Journal of Mobile, Embedded and Distributed Systems, vol. I, no. 2, 2009
ISSN 2067 – 4074

Pool-site E-voting Security

Ciprian Stanica-Ezeanu
Cybernetics and Economic Informatics Faculty,
The Bucharest Academy of Economic Studies,
6, Romana Square, district 1, Bucharest,
ROMANIA
cystanica@yahoo.com

Abstract: The aim of this paper is to present e-voting procedure describing its advantages and disadvantages. Conventional security measures such as firewalls or SSL communications are necessary but not sufficient to guarantee the specific security requirements of e-voting. Besides these conventional security measures, it is also necessary to implement an additional layer of specialized security technology to address the specific risks posed by electronic voting and guarantee critical security requirements such as voters' privacy, vote integrity and voter-verifiability. Analyzing the security of Diebold AccuVote-TS voting machine it was observed the vulnerabilities of this machine to different classes of attacks like: vote-stealing attack, Denial-of-Service (DoS) attack and injecting attack code.

Keywords: e-voting, security, pool-site, DRE, attack, BallotStation, VVPAT

1. Introduction

Electronic voting consists in the casting of votes by electronic means rather than traditional means such as paper ballots or postal ballots.
There are two types of electronic voting:
(i) Remote e-voting: casting of votes through any device (PC, mobile phone, PDA, etc.) with an Internet connection;
(ii) Poll-site e-voting: casting of votes from touch-screen electronic voting terminals located in polling stations (these terminals are also known as Direct Recording Electronic voting terminals or DREs).
Electronic voting presents numerous advantages over traditional paper-based voting:

- Speed and accuracy in the vote counting process,
- Accessibility for blind and visually impaired voters,
- Flexibility in the design and modification of the ballots,
- Prevention of involuntary voting errors (e.g., "over-voting" and "under-voting" errors),
- Ease-of-use for voters,
- Support of multiple languages, etc.

Furthermore, in the case of Internet voting, there is the additional advantage of voters' mobility and convenience which generally leads to higher turnout rates.
For large electoral rolls, the cost per voter in e-voting is lower than the cost per voter in traditional voting due to the economies of scale present in e-voting. With electronic voting, as the size of the electoral roll increases, the cost per voter decreases.
Electronic voting is currently being used by many governments worldwide to carry out binding public elections (e.g., Switzerland, Finland, Nederland, Brazil, United States, Australia, etc.). Additionally, many private sector organizations also use electronic voting in their internal electoral processes (e.g., labor union elections, shareholders' meetings, professional associations, etc.).
E-voting can be as secure as (or even, in many cases, more secure than) traditional paper-based voting provided that adequate security measures are adopted.
Conventional security measures such as firewalls or SSL communications are necessary but not sufficient to guarantee the specific security requirements of e-voting. Besides

these conventional security measures, it is also necessary to implement an additional layer of specialized security technology to address the specific risks posed by electronic voting and guarantee critical security requirements such as voters' privacy, vote integrity and voter-verifiability.

2. Pool-site E-Voting systems description

Scytl Secure Electronic Voting (Scytl) is a software company specializing in the development of secure electronic voting solutions [1].
Scytl's solution provides end-to-end security (i.e., from the individual voters to the Electoral Board), preventing internal attacks from system administrators. Votes are encrypted and digitally signed by voters in the voters' voting devices (e.g., PCs) before they are cast. The private key to decrypt the votes is divided in shares and these shares are distributed to the Electoral Board members before the election begins. The private key is destroyed in this splitting process and, therefore, does not exist during the election. At the end of the election, a pre-defined minimum number of Electoral Board members have to meet to reconstruct the private key and decrypt the votes.
Scytl's Internet voting solution puts the control of the electoral process exclusively in the hands of the Electoral Board as it happens with traditional paper-based elections. The Electoral Board members are the only ones that can reconstruct the key to decrypt and count the votes. System administrators or any other actors with privileges in the system do not have access to the private key and, therefore, cannot see nor modify clear-text votes.
Votes are encrypted in the voters' voting device before they are cast. Only the Electoral Board can decrypt the votes by reconstructing the private key. The decryption of the votes is carried out in an isolated and physically secured computer by applying a mixing technique that breaks the correlation between the voters' identity and the clear-text votes in order to guarantee voters' privacy.
Votes are cryptographically protected (i.e., encrypted and digitally signed) while they are stored in the voting servers and, therefore, cannot be manipulated by anyone, not even system administrators with a privileged access to the system.
Once encrypted, votes are digitally signed by individual voters. The digital certificates used by the voters to digitally sign their encrypted votes can be either pre-existing digital certificates or digital certificates generated "ad-hoc" for that specific election. Before decrypting the votes, the Electoral Board verifies that the digital signatures on the votes belong to valid voters. Votes with invalid digital signatures are "red-flagged" and put aside for further auditing.
There are two levels of security to prevent multiple vote casting. The first level is the electoral roll data base that marks the voters who have already cast votes to prevent them from casting additional votes. The second level is the verification of the digital signatures on the encrypted votes that the Electoral Board performs before decrypting and counting those votes. In case a voter had cast two votes, the Electoral Board would detect the duplication at this time.
Voters are provided with a voting receipt that contains a unique identifier which is randomly generated in the voters' voting device and, therefore, is only known to the voter. This unique identifier is encrypted with the vote in a digital envelope. Only the Electoral Board will be able to open the digital envelope and retrieve the vote and the unique identifier. At the end of the election, the Electoral Board publishes the list of the retrieved unique identifiers and voters are able to check that their individual votes have reached the Electoral Board and been counted.
Scytl's voting receipt does not disclose the voting options selected by the voter and, therefore, does not allow vote selling or voter coercion.
Scytl believes that transparency is an integral part of security. This is why Scytl provides election authorities (and independent auditors designated by the election authorities)

