# An Authentication and Ballot Layout Attack against an Optical Scan Voting Terminal

*Aggelos Kiayias* *Laurent Michel* *Alexander Russell* *Narasimha Shashidhar*
*Andrew See* *Alexander A. Shvartsman*

## Abstract

Recently, two e-voting technologies have been introduced and used extensively in election procedures: direct recording electronic (DRE) systems and optical scanners. The latter are typically deemed safer as many recent security reports have discovered substantial vulnerabilities in a variety of DRE systems. In this paper we present an attack against the Diebold Accuvote optical scan voting terminal (AV-OS). Previously known attacks direct to the AV-OS required physical access to the memory card and use of difficult to find hardware (card reader/writer).

Our attack bypasses these issues by using the serial port of the AV-OS terminal and reverse engineering the communication protocol, in essence, using the terminal itself as a reader/writer. Our analysis is based solely on reverse-engineering. We demonstrate how an attacker can exploit the serious security vulnerability of weak (non-cryptographic) authentication properties of the terminal. The attack payload delivers a tampered ballot layout that, depending on the scenario, allows swapping of candidate votes, neutralizing votes, or even shifting votes from one candidate to another.

## 1 Introduction

Broadly speaking, there are two major types of E-voting equipment: Direct-recording electronic (DRE) machines and optical-scan (OS) machines. As certain DRE's have recently been shown to be subject to dramatic tampering that can invalidate an election in which they participate, OS technology is believed to be the safer alternative. In addition to these widely publicized security flaws in the design of currently employed DRE terminals, [2, 3], some DRE's lack a paper audit trail (also known as a *voter verifiable audit trail* or VVAT): for such machines there is no independent voter-verified method for verifying the validity of election results. (Even in the case that a VVAT is present it is not necessarily foolproof, as it is also "machine-produced.") Optical-scan machines, on the other hand, though also the subject to security criticism [1], have the important and obvious benefit that they naturally yield a voter-verified paper trail (or more accurately voter-produced paper-trail): the actual "bubble sheet" ballots marked by the voters. We note that there are vulnerabilities that may be jointly shared by both technologies e.g., such as those discussed in [5].

The subject of this paper is the AccuVote Optical Scan voting terminal (AV-OS) manufactured by Diebold, Incorporated, Election Systems division, shown in Figure 1. The AV-OS is a widely adopted machine both in the USA and abroad. In 2006 mid-term elections at least 24,000 units were deployed. As we emphasized already optical scan voting is freed from some of the perils of paperless trails or computer generated paper trails; nevertheless, the election still relies on the terminal to electronically add the votes and report the results; this introduces the possibility of attacks that interfere with these basic tabulation and reporting tasks.

The AV-OS has been the subject of the report of H. Hursti [1], pointing out that the AV-OS memory card lacks cryptographic integrity checks (and that this can lead to serious security vulnerabilities). These findings lead many jurisdictions employing the AV-OS to choose to seal the card in the terminal with a tamper-evident seal (cf. Figure 1) during elections and further demand that it be delivered and shipped back from polling locations with such seals in place.

Given the above, an important question is whether this measure is sufficient to protect the elections. In contrast to Hursti's attack [1], we use the voting terminal itself as a card reader/writer. Our attack thus does not rely on special hardware or physical access to the card. We discovered the vulnerabilities through direct experimentation without source code or other detailed knowledge.

We have demonstrated the attack against a AV-OS terminal in pre-election state using an ordinary PC with a

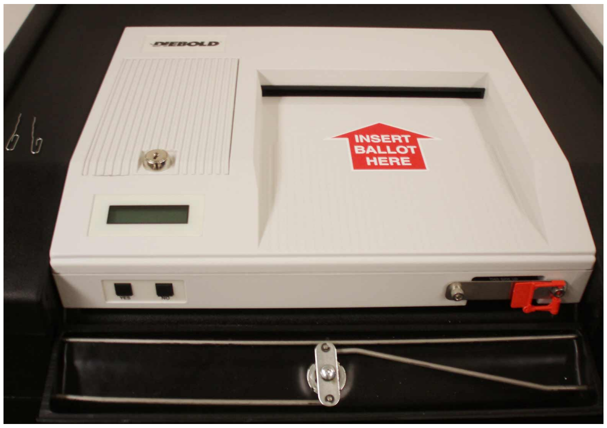


Figure 1: The AccuVote Optical Scan voting terminal (AV-OS). The terminal is shown prior to it being locked to the ballot box, with its front panel visible and showing two control buttons (Yes/No on the lower left corner) and the memory card slot with the card sealed in (lower right corner).

standard serial port. Start to finish the attack can be carried out in under 5 minutes.

