_FIND_DATA fd1, fd2;* *exists1 = ((hFind = ::FindFirstFile(file1, &fd1)) != INVALID_HANDLE_VALUE);* *::FindClose(hFind);* *exists2 = ((hFind = ::FindFirstFile(file2, &fd2)) != INVALID_HANDLE_VALUE);* *::FindClose(hFind);* *return (exists1 && exists2 && fd1.nFileSizeLow == fd2.nFileSizeLow);* *}* *Currently the code will declare any two files to be the same*	

Page #	Statement from Rubin Report	State of Maryland Controls
	that have the same size. The author's comment to use a CRC doesn't make much sense, as a byte-by-byte comparison would be more efficient. If this code were ever used, its inaccuracies could lead to wide variety of subsequent errors. While most of the preprocessor directives that remove code correctly use #if 0 as their condition, some use #ifdef XXX. There is no reason that a later programmer should realize that defining XXX will cause blocks of code to be reincluded in the system (causing unpredictable results, at best). We also noticed #ifdef LOUISIANA in the code. Prudent software engineering would recommend a single implementation of the voting software, where individual states or municipalities could have their desired custom features expressed in configuration files."	
20	*"While the system is implemented in an unsafe language (C++), the code reflects an awareness of avoiding such common hazards as buffer overflows. Most string operations already use their safe equivalents, and there are comments reminding the developers to change others (e.g., should really use snprintf). While we are not prepared to claim that there are no buffer overflows in the current code, there are at the very least no glaringly obvious ones. Of course, a better solution would have been to write the entire system in a safe language, such as Java or C#."*	The scope of the risk assessment did not include a review of Diebold's software engineering practices. However, such an attack vector would require network access. The DRE devices are not connected to a network.
20	*"The core concepts of object oriented programming such as encapsulation are well represented, though in some places C++'s non-typesafe nature is exploited with casts that could conceivably fail. This could cause problems in the future as these locations are not well documented."*	This is not a security requirement.
20	*"Overall, the code is rather unevenly commented. While most files have a description of their overall function, the meanings of individual functions, their arguments, and the*	The scope of the risk assessment did not include a review of Diebold's software engineering practices. It should be noted that since the publication of the Rubin report, Diebold has

Page #	Statement from Rubin Report	State of Maryland Controls
	algorithms within are more often than not undocumented."	developed, documented, and implemented a change control process, which has been delivered to the SBE.
21	*"An important point to consider is how code is added to the system. From the CVS logs, we can see that most code updates are in response to specific bugs that needed to be fixed. There are numerous authors who have committed changes to the CVS tree, and the only evidence that we have found that the code undergoes any sort of review process comes from a single log comment: "Modify code to avoid multiple exit points to meet Wyle requirements." This could refer to Wyle Laboratories whose website claims that they provide all manner of testing services."*	The scope of the risk assessment did not include a review of Diebold's software engineering practices. It should be noted that since the publication of the Rubin report, Diebold has developed, documented, and implemented a change control process, which has been delivered to the SBE.
21	*"There are also pieces of the voting system that come from third parties. Most obviously is the operating system, either Windows 2000 or Windows CE. Both of these OSes have had numerous security vulnerabilities and their source code is not available for examination to help rule out the possibility of future attacks. Besides the operating system, an audio library called "fmod" is used.15 While the source to fmod is available with commercial licenses, unless the code is fully audited there is no proof that fmod itself does not contain a backdoor."*	Exploitation of these attack vectors would require network access. The DRE devices are not connected to a network.
21	*"Due to the lack of comments, the legacy nature of the code, and the use of third-party code and operating systems, we believe that any sort of comprehensive, top-to-bottom code review would be nearly impossible. Not only does this increase the chances that bugs exist in the code, but it also implies that any of the coders could insert a malicious backdoor into the system. The current design deficiencies provide enough other attack vectors that such*	The scope of the risk assessment did not include a review of Diebold's software engineering practices. However, such an attack vector requires network access. This risk is mitigated because the DRE devices are not connected to a network.

Page #	Statement from Rubin Report	State of Maryland Controls
	an explicit backdoor is not required to successfully attack the system. Regardless, even if the design problems are eventually rectified, the problems with the coding process may well remain intact."	
21	*"While the code we studied implements a full system, the implementors have included extensive comments on the changes that would be necessary before the system should be considered complete. It is unclear whether the programmers actually intended to go back and remedy all of these issues as many of the comments existed, unchanged, for months, while other modifications took place around them. It is also unclear whether later version of AVTSCE were subsequently created."*	This is not a security requirement.
22	*"There are, however, no comments that would suggest that the design will radically change from a security perspective. None of the security issues that have been discussed in this paper are pointed out or marked for correction. In fact, the only evidence at all that a redesign might at one point have been considered comes from outside the code: the Crypto++ library16 is included in another CVS archive in cvs.tar. However, the library was added in September 2000 and was never used or updated. We infer that one of the developers may have thought that improving the cryptography would be useful, but then got distracted with other business."*	This is not a security requirement.