Comments on the Report
“e-Voting Security Study”
Written by the
Communications-Electronics Security Group

Andreu Riera

CEO

Scytl Online World Security, S.A.

Barcelona, Spain

28 August 2002

Table of Contents

1. Executive Summary

Electronic voting is a very promising development that may serve to enhance citizens' participation in the democratic process. Nevertheless, in order to ensure the successful adoption of e-voting, security and privacy issues must be adequately addressed and solved.

A sound e-voting cryptographic protocol must lie at the core of any security solution for Remote Electronic Voting (REV) systems. Only mixing-based protocols offer the optimal balance between security and practicality.

In our opinion, both the "key principle" (paragraph 59) and the security profile set out in Annex A of the CESG's report can be strengthened to better meet the specific concerns of an electronic election. Once strengthened, we believe that Annex A could be a good security profile to be adopted for securing further pilots.

We do agree with CESG's assertion that none of the current security technologies are able to meet all the requirements implied by the key principle, the security profile, and recommendations 1 to 5. In spite of that, we present a security protocol that manages to meet almost all the requirements (including the strengthened Annex A). We believe that it would be highly advisable to pilot this protocol in 2003.

In our opinion, the approach presented in Annex C of the report sacrifices too much security and user-friendliness in order to allow the casting of ballots from untrusted client devices and/or devices that lack computational power. As a consequence, considerable trust must be placed in certain back-end elements of the voting system, which inevitably exposes the system to internal attackers with administrative privileges.

2. Introduction

E-voting is very promising but it poses many technological and socio-political challenges [RST02]. Among the technological challenges, security and privacy are regarded to be the most significant, and solving them is of paramount importance to achieving a smooth transition from conventional voting methods to e-voting.

Many digital security measures already exist in the market that can be applied to REV systems. Even though some of these security measures – such as firewalls, transport-level encryption, intrusion detection systems, etc. – are mature and can be reliable if well administered, they are too generic and therefore do not cope well with the specific security requirements posed by a voting application. Indeed, an e-voting system by its very nature breeds a list of security concerns that are not adequately addressed by current off-the-shelf security technologies. Moreover, these security technologies are focused on preventing attacks from external intruders, leaving systems exposed to the possibility of attacks that come from the inside (which are much more dangerous given the privileged position of the attackers).

The ability to vote electronically is currently seen with great interest, and so today e-voting has the opportunity to be widely adopted in the near future. However, if something goes wrong with the security (the integrity of results, privacy of voters, etc.) during a major e-voting event or pilot, this could destroy the public's confidence and undermine all current work to introduce e-voting to society. This is actually a very daunting prospect, since as e-voting gains importance it will become a more attractive target for attackers of all kinds. Current generic digital security measures might provide a false sense of security that can lead to catastrophic outcomes – not only to a pilot or a general election specifically, but also to the future of e-voting generally.

Through these written comments we will outline our view with regard to e-voting security, arguing that a sound e-voting cryptographic protocol is necessary to achieve a reasonable level of security in REV systems. An e-voting cryptographic protocol provides protection for all parties involved in the voting application, even in the event that privileged parties collude to attack the system. We will list the security concerns that can be addressed by means of an e-voting cryptographic protocol. In accordance with our position, we will comment on the specific issues requested, and propose some additional security requirements to complement the list given in the CESG's report.

At the end of our comments, we will provide an overview of Pnyx, an e-voting cryptographic protocol developed by Scytl, that is based on eight years of academic research. Pnyx fully satisfies the security requirements needed to conduct fair and accurate electronic elections, while maintaining a low level of complexity that permits a thin-client implementation.

3. e-Voting Security

3.1. How e-Voting is Different

Conducting an electronic election by means of a REV system is a complex issue that raises a whole list of specific security requirements [CC96,Rie99] in addition to those security requirements that are common to many other online applications (e.g. system availability). The security requirements specific to e-voting (e.g. the privacy of voters) cannot be guaranteed through generic off-the-shelf security measures but need special-purpose security measures. Furthermore, given the sensitive and inclusive nature of voting (i.e. those in charge of running the system are also voters and/or candidates and have an interest in the outcome of the election), the system must be protected from both outsider and insider attacks. In summary, the main aspects that differentiate e-voting security from that of other common online applications are:

- E-voting entails a set of advanced security requirements which demand special-purpose security measures
- Any e-voting system must be protected from insider attacks

3.2. Application-Level Cryptographic Protocols

Due to the above problems, a REV system must be powered by an application-level cryptographic voting protocol[1]. Such a protocol acts as the central pivot for those security measures that are focused specifically on e-voting. Because the protocol lies at the application layer, it can fully understand and address the needs of the voting application. Indeed, any security measure below the application layer such as transport-level encryption would fall short of solving the security problems that are *specific* to e-voting.

An application-level cryptographic voting protocol is designed to protect the interests of all of the parties involved in the election, even when confronted by the malicious collusions of other parties, or by insider attacks. A cryptographic voting protocol prescribes the steps and actions to be followed in casting a ballot remotely, both by the voter's device and by the corresponding vote collection server. The protocol also determines the cryptographic actions that must be done to open a digital ballot box to tabulate the ballots, and also to verify the election results. Naturally, a cryptographic voting protocol should be complemented with generic digital security measures (firewalls, data-transport encryption, protection against denial of service attacks, etc.). Nevertheless, the voting protocol is arguably the most essential technological component to allow voting over a communication network to achieve the same levels of protection as conventional voting systems. A cryptographic voting protocol is therefore an essential part of any REV system.

A custom-designed application-level cryptographic protocol is an essential part of REV Systems.

[1] A cryptographic protocol is a sequence of steps performed by two or more peers involving cryptographic operations and the exchange of messages between them. An application-level cryptographic voting protocol is a cryptographic protocol that runs at the upper-layer (the application level) of the communication architecture, and that is specifically designed to address the security requirements of a voting application.

