Draft Chapter 4.

—

h1. HistorySummary of E2E Voting System Development (Doug, Poorvi, Olivier)

Origin

The phrase *end to end* is generally attributed to Saltzer, Reed, and Clark [1981], where it was applied to a class of arguments about computer networks: good system design ensures that—-regardless of the application and what assumptions can be made of its properties—-certain high-level properties are satisfied at the endpoints of a communication channel. As applied to voting, Jones [2002] translated this to the question "How close does the outcome of the election, as reflected in the official canvass, come to the actual intent of the voters who participated in the election?" The term is also used in Ben Adida's dissertation, Adida [2006], which states that the verifiability of a voting system is end-to-end if it is "preserved from start to finish, regardless of what happens in between". More specifically, the ability to verify correctness of the output of an end-to-end voting system does not depend on securing chain of custody, nor on checking the software or hardware used to carry out the election. That is, it does not depend on monitoring the path the votes take from the voter to the tally, or on assumptions regarding the security of that path. Rather, voting system design ensures that one may examine only the output (the tally, other data and a mathematical proof of tally correctness) to verify the election.

During the recent years, various authors provided various definitions of end-to-end verifiable voting systems. These definitions differ in their level and precision of formalism, and in their relation to specific voting systems. For instance:

- Lowry and Vora [2009] define end-to-end independently verifiable voting systems as ones where an honest observer may determine whether the outcome "correctly represents the votes cast by voters".
- Kremer et al [2010] define end-to-end verifiability, using the applied-pi calculus as modelling framework, as the conjunction of three properties: (i) individual verifiability (voters can verify that their vote was captured properly); (ii) eligibility verifiability (anyone can verify that all the ballots included in the tally come from eligible voters) and (iii) universal verifiability (anyone can verify that the tally is correctly computed from those votes).
- Popoveniuc et al [2010] provide a somewhat technical definition of end-to-end verifiable elections, as ones that pass a set of six checks on whether: (i) presented ballots are well-formed; (ii) cast ballots are well-formed; (iii) ballots are recorded as cast; (iv) ballots are tallied as recorded; (v) consistency: whether the collection of ballots subjected to the check for (iii) is the same as that checked for (iv); (vi) each recorded ballot is subjected (by at least one voter) to the check of (iii).

- Küsters et al [2010] define verifiability as the guarantee that auditors will only accept election results that are consistent with the votes cast by eligible voters. They also advocate the requirement of a stronger notion of verifiability, that is, accountability, which requires that, when the audit process of a verifiable system results in a failure, it should also point to the election participant (voter, authority, ...) that misbehaved. Their definition is instantiated both in a symbolic and in a computational model.

Background

Research in the use of cryptography to design voting systems with end to end properties dates back to the early eighties. Much literature exists on desirable properties of voting systems. Inspired from Kremer and Ryan [2005], we list the following desirable voting system properties as an example:

1. Fairness: no early results can be obtained which could influence the remaining voters.
2. Eligibility: only legitimate voters can vote, and only once.
3. Privacy: the fact that a particular voter voted in a particular way is not revealed to anyone.
4. Individual verifiability: a voter can verify that her vote was correctly included in the collection of votes.
5. Universal verifiability: anyone can verify that the published outcome correctly represents that same collection of votes.
6. Receipt-freeness: a voter cannot prove that she voted in a certain way (this is important to protect voters from coercion).

Of these, properties 4 and 5, taken together, impose an end-to-end constraint on the process of vote collection and tabulation, while property 2 protects the vote from dilution, and properties 1, 3 and 6 work to prevent coercion of voters. (We observe that these six properties do not allow anyone to verify that *all* votes were [properly] included in the tally.)

The end-to-end constraints can be satisfied trivially in small elections where preventing coercion is not an issue. Everyone involved in a show of hands can verify that their hand is up or down, and everyone involved can count all of the votes to verify the result. Similarly, if voters sign their paper ballots and post them on a bulletin board, each voter can verify that their ballot is on the board, which represents the collection of votes, and anyone can verify the result.

Paper ballots, whether hand counted or machine counted, come close enough to meeting these constraints that their failures are worth analyzing. Jones [2002] observed that, while voters can check that their ballots were marked as intended, they cannot check that hand-made markings on ballots meet the criteria that will be used in the count. Similarly, while voters can observe that their ballots

went into the ballot box, everyone cannot observe the chain of custody of the ballots between the time they go into the box and the time the box is opened for counting. Finally, the number of people who can directly observe the count is strictly limited, so we do not have universal verifiability.

