Security in Large-Scale Internet Elections: A Retrospective Analysis of Elections in Estonia, The Netherlands, and Switzerland

Guido Schryen, Eliot Rich

Remote voting through the Internet provides convenience and access to the electorate. At the same time, the security concerns facing any distributed application are magnified when the task is so crucial to democratic society. In addition, some of the electoral process loses transparency when it is encapsulated in information technology. In this paper, we examine the public record of three recent elections that used Internet voting. Our specific goal is to identify any potential flaws that security experts would recognize, but may have not been identified in the rush to implement technology. To do this, we present a multiple exploratory case study, looking at elections conducted between 2006 and 2007 in Estonia, Netherlands, and Switzerland. These elections were selected as particularly interesting and accessible, and each presents its own technical and security challenges. The electoral environment, technical design and process for each election are described, including reconstruction of details which are implied but not specified within the source material.

We found that all three elections warrant significant concern about voter security, verifiability, and transparency. Usability, our fourth area of focus, seems to have been well-addressed in these elections. While our analysis is based on public documents and previously published reports, and therefore lacking access to any confidential materials held by electoral officials, this comparative analysis provides interesting insight and consistent questions across all these cases.

Effective review of Internet voting requires an aggressive stance towards identifying potential security and operational flaws, and we encourage the use of third party reviews with critical technology skills during design, programming, and voting to reduce the changes of failure or fraud that would undermine public confidence.

***Index Terms*— e-voting, Internet voting, Internet election, security, verifiability, RIES, Estonia, Neuchâtel**

I. Introduction

In the course of the recent development of electronic democracy and electronic services, electronic voting has drawn a remarkable degree of attention. Beyond direct recording electronic (DRE) voting machines in designated polling places, the use of Internet voting to allow for remote balloting has been applied in more than 100 elections and in 14 countries between 1996 and 2007 [27]. Milestones in the adoption of Internet voting include the March 2000 Arizona Democratic Party's presidential preference primary election, which was the first legally binding election to employ Internet voting [29], and the 2007 Parliamentary Elections in Estonia, where Internet voting was first used at the national governmental level [17].

Proponents argue that Internet voting increases voter access and participation in the political process, lowers costs, and protects against electoral fraud [29]. However, implementing Internet voting requires extensive revisions to long-established procedures for voting, counting, monitoring and auditing. This task is extremely challenging: In the early 2000s, Internet voting and related security issues were systematically investigated by independent groups of security experts [7], [23], [25], who concordantly concluded that technological threats to the security, integrity, and secrecy of remote Internet voting systems are significant and that the possibility of large-scale automated attacks leads to a level of risk so high as to be unacceptable. A recent example of insufficient security in electronic voting systems was demonstrated in an independent assessment of the voting systems certified for use in California [4]. Within the proprietary code researchers found that one piece of software appends a three-letter suffix to a password and sends this result over the network, another duplicates the same encryption keys in all of its machine source code, and a third system uses its own name as a hard-wired password. Other flaws arise from elementary coding vulnerabilities and flawed cryptography. This demonstrates clearly that critical flaws remain an unresolved concern.

One might argue that e-Commerce systems provide a basis for secure Internet voting systems. Unfortunately, there are some important differences between the two applications that make it difficult to share similar architectures:

--**Impact:** Elections are inseparably linked to democracy, and can be directly and decisively affected by compromised election processes. Democracy relies on broad confidence in the integrity of elections as well as the successful completion of each transaction.

--**Identity:** The voting franchise is usually not transferable,

Manuscript received July 17, 2009.

Guido Schryen is with the International Computer Science Institute, Berkeley, CA 94704 USA (phone: 510-666-2972; fax: 510-666-2956; e-mail: schryen@gmx.net).

Eliot Rich is with the University of Albany, Albany, NY 12222 USA (e-mail: e.rich@albany.edu).

so the identity of the voter must be accurate. In e-Commerce, the use of a credit card by an account co-owner is not a security failure.

--**Availability:** A well-timed assault on an Internet election infrastructure, such as a DDoS attack, can delay or invalidate returns and disenfranchise voters. A similar attack on an e-Commerce site would have less serious results, as the buyer can either revisit the website after operations are restored or select an alternate vendor.

--Authentication and anonymity: An Internet vote requires confirmation of the right to vote and anonymity within the vote transaction, an unusual duality. Business transactions require authentication through passwords, PINs, or biometric data, but the buyer and seller are not usually anonymous.

--**Monitoring and audit:** When they are made available within a voting architecture, plaintext voting receipts may capture the voting event, but must also obscure the voter's selections. If a voting receipt showed how a vote was cast in plaintext, vote selling and coercion might occur [35]. Voting audit trails must also provide protection of the voter's identity while ensuring the integrity of the vote. This also implies that access to the audit trail must be protected. In contrast, e-Commerce customers may also receive receipts for transactions, but these receipts often list details in order to facilitate resolution of complaints and the efficient delivery of goods.

--**Voter privacy:** Internet voting does not assure that voters' physical privacy is respected, again raising the spectre of coercion. No such concern exists in e-Commerce.

Despite these characteristics of Internet voting, several large-scale Internet elections have been conducted in the recent past. To what extent were the requirements, concerns, and solutions offered by the academic community employed or helpful in the support of reliable democratic processes? We address this question by examining three recent European elections that employed Internet voting. To the best of our knowledge, no prior study has addressed this question by analyzing and comparing multiple election cases. In contrast to other studies that analyze a single election or election system (e.g. [4], [25], [29], [35]), our multiple case study provides some basis for empirically-driven generalization. Using almost exclusively public documents, published literature, and interviews with electoral observers, we identify possible security oversights or exploitable gaps, verifiability issues, and lack of transparency.

The rest of this paper is structured as follows: In Section 2, we identify the properties deemed desirable for Internet voting based on a review of the literature. Section 3 presents our research methodology by explaining how multiple exploratory case study analysis is used in this paper to explore three large-scale Internet elections. Sections 4-6 contain the description and the analysis of the particular elections held in Estonia, The Netherlands, and Switzerland, respectively. In Section 7, a cross-case analysis is conducted. Section 8 concludes the paper and provides an outlook on further research.

II. Theoretical Background

A. Requirements and desired properties

In one of its earliest pronouncements, the United Nations formulated "The Universal Declaration of Human Rights" [42], including a call for universal and equal suffrage and voting anonymity. These universal requirements are given substance by [14] through specific voting security-related properties: these include accuracy, equal voting power (termed invulnerability in [14]), and privacy:

--Accuracy: (i) Votes must not be altered or eliminated, and invalid votes must not be counted. (ii) The vote tally must be "perfect", either by preventing or detecting inaccuracy. If inaccuracies can be detected, but cannot be corrected, an election system is termed "partially accurate" [14].

--Equal Voting Power: Democratic theory and practice requires that: (i) Only eligible voters can vote, and that (ii) Eligible voters can vote only once.

--Privacy: (i) A link between the voter's identity and the voter's selection(s) must be impossible. (ii) The voter must not be capable of proving that she voted in a particular way. This property fights vote buying and extortion, but may conflict with verifiability, a property defined below.

While not explicit in [14], the property of usability is suggested in the text: *"A system is convenient if it allows voters to cast their votes quickly, in one session, and with minimal equipment or special skills."* [14, p.3]. In addition, verifiability is desirable. There are two elements within this property:

--Auditing of votes: The most robust level of verifiability allows any citizen or outside body to determine that all votes have been counted correctly. A weaker definition of verifiability requires only that voters can verify their own votes and correct any mistakes without sacrificing their privacy [14].

--Auditing of voting procedures and voting systems: After identifying important several election software products [4] concluded that the audit and review of voting systems are highly important. System evaluation, system testing [12], and system certification [15] can be used to demonstrate the absence of known problems. However, [28] projects e-voting processes into the framework of the Common Criteria and concludes that even that effort does not suffice. A useful instrument for finding the causes of problems when they occur are forensic audit trails (FAT), log files that track each system events and form the basis for the detection of malicious activity and errors involving the recording and counting of votes [34].

A final property discussed in the literature is transparency. Transparency of the technological and organizational elements of an election system and its processes add to the credibility of an election. Much of this paper's analysis comes from review of materials that attempt to meet this property. While some argue for "security by obscurity," the need for credibility and the significant risk of unintentional or intentional exposure of the system's programming to the public [38] makes relying on obscurity for protection untenable. When third-party reviewers

can be employed, these reviewers must have the technical expertise, time and resources to evaluate election architecture designs and implementation.