3.3. Security Requirements Addressed by Voting Protocols

Much research has been done on cryptographic voting protocols during the last two decades, resulting in many voting protocol variants. Despite the variety, most of the protocols tend to meet the same set of security requirements. The **privacy of voters** is frequently regarded as the central security requirement. Indeed no one (not even election authorities or system administrators in charge of vote collection servers) must be able under normal circumstances to correlate the votes to the voters who have cast them. However, this somewhat contradicts the obligation to adequately identify voters using **strong authentication** means in order to prevent non-eligible voters from voting, and to prevent eligible voters from voting more than once. The **accuracy of election results** is also essential. It must be impossible to add invalid ballots (e.g. a fraudster voting in place of abstaining voters), or to delete or alter valid ballots. Another requirement addressed by voting protocols is to **keep intermediate results secret** until the election is completed (unless required by the specific nature of the election). The objective of such secrecy is to prevent leaked partial election results from influencing individuals who have not yet voted. Another important security requirement considered by voting protocols is **verifiability**. To gain voters' confidence in the system, it is important to provide them with the ability to verify that their votes have been treated correctly. The verifiability method must leave no room for any doubt as to the vote's treatment. In addition, in the case of the detection of any problem, a voter must be able to irrefutably prove (without fear of compromising his/her privacy) that his/her vote was not properly treated. At the same time, the voting protocol must ensure that the capability to verify one's own vote **does not expose voters to coercion** by third parties and **does not allow them to sell their votes**. Table 1 summarizes the list of security requirements that are commonly addressed by cryptographic voting protocols.

Privacy: Impossible to correlate votes with the corresponding voters.
Authentication of voters: To ensure that only eligible voters can vote and only one vote per voter is counted.
Accuracy: Valid votes cannot be removed or manipulated. No invalid votes can be added.
Secrecy of intermediate results: All results are kept secret until the election is completed.
No-coercion: The system must not enable the selling of votes or the

coercion of voters.
Verifiability: Voters must be assured of the correct treatment of their votes and have means to irrefutably prove any fraud.

Table 1: Security requirements generally addressed by cryptographic voting protocols

A couple of short notes on the last two security requirements:

Two clearly different types of coercion must be distinguished. *Individual* coercion or vote selling is the straightforward consequence of having voters casting ballots from unattended locations using REV systems. *Massive* coercion or vote selling does not require looking over the voters' shoulders but it is instead based on the processing of *voting receipts* (which are generally generated during a voting protocol). Each type of coercion requires different protection techniques.

Two different levels of verifiability may be considered. A first level of verifiability allows voters to check if their votes have been really delivered unaltered to the authorities that are in charge of the ballot tabulation process (as what happens in current paper-based polling-place voting systems; note however that this is not the case for postal voting). A second level of verifiability allows voters to actually check the presence of their own vote on the published results. This second level of verifiability goes much further than conventional voting methods. Nonetheless, if offered, it would require extraordinary effort to prevent massive coercion or vote selling.

3.4. Taxonomy of Cryptographic Voting Protocols

Mixing

Chaum proposed the earliest cryptographic voting protocol in 1981 [Cha81]. Many other proposals followed, coming up with a numerous group of so-called mixing-based voting protocols. The fundamental principle of these mixing-based voting protocols is to send ballots blindly validated to the vote collection servers through some sort of anonymising channel that serves the purpose of severing the link between the voter's identity and their vote. Representative mixing-based voting protocols were proposed by Fujioka et al. [FOO92] and Park et al. [PIK93].

Homomorphic Encryption

In 1986 Benaloh et al. started a second group of cryptographic voting protocols based on homomorphic encryption techniques [BY86]. These voting protocols avoided the use of anonymising channels by splitting individual ballots into a number of pieces and casting each piece to a separate tallier. The final result of the election came from joining all partial tallies (there were cryptographic mechanisms to assure the accuracy of the final tally). Representative homomorphic-based voting protocols have been recently proposed by Sako and Kilian [SK94], and by Cramer et al. [CFSY96,CGS97]. Homomorphic-based voting protocols require far greater computing power than mixing-based voting protocols (both on the client and on the server sides). This need effectively prevents the construction of homomorphic-based voting systems using thin-client devices to cast ballots. Another inconvenience of these protocols is their low flexibility with respect to ballot formats (e.g. these voting protocols cannot support write-in candidates).

Table 2 summarizes the main features of each of either group of voting protocols.

Mixing-based voting protocol	**Homomorphic-based voting protocol**
Generally requires anonymising channel	Substitutes anonymising channel with separate talliers
Allows thin-client implementations	Requires thick-client implementations
Supports complete ballot flexibility	Requires predetermined and inflexible ballots

Table 2: Mixing-based vs. Homomorphic-based voting protocols

Simpler Protocols

There are some voting protocols that use rather simpler methods to ensure the desired security requirements. These protocols can often be successfully attacked with less effort or reduced privileges than the other protocols.

Among the simple methods used to secure voting applications, is often found the two-agencies model. The basic idea of this model is to segregate the functions of verifying the identity and authenticity of voters, on the one hand, from the function of counting the ballots on the other, between two functionally and structurally separate units. In general, protocols based on the two-agencies model cover some of the e-voting security requirements only if both agencies do not collude, and provided that some parts of the system are completely trustworthy [Cra96]. Accuracy of election results, for example, can be assured only by trusting the counting agency to properly count all ballots. Further security requirements such as the ability of voters to verify the results and to publicly object to the tally (with irrefutable proof of errors), are not assured at all.

From the above, we may conclude that any REV system must incorporate either an advanced mixing-based or homomorphic-based cryptographic voting protocol in order to have adequate security.

4. Comments on the Issues

4.1. Issue 1

"Should the proposed 'key principle' (paragraph 59) be adopted?"

From Vote Recording to Vote Counting

We agree on the requisite for a trusted path between the voter signalling his/her intention and the recording of the vote in the REV system. Such a trusted path would ensure the voter's privacy and the vote's integrity during its travel across the network and the front-end voting systems. Still, we believe that this trusted path must be complemented with the assurance that neither the voter's privacy nor the vote's integrity will be compromised *once* the vote has been recorded in the system. That is to say that the real need is for a trusted path between the voter's signalling of intention and the *counting* of the vote. In other words, we believe it is important to differentiate between the processes of vote recording (i.e. the action of recording the vote in a digital ballot box at the vote collection domain) and of vote counting (i.e. the action of actually including the vote in the final tally). Figure 1 illustrates this distinction. Paragraph 59, as it is currently written, covers up to vote recording, without reaching vote counting. In effect, the security does not enter the "black box" back end of the REV system and control what goes on inside of it.