Numerous voting systems have been proposed that attempt to meet these constraints. The following papers describe key technologies on which practical systems have been based. Note that these predate Kremer and Ryan's enumeration of voting system properties; that enumeration is, in large part, one attempt to formalize the problem that these and other new voting systems are attempting to solve:

- Chaum [1981] proposed the idea of a cryptographic *mix net* with applications to anonymous e-mail and electronic voting. In this approach, votes are encrypted and then shuffled by multiple *mixes*, each mix partially decrypts votes as well. In the final set of fully-decrypted and shuffled votes the link between a voter and her vote is completely destroyed. This set of votes may be counted. This approach established one of the two main paradigms for later proposals for end-to-end verifiable voting.
- Benaloh [1985] introduced *homomorphic encryption* as an alternative to mix nets, and this approach is the basis of the other main paradigm. In this approach, encrypted votes are counted without decryption, and only after counting are the totals decrypted.
- Chaum [2001] introduced *code voting*, embodied in the SureVote system. This approach forms the basis of many remote voting systems where the voting machine is not trusted. Voters obtain codesheets with a code corresponding to each candidate, and enter the code, instead of the candidate name, into the voting machine.

Building on these ideas, a number of practical voting systems have been developed. Each of the following systems has actually been used in a real election or in a pilot:

RIES Hubbers et al, 2004

RIES, the Rijnland Internet Election System, was developed to support elections to the Rijnland water management board, supplementing the system of postal voting used by the water board. It weakens the receipt-freeness requirement generally accepted for E2E voting systems while providing universal verifiability and a degree of individual verifiability. These compromises are based on the fact that remote voting generally, including postal voting, has weak coercion resistance, and adding universal verifiability and any degree of individual verifiability is a distinct improvement.

Before the election, credentials are mailed to every voter in the form of a very long number (encoded in Crockford's Base 32) (see Crockford 2002. The same

mailing also includes instructions. Voters log into an election web site that includes a client-side voting application written in Javascript. The algorithms used by this script are public, and each voter, havign access to all of the inputs and outputs, may (in principle) check the computations. This is weaker than the desired individual verifiability, but nonetheless, far stronger than conventional voting systems.

The client-side application encrypts the vote by passing the voter authorization code and the public ID of the candidate through a one-way function to create the encrypted vote. The encrypted vote is then placed on the public bulletin board that serves as a ballot box. At the close of the polls, the election authority releases the codebook containing the encryptions of all valid credentials with all candidate IDs. If a voter discloses her credential or her encrypted vote, the published codebook may be used to violate ballot secrecy. The developers of RIES judged this violation to be no more severe than the threats to ballot secrecy inherent in postal voting.

RIES was used for Rijnland Water Board elections starting in 2004, and in 2006, it was used to allow expatriate voters to participate in the Dutch parliamentary elections, see Gonggrijp et al, 2009 The OSCE sent an election assessment team to observe the use of the system in 2006. Their report contains observations of critical security features of the system that could not be observed; see OSCE 2007

One of the more important practical contributions of RIES involves the recognition that, when voter authorizations are distributed by post or more generally, distributed long in advance of the election, a mechanism must be provided allowing voters to obtain replacement credentials, for example, by telephone or e-mail, and a mechanism is needed to invalidate lost credentials. Adding these mechanisms adds significant complexity to the system and is a source of some of the problems reported in the OSCE report. A second feature of RIES is that it allows parallel testing during the election, where pre-invalidated test ballots are deliberately added to the bulletin board in order to test the network path from selected internet clients to the server. This offers the possibility of detecting a variety of spoofing and denial of service attacks.

Prêt à Voter, Chaum, Ryan and Steve Schneider [2005]

The Prêt à Voter system uses two-part paper ballots with the candidate names on one part and the voting targets plus a ballot ID number on the other. Typically, the two parts are printed as a single sheet, with a perforation along which the sheet can be divided after voting. The order of the candidate names appears to be random, from the voters' perspective. After voting, the half of the ballot containing candidate names is shredded. The voted half forms the cast ballot, and the voter may take a copy home.

A curious voter may inspect the public record of any cast ballot by searching for

it by the ballot ID number in the public database of cast ballots. That database shows the positions that were marked on that ballot, but crucially, it does not show the identifying letters or candidate names. The voter may therefore verify that the positions marked at the polling place were correctly recorded by the jurisdiction, but because the voter only has half of the ballot and there is no public display of the linking of candidate names to ballot positions, the voter cannot prove to anyone else how the ballot was voted.