B. Voting system design primitives, protocols and attack counter-measures

Much of the technologically-oriented e-voting literature discusses alternative design primitives, protocols and technological attack counter-measures. Reference [26] identifies three general design approaches for building e-voting systems based on three primitives. Mixnet-based primitives, introduced by Chaum [10], are part of the protocols of [24] and [39] and are used in the e-voting system SureVote [9]. Homomorphic encryption-based primitives were introduced by Benaloh [2], used in [1], [3], and [21] and implemented in E-Vote [20]. Blind-signature-based primitives were introduced by Fujioka et al. [19], are used in [8] and [19] and adopted in Sensus [14]. Further protocols are provided in [26], who also provide examples for some other approaches that are not based on any of the above primitives.

Voting protocols are dedicated to enhancing security in communication, but are of limited value against security threats to voting devices (e.g., keyloggers, viruses). Reference [7] proposes specialized operating systems as one mechanism for protecting security-critical applications from malicious code. Voting operating systems are designed to protect security-critical applications from malicious code using security properties such as process isolation. Further instruments proposed are closed platforms employing smartcards, and trusted computing elements.

III. Methodology

We present our methodology in two parts: First we explain why we chose exploratory multiple case study as the primary research methodology. Second, we present the design of our case study.

A. Exploratory multiple case study analysis

As Internet elections are embedded in a societal environment, a comprehensive analysis of these elections needs to consider organizational and societal concerns along with the technology employed. According to [44], the appropriateness of a research strategy depends on three attributes: (1) the form of the research questions, (2) the control of behavioural events, and (3) the focus on contemporary events. Our research poses "what" and "how" questions, does not allow for controlling or manipulating behavioral events (elections) and focuses on contemporary elections, so we chose exploratory case study as our research methodology. In order to get a more comprehensive picture and to make the findings more robust, we selected multiple elections, resulting in an exploratory multiple case study.

B. Research design

Our research design follows the recommendations of [43-44] regarding exploratory multiple case study design. As we are focusing on Internet elections, we use one framework for all of our cases. We first describe briefly the electoral environment. We then describe the overall technological architecture of the Internet voting system, and then move to pre-electoral, electoral, and post-electoral processes in terms of technological and organizational procedures. We conclude each case with an analysis which systematically matches the properties of the elections with the election requirements as presented in Section 2.1.

To identify the cases for study, we examined a literature review on Internet voting [27] and the e-voting database of 239 elections provided on http://db.e-voting.cc, where we excluded those that were non-political, small-scale (less than 1000 eligible voters) or conducted before 2006. We examined the public documentation for the elections looking for cases using innovative technology and appearing in academic or lay media.

We also considered diversity of country, election level (national vs. local) and voting eligibility (everyone or only voters living abroad). We also applied subjective preferences of the authors, including the fluency of the authors with particular non-English languages. We ultimately selected elections in Estonia, the Netherlands, and Switzerland for comparison:

--Estonia is the first country worldwide to introduce legally-binding, nation-wide Internet voting without any preconditions. The election under investigation is the national parliamentary election conducted in March 2007.

--The Netherlands was an early adopter of information and communication technology for voting. We examine the Dutch 2006 national parliamentary election, where both stand-alone electronic voting machines nation-wide as well as Internet voting for citizens living abroad were employed.

--Switzerland is the country with the most experience in conducting legally-binding local elections via the Internet. The election under analysis is the March 2007 local referendum, where Internet voting was used in the city of Neuchâtel.

To get a robust picture of each election we visited the web site for each election's supervising authority, which gave access to official observations (e.g. OSCE reports). We extended the literature review by searching scientific papers on these elections and on the computer systems they employed through pertinent databases, including the IEEE Xplore Digital Library and the Web of Science. In the Estonian and Swiss elections we were able to arrange informal interviews with election observers.

In regards to the Estonian election we found several helpful sources. The OSCE/ODIHR Election Assessment Mission Report [31] provides a description of the election and how Internet voting was integrated. It is based on interviews with government representatives and state officials, election administration, political parties, academics, and civil society. However, the security analysis is at a high level and exemplary only. The Report for the Council of Europe by the European Union Democracy Observatory [18] is based on telephone interviews with voters and is purely non-technical. The Estonian National Electoral Committee websites [16-17] provide a description of the e-voting process and statistics.

The Dutch election was also observed by the OSCE, which provides the Election Assessment Mission Report [32]. We were also provided with an internal and non-confidential OSCE report that focuses on problems with e-voting machines and Internet voting [33]. This report is based on interviews with participants and on the observation of the election, but while this report was somewhat critical of the process, it is not generally available to the public. We also used [22], which provides a complete technical description and a verifiability analysis of RIES, the Internet election system used.

The selected Swiss case was not observed by the OSCE or any other independent organization. The "Schweizerischer Bundesrat" [40] and the "République et Canton de Neuchâtel" [36] provide high-level descriptions of the elections. References [11] and [40] provide a technical description and analysis, respectively, of the Pnyx system, the source for some of the components used in the election.

IV. Parliamentary Elections in Estonia

Our first case study examines the 3 March 2007 elections for the Estonian parliament. 30,275 voters used Internet voting, representing 3.4% of the eligible voters and 5.4% of the votes cast [17].

A. Electoral environment

In advance of the Estonian election voters could submit paper ballots or use the Internet to register their vote. On Election Day, they could cast their ballots in polling stations. An important feature of this election was the voters' ability to change their votes during the advance voting period, either by voting again through the Internet or by casting a ballot paper at a polling station. The voter could change her Internet vote an unlimited number of times, with the last electronic ballot being the only one counted; a vote cast by paper was final and annulled all Internet votes cast by the voter. Voters who cast a vote by Internet were not allowed to cast a vote on Election Day.

A cornerstone of the Internet voting system in Estonia was the exploitation of the existing national identification document (ID card), which is legally accepted for authentication and digital signatures. The computer used by the voter must have a smart card reader installed in order to process the digitally-enabled ID. Each ID card is accompanied with two Personal Identification Numbers: PIN1 is required to access the personal data stored on the ID and to use the ID card for personal identification to web-based services, while PIN2 is required for digital signing (http://www.id.ee/11039).

B. Design

The Estonian Internet voting system consists of the components shown in Figure 1. The Voter Application software allows citizens to cast their vote. Independent applications were designed for Windows (a signed ActiveX web browser component) and Apple Mac OS and Linux (stand-alone applications). The Internet Server provides the Voter Application to voters, stores the list of eligible voters, and forwards votes to the Vote Storage Server. The Vote Storage Server records votes during the voting period. The Counting Server is an offline, stand-alone computer, used to decrypt and count the votes recorded. The decryption of votes is performed using a Hardware Security Module (HSM). The module generates the public and private key of the Counting Server. The Certificate Authority Server provides the voters' digital certificates. A private software company developed all of these components, except for the Certification Server, based on specifications developed by the Estonian National Electoral Committee (NEC). The Estonian Informatics Centre is responsible for the physical hosting of the servers, as well as for providing Internet connections.

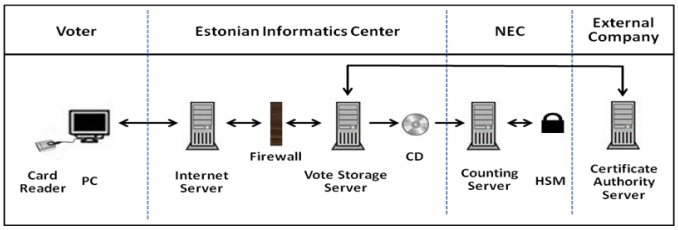

Fig. 1: Architecture of the Estonian Internet election system

C. Electoral processes

The current Estonian citizen ID card contains personal data and a private key on an embedded chip, providing individual identification for voting. The Internet voting process used in this election resembles the dual envelope method used for paper-based absentee voting. The Internet voter software creates an inner envelope (which is essentially an encrypted vote) and an outer envelope (which is essentially a digital signature).

The voting process is depicted in Figure 2. The Voter Application requests authentication data from the voter's ID card. To proceed, the voter enters PIN1 to identify herself. The Voter Application establishes an SSL connection with the Internet Server and sends authentication data to this server, which then looks up the voter lists to verify the eligibility of the voter. As the voter lists contain the Personal Identification Number (PIC) of each eligible voter, we assume that authentication data sent by the Voter Application also contain the voter's PIC.

The voter chooses one candidate by clicking on the name of the candidate in the client software. Unlike paper balloting, the application software prevents blank or physically spoiled ballots. The vote and a random number are encrypted with the public key of the Counting Server. In order to cast the vote, the voter must type in PIN2. PIN2 enables the card to sign the encrypted vote and is not transferred to the Internet Server. The encrypted vote is sent to the Internet Server, which verifies that the digital signature corresponds to the session owner. At this point, the description of the voting process in [31], p. 13, is not precise regarding what exactly is signed (we assume the encrypted vote only). When the vote is received by the Vote Storage Server, an entry is recorded in a log-file (LOG1), using the format (PIC, hash(enc(vote), random number)) [16]. The use of a hash function is intended to eliminate a link between the voter's decision and their identity

in the LOG1 file. Figure 3 provides an overview of the log files stored.