ISSN 2067 – 4074

with access to the source code of the e-voting solution. Once audited, this source code is digitally signed by election authorities to make sure that the same source code is used in the election.
Scytl's solution generates logs for all the actions taken during the election. These logs are cryptographically chained every time a new log is generated in order to prevent any tampering. These immutable logs allow an accurate audit of the election results by election authorities and third parties at the end of the election.
Diebold AccuVote-TS and its newer relative the AccuVote-TSx are together the most widely deployed electronic voting platform in the United States. In the November 2006 general election, these machines are scheduled to be used in 357 counties representing nearly 10% of registered voters.
New studies showed that the machine is vulnerable to a number of extremely serious attacks that undermine the accuracy and credibility of the vote counts it produces.
Computer scientists have generally been skeptical of voting systems of this type, Direct Recording Electronic (DRE), which are essentially general-purpose computers running specialized election software. Experience with computer systems of all kinds shows that it is exceedingly difficult to ensure the reliability and security of complex software or to detect and diagnose problems when they do occur. Yet DREs rely fundamentally on the correct and secure operation of complex software programs. Simply put, many computer scientists doubt that paperless DREs can be made reliable and secure, and they expect that any failures of such systems would likely go undetected.

Fig. 1. The Diebold AccuVote-TS voting machine

Main findings [2] of Diebold's insecurity are:
1. Malicious software running on a single voting machine can steal votes with little if any risk of detection. The malicious software can modify all of the records, audit logs, and counters kept by the voting machine, so that even careful forensic examination of these records will find nothing amiss.
2. Anyone who has physical access to a voting machine, or to a memory card that will later be inserted into a machine, can install said malicious software using a simple method that takes as little as one minute. In practice, poll workers and others often have unsupervised access to the machines.
3. AccuVote-TS machines are susceptible to voting-machine viruses – computer viruses that can spread malicious software automatically and invisibly from machine to machine during normal pre- and postelection activity.
4. While some of these problems can be eliminated by improving Diebold's software, others cannot be remedied without replacing the machines' hardware. Changes to election procedures would also be required to ensure security.

3. Classes of Attacks

An analysis of different kinds of attacks was made by Feldman [4] and a short description of each type will be presented as following.

3.1 Vote-Stealing Attacks

The above presented AccuVote-TS machine is vulnerable to attacks that steal votes from one candidate and give them to another. Such attacks can be carried out without leaving any evidence of fraud in the system's logs. To avoid detection, a vote-stealing attack must transfer votes from one candidate to another, leaving the total number of votes unchanged so that poll workers do not notice any discrepancy in the number of votes reported. Attacks that only add votes or only subtract votes would be detected when poll workers compared the total vote count to the number of voters who checked in at the front desk.

This machine maintains two records of each vote – one in its internal flash memory and one on a removable memory card. These records are encrypted, but the encryption is not an effective barrier to a vote-stealing attack. Malicious software running on the machine would modify both redundant copies of the record for each vote it altered. Although the voting machine also keeps various logs and counters that record a history of the machine's use, a successful vote-stealing attack would modify these records so they were consistent with the fraudulent history that the attacker was constructing. In the Diebold DRE these records are stored in ordinary flash memory, so they are freely modifiable by malicious software. Such malicious software can be grafted into the BallotStation election software (by modifying and recompiling BallotStation if the attacker has the BallotStation source code, or by modifying the BallotStation binary), it can be delivered as a separate program that runs at the same time as BallotStation, it can be grafted into the operating system or bootloader, or it can occupy a virtualized layer below the bootloader and operating system. The machine contains no security mechanisms that would detect a well designed attack using any of these methods.