## 2 Basic Characteristics

The AV-OS election system consists of two components: the AccuVote Optical Scan voting terminal (the AV-OS terminal) and the ballot design and central tabulation system (GEMS, for Global Election Management System). These components have the following characteristics:

- The GEMS software is installed on a conventional PC that is equipped with a serial port and includes a ballot design system and a tabulation system.

- The specifications of an election are downloaded onto a 40-pin 128KB Epson memory card present in the AV-OS. It should be noted that the memory card has been discontinued by Epson, and no reader/writer for this type of medium is readily available in the market.

- The AV-OS system used in this study contained the firmware version 1.96.6 (in the form of an EPROM chip). It is equipped with an optical scanner, a paper-tape dot-matrix printer, a LCD display, a serial communication port, and telephone jacks leading to a built-in modem. It runs on a V25 CPU (an 8088 compatible processor). For election deployment the system is secured within a ballot box so that no sensitive controls or connectors are exposed to the voter.

## 3 Security Vulnerabilities

This section briefly describes several vulnerabilities that were discovered in the AV-OS system. Section 4 describes the attack scenario that exploits these vulnerabilities. We note that the first two vulnerabilities are newly discovered in this report to the best of our knowledge; the third vulnerability is well known from the works of [1, 5].

**The AV-OS leaks the memory card contents:**

The AV-OS terminal allows any operator to obtain a dump of its installed memory card contents *without any authentication control*. In particular, given access to an AV-OS machine one can obtain all the information that is stored in the memory card (including, e.g., the machine's PIN) in a matter of seconds.

**The communication between AV-OS GEMS is unauthenticated:**

During the initialization of a machine for election the GEMS system communicates with the AV-OS terminal to write the initial election setup to the memory card. No encryption or cryptographic authentication is performed during this transmission. The serial line protocol does use a cyclic redundancy check (CRC) mechanism for error control.

**Executable code within the memory card:**

Each memory card contains executable code that is used for printing the reports. The code is written in a proprietary symbolic language. Such executable files are identified as `.abo` (AccuBasic Object) bytecode. The possibility to modify the code that prints the results opens the possibility to corrupt machines and coerce them into misinterpreting their counters.

## 4 The Attack

We now present a general attack against the AV-OS system that use the vulnerabilities described above. The attack entirely compromises the election process assuming that the attacker has a few minutes of access to the AV-OS terminal prior to election time. The attacker never has to directly access the memory card. Instead it uses the *terminal itself* to compromise the card contents.

## 4.1 Compromising the Election

By compromising the election we refer to an attacker's capability to put the AV-OS in a state where it miscounts the ballots that are inserted into the machine. For example an election would be compromised if the votes received by two candidates are swapped or if the votes of a candidate were nullified.

To streamline our attack, we have developed a proof-of-concept software package that processes card dump data, extracting the ballot layout, password (PIN), and audit information, and computes a serial payload to reprogram the card. We emphasize that our software was developed by observing the AV-OS system during normal operation, without access to any technical information about the system, its internals, or access to the source code of AV-OS or GEMS. Specifically, the attack was developed with precisely the same information and access to the system that is normally available to, for example, election administrators (poll workers and other town officials). Note, however, that to actually *carry out* the attack, one only needs physical access to the voting machine, without the privileges of an election administrator. Furthermore, in the absence of our software, an attacker could use a standard terminal emulator and readily available editing tools to perform the same attack. Equipped with a laptop and a regular RS-232 null modem serial cable, an attacker needs only to gain physical access to an AV-OS terminal prior to the election. Furthermore, the attacker needs no knowledge of the particulars of the election he is to undermine (such as exact candidates' names, ballot layout, precinct names, or any kind of passwords). The whole process can be completed in a matter of a few minutes. In the following we perform a step-by-step demonstration of the attack.

