Comments by Scytl on the SERVE security report

An alternative and constructive perspective on Internet voting security

April 14, 2004, First Version

March 15, 2007, Last update

Andreu Riera, PhD
Chairman & co-founder

Jordi Puiggalí
VP Research and Development

Scytl Secure Electronic Voting S.A.

Barcelona, Spain **Dallas, USA** **Singapore, Singapore**

Introduction

The SERVE voting system (Secure Electronic Voting and Registration Experiment)[1] was an Internet-based voting system designed by Accenture and its subcontractors for the United States Department of Defense's FVAP (Federal Voting Assistance Program). The objective of the program was to make registration and voting easier for military personnel abroad and American citizens living outside the United States.

The SERVE system was planned for deployment in the 2004 United States primary and general elections, allowing around 100,000 voters to cast binding ballots over the Internet from anywhere in the world. Eventually, future versions of the system were supposed to handle about 6 million votes (the entire population of American eligible overseas citizens plus military personnel and their dependents).

A group of experts in computerized election security was assembled by the FVAP to help evaluate SERVE. Two three-day meetings were held in July 2003 at Caltech in Pasadena, California, and in November 2003 at Accenture in Reston, Virginia. Four members of the group of experts attended both meetings. They were David Jefferson, computer scientist at Lawrence Livermore National Laboratory, Aviel D. Rubin, Associate Professor of Computer Science and Technical Director of the Information Security Institute at Johns Hopkins University, Barbara Simons, a technology policy consultant, and David Wagner, an Assistant Professor in the Computer Science Division at the University of California at Berkeley. These four computer scientists published in January 2004 a report[2] entitled "A Security Analysis of the Secure Electronic Registration and Voting Experiment (SERVE)" (the "SERVE Security Report").

The SERVE Security Report denounces a list of security vulnerabilities that, according to the authors, prevent the proper use of the system as it was intended. The report recommends "*shutting down the development of SERVE immediately and not attempting anything like it in the future until both the Internet and the world's home computer infrastructure have been fundamentally redesigned, or some other unforeseen security breakthroughs appear*."

SERVE was eventually cancelled by the Department of Defense. As it can be read on the FVAP's website at http://www.fvap.gov/services/serve.html, "*Concerns were raised*

[1] http://www.fvap.gov/services/serve.html
[2] http://www.servesecurityreport.org/

that, given the current security vulnerabilities of the Internet and voters' personal computers, no Internet voting system could be 100% secure. Rather than potentially bringing the integrity of the election results into doubt, the Department of Defense has decided not to deploy the SERVE system for use in 2004."

In the present document, we, as electronic voting security experts, provide our views with regard to the SERVE Security Report. The purpose of this document is to provide an alternative and constructive perspective on Internet voting security by analyzing the specific threats mentioned in the SERVE Security Report and proposing solutions to mitigate and, in some cases, eliminate some of these threats. Our opinions focus strictly on the technical side of the discussion, and are given under the constraints imposed by our limited knowledge of the SERVE system.

Our comments on the SERVE Security Report

We essentially agree with the report in that securing Internet voting is far more difficult than securing e-commerce or other types of applications. However, we disagree with the report's conclusion that an all-Internet secure voting system is far ahead of its time. There are solutions –involving a combination of procedural, technological, and physical security measures– that allow conducting these initiatives with reasonable confidence and reliability.

Internet voting offers numerous advantages over other remote voting systems such as postal voting. We are proponents of Internet voting, provided that its introduction is made gradually and reasonably and, more importantly, its deployment considers security as a primary aspect. As any voting method, Internet voting will need confidence, and without security there is no confidence.

We see Internet voting as a technical alternative to postal voting for absentee voters. We strongly believe that, if properly implemented, Internet voting may be more secure and reliable than postal voting. Large-scale disenfranchisement (even in some cases selective disenfranchisement given the actual geographic distribution of political preferences) takes place regularly in conventional elections that depend on the reliability of postal systems. During the November 2003 Election to the Parliament of Catalonia (Spain), the inefficacy of the postal system in some of the countries where Catalonian voters reside caused a large number of ballots to fail to fulfill the delivery deadlines. It is remarkable that from Mexico more non-binding online ballots were received through an Internet voting experiment[3] than real binding postal ballots were tallied. Moreover, most of the comments received by the voters that participated in the experiment were against the current postal voting system and demanding new Internet-based voting methods. In other European countries, postal voting cannot be often implemented because of stringent electoral deadlines. For example, until recently, Italian non-residents had to travel from their actual –abroad– locations to their home precincts in Italy to be able to cast their votes in certain elections (the Italian government paid the domestic train fare from the Italian border to the home precinct).