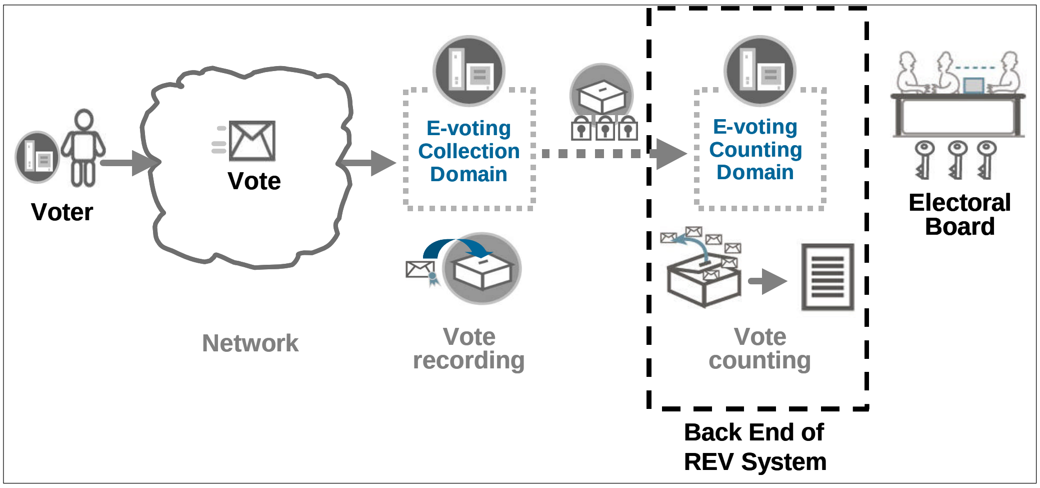

Figure 1: Vote Recording and Vote Counting Distinguished

Replicating the Protection in Conventional Systems

In conventional voting systems, physical protection measures are applied to hide the voter's intention until vote counting, and from that time on, no correlation back to the voter will be possible. Such physical protection measures include using opaque ballot envelopes, mixing of ballots in the ballot box, and ensuring that *all* members of the electoral board (as well as the observers) would need to collude to perform any kind of fraud. E-voting security systems must replicate these physical security measures in order to be successful and to avoid the existence of points of attack, whether they are inside the electronic environment or outside of it.

Receiving Assurance

In conventional voting systems, the voter receives assurance that her vote is recorded as intended, that is, she receives assurance that the vote has been introduced in the ballot box. She then trusts the system (something easily done given the existence of the electoral board and the observers) to correctly count her vote when the ballot box is finally opened. E-voting security systems must initially offer the same level of verifiability, additionally allowing the voter, if verification fails, to *publicly prove* with *irrefutable evidence* that she still has not voted in spite of having cast a ballot.

Adding an Electoral Board

The report states that in a traditional system, the trusted path is achieved by means of pencil, paper, and ballot box. We would add also the existence of an electoral board to this list of real-world measures. The pencil, the paper and the ballot box would be useless if there were not an electoral board in charge of the ballot box. Think about a voter using the pencil to mark the paper, and placing her vote into a ballot box which is in front of a single unknown person. Would the voter leave the place feeling comfortable about this situation (with regard to her privacy and her vote’s integrity)? The distribution of trust among the members of an electoral board is essential to gain security in e-voting systems [Bor96].

No Observable Properties

We find the phrase “have no observable properties” to be ambiguous in meaning, and offer an alternative: “not leak any information whatsoever to anyone.” While our alternative is not as concise as the original, it has the great advantage of being clearer.

Following from our observations detailed above, we suggest that the Key Principle be modified as follows:

> The signalling of intent, by the voter, into the electronic environment should not leak any information whatsoever to anyone, and the voter should receive irrefutable assurance that her vote was received unaltered by the electoral board in charge of vote counting as it was intended.

4.2. Issue 2

“Is the security profile set out in Annex A of the report valid and complete?”

We believe that the security profile of Annex A could be strengthened, in particular with regard to the security objectives that must be addressed by the voting protocol.

System Control Principles

Accordingly, we would modify the list of System Control Principles given in paragraph 95, by adding two additional principles and updating the definition of “Data integrity” and “System accountability”. Table 3 gathers the proposed modifications of paragraph 95.

Data verifiability (*added*)	Allowing the voter to verify the correct treatment of their vote and providing the ability to irrefutably demonstrate incorrect treatment.
No-coercion (*added*)	Preventing vote-selling and coercion.
Data integrity (*modifications in italics*)	Ensuring that each vote is recorded *and counted* as intended, *not manipulated or removed; and ensuring that no invalid votes are added.*
System accountability (*modifications in italics*)	Ensuring that system operations are logged and audited *without compromising voters’ privacy.*

Table 3: Proposed modifications to the list of control principles

OS2 (Effective Voter Authenticity)

We would make a couple of clarifications to Security Objective OS2 (pp. 26). First, it is not enough to authenticate voters only when they access the voting service. This voter authentication must in some way be transferred to the particular vote. It is in this way that the accuracy of election results can be guaranteed. Otherwise, if votes cannot be authenticated after the fact, any attacker with enough system privileges could add invalid votes. The second clarification is that the system must ensure that each voter can only vote once (or alternatively, the system must provide some unambiguous way to select one of the votes cast by a voter if multiple votes are allowed to be cast).

In our opinion, the authentication token mentioned in Security Requirement OS2 (pp. 29) could also be provided beforehand to voters by some service other than the registration service (a national digital ID card would be an example of this).

