Peering Through Chinks in the Armor of High-Tech Elections

Pokey Anderson

May 27, 2007

When you vote, who you voted for is a secret.

Virtually everything else about elections should be public and transparent. But somehow, secrecy has come to cloak many aspects of elections.

First off, the software counting votes is secret.

A raft of other aspects of elections are secret or obscure, including who owns and runs voting machine companies, who their programmers are (google Jeffrey Dean & Diebold for one hair-raising example), reports by certification entities and many experts, and even raw exit poll data by the nation's biggest exit polling group. All secret.

Some voting machine companies even expand the blanket of secrecy, hiding audit logs, error reports, and even vote tabulation databases, claiming they are trade secrets. For example, Alaska election officials resisted one party's efforts to obtain the electronic file stating vote totals; the officials claimed it was proprietary information belonging to the vendor. [1]

Let's just focus on the first item: the secret software that in effect casts our votes for us electronically (on DREs), and tabulates our votes electronically (on DREs and optical scan systems). Voting machine companies claim it is a trade secret, and strenuously resist its disclosure.

Certainly, disclosure would be a good thing. It's an important principle, to roll back the trend toward secrecy in vote counting. It's also important in practice: expert examination could root out some obvious errors, shoddy practices, vulnerabilities, and bugs. A source code review is even more effective in tandem with dynamic testing of the actual system in use.

But, if the software were disclosed, would that solve the problems of using electronic voting systems for elections?

There are unique things about running elections that no other software has to accomplish. When the voter leaves the polling place, because of anonymity, he or she is forever separated from the vote cast, with no way to check and see if it was ultimately counted correctly. This is quite different from using an ATM, which gives you a record at the time of the transaction, sends you a monthly statement to review, and makes a video record of the transaction as well. Plus, if someone were able to empty your entire bank account, they might net what, $1000? $10,000? $100,000?

[1] "State Rebuffs Raw Vote Demand: Standoff: Democrats want 2004 base election data; machine firm is playing coy," by Lisa Demer, January 24, 2006, Anchorage Daily News, http://www.commondreams.org/headlines06/0124-08.htm

By contrast, if someone were able to steal votes, they could put their own puppets into office, and could conceivably use the credit of the United States Treasury to funnel millions or billions of dollars their way. So, there's a huge prize to entice ethically-challenged operatives.

Let's make it personal. The winners of elections get to spend thousands of your tax dollars each year, so you could think of hackable electronic voting machines as leaving your checkbook out on the sidewalk, with a bunch of signed checks.

So again, disclosing the software that counts your vote would be a good thing? Yes.

"The winners of elections get to spend thousands of your tax dollars each year, so you could think of hackable electronic voting machines as leaving your checkbook out on the sidewalk, with a bunch of signed checks."

But, let me explain why disclosed software in elections is no panacea for the problems of electronic elections (both DREs and optical scan).

Here it is in one sentence (take a deep breath here):

Even if a person could check hundreds of thousands of lines of software code and find hidden malicious code,
and even if software could be written bug-free,
and even if the hardware works properly and interfaces perfectly with the software and peripherals,
and even if the binary and source code match identically,
and even if each electronic voting machine were physically guarded every minute to prevent insertion of malicious code (including by insider vendors or subcontractors or election personnel or anyone with a key including the janitor),
and even if every software change has been clean and legitimate,
and even if unexamined ballot definition files are accurate and trustworthy,
and even if there were reasonable ways to make sure that the software previously checked is now the software running on each machine on the morning of election day
. . .
chinks in the voting system armor could allow intrusion DURING voting day and during tabulation.

Let's go through that, step by step.

1. **Can someone find problems by examining election software, which has hundreds of thousands of lines of code?**

   Election system manufacturers resist letting anyone see their code, aside from the rare expert sworn to secrecy. If they allowed examination by independent experts, could those experts reasonably be expected to find any problematic code?

   If they know where to look, they could probably find some.

A Diebold system for voting contains roughly 285,000 lines of source code. Experts for the State of Maryland acknowledged that "only a fraction" could be carefully studied in their review.[2]

I asked David Dill[3] about this: Given the hundreds of thousands of lines of source code, would it be reasonable to expect the certification board to go through and find problems?

David Dill: "It is practically impossible for someone to review software of any length at all -- even 10,000 or maybe even 1500 -- lines of code to make sure that's 100% error-free. The certification is done by organizations called independent testing authorities. They couldn't do it, no matter how hard they tried. Now, from what I have learned, they don't try hard enough. There are claims that the code is inspected line by line. I know that that is not sufficient to find bugs and certainly not to find tampering that is deliberately hidden in that software. In fact, the tampering may not even be in the software that's presented to the independent testing authority." [4]

Another expert agrees.

