

A Case Study in System-Based Analysis: The ThreeBallot Voting System and Prêt à Voter

Thomas Tjøstheim¹, Thea Peacock² and Peter Y. A. Ryan²

¹ Department of Informatics, University of Bergen, Norway

² School of Computing Science, University of Newcastle, United Kingdom
thomast@ii.uib.no, t.peacock@ncl.ac.uk, and peter.ryan@ncl.ac.uk

Abstract. Threat analysis of voting systems is a field of increasing interest. While it is important to verify the system itself, it has been found that certain vulnerabilities only become apparent when taking a “system-based” view, i.e. considering interactions between the various components of a scheme. In this paper we apply a model for system-based analysis to carry out a systematic threat analysis of the ThreeBallot voting system and Prêt à Voter.

1 Introduction

There has been much progress in developing high assurance, verifiable voting systems. Ideally, there should be minimal reliance on the players, i.e. voters, election officials, etc., and technical components, such as the hardware and software behaving as intended. Notable examples are Prêt à Voter [8], Punchscan [3], and VoteHere [4], all of which aim to provide a high degree of transparency in the system. While cryptography is often used to enable verifiability without compromising voter privacy, Rivest has shown with the ThreeBallot voting system that this is not, in fact, an absolute necessity [18].

Recently, interest has grown in analysis techniques to ensure that voting systems meet election requirements, such as eligibility, coercion-resistance and accuracy. In [13] Karlof et al. carried out a system-based analysis of Chaum’s visual crypto scheme [7] and Neff’s scheme [16, 15], identifying potential threats such as subliminal channels and “social-engineering”-style attacks. In a similar analysis, Ryan et al. [20] showed that Prêt à Voter [8] is robust against many of the threats mentioned in [13], but identified further possible vulnerabilities such as chain-voting and authority knowledge.

By considering the interactions between the various components in the above analysis new threats were identified. Although highly useful, this type