RSA Laboratories

Volume 7, No. 2, Fall 2004

CryptoBytes

CONTENTS

This issue of CryptoBytes concerns the security "electronic voting," a timely topic in this election year. Many voters wonder if electronic voting is really secure, or can be made secure. The advisability of voter-verified paper ballots is a hot topic of debate and experimentation; the appropriate use of cryptography is a mystery to many, and the potential utility of cryptographic receipts is just beginning to be explored. These three articles provide an introduction to these topics, and more. I'm sure that you'll find this issue of CryptoBytes highly stimulating and intriguing. These are excellent articles, and "must-reads" for those interested in secure voting technology. —Ron Rivest

I. Electronic Voting Systems—Is Brazil Ahead of its Time?

Pedro A.D. Rezende

ABSTRACT

The first article, by Professor Pedro Rezende of the University of Brasilia, describes the political context for the introduction of voter-verifiable paper ballots to their DRE (direct-record electronic, or touch-screen) voting machines for their 2002 elections. Rezende argues that many of the criticisms levelled against voter-verifiable paper ballots, such as the criticism that voter-verifiable paper ballots favor vote-selling, are just plain wrong.

II. Misassessment of Security in Computer-Based Election Systems

Douglas W. Jones

ABSTRACT

The second article, by Professor Douglas Jones of the University of Iowa, critiques what appears to be the current state-of-the-art in the application of cryptography to voting systems. He argues that not only are cryptographic security mechanisms frequently missing or misapplied by voting system designers, but that the voting system evaluation process is clearly flawed, based on the available evidence regarding the use of cryptography in current voting systems.

III. Secret Ballot Receipts: True Voter-Verifiable Elections

David Chaum

ABSTRACT

The final article, by David Chaum, describes his proposal for "secret-ballot receipts," where the voter is given a "receipt" for his cast ballot in the form of an encryption of the voter's choices. A clever procedure, involving "visual cryptography," provides assurance to the voter that his receipt is indeed an encryption of the voter's choices, and not of something else.

RSA LABORATORIES

Electronic Voting Systems
Is Brazil ahead of its time?

Pedro A. D. Rezende
Department of Computer Science - University of Brasilia

Abstract

We describe the limited deployment of verifiable voting electronic mechanisms in Brazil, along with the corresponding political and public reactions. In particular, we discuss how the use of such voting machines may be impacted by a long-held Brazilian tradition of corruption and electoral fraud. Our observations may prove valuable in the context that systems similar to that in Brazil are under consideration in several other countries with similar political climates.

1 Introduction

In May 2001, the president of the Brazilian Senate publicly admitted to spying on secret voting on the Senate floor [1] using an allegedly intentional back door in the electronic voting system used in senate. The resulting scandal – fueled by the fact that public elections are conducted using similar electronic devices – resulted in his resignation. It also set the climate for the approval of legislation requiring such devices to be made *voter-verifiable* (or *vv*), meaning that the voter can check that his vote was received and tallied [2]. This was done by the addition of a printer to the voting machines, which are known as *Direct Recording Electronic* (or *DRE*) machines. After the voter has input his choices, these would be printed on a slip of paper, and shown to the voter. In order not to simplify vote selling, this slip of paper is not given to the voter, but displayed behind a window. After the voter has approved the vote, it is counted, and the slip of paper transferred to a sealed bag.

The most prominent shortcoming of current *DRE* voting machines that lack printers is their inability to allow for recounts, as they do not not record individual votes, but only the sums per candidate per precinct. Therefore, apart from allowing the voter to verify that his vote was correctly received, the *vv* system would have the additional feature of allowing recounts using the paper slips. Recounts would allow for the detection of potential errors in precinct sums caused by malicious tweaking with the software; such modifications would otherwise not be detectable, given the ineffective auditing of the *DRE* software.

Sadly, Brazil has had a pattern of electoral fraud, and many political careers have benefited from being able to manipulate ballot boxes [7]. With its current electronic system and electoral process, fraud can be done invisibly by insiders and, while the system is generally believed to be secure, unsuspectedly as well. And even more sadly, general naiveness with technology may be contributing to harden that pattern. A current indication of such hardening can be traced to the well-respected Gallup Institute. Gallup performs polls worldwide, including election polls – but not election polls in Brazil. Huge disparities between competing polls, and between final polls and outcomes from an opaque electoral system believed to be reliable, may

explain: It is not beneficial to let one's reputation of being a reliable poller be muddied by notable, intriguing disparities. Note that if an unsuspected insider scheme to defraud an election ensues, or a set of them competes, scientific polls become unscientific if not aligned with the winning shceme.

With this in mind, it is not surprising that the *vv* system was met with fierce political resistance. While the *vv* system admittedly has technical shortcomings, it has been demonized and criticized beyond what many consider reasonable, often for reasons that are based on misconceptions, supported by administrative decisions appearing to be made to taint the image of *vv* in the eyes of the public. In the following, we will describe the *vv* system, its shortcomings and the criticism it has drawn. We will also describe and briefly analyze alternatives that have been proposed by its critics.

2 Voter Verifiable DREs

As mentioned, the *vv* system is based on adding a printer to each DRE machine, allowing the voter to inspect his vote before approving it and having it counted. The approved votes are entered into a sealed container (a plastic bag), allowing for later recounts.

What if the voter does not accept the printed vote? If a voter finds that he has entered the incorrect choices after seeing the printed paper slip, he may cancel the vote and start over again. Similarly, if a voter claims that the information on the screen diverges from the information on the slip (or either is different from the selection he made) then he can request that his vote be canceled and votes again. If the alleged mismatch persists, the entire precinct has to switch, from then on, to performing manual voting.

This way to deal with potential inconsistencies has, on one hand, been demanded by critics who later held it as an inherent inconsistency of electronic systems forced to turn voter-verifiable, while, on the other hand, exposed a peculiar double standard: before the *vv* measure, if the voter repeatedly complained of mismatch between the information keyed in and the information on the screen (whether this occurred or not), then he would have to accept whatever the screen says or give up his right to cast a vote. The justification was that since vote is secret, no one was allowed to verify his claim and/or suspend the use of the equipment upon such claim. Printer-screen inconsistencies are thus feared as a much wilder beast than keyboard-screen inconsistencies.

Do the printed slips favor vote selling? In a misinformed attempt to ban the use of the *vv* system, it was even argued in Congress that its use is dangerous in that it allows a voter to take the printed receipt of his vote to the candidate to whom he wishes to sell his vote [17, 18]. This is clearly not the case, since the voter never obtains the printed slip, but only gets to see it behind a window.

The irony here is that a simple way to sell votes remains, with or without the use of *vv*. Since the *DRE* voting software displays the name and a picture of the chosen candidate before the voter confirms his choice for that vote, and since this picture is from a file provided by the candidate to official authorities in charge of setting up the software, we have that a candidate could later show a voter a collection of different pictures of him or herself, one of which is identical to the one given to the electoral officials. This way, the candidate could pay voters able to pinpoint the correct picture. A savvy voter could, of course, select the candidate and then cancel his vote, thereby being able to recognize the picture without voting correspondingly – most voters, however, are not likely to do this.

Do the added printers cause difficulties? Before being banned, the *vv* measure was the subject of a "compromise" [3]. Given the public outcry from the earlier senatorial scandal, the *vv* system was to be employed, "on a trial basis", in 3% of the precincts in the October 2002 election in which Brazilians voted for president, state governor, two Senate and two House seats [4]. Interestingly, media attention after the election was not focused on analysis of the results of the election, not even on some strange mishaps, such as a momentary drop in the partial total of votes officially tallied for a presidential candidate, from over one million to minus forty one thousand [5, 6]. Rather, it covered the long lines at polling places, which were worse in those with *vv* add-on printers. Nevertheless, the likely reasons for these additional delays were never mentioned [8].

Among these reasons, we have that the election officials in charge of setting up the machines were not instructed to remove a "security" seal blocking the exit path of the slips of paper from the small add-on printer before sealing the bag onto it. As a result, the seal (which was explicitly specified in the printer supplier's contract) caused the printers to jam. Another reason is that voters were not told about the need to push the confirm key *twice* to have his vote approved *and* have the slip cut from the reel and moved to the sealed bag. Failing to do so caused the voting machines to time out after two minutes, requiring them to be reset using a tortuous menu path, and requiring the precinct official to enter a password. A third reason is that the number of voters registered by the electoral administration to vote at most *vv*-enabled precincts was increased beyond historic top levels. As a result of the "bad experience" with the *vv* system, Congress quickly voted, one year after this "trial", not to use it in future elections [10].