OS3 (Effective Voter Anonymity)

Given the previous distinction we have made between *vote recording* and *vote counting*, we believe that the restriction on the *vote recording mechanism* identifying the individual voters (pp. 29) can be lifted. The *act of voting* can be recorded and traced back to the voter to ensure accuracy, while using cryptographic techniques to protect the contents of the vote. However, during the *vote counting* process (when the contents of the vote are made clear), cryptographic techniques must prevent *anyone* from correlating the vote back to the voter. This is exactly how conventional systems work. In this way, we would clarify that the vote *counting* mechanism (where votes appear in clear) must not identify the individual voter.

OS4 (Effective Vote Confidentiality)

We would make two clarifications on Security Requirement OS4 (pp. 29). First, the report states that “effective end-to-end encryption may be required …”. We would be very cautious in the use of the expression “end-to-end”. If applied to the delivery channel, “end-to-end” means from the vote-casting client device to the vote collection server, where the vote is eventually recorded. This is of course not sufficient to secure the e-voting application. The confidentiality of votes could be broken by anyone with enough privileges on the vote collection server (whether an external or an internal attacker). Vote collection servers will be complex systems connected to a network, and therefore difficult to secure sufficiently.

“End-to-end” must go farther than vote collection servers. Actually, it must not be applied only to the delivery channel, but rather to the entire voting protocol. Voting is in reality an action between the voter and the electoral board. “End-to-end” should encompass this entire process. This is the way to prevent successful attacks at the vote collection servers (or at any other point on the way from the voter to the electoral board). Accordingly to this argument, we would modify the statement as:

> “Effective end-to-end encryption *(i.e. encryption between the voting client device and the authorities in charge of vote counting)* may be required to protect the vote from disclosure to third parties *and to malicious authorities* during transmission *and storage*.”

Our second clarification concerns where the report states that direct evidence of how the vote was cast must not be sent to the client system. This could actually be done in the case of using a certain class of tamper-resistant hardware client systems, if a high level of verifiability was desired (i.e. to allow the voters to locate their specific votes in the published results).

OS8 (Information Integrity)

We believe Security Objective OS8 (pp. 26) should be strengthened by adding "*Moreover, the e-voting service must not be able to inject bogus information such as invalid votes*". In this way, Security Objective OS8 would match the goals of the requirement "Accuracy" as presented in Table 1.

OS12 (Operator Integrity)

Even though we believe Security Objective OS12 (pp. 27) has good intentions, it should not preclude in any way the development of security solutions that protect e-voting even from dishonest personnel.

OS13 (Open Auditing and Accounting)

As argued previously, the implementation of Security Requirement OS13 (pp. 32) must not compromise voters' privacy. An audit could reveal such things as who voted and when, and also what are the overall results of the votes contained in a ballot box. However, an audit should never reveal the contents of a particular vote if it were possible to link this with the voter.

4.3. Issue 3

"Should Annex A be adopted by Government as the foundation for securing the 2003 pilots?"

We believe Annex A could be strengthened as described above, and be adopted as the foundation for securing the 2003 pilots.

In any case, security issues must be specifically addressed in further pilots.

4.4. Issue 4

"Is CESG right to conclude (paragraph 69) that current security technologies are not able to meet all the requirements implied by the key

principle, the security profile, and recommendations 1-5? If so, what strategy should be adopted?"

We agree that current security technologies are not able to meet all the requirements implied by the key principle, the security profile, and recommendations 1 to 5.

However, in Section 5 we will present a promising e-voting security technology being developed by Scytl that is based on a sound cryptographic voting protocol. This security technology meets:

- Key principle: It meets the proposed strengthening of the key principle presented in Section 4.1.
- Recommendation 1 (security profile): It meets all the requirements of the security profile that are applicable to the voting protocol (and it also meets the additional requirements suggested in Section 4.2).
- Recommendation 2: It does not meet Recommendation 2. In fact, we believe REV systems *do require* a trusted voting agent on the client side to be secured to a minimum acceptable level.
- Recommendation 3: It meets Recommendation 3 if we assume a minimum amount of computational power on client devices (which is actually the current technological trend).
- Recommendation 4: It absolutely meets Recommendation 4.
- Recommendation 5: It is not applicable to the voting protocol.

We believe that a good strategy for the 2003 pilots would include testing this technology.

4.5. Issue 5

"Is a scheme broadly in line with Annex C: theoretically sound? likely to be implementable by suppliers? likely to be manageable by electoral administrators? likely to be usable by, and acceptable to, voters? achievable in a pilot in May 2003?"

Usability

We believe that, besides the security issues that we will deal with later, the scheme of Annex C suffers from a serious usability problem. Voters are forced to deal with long strings of nonsense numbers that will cause much confusion and may eventually result in many voters giving up. Consider the nightmare caused by the disastrous design of the Florida butterfly ballots in the US 2000 Presidential election [Bri02]. An e-voting system that does not consider usability as one of the primary issues is inevitably condemned to catastrophe. In our opinion, voters need to deal with a very user-friendly voting agent to cast their votes. In a powerful enough voting device such as a PC, this would involve a virtual ballot clearly listing the different options and allowing voters to point-and-click their preferred option.

Usability issues aside, the PCIN codes proposed in Annex C of the report do not allow for flexible ballot formats. For example, they cannot support write-in candidates.

Misguided Trust in the Philosophy of the Two-Agencies Protocols

The proposed approach in Annex C follows the logic of the two-agencies model analysed earlier in these comments in that it divides a number of electoral tasks between different pairs of bodies to prevent fraud. This was shown to be too weak to secure electronic elections in the public sector (mainly because possible attacks are simple to perform and it is difficult to provide irrefutable evidence of any vote tampering).

> Many opportunities for two-party collusion exist in the approach detailed in Annex C that would be very dangerous to the integrity of the system.

The separation between Tallier and Matcher to avoid correlating voter IDs to the identities of the voted candidates is *too* weak. The Tallier itself can actually infer the intention of every voter ID from the total number of votes of each candidate number. From this, the Tallier is also in a position to change votes to the preferred candidates in an undetectable way. Furthermore, a simple collusion between the Tallier and the Matcher compromises privacy and accuracy – votes may be altered without detection, and invalid votes may be added.

The Validator and the Authenticator (and this means any system administrator or skilled external attacker gaining enough privileges on both systems) can change the intention of voters without easily being detected. This can be done by brute-force attacks on PCINs – even though the attacker might not know which candidate she is really voting for, a different valid vote may be substituted for the original.