The key to tabulating the votes is that there is a cryptographically secure mapping from the ballot serial numbers to the apparent random order of the candidate voting positions. The decoding of a cast ballot to a corresponding plain-text ballot, where the voted positions and candidate names are in a canonical order (alphabetical order, for example) is performed in multiple steps, by multiple custodians, for example, using a mixnet. As with the mixnet, no one, even if colluding with a subset of the custodians, can connect a decoded ballot to the corresponding cast ballot (and, hence, to the voter). Unvoted ballots may be audited before, during and after the election to ensure that the decoding of cast ballots is being correctly performed (audited ballots may not be used for voting purposes, however, because the connection between the voting positions and candidate names is publicly revealed during the audit). Randomly selected stages in the decoding can be challenged to prove the integrity of the count, and decoded ballots are easily counted. Anyone may therefore check the count.

A version of Prêt à Voter is planned for use in a governmental election by the state of Victoria in Australia in November 2014, see Burton et al, 2012 Previously, an attempt was made to use Prêt à Voter in a student election at the University of Surrey in February 2007, see Bismark et al, 2007 — this attempt illustrates many of the perils of working with an election authority.

Punchscan, Popoveniuc and Hosp [2006, 2010

The Punchscan system used a two-part paper ballot where the top part had candidate names and candidate numbers (or letters) and the bottom part had the numbered (or lettered) voting targets. The top part had holes punched in it to expose the voting targets below. The order of the voting targets for each race appears random to the voter. Both halves of the ballot bear an identical serial number. After marking the ballot with a bingo dauber, the two halves are separated. Prior to separation, the voter can easily see that the correct target has been marked. To vote, the voter separates the two sides. Either side can be scanned (since the bingo dauber marked both through the hole and around it) as the cast ballot. The other side is destroyed. A copy of the cast side may be retained by the voter.

A curious voter may inspect the public record of any cast ballot exactly as with Prêt à Voter. It does not matter which half of the ballot the voter retained, because there is no public display of the numbers that link candidate names to voting positions; only the position that was marked is displayed. Again,

individual ballots may be audited, and the key to tabulating the votes is that there is a cryptographically secure mapping from the ballot serial numbers to the apparent random order of the candidate voting positions.

Punchscan was used for the graduate student association elections of the University of Ottawa in 2007, see Essex, Clark, Carback and Popoveniuc, 2007 It is likely the first end-to-end voting system with ballot privacy used in a binding election.

Scantegrity II, Chaum et al 2008, Chaum et al 2009

Scantegrity II (Invisible Ink) allows end-to-end cryptographic verification of optical-scan paper ballots. Scantegrity ballots may be fully compatible with conventional optical scan vote tabulation equipment, but are voted using a marking pen filled with invisible ink. When the voter marks a target on the ballot, the ink in the pen reacts with invisible ink on the paper to disclose a three-letter alphanumeric code in the marked voting target. A voter who takes note of this code and the ballot ID number may check a public bulletin board to check that the ballot was indeed tabulated.

As with Punchscan and Prêt à Voter, there is a cryptographically secure mapping from the ballot ID number to the code disclosed when the voter marks a voting target. The code is displayed on the public bulletin board, and public verification of the decryption and tally of the contents of the bulletin board proceeds in a manner similar to that of Punchscan for the system fielded in Takoma Park, and in a somewhat different manner using similar principles for the system described in Chaum et al 2009.

Scantegrity II is the first end-to-end voting system with ballot privacy to see use in government elections. It was used in Takoma Park, Maryland in 2009 and 2011; see Carback et al 2010

Helios Adida 2008, Adida et al, 2009

Helios was developed for Web-based Internet voting. Helios provides a web-browser-based ballot preparation system that can be used to make choices and encrypt ballots. After preparing a ballot, the voter has the option of either casting that ballot or auditing it; this allows the voter to detect misbehavior on the part of the ballot preparation software. Voter authentication is not required until after the voter decides to cast the ballot, so anyone may prepare and audit ballots. Existing Helios implementations piggyback on existing university and commercial authentication mechanisms, and also support an internal login-password authentication mechanism for which credentials can be distributed by email.