Were the auditing features employed? After the compromise allowing the "trial" of the *vv* measure in 3% of precincts, and before the 2002 election running it, there were several warnings by high-ranking officials from the electoral administration of the risks posed by such a mechanism for vote paper audit. Its functionality – to provide voter verifiability – was deemed as an "unnecessary" and "stupid" security measure which could taint the success of an otherwise flawless election [9]. Given the very tight margins (less than 0.2%) of one state gubernatorial runoff election, one for which pre-vote polls yelded up to 8% discrepancy, the losing candidate, relying on the "trial", appealed for a manual recount of the votes of the *vv*-enabled precincts. His appeal was dismissed by the local electoral tribunal, headed by an early critic of the *vv* measure [9], on the grounds that a manual recount from a non-mandatory mechanism "could put under suspicion [the electronic] elections nationwide" [11]. After all, they would argue, no one has yet been able to prove there have been any fraud in electronic votings in Brazil. The main question remaining: is no one able to prove fraud because the system is secure, or is the system secure because no one can prove fraud? In other words, secure for whom and against what? For layman voters against fraud, or for dishonest insiders against recounts?

3 Alternatives

Three alternatives to the *vv* measure has been brought forward by its critics. We will briefly describe these, along with their relative weaknesses.

Alternative 1: Parallel voting. The first alternative is called *parallel voting*; under this proposal, a sample of the voting machines that are to be used are replaced by backups, and a test is run on that sample during election hours. The test consists of running a "simulated election", in which a group of electoral officials enter votes on the selected machines as if they were individual voters, to verify that they operate correctly. This is done by checking that the machine output is

correctly generated for the votes cast by the officials, at the end of the voting period. The final simulated tally is then compared to the expected tally, this one run by anyone following the test. Any discrepancy can be detected, since the choices for the simulated votes are drawn and publicly known.

The main weakness of this proposal is that the conditions set up for the simulation would differ signifiacntly from the "real" conditions. Most notably, given the complexity of the routines for entering a simulated vote [13], this task is made much more time consuming than at the standard vote (as described in [15]). Therefore, if the *DRE* is controlled by a malicious piece of software, the test situation can then be detected and the *DRE*'s behavior affected. Note that although the times to key in different choices for a vote are indistinguishable, it is highly unlikely, in a real vote situation, that the votes be cast at the very low rate which is possible at simutation. Therefore, the software of the *DRE* can determine, from the number of votes entered by the end of the voting period, whether to run the correct tallying (if a test was detected) or to "cook the books" (if a real-election situation was detected) before it outputs.

Alternative 2: Software auditing. This naturally brings us to the second proposal, which is to have the software purported to run on the *DRE*s audited for correctness. Given the complexity of the software used and the difficulty of establishing *exactly* what a piece of software does (as evidenced by the continuous use of bloated commercial general purpose software), this is not likely to be a meaningful solution.

To make it worse, inspecting the system's code has proven to be a charade, with repeated promises – and rulings – to "open all the code" failing to materialize at the last moment, election after election [13, 14]. Even though auditing of the *complete* source code is required by law since 1997, only parts of such code has been offered for inspection. This is in spite of the most rigid non-disclosure agreement possible, allegedly due to "copyright protection issues". Moreover, even if this were not the case, practical circumstances come in the way of making this alternative a satisfactory solution. For one thing, there has been no way offered (or permitted) to verify that the audited code is the same as that which is used on election day, making such "code audit" a silly exercise. Besides the code of operating systems not having been included, the time and conditions allotted for inspection has been very far from sufficient, making it clear that this alternative is unconvincing except as a public relations stunt.

Alternative 3: Cryptography. A third and latest proposal has been to use additional electronics and/or software to generate and verify digital signatures on various portions of source and executable code, so that interested parties can verify that these are the components which are later compiled and used. The resulting executable, along with further signatures and verification software, would then be deployed to the hardware constituting the 450,000 *DRE* machines typically used in an election. During or after the deployment process local supervisors could then verify their party's signature on appropriate files – using the verification software deployed within the *DRE* software [14].

However, if the verification software is tweaked before deployment so as to not report errors, nothing would be gained by running it. Furthermore, even if the deployed verification software is working properly, the *DRE*'s operating system could have been tweaked to defeat its intended objective, namely by presenting the original file to the verification software, to later replace it by a hidden and rigged version. That is to say, the operating system (which was left out of the inspection set up by Alternative 2, as the reader may recall) can shelter code designed to override any of the security measures intended to be taken by this approach. In other words, if the short-circuited nature of this verification would not be enough to in-

validate alternative 3, the software to be verified has always included binaries untraceable by the 'auditing' permitted by alternative 2, yielding a cumulative process without any basis of trust on which to build.

At this point it is worth noticing the difficulty of tracing the origin of the money spent to develop and deploy such *DRE* system, let alone the possible strings that may come attached. Interested parties have not been able to either validate the workings of deployed *DRE* machines, nor have they been allowed to inspect contracts in due time. Some of these contracts have never been made public beyond their summary or first outsourcing link, despite being deemed public. The electoral administration was constitutionally set out in a way as to be its own judge, and the vast majority of voters and officials seem satisfied not only with such concentration of power, but also with the belief that technology works as panacea for negative human traits.

What are the benefits of these alternatives? The vulnerabilities of the three alternatives have been pointed out repeatedly to officials by various security experts. This leaves us with the question of whether the political support for these alternatives – and the resistance to the more straightforward *vv* approach – is grounded in incompetence or malice. We shall not attempt to address this question here.

4 Conclusion

Is Brazil, after all, ahead of its time regarding voting technology? Maybe.

It is understandable that voter verifiability measures tend to increase both the complexity of the system and the risk of malicious interference by individuals and organizations with rights to supervise election procedures. If not to affect the results, the risk tends to at least cast doubt on the result, something a sore loser may consider. This, however, should not be taken as reason to discard such measures from the outset. Rather, it shall be held as motivation to better research e-voting systems, given that verifiability is a technical price to pay for automation. Brazil's pioneer experience with e-voting evidences the flawed nature of simplistic reasoning, while giving plenty of indications that election security is a matter of balancing risks, conveniences and responsibilities.

News reports indicate that Paraguay, Argentina, Mexico, and other countries where corruption and election fraud are not just abstract concepts, may soon borrow or rent Brazil's system. In the United States, serious debate on the convenience and possible effects of legal measures enforcing voter verifiability in electronic systems is under way. We thus may soon see a number of countries facing the same questions that Brazil has been led to face over the last few years.

References

[1] National newspaper special report: "*Crise no Senado, Tempestade no Planalto*" O Estado de São Paulo, May 2001 http://www.estadao.com.br/ext/especiais/tempestade/tempestade.htm, accessed July 27, 2003.

[2] The 2002 Law implementing voter-verifiable measure: "*Lei 10.408/02" D.O.U. de 11.01.2002, http://www.brunazo.eng.br/voto-e/textos/lei10408.htm*, accessed July 27, 2003.

[3] Interview with Brazil's chief electoral official: "*Ilmar Franco entrevista Nelson Jobim*" O Globo, Rio de Janeiro, October 15, 2001 http://oglobo.globo.com/pais/1659118.htm.

[4] Note in National newspaper: "*CCJ aprova impressão do voto eletrônico*" Folha de São Paulo, pp. A9, October 29, 2001.

[5] National syndicated newspaper column: "*Coluna Jânio de Freitas*", Folha de São Paulo, October 8, 2002, São Paulo, SP.

[6] National syndicated newspaper column: "*Coluna Carlos Chagas*", Tribuna da Imprensa, October 10, 2002, Rio de Janeiro, RJ.

[7] Laerte Braga: "*A agência errada, o golpe é aqui, no Brasil*" http://www.crestani.hpg.com.br/2002C/laerte-braga.htm, accessed June 19, 2004.

[8] Local newspaper main stories on elections: "*Confusão Eletrônica*" pp.21, "*No Limite da Paciência*", pp.22, Correio Braziliense, October 7, 2002, Brasília, DF

[9] Interview with chief electoral official in Federal District: "*Fabrício Azevedo entrevista Lécio Rezende da Silva: Garantimos a lisura das eleições*" Jornal da Comunidade, August 4, 2002, pp.4, Brasília, DF.

[10] The 2003 electoral law bill: "*Câmara dos Deputados do Brasil - Proposição PL-1503/03, do Senado Federal*" Brazil's House of Representatives, National Congress http://www.camara.gov.br/sileg/Prop_Detalhe.asp?id=124899, accessed July 27, 2003.

[11] Editorial page from mainstream newspaper: "*Magela Recorre ao TSE para recontagem dos votos*" O Estado de São Paulo, November 19, 2002, São Paulo, SP http://www.estado.estadao.com.br/editorias/20-02/11/19/pol025.html, accessed July 30, 2003.

[12] Official web simulation of Brazil's electronic voting: http://www1.tse.gov.br/eleicoes/urna_eletronica/simulacao_votacao/urna.html accessed June 19, 2004.

[13] Official web site for Brazil's voting regulations: http://www1.tse.gov.br/servicos/resolucaoEm-Destaque/pesquisa.jsp accessed June 19, 2004.

[14] Brazil's Electoral Administration web newsroom: http://www1.tse.gov.br/servicos/informativo/-index.jsp accessed June 19, 2004.

[15] Report tracking Law 10.740/03's approval: "*Lei do voto virtual às cegas*" Forum do voto seguro, http://www.brunazo.eng.br/voto-e/textos/PLazeredo.htm, accessed May 10, 2004. Links doc.1 through doc.10 in report point to scanned versions of paper documents showing: a) trail the corresponding bill would have followed in Congress; b) its logical inconsistency.