In itself the Validator is a dangerous single point of attack. Votes can be easily removed from the Validator once the Response IDs have been sent to the

Authenticator (and this remains undetected assuming that the logs produced at the Gateways are manipulated accordingly).

Misguided Trust in the Creation of Pre-encrypted Ballots

While it is true that the approach outlined in Annex C does not place any trust in the client devices, this is actually achieved by unduly trusting some back-end elements of the system, which in our opinion is an imperfect approach to securing e-voting systems. Voters must place too much blind trust in the processes of ballot generation and ballot distribution, as well as in the storage of the generated codes.

The privacy of voters and the integrity of votes ultimately relies on the secrecy of the Voter IDs and PCINs. This is essentially the same as a password-based mechanism of protection, which is regarded by computer security experts [MOV97, Sch00, Per02] as a very weak security mechanism (for many reasons). Anyone with specific knowledge on some information related to the ballot generation process may easily frustrate a number of security requirements. In particular if such an attacker colludes with any of the other elements of the election system, then the results may be catastrophic. The reach of the attack basically depends on the level of knowledge gained about the information related to the ballot generation process. A single piece of information that is *critically sensitive* is the key of the HMAC used for ballot generation. Anyone in possession of this key can easily subvert most of the security requirements of the election. This alone represents an unacceptable threat to the system. As a general conclusion, we would remark that blind trust in the elements and processes that are performed on the server-side of the voting system is unadvisable.

> The proposed solution should consider the possibility of internal attacks (and of collusions between internal privileged attackers) more seriously.

It seems to us that the presented scheme, with the aim of meeting Recommendation 2, devises a solution that does not demand computing power on the client devices. However, we believe that the current trend of incorporating more and more intelligence into all kinds of client devices is actually minimizing the need to devise solutions that trade-off security for the lack of computational power on the client side.

No Irrefutable Proof Given

The approach given in Annex C does not totally deal with the system principles "no-coercion" and "data verifiability" that we proposed in Section 4.2. First of all, the sale of voting credentials would be quite easy. Secondly, the mechanism of the Response IDs does not provide voters with a complete and irrefutable assurance as to the treatment received by their votes. Receiving a valid Response ID just means that someone with knowledge on how to calculate Response IDs from Voter IDs and PCINs (if things work according to plan, the Authenticator) has received the vote. The voter has no means at all to be sure if her vote has even reached the Ballot Box. She has even fewer means to prove anything in case she is not satisfied.

> Dispute resolution is not an easy matter when irrefutable proof that a valid vote has been cast is not provided to the voter.

4.6. Issue 6

"Are the other recommendations sound? Is anything missing?"

Some comments/suggestions follow:

Paragraphs 13,31,40,64,92: It is highly advisable to also consider the possibility of skilled internal attackers with privileges on the communication network or on the vote collection servers.

Paragraph 32: If the voter authenticates to an SSL website using a weak authentication protocol, her credentials can be stolen without an extraordinary effort and therefore the voter can be subsequently impersonated.

Paragraph 32: The authentication could be non-anonymous if there were additional measures to prevent the disclosure of the contents of the vote once recorded on the server side.

Paragraph 36: Digitally signing the code can mitigate the risks of fake software distribution.

Paragraphs 41,76: Adequate protection measures can prevent (or at least reduce to a trivial level) individual coercion and vote selling in REV systems.

Paragraph 42: The Pnyx system outlined in Section 5 meets these requirements if used along with coercion codes distributed personally.

Paragraph 48: It is worth noting the main weakness of the two-agencies model: if the authorizer and the collector collude, the privacy of voters is compromised.

Paragraph 53: It is not clear why a voting applet adds many risks. In May 2002, a successful pilot with a security system using voting applets took place at the Universitat de Barcelona, involving an electoral roll of 2.200 voters.

Paragraph 65: If the assumption requires only a reasonable amount of computational power, then it does not *severely* limit the ballot-casting channel. We believe a voting agent on the client side is needed to offer minimal acceptable levels of protection in REV systems.

Paragraph 75: If postal and traditional voting take precedence, then electronic ballots can be automatically removed with very little effort provided that electronic ballots can be authenticated, as when using the Pnyx protocol introduced in Section 5.

Paragraph 77: Any voting protocol should be designed assuming that the delivery channel is not secure.

Paragraph 78: Adequate protection measures can prevent (or at least minimize to a trivial level) coercion and vote-selling based on processing voting receipts (vote acknowledgements).

Paragraph 90: Digital signatures allow the enforcement of control both over the content and the delivery of the EVCA.

Paragraph 156: The scratch-off panel is not enough to assure voters that the details have not been compromised. The scratch-off panel ensures the details have not been compromised during their distribution. However, the voter does not actually have any guarantee as to the back-end process of generating the details or of creating the scratch-off panel over the printed details before the distribution process.

Section B.7.2: In the private sector, the first binding election over the Internet in Europe (1997 Presidential Election of the IEEE’s Spanish Chapter of Information Theory) was conducted using a secure prototype developed by an academic research group at the Universitat Autònoma de Barcelona. Scytl is a company spun off from this group.

Paragraph 211: Adequate protection measures can prevent (or at least minimize to a trivial level) coercion and vote-selling (both individual and massive) in REV systems.

Paragraph 213: Adequate protection measures allow the creation of audit trails that enable recounts that do not compromise voters’ privacy. Such audit trails do not gather the votes in clear text but the adequately enciphered votes instead. Digital signatures or equivalent authentication mechanisms ensure that enciphered ballots actually reflect the intention of voters.

Paragraph 225: PDAs offer more than enough processing power to implement a mixing-based voting protocol. Mobile phones supporting Java 2 Micro Edition (J2ME) also have more than enough processing power.

Paragraph 226: We would add another principle area: meeting all security requirements against internal privileged attackers (i.e. minimize the amount of trust placed on any element of the system).

Paragraph 228: We believe a certain amount of client processing power is needed to accomplish a reasonable security level in REV systems.