**Step 1 : Gaining physical access[1].** Prior to the election the terminal is presumably locked within the ballot box. At this stage, the system has been initialized with all the election data and its removable memory card is sealed with a tamper evident seal. The first thing an attacker must do is gain access to the front side of the AV-OS that is concealed by the ballot-box. If the box is unlocked or the attacker has the keys this is straightforward. We note that the locks used are regular pin tumbler locks similar to those found in filing cabinets, office drawers or other standard computerized equipment. If the attacker lacks the key, picking the lock can be done in a short amount of time ranging from seconds to minutes (it is feasible even for someone who has never done it before using information available online, e.g., [4]). Picking the lock requires no special equipment: in fact two standard paper clips are sufficient (cf. Figure 1). Once the lock is opened, the front side of the machine is freed and the attacker can access the Yes/No buttons located on the left of the front panel of the AV-OS, see Figure 1.

**Step 2 : Dumping the memory card contents.** Once the AV-OS terminal is accessible from the front, the attacker can pull it slightly outwards and obtain access to its back side. There, a number of standard connection ports are available including a RS-232 serial port and a telephone line jack. The attacker uses a standard serial cable to connect the machine to her laptop. In order to prepare for the attack the laptop must capture the data sent to its serial port by the AV-OS. Though we have written software which automates this portion of the attack, a standard terminal emulator would suffice.

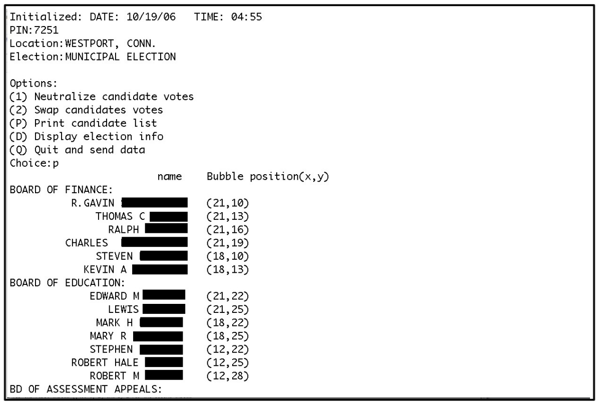
Once the serial cable is in place, the attacker turns on the machine using the on/off switch located on the right of the machine's back panel while simultaneously depressing the two buttons on the front panel. This results in the AV-OS entering a special diagnostic mode. The terminal asks for no password or other identification from the operator in order to enter into such a mode. One of the options that is available to the attacker in this mode is to dump the contents of the installed memory card through the serial line. This is the option that the attacker selects and the AV-OS dumps the card contents.

It takes roughly two minutes to receive (and parse) the card dump that the AV-OS transmits. The dump of the card is sent in cleartext and the only component that is hidden is the PIN that enables the attacker to enter into a special "supervisor mode." This is the mode that poll-workers have access to during election time. The 4-digit PIN is contained in an obfuscated form at a fixed location in the dump. The election compromising software de-obfuscates and prints the supervisor mode access PIN, see Figure 2. In addition, the software decodes the "audit history" that appears in the card dump, including the entry containing the initialization timestamp for the card as well as any other entry in the transaction log of the terminal.

In the same screenshot the main menu is presented and offers the following options: (1) neutralize candidate votes, (2) swap candidate votes, (P) print candidate list, (D) display election info, (Q) quit and send data. Using the options (P) and (D) the attacker can obtain all information about the election including the ballot layout.

**Step 3 : Ballot design remapping.** In order to understand the specifics of the attack, the following gives an overview of the election setup of the AV-OS system. Each candidate and race has a unique identifier. The candidates for each race are encoded together with an $(x, y)$ coordinate (cf. Figure 2), which corresponds to the

