INDEPENDENT VOTER VERIFIABILITY FOR REMOTE ELECTRONIC VOTING

Jordi Puiggalia and Victor Morales Rocha
Scytl Secure Electronic Voting, Tuset 20 1-7 Barcelona, Spain

Keywords: e-Voting, Remote Electronic Voting, Cryptographic Voting Protocols, Voter Verifiability, Voting Receipt.

Abstract: Most of the current efforts to implement voter verifiability methods for electronic voting are not suitable for remote electronic voting. Moreover, the remote voting verifiability methods proposed to date are inefficient, do not allow the verification of the presence of the votes after they have been decrypted or they sacrifice voter privacy requirements in order to accommodate the “cast as intended” voter verification objective. We propose a voter verifiability method for remote electronic voting that addresses each of these issues. The method is based on the implementation of cryptographically protected voting receipts and is complemented by the use of an independent verification application which is easy to audit and certify.

1 INTRODUCTION

Electronic election processes are not easy to carry out, especially when remote communication channels are used to cast votes. There is a natural distrust towards remote electronic voting due to the security risks that it faces. Such risks must be mitigated in order to provide confidence to voters and politicians.

Independent voter verifiability must be considered an important element for providing reliability to an electronic election process, allowing a voter to verify the correct handling of his/her vote, from the moment that it is cast until it is counted.

2 VOTER VERIFIABILITY

The main objective of voter verification proposals is to provide a means to (i) verify that voter intent is accurately stored by the voting system (the so-called “cast as intended” requirement) and (ii) audit the accuracy of the records used to provide the election results (the so-called “counted as cast” requirement). Moreover, voter verification must not open up the possibility of coercion or vote-buying. The issuing of voting receipts can cause voter coercion or vote buying if the receipts contain proof of the chosen vote. Therefore, a voting receipt must allow the voter to verify the presence of his/her vote in the final tally, but not its contents.

Nowadays, most voter verifiability proposals, e.g. (Mercuri, 2002), (Riera, 2003), (Chaum, 2004), (Neff, 2004), are focused on providing voter verification in poll-site electronic voting environments, and are not suitable for being implemented in remote e-voting scenarios.

The voter verifiability proposals for remote voting schemes make use of voting receipts to verify the correct recording of the voter intent, e.g. (Sako, 1994), (Cranor, 1997), (Malkhi, 2002). However these proposals can pose some privacy risks, since these receipts can be used to discern the voter intent. In this paper we will propose the combination of independent verification systems and voting receipts to achieve voter verifiability objectives without compromising voter privacy.

3 INDEPENDENT VOTER VERIFIABILITY

Our proposal consists on using secret and tamper-proof voting receipts generated by a set of cryptographic processes executed on an easy-to-audit application.

For implementing our proposal, we assume that each voter has a unique asymmetric RSA key pair. The generation, distribution and management of

Puiggalia J. and Morales Rocha V. (2007).
INDEPENDENT VOTER VERIFIABILITY FOR REMOTE ELECTRONIC VOTING.
In *Proceedings of the Second International Conference on Security and Cryptography*, pages 333-336
DOI: 10.5220/0002120303330336

these keys can be implemented following a PKI approach.

3.1 Verification Components

Voter confidence is based on two elements, an independent verification application and a validated voting receipt.

3.1.1 Independent Verification Application

Following the approach of the Independent Verification systems described in (VVSG'05). We have divided the voting process into two different components: the selection of the options and the verification of these options.

The selection of the options consists of the presentation to the voter of the races and the available candidates and the selection by the voter of the preferred candidates.

The verification of the selection options is the last process implemented by the voter before casting his/her vote. In this process a summary of the voting options selected by the voter must be shown for his/her review. This process waits for the voter's confirmation before casting the vote. If there is any problem with the selection of the voting options, the voter can detect this problem at this stage and cancel the casting of the vote. Therefore, ensuring the accuracy of this verification process in turn ensures that the vote has been cast as intended.

Based on this premise, our approach proposes the existence of an independent application that (i) performs this verification process and (ii) executes the cryptographic processes that protect and then issue the voting receipt. We will call this application Independent Verification Application (IVA) and it must be audited and certified by an independent entity. This certification can be achieved by the digital signature of the code or the publication of the checksum of the verified code in a public area. Since the audit only reviews the accuracy of the verification process, the complexity of the audit process is reduced. An audit of the vote selection process is not required. Furthermore, since the verification process only needs to display the options selected by the voter, this application could remain unchanged even if the selection process is modified.

Voters could validate the IVA application by verifying its digital signature or examining the checksum. A possible implementation of this application could be a digitally signed Java applet.