The security report against the current SERVE architecture denounces real threats and vulnerabilities. However, in our opinion, the report uses an excessively catastrophist language, and we strongly disagree with their final conclusion that "*the SERVE project*

[3] See "*Sistema de votació electrònica a través d'Internet*" at the Catalan Government website, http://www.gencat.net/governacio-ap/eleccions/e-votacio.htm (in Catalan)

is thus far ahead of its time" and that it pursues "*an essentially impossible task*". In appendix C, the report proposes an alternative to SERVE which actually leads to a kind of polling-place voter verified paper audit trail (VVPAT) system, therefore failing to take advantage of the many benefits of using the Internet for remote casting of ballots.

We believe that the current SERVE architecture could indeed be improved. Still, we think that adequate solutions already exist that may secure an online voting process to reasonable levels.

The authors of the report base most of their arguments against SERVE on the fact that nowadays Internet and current PCs are inherently insecure, and that large-scale attacks could very easily be deployed from anywhere in the world. They also accuse SERVE to be proprietary software. In their opinion:

> *"The troubles with SERVE derive from three fundamental design choices: It uses the Internet heavily, with all of the vulnerabilities that implies (e.g. denial of service, spoofing, and man-in-the-middle attacks). It relies on voters using private, unsecured PCs with proprietary, commercial software configured to accept mobile code, with all of the vulnerabilities that implies (e.g., virus attacks, various kinds of privacy violations). And SERVE itself is proprietary software, with all of the vulnerabilities that implies (e.g. security holes, bugs, insider fraud)."*

In our opinion, indeed these three design choices involve vulnerabilities, but all of them can be properly addressed. We will go more deeply into each of the three issues.

Using the Internet

The Internet is just a communication medium. We agree that it is complex and that many potentially malicious actors have access to it. Still, when designing the security architecture of any remote voting system, the initial premise that the communication medium is insecure has to be considered anyway.

The overall security architecture for a remote Internet voting system such as SERVE should to rely heavily on application-level cryptographic protocols and on physical protection of some of the modules of the cryptographic system (which could even be completely disconnected from any network during all the electoral process). This would enable real end-to-end security, counteracting most of the denounced attacks (man-in-the-middle, spoofing, etc.). Please note the special meaning of the term "end-to-end"

here. We do not mean “from client to server”, but rather “from voter to electoral board”, the actual two ends of a ballot casting procedure.

We propose a solution[4] that tunnels through the Internet, the voting servers, and all the possibly malicious (external or internal) actors with access to them. A cryptographically protected virtual dialog takes place from a secure voting agent operated by the voter to the security system operated by the electoral board (the latter physically monitored and disconnected from any network).

Denial of Service (DoS) attacks against the ballot collecting servers are still possible, as it is possible to lose a significant number of absentee ballots sent through the postal system as a result of negligent, or even intentional, delays in the delivery of those ballots to the electoral board (which sadly happens quite frequently). No voting system is completely secure but, as the SERVE Security Report states, “*any new form of absentee voting should be as secure as current absentee voting systems*” and this should be the standard against which to compare Internet voting. Additionally, there are certain measures (e.g. VPNs from controlled locations, dynamic IP addresses, etc.) that can significantly mitigate or even eliminate (in the case of the VPNs) the effect of a DoS attack.

Unsecured PCs

We totally agree with the report in that current PC architecture and the current SERVE architecture make the casting of ballots risky in high-stake elections using private PCs as client devices. The virus problem is real (although we believe that large-scale attacks against the voting system are unlikely to remain undetected).

In public environments, to circumvent this problem, we would propose the use of clean voting kiosks as an alternative to private PC-based voting. These voting kiosks could be installed in strategic places (e.g., embassies, consulates, military bases, etc.). Therefore voters could have the choice of using a safe voting terminal in case they are not confident on the security of their PCs. These voting kiosks do not need to be based on special purpose voting devices. These kiosks can be based on standard PC infrastructure, extensively audited and physically secured.

[4] Additional details on the commercial implementation of our proposed solution can be found at http://www.scytl.com/eng/soluciones.htm

Scytl has successfully experienced with the use of clean bootable CDs (live CDs) to convert any standard PC into a secure voting terminal. This proposal allows the secure reuse of any pre-existing PC infrastructure without requiring the installation or hardening of any software currently installed in the PC. It also facilitates the audit of the voting terminal, since all the executed software is contained in the CD. Therefore auditing the CD contents is equivalent to auditing the complete voting software.