5. Pnyx: Scytl's e-Voting Cryptographic Protocol

Scytl has developed a patent-pending cryptographic voting protocol that is the product of eight years of pioneering academic research. This protocol is being used as the foundation for a high-class cryptographic component software that enables accurate, verifiable, anonymous and non-coercible e-voting and can seamlessly plug into existing platforms and serve as a solid basis for the development of new ones. This cryptographic software, currently under development, is called Pnyx, after the hill where political discussions and voting took place in ancient Athens.

A Protocol Assuming Mutual Suspicion

Besides the possibility of external attackers, Pnyx assumes a mutual suspicion between all the legitimate parties involved in the election (i.e. voters, election authorities, system administrators, etc.) and therefore also deals with the possibility of internal attackers. The objective of Pnyx is to protect every party, even from collusions between other legitimate parties. Pnyx reduces to a minimum the trust that must be placed on any single element of the e-voting system.

We agree on the statement of paragraph 92 of CESG's report. External cracking is a serious issue in REV systems that use data-transport networks such as the Internet to carry votes. However, we would like to remark that the use of complex computing systems for the collection of votes opens up new risks derived from the difficulty of preventing/detecting attacks by skilled insiders with privileges. Also, the complexity of the processes of ballot generation and distribution such as those proposed in Annex C paves the way to a number of internal attacks.

Using Pnyx, immediately after the signalling of intent by the voter, the vote is protected at the client device by means of suitable cryptographic techniques. Votes are passed to the vote collection domain while protected, so that not even privileged parties can compromise voter privacy and election integrity. The vote counting domain (where at last the ballots must be opened up) is separated from the vote collection domain. The vote counting domain may be connected to the network through a specific and highly secure network connection or, even better, it can even be completely isolated on its own. In addition to isolating the vote counting domain, this subsystem can be made very simple and secure (simplicity is an effective approach to security [Sch00]). The isolation, physical security and simplicity of the vote counting domain allows Pnyx to effectively secure the most sensible elements. This serves as the most successful basis to accomplish Security Requirement OS11.

The Voting Process According to Pnyx

Pnyx embodies a simple and yet very effective and powerful patent-pending approach to e-voting security. The whole security system replicates the physical protection measures encountered in traditional elections. Voters are authenticated (by means of digital certificates and strong authentication means, which can be complemented with the use of biometrics and/or hardware tokens). The voter obtains a validated voting receipt for the intended vote, which will be useful for verification purposes, in an effort to compensate for the lack of transparency inherent in electronic voting systems. Note however that the voting receipt will not enable coercion or vote selling. The vote is protected at the client side by means of a digital envelope. Once sealed, the vote cannot be read by anyone – neither voting authorities, nor system administrators, crackers, nor even the voters. A proof of authenticity is associated with the digital envelope to ensure the accuracy of the results and to make an audit possible if needed. In this way each voter becomes bound to his/her vote, preventing the deliberate repudiation of a vote. No voter will be able to successfully claim foul play unless some fraud has actually occurred. Under specific and warranted conditions, this proof of authenticity will allow the determination as to whether gerrymandering has really occurred.

The digitally enveloped vote is cast and placed into the digital ballot box. At the vote counting domain, a full electoral board made up of several members (along with independent observers or any other third party with interests in seeing to the fairness of the election) operates a node where the digital ballot boxes will be opened. Only a qualified majority of the members of the electoral board is able to build the private key that allows the opening of the digital envelopes that protect the ballots, much in the same way that a specified number of military personnel are required to turn their trigger keys simultaneously to launch a nuclear missile. During the opening of a ballot box, a secure mixing process is performed to prevent a correlation between the voters and the votes.

Securing the Black Box

Pnyx makes a clear separation between those elements that must be connected to the network – and that by definition will have a high degree of complexity (essentially, the vote collection servers) – and those used to perform critical functions (e.g. opening ballot boxes) and that must be operated off-line for more security.

Furthermore, Pnyx provides real *end-to-end protection* – that is the encryption of the vote from the voter's voting agent through to the electoral board in charge of vote counting. No assumptions need to be made with respect to the security of the delivery channel or the front-end servers of the vote collection domain since no man-in-the-middle attacks are possible (by proxy servers, crackers on the vote collection domain, system administrators, etc.). Not even an insufficient number of members of the electoral board can break the vote encryption. The black box is indeed secured.

Preventing Coercion and Vote Selling

With respect to coercion, Pnyx adopts a practical approach. Individual coercion is severely limited by having coercion codes distributed off-line. These coercion codes allow any voter to confound coercers by simulating a real act of voting by creating a fake vote that will not be counted. If properly distributed, coercion codes leave no means for the coercer to distinguish the real vote from the fake ones. Massive coercion is prevented by means of voting receipts that are specially designed to make them useless to coercers or vote-buyers to gain evidence of the vote cast. In the case of opting for a higher level of verifiability (i.e. allowing voters to identify their corresponding particular votes on the published results), additional tamperproof hardware is needed as part of the client devices.

Use of Mixing-Based Protocol

The cryptographic protocol that powers Pnyx is mixing-based although the trust distribution techniques among the members of the electoral board allow Pnyx to avoid the use of anonymising channels for casting the votes. This means that only conventional data-transport services of regular data communication networks such as the Internet are needed.

Because of Pnyx's mixing-based nature, the part of the code that runs on the client side does not require much computing power. An earlier version of Pnyx that was used successfully in a 2002 election involved a Java applet of a mere 83 Kbytes and that required less than 0.1 seconds to perform all needed cryptographic operations on a Pentium III PC. Because of the version of Java used, all current Internet browsers supported this applet. Such low constraints make Pnyx suitable to be implemented on J2ME-enabled phones, PDAs or any other thin-client device.

Table 4 summarizes the main features of Pnyx. More information about Pnyx can be found at http://www.scytl.com, or obtained by e-mailing products@scytl.com.