3.1.2 Voting Receipt

A voting receipt is issued to the voter to verify the correct handling of his/her vote during the decryption process. To achieve this objective, we propose a challenge approach: we will require the authorities in charge of the vote decryption process (e.g. the Electoral Board) to retrieve and publish the voting receipt contents. These receipt contents must only be known by the voter and therefore can only be retrieved if the vote is correctly decrypted. The steps for creating and validating this voting receipt are described below.

3.2 Verification Protocol Phases

The verification of the accuracy of the election comprises the following phases.

3.2.1 Verifying the Voting Application

Before the election process begins, the IVA application must be audited and certified by an independent software auditor. Once certified, a digital signature or checksum of the application is generated and published.

When voters access the voting platform, the IVA application (e.g., Java applet) is downloaded and executed in their browser. Using the published checksum or digital signature, voters can check the authenticity and integrity of this application. This process can be done by checking the contents of the browser pop-up window which displays the information regarding the signer of the voting application. This point is essential for the security of the voting procedure.

Through the accuracy check of the IVA application a voter can be sure that the cryptographic processes used to protect the vote and generate the voting receipt are the correct ones. This verification process allows voters to be sure that their votes are correctly recorded (i.e. "cast as intended").

3.2.2 Receipt Generation

Before casting the vote, the voting receipt is generated by the IVA application. The generation of this voting receipt comprises the steps of creating a unique receipt identifier, issuing a receipt request, validating the voting receipt, and displaying the voting receipt.

Creating a unique receipt identifier

The unique receipt identifier (R_{id}) is generated by using a pseudo-random number generator included in the IVA application. The voter could also participate in the generation of this number by providing some of its digits. In either case, we assume that the number of random digits contained in the R_{id} is large enough to prevent collisions with R_{id} generated by other voters. The reason that it is the IVA application that creates the receipt identifier is to prevent a malicious authority creating the same voting receipt for two different voters. Otherwise this authority could delete or modify votes with the same receipt identifier without being detected.

The receipt identifier must be kept secret (i.e. only known by the voter). We propose encrypting it along with the vote. Assuming that an Electoral Board has a private key to decrypt the votes, only this Electoral Board will be able to decrypt it. Using the Electoral Board public key (P_{EB}) to encrypt the vote and R_{id} pair:

$$V \equiv P_{EB}(\text{vote}, R_{id})$$

Creating the receipt request

To validate the voting receipt, the data used to generate this receipt must be sent to a Voting Service (ballot collector server). This data is collected in a token called the “receipt request”, R_r. The objective of this receipt request is to provide a proof of authenticity for the receipt without revealing the receipt contents. To this end, the IVA application masks the receipt id (M_r) and concatenates the masked information with the encrypted vote. The receipt signing request (R_r) is generated by digitally signing the concatenated data with the voter’s private key. The receipt signing request, the encrypted vote and the mask of the receipt id will be sent to the Voting Server.

The token, R_r, is generated as follows:

1. The voter computes the hash of the receipt identifier (R_{id}), the election identifier (E_{id}) and the voter identifier (V_{id}):
   $$M_r \equiv H(R_{id}|\, E_{id}|\, V_{id})$$
2. The voter creates R_r by signing M_r:
   $$R_r \equiv S_V(M_r)$$

Validating and signing the voting receipt

The Voting Server checks if the receipt signing request corresponds to the encrypted vote and to the masked receipt. If the voter has the right to vote, the information is stored in the digital ballot box, and a validated receipt signature is issued to the voter by the Voting Server. This validated receipt signature is generated by means of digitally signing the receipt id mask using a Voting Server private key. The validated receipt signature is returned to the voter, who can then print it along with the unique receipt identifier and the election data.

Upon reception of the receipt request, the Voting Service performs the following operations:

1. Check R_r's signature to verify its validity
2. Generate the Voting Receipt as
   $$R = S_{VS}(M_r)$$

Displaying the Voting Receipt

Upon reception of *R*, the IVA application will display, in a printable format, the following information:

- The electoral identifier, E_{id}
- The voter identifier, V_{id}
- The receipt identifier, R_{id}
- The receipt signing request, R_r
- The *R'*s signature

The two last fields (R_r, R) can be printed using positional notation codes or barcodes. Alternatively, the voting receipt could be recorded by the voter in a data storage device.

It is important to note that the voting receipt contains no information about the voting options. Therefore, this receipt does not facilitate the implementation of coercion practices, since the voter receipt does not discern the voter intent.