[16] Public manifesto: "*Alerta contra a insegurança do sistema eleitoral informatizado*" Forum do voto seguro, http://www.votoseguro.com/alertaprofessores, accessed May 10, 2004, with 878 signataries.

[17] *Transcripts of House evening session of October 1st, 2003*: Brazil's House of Representatives, National Congress http://www.camara.gov.br/Internet/plenario/notas/ordinari/v011003.pdf, accessed May 10, 2004, pp.333-334: speech by the leader of Workers Democratic Party (PDT), Rep. Alceu Colares, denouncing: a) false representations by Rep. Moroni Torgan about the 2001 vv measure, which the bill there and then under vote would ban; b) the rigging of the House's internal electronic tracking system, on the path taken by said bill.

[18] Excerpt from *official video transcripts of Brazil's House of Representatives evening session of October 1st, 2003*. Forum do voto seguro www.brunazo.eng.br/voto-e/arquivos/collares1.rm [codec Real video 3.0, aprox. 3 min, 4.5Mb]:

Speech by PDT Leader (referenced in [17]), who frantically waves paper trail documents (scanned

and linked in [12]) which show a rig at the House's internal electronic tracking system, on the path taken by the bill there and then under vote, to an unamused and unresponsive House president (Rep. João Paulo Cunha) conducting the session.

www.brunazo.eng.br/voto-e/arquivos/collares2.rm [codec Real video 3.0, aprox. 7 min, 8.5Mb]: Complete speech.

The author

Pedro Antonio Dourado de Rezende is a tenured professor at Computer Science Department, University of Brasilia (UnB). ATC PhD in Applied Mathematics from University of California at Berkeley in 1983, heads the UnB Cryptography and Info Security Extension Program since 1997. Author of over one hundred articles on related topics, is a member of Brazil's Public Key Infrastructure Steering Committee since 2003, by appointment of President Lula da Silva to represent civil society.

This article is based on an earlier copylefted version published at www.cic.unb.br/docentes/pedro/trabs/election.htm.

Misassessment of Security in Computer-Based Election Systems

Douglas W. Jones
University of Iowa
Department of Computer Science
Iowa City, Iowa 52242
jones@cs.uiowa.edu

September 22, 2004

Abstract

When today's computer-based election systems use cryptographic technology, it is more likely to serve a cosmetic purpose, providing an illusion of security, than it is to actually secure anything. Example abuses range from simple "fairy dust" applications of cryptography to confusing encryption with authentication, encrypting the wrong data, poor key management, and related problems such as failure to understand the difference between random and pseudorandom. These demonstrate serious weaknesses not only among voting system vendors, but also in independent testing labs, security consultants, and government.

1 Introduction

When an election is conducted in a small group by a show of hands, security is not an issue. Everyone present can observe the entire process and determine the result for themselves. Security becomes an issue when the number of participants grows to the point that the voters cannot all vote in the same room, and it becomes an issue when secret ballots are introduced in order to protect the rights of voters who oppose the powerful or hold unpopular opinions.

When elections are distributed between many locations, we must secure the conveyance of data between these locations, not so much because of the possibility of eavesdropping, but because we need to assure ourselves that it is authentic. In fact, almost all of the data we are interested in conveying is public. The ballot layout is usually published weeks before the election and the totals from the precinct are usually posted in public when the polls close. The only actual secrets included with this data are authentication keys being distributed for later use.

Unfortunately, these elementary facts appear to be lost on many voting system developers, evaluators and customers. The following examples illustrate this.

2 Fairy Dust

In the summer of 1996, a subcontractor working for Wyle Laboratories of Huntsville, Alabama evaluated the software of the Electronic Ballot Station, an innovative new voting system made by I-Mark Systems of Omaha Nebraska. In the review of this software, the subccontractor reported that this was the best voting system software they had ever seen, and they were particularly impressed by its security and its use of DES [1].

This system was brought before the Iowa Board of Examiners for Voting Machines and Electronic Voting Systems on November 6, 1997 by Global Election Systems of McKinney Texas, which had renamed the system the AccuTouch EBS 100. At that examination, it quickly became apparent that the use of DES in this system was quite naive. The question that exposed this was simple: Given that DES is a symmetric key cypher, the security of the system depends crucially on how the key management and distribution problems are solved. So, how are they solved?

The answer from Global was disappointing but difficult to draw out: There was no key management or key distribution problem because there was only one key and it was hard coded into every copy of the system. In a prototype system, as a place-holder for future development, such a scheme might be appropriate, but such a primitive scheme should never have come to market. Unfortunately, this primitive security system remained in use until 2003, by which time, Diebold had purchased Global Election Systems [2].

Here, it is clear that cryptography was used as fairy dust. It was sufficient to fool the examiner for Wyle Labs into believing that the system was secure, where a more able examiner would have admitted an inability to evaluate the system's security instead of being impressed by a thin veneer of cryptography.

3 Incorrect use of Cryptography

What did the I-Mark system encrypt? As it turns out, encryption was used to guard the contents of the electronic ballot box during transfer from the electronic ballot station to the centralized election management system. This raises a second issue: This information is not secret. The best practice when closing the polls at a polling place is to print and post in public a copy of the election totals for that polling place before transfer of the electronic record to the election management system. This allows observers to verify that the data eventually published for that polling place matches the data disclosed before transmission.

If the data is already public, encryption must serve some other purpose. In this case, the intent is clearly to offer some degree of authentication. Unfortunately, simple encryption offers no authentication at all unless there is some redundant structure to the encrypted data. A compactly encoded binary file of election results would offer little assurance in this regard.

I-Mark, Global and Diebold were not alone in making this error. In the fall of 2003, the state of Ohio contracted with Compuware Corporation to evaluate four of the direct-recording electronic voting systems then on the market [3]. The Compuware report noted that the Election Systems and Software iVotronic system made no use of cryptography in data transfers from the voting machine to the election management system, it recommended that strong encryption be used, but it did not mention the need for authentication.

In fact, there is authentication in several of these voting systems, but it is accidental and weak. In the case of the Optech Eagle, sold by both Sequoia and Election Systems and Software, the data returned to the election management system includes the time at which the system was prepped for the election, and this is checked on receipt. While the time at which a system was prepped for the election is no secret, obtaining this information to the full precision of the hardware clock is difficult, so it represents a useful if weak authentication token, defending against forgery but not man-in-the-middle attacks [4].

4 How Strong is Strong Enough?

When the I-Mark/Global/Diebold AccuTouch system came under widespread public criticism in 2003, many considered the use of DES to be a significant

weakness [2]. This encryption standard, with only a 56-bit key, was never seen as very secure; designs for a brute-force DES cracker had been published in 1998 [5], and successful attacks were demonstrated shortly after that.

The use of cryptographically secure authentication to protect transmission of election data from precincts to election management systems is a specialized context, in which the basic assumptions under which DES was cracked may not apply. There are two ways in which an adversary may attack this transmission path in a voting system:

First, the adversary may attempt a man-in-the-middle attack, trying to crack the authentication, edit the vote totals and forge new authentication data for the edited totals. In jurisdictions where polling places transmit totals by public networks, for example, by telephone, there is usually a fairly short window during which the data must be transmitted, on the order of an hour. If data is hand-delivered, for example, in an electronic cartridge, the delivery window will be longer to allow for physical travel, but this does not give the adversary much more time for computation. Attacks that take many hours would be of no use here.

Second, the adversary may forgo cracking the authentication keys and attempt a trial-and-error attack, hoping to deliver an acceptable forgery before the authentic data is transmitted. Alternately, the trial-and-error strategy could be forced on a man-in-the-middle attack when a complete crack of the authentication keys is impossible. In either case, if even one bit of authentication information is wrong, the attack can be detected. All modern voting systems offer alternative channels that can be used when an attack is discovered, so trial-and-error is unlikely to pay off. In short, very weak authentication is sufficient if the attacker gets only one shot at a trial-and-error attack.

5 Is Pseudo-Random Random?

One critical requirement for any voting system used in the United States is that it protect the secrecy of the voter's ballot. The order in which voters enter a particular voting booth is no secret, any observer can record this. The ballots themselves are also only weakly guarded. In case of a recount, they may well become public record, as in Florida 2000. What must be broken is the link between voters and their ballots.

One way to break this link is to store the ballots in random order inside the voting machine. Unfortunately, what a naive programmer may believe to be random may be merely pseudorandom and quite predictable, to a cryptanalyst. Unfortunately, this fact is lost on many who advertise their services as security professionals.

For the I-Mark/Global/Diebold AccuTouch system, for example, a well-known and very weak linear congruential random number generator was used [2]. Unfortunately, when Compuware Corporation evaluated this same system, they concluded that this generator posed no risks [3]. Curiously, they did note that the pseudorandom number generators used for this purpose by ES&S and Sequoia were seeded from the real-time clock, showing some awareness of the limits of randomness.

Unfortunately, a brute-force exhaustive search through all possible 32-bit seeds is remarkably fast on a modern computer. Furthermore, the sample size, typically around 100 ballots per voting machine, is large enough that an exhaustive search may well be sufficient to reveal the seed that put the ballots into particular slots within the ballot box. As a result, simply seeding a weak pseudorandom number generator from the time of day clock may offer no real privacy.