David Jefferson: "Security professionals know that even if you do look at the code, you're not necessarily going to find hidden malicious logic that does that kind of thing [shaving votes or transferring them from one candidate to another.[5]

Then there are variables. Ellen Theisen told me that no professional would attempt to write election software without using variables (like the 'x' in an algebraic equation, a variable can represent different things at different times). Once variables are in the code, it can be tricky to remember what and how they operate. She said that reading someone else's code is not terribly easy. In fact, rereading 100 lines of code she wrote herself, with good notes, is not always easy, she said.[6]

A conversation between Prof. Avi Rubin and Frank Shugar included this Q&A: "Is it easy to hide undetected code in a great big code package?" "Absolutely." He put the chance of it going undetected at 99.9%.

I asked Prof. Dan Wallach about testing the software if the bug is hiding in it.

Dan Wallach: "I don't know about that other tenth of a percent. This is a classic computer security problem. Whoever gets into the machine first wins. So if the Trojan horse software is in there first, you ask it to test itself -- it will always lie to you and tell you everything is fine. And no matter what testing code you try to add after the fact, it's too late. It can now create a world where the testing software can't tell that the machine has been compromised, even though it has."[7]

[2] Review for the State of Maryland: Diebold AccuVote-TS Voting System, January 20, 2004, RABA Technologies, http://www.raba.com/press/TA_Report_AccuVote.pdf

[3] Brief bios for David Dill and several others who are quoted frequently appear on the last page.

[4] Sunday Monitor, July 13, 2003, KPFT Radio Houston

[5] Telephone interview, May 5, 2004

[6] Telephone interview, May 16, 2007

[7] November 20, 2003 interview, in person

2. **Can software be written bug-free?**

David Jefferson told me that bugs can be elusive and numerous:

David Jefferson: "It's quite possible that there are just plain bugs in the code. And, again, you know, or anyone who has any software background knows, that a lot of bugs are unbelievably difficult to find -- so difficult in fact that after code has been in production for a long time it still typically has hundreds or thousands or tens of thousands of bugs left in it."[8]

For example, Sequoia had an embarrassing moment when it was showing off its shiny new voting equipment to California State Senate staffers in 2004. Kim Zetter writes that "the machine worked fine when the company tested votes using an English-language ballot. But when the testers switched to a Spanish-language ballot, the paper trail showed no votes cast for two propositions. 'It caught a mistake in the programming of the touch-screen machine itself. For some reason it would not record or display the votes on the Spanish ballot for these two ballot measures. The only reason we even caught it was because we were looking at the paper trail to verify it,'" said Darren Chesin, a consultant to the state Senate elections and reapportionment committee. Sequoia spokesman Alfie Charles said the problem was not a programming error but a ballot-design error.[9]

3. **Is the hardware working properly, and does it interface properly with the software and peripherals?**

There are a lot of moving parts in a voting system. There's hardware and software. There are peripherals -- the removable items like memory cards or flash drives that can record votes, or can reprogram the software. And, what about various features -- the code to convert ballots into foreign languages, the code used for disability access? All of these pieces have to work well by themselves and together. All provide a possible entry point for an intruder.

In an example just last week in Texas, there was a figurative and literal meltdown of the equipment. It's unclear whether the memory card, the voting machine hardware or software are to blame. During early voting one of the three Diebold machines stopped allowing people to cast ballots. Later, a second machine malfunctioned. When the polls finally closed, one memory card was opened (a memory card is the electronic equivalent of a wooden ballot box). Oops, there were no votes on it. Officials said it should have 38 votes. Okay, we can get a copy of the ballot records from the internal memory of the voting machine, said a Diebold representative. The records were scheduled to be examined on Tuesday. Unfortunately, there was a fire at Aurora city hall Monday night. The machine in question was damaged -- its case was smoldered. Diebold said it was trying to recover the votes.[10]

So, something failed, and citizens got smoke instead of timely results.

[8] Telephone interview, May 5, 2004

[9] "Wrong Time for an E-Vote Glitch," by Kim Zetter, August 12, 2004, Wired, http://www.wired.com/politics/security/news/2004/08/64569

[10] "City may have to hold new election," By Christina Lane, May 17, 2007, Wise County Messenger [Decatur, Texas], http://www.wcmessenger.com/news/news/EEZAFkkVpADWXqXtfA.php

4. **Will examining the source code always detect any problems in the binary code?**

   Ok, this may be an item only a geek could love, so I'll try to translate. Generally, if experts review software, they are reviewing source code, the human readable version of a program.