Proprietary closed software

It is commonly agreed that security must not depend on obscurity. In addition, any new type of voting system must pursue public confidence, and enclosing the entire system into a black box obstructs achieving such confidence.

This is not to say that an electronic voting system must be made completely open. There can be constraints on the opening of the system. First, we think that by maintaining a clear separation of critical and non-critical modules, only a subset of small, physically protected and easily auditable, modules have to be made open. Second, we feel that these critical modules must be made open only to the appropriate parties, but not to absolutely everyone. Further details on our vision on this separation of modules may be found in the article "Advanced Security to Enable Trustworthy Electronic Voting" published in the proceedings of the 3rd European e-Government Conference[5]. We believe that the proposed architecture encompasses the vendors' desire of keeping closed many of the parts of their voting systems, while at the same time the voting system stops being a black box, becomes much more easily auditable, and can be successfully secured as only a small number of simple modules are really critical and some of them can reliably be physically protected.

The SERVE Security Report denounces that software testing can become very difficult. We agree. However, we feel that keeping critical modules as simple as possible goes a long way in helping in this task. Adequate cryptography and physical security allow achieving a reasonable level of security and auditability. Let us clarify this point with an example. Suppose that what we were pursuing is the confidentiality of a piece of information (in this case a number from 0 to 9) that has to be transmitted from Alice to Bob through the Internet. For the sake of the example's simplicity, suppose we were only interested in the requirement of confidentiality and not in any others. Suppose that Alice uses his Internet-connected private PC at home to send the confidential number

[5] http://www.academic-conferences.org/eceg2004/2-proceedings-eceg2003.htm

to a complex set of servers managed by a system administrator that will deliver the information to Bob. At first glance, the confidentiality requirement mentioned earlier faces a frightening scenario.

Imagine now that we design a very simple device that only performs the following computation:

1. It asks the user to enter the confidential information I (a number from 0 to 9).
2. It asks the user to enter a key K (another number from 0 to 9).
3. It adds, modulo 10, the previous numbers to obtain a result R = I+K mod 10.
4. It displays R to the user.

Such a simple device could be easily audited and secured. Now imagine we devise a simple procedure to securely share a secret key K between Alice and Bob. Alice uses her device to encrypt[6] the confidential information I using the key K. The device displays the encrypted message R to Alice who types it into her PC. When Bob finally receives R, he can recover I by using his secure device and K. This critical task can be done by Bob away from the views of any other person and away from any (inherently insecure) system connected to the Internet.

Given the premises of this example, no matter how insecure and how closed is the complex technical infrastructure between Alice and Bob, the confidentiality of information is still assured. Only the small devices in the hands of Alice and Bob have to be made open, tested, certified and properly secured. Of course, this protocol does not address many other security requirements (the integrity or the availability, for example). The purpose of the example is just to demonstrate that the fulfillment of certain security requirements does not necessarily mandate to inspect and to keep completely secure all of the involved technical systems. Most of them can still be inherently insecure, but in this example if we focus our auditing efforts on a couple of simple devices and a procedure to share a secret key, the whole system remains confidential.

In the same way, for software testing purposes in an Internet voting system it is crucial to separate critical from non-critical modules and to keep critical modules very simple.

[6] Provided that the information and the key K follow certain statistical properties, the encryption proposed in this example is unconditionally secure. This means that no matter how many resources and how much time an attacker has, the encryption cannot be broken. The original scheme of such unconditionally secure encryption was proposed by Gilbert Vernam at the beginning of the XXth Century.

Scytl and Pnyx

Scytl Secure Electronic Voting S.A. (Scytl) is a highly specialized software company that commercializes the most secure Internet voting solutions currently available. These solutions incorporate unique cryptographic protocols that enable to carry out all types of electronic voting processes or elections in a completely secure and auditable manner. Scytl's advanced e-voting security technology positions the company as a leader in the e-voting industry.

Scytl was formed as a spin-off from a leading research group at the Universitat Autònoma de Barcelona, Spain. This group, funded by the Spanish Government's Ministry of Science and Technology, pioneered the research on e-voting security in Europe since 1994 and produced significant scientific results, including 25 scientific papers published in international journals and the first two European Ph.D. theses on electronic voting security, by Prof. Joan Borrell and Scytl's founder Dr. Andreu Riera (in 1996 and 1999, respectively). This research group also participated in the first Internet binding election in Europe (i.e., the 1997 election to the Presidency of the IEEE IT Spanish chapter).