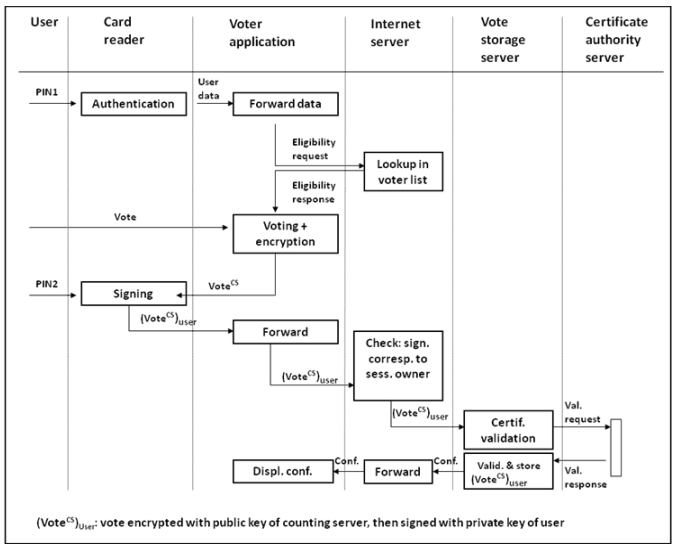

Fig. 2. Internet voting process in Estonia's 2007 parliamentary elections

The Internet Server forwards the encrypted vote to the Vote Storage Server, which accesses the Certificate Authority Server and requests the voter's certificate in order to confirm the validity of the digital signature that is attached to the encrypted vote. At the end of validation, the voter receives an on-screen confirmation that the vote has been cast. The encrypted vote remains on the Vote Storage Server until counting and tabulation is performed on Election Day. It is not clear whether the confirmation consists of text only, a unique confirmation code, or other information.

As each citizen can vote by both advance ballot and the Internet, a consolidation of votes is needed. After receiving lists from polling stations regarding any voters who cast a paper ballot during advance voting and who also cast a vote by Internet, NEC staff mark the corresponding electronic votes on the Vote Storage Server as "not to be counted". It is important to note that at this stage of the election a link between the (encrypted) vote and voter's identity exists. Cancelled Internet votes are logged in a file (LOG2), using the same format as in LOG1 with the reason for cancellation. Advance paper ballots are counted with those cast at the polls. At the end of Election Day, the NEC staff burns a CD from the Vote Storage Server that contains the last encrypted electronic vote of each voter. This CD is sealed and given to the Chairman of the NEC.

The counting of the electronic votes takes place on Election Day: The encrypted votes are transferred to the Counting Server by a CD-ROM. All entries transferred to the Counting Server are recorded in log file (LOG3) using the same format as LOG1. After the insertion of six physical keys to enable the HSM, the Counting Server decrypts the votes. Reference [16] reports that each decrypted vote is checked against the candidate list to determine if it is possible to vote for the candidate in that constituency. If the candidate number is incorrect, the vote is declared invalid. However, [31] states that the voter is provided an electronic ballot with candidates of the voter's electoral district and that ballots cannot be spoiled by the voter, which seems inconsistent. A corresponding notice is recorded in a log file (LOG4) in the format hash(encrypted(vote)). Valid votes are tabulated and recorded in a log file (LOG5), again in the format hash(encrypted(vote)). After the votes were counted on the Counting Server, a new CD is burned with the results. The CD is taken to a personal computer where the results are analyzed.

D.Analysis

Security: The Estonian Internet voting system shows design weaknesses with regard to all of our targeted security properties. Accuracy is endangered in multiple ways: On the client side, the use of card readers coupled to home computers rather than devices with their own display and keyboard makes the process vulnerable to PC viruses that change the input and the output of the card reader. An unwary voter may download fraudulent software purporting to support the electoral process. Although the authenticity of the website and the client voting software can be validated by the voter, as the ActiveX software is signed, voters may not be familiar with browser certificate checking. If fraudulent software is in place, votes may be altered or eliminated by while simultaneously presenting the voter a faked confirmation message. The voter has no means to confirm that her vote has been transmitted without distortion.

Serious threats exist on server side as well. The Internet Server and the Voting Server have code to invalidate votes, and the Voting Server and the Counting Server can add votes. Logging the process does not ensure that only invalid votes have been actually removed without confirmation that the logs are accurate.

Reference [31] reports that the PC used to read the CD containing the results was connected to the Internet during part of the time the counting procedure was conducted, a somewhat risky choice when considering the role of this device in the process. It is unclear whether any countermeasures against Denial-of-Service attacks against the Internet Server and the Vote Storage Server had been taken. There is an important privacy issue with the Estonian e-voting system. As the voting process allows voters to cast multiple and overwriting ballots, the Vote Storage Server needs to keep a link between the encrypted vote and the identity of the voter. Storing this link is a serious violation of the privacy principle. These security issues are rooted largely in the design of the Estonian election system. While they may well be managed by well-crafted processing, we have not seen external evidence to document their resolution.

Usability: The Estonian e-voting system requires the voter to have a card reader available and to install the card reader software. As of November 2006 over one million digitally-enabled ID cards had been issued with almost 900,000 eligible voters [31].. In this 2007 election, 98.8% of Internet voters used a Microsoft Windows-based web browser, through which ActiveX voting software was downloaded and executed [31]. According to [5], the voting software comes with a self-explanatory point-and-click interface, but is available only in

Estonian. As about 15% of Estonian citizens speak Russian as their mother tongue [31], there should be concern about disenfranchisement of the Russian-speaking minority.

Verifiability: It appears that individual voters cannot confirm the content of their Internet votes for lack of an accessible audit trail. In case of complaints, election officials can consult the log files described in the previous section and shown in Figure 3.

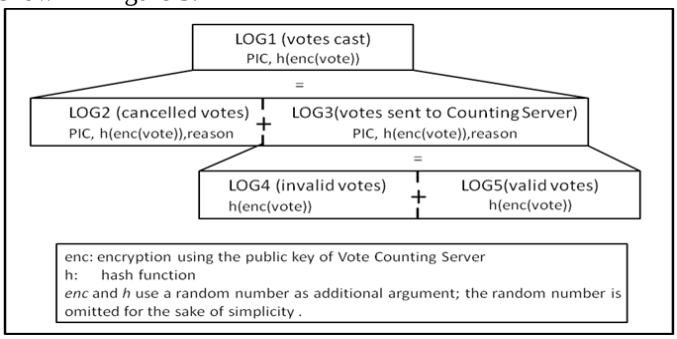

Fig. 3. Audit log files stored in Estonia's 2007 parliamentary elections

While these log files are useful, they are not a replacement for a process that verifies votes for the individual. For example, if a vote was deleted before being recorded in LOG1, the voter cannot prove that she has cast a ballot. A malicious individual with access to the Vote Storage Server also has access to the PIC codes, as these are required to validate the entries. The PIC codes could be used in combination with a fraudulent but properly formatted hashed vote to create votes for non-participants. Similarly, a determined insider can modify votes or substitute invalid votes for valid entries, all of which would be supported by the false audit trail in LOG2 and LOG3. The use of LOG4 to track votes for invalid candidates subsequent to the insertion of a false vote also appears possible. If all log files are correct, it can be proved that no votes have been added or deleted by validating the two constraints LOG1=LOG2+LOG3 and LOG3=LOG4+LOG5.

In the absence of explicit review, the correctness of the software is an assumption rather than an audited truth. Although some practical tests of the Voter Application and the Vote Storage Server are reported, there was no obligation to certify or test the system, the Internet voting system was not officially certified by an independent body and no full end-to-end logic and accuracy test was performed on the system [31].

An external auditing firm (KPMG Baltics AS) monitored and checked the activities of the NEC against written documentation describing the necessary steps and procedures. However, the final report is not public, and the external auditing company did not conduct any post-election audits. Overall, the Estonian e-voting system had practically no verifiability. While we do not claim malfeasance, we do note the opportunity for problems.

Transparency: According to [31], the election processes and the management of the Internet voting system were made transparent to the OECD, all political parties, and accredited observers. This included the opportunity to review the documentation of the system, the source code of the software, and all of the setup procedures in the process. However, the OECD report says [31, p. 20]: "*One reason cited by some political party representatives for not observing the internet voting process was (…) the lack of qualified personnel who could understand the process and provide effective control (…).*" Overall, the apparent lack of thorough oversight by independent security experts and unpublished audits are blows to claims of transparency.