Clearly, the strength and seeding of the pseudorandom number generators used for ballot storage should have been investigated by Compuware. It is not safe to

rely on the random number package that comes with whatever system or language is being used, nor to rely on default seeding of these generators. The only acceptable alternative to a carefully seeded cryptographically secure pseudorandom number generator is the use of additional carefully selected sources of randomness to bolster a weak generator.

6 Do We Need Public Keys?

Is symmetric key cryptography safe for use with voting machines, or do we need to build a public key infrastructure for elections? The central issue here is one of secure key distribution, and the answer rests on an understanding of how voting systems are used.

A typical jurisdiction has an elections office that includes a secure warehouse where all of the voting machines are stored between elections and where the election management system runs. Prior to the election, the election management system is used to prepare ballot information for each precinct and load this into the voting machines.

Some voting systems are loaded by physically connecting them to the election management system, one at a time in the secure warehouse. Others are loaded using PCMCIA cards or compact flash cards that are sealed into the system in the warehouse. Yet others are loaded at the polling place immediately before the polls open, using portable media hand delivered to precinct election officials.

So long as the voting systems are prepped for the election in the secure premises of the election warehouse and then securely delivered to the polling place, or so long as portable media are held in trustworthy hands, cryptographic keys can be delivered to the voting system through this route and there should be no need for more complex cryptographic models.

There are two thorny issues that must be addressed before accepting this argument. First, the custody issue must be addressed seriously. If authentication or cryptographic keys are loaded in a voting machine and then it is left unattended in an insecure location, someone might open the machine up and extract this information. Clearly, physical security is not obsolete.

The second concern is rising pressure from county election managers for faster ways to prepare machines for election day. This leads, naturally, to proposals for remote-control initialization and testing of voting machines using wireless technology. The security problems this could create verge on nightmarish, yet some vendors are proposing that their next-generation voting systems will operate this way.

7 Anti-Virus Tools?

It is clear that voting systems must be protected from viruses, and this is required by Section 6.4.2 of current voting system standards [6]. What is not so obvious is that protection from viruses or other malware injected into the system via network ports or removable media need not rest on the use of anti-virus software. Unfortunately, this has not been understood by many well-meaning security evaluators. For example, one assessor asked that Miami-Dade County install anti-virus software on the ES&S iVotronic voting machine [7].

The iVotronic does not run a commodity operating system, nor does it use data formats that are known vectors for virus distribution, although it does use an industry-standard data format for compact flash card directories. As such, no commercial anti-virus software is applicable and it is quite possible that the system is inherently virus-proof. Furthermore, anti-virus software can only detect known viruses, which is to say, it can only defend against second and third attacks, after the discovery of an initial successful attack.

Certainly, assessment of the security of voting systems against viruses and similar attacks is appropriate, but simply checking that the latest anti-virus tools are installed is not enough. Instead, the relevant questions are: Are the communication protocols used by this system inherently free of virus delivery mechanisms, and are they correctly implemented, for example, free of buffer overflow vulnerabilities?

If it can be shown that a communications channel cannot deliver data that will serve as input to an interpreter or be read as machine code, then that channel cannot be used to inject viruses or other malware into the system. This is the question that must be assessed on most embedded systems, and in general, the security offered by systems that meet this standard is higher than can possibly be met by installing and regularly updating anti-virus software. In fact, routine installation of antivirus software offers a path for Trojan horse attacks via that software, so it poses security risks of its own.

8 Conclusion

Unfortunately, these stories show that not only voting system vendors but also a significant number of voting system evaluators have seriously misunderstood the security requirements for voting systems. The presence of inept security in voting systems reflects badly on the vendors and on the level of sophistication of their customers, but after the publication of *Analysis of an Electronic Voting System* [2], this is not news.

What is more distressing is the extent to which the security evaluations that have been done for voting systems expose flaws in the knowledge of security professionals. It may not be too much of an exaggeration to state that many of today's security professionals have focused so much on conventional data processing applications using Microsoft Windows in a corporate setting that they are very poorly adapted to examining the security of novel applications outside the Windows domain or outside the commercial data processing domain.

References

[1] *Qualification Testing of the I-Mark Electronic Ballot Station*, Report number 45450-01, Wyle Laboratories, Huntsville AL, 1996, 336 pages. Note, this report is proprietary. Only content discussed in open meetings of the Iowa Board of Examiners for Voting Machines and Electronic Voting Systems is cited here.

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

[3] *Direct Recording Electronic (DRE) Technical Security Assessment Report*, Compuware Corporation, Columbus OH, Nov 2003.

[4] D. W. Jones, Problems with Voting Systems and the Applicable Standards, *Improving Voting Technologies – The Role of Standards* Serial No. 107-20, pages 154-191, U. S. House of Representatives Committee on Science, Washington DC, May 2001.

[5] Electronic Frontier Foundation, *Cracking DES*, O'Reilly, 1998.

[6] *Voting System Standards*, Federal Election Commission, Washington DC,April 2002.

[7] C. Jackson, *Audit Report – City of Opa-locka Special Election Held April 29, 2002*, Memo to D. Leahy, Miami-Dade County Elections Department Public Records, August 7, 2002.

Secret-Ballot Receipts: True Voter-Verifiable Elections

David Chaum
info@chaum.com

September 21, 2004

Abstract

A new kind of receipt sets a far higher standard of security by letting voters verify correctness of the election outcome – even if all election computers and records were to be compromised. The system preserves ballot secrecy, while improving access for voters, robustness, and adjudication, all at lower cost.

1 Introduction

Current electronic voting machines at polling places don't give receipts. Rather, they require prospective voters to trust them – without proof or confirming evidence – to correctly record each vote and include it in the final tally. Receipts could assure voters that their intended votes are counted. However, receipts have so far not been allowed because of the secret ballot principle, which forbids voters from taking anything out of the polling place that could be used to show others how they voted. The reason for this is to prevent schemes that could improperly influence voters, such as vote selling and various forms of coercion.

Introduced here is a fundamentally new kind of receipt. In the voting booth, the voter can see his or her choices clearly printed on the receipt. After taking it out of the booth, the voter can use it to ensure that the votes it contains are included correctly in the final tally. But, because the choices are safely encrypted before it is removed from the booth, the receipt cannot be used to show others how the voter voted.

The receipt system can be proven mathematically to ensure election integrity against whatever misbehaving machines or people might do to surreptitiously change votes. This level of integrity should enhance voter satisfaction and confidence and positively impact participation.

The system also eliminates the need for trusted voting machines, which typically use proprietary "black box" technologies. It can run with published code on standard PCs, allowing significantly lower cost. The receipts also improve robustness, currently achieved by costly proprietary hardware redundancy in storing and transporting votes, not only because failures can be detected at the polls in time to prevent lost votes, but also because the votes that receipts contain can be counted no matter what happens to the machines. Moreover, open-platform hardware, instead of needing to be stored in special warehouses most of the time, could even be used for various purposes year-round, for example in schools and libraries.

The inability of the current approach to reconcile secrecy and security needs has also led to functionality problems. The new US Federal requirement for provisional ballots – ballots cast by individuals whose names don't appear on the registration list – means

separate handling and counting, singling provisional ballots out for reduced privacy protection. Just as the system presented here can seamlessly include all such votes, it can lift the requirement that voters vote from their home precinct, ensuring access while improving convenience and turnout. (It even makes interjurisdiction voting workable.) Courts can also surgically add or remove the votes of particular fine-grained categories of voters; their inability to do so today forces them to either call revotes, throw out all ballots, or determine winners themselves.

1.1 Voting with the new approach

After being admitted to the voting booth, you vote using a so-called "touchscreen", as is becoming increasingly common. When all the candidates for an office are listed on the touchscreen, you vote for one by touching his or her name. That name is then highlighted while the other choices are dimmed.

With the new approach, the highlighted name you touched also appears by itself in large letters on a separate "printer/viewer" screen. You notice it in your peripheral vision and can easily see that it's displaying the same name you chose, thereby confirming your choice.

You can also choose to undo a vote, by touching once again the highlighted choice on the touchscreen, causing all the candidates to return to the original undimmed state. The canceling of votes is also confirmed on the printer/viewer screen, shown as the original vote clearly lined out.

Votes other than for a single candidate are also similarly handled, such as those for ballot questions, straight-party voting, prioritized and weighted votes, party symbols, and even write-in's input through displayed keyboards or even pen tablets. Once you make whatever kind of choice, a confirming summary image of it is displayed on the printer/viewer screen.

When you are done making your selections, you touch and confirm the "cast ballot" button. At this point two small rolls of paper appear lighted at the front of the printer/viewer, each behind its own transparent door. (The mechanics of the doors is such that opening either, like with a vending machine, locks the other door, preventing voters from opening both.) You are instructed to take one – either one of your own free choice. The paper roll you choose to take is your receipt.

(If you look inside the printer/viewer, through the clear window provided, you would see an ordinary thermal receipt printer that prints a pattern in three strips along the length of the paper and projection lenses superimposing images of the three strips onto the rear of the printer/viewer's projection screen. After projection, the strips of printing are separated by being physically slit apart. Two of the three strips are then rolled and held behind the transparent doors.)