One of Scytl's main strengths is its unique technology, which derives from over twelve years of pioneering R&D and is protected by a portfolio of international patents. The groundbreaking e-voting cryptographic protocols developed by Scytl provide e-voting with the highest levels of security, in terms of anonymity, urn integrity, and voter-verifiability. This innovative technology has received numerous international awards, including the prestigious IST Prize granted by the European Commission in 2005.

Scytl has customers both in the private and public sectors. The former are local, state (regional), and federal governments which license Scytl's e-voting products to carry out their elections, referenda, or citizen consultations by electronic means. The latter are large corporations and organizations that choose Scytl's technology to carry out by electronic means electoral/consultation processes such as labor union elections or shareholders' meetings. Some of these customers represent leading references in the electronic voting industry (e.g., governments in Spain, Switzerland, United Kingdom, Philippines, Mexico, Argentina, Finland and Australia that are pioneering new electronic voting applications). Scytl's products have already been successfully used in multiple projects worldwide, some of which represent breakthrough projects for the electronic voting industry.

Scytl has developed Pnyx.core[7] to provide e-voting with the highest security standards. Pnyx.core is a software product which comprises four main modules. The Pnyx Voting Client module is a Java-based API that implements the cryptographic ballot casting protocol for different kinds of voting client systems, such as PCs (using web browsers and Java applets) or mobile phones (using Java midlets and Microsoft .NET framework). The Pnyx Voting Service module is an independent process that implements the server-side ballot casting protocol. The Pnyx Voting Service module is a standalone process (system service or *daemon*) that can run in Windows or UNIX (Linux) platforms. The Pnyx Voting Proxy module is a Java-based API that allows the integration of Pnyx in any kind of electronic voting system (based on servlets, CGI, J2EE...). The Pnyx Voting Proxy module manages the protocol messages between the Pnyx Voting Client and the Pnyx Voting Service independently of the communication channel used by the voting system (HTTP, WAP...). Finally, the Pnyx Mixing Service module is the component that is operated by the electoral board members, mainly to process the contents of the election ballot boxes by means of a cryptographic mixing protocol. This module is a standalone process that can run in any kind of Windows or UNIX platform.

Pnyx.core is a software product in constant evolution and improvement so that it is continuously taking advantage of new achievements and developments. The objective of Pnyx.core is to offer end-to-end security, from the voter to the electoral board. In the most secure architectures, Pnyx.core is complemented with Hardware Security Modules (HSM), voting kiosks and physical security measures.

The SERVE Security Report mentions five main threats to any truly democratic voting system. We believe that Pnyx.core helps, to some degree or another, in mitigating all of these threats, and it provides some additional important advantages. The five voting system threats are:

- Disenfranchisement: Pnyx.core does not directly address Denial of Service attacks. However, it includes measures to prevent legitimate digital ballots that have already been cast from being excluded from the tally. Additionally, as previously mentioned, certain measures can be implemented in parallel to Pnyx.core to mitigate DoS attacks.

[7] http://www.scytl.com/eng/soluciones.htm

- Ballot modification: With Pnyx.core, ballot modification at the server side is not possible, even by privileged technical actors. The use of clean voting kiosks (or computers operating in a controlled environment) prevents ballot modification at the client side.

- Loss of privacy: With Pnyx.core ballots can be deciphered only by a minimum threshold of members of an electoral board. The electoral board operates on an audited and physically protected and isolated server. With regard to voters' privacy, Pnyx.core represents a very effective measure to counteract insider attacks. No technical actors with privileges on the systems that manage/store the digital urn can defeat this security requirement. As an immediate result, Pnyx.core completely solves the problem –reported in page 10 of the SERVE Security Report– regarding the temporary existence of clear text ballots in the difficult-to-secure ballot collecting server systems connected to the Internet. With Pnyx.core, ballots are protected by digital envelopes that can only be open by means of a private key which simply does not exist. As for the case of ballot modification, security on the client side can be addressed by using voting kiosks or computers operating in a controlled environment, as mentioned previously in this document.

- Double voting: With Pnyx.core double voting is not possible since every individual ballot includes a proof of authenticity, and the opening of the digital ballot box, which is controlled by the electoral board, includes a process that checks the set of (encrypted) ballots against the signed electoral roll.