[1] If the attacker has access to the election-ready voting terminal prior to its being locked within the ballot box, proceed to Step 2.



```
Initialized: DATE: 10/19/06   TIME: 04:55
PIN:7251
Location:WESTPORT, CONN.
Election:MUNICIPAL ELECTION

Options:
(1) Neutralize candidate votes
(2) Swap candidates votes
(P) Print candidate list
(D) Display election info
(Q) Quit and send data
Choice:p
                        name    Bubble position(x,y)
BOARD OF FINANCE:
          R.GAVIN [redacted]    (21,10)
             THOMAS C [redacted] (21,13)
               RALPH [redacted]  (21,16)
         CHARLES  [redacted]     (21,19)
             STEVEN [redacted]   (18,10)
          KEVIN A [redacted]     (18,13)
BOARD OF EDUCATION:
            EDWARD M [redacted]  (21,22)
               LEWIS [redacted]  (21,25)
             MARK H [redacted]   (18,22)
            MARY R [redacted]    (18,25)
            STEPHEN [redacted]   (12,22)
          ROBERT HALE [redacted] (12,25)
            ROBERT M [redacted]  (12,28)
BD OF ASSESSMENT APPEALS:
```

Figure 2: (*Top part*) : The main menu of the election compromising software. The de-obfuscated PIN is prominently presented. (*Bottom part*) : The listing of candidates for some of the races and the corresponding bubble sheet coordinates for each candidate. (This snapshot is touched-up to black out the last names used in this fictitious race.)

bubble on the paper ballot sheet that the voters mark in order to vote for that particular candidate. The ballots are printed taking into account this configuration. The correct correspondence between printed ballots and internals of the memory card is essential for the election to go through uncompromised. This correspondence is one of the aspects of the election system that the attack subverts.

To neutralize a candidate in a specific race, the attacker simply maps the $(x,y)$ coordinate of the candidate to some location that is beyond the ones used for the election (note that most coordinates are in fact unused; thus it is trivial for the software to recover such a position). Analysis of the AV-OS determined that the *checksum* printed on the audit tape is computed from the election data, including candidate ballot locations. In the current implementation of the election compromising software the location selected for neutralizing a candidate whose coordinates are $(x,y)$ is $(x-1,y+1)$. The $(x-1,y+1)$ pair is suitable as this choice will not affect the value of the (modular addition based) checksum that the terminal computes from the ballot layout data.

A more insidious attack is to swap two candidate's votes. Following the previous rationale if the bubble coordinates assigned to candidate $A$ are $(x_A,y_A)$ and the bubble coordinates assigned to candidate $B$ are $(x_B,y_B)$, by simply swapping the coordinates one effectively makes AV-OS count a vote for candidate $A$ as a vote for candidate $B$ and vice versa. The swapping attack has the advantage that it is then possible to modify the bytecode in order to "un-swap" the reported results in order to avoid detection during ballot testing, i.e., before the official start of election.

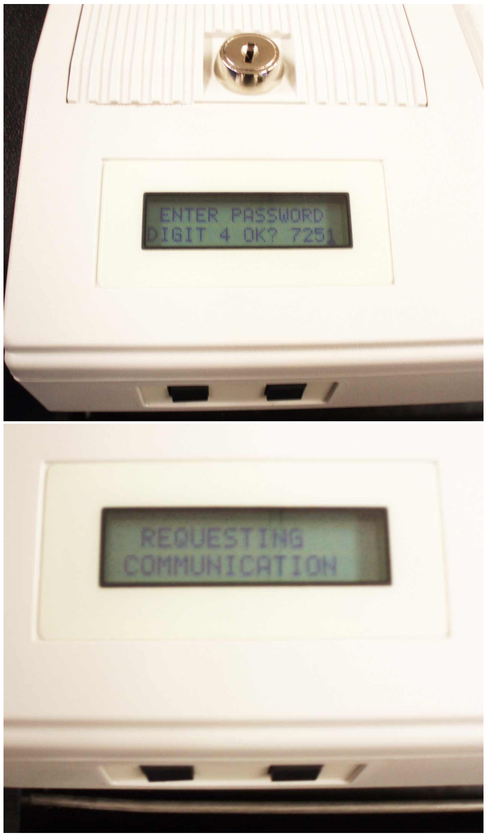


Figure 3: (*Left*) : The attacker enters the PIN to enter supervisor mode. (*Right*) : The AV-OS is requesting communication from the GEMS system to overwrite the memory card contents with the forged election setup.

These modifications are built into the software that also includes additional payloads for biasing the reporting functionality of the terminal. What needs to be performed next by the attacker is to use the AV-OS to reprogram the memory card with this altered election data.