   Here is what they consider "human readable" (they are joking, right?):

```
void CBallotRelSet::Open(const CDistrict* district, const CBaseunit* baseunit,
const CVGroup* vgroup1, const CVGroup* vgroup2)
{
ASSERT(m_pDB != NULL);
ASSERT(m_pDB->IsOpen());
ASSERT(GetSize() == 0);
if (district->KeyId() == -1) {
Open(baseunit, vgroup1);
} else {
const CDistrictItem* pDistrictItem = m_pDB->Find(*district);
if (pDistrictItem != NULL) {
const CBaseunitKeyTable& baseunitTable = pDistrictItem->m_BaseunitKeyTable;
int count = baseunitTable.GetSize();
```

   But, what the system actually executes is binary files, which look like 0110000111 (but much longer). According to David Dill and Dan Wallach, "In a sense, investigating the system by reviewing source code is like investigating the collapse of a building by reviewing blueprints. The blueprints have valuable information but the actual building may differ in subtle but significant ways from its blueprints." So, there could be a mistake or even fraud lurking in the translation between source code and binary code.[11]

5. **Is each electronic voting machine physically guarded every minute to prevent insertion of malicious code?**

[12]

   Serious election security would require treating an electronic voting system like a live ballot box from the time it is manufactured to the time of each election. Especially given the very weak software protections that have been revealed in investigations so far, the equipment should be physically protected at all times from malicious intrusion, whether by wireless, modem, flash drive, or other means. Test hackers of Diebold voting equipment have needed only one to four minutes access to the equipment to completely take control of the software.

[11] "Stones Unturned: Gaps in the Investigation of Sarasota's Disputed Congressional Election," Prof. David Dill and Prof. Dan Wallach, April 13, 2007, http://www.cs.rice.edu/~dwallach/pub/sarasota07.pdf, p. 6 and 11

[12] Design courtesy of CA-50 Action Committee. http://www.nosleepovers.org

In an attack by Prof. Ed Felten at Princeton, he showed that the code could easily be configured to "disappear" once its work was done, leaving no trace of tampering.[13]

Numerous opportunities for a private moment with electronic voting equipment have been documented, from machines stored in stairwells or unlocked closets in schools, to machines sent home with thousands of precinct clerks for days before the elections. Brad Friedman calls these "sleepovers."

For instance, in California, Diebold DREs were taken home by election workers prior to the 2006 special election. "Depending when they have training, the machines could be at their homes for more than a week or two," reported Pamela Smith, Nationwide Coordinator for VerifiedVoting.org.

In Harris County, Texas, precinct election equipment is typically in possession of an election worker for days, up to a week before an election, according to two election judges I talked to. At the class they attended for election judges, "they basically gave us some guidelines, that the machines needed to be kept securely—don't leave them in the car, don't leave them in the hallway, don't leave them in your office. Don't allow your children to play with them. Store 'em under the bed, or put them in a room. And that's it," said Sarah Gonzalez.

She added that at the end of election day, her precinct convention was in a separate room, across a campus, forcing her to leave her equipment unguarded. "There's a lot of times when they are left alone, and I don't know how else to say that." Another precinct judge told me that at her class, they were prompted by the person running the meeting, "Where will you all be storing your election machines before the election?" The election workers answered, in a chorus, "Under our beds!"

6. **Are official software changes legitimate and clean?**

The idea of vetting several hundred thousand lines of code would be daunting even if the code were static. However, in the normal course of events, software is updated or "patched" to improve it or make fixes. Are new software patches properly vetted, and will they work? Or, will they fix one thing and break two others? If you're like me, you get official-looking emails "from" banks, Ebay, Paypal, and so on daily, but they're fakes. Could an official-looking patch arrive on the Election Chief's desk, but actually be from a hacker?

One example of reprogramming happened during the recount of the Ohio 2004 presidential ballots. A tiny company called Triad ran the software for about half of Ohio's counties. After the recount, Triad President Brett Rapp said that Triad had reprogrammed all their counties for the recount. For the recount, he said, "there has to be a change made to the tabulation reporting, to tell this reporting system: only report the presidential totals ... and we did this not just in Hocking County, this is in all of our counties."