- Vote-buying: Any remote voting system (not necessarily by technical means) is vulnerable to individual coercion and/or vote buying. It is a policy decision to accept this or not. If postal voting is accepted, then nothing should prevent us from accepting Internet voting. As the SERVE Security Report states, Internet voting, as a new form of absentee voting, should be as secure as current absentee voting systems. A completely different matter is the possibility of massive vote-buying that can take place in a digital voting system if it is not properly protected. Pnyx.core has been designed to completely remove such possibility, and therefore to fulfill the SERVE Security Report requirement that "*what we must avoid at all costs is any system in which it is possible for a successful large scale or automated (computerized) attack to compromise many votes*".

Besides the five voting threats already discussed, Pnyx.core provides additional advantages. The main two advantages are:

- Testing and certification: Pnyx eases the task of testing and certifying the voting system. Pnyx.core allows focusing the testing efforts on small pieces of the entire voting system while some parts of it may even remain closed.

- Voter verifiability: Any electronic voting system has an inherent lack of transparency that may undermine the electorate's confidence in the electoral process. To counteract this inherent lack of transparency, Pnyx.core offers a mechanism that enables every voter to gain absolute confidence that his/her vote has been received by the respective electoral board with total confidentiality and integrity (exactly as it happens in conventional polling-place elections). It is worth noting that in addition to providing individuals with higher confidence in the accuracy of the results, individual verifiability is also an efficient statistical auditing mechanism. Even if only a tiny percentage of participants actually verified their ballots, small manipulations of the results would become apparent. For example, in an election with 100,000 ballots cast, if just 1% of the participants verified the correct treatment of their ballots there would be a probability of 99.35% of detecting a manipulation of just 500 ballots.

We strongly believe in the need for voter verified voting. However, we feel that voter verified voting must not necessarily be paper-based; adequate all-technological voter verified solutions are possible.

Conclusions

In our opinion, the SERVE Security Report denounces real security threats. We totally agree with the report in that securing a voting application requires much more effort than securing other online applications. However, we disagree in that today it is an impossible task to conduct, in a secure way, a binding election over the Internet.

We have introduced Pnyx.core, a security solution specially designed to cope with the electronic voting security concerns which has been used in multiple binding elections in the public sector worldwide. Pnyx.core has not been described in detail as this was not the objective of this document. However, additional information can be obtained from Scytl.

If the proper security architecture were used, SERVE could be a magnificent Internet voting experiment. In particular, the use of Pnyx.core along with the substitution of private PCs by clean voting kiosks or controlled PCs (and the appropriate procedural and technical measures to keep such computers secured) would eliminate almost all of the problems denounced by the SERVE Security Report.

As we understand it, the use of Pnyx.core would immediately solve the following problems of the current SERVE architecture:

- Pnyx.core addresses the listed five main threats to the voting system, especially the possible loss of privacy caused by privileged insider attacks which should be a primary concern given the current SERVE architecture.
- Pnyx.core simplifies a lot the task of testing and certifying the voting system.
- Pnyx.core provides the additional feature of voter verifiability, to counteract the limitation in terms of transparency of an electronic voting system and to maximize security.

About the authors

Andreu Riera initiated his academic research on electronic voting security in 1995, accomplishing an extensive list of results that comprise publications in international scientific publications and book chapters. His PhD Thesis “Design of Implementable Solutions for Large Scale Electronic Voting Schemes” was published in 1999. He has coauthored five international patent applications, two of which refer to technical models to bring trust to electronic voting systems. He has also presented in numerous electronic voting conferences and seminars in several European countries. He also advised the Spanish Senate on the correct implementation, referring to security aspects, of electronic voting. In 2001, he founded Scytl as a spin-off of a leading academic research group on electronic voting security.

Jordi Puiggalí has headed Scytl’s Research & Development Department since the formation of the company. Mr. Puiggali has been instrumental in the development of Scytl’s technology and intellectual property, co-authoring numerous international patents on application-level cryptography and e-voting security. Prior to joining Scytl, Mr. Puiggali was the Technical Director for PKI and security projects at the IT department of the Autonomous University of Barcelona. Mr. Puiggali has also actively collaborated with the cryptographic research group of the Department of Computer Science at the Autonomous University of Barcelona where he co-directed research projects on PKI and applied cryptography. Mr. Puiggali is a security expert and has participated as a speaker and lecturer in numerous international conferences on computer security and applied cryptography. Mr. Puiggali has a bachelor degree in Computer Engineering from the Universitat Autònoma de Barcelona.