However it is packaged, the attack software can modify each vote as it is cast, or it can wait and rewrite the machine's records later, as long as the modifications are made before the election is completed. The attack code might be constructed to modify the machine's state only when the machine is in election mode and avoid modifying the state when the machine is performing other functions such as pre-election logic and accuracy testing. The code could also be programmed to operate only on election days. Obviously, it could be programmed to operate only on *certain* election days, or only at certain times of day. By these methods, malicious code installed by an adversary could steal votes without being detected by election officials. Vote counts would add up correctly, the total number of votes recorded on the machine would be correct, and the machine's logs and counters would be consistent with the results reported – but the results would be fraudulent.

3.2 Denial-of-Service Attacks

Denial-of-service (DoS) attacks aim to make voting machines unavailable on election day or to deny officials access to the vote tallies when the election ends. It is often known in advance that voters at certain precincts, or at certain times, will vote disproportionately for one party or candidate. A targeted DoS attack can be designed to distort election results or to spoil an election that appears to be favoring one party or candidate. Several kinds of DoS attacks are practical because of the ease with which malicious code may be executed on the voting machines.

One style of DoS attack would make voting machines unavailable on election day. For example, malicious code could be programmed to make the machine crash or malfunction at a pre-programmed time, perhaps only in certain polling places. In an extreme example, an attack could strike on election day, perhaps late in the day, and completely wipe out the state of the machine by erasing its flash memory. This would

ISSN 2067 – 4074

destroy all records of the election in progress, as well as the bootloader, operating system, and election software. The machine would refuse to boot or otherwise function. The only way to restore such a machine to a working state would be to have a service technician visit, install a special EPROM chip on the machine's motherboard, and reboot the machine from that EPROM. If many machines failed at once, available technicians would be overwhelmed. The result would be a fresh install of the machine's software, with all records of past and current elections still lost. A similar style of DoS attack would try to spoil an election by modifying the machine's vote counts or logs in a manner that would be easy to detect but impossible to correct, such as by injecting malicious code that adds or removes so many votes that the results at the end of the day are obviously wrong. A widespread DoS attack of either style could require the election to be redone.

3.3 Injecting Attack Code

To carry out these attacks, the attacker must somehow install his malicious software on one or more voting machines. If he can get physical access to a machine for as little as one minute, he can install the software manually. The attacker can also install a voting machine virus that spreads to other machines, allowing him to commit widespread fraud even if he only has physical access to one machine or memory card.

3.3.1 Direct Installation

An attacker with physical access to a machine would have least three methods of installing malicious software. The first is to create an EPROM chip containing a program that will install the attack code into the machine's flash memory, and then to open the machine, install the chip on its motherboard, and reboot from the EPROM.

The second method is to exploit a back door feature in Diebold's code to manually install the attack files from a memory card. When the machine boots, it checks whether a file named explorer.glb exists on the removable memory card. If such a file is present, the machine boots into Windows Explorer rather than Diebold's BallotStation election software. An attacker could insert a memory card containing this file, reboot the machine, and then use Explorer to copy the attack files onto the machine or run them directly from the card.

The third method exploits a service feature of the machine's bootloader. On startup, the machine checks the removable memory card for a file named fboot.nb0. If this file exists, the machine replaces the bootloader code in its on-board flash memory with the file's contents. An attacker could program a malicious bootloader, store it on a memory card as fboot.nb0, and reboot the machine with this card inserted, causing the Diebold bootloader to install the malicious software. (A similar method would create a malicious operating system image). The first method requires the attacker to remove several screws and lift off the top of the machine to get access to the motherboard and EPROM. The other methods only require access to the memory card slot and power button, which are both behind a locked door on the side of the machine. The lock is easily picked—one member of our group, who has modest locksmithing skills, can pick the lock consistently in less than 10 seconds. Alternatively, this slot can be reached by removing screws and opening the machine. Some attackers will have access to keys that can open the lock—all AccuVote-TS machines in certain states use identical keys, there are thousands of keys in existence, and these keys can be copied at a hardware or lock store.