1.2 Your receipt

Unrolling your receipt, you see it is an unreadable and seemingly random pattern of tiny squares. In fact, neither of the two rolls you were offered is readable on its own – the superimposition of the two, however, created a readable image when projected on the screen. The patterns on the two rolls can be thought of as "layers", as they are overlaid in viewing. (The third strip, to be explained later, is a final layer that serves only to make the images easier to read.) Your receipt then is one of two safely encrypted layers of the vote you saw in the booth on the printer/viewer.

The voting machine in the booth keeps an electronic version of this same final receipt until it sends it in for posting on the official election Web site. The bits representing the other layer, printed on the roll you did not take, are erased electronically. Thus, the only information about your vote that is retained is that printed on the physical layer you keep as your receipt

and, in the machine, a digital version of that same image. The roll you did not take (the third strip, which cannot be taken by voters) is kept just long enough to ensure it isn't needed to recover from a loss of digital data, before it's shredded.

1.3 Checking the vote

You can safely show your receipt to anyone, including political, governmental, public interest, or media organizations. Outside the polling place, for example, a group such as the League of Women Voters might offer to check your receipt. They simply scan it with a hand-held scanner and let you know immediately that it's authentic and correct (by subjecting the receipt's printed image and its coded data to a consistency check and later ensuring that it's correctly posted online when it should be, all as detailed later). An invalid receipt would irrefutably indicate incorrect operation of election equipment, although a second scanner could readily dispel a false alarm.

When the polls close, the polling place sends only the digital form of the receipts (not the erased layers or cleartext votes), electronically or by transport of, say, a CD.

1.4 Election Web site

If you wish, you can find the page on the official election Web site that includes your receipt by entering the receipt's serial number. You could then check that your vote was posted correctly – for example, by printing the posted receipt and matching it with your original receipt to check that they are identical. (You need not run consistency-checking software, because anyone can do this for all posted receipts, as discussed later.) You could also provide the original or its image by fax or photocopy to others for checking.

At some point after the polls close, the definitive set of receipts to be counted – the receipt batch – is posted on the Web site along with attesting signatures. The election's final output – the tally batch – is similarly posted. It contains the same number of items as the receipt batch, but each is a readable plaintext image of the ballot exactly as the voter saw it in the booth. (Using simple software, anyone can compute the totals from the tally images.) To protect privacy and ballot secrecy, the tally batches are in a random order, thereby hiding the correspondence between receipts and ballot images. To ensure that a one-to-one correspondence does in fact exist between the batches – that is, that no ballots were inserted, deleted, or changed – the system uses a kind of audit of a chain of intermediate batches between the receipt batch and tally batch.

After creating and publishing the intermediate batches, the system decrypts randomly chosen samples from them. These samples are chosen so as not to reveal enough to compromise privacy. They reveal enough, however, that checking them against the published batches effectively thereby checks that the correct one-to-one correspondence holds. Anyone can do this checking by running a simple, open-source program that they can download from any of multiple suppliers or even write themselves. The program can also check the consistency of each receipt batch entry. Such a suite of checks can convince anyone that the receipt batch correctly yielded the tally batch.

2 Receipt system

The system, a sketch of which is introduced in this section, is detailed in the Appendix 1 "More Formally", which in turn serves as a basis for Appendix 2 "Proof Sketch".

2.1 Properties

The receipt system ensures that several properties are met.

First, if your receipt is correctly posted, you can be sure (with acceptable probability) that your vote will be included correctly in the tally. A receipt that isn't properly posted is physical evidence of a failure of the election system, and a refusal by officials to post it is an irrefutable admission of a breakdown in the election process.

In addition, no one can decode your receipt or otherwise link it to your vote except by breaking the code or decrypting it using all the secret keys, each of which is held by a different trustee.

Even if all the election computers were compromised and running colluding malicious software (even having access to unlimited computing power), there are only three ways that a system could change a voter's correctly posted ballot without detection:

- It could print an incorrect layer, gambling that the voter will choose the other layer.
- It could use the same serial number for two different receipts, hoping the two voters choose the same layer.
- It could perform a tally process step incorrectly, taking the chance that the step will escape selection during audit.

For each ballot and with either approach, the chance that it would go undetected is one half. Thus, the chance that two ballots could be changed without detection of at least one is only a quarter, three ballots without a single detection an eighth, and so on. Changes in just 10 ballots will avoid any detection fewer than one in 1,000 times, and changes in 20 ballots will avoid detection fewer than one in 1,000,000 times.

In practice, many voters will not check that their receipts are posted or even have others check them. For example, in a large election if just 10 receipts are changed and only 5 percent of receipts are checked at random, the chance of detections is 50 percent. But in close elections in which a small number of ballots matter, a sufficiently high percentage of ballots would presumably be checked at least after the results were published. For example, if 100 votes would have changed the outcome in a large election, 5 percent of receipts checked would be enough to catch cheating all but one in 1,000 times.

2.2 Receipt encoding

What makes the optically combined layers readable and each of the two layers apparently random when separate is the mutual relationship of the patterns. The printing on both layers is divided into a grid of squares, or pixel locations. Each pixel location is either printed with the single color, or it is not printed, something like a random crossword puzzle without any text.

The earlier proposed and more intuitive superimposed printing uses two clear plastic sheets, each separately printed. Viewing the superimposition of the layers is simply achieved by overlaying the sheets one on top of the other and viewing through the combination. (This technique could be used by the printer/viewer in the booth, but printing on paper and using projection lenses as detailed later turns out to be more practical.) The single color of ink is used for printing on the sheets is translucent light gray. In a pixel location where both layers are printed, what you see looking through the laminate is then dark gray, because you're looking through two light gray zones; when both are unprinted, you're looking through clear plastic. But only when one layer is printed and the other is clear, is the result medium gray – the same medium gray no matter which of the two layers is on top. See Figure 1.

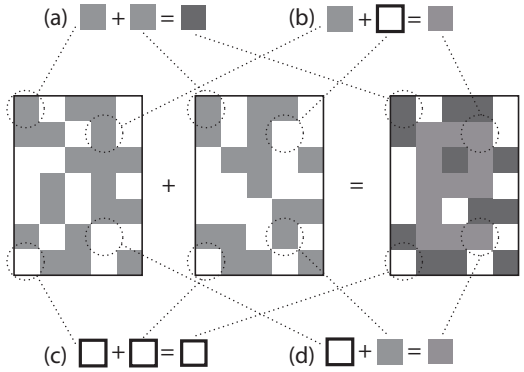

Figure 1: The letter “e” from two transparent layers. What the two layers on the left look like overlaid is shown on the right. Printing is 50% black ink, allowing half the light through and appearing light gray. There are four cases: (a) two light gray pixel values combine to a dark gray; (b) a light gray on the left and a clear on the other result in light gray; (c) clear on both layers results in clear; and (d) clear and light gray results in light gray.

This technique can be used to encode information on one plastic sheet so that only someone with a second correspondingly coded such sheet can read it, the application that Moni Naor and Adi Shamir first proposed a related technique for [1].1 It’s helpful to associate names with the two sheets: I’ll call the first “white” and the second “red” (but these colors have no more graphic significance than that you might tint the two translucent sheets to distinguish them). Each sheet is divided into a grid of pixel locations, and each pixel location is either printed light gray or unprinted clear. When the two sheets are laminated together, the grids line up exactly: each pixel location on one sheet has a paired pixel location at the same coordinates on the other sheet so the two are exactly one on top of the other.

Most current printing technologies print ordinary text by creating a grid of pixel locations in which some are printed with black ink while others get no ink. In the simple overlay system, the resulting image will look like medium gray letters on a background composed of randomly scattered dark gray pixels and clear pixels. (The third layer to be described, whether on clear plastic or paper, gets rid of the speckled background and can even give a color to the text.)

Consider how you can make two plastic sheets that yield text like this when laminated that cannot be deciphered from either sheet alone. For each pixel location, there are exactly two “pixel values” to work with: printed and unprinted. First you choose, totally at random, the pixel value for each pixel location on the white sheet. Thus you create the white sheet by inking some pixel locations and not others. Now to encode your message in the laminate, you simply choose the pixel values of the red sheet accordingly: If you want medium gray to make up text at a pixel location when laminated, you choose the pixel value for the red sheet so that it differs from that of the white sheet at that location; for a background location, you choose the red pixel value to be the same as the white pixel value.

When the receipt layers are laminated, the voter’s choices would thus be printed in medium gray on a speckled background (though preferably made neater by the third layer). This ballot image is the visible plaintext summary of the vote accepted by the voter.

Because the vote needs to be encoded in each layer, both layers need some red pixels (an all white layer is random and thus contains nothing of the vote). Interchanging two paired pixel values leaves the laminate visually unchanged, since the light still has to go through the same pixel values, just in a different order. So pairs in half of the pixel locations are swapped, which leaves the laminate outwardly appearing the same. The choice of which half of the pixel locations to swap and which to leave unchanged can be like the alternating pattern of a checkerboard. Each layer thus becomes, in terms of the way the pixel values are gen-

erated, a red and white checkerboard and because the text is large compared to the pixels, enough information about the vote is in each layer.