E. Conclusion

Although the Estonian e-voting system falls far short of what we would consider a secure process, these concerns are not shared by officials and voters. In contrast, e-voting continues to be popular in Estonia, as the e-voting turnout at the 2009 European parliament elections was double that of 2007The desire for user simplicity, high turnout, enthusiasm for the new Estonian technology base and the opportunity to draw a structural parallel with traditional elections were important drivers in the design of the Internet voting software [5], [16]. Reference [5] cites an ERC member saying *"The goal is to make things easier for people, to increase participation […]It's impossible to build a system that is 100 percent secure. But it's as safe as it can be."* and the e-voting project manager *"You trust your money with the [I]nternet, and you won't trust your vote? I don't think so."*

V. Dutch elections to the House of Representatives

This case study refers to the 22 November 2006 Dutch elections to its House of Representatives. The government opted to use Internet voting as an experiment to support Dutch voters residing abroad, responding in part to previous problems with mail balloting. Moreover the purpose was to make it easier for the voter by non-place-dependent voting. They employed the Rijnland Internet Election System (RIES), developed originally for the local Rijnland District Water Board elections in 2004 [32]. RIES was used by 19,929 voters, a scant 0.16% e-voting turnout.

A. Electoral environment

Dutch voters select one candidate from a pre-designated list. In-country voters had the freedom to use any polling station in the country, and each station had direct recording electronic machines (DREs). Outside the country, Dutch voters used either Internet or mail voting, which is where our analysis is directed. Internet voters accessed the front-end of the voting system directly, while mail ballots were recorded by election employees into the same system. We have no information on how the electronic votes and the votes cast in polling stations were merged.

B. Design

In contrast to the Estonian case, RIES is directly derived from academic research. RIES is a simplified version of the system proposed by Robers [37]; its implementation differs from that proposed in the research by eliminating smartcards and voter-specific public key pairs and implementing an alternative organizational structure. Our reconstruction of the

technical details of RIES is based on [22] and [26], which provides a process-oriented perspective. As we have no information available on the architecture of RIES as implemented in the election under consideration, we depict the architecture as planned in these reference materials in Figure 4.

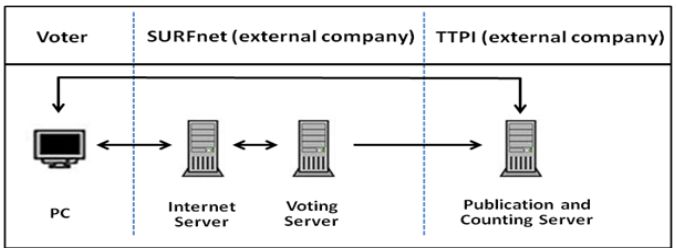

Fig. 4. Architecture of the Dutch Internet election system

Prior to the election each eligible citizen living abroad could request access to the Internet-based system and a secret key for individual access. These voters access the Voter Application, implemented as a JavaScript application embedded in a web site. The JavaScript code performs all user-side cryptography. The Internet Server receives encrypted votes from the remote PC, which are then stored in the Voting Server. Both servers are operated by SURFnet, a Dutch Internet service provider. (http://www.usenix.org/event/ lisa06/bofs.html). The Publication and Counting Server is operated by the company TTPI, developers of the RIES system.

C. Electoral processes

The RIES election scheme supports voter authorization, encrypted voting, and transparent identification of votes cast through the use of encryption keys and hashing. We draw this description largely from [22]. Before the election, TTPI generates an electronic identification code (El_ID) and a cryptographic DES key for each registered Internet voter. The El_ID and key are sent via post to each voter. In addition, TTPI prepares a table that applies the same user-specific key to the El_ID to create a Message Authentication Code (MAC), resulting in an encrypted "Voter ID". In addition, TTPI generates a set of MACs encrypting all valid votes for the specific voter. The resulting MACs are hashed using the Modified Detection Code (MDC2) one-way function [6], and published in a reference table. The TTPI copy of the user-specific DES keys is then destroyed. Figure 5 summarizes the pre-election procedure.

The application of MDC is crucial. If encrypted votes were to be published in the pre-election table without hashing, the encrypted values corresponding to specific candidates would represent valid votes and could be sent by anyone. As published MDC values are generated by TTPI through a one-way function, attackers do not know which values would match the corresponding MDC values.

During the election period, the Internet voter visits the election web page "internetstemmen.nl". This site holds a SHA 1 digital certificate to confirm its authenticity to voters. The voter then selects her candidate *j*, and enters her election key K_i into the appropriate field. The PC-based JavaScript code computes what the designers call the "technical vote," containing two values: her Voter ID, calculated by $MAC_{Ki}(El\ ID)$, and the hashed vote, $MAC_{Ki}(CAN\_ID_j)$. The technical vote is shown on the voter's screen and sent to the Internet Server through an SSL secured connection to SURFnet. The voter then receives a confirmation that the technical vote was received successfully. The voter is advised to store the technical vote after receiving this confirmation in order to be capable of performing a validation check.

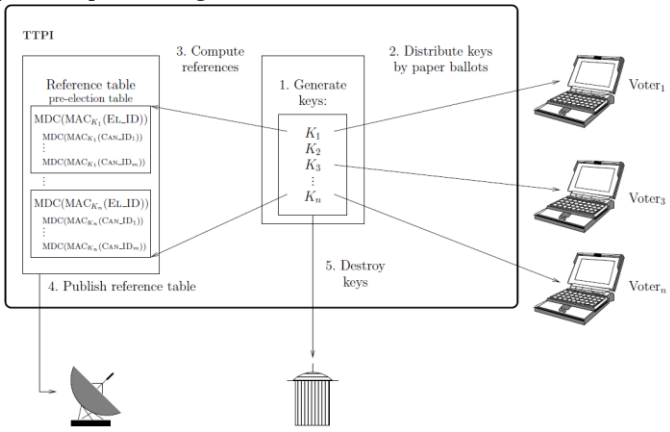

Fig. 5. Pre-election phase in RIES [22, p. 7]

When paper votes are used, TTPI records them into the electronic system. TTPI scans paper ballots (mail votes) and captures the Voter ID and the MAC for the selected candidate. No information is found describing how paper ballots are treated after their entry into the system.

After the election period, SURFnet hands over all technical votes it collected to TTPI. TTPI computes the (keyless) MDC hash values over the technical votes, yielding values that can be matched against the MDC values in the tables generated before the election. Then the results are tabulated.

The use of reference tables with MDC values of all valid technical votes permits automated validation of technical votes. In order for a technical vote to be valid, its MDC hash value needs to be in the pre-election reference table. Votes that do not comply with this rule are marked as invalid and logged with a reason code. Figure 6 shows this process in detail.

D. Analysis

Security: As with the Estonian system, a downloaded voting client makes it vulnerable to infection or substitution with malicious software. An infected or a rouge clone of the client could read the DES key Ki from voter i as she enters it, calculate and send the technical vote for a preferred candidate j by computing MACKi(CAN_IDj) while displaying the technical vote MACKi(CAN_IDl) for the voter's true choice, candidate l. Neither the voter nor any other involved party can detect this fraud during election. Although the voter can detect this fraud when the MDC values of technical votes are published, she cannot prove that the vote has been altered.

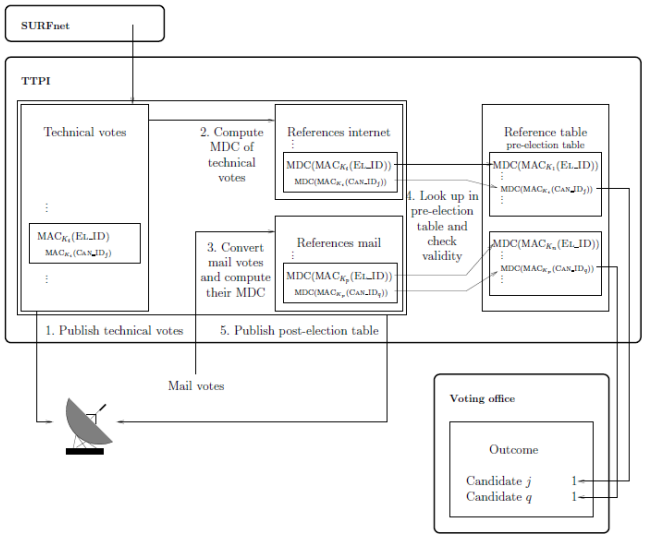

Fig. 6. Post-election phase in RIES [22]

No server-side party could add or alter votes without access to the DES keys. SURFnet would need the key of a voter to compute valid technical votes, and modifications by TTPI could have been detected by matching technical votes recorded by SURFnet with technical votes used by TTPI. However, we do not know whether this consistency check was conducted. TTPI could not delete votes without this being noticed if the SURFNet files were intact. As the single point of recording, SURFnet could have deleted technical votes without notice: SURFnet had the data to compute the MDC hashes on each vote they receive, and could combine this data with the pre-election tables and delete technical votes without detection. Again, we argue that this is a possible flaw in the design which needs to be articulated, and do not doubt the goodwill of any involved party.