This voting receipt is resistant to voting receipt tampering and bogus receipt creation. The voting receipt is digitally signed by the Voting Service and therefore any alteration of its contents is easily discovered. Moreover, since each receipt request is also digitally signed by the voter, the Voting Service cannot create bogus receipt requests and consequently bogus voting receipts. Finally, voters cannot generate a valid voting receipt without interacting with the Voting Service. Therefore, the possibility of voter creation of bogus voting receipts is also eliminated.

3.2.3 Receipt Recovery

After closing the voting period, the Electoral Board carries out the decryption of the votes. At this point, we recommend the use of a Mixing process, like a Chaum-type Mixnet (Chaum, 1981), to break any correlation between the decrypted votes and their corresponding receipt Ids. Moreover, this Mixing process prevents the possibility of timing-attacks, in which an attacker monitoring the network could link voters with their respective votes.

Decrypted votes are counted and the list of receipts Ids published to allow the verification of the results.

3.2.4 Verifying Results

The verification protocol implemented by our scheme allows voters to verify that their votes did indeed reach the proper electoral authorities (note that this is the verifiability level found in conventional elections). In order to perform the verification process, this scheme needs both the voter's voting receipt generated during the vote casting period and the list of receipt Ids retrieved and published by the Electoral Board.

The voter looks for the receipt Id, contained in his/her voting receipt, in the published list of receipt Ids corresponding to all valid votes received and decrypted. With this kind of verification, the voter can check that his/her particular vote was provided as input to the counting process (i.e. satisfying the "counted as cast" requirement).

Voters whose receipt identifier does not appear in the published results can issue a public objection by presenting their voting receipt. Such an objection does not compromise the voter's privacy since a vote's contents are not needed to verify its validity.

Furthermore, individual verifiability can also identify (even miniscule) manipulations of the tally, including the case where only a small percentage of voters verify their own voting receipts. An example may serve to illustrate the effectiveness of individual verifiability in detecting general manipulations of the election results: In an election with 2,000 cast votes, only 30 voters would be required to verify the presence of their own votes in the tabulated results in order to achieve a more than 90% probability of detecting a manipulation of just 150 of the ballots. If the number of voters that verify their respective votes in this election would double (i.e. increase to just 60 voters), the probability for detection of vote manipulation would rise to more than 99%.

4 CONCLUSIONS

In this paper, we have presented a method which satisfies both verification requirements. The "cast as intended" requirement can be satisfied through an independent verification application. It involves components that allow voters to verify the voting application integrity, as well as the correct inclusion of their votes in the final tally and publication of the election results.

The verification of the votes presence after being decrypted (i.e. "counted as cast" verification) is achieved by using cryptographically protected voting receipts. These voting receipts are resistant to manipulations by voters since they are digitally signed by an authority (i.e. the Voting Service) that is under the control of the election authorities. The authenticity of the receipts can also be validated by using this digital signature, preventing voters from manipulating their voting receipts. Finally, bogus receipts cannot be generated by individuals since they require collaboration of the voter and the election authorities.

Furthermore, these receipts also maintain voter privacy and do not facilitate coercion and vote-buying practices since they do not reveal any information about the vote.

Finally, the voter verification approach that we propose also facilitates, with high probability, the detection of small vote manipulations by merely verifying a small percentage of the voting receipts.

REFERENCES

Chaum, David. 1981. "Untraceable electronic mail, return addresses and digital pseudonyms". *Comms. of the ACM*, Vol. 24 , Issue 2, pp. 84-88. February 1981.

Chaum, David. 2004. "Secret-Ballot Receipts: True Voter-Verifiable Elections". *IEEE Security and Privacy*, pp. 38-47. January-February 2004.

Cranor, L.F., Cytron, R.K. 1997. "Sensus: A security-Conscious Electronic Polling System for the Internet". *Proceedings of the Hawaii International Conference on Systems Sciences. 1997.*

Malkhi, D., Margo, O., and Pavlov, E. 2002. *E-voting without 'Cryptography', FC 2002.*

Mercuri, R. 2002 "A better ballot box?" *IEEE Spectrum Online*, October 2002.

Neff, Andy. 2004. "Practical High Certainty Intent Verification for Encrypted Votes" Retrieved March, 19, 2007, from http://www.votehere.com/vhti/documentation/vsv-2.0.3638.pdf

Riera, A., Puiggali, J., Brown, P. 2003. "Applied Cryptography Enabling Trustworthy Electronic Voting". *Workshop on Voter Verifiable Elections,* Denver, July 2003.

Sako, K. 1994. "Electronic voting scheme allowing open objection to the tally". *In IEICE Trans. Fund. of Electronics, Comm. Comp. Sci*. pp. 24-30.

VVSG'05 Electoral Assistance Commission. *Voluntary Voting Systems Guidelines*, February, 2006