The system in effect uses the one-time pad coding technique to encrypt the ballot image. Claude Shannon proved this technique to be unbreakable, assuming the key is random [2]. The keys used – the values in the white pixel locations – aren't random but are believed to be indistinguishable in practice from random except to the set of trustees, who collectively guard ballot secrecy. Thus, if you have only your receipt layer and are staring at a particular white pixel on it, you learn nothing. Similarly, a red pixel value only tells you that the lamination would have been clear or dark gray if the paired white pixel matched the red pixel and medium gray if it didn't. But knowing nothing about which white pixel value was paired means you can't infer anything more about whether the combination was light gray or not.

The correction layer with transparent overlays simply cancels out the differences in the background and tints the text. Thus, where there is a double unprinted pixel, the correction layer has a medium gray printing (matching the appearance of two light gray layers); where there is a double light gray pixel, the correction layer is clear. And where there is text, the correction layer is, say, a translucent color.

When paper printing and projection lenses are used, the printing is black for the two potential receipt strips. The correction strip also has black printing but colored printing for the text (some thermal printers use lower heat to create a color on special paper and black for a higher heat). Since the correction layer is illuminated twice as brightly as the other two, a white on it contributes as much light to the screen as a white on each of them separately. Thus, the correction layer in addition to contributing color to the text, turns double black background locations into the same luminance as double white background locations. See Figure 2.

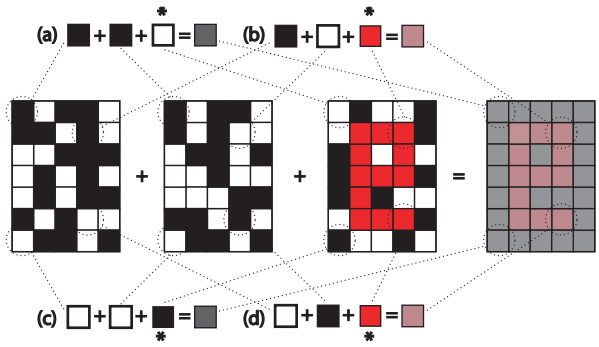

Figure 2: The letter "e" by superimposed projection from paper. The three paper strips are shown on the left and what they look like projected on the same screen is shown on the right. Printing is black except on the third strip, which receives twice as much light (indicated by "*") as the other two and additionally uses red colored ink. Again there are four cases: (a) black on the left and middle strip is turned to gray by double illuminated white 50% R,G&B on the third strip (b) black and white combine with double red yielding pink 75%R and 50%G&B; (c) two whites also make 50% R,G&B gray; and (d) same as (b) with order swapped.

Receipts should encode the votes exactly as the voter sees them. It's technically possible, however, that the still laminated printout shows one set of choices, but the receipt layer the voter takes encodes other choices. This could occur only if just one layer was invalid. If both layers are invalid, whichever layer the voter takes will fail checking and provide evidence of cheating by the system. If only one layer is invalid, and the voter doesn't select it, it won't be checked, just shredded. However, essential to security, as mentioned earlier, is that the voter chooses which layer to take only after the printer finishes printing the votes. Thus, a single invalid layer has essentially a 50-50 chance of being selected by the voter and caught.

3 Tabulating process

When the polls close, election officials and/or the courts should resolve any contested or provisional voting and then electronically post the receipts to be included in the tabulating process as the official definitive receipt batch. (A preliminary tally formed before contested and provisional ballots are included can, to protect the privacy of the provisional/contested votes, omit a random selection of ballots that will be included in the final tally.)

The tabulating process starts with an undisputed receipt batch and produces a final tally batch of ballot images. The first trustee produces the first intermediate batch from the receipt batch. The next trustee forms the second intermediate batch from the first intermediate batch, and so forth, until the last trustee forms the tally batch from the last intermediate batch. Figure 3 diagrams this process.

Trustees change the coding and the order of items from each batch in the chain to the next, thus ensuring privacy. Requiring each trustee to release some random link samples, establishing that items have been correctly transferred from batch to batch, ensures integrity.

3.1 Russian nesting dolls

A Russian nesting doll analogy can illustrate the processing of the input batch of receipts into the tally batch of ballot images. Each batch corresponds to a collection of dolls, each doll to an item in the batch. The receipt batch, for instance, is a collection of outermost "big" dolls, each with all its smaller dolls neatly nested within. The next batch, the first intermediary batch, is similar to the receipt batch but without the big dolls. This continues to the tally batch: the tiny solid wood innermost dolls. All batches have the same number of dolls, and within a batch the outermost dolls are all the same size.

The nesting dolls are like secret agents, each doll holding a unique random code sheet in its hands. The sheet is a grid of pixels printed using the two pixel values. Each doll is also physically locked with a combination lock that prevents access to the dolls within. A different secret combination, known only to a single corresponding trustee, unlocks all dolls of a particular size.

Consider the trustee with the secret combination for, say, the 10-inch dolls. To process an individual doll in the batch of 10-inch dolls, the trustee first unlocks the doll using the secret combination and removes its contents, a nine-inch doll. The trustee now has two code sheets, one from the 10-inch and one from the 9-inch doll. The trustee combines the two sheets to produce a new code sheet as follows: for every pixel location where light passing through is light gray, one pixel value is printed on the new sheet; everywhere the laminate is clear or dark gray, the other pixel value is printed. (When each of the two pixel values is considered to be a binary digit, 1 or 0, combining any number of sheets is simply adding the values modulo two.) The trustee places the combined code sheet in the hands of the 9-inch doll and destroys the empty 10-inch doll along with both old code sheets.

After likewise processing all the 10-inch dolls into 9-inch dolls with new code sheets, the trustee randomizes their order and outputs them as a batch. The trustee with the secret combination for the 9-inch dolls takes this batch as input, processes it into a batch of 8-inch dolls, and so on.

3.2 Coded sheets

A simple way to apply this process to an election starts by forming the sheet held by each big doll, differently, from all the sheets of the dolls nested within it. Suppose the original doll maker faithfully chooses sheets for all the dolls inside a big doll at random, but makes copies of all the sheets. Instead of keeping these copies on separate sheets, the doll maker com-

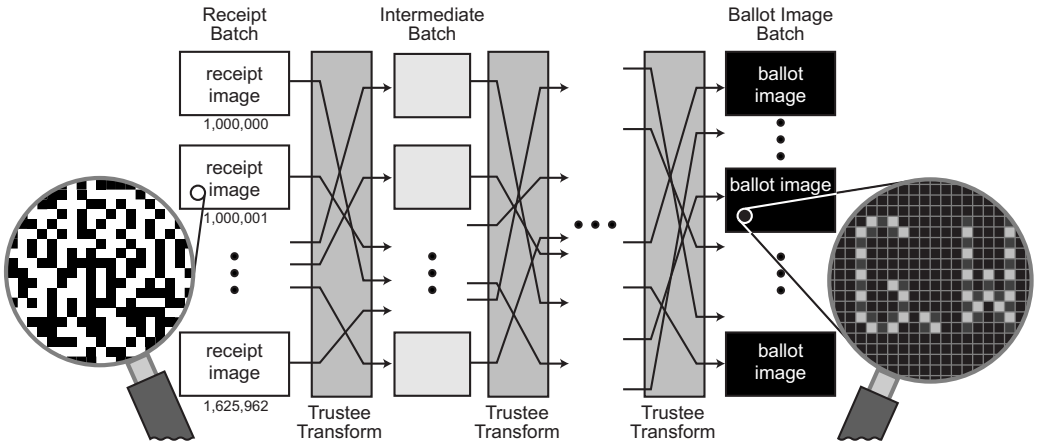

Figure 3: Overall tabulating process. Receipts pass through trustee-operated mixes, which transform them step-by-step into cleartext ballot images to be posted and tallied. Serial numbers and all but the red half of the pixel values are stripped off in forming the first intermediary batch. Mixes transform by removing a layer of encryption from each input and re-ordering the inputs in their output. (Vertical ellipses indicate batch items not shown; horizontal ellipses indicate additional trustees. Darker ballot image pixels are inferred from the lighter ones using redundancy in the font.)

bines them into a single sheet for the big doll, one pair of sheets at a time (or all at once using modulo-two addition). This is the "white" sheet for that big doll. Intuitively, it's formed by an initial "adding in" of all the inner sheets' coding, which will be "subtracted out" in stages as the dolls are processed.

Now suppose a voter has one of these big dolls and wants to use it to vote with privacy. The voter determines a red sheet that produces the desired ballot image when optically combined with the doll's white sheet (as previously explained). The voter then shreds the white sheet and gives the doll the red sheet to hold, placing the doll in the initial batch of big dolls. After processing by all the trustees, the final output batch contains the tiny solid wood dolls in random order, each holding a sheet that reveals a ballot image (which is easily seen by laminating with a sheet containing the same pixel value copied everywhere). All of the code sheets combined in the white sheet that influenced the red sheet have now been subtracted out.