It was possible to check that only valid votes were counted, as MDC values of all valid votes were published in the pre-election reference table and invalid votes would not have lead to any MDC value published. However, inaccuracies could have been introduced in at least two ways: First, The election design does in principle allow TTPI to conduct an incorrect assignment of MDC values to candidates in the pre-election table so that technical votes would not be counted as cast. Validation of the assignment requires knowledge of the voters' keys, which only the voters have after the tables are generated. Thus, it is important that the MDC calculation be reviewed carefully. While each voter has the data needed to confirm their candidate's MDCs and the resulting technical vote, the algorithm is sufficiently challenging to make this unlikely without third-party tools.

Another weakness arises from how RIES was used in the 2006 Dutch elections: Mail votes were transformed into Internet votes and then integrated into a single process. A mail vote and its technical representation are not seen by SURFnet, allowing votes to be added, deleted, or altered at TTPI. The technical and organizational protections afforded to Internet votes do not protect mail ballots from tampering.

We recognize two concerns about access. As the user's voting documentation is distributed by mail, ineligible voters might accidently or intentionally acquire a valid DES key. As SURFnet was a single point of vote recording, they had to take technical countermeasures against DoS attacks in order to ensure that eligible voters could vote.

There are two important privacy issues in the 2006 Dutch elections: The deletion of user keys by TTPI, while appropriate, is not sufficient to guarantee privacy. Complete privacy requires deletion of linkage between voter identity on one hand and the key, technical votes, and MDC hashes on the other. In addition, as the voter has access to her technical vote, it is possible to coerce them into producing it. This in turn can be combined with their DES key to determine their selections. Thus vote buying and coercion is possible.

Usability: While voter registration and vote casting appear simple, the validation of votes is not. To validate that their vote was properly recorded, voters need to determine the MDC hash of their technical vote. Reference [22] reports complaints about the complexity of vote-checking. Enabling voters to validate their recorded vote encourages them to do so, but a highly complex validation procedure dissuades them from using this feature. This in turn may discourage them from participating in future electronic voting, even though paper ballots have no verification at all. As noted earlier, the pre-election verification of MDCs values was similarly complex, and therefore unlikely to be used. A third concern refers to the procedure that was necessary to validate the overall correctness of the counting. The description of the laborious procedure [22] shows that the validation was beyond the capabilities of most voters.

Verifiability: The RIES design provided a mechanism for voters to verify that their votes were counted as cast through the published post-election tables. This assumes that that the voter was both willing and capable to compute the MDC value of his/her technical vote. In the case of an erroneously deleted vote, however, it was difficult to prove that she actually voted if SURFnet did not support the claim. Post-election review of correct totals depended on the accuracy of the post-election tables. As post-election tables included transcribed mail votes that were not in the pre-election publication, there was no way to verify completeness or correctness from the tables alone.

During the election, OSCE observers visited the Netherlands and provided a general report and a supplementary report dedicated to electronic voting components and processes [33]. This report mentions that the RIES system had previously been used in the 2004 elections for two regional water control boards. We found no official report on these earlier elections, but fortunately [33] performed an independent assessment of the 2004 RIES use. For the election under this review there do not seem to be any auditing reports available publically, so we cannot review how the audit was conducted.

Transparency: While the election design of RIES is transparent, large parts of the technological infrastructure, organizational election processes, and auditing are not. We found no information on the specific architecture used in the 2006 parliamentary elections and, in contrast to the client-side

JavaScript code, the software used on the SURFnet server(s) and the TTPI server(s) is not open to review [22]. We are also concerned about the election procedures themselves. As one firm generates and distributes keys, computes technical votes, and creates the pre- and post-election reference tables, we regard it as most crucial that these procedures are fully transparent. The detailed specification for the deletion of keys, for example, has been deemed security sensitive and classified as confidential. The OSCE report [33, p. 13] concludes that *"[…]far too many details of the electronic voting systems […] remain inaccessible to the public."*

E. Conclusion

Although the theoretical system provided by Robers [37] provides a promising approach for conducting secure and verifiable Internet elections, the application of RIES in the 2006 parliamentary elections reveals deficiencies in terms of security, usability, verifiability, and transparency. This election demonstrates that the inclusion of sound theoretical concepts is insufficient to conduct high-quality Internet elections. The powerful role of external vendors without transparency and the laborious efforts to achieve practical verifiability raise concerns about this experience.

Rijnland started their development by asking a third party to identify the security risks involved with setting up an Internet voting system. The results included that many risks involved in voting by Internet are not higher than in voting by ordinary mail and that risks typical to Internet settings such as DDoS attacks and Trojan horses on client machines can be successfully addressed with procedural countermeasures for the specific situation of Internet voting [33]. The report itself is not available for public review.

Interestingly, the Dutch government has eliminated Internet voting from its 2008 Water Board elections and the 2009 EU Parliament elections. The Dutch Ministry of the Interior decided in 2008 to reject electronic voting machines because of concerns about their security.

VI. Cantonal Referendum in Switzerland Dutch Elections to the House of Representatives

This case study refers to the 11 March 2007 cantonal referendum in Neuchâtel/Neuenberg, Switzerland, where 1,538 voters used Internet voting, representing 1.44 % of the population and 2.54% of votes cast.

A. Electoral environment

"Vote électronique" is part of a larger portal (Guichet Unique, GU) that offers a set of e-Goverment services. Each GU registrant receives a user code and a password by mail. For the 2007 Neuchâtel election, voters could choose between three voting modalities: she could vote traditionally, by mail, or by Internet through the GU portal. In order to prevent voters from casting multiple ballots, votes were stored in a central register after successful submission, as discussed below.

B. Design

The design of the Internet voting system is documented publically only at a descriptive level [36], [40]. According to [36], parts of the remote voting system Pnyx [41] of the company Scytl are used. Pnyx is proprietary software and closed to public scrutiny. To counter the (now) obvious concerns associated with proprietary software, an independent review team received access to the Pnyx.core ODBP 1.0 voting software. We use the review team's reports [11] on Pnyx components used in the Neuchâtel election: the Voting Server, the Voting Proxy, and the Ballot Box [36]. Other important elements of the packaged Pnyx voting protocol, such as are cryptographic key pairs for the voters, the technological infrastructure used at polling places, and Voter Verified Paper Audit Trails are not used in the Neuchâtel election. These functions are performed through election-specific customizations, out of the scope of the Pnyx core. Figure 7 presents the design of the Internet Voting System used in the Neuchâtel election as we have reconstructed it.

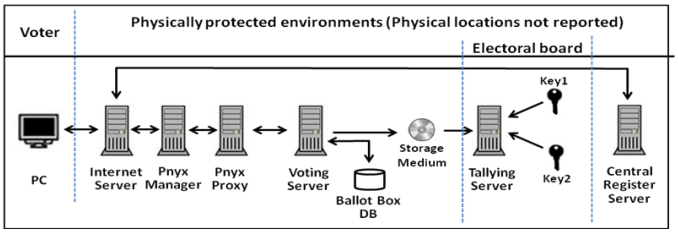

Fig. 7. Architecture of the Neuchâtel Internet election system

The integration of proprietary packaged and custom software creates a somewhat more layered and obscure architecture than in the cases discussed earlier. Voters use a digitally signed Java applet as their Voter Application. The vote is transmitted to the Internet Server, which communicates with the Central Register. Once this process completes, the vote is transferred to the Pnyx Manager. Reference [36] identifies the Pnyx Manager's responsibility as basic configuring of the votes, but does not provide any further explanation. According to [11], the Pnyx Proxy is primarily responsible for relaying communications between the "Voting Client" (here, the Pnyx Manager) and the Voting Server, which in turn stores the encrypted votes. When the Electoral Board provides the appropriate keys, the Tallying Server opens the digital Ballot Box and the ballots are decrypted and tallied.

C. Electoral processes

As with the system design, some parts of the election processes are different from the standard Pnyx and are not documented in detail. Unclear parts are marked with a question mark in Figure 8, which shows the key electoral processes as we understand them.