Voting protocol	Mixing-based, without need for anonymising channels
Key principle	Mutually suspicious parties, assuming internal attackers and avoiding excessive trust on any one element of the system
Service protection	Clear separation between vote collection and vote counting; the latter performed off-line on a fairly simple and physically secure node
Security techniques	Replication of the conventional security measures into their cryptographic equivalents (ballot envelope, identification, trust distribution, etc.).
Demands on the computational power of clients	Small amount of computation is needed (J2ME suffices). Current implementation requires a 83 Kb Java applet.
Audit trails	Enabled, without compromising voters' privacy (votes are sealed and authenticated afterwards)
Principal security requirements addressed	Privacy; Authentication; Accuracy; Secrecy of intermediate results; No-coercion; Verifiability

Table 4: Pnyx's main features

When checked against the first five recommendations proposed in CESG's report, Pnyx meets the following requirements:

- Recommendation 1: Pnyx successfully meets OS1, OS2, OS3, OS4, OS8, OS11, OS13. Additionally it meets "no-coercion" and "data verifiability" as we previously defined them. Security objectives OS5, OS6, OS7, OS9, OS10, OS12, are not applicable to a voting protocol.
- Recommendation 2: Pnyx, as any other e-voting cryptographic protocol which does not force voters to trust the other parts of the system, needs a small amount of computational power on the client side.
- Recommendation 3: A majority of client devices is supported.

- Recommendation 4: Fully met. No assumptions are made on the delivery channel infrastructure.
- Recommendation 5: It is not applicable.

6. Concluding Remarks

Conventional electoral methods are designed to deal with the risks that exist in such elections. For this reason, most voters in democratic countries view these electoral processes as reliable and secure. What these systems do well is to adequately manage the existing risks. In the leap to e-voting the integrity of our electoral processes can (and must) be ensured. The trick is to maintain the strengths of the conventional systems while using experts to create standards and rigorous certifications for any new voting method.

Our opinion is that a secure REV system requires the existence of a sound application-level cryptographic voting protocol at its core. Of the two main groups of such protocols (mixing-based and homomorphic-based), mixing-based voting protocols offer the best trade-off between security and practicality. Only a modest amount of computing capacity on client devices is required so that virtually all Internet browser-based systems as well as a growing number of mobile devices – both personal digital assistants and mobile phones – will be able to run the client-side part of the protocol.

In conventional electoral methods there are two strengths that should not be discarded in e-voting systems: user-friendliness and the checks and balances that are based on the mutual suspicion of all parties involved in the electoral process. User-friendliness dictates that the voter is presented with a clear, unambiguous list of candidates from which a selection is made – a virtual ballot. Changing this time-tested process into an abstract number selection would put us on the road to electoral confusion of Floridian dimensions. Mutual suspicion dictates that we cannot put blind trust in the black box of the e-voting system. In conventional voting systems, the presence of physical security and several observers prevent electoral shenanigans once a vote has been collected. If these measures are not replicated in the e-voting process, electoral integrity can be compromised by internal attacks. We believe that the approach of Annex C suffers from this weakness, for in the back-end processes of generating voter IDs and PCINs, any attack could devastate the election's integrity and voters' privacy.

We believe that the Key Principle should be modified slightly in three ways – one of which is to clarify it, while the other two are to strengthen it.

> The signalling of intent, by the voter, into the electronic environment should not leak any information whatsoever to anyone, and the voter should receive irrefutable assurance that her vote was received unaltered by the electoral board in charge of vote counting as it was intended.

The phrase “no observable properties” in the original may lead to ambiguous interpretations, so we have attempted to clarify its meaning. The word “recorded” has been substituted in our modification to extend it to take into account the back-end processes of counting. The last modification specifies that the assurance is irrefutable – implying a much greater utility to the voter.

We also believe that two security directives are lacking from Annex A: data verifiability (the irrefutable assurance of the modified Key Principle) and the non-coercion of voters. Both of these measures can go a long way into improving the confidence of the voters in the process.

Scytl’s voting protocol has been developed to do just one thing: make e-voting secure. Pnyx is based on an application-level cryptographic voting protocol that meets or surpasses the Key Principle and all of the recommendations in CESG’s report – if you allow the requirement of a small amount of computing power in the client devices. The use of the Pnyx-based software in future e-voting trials would solve a large part of the e-voting security concerns and help to accelerate the adoption of e-voting in the UK.

Annex A: About the Author

Andreu Riera holds a Ph.D. on Applied Cryptography, specifically on cryptographic protocols to secure large-scale electronic voting systems. His experience on e-voting security started in 1994 when he undertook academic research on the topic, and continues today in Barcelona, where he runs Scytl, a company devoted to engineering application-level cryptographic solutions for critical applications such as e-voting.

Scytl's team is currently developing world-class security software for REV platforms that is based on eight years of previous academic research and a patent-pending cryptographic voting protocol. This security technology offers the best protection available to existing REV platforms, while alternatively it may be used as a solid base to build new platforms as well.

Contact
phone: +34 605 895 517 - email: andreu.riera@scytl.com

Professional Experience

Company	Title	Period
Scytl Online World Security	Co-founder and CEO	Since 04/2001
iSOCO Intelligent Software Components	Research Manager	02/2000 – 03/2001
Universitat Autònoma de Barcelona	Lecturer / Researcher	09/1993 – 09/2000
Fundació Universitària del Bages Escola d'Organització Industrial Consell Tecnològic del Bages	Visiting Lecturer	1995 – 1999
Freelance	Architect/Programmer	1988 – 1994

Academic Titles

Title	Institution	Date
Ph.D. on Computer Science	Universitat Autònoma de Barcelona	12/1999
MSc on Computer Science	Universitat Autònoma de Barcelona	10/1995
Graduated in Computer Science	Universitat Autònoma de Barcelona	07/1993

Research Activities / e-Voting Cryptography Expert

- Sole author of a PCT patent application for a cryptographic method to ensure the security of trust seals on the Internet.
- Co-author of a PCT patent application for a cryptographic protocol to enable trustful and reliable electronic electoral processes.
- Participation in the project “Development of a secure voting service in an Intranet/Internet environment using PKI”. U.A.B., funded by CICyT (TEL97-0663). From July 1997 to July 2000.
- Participation in the project “Implementation of a TCP/IP secure voting scheme over a LAN”. U.A.B., funded by CICyT (TIC94-0331). From May 1994 to May 1997.
- Speaker as an expert on e-voting cryptographic protocols in various conferences and seminars organized by the Internet Society, International Federation for Information Processing, Spanish Association of Cryptology and Information Security, University of Stockholm (Royal Institute of Technology of Sweden), Universitat de Vic, Universitat Rovira i Virgili, Fundació Jaume Bofill, Fundació ESICO and Casa de la Cultura de Girona.