To provide integrity, the system must be able to catch any trustee attempting to improperly change dolls or their sheets during processing. The solution entails requiring trustees to release complete and detailed audit trails of the processing (as videotapes, for example), but only for select dolls.

To allow trustees to release half of the complete set of tapes without compromising ballot secrecy, they each take on the role of processing more than one of the batches, say, two successive batches of the chain. This prevents tracing any tiny doll back to a big doll, even by a collusion of all but one trustee. After processing, a public lottery draw selects half the dolls in the trustees' first input batch, and the trustee releases their videos. Videos of these dolls' second processing wouldn't be revealed (because that would allow linking), but the second-batch videos of the other dolls are revealed. Figure 4 shows an example processing of two such batches by a trustee.

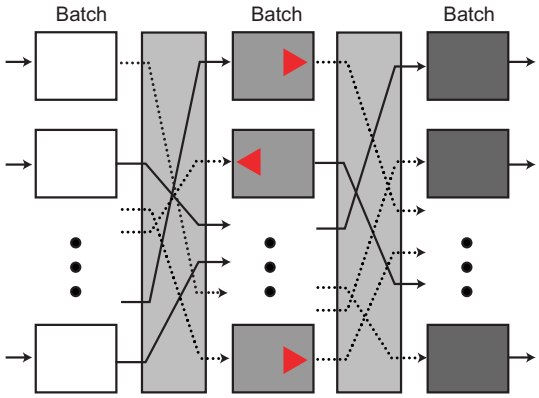

Figure 4: Batch processing by a single trustee. Triangles show the result of the draw and broken lines show links whose details are accordingly released in audit.

Exact tracing is thus prevented because trustees release only one video per doll for the two adjacent batches. Still, each time a trustee improperly forms a batch item there is a 50-percent chance of it being selected for release, so the odds of being caught stack up just as fast as with cheating by introducing a bad printed layer.

3.3 Encryption

Returning to the receipt system, the analogy's red and white sheets correspond, of course, to a ballot's red and white pixels (although without checkerboarding). The analog of a locked wooden doll is public-key encryption, in which anyone can encrypt a message using a published public key, but only the holder of the corresponding private key, the trustee, can decrypt it. Thus any voting machine can in effect be a doll maker and successively form the layers of a digital doll using published keys, but only trustees can strip off the respective layers. (Various known redundancy and key-sharing techniques can optionally provide resiliency in case some trustees don't participate.) With encryption as the mechanism, instead of a videotape, in effect only the code sheet originally held by the output doll must be released. (It's easy to check that applying the public key to the combination of this original sheet and the output doll results in the input doll.)

The initial printout in the voting booth actually uses two dolls. One of these is checked completely by being reconstructed from values printed on the last inch of the receipt and then not used further. The other doll and its checkerboard half of the red pixels create a "duo" that travels together through the chain of batches in the tally process. Such duos make up all batches. Trustees process each batch by removing a layer of encryption from the duo's doll and applying the revealed digital sheet to the duos pixels. By the time the duo reaches the tally batch, nothing is left of the dolls, and the pixels have become a readable plaintext ballot image.

(Dolls including error correction are printed on the layers in prescribed regions between lines of text. Since they should be identical on both layers, they should create uniform background around the votes whose absence would be easily noticeable to voters – ensuring that each layer has identical copies of both dolls.)

4 What codes to use?

Digital signatures are printed in the barcode on the last inch of the receipt layer. Such signatures have legal standing in many countries, and are considered irrefutable proof of the signed message's origin. A verifier outside the polling place can scan your receipt to immediately check, among other things, that it's valid, that an authorized voting station generated it, and that it correctly covers all the data printed. If the signature doesn't pass, the physical receipt is immediate evidence of system failure. If the receipt does check, however, it cannot be credibly denied a place in the definitive receipt batch.

Cryptographic techniques are classified as either *unconditionally secure* or *computationally secure*. The former, like the one-time pad with random key, cannot be broken, even if an adversary were to apply infinite computing power. The receipt system uses such unconditionally secure techniques to ensure, except with the probabilities of detection enforced, that integrity is not compromised.

Most cryptography in practice, however, is computationally secure – that is, in principle it is breakable if enough computing power is applied. No criminal has likely been able to make such computations using resources available today (because many systems, including international high-value wire transfer, that rely on such codes are still in place). Such standard cryptographic building blocks, which are also like those used widely by browsers when accessing secure Web sites, are enough (along with addition modulo two) to build the systems described here.

The receipt system uses computationally secure encryption to form the layers, which ultimately encrypt the data in receipts and batches, and thus protect privacy and ballot secrecy. After voting, the codes protecting receipts and posted batches, which are only readily linkable to ballot numbers and not people (apart from perhaps the case of provisional ballots), can easily be as good as those protecting comparable and much more identifiable, sensitive, and detailed data traveling on networks today.

Technical provision of privacy in voting is limited, however. Current surveillance technology means the confidentiality of what transpires in voting booths cannot in practice be held to any absolute standard. For example,

- Most US voter party affiliations are a matter of public record.
- The more a device helps a voter the harder it is to keep it from learning who they vote for (although, as in the system proposed here, devices need not be able to retain data between votes).
- Even the "gold standard" of voting systems – manual paper ballots – is subject to marking or ballot number recording and automatically captures fingerprints.
- Theoretical limits generally force a choice in cryptographic systems between unconditional integrity and unconditional privacy.

Thus the system presented here is arguably optimal. It protects privacy computationally according to current best practices by encrypting votes in receipts and published batches. And it protects the tally's integrity unconditionally by enforcing sufficient probabilities of detecting tampering.

This new type of receipt system reduces the cost of integrity while raising its level dramatically and making its assurance open to all interested parties. Robustness is similarly more cost-effective and raised to a level where it too can be ensured by voters (assuming they can access a functioning booth) through their receipts. Privacy and secret-ballot protections can easily meet current best practices and are arguably practically optimal. Improved functionality of the system facilitates accessibility and higher turnout, as well as needed improvements in adjudication. Perhaps most fundamentally, it can do a great deal to repair and improve voter confidence.

The hardware costs of these systems can be lower than current black box systems, which the government buys at many times the price of open-platform PCs. The cost of suitable printers in volume should be considerably less than the hardware cost saving. This doesn't even include savings in maintenance, upgrade flexibility, multiple uses, and reductions in outmoded security provisions. In fact, because of the provable integrity, federal dollars could be very well spent sponsoring development of such systems and making them available.

The Help America Vote Act was intended to fund the introduction of computers into almost all voting

booths in the US, and the systems that are deployed through this unprecedented funding will likely be in place for a long time. (There is also, for instance, a movement to automate Latin-American voting using the Brazilian model, which also includes computers in voting booths.) A growing grassroots movement is pushing to allow voters to see a printed summary of their vote, which is retained for possible recount. So far such summaries have not been shown to be effective or workable in general and have been replaced in Brazil. It does, however, indicate a growing level of public concern, and the two printing approaches could even be combined. The sad truth, however, is that the process of deciding which types of systems to deploy has so far been for the most part closed and informed neither by explicit performance requirements nor generally accepted security practices.

The receipt system presented here offers a new level of integrity, access, robustness, and adjudication, all at lower cost, that make it a compelling way to secure polling-place elections – and it should be the only way acceptable now

Acknowledgments It is a pleasure to acknowledge Ron Rivest, who served as a superb sounding board for ideas. The "WOTE" workshop was also very stimulating. Later, Jim Dolbear and Lori Weinstein provided a lot of help. Detailed comments from Josh Benaloh, Paul Craft, David Jefferson, Doug Jones, and Andreu Riera, as well as feedback from Jeremy Bryans, Dan Boneh's group, Stuart Haber, Robert Naegele, Peter Ryan, Marius Schilder, and Adi Shamir were also helpful.

References

[1] M. Naor, A. Shamir, Visual Cryptography, *Proc. Advances in Cryptology (Eurocrypt 94)*, A. De Santis, ed., LNCS 950, pages 1-12, Springer-Verlag, 1995.

[2] C.E. Shannon, Communication Theory of Secrecy Systems, *Bell System Technical J*, No. 28, pages 656-715, 1949.

[3] D. Chaum, Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms, *Comm. ACM,*, Vol. 24, No. 2, pages 84-88, 1981.

[4] M. Jakobsson, A. Juels, R.L. Rivest, Making Mix Nets Robust for Electronic Voting by Randomized Partial Checking, *Proc. Usenix Security 2002* Usenix Assoc. pages 339-353, 2002; also available as *IACR eprint 2002/025*.

David Chaum *is currently affiliated with several companies, universities, and international projects. Widely recognized as the inventor of electronic cash, he also originated a number of basic cryptographic techniques, general results, and techniques that allow individuals to protect their identity and related information in interactions with organizations. He has more than 50 original technical publications and 25 separate cryptography-related patent filings. Chaum has a PhD in computer science from the University of California, Berkeley. He has taught, led a crypto research group, and founded DigiCash and the International Association for Cryptologic Research (IACR). Contact him at info@chaum.com.*

Appendix: A more formal treatment

A complete system can be described somewhat more abstractly and formally, much as a typical cryptographic protocol: in terms first of what messages should be exchanged in what order, and then how the parties are to check what they receive. The receipt system has two separate phases: a voting phase and a tally phase.