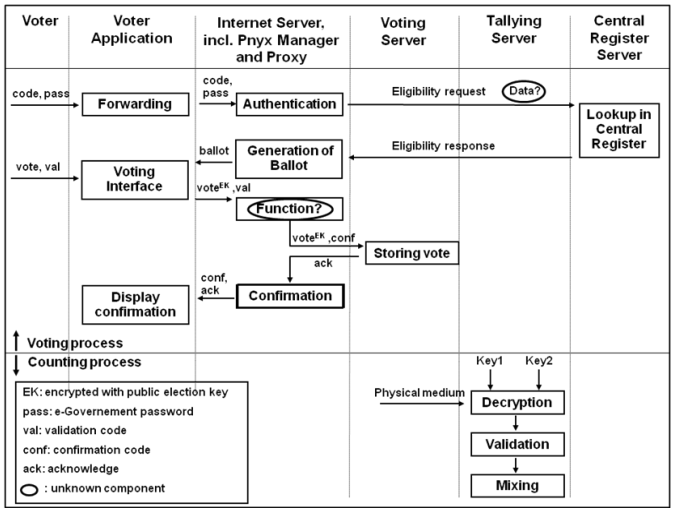

Fig. 8. Internet voting process in Neuchâtel Internet election system

In the pre-election period, one cryptographic public key pair (*pu*,*pr*) is generated. The public election key *pu* is used by the Voter Application for the encryption of votes. The private key *pr* is decomposed into two parts, which are stored onto digital cards. Each card is password-protected, with the card-specific password being chosen by a member of the electoral committee. The cards are then sealed and stored in a physically protected environment.

Voters registered at GU received an election- and voter-specific validation and confirmation code before the election. To cast a vote, the voter logs in the GU portal. The Internet Server validates the voter's eligibility through a request to the Central Register Server and generates an individual ballot, which is presented through a Java Applet on the voter's PC. The voter marks the electronic ballot and confirms her vote with her validation code. The vote is encrypted with the public election key and sent together with the validation code to the Internet Server. The Internet Server reviews the confirmation code of the particular user and sends the encrypted vote and the confirmation code to the Voting Server, which stores them. The description of the Internet Server does not provide any information on how it determines the confirmation code and where it is stored. We assume that it sends a request to the Central Register Server.

The Voting Server acknowledges the receipt of the encrypted vote. Finally the Internet Server sends the confirmation code and the acknowledgement back to the Voter Application. The voter can validate the server's authenticity by matching the returned code with the confirmation validation code that was sent to him/her prior to the election.

After the election, the encrypted vote and confirmation code pairs are exported from the Voting Server to an external storage medium and then imported into the Tallying Server. The electoral committee members appear at the location of the Tallying Server and provide their passwords to access the digital cards that hold the decomposed private election key. The key is then assembled and used to decrypt those votes that are linked to a confirmation code that is related to an eligible voter. Votes that are not validated are removed.

To perform its function, the Tallying Server appears to need the list of eligible voters and their confirmation codes. This validation occurs while voter identities are still linked to the encrypted votes. The validated votes are separated from the confirmation codes and decrypted with the assembled election key. Finally the decrypted votes are shuffled before they are published. Confirmation codes of all validated votes are also published; by randomizing the order presentation a link between confirmation codes and votes cannot be discerned.

D.Analysis

Security: The pre-election procedures appear to be adequate to secure the private key. However, no information is available about the key decomposition procedure, the number of digital cards, and the mapping of partial keys onto cards. As confirmation codes are published after the vote, an individual can confirm that their ballot was part of the final tally. However, the vote might have been altered and counted other than cast in several ways. As the Voting Application is executed on a PC, it is vulnerable to various local threats, as discussed in the earlier cases [25]. In the absence of a digital signature, the vote can be replaced by another vote encrypted with the public election key by malicious software components embedded in the voting software or any other software that is executed on the Application Server, Pnyx Manager, Pnyx Proxy, or Tallying Server. The voter has no mechanism to detect the alteration of her vote if the confirmation code remains intact. Similarly in the absence of careful monitoring, server-side software components could add votes with confirmation codes for voters who have not actually voted and block access to voters whose codes were compromised.

We did not find any information on how the communication between the Java Applet and the Internet Server is secured, or how and when how the Central Register Server records each to prevent any further ballots cast by the particular user.

The Tallying Server apparently validates the encrypted votes by checking whether the corresponding confirmation codes are assigned to eligible voters. Assumed that the software on the Tallying Server accomplishes this task correctly, it is assured that only eligible voters can vote. However, we have no information on whether the Pnyx component is used or any other software.

The system design limits each eligible voter to a single ballot, as a successful vote is stored until counting. However, this concept might be compromised during implementation. We cannot assess this any further, as the program code is neither published nor has been certified by an independent body.

In contrast to the Estonian election, the voter is not provided any option to overwrite his/her vote. As a result, vote buying and coercion might occur. We do not know whether the election system was protected against DoS attacks. Reference [11] says that the Pnyx system has no apparent DoS countermeasures. This is a concern, though such measures may be part of the GU environment.

The voter's confirmation code and encrypted vote remain linked to each other until the votes are decrypted on the

Tallying Server. It is therefore possible to link the decrypted vote and the voter, who then becomes vulnerable to insider threats or malicious software.

While the Neuchâtel election process takes important steps towards security, we remain concerned about the possibility of malicious election software and insider attacks. We also note that we have not uncovered any published evidence of a formal proof of the security protocols employed by Pnyx or within the GU portal. Again, the concerns raised by [11] about the implementation of the Pnyx system, including weaknesses in the bit strength of cryptographic schemes and in the shuffle procedure concern us.

Usability: In order to cast a ballot electronically, the voter only needs to have registered with GU and the election-specific validation code sent by mail. Any web browser that is Java-enabled can access the portal without additional hardware or software No complaints on the voting interface are reported in the literature.

Verifiability: In contrast to the full Pnyx.core ODBP 1.0 system that provides for Voter Verified Paper Audit Trails, the Neuchâtel election does not allow voters to verify whether their votes were counted as cast and whether the overall tally is correct. As the published confirmation codes are separated from the votes, voters can verify that their ballot was received, but does not demonstrate that its contents were tallied properly. Although the Pnyx system keeps various electronic logs [41] we do not know whether they are used and analyzed during the Swiss election in order to track the integrity of cast votes.

Reference [40] reports that in 2005 four test elections and three official elections were conducted and that two external security audits were conducted but the reports are not published. Interestingly, [11] reports that the canton Neuchâtel performed an internal security audit of the Scytl Pnyx.core software, but has not made their report available to Scytl, in contrast to the Finnish group at the University of Turku, which made the report available on the Internet (http://www.vaalit.fi/uploads/5bq7gb9t01z.pdf).

Reference [40] also reports that the electoral committee members cast test votes, to check the integrity of the election system. While testing the correctness of intended functionality is vital, rigourous software testing requires a focused attempt to find flaws.

Transparency: We found only high-level descriptions of the Internet election system. The election was not monitored by an independent organization, such as the OECD. As mentioned in the discussion of verifiability, the canton Neuchâtel has not published their audit of the Pnyx system. Confidence in the system and process would be greatly improved with additional external review of election procedures, technological components and audit results.

E. Conclusion

The Neuchâtel e-voting system largely ignores threats through infected PCs, shows deficiencies in secrecy of votes, and is susceptible to vote buying and coercion. The election system also does not provide any substantial verifiability and transparency.

We understand that Switzerland and its cantons are particularly interested in conducting Internet-based voting, because the political system in Switzerland is based on direct democracy and provides several elections a year, with the majority of people casting their votes remotely by mail. We also understand that electronic elections are embedded in comprehensive e-Government initiatives, such as GU. The example of the Neuchâtel election shows that in the presence of so many elections even an electoral system at risk because of rudimentary security precautions and almost no transparency and verifiability can become entrenched. In June 2008, the canton Neuchâtel used the Internet-based voting system in official elections for the seventh time (http://www.admin.ch/aktuell/00089/index.html? lang=de& msg-id=19093&no_cj_c=0).

VII. Cross-case Analysis

There are some interesting contrasts and similarities among these three Internet voting implementations, which are summarized in Table I. In all three elections, there are important and possibly unaddressed security issues with the voting client software. Clients can be infected with malware or can be remotely controlled as a part of a botnet, and users must be cautioned against malicious substitution, particularly if the same client is used for multiple elections. Even in the Estonian election, where external card readers and a PKI are used, the voter actually did not know what happened on her PC, as the card reader did not have its own keyboard and display. While the SERVE report [25] presents important information in this regard, it may not have been consulted in time to affect the design of these elections.

All three elections have server side design weaknesses. Accuracy and privacy requirements are met only if the server programs function correctly and the particular parties behave properly. While we have no reason to doubt the integrity of the individuals involved, the security community accepts the principle that the integrity of a system should be largely based upon design and not on the implementation. Thus it is responsible to be respectfully concerned about apparent gaps and recommend the presentation of formal evidence of correct design and use. Precautions against Denial-of-Service attacks were not reported for any of the elections. In the case of the Dutch election, we found two additional severe security risks. The in-house transformation of mail votes to electronic ones is a weakness that could be exploited without careful oversight. In addition, the ability of voters to prove not only that they voted, but how they voted, may create opportunities for coercion. One relatively bright spot in these cases was the absence of major usability problems. We suspect that electoral officials considered the user interface of primary importance to acceptance by the voting public, and therefore paid great attention to this particular topic, save the absence of a Russian-language client in Estonia. Another exception was the difficulty in using vote verifiability reported by [22].