[13] "How to Hack an Election in One Minute," by Daniel Turner, September 18, 2006, http://www.technologyreview.com/Infotech/17508

Two observers for the recount reported that Triad was "able to reprogram the computer to count only the Presidential ballots by remote dial-up" and Triad "had serviced the machine over the phone via modem on December 9th."[14]

But, rest assured, there's nothing to worry about; Triad says no one should worry about technicians changing anything in the software for elections, because the tech will leave a note inside the computer as to what was done.

7. **Are unexamined ballot definition files accurate and trustworthy?**

If you tried to vote for Senator Barbara Mikulski in the Democratic primary in 2004, you might have had a difficult time. Her contest was left off the ballot in at least three counties, according to voter complaints reported to the Senator. [15]

A little-known but crucial moving part in election software is the ballot definition, prepared relatively close to the time of an election. This is unique programming for each election, defining all the races and candidates for each precinct. Faulty ballot definition programming can thwart accurate electronic vote tabulation of DREs and optical scanners. "Every voting system includes a key component, called the ballot definition file (BDF), that is never subjected to an outside review. Given that BDFs determine the way votes are recorded and counted, the lack of independent oversight of these files is a major security vulnerability," writes Ellen Theisen.[16]

8. **Are there reasonable ways to make sure that the software that was previously checked is now the software running on each machine on the morning of election day?**

If someone changed the software, wouldn't it be detectable? In a word, no. Computer experts lament that it is incredibly difficult to determine what code is running on a particular machine on a particular day, i.e. election day. Even if it were relatively easy, would a jurisdiction be willing and able test each of thousands of machines just before they are sent out to precincts?

9. **Even if all eight conditions above seemed fine before the election, could there be intrusion DURING election day and during tabulation?**

Yes.

The entire system must have robust "burglar bars" to keep out intruders. That means strong password protection, strong authentication, encryption, and protection against hacking through any means, including during transmissions. Like water flowing around rocks, fraud will find its way

"Like water flowing around rocks, fraud will find its way around obstacles in a system."

[14] December 2004 taped interview with Triad President Brett Rapp and Triad Vice President Dwayne Rapp, by Evan Davis and Terri Taylor. Also, Ohio 2004 Green Party recount observer reports, http://web.archive.org/web/20041224044833/http://votecobb.org/recount/ohio_reports/counties/henry.php and http://web.archive.org/web/20050207042717/http://www.votecobb.org/recount/ohio_reports/counties/vanwert.php

[15] "The Vexations Of Voting Machines," By Viveca Novak, Time, April 26, 2004, http://www.time.com/time/magazine/printout/0,8816,1101040503-629410,00.html

[16] "Key Component of Voting System Undergoes No Review," by Ellen Theisen, June 18, 2006, http://www.votersunite.org/info/ballotprogrammingintro.asp. For a list of known vote switches, go to "Vote-Switching Software Provided by Vendors, A Partial List — 51 Ballot Programming Flaws Reported in the News; These were detected; how many were not?" June 2006, http://www.votersunite.org/info/mapVoteSwitch.pdf

around obstacles in a system. And, those ethically-challenged types who want your government run to suit them, not you, are surely scheming to hack the system.

Other attacks via computer are documented -- the Washington Post reported that IBM's global security intelligence team detected more than 237 million security attacks worldwide ... in six months. The online world is called 'a war zone.'[17]

Let's see how difficult Election Day intrusion might be.

Passwords - Diebold

Have passwords employed to protect election systems been robust? Evidence so far is that they are not. Diebold had the infamous password of 1111. Did an expert review force them to improve their password security? Only in that state, according to Pamela Smith.

Pamela Smith of San Diego County: "Once Diebold had been busted in Maryland by the RABA red team, they had to change at least their infamous [password] code of 1111. And so they did in Maryland. They didn't change it in California or anywhere else, which to me is just not good business practices."[18]

Passwords - ES&S

Diebold gets a lot of attention, but ES&S, the largest voting machine company in the nation, had its passwords examined recently. A team of experts (SAIT) who reported to the State of Florida weren't impressed.

SAIT: "Each of the other passwords mentioned above is fixed and hard-coded into the source code. They are the same for all [ES&S] iVotronic machines in the country, and likely to be known to every election official who manages elections on an iVotronic machine. They can never be changed, without changing the firmware on the iVotronic machine. This represents poor practice. ... Our judgment is that the password mechanisms on the iVotronic are poorly conceived and poorly implemented. The consequence is that the passwords by themselves do not do a good job of preventing unauthorized individuals from accessing critical system functions."