Voting phase

The voting phase comprises a number of instances, each of which has up to six successive steps:

1. The prospective voter supplies a ballot image B.
2. The system responds by providing two 4-tuples: $\langle L^z, q, D^t, D^b \rangle$, ($L$ is layer, q is serial number, D is doll, and z is either t for top layer or b for bottom layer.) Each 4-tuple is printed on a separate layer.
3. The voter verifies (using the printing's optical properties) that $L^t \oplus L^b = B$ and that the last three components of the 4-tuple are identical on both layers.
4. The voter either aborts, and is assumed to do so if the optical verification fails, or selects the top layer $x = t$ or the bottom layer $x = b$.
5. The system makes two digital signatures and provides them as 2-tuple $\langle s^x(q), o^x(L^x, q, D^t, D^b, s^x(q)) \rangle$ ("s" is seed and "o" is overall)
6. The voter (or a designate) performs a consistency check to ensure that the digital signatures of the 2-tuple check, using agreed public inverses of the system's private signature functions s^x and o^x, with the unsigned version of the corresponding values of the selected 4-tuple (as printed) on the selected layer, and that $s^x(q)$ correctly determines D^x and the half of the elements of L^x that it should determine.

More particularly, let the relationships between the elements of the 4-tuples and the 2-tuple be as follows: The red bits R^z and white bits W^z (both m by $n/2$ where n is even) determine the m by n binary matrices L^z in a way that depends on whether $z = t$ or $z = b$: $L^t_{i,2j-(i \bmod 2)} = R^t_{i,j}$, $L^t_{i,2j-(i+1 \bmod 2)} = W^t_{i,j}$, $L^b_{i,2j-(i+1 \bmod 2)} = R^b_{i,j}$, $L^b_{i,2j-(i \bmod 2)} = W^b_{i,j}$, where $1 \le i \le m$ and $1 \le j \le n/2$.The ballot image and the paired white bits of the opposite layer y determine the red bits: $R^x \oplus W^y = B^x$.

The cryptographic pseudo-random sequence functions h and h' (whose composition yields binary sequences of length $mn/2$) determine the white bits from the signature on the serial number as follows: $W^z_{i,j} = (d^z_k \oplus d^z_{k-1} \oplus \cdots \oplus d^z_1)_{(mj-m)+i}$, where $d^z_l{}' = h(s^z(q), l)$ and $d^z_l = h'(d'^z_l)$. The d'^z_l also forms the "dolls" using the public-key encryption function e_l, whose inverse is known to one of the trustees: $D^z_l = e_l(d'^z_l \cdots e_2(d'^z_2, (e_1(d'^z_1))$, where $1 \le l \le k$ and, for convenience, $D^z = D^z_k$. (Separate h and h' are for improved efficiency with large ballots.)

Tally phase

The tally phase takes its input batch from the outputs of an agreed-on subset of voting instances reaching step 6. For each such instance, only half of L^x and all of D^y are included in the tally input batch, consisting of the duo $B^x_k = R^x$, $D^y = D^y_k$, which can be written as B_k, D_k. A series of k mix operations [3] transforms each such duo into a corresponding ballot image B^z. The lth mix transforms each duo B_l, D_l in its input batch into a corresponding B_{l-1}, D_{l-1} duo in its lexicographically ordered output batch by decrypting D_l using its secret decryption key corresponding to e_l, extracting d'_l from the resulting plaintext, applying h', and finally applying $B_{l-1} = d_l \oplus B_l$. The kth mix performs the same operation on each duo, and because D_0 is empty, the result is $B_0 = B^z$.

Prior arrangement partitions the k mixes into contiguous sequences of four among a set of $k/4$ trustees, where k is divisible by four. For simplicity, assume that the input batch size is also divisible by four. When the mixing is complete, half the tuples in each batch are selected for opening. The work of Markus Jakobsson, Ari Juels, and Ronald Rivest [4] inspired this approach. A random public draw, such as that used for

state lotteries, ensures that these choices are independent and uniformly distributed. The tuples selected for opening depend on the order in each trustee's four mixes:

- In the first mix, half of all tuples are opened.
- In the second, the tuples not pointed to by those opened in the first mix are opened.
- In the third, half the tuples pointed to by those opened in the second mix and half the tuples not pointed to are opened.
- For the fourth mix, as with the second, those tuples not pointed to by the previous mix are opened.

Figure 5 illustrates this process.

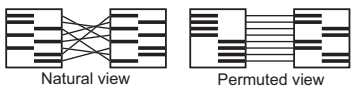

Figure 5: A trustee's four mixes of eight pairs. Tuples are opened according to the order of these mixes.

A few extensions are worth noting at this point. For improved privacy, multiple doll pairs allow separate ballot images per contest and/or question. Also, to prevent a voter's choice of layer, which is revealed to the poll workers, from determining the ballot image type, and to prevent bias in voter preference for particular layers, the dolls can determine a mapping between the physical layers and a pair of symbols that the voter chooses between. The symbols are printed before layer selection in a way that hides them until after the layers are separated.

Proof Sketches

The properties asserted informally in the text can be abstracted and stated more precisely in terms of the more formal description provided. Without implying any particular level of rigor, explanations for these statements can be illustrated in terms of the familiar format of theorems and proof sketches.

Theorem 1 If, for a selected and an unselected 4-tuple from an instance of step 2 in the voting process, the selected 4-tuple satisfies the consistency check in step 6 and there is a 2-tuple that would satisfy such a check with the unselected 4-tuple, the doll of the unselected layer, as printed on the selected layer, is correctly formed and determines all white pixels printed on the unselected layer (relative to which the voter sees the vote in the receipt's red bits).

Proof (sketch): The serial number q and the doll D^y are printed on both layers identically, as the voter verifies in step 3. The doll D^y in the unselected layer's 2-tuple is correctly determined by q, according to the functions s^y, h, and e, because the unselected 4-tuple would satisfy the consistency check in the hypothetical step 6. Similarly, q correctly determines the white bits W^y according to s^y, h, and h' that the voter checks in the hypothetical step 6 as being correctly printed on the unselected layer. Because the encryption e is bijective, Dy determines the $d_i'^y$, which determines W^y. Thus, the D^y printed on the receipt determines the W^y printed on the unselected layer.

Theorem 2 Any properly formed, selected layer and its resulting processing reveal the ballot images only in encrypted form until they appear in the tally batch.

Proof (sketch): Of the selected layer's six components $\langle L^x, q, D^t, D^b, s^x(q), o^x(L^x, q, D^t, D^b, s^z(q))\rangle$, only the first depends on the ballot image B. The L^x bits are partitioned among the R^x bits that depend

on B, and the W^x that don't. The W_i^x are each encrypted by e_i and can therefore be ignored. Each B_l, $1 \leq l \leq k$, appears in its respective input batch summed modulo 2 with each d_p, $l \leq p < k$. Thus, each time any B appears in an input batch it appears $\oplus$ed with a distinct pseudorandom value that only appears in all following sums. The resulting set of linear equations thus cannot be solved for any B.

Theorem 3 For any trustee's mixes, a duo's prescribed opening doesn't reveal a restriction on the correspondence between any individual input and output.

Proof (sketch): It's easy to see that the restriction imposed by an odd-numbered batch followed by an even-numbered batch – a doubleton of batches – requires that each of the two known halves of the inputs results in a respective known half of the outputs. (Note, however, that this could reveal something about an individual input and output, such as whether the input could correspond to a particular unique output.) A next doubleton that exactly splits each output partition of its predecessor across its own input partitions enforces the restriction that exactly half the members of an input partition are in each output partition, but leaves any particular input to the two doubletons free to be any particular output.

Theorem 4 The probability that a trustee that improperly forms u distinct duos in any of its output batches will be detected in at least one duo is $1 - 2^{-u}$.

Proof (sketch): The random draw selects the duos to be opened in a trustee's first batch independently of the trustee's control; an opened duo is either correct or not. The probability of detection is thus 50 percent for each improperly formed duo in the batch. Because the opened values are all correct, the half chosen for the next batch is selected independent of any improperly formed duo, and so on inductively.

ABOUT RSA LABORATORIES

An academic environment within a commercial organization, RSA Laboratories is the research center of RSA Security Inc., the company founded by the inventors of the RSA public-key cryptosystem. Through its research program, standards development, and educational activities, RSA Laboratories provides state-of-the-art expertise in cryptography and security technology for the benefit of RSA Security and its customers.

Please see www.rsasecurity.com/rsalabs for more information.

NEWSLETTER AVAILABILITY AND CONTACT INFORMATION

CryptoBytes is a free publication and all issues, both current and previous, are available at www.rsasecurity.com/rsalabs/cryptobytes. While print copies may occasionally be distributed, publication is primarily electronic.

For more information, please contact:

cryptobytes-editor@rsasecurity.com.

RSA SECURITY®

RSA Security Inc.
www.rsasecurity.com

RSA Security Ireland Limited
www.rsasecurity.ie

CRYPTOBYTES VOLUME 7, NO. 2, 2004