Publications

Internet Voting: Embracing Technology in Electoral Processes. Chapter published in the book “Electronic Government: Design, Applications and Management”, edited by Åke Grönlund. Idea Group Publishing, 2002.

Applying New Technologies in Electoral Processes: Technological Security and Social-Political Concerns about Remote Electronic Voting (in Spanish). Simposio Argentino de Informática y Derecho. Universidad de Buenos Aires, September 2001.

Efficient Construction of Vote-Tags to Allow Open Objection to the Tally in Electronic Elections. Information Processing Letters, Vol. 75, n. 4, pp. 211-215, October 2000.

An Efficient Method to Allow Public Objections to the Tallying Process in Electronic Voting (in Spanish). Proceedings of the VI Reunión Española sobre Criptología y Seguridad de la Información, pp. 417-424. Tenerife, September 2000.

Design of Implementable Solutions for Large Scale Electronic Voting Schemes. Ph.D. Thesis, Universitat Autònoma de Barcelona, December 1999.

A Practical Approach to Anonymity in Large Scale Electronic Voting Schemes. Proceedings of the 1999 Network and Distributed System Security Symposium, NDSS '99, pp. 69-82. Internet Society. San Diego, February 1999.

An Introduction to Electronic Voting Schemes. PIRDI - 9/98. Computer Science Department, U.A.B., September 1998.

How Smartcards Helped to Solve a Cryptographic Paradox. Proceedings of the V Reunión Española sobre Criptología y Seguridad de la Información, pp. 35-36. Málaga, September 1998.

An Uncoercible Verifiable Electronic Voting Protocol. Proceedings of IFIP TC11 14th international conference on Information Security (SEC '98), pp. 206-215. Vienna-Budapest, August 1998.

Large Scale Elections by Coordinating Electoral Colleges. Proceedings of IFIP TC11 13th international conference on Information Security (SEC '97), pp. 349-362. Copenhagen, May 1997.

SecuDE as a Tool to Construct Certification Structures: Some Shell Scripts to Facilitate its Use. Proceedings of the IV Reunión Española sobre Criptología, pp. 143-150. Valladolid, September 1996.

Computer Networks (in Catalan). Computer Science Department Internal Publication. Universitat Autònoma de Barcelona, September 1995.

Public key certification hierarchies and their operation using the security toolkit SecuDE. Master Thesis. Universitat Autònoma de Barcelona, October 1995.

Internet (in Catalan). Computer Science Department Internal Publication. Universitat Autònoma de Barcelona, May 1994.

Unix Security (in Catalan). Computer Science Department Internal Publication. Universitat Autònoma de Barcelona, March 1994.

Annex B: References

[Bor96] Borrell, J. *Design and Development of a Cryptographic Scheme to Perform Secure Elections over a LAN* (in Catalan). PhD Thesis, Autonomous University of Barcelona, 1996.

[Bri02] Bricklin, D. *Ballot Usability in Florida.* Web page at http://danbricklin.com/log/ballotusability.htm.

[BY86] Benaloh, J.C. & Yung, M. *Distributing the Power of a Government to Enhance the Privacy of Voters*. Proc. of 5th Annual ACM Symposium on Principles of Distributed Computing, pp. 52-62, 1986.

[CC96] Cranor, L.F. & Cytron, R.K. *Design and Implementation of a Practical Security-Conscious Electronic Polling System.* Technical Report WUCS-96-02, Washington University, 1996.

[CFSY96] Cramer, R., Franklin, M., Schoenmakers, B. & Yung, M. *Multi-Authority Secret-Ballot Elections with Linear Work*. Proc. of Eurocrypt '96, LNCS 1070, pp. 72-83, 1996.

[CGS97] Cramer, R., Gennaro, R. & Schoenmakers, B. *A Secure and Optimally Efficient Multi-Authority Election Scheme*. Proc. of Eurocrypt '97, LNCS 1233, pp. 103-118, 1997.

[Cha81] Chaum, D. *Untraceable Electronic Mail, Return Addresses and Digital Pseudonyms*. Communications of the ACM, v.24, n.2, pp. 84-88, 1981.

[Cra96] Cranor, L.F. *Electronic Voting: Computerized Polls May Save Money, Protect Privacy*. ACM Crossroads, April 1996.

[FOO92] Fujioka, A., Okamoto, T. & Ohta, K. *A Practical Secret Voting Scheme for Large Scale Elections.* Proc. of Auscrypt '92, LNCS 718, pp. 244-251, 1992.

[MOV97] Menezes, A.J., van Oorschot P.C., and Vanstone, S.A., *Handbook of Applied Cryptography*. CRC press, 1997.

[PIK93] Park, C., Itoh, K. & Kurosawa, K. *Efficient Anonymous Channel and All/Nothing Election Scheme.* Proc. of Eurocrypt '93, LNCS 765, pp. 248-259, 1993.

[Per02] Perrig, A. *Shortcomings of Password-Based Authentication.* Web Page at http://paris.cs.berkeley.edu/~perrig/projects/usenix2000/node2.html

[Rie99] Riera, A. *Design of Implementable Solutions for Large Scale Electronic Voting Schemes*. PhD Thesis, Autonomous University of Barcelona, 1999.

[RST02] Riera, A., Sanchez, J., Torras, L. *Internet Voting: Embracing Technology in Electoral Processes.* Chapter published in the book "Electronic Government: Design, Applications and Management", edited by Åke Grönlund. Idea Group Publishing, 2002.

[Sch00] Schneier, B. *Secrets & Lies*. John Wiley & Sons, 2000.

[SK94] Sako, K. & Kilian, J. *Secure Voting Using Partially Compatible Homomorphisms*. Proc. of Crypto '94, LNCS 839, pp. 411-424, 1994.