If an intruder wasn't good at guessing the password, they could bypass ALL passwords of the ES&S iVotronic by simply using a special type of Personalized Electronic Ballot - similar to a memory card, called a Factory Test PEB, or by impersonating one. "This undocumented backdoor poses a risk of unauthorized access to critical system functions."[19]

[17] "Hackers' attacks bewilder VeriSign," by Leslie Walker, Washington Post, Aug. 6, 2005, http://www.chron.com/CDA/archives/archive.mpl?id=2005_3893053

[18] Meeting, State of California, Secretary of State, Voting Systems and Procedures Panel, Sacramento, April 21, 2004, http://www.ss.ca.gov/elections/vsptranscript0421.pdf, p. 206

[19] "Software Review and Security Analysis of the ES&S iVotronic: 8.0.1.2 Voting Machine Firmware," Alec Yasinsac, David Wagner, Matt Bishop, Ted Baker, Breno de Medeiros, Gary Tyson, Michael Shamos, Mike Burmester, SAIT (Security and Assurance in Information Technology Laboratory), For the Florida Department of State, February 23, 2007, http://election.dos.state.fl.us/pdf/FinalAudRepSAIT.pdf

Encryption

David Jefferson was underwhelmed by the encryption used by Diebold, and told Diebold's Bob Urosevich so during a hearing.

David Jefferson: "What, in effect you did or your team did, is create a big complex building, put locks on every door, use the same key for every lock, and then published a picture of the key on the wall. Does this seem to be a suitable security architecture to you?"[20]

Strong encryption could be a vital protection during phone transmission of election results. And if not protected...? Here's what Dan Wallach has to say.

Dan Wallach: "A sophisticated adversary, e.g., an employee of the local phone company, could tap the phone line and intercept the communication."[21]

Switch to Partisan Server

Maybe you could skip figuring out passwords, if you could host real time election results on your own computer.

In Ohio during the pivotal 2004 election, the servers for the Ohio Secretary of State[22] reporting real time results on election night were "mirrored" or "hosted" or "diverted" to a group of SMARTech servers. The location was nestled among highly partisan entities. The 12-digit ISP address to which the Ohio Secretary of State site was diverted for the 2004 vote count falls between two ranges leased to the Republican National Committee. Hundreds of other Republican websites are hosted there. Servers handled by SMARTech also hosted countless emails from White House staffers that were recently in the news, sent outside the White House system, to such addresses as gwb43.com.

This Ohio election night server was not even in Ohio, but was located in Chattanooga, Tennessee. Why an official website with real time election results on election night would be hosted there has not been satisfactorily explained. EPluribusMedia.org and Free Press.org have been investigating, but a serious investigation would require subpoenas and securing evidence.[23]

Modem Transmission - Impersonate a Valid Jurisdiction to Transmit Rigged Results

Imagine you get a phone call, "Hi, I'm Fred from your bank. I just want to verify what your password is, ok?"

You would want to authenticate Fred, of course. But, the Diebold central tabulator (where the votes come in to be accumulated) didn't demand any authentication of what entity is calling with votes. The RABA hack demonstration discovered that a remote

[20] State of California, Secretary of State, Voting Systems and Procedures Panel, Sacramento, April 21, 2004, http://www.ss.ca.gov/elections/vsptranscript0421.pdf, p. 53

[21] "Analysis of an Electronic Voting System," http://avirubin.com/vote/analysis/index.html

[22] http://election.sos.state.oh.us

[23] A summary and links to the original investigative stories can be found at "The Pivotal Ohio Vote in 2004: Who Did the Counting?" by Josh Mitteldorf, April 29, 2007, OpEd News, http://www.opednews.com/articles/opedne_josh_mit_070429_the_pivotal_ohio_vot.htm

attacker could get complete control of the election. Special equipment needed? A laptop, and the right phone number. "Hello, central tabulator? I'm Fred, and I have votes for you to count," the computer could say to the other computer, talking in computereeze. The Diebold central tabulator doesn't care if Fred is Fred or Anna Nicole Smith's baby; it accepts the votes.

William Arbaugh of RABA: "We could have done anything we wanted to. We could change the ballots (before the election) or change the votes during the election."

The RABA testers could intercept votes being sent by modem to the server, changing the votes and sending on the new votes to the server (called a "man-in-the-middle" attack).[24]

Well, that was three years ago -- surely with testing and certification of voting equipment, everything is safe and secure now. They've worked all the so-called glitches out, right? Wrong!