**Step 4: Adjusting the AV-OS clock to agree with the card's initialization timestamp.** When the election compromising software processes the dump of the memory card, it also recovers the time and date at which the card was originally programmed for the election. To insure that this timestamp is preserved in the audit history of the new image of the card to be created in Step 6 below, the attacker would need to reset the clock of the AV-OS so that it agrees with the recovered timestamp. The option to (re)set the clock appears in Diagnostic Mode, obtained by restarting the machine with both buttons pressed.

**Step 5: Temporarily disabling the AV-OS printer.**

When the AV-OS terminal is initialized it prints a tag that can be used for auditing the system and contains the date and time of the initialization as well as some other control information. Given that the attacker will reinitialize the system, in order to prevent the AV-OS from printing such tag, the attacker must disable the printing functionality by selecting the corresponding choice available in the "supervisor menu" of the terminal that is accessible by using the de-obfuscated PIN. This step is optional as the attacker may simply discard the printout, nevertheless the fact that the attacker can disable the printer makes the attack more stealthy and avoids the necessity of picking the lock protecting the printer and audit tape. Nonetheless, it is clear that the ability to disable the printer should not be available in a voting terminal, especially since the tape is the only source of election reports and audit reports in at least some districts using these systems.

It should be noted that neither disabling the printer nor adjusting the clock are included in the audit log. Thus, there will be no sign that the terminal was tampered with if an audit report is printed.

**Step 6 : Impersonating the GEMS system.** Once the AV-OS clock is reset and the printer is shut-off the attacker sets the AV-OS terminal in supervisor mode. In supervisor mode, AV-OS can format the contents of the memory card and accept communication from the GEMS system to initialize the election. The attacker takes advantage of the fact that the AV-OS does not use any strong cryptographic identification check to authenticate the sending entity and hence it can impersonate the GEMS system.

Using the election compromising software the attacker prepares a forged election payload. The preparation of this payload is based on the reverse engineering of the communication between the GEMS system and AV-OS that was performed as part of the vulnerability assessment. The software prepares a fake communication transcript that appears to be originating from GEMS. The transcript contains the election details recovered from the memory dump together with a number of malicious alterations such as candidate swaps, candidate neutralizations and corrupted bytecode reporting functionality.

Figure 3 shows the attacker entering the 4-digit PIN that was recovered from the memory dump to gain access to the options of the supervisor mode of the terminal. In order to start the machine in supervisor mode the unit needs to be turned off and restarted while simultaneously depressing the 'Yes' button. Subsequently the attacker chooses to erase the memory card contents, and the card is formatted. Once the contents of the memory card are erased the unit would request to be initialized from the GEMS system. In Figure 3 the AV-OS terminal requests communication from the GEMS system. The attacker furnishes to the terminal the forged communication transcript.

**Step 7 : Completing the attack.** Once the forged communication is transmitted through the serial port the compromise of the terminal has been successfully completed. The attacker will reset the clock to the current time using the diagnostic mode and will reactivate the printer.

After this step, the AV-OS terminal will be found by poll-workers in its expected pre-election state. The terminal will appear to be functioning normally for all operations during the election. The total time required to compromise the card is only a few minutes.

## 5 Conclusion

We presented the outline of an actual attack against the AV-OS voting terminal that takes advantage of the lack of authentication between the terminal and its ballot management system. The attack relies on newly discovered vulnerabilities and a protocol analysis that were developed from first principles without having access to source code or any information that is not in the public domain.

## References

[1] H. Hursti, Critical Security Issues with Diebold Optical Scan Design, Black Box Voting Project, July 4, 2005. `www.blackboxvoting.org/BBVreport.pdf`

[2] T. Kohno, A. Stubblefield, A. D. Rubin and D. S. Wallach, Analysis of an Electronic Voting System, IEEE Symposium on Security and Privacy 2004.

[3] A. J. Feldman, J. A. Halderman, and E. W. Felten, Security Analysis of the Diebold AccuVote-TS Voting Machine, September 13, 2006. `itpolicy.princeton.edu/voting`

[4] Theodore T. Tool, MIT Guide to Lock Picking, 1991. `people.csail.mit.edu/custo/MITLockGuide.pdf`

[5] David Wagner, David Jefferson and Matt Bishop, Security Analysis of the Diebold AccuBasic Interpreter, Voting Systems Technology Assessment Advisory Board, University of California, Berkeley, February 14, 2006.