As with the schemes discussed above, all cast ballots are posted in encrypted form on a public bulletin board so that voters may check that their ballots have

been correctly recorded. Similarly, after the polls close, the decryption and vote tally may be checked. Helios uses homomorphic vote tallying for simplicity, even though a single mixer was used in its first developments, and mixnets have been used in several of its forks and descendants; see Bulens et al or Tsoukalas et al for instance. A recent variant by Cortier et al proposes an enhanced ballot authentication mechanism. The STAR-Vote system, designed for public elections as a result of an invitation from the Travis County election administration, also borrows various techniques used in Helios and VoteBox.

Helios was used for the election of a Belgian university president in March 2009 and by numerous universities and associations since then, including the ACM and the IACR. The cryptographic protocols it implements have been subject to considerable analysis.

Norwegian System Gjøsteen 2012

Between 2011 and 2014, the Norwegian government ran an Internet remote voting trial. The system rests on a cryptographic protocol designed by Scytl, a commercial voting system vendor. Scytl and the Norwegian government assert that this is an E2E system. As such, this marks the entry of efforts (or claims) by commercial voting system vendors to enable E2E elections. System descriptions have been incomplete, hence it has not been possible to tell what properties the system possesses. However, the following are clear. The system's claims to protect voter privacy are weak: "If the voter's computer and the return code generator are both honest, the content of the voter's ballot remains private." In addition, the receipt delivered to the voter proved only that the encrypted ballot was received as cast, not that it was counted as cast or that the encrypted vote matched the voter's intent. Thus, by any of the definitions above, it is not an end-to-end system.

The Norwegian system used a three-channel model, where the voter receives authorization codes to cast a ballot via the postal system, then uses a computer to cast an encrypted ballot, and finally obtains a confirmation code offering a parital end-to-end proof via an SMS message to the voter's mobile phone. In addition, as with RIES, voters could cast multiple ballots; in Norway, only the last ballot cast was counted, and if a voter votes both on paper at a polling place and by Internet, the paper ballot overrides the Internet ballot.

The system evolved significantly between its first use in 2011 and 2013, with added complexity to attempt to assure voters that their ballots were stored as cast. In 2013, the Carter Center mounted a serious effort to observe the Norwegian system in action. Their report on the operation of the system, in practice, and the problems they had observing it offers useful insight into the administration of E2E systems in general as well as the particulars of the Norwegian system; see Carter Center, 2013

Remotegrity, Zagórski et al [2013]

Remotegrity is a remote coded voting system that additionally uses the notion of a lock-in to provide additional security properties. Remotegrity voters get a package in the postal mail. The package contains:

- a coded voting ballot (a ballot with a code printed against each choice). The code may or may not be covered by a scratch-off field (such as is used for lottery tickets)
- an authentication card that contains: (a) many authentication codes under scratch-off (b) a lock-in code under scratch-off and © an acknowledgement code.

Both cards have serial numbers.

To vote, the voter enters (a) both serial numbers, (b) the codes corresponding to her choices and © an authentication code obtained after scratching-off a surface chosen at random.

She returns to the election website a few hours later to check if her codes are correctly represented, and to see if the election authority has posted her acknowledgement code next to the codes. This indicates to her that the election officials received valid codes for her ballot.

She scratches-off the lock-in code and posts it on the website. This is her way of communicating to the election custodians, observers and other voters that her vote is correctly represented on the website. Among all of the E2E systems (and approximations to E2E systems) discussed here, this is the first one that asks the voter to take positive action to confirm that the vote was correctly posted.

As with RIES, if we assume that there is no communication between the computer used to print the credentials and the computer used to accumulate the votes, the latter computer does not know the mapping from codes to candidates, so the vote is not revealed to the computer. Further, because the computer does not know a valid code corresponding to another candidate on the ballot, it cannot change the vote. Finally, and uniquely, because the computer does not know the acknowledgement code, its presence on the election website assures the voter that the election officials received a valid code for her ballot.

The tally is computed from the codes in a verifiable manner that corresponds to the coded voting system used.

The voter can be sent two ballots. She can choose one to vote with and one to audit.

If a jurisdiction is nervous about using the Internet for remote voting, Remotegrity ballots can be mailed in, and voters can check for their codes on the election website to be assured that their vote correctly reached election officials.

Remotegrity was used for absentee voting—-and Scantegrity for in-person voting—-by the City of Takoma Park in its 2011 municipal election, see Zagórski et al [2013].