Keys

How about keys? Well, if you've read all the way through this, your prize is coming up. Diebold was protecting elections with a hotel mini-bar key. (This is also known as the Leave No Comedian Behind Election Security Provision.)

"The access panel door on a Diebold AccuVote-TS voting machine - the door that protects the memory card that stores the votes, and is the main barrier to the injection of a virus - can be opened with a standard key that is widely available on the Internet. ... the exact same key is used widely in office furniture, electronic equipment, jukeboxes, and hotel minibars."[25]

Removable Vote Storage Device (Diebold Memory Card)

In 2005, Finnish computer expert Harri Hursti hacked the Leon County, Florida optical scan system in front of Supervisor of Elections Ion Sancho. It took Hursti a few minutes to change the result of a test election, and he never entered the room that had the tabulator in it -- he had reprogrammed the memory card in his hotel room. In the demonstration, later shown on the HBO documentary "Hacking Democracy,"[26] his new software was now in control of the election.

I asked Harri Hursti about the exploit.

Harri Hursti: "Fundamentally, the whole idea, and the discovery which I made from the publicly available documents, was that there is an executable program, which is living and stored in the removable media -- what we call the memory card. And that memory

[24] Review for the State of Maryland: Diebold AccuVote-TS Voting System, January 20, 2004, RABA Technologies, http://www.raba.com/press/TA_Report_AccuVote.pdf. and "E-Vote Still Flawed, Experts Say," by Kim Zetter, January 29, 2004, Wired, http://www.wired.com/politics/security/news/2004/01/62109

[25] "'Hotel Minibar' Keys Open Diebold Voting Machines," September 18, 2006, by Ed Felten, Freedom to Tinker, http://www.freedom-to-tinker.com/?p=1064

[26] HBO's "Hacking Democracy". http://www.hackingdemocracy.com; memory card photo used with permission.

card is really the modern day ballot box itself. So, while there was no indication in the user manuals or documentation that such a program is stored there, it was there. And it really means that there's no such thing as an empty ballot box. Well, the whole thing there is that that program is responsible for all the reporting functions of the optical scan count unit. Once you change that program, you can do a lot of other stuff. ... What's very important to understand is that there was no protection against random errors or intentional tampering to change or -- and replace the program in the memory card. It was there, just wide open. You could rewrite it -- write it over with your own program. And of course when you have your own program then there is a very far-reaching implication." [27]

My translation: it would be as if you marked a paper ballot, dropped it in a locked box, and there was a little ballot troll in the box, madly scribbling new votes on some ballots, and erasing votes on others. The troll could even use scissors and cut some of the contests off the ballots completely. If a person opened the box, the troll could become invisible.

Removable Vote Storage Device (ES&S PEBs)

A similar threat possibility was identified on ES&S DREs just a few months ago. Once again, the path of intrusion could be a removable vote storage device inserted into the larger machine. Once again, the successful attacker could completely control the machine and the results.

A tight Sarasota, Florida race for Congress, with 18,000 voters seeming not to have cast ballots in that race even though they voted for a lower profile hospital board contest, sparked unusual scrutiny of the electronic voting machines used.

The tally suggests that the race was decided by a margin of under 400 votes, but the 18,000 undervotes remain a gnawing question with no convincing reason for them. For those using the DREs in Sarasota, the undervote rate was three to seven times higher than the rate in neighboring counties voting for the same contest. The SAIT team reporting to The State of Florida wrote of the system, the ES&S iVotronic:[28]

"Our security analysis revealed several software defects that could allow an attacker to introduce a virus into the voting system that spreads through removable storage devices." [pp. 44]

The SAIT team found vulnerabilities called buffer overflows that could allow an attacker to take control of a voting machine by corrupting data on a PEB.

[27] Monitor, June 19, 2005, KPFT Radio Houston

[28] "Software Review and Security Analysis of the ES&S iVotronic: 8.0.1.2 Voting Machine Firmware," Alec Yasinsac, David Wagner, Matt Bishop, Ted Baker, Breno de Medeiros, Gary Tyson, Michael Shamos, Mike Burmester, SAIT (Security and Assurance in Information Technology Laboratory), For the Florida Department of State, February 23, 2007, http://election.dos.state.fl.us/pdf/FinalAudRepSAIT.pdf