All three case studies showed deficiencies in verifiability. Voters could not verify whether their votes had been counted

as cast, and consequently could also not verify the correctness of the overall tally. Although the Dutch election allowed voters to validate the correct counting of their votes, the practical procedure for this was so complicated that it was dysfunctional. Beyond these weaknesses, the auditing and testing of voting procedures and technology by independent authorities was practically non-existent in all three elections. The Estonian case included a large set of log files, these files could have been manipulated. In the case of the Dutch elections, auditing was even more challenging because mail votes were transformed into electronic votes. Forensic audit trails were not available in any case.

Finally, all of the elections raise concerns about transparency. Large parts of the software was not open source, the documentation of the system and audit reports are unpublished, and in the Dutch case, the documentation of procedural details on the deletion of keys has been deemed security sensitive and classified confidential.

TABLE I
KEY WEAKNESSES OF ELECTION CASES

Requirements		**Elections**		
		Estonia	**The Netherlands**	**Switzerland**
Security	Accuracy	• Votes can be altered and deleted on client side (card readers have no keyboard and no display) and on server side • Votes can be added on server side	• Votes can be altered and eliminated on client side (software runs on unprotected PCs) • Votes can be deleted on voting server by ISP • Correctness on tally also depends on correctness of mail votes but cannot be checked by voters	• Votes can be altered on client side through software on unprotected PCs) as well on server side • Votes can be added on server side • Correctness depends on voting software that is closed source
	Democracy	• Implementation of authorization mechanism not analyzed by independent observers • No precautions against DoS attacks	• Assurance that eligible voters can cast only one depends on the integrity of vendor • No precautions against DoS attacks are reported	• Eligible voters cannot vote when their confirmation codes was abused on server side • No precautions against DoS attacks are reported
	Privacy	• Vote Storage Server keeps a link between the encrypted vote and the identity of the voter	• Secrecy of votes rests on private vendor • Voters can prove how they voted	• Secrecy of votes can be violated by server-side software
Usability		• Discrimination of Russian-speaking community (voting software available in Estonian language only)	• Verifiability procedures are for most voters practically impossible to conduct	--
Verifiability	Auditing of votes	• No Voter Verified Audit Trail • Voters could also not check whether the counting was correct	• Limited usability hindered voters to verify own vote • Merging mail votes with Internet votes makes it impossible to validate correctness of votes	• Voters cannot verify whether their votes were counted as cast and whether the overall tally is correct
	Auditing of voting procedures and voting system/ Forensic audit trails	• No certification of system • No end-to-end accuracy test • Auditing conducted by a private company • Log files can be manipulated unnoticed • No reliable forensic audit trail available	• No auditing or test reports are available • No certification of system • Forensic audit trail focuses on ballot revocations	• Test elections and security audits were conducted but results are unpublished • Test votes cast by committee members are almost useless • Existence of forensic audit trail is unclear
Transparency		• E-voting system, including software and documentation, is not transparent to the public and to independent security experts • Final auditing report is not published	• Reports on the concrete technologic infrastructure, organizational election processes, and auditing are not published • Server-side software is closed source	• Only high-level descriptions of election procedures and technological components are available • Audits are unpublished • Usage of log files is unclear

VIII. CONCLUSIONS

The analysis of three recent large-scale Internet elections conducted in Estonia, the Netherlands, and the Swiss canton Neuchâtel revealed several apparent deficiencies in terms of security, verifiability, and transparency. Responsible authorities may have been unaware of these problems or they may have been confident of their defenses, or they may not

have disclosed all of the details of their work. In addition, the absence of public reporting of successful attacks have been detected or reported does not mean that none occurred [25, p. 30]. It should also be noticed that with Internet-based government services becoming increasingly attractive, the opportunity and returns from attacks on voting systems also increase.

In contrast to Estonia and some cantons in Switzerland, including Neuchâtel, only the Dutch authorities publically recognized the consequences of severe security issues of Internet voting system and stopped Internet voting in the Netherlands. One of the most urgent tasks for e-voting scientists and security researchers is the identification and presentation of weaknesses of Internet election systems where in other countries where Internet voting is still under consideration. Responsible authorities should also be shown that testing, auditing, and providing publicly available reports takes time and needs substantial funding.

Public confidence in the electoral process depends on information and advice provided by security experts. The fact that overall only very few complaints of citizen or citizen initiatives have been reported in the face of the problems we identify shows that voters tend to accept missing transparency in Internet voting systems, to trust authorities, and to underestimate security threats. Sadly, we expect that this tacit confidence may be easily dissipated in the face of a failed election. Deficiencies of proposed Internet voting systems should be made transparent to voters in advance.

We conclude from our analysis that future Internet voting initiatives should address some technological, organizational, and administrative properties which were apparently neglected in our three cases.

Technology: The correctness of security and verifiability elements should depend on the design of the system and not assumptions of the proper implementation of programs or of organizational procedures, When designing an Internet voting system, well-known and well-understood e-voting protocols should be used. The design should also integrate procedures for creating log files and forensic audit trails. In order to increase security and verifiability, end-to-end schemes should be used so that voters do not have to rely on the integrity of election parties. If the public is required to procure voting hardware, the devices should be certified against tampering and have the ability to capture, display and protect its information from manipulation during its transfer to the ultimate host. Internet election design should include precautions against DoS attacks on server side, an increasingly popular and viable attack mode.

Voters should be able to verify that their votes have been counted as cast. Although vote buying and coercion cannot be prevented, some systematic attacks may be prevented when voters receive proof of voting that cannot be reliably decoded by another person. The Estonian approach, which allows voters to overwrite their Internet votes as often as desired, should be considered.

Organization: All of the case studies demonstrate the extraordinary responsibilities placed on electoral authorities and the technology providers supporting them. We suggest that Internet election technical operations be distributed to at least two independent parties. We argue further that e-voting providers be accountable for their code and procedures to parties competent to judge and review their work. E-voting providers must be held to stricter standards than e-Commerce vendors, and organizations with e-Commerce experience are not automatically qualified to operate Internet election technology. As elections are a core part of democracy, it seems to be too risky to rely on private organizations only, which are usually primarily interested in profit. Our confidence in election authorities increase with the deployment of independent third party reviews, independent of the organizations involved.

Administration: The numerous technical weaknesses uncovered in our review suggests strongly that independent security experts should be consulted in advance of the design and implementation of Internet voting, that comprehensive tests of the full system be conducted with an eye towards identifying points of failure, and that the system and the overall election be audited by independent e-voting experts. It also seems reasonable to follow the OSCE recommendation [33], to cast relatively large numbers of test ballots during the election, where these test ballots are cast in a way that is indistinguishable from regular ballots. In addition, we argue in favor of full transparency of design, implementation, election procedures and test and auditing reports: While obscurity can protect the systems from some exposures, it is almost impossible to hide implementation details in the long run. A salient example is the accidental publication of Diebold voting machines source code on the Internet in 2003 [38].

As accounting for technological, organizational, and administrative properties involves comprehensive and non-trivial tasks, we argue that the elaboration and deployment of e-voting recommendations, such as [13] and [30], or even better, standards should be striven for, which can serve as overall guides for persons responsible for Internet elections. These standards will require regular revisions, however, as the inevitable progress of technology will introduce new vulnerabilities and enable the exploitation of old ones.

This study has some important limitations of its own. First, we are aware that other researchers would have probably selected other elections, for example one of the local elections in the U.K. or the UMP Presidential Primary Election in France, and that our choices may not be representative. Second, we acknowledge that the use of incomplete public records to reconstruct technical architectures will make some elements appear neglected. In the case of sensitive applications, such as electoral systems, the obscurity of detail may be intentional to reduce exposure to attack. Some of the concerns that we raise may well be answered to the satisfaction of the various electoral commissions in private, and thus we do not question the accountability of these bodies. Our review illustrates potential gaps for consideration rather than judgment upon the efforts of government administration. Third, our study does not include any cost-benefit analysis. Some of the identified limitations in authorities' efforts to

increase security and transparency may be rooted in financial restrictions, instead of being based on ignorance or neglect. Finally we recognize that the evolution of e-Voting architectures is taking place in parallel with active attempts at implementation. The planning and design of a voting scheme occurs months or years before its implementation, so it takes time for new research and experience to be seen. Comparative case study is one mechanism to highlight lessons be integrated in new electoral cycles.

REFERENCES

[1] A. Acquisti, "Receipt-Free Homomorphic Elections and Write-in Voter Verified Ballots, "Technical Report 2004/105, International Association for Cryptologic Research, May 2, 2004, and Carnegie Mellon Institute for Software Research International, CMU-ISRI-04-116, 2004.