Wombat, Rosen et al [2011]

The Wombat voting system is an in-person voting system where the voter votes on a touch-screen and obtains a printout of his vote with an encryption of it. The voter can choose to cast or audit the encrypted vote. If she chooses to audit the vote, she may check if the vote was correctly encrypted. If she chooses to cast it, the ciphertext is posted online, and she casts the unencrypted vote in the ballot box (this may be manually counted) and takes the ciphertext home. The votes are tallied using a verifiable mixnet.

The system has been used for multiple pilot elections in Israel.

DEMOS, Kiayias et al [2014]

DEMOS is a coded vote system where the voter is given a two-part coded ballot; she audits one part and uses the other to vote. Associated with each choice on the ballot is (a) a vote code—-the encryption of the vote, which is entered in the voting machine by the voter and (b) a receipt code which the voter does not enter, but which is posted online next to the vote code. The voter can check this to ensure her vote reached the election authorities. The ballot also has a QR code containing all the information on the ballot. It can be scanned by the voter if she prefers not to manually enter the vote code. Once the ballot is entirely represented on the computer, the voter can then make her choices. Note that if the voter scans the QR code, the computer knows how she voted. The vote codes represent homomorphic encryptions of the votes and the verifiable tally is obtained in a standard manner.

A pilot study of DEMOS was carried out in 2014.

Typical Use

All of these E2E voting systems rest on a similar information flow, so we organize our discussion of the typical use in terms of this flow.

(Perhaps this section is best done after a similar, but more technical, effort by the cryptographic protocol team.)

Pre-election phase

Before the polls open—-and in cases where the postal system is used, long before—-there are a number of operations that must be carried out and data

provided to the public. All the data is generally posted on an election website. The custodians may use public (asymmetric) keys and/or symmetric keys to carry out the cryptographic operations performed before, during and after the election. Public keys are declared before the election. To construct secret keys, election custodians may construct a single master secret from individual passwords. The master secret is known as a shared secret, and is used to construct all other secret keys used in the election.

In addition to cryptographic keys, an important work product of the pre-election phase is a set of voting credentials that must be distributed to voters to allow them to vote. With paper-ballot elections (whether E2E verifiable or conventional), the blank paper ballots are the credentials; these are distributed either by post or by election clerks who directly hand them to voters. Where Internet voting is involved, the credentials may take the form of passwords, physical tokens such as smart-cards, or hybrids such as a smart-card plus a PIN.

The custodians of the election must, in come cases, make a public commitment of certain results of their pre-election computations. They may do this, for example, by publishing a secure hash of the results. In addition, the custodians may also post commitments to how they will process votes. Both sets of commitments will enable the post-election audits.

Voter privacy requires that the secrets used in setting up the election not be leaked to insiders. In all of the systems (whether voters use paper ballots or paper credentials, or directly enter their vote on a voting machine), information leakage—-from the system that prints the ballots or credentials or encrypts the vote—-to the system that accumulates the votes would permit privacy violation and in some cases, selective disenfranchisement. While procedures may be used to reduce the probability of such leakage, it is very difficult to guarantee that such leakage will not/did not occur.

Voting

In an E2E voting system, as in any voting system, a voter is presented candidate choices. Once the voter makes a choice, she receives a cryptographic receipt.

- For the (electronic or paper) ballot serial number used by her, if the voting system does not cheat, this receipt represents one candidate only, the one chosen by her.
- However, in a secret-ballot E2E system, the receipt cannot be used to determine the candidate. E2E systems that rely on the security of cryptographic primitives to enable privacy are said to provide *computational privacy*. Systems that do not rely on such an assumption are said to provide *unconditional privacy*.

It is important that the voter be able to check that the receipt correctly represents her vote; that is, that the voting system did not cheat. In the E2E voting

systems that have been used, this end is generally achieved using the cast-or-audit approach due to Benaloh [2006], as follows. The voter needs her credentials only to cast a receipt (and not to obtain it). She hence obtains receipts for many different votes. She casts only one of these, corresponding to her chosen candidate (the other uncast receipts may correspond to any candidates, including her own). She challenges all the uncast receipts. The voting system provides a public proof of the votes encrypted by each challenged receipt. Note that a voter cannot cast a challenged receipt, because the corresponding vote is public. However, because the voting system does not know which receipts will be challenged, it is caught, with high probability (assuming challenges are suitably random), if it tries to encrypt incorrect votes. Note also that all voters need not challenge receipts. Also note that observers and auditors who are not voters may also obtain receipts and challenge them. As long as the machine cannot tell which receipt will be challenged, it will be caught with high probability (again, if challenges are suitably random) if it cheats.