SAIT: "Unfortunately, the testing procedures that are standard practice in the elections community are unlikely to discover these vulnerabilities or the presence of a virus. ... If these vulnerabilities were exploited, it would be possible to hide their existence. A cleverly constructed virus can cover its tracks so that infected machines could not be detected by ordinary means and an appropriately programmed virus could self-destruct and erase all its tracks. ... [If] carefully constructed, it can allow an attacker to transfer program control to her own malicious code. Once this happens, the attacker controls the machine." [pp. 37-38]

So, with the two biggest elections vendors, Diebold and ES&S, sneaky software could be injected into the voting system from a little device that could fit into the palm of your hand. The ballot definition file, which we discussed in #7 above, is part of the ES&S security vulnerability identified by SAIT. The PEB on this ES&S equipment contains the definition of the ballot. The fundamental architecture of the system automatically trusts the PEB and its data. It asks the PEB, "So, are you Fred?" and if the PEB says, "I'm Fred's evil brother and I bring you a contagious disease," the system responds, "Welcome!" If the PEB is infected, it could cause big trouble. (SAIT did not conclude that the election HAD been infected with this problem, nor did it rule it out.) [pp. 38-40]

In a separate study, computer scientists David Dill and Walter Mebane Jr. did a statistical analysis and found an association between PEB error messages and undervoting in the Sarasota congressional race (called CD-13 here):

"One particular error message ("Invalid vote PEB") is both directly associated with variations in the CD-13 undervote rate and related to differences in the relationship between the CD-13 undervote rate and the pattern of votes cast for the five statewide offices."[29]

The Sarasota case is a good example of secrecy in elections run amok. Even in the face of the glaring oddity of the Sarasota undervote, which drew national attention, officials tried to prevent the challenger from having experts examine the source code or the actual machines used.

Worm Attack

In the old black & white horror movies, a giant creature advances on the city and the women run screaming for their lives. These days, it's just a little worm, and it's Brad Friedman raising the alarm, with the computer scientists chiming in but in their trademark modulated tones.

Florida's Sarasota County (formerly represented in Congress by Katherine Harris) seems to have had a number of election anomalies last November. A Slammer worm attacked the ES&S voter registration server in Sarasota on the first day of early voting. It changed the system password and shut things down for two hours. The incident report includes this:

[29] "Factors Associated with the Excessive CD-13 Undervote in the 2006 General Election in Sarasota County, Florida," by Walter R. Mebane, Jr. and David L. Dill, January 24, 2007, http://www.votetrustusa.org/pdfs/Florida_Folder/smachines1.pdf, p. 5

"During investigation of the machine, it was found that the SQL database administrator password had been changed to an unknown password."[30]

Dan Wallach's response to the incident:

Dan Wallach: "... Most likely, the worm in question was not engineered specifically to attack the election, but rather was a general-purpose worm, of the sort that infect computers all over. Of course, if there's a security hole that the worm could use to get in, that very same hole would allow more specifically malicious people to get in via the same hole. "[31]

Bruce O'Dell's response to the incident:

Bruce O'Dell: "... Any malware that could rewrite an admin password could certainly install a rootkit, so I would certainly regard the compromised machine (or the entire subnet) as a goner unless it's been very well scrubbed indeed."[32]

"The technology to invisibly compromise voting systems is mature and the rewards are essentially limitless. It's professionally irresponsible to not presume vulnerable extreme-high-value systems are already actively being exploited."
-- Bruce O'Dell

What is a rootkit?

"A rootkit is any type of program - usually malicious - designed to hide its presence from the operating system. Rootkit technology can be used to prevent malicious software from being detected on a compromised system; it can even replace operating system functions themselves - so rootkits can do anything the operating system can do. It's not impossible that the rootkit approach could be applied to levels below the operating system - say, firmware or device driver ... even a perfectly clean operating system may be compromised invisibly. Worst of all, there are theoretical limitations to the ability to detect rootkits on a running system."

Bruce O'Dell continued:

"The technology to invisibly compromise voting systems is mature and the rewards are essentially limitless. It's professionally irresponsible to not presume vulnerable extreme-high-value systems are already actively being exploited."[33]

[30] "Worm attacked voter database in notorious Florida district," by Brad Friedman, Computer World, May 16, 2007, http://computerworld.com/action/article.do?command=viewArticleBasic&articleId=9019560&intsrc=hm_list

[31] Email to me, May 16, 2007

[32] Email to me, May 16, 2007

[33] Bruce O'Dell suggested for further reading, "A Deeper Look: Rebutting Shamos on e-Voting," http://www.verifiedvoting.org/downloads/shamos-rebuttal.pdf. He writes that Sections 3.1.1 "Cheating by Communications Devices," and 3.1.2 "Cheating by Malware Loader" alone are "sufficient cause for immediate abolition of voting by computer."