3.3.2 Voting Machine Viruses

Rather than injecting code into each machine directly, an attacker could create a computer virus that would spread from one voting machine to another. Once installed on a single "seed" machine, the virus would spread to other machines by methods described below, allowing an attacker with physical access to one machine (or card) to infect a potentially large population of machines. The virus could be programmed to install malicious software, such as a vote-stealing program or denial-of-service attack, on every machine it infected.

An infected machine will infect any memory card that is inserted into it. An infected memory card will infect any machine that is powered up or rebooted with the memory card inserted. Because cards are transferred between machines during vote counting and administrative activities, the infected population will grow over time. Diebold delivers software upgrades to the machines via memory cards: a technician inserts a memory card containing the updated code and then reboots the machine, causing the bootloader to install the new code from the memory card. This upgrade method relies on the correct functioning of the machine's bootloader, which is supposed to copy the upgraded code from the memory card into the machine's flash memory. But if the bootloader were already infected by a virus, then the virus could make the bootloader behave differently. For example, the bootloader could pretend to install the updates as expected but instead secretly propagate the virus onto the memory card. If the technician later used the same memory card to "upgrade" other machines, he would in fact be installing the virus on them.
Memory cards are also transferred between machines in the process of transmitting election definition files to voting machines before an election. If one of the few units that download the data is infected, it will transfer the infection via the "stacks of memory cards" into many voting machines.

4. Voter-Verifiable Paper Trial and Random Audits

The most important strategy for mitigating vote-stealing attacks is to use a voter-verifiable paper audit trail (VVPAT) coupled with random audits. The VVPAT creates a paper record, verified visually by the voter, of how each vote was cast. This record can be either a paper ballot that is deposited by the voter in a traditional ballot box, or a ballot-under-glass system that keeps the paper record within the voting machine but lets the voter see it. A VVPAT makes the vote-stealing attack detectable. In an all-electronic system like the Diebold DREs, malicious code can modify all of the logs and records in the machine, thereby covering up its vote stealing, but the machine cannot modify already created paper records, and the accuracy of the paper records is verified by voters.
Paper trails have their own failure modes, of course. If they are poorly implemented, or if voters do not know how or do not bother to check them, they may have little value. The real advantage of a paper trail is that its failure modes differ significantly from those of electronic systems, making the combination of paper and electronic recordkeeping harder to defraud than either would be alone. Requiring a would-be vote stealer to carry out both a code-injection attack on the voting machines and a physical ballot box stuffing attack would significantly raise the difficulty of attacking the system.
Paper ballots are only an effective safeguard if they are actually used to check the accuracy of the machines. This need not be done everywhere. It is enough to choose a small fraction of the polling places at random and verify that the paper ballots match the electronic records there. If the polling places to recount are chosen by a suitable random procedure, election officials can establish with high probability that a full comparison of paper and electronic records would not change the election's result. Another limitation of VVPATs is that they cannot stop a denial-of-service attack from spoiling an election by disabling a large number of voting machines on election day. Given this possibility, if DREs are used, it is worthwhile to have an alternative voting technology available. A decidedly low-tech system such as paper ballots may be suitable for this purpose.

5. Conclusion

ISSN 2067 – 4074

In the security community it is not considered very productive to make statements about the security of any system without at least defining what it is we're securing, and what it is we're securing it against. In the case of voting systems, the general concept of security is often mistakenly taken to mean "the system-wide level of over-all security against any attack, mounted by outsiders, that affects the election results". There are many more possible interpretations of security when it comes to elections. The above definition for instance ignores the risk posed by an attack by knowledgeable insiders, or that of an attack that involves only the secrecy of the ballot.
In the case of voting systems, the only meaningful security against insiders is to have a voting mechanism of which all the details are published, and that a substantial portion of the general population is capable of comprehending in-depth. Any other solution creates a situation in which the population depends in essence on reassuring statements that cannot be verified independently. In a country where e-Voting has replaced traditional paper ballots, the level of confidence with which the population views these statements is then by definition the upper bound for the trust the population can have regarding the outcome of any election, and thus in effect a measure for the legitimacy of government.

A preliminary draft of this paper was presented the SECITC 2008 International Conference.

References

[1] http://www.scytl.com/
[2] http://www.youtube.com/watch?v=5WMG34cv0zM
[3] http://freedemocracy.blogspot.com/2008/01/how-to-hack-diabold-accuvote-ts.html
[4] Feldman, A.J., Halderman, J.A., Felten, E.W., *USENIX/Accurate Electronic Voting Technology Workshop (EVT'07*), Boston, 2007, USA