The voter may take the voted receipt and challenged receipts and proofs home with her if she votes from a polling booth. Alternately, she is presented the receipts and proofs in electronic form if she votes remotely using an electronic channel. In addition to the proofs for challenged receipts, the custodians post all voted receipts on the election website and compute the tally from the voted receipts. The custodians may use secret keys to compute the tally, however the computation must be performed in an auditable fashion. Additionally, for ballot privacy, the audits must not reveal information on the secrets or the individual votes beyond that already contained in the correct tally.

Post-election phase

Voters who wish to can check that their receipts are represented correctly on the election website. If a large enough number of voters independently perform this check, and if the voting system cannot predict who will check, a voting system attempting to use different receipts for the tally will be caught with high probability (if voters who check amount to a random sample of all voters).

A post-election audit that checks that the tally is correctly computed from the posted receipts is performed. This audit uses data that can be made public without revealing information on secrets or any individual votes (beyond the information contained in the correct tally). The audit will typically require reliable confirmation that a certain minimum number of independent voters had correctly checked their receipts, and that no one had found an error.

Additionally, the post-election audit also checks the correctness of the proofs for challenged receipts.

What is not covered

E2E techniques make it possible to detect manipulations of an election that can lead to incorrect results. They do not, however, prevent such manipulation. Additionally, by themselves, they do not guarantee anything regarding the privacy of the votes (voting by a public show of hands is E2E verifiable) or the availability of the election results (E2E does not prevent destroying all paper and/or electronic ballots before tally, though such destruction would be detected).

Of course, E2E verifiability does not contradict these properties: all the E2E systems described above provide these properties at various levels, and often in ways that are more robust than in the currently deployed non-E2E systems. Depending on the intended uses of the systems, various solutions are proposed, balancing security, usability and versatility.

In practice, the privacy of the votes will typically rely on a set of trustees to not collude and, in some cases, on the device used to prepare the ballot to not reveal the ballot content (when there is such a device).

Getting election results will depend on the availability of paper ballots and/or voting devices, or of websites serving the ballot preparation system. It will also depend on the protection of the urns (paper, or electronic): ballot stuffing or hacking of an election server will be detected thanks to E2E verifiability, but is still likely to make it impossible to obtain election results.

E2E techniques also often rely, at various levels, on the availability of some specific components. It is typically assumed, for instance, that auditors and voters can run their verification procedure on at least one device that is not corrupted. It is also often assumed that a broadcast channel, providing consistent data to everyone, is available for getting information such as election description, audit data, or public randomness. Such channels are sometimes simply assumed to be provided by a public website, but various other techniques have also been implemented, including the use of newspapers, digital signatures, or more sophisticated distributed cryptographic protocols.

Some (though not all) E2E designs are not able to resolve disputes between the voting system (which may, for example, insist that it has honestly represented voter receipts) and a group of voters (who might be insisting otherwise). With these voting systems, hence, the public would not be able to determine if the complaints of a group of voters were genuine.

Strengths

Paper ballot systems using statistical audits require a secure chain of custody. This is not always straightforward to ensure. It is also not easy to detect any violations of this assumption. Finally, only a certain elite few are in a position to detect these violations.

E2E systems, on the other hand, democratize access to the data that reveals attempts at election fraud. They also do not require the voter or observer desirous of checking the correctness of election outcome to trust a certain set of officials, software or hardware. To the extent that software or hardware needs to be used to check the election outcome, an E2E system allows the individual to choose which software or hardware she would use. To the extent that it requires trust in election procedures, these are performed during the election in a publicly-viewable manner when they deal with the collection of votes or ballots in general. When they deal with an individual vote, they are viewable during the voting process by the corresponding individual voter.

Weaknesses

E2E systems require the participation of voters and observers to detect problems.

E2E systems providing computational privacy open up the possibility of vote exposure if the cryptography used is insecure, or if a set of custodians collude. Note that this vote exposure relates the receipt to the vote. If the voter reveals her receipt, it then relates the voter to the vote through the receipt.

E2E systems increase the number of steps in the voting process and hence decrease its usability. The most secure E2E voting systems are paper-ballot-based, and hence very inaccessible.

Potential points of failure

The most vulnerable point in an E2E voting system is the election website. Additionally, an E2E system can fail if the voting process is not adequately communicated to voters, or if a sufficient number of voters does not check encryption-correctness or the presence of receipts on the website. Both aspects of an E2E election need special attention.