CONCLUSION

Of course, if everything about an election computer system were disclosed and vetted from head to toe, it would be safer. But, given what's at stake -- the reward for stealing an election could amount to control of a jurisdiction, or even the entire US treasury -- the threat level is quite high. The quality of software for elections to date has been unreliable and has not inspired confidence. And, experts admit that the task of protecting elections without some sort of paper ballot records is near impossible.

One of the experts for RABA spoke of the vulnerabilities of the Diebold DREs, which at the time had already been used statewide in Georgia:

William Arbaugh: "There's no security that's going to be 100 percent effective. But the level of effort was pretty low. A high school kid could do this. Right now, the bar is maybe 8th grade. You want to raise the bar to a well-funded adversary."[34]

Raise the bar? Tell ya what. Instead of shaking their secrets out of private election companies one by one, and exposing their insecure election systems mistake by mistake, let's get elections that can be overseen by average citizens.

Poll workers and citizens shouldn't have to know about rootkits and encryption keys and buffer overflows to protect our votes from wholesale theft with a few keystrokes. And, attaching printers or doing audits after the fact seem like a weak overlay onto a shaky, vulnerable electronic system. Sort of like putting leather seats into a car that doesn't run. Or maybe, more in keeping with gambling our democracy, it's like hanging a new pair of dice over the mirror of the junk car.

An election should be observable from start to finish, with human eyes unmediated by "help" from software. And human eyes should be able to tell if it's honest. Get it right on election night. Send everybody home convinced of the final result.

Computers can't do that. Paper ballots can.

[34] "Md. computer testers cast a vote: Election boxes easy to mess with," by Stephanie Desmon, January 30, 2004, Sun (Maryland), Archived at: http://www.votersunite.org/article.asp?id=1102

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Brief bios of some of the people quoted above.
(Their mention, of course, does not mean they agree with, or don't agree with, this article,)

David Dill is Professor of Computer Science at Stanford and founder of VerifiedVoting.org. He is the author of the "Resolution on Electronic Voting" which calls for a voter-verifiable audit trail on all voting equipment, and has served on the California Secretary of State's Ad Hoc Task Force on Touch-Screen voting.

David Jefferson has written computer software at Lawrence Livermore National Laboratory, and has been involved in election issues and procedures for a decade. He was on the California Secretary of State Internet voting task force in 1999, and has served on the California Voting Systems and Procedures Panel that makes recommendations to the Secretary of State about voting equipment.

Ellen Theisen is a software technical writer with 22 years experience. She is founder and co-director of Voters Unite, inventor of Vote-PAD, and author of numerous articles, including "Myth Breakers for Election Officials."

Dan Wallach is Associate Professor of Computer Science and in Electrical & Computer Engineering at Rice University. He studies computer security, distributed systems, and electronic voting systems. He co-authored "Analysis of an Electronic Voting System,"[35] the first independent look by computer scientists at the software of electronic voting.

William A. Arbaugh was a member of the RABA team that tested Diebold DREs for Maryland in 2004, and is an assistant computer science professor at the University of Maryland, College Park. His research includes information systems security and privacy with a focus on wireless networking, embedded systems, and configuration management.

Bruce O'Dell has spent his career working with very large-scale computer systems with stringent security, audit and accountability requirements - systems for financial accounting, insurance claims processing, mortgage origination, bond trading, stock trading, loan servicing, and online financial account aggregation. At American Express he was lead software architect for a project to create a company-wide security component, and received their Chairman's Award for Quality, in 1998, for helping to develop methods for securely deploying new software to networks of thousands of computers.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Pokey Anderson has broadcast or published numerous reports on voting machine issues over the past four years. She co-produces a weekly news and analysis radio program, The Monitor (www.TheMonitor.wordpress.com), on KPFT-Pacifica in Houston. A previous article was "Even a Remote Chance."[36] She has done research with a number of authors, contributing to a Nation cover story on elections by Ronnie Dugger, and providing extensive research for a book on Enron's collapse by Mimi Swartz with Sherron Watkins. Her email address is Pokey at kpft.org.

[35] "Analysis of an Electronic Voting System," Tadayoshi Kohno, Adam Stubblefield, Aviel D. Rubin, and Dan S. Wallach. July 23, 2003.http://avirubin.com/vote/analysis/index.html

[36] http://www.votersunite.org/info/evenaremotechance.asp