[2] J. Benaloh, "Verifiable Secret-Ballot Elections," PhD thesis, Yale University, 1987.

[3] J. Benaloh and D. Tuinstra, "Receipt-free secret-ballot elections," in Proc. of the 26th Annual ACM Symposium on the Theory of Computing, 1994, pp. 544-553.

[4] M. Bishop and D. Wagner, "Risks of E-Voting," Communications of the ACM, vol. 50, no. 11, 2007, p. 120.

[5] J. Borland (2007, March 2), "Online Voting Clicks in Estonia,"Wired [Online]. Available: http://www.wired.com/politics/security/news/2007/03/72846?currentPage=all.

[6] B.O. Brachtl, D. Coppersmith, M.M. Hyden, S.M. Matyas, C.H. Meyer, J. Oseas, S. Pilpel, and M. Schilling (1990, March 13), "Data Authentication Using Modification Detection Codes Based on a Public One-Way Encryption Function," U.S. Patent Number 4,908,861.

[7] California Internet Voting Task Force, "A Report on the Feasibility of Internet Voting", 2000. [Online] Available: http://www.sos.ca.gov/elections/ivote/final_report.pdf.

[8] D. Chaum, " Elections with unconditionally- secret ballots and disruption equivalent to breaking RSA, " in Proc. of EUROCRYPT '98, LNCS, vol. 0330, Springer, Berlin, Heidelberg, 1988, pp. 177–182.

[9] D. Chaum, "Secret-Ballot Receipts: True Voter-Verifiable Elections," IEEE Security & Privacy, vol. 2, no. 1, 2004, pp. 38–47.

[10] D. Chaum, "Untraceable Electronic Mail, Return Addresses and Digital Pseudonyms," Communications of the ACM, vol. 24, no. 2, 1981, pp. 84-88.

[11] M. Clarkson, B. Hay, M. Inge, A. Shelat, D. Wagner, and A. Yasinsac (2008, September 19), "Software Review and Security Analysis of Scytl Remote Voting Software," [Online] Available: http://www.eecs.berkeley.edu/~daw/papers/scytl-odbp.pdf.

[12] C. Coggins, "Independent testing of voting systems", Communications of the ACM, vol. 47, no. 10, 2004, pp. 34-38.

[13] Council of Europe, Legal, Operational And Technical Standards for E-voting - Recommendation Rec (2004)11 And Explanatory Memorandum (Legal Issues), 2004.

[14] L.F. Cranor and R.K. Cytron, "Sensus: A Security-Conscious Electronic Polling System for the Internet," in Proc. of the 37th International Conference on System Sciences, Hawaii, 1997, pp. 561-570.

[15] H. Deutsch and S. Berger, "Voting systems standards and certifications," Communications of the ACM, vol. 47, no. 10, 2004, pp. 31-33.

[16] Estonian National Electoral Committee (NEC) (2005). E-Voting System – Overview [Online]. Available: http://www.vvk.ee/public/dok/Yldkirjeldus-eng.pdf.

[17] Estonian National Electoral Committee (NEC) (2007). Parliamentary elections 2007: Statistics of e-voting [Online]. Available: http://www.vvk.ee/english/Ivoting_stat_eng.pdf.

[18] European Union Democracy Observatory (2007, July 31), Report for the Council of Europe: Internet Voting in the March 2007 Parliamentary Elections in Estonia. [Online]. Available: http://www.vvk.ee/english/CoE and NEC_Report E-Voting 2007.pdf.

[19] A. Fujioka, T. Okamoto, and K. Ohta, "A Practical Secret Voting Scheme for Large Scale Elections," in LNCS, vol. 0718, Springer, Berlin,Heidelberg, 1993, pp. 244-251.

[20] J. Gilberg, "E-VOTE: An Internet-based Electronic Voting System: Consolidated Prototype 2 Documentation," Technical Report e-VOTE/WP-7/D7.4/3.0/29-05-2003, May 2003. [Online] Available: http://www.instore.gr/evote/evote_end/htm/3public/doc3/public/public_deliverables/d74/ Consolidated_Docu_final.zip.

[21] M. Hirt, and K. Sako, "Efficient receipt-free voting based on homomorphic encryption," in LNCS, vol. 1807, Springer, Berlin, Heidelberg, 2000, pp. 539–556.

[22] E. Hubbers, B. Jacobs, and W. Pieters, "RIES - Internet Voting in Action," in Proc. of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05), Washington, DC., 2005, pp. 417-424.

[23] Internet Policy Institute, "Report of the National Workshop on Internet Voting: Issues and Research Agenda", 2001. [Online] Available: http://news.findlaw.com/cnn/docs/voting/nsfe-voterprt.pdf.

[24] M. Jakobsson, A. Juels, and R.L. Rivest, "Making mix nets robust for electronic voting by randomized partial checking", in USENIX Security Symposium, 2002, pp. 339-353.

[25] D. Jefferson, A.D. Rubin, B. Simons, and D. Wagner (2004, January 20), A Security Analysis of the Secure Electronic Registration and Voting Experiment (SERVE). [Online]. Available: http://www.servesecurityreport.org.

[26] A. Kiayias, M. Korman, and D. Walluck, "An Internet Voting System Supporting User Privacy," in Proc. of the 22nd Annual Computer Security Applications Conference (ACSAC '06). 2006, pp.165-174.

[27] R. Krimmer, S. Triessnig, and M. Volkamer, "The Development of Remote E-Voting Around the World: A Review of Roads and Directions," in LNCS, vol. 4896, Springer, Berlin, Heidelberg 2008, pp. 1-15.

[28] R. Mercuri, "Electronic Vote Tabulation Checks & Balances," Ph.D. dissertation, School of Engineering and Applied Science of the University of Pennsylvania, Philadelphia, PA., 2001.

[29] J. Mohen and J. Glidden, "The Case for Internet Voting," Communications of the ACM, vol. 44, no. 1, 2001, pp. 72-85.

[30] NIST (2009, May 27), "Draft Voluntary Voting System Guidelines Version 1.1," Prepared for the Election Assistance Commission. [Online] Available: http://www.eac.gov/program-areas/voting-systems/voting-system-certification/2005-vvsg/draft-revisions-to-the-2005-voluntary-voting-system-guidelines-vvsg-v-1-1.

[31] OSCE (2007, June 28), OSCE/ODIHR Election Assessment Mission Report in the 2007 parliamentary elections in Estonia. [Online] Available: http://www.osce.org/documents/odihr/2007/07/25385_en.pdf.

[32] OSCE (2006, November 22), OSCE/ODIHR Election Assessment Mission Report, The Netherlands, Parliamentary elections. [Online] Available: http://www.osce.org/odihr-elections/item_12_22041.html.

[33] OSCE (2006, December 29), An Assessment of Electronic Voting in the Netherlands 22 November 2006 Parliamentary Elections.

[34] S. Peisert, M. Bishop, and A. Yasinsac, "Vote Selling, Voter Anonymity, and Forensic Logging of Electronic Voting Machines," in Proc. of the 42nd Hawaii International Conference on System Sciences, 2009, pp.1-10.

[35] D.M. Phillips and H.A. von Spakovsky, "Gauging the Risks of Internet Elections," Communications of the ACM, vol. 44, no. 1, 2001, pp. 73-85.

[36] République et Canton de Neuchâtel, "E-government and electronic voting," [Online] http://www.coe.int/t/e/integrated_projects/democracy/Democracy_Forum_2008/Rota_Presentation_FFD08.pdf.

[37] H. Robers, "Electronic elections employing DES smartcards," Master's thesis, Delft University of Technology, Delft, The Netherlands, 1998.

[38] A. Rubin, "Brave New Ballot," Morgan Road Books, New York, USA, 2006.

[39] K. Sako and J. Kilian, "Receipt-free Mix-type Voting Scheme," in LNCS, vol. 921, 1995, pp. 393-403.

[40] Schweizerischer Bundesrat (2006, May 31). Bericht über die Pilotprojekte zum Vote électronique, pp. 5479-5485. [Online] Available: http://www.admin.ch/ch/d/ff/2006/5459.pdf.

[41] Scytl Secure Electronic Voting (December 2005), "Pnyx.core: The Key to Enabling Reliable Electronic Elections A Description of Scytl's Cryptographic e-Voting Security Software [Online] Available: http://www.scytl.com/pdf/PNYXDREWhitePaper.pdf.

[42] United Nations, "The Universal Declaration of Human Rights", 1948. [Online] Available: http://www.un.org/en/documents/udhr/index.shtml.

[43] R.K. Yin, Applications of case study research, 2nd ed. Thousand Oaks, London, New Delhi: Sage Publications, 2003.

[44] R.K. Yin, Case study research: design and methods. 3rd ed. Thousand Oaks, London, New Delhi: Sage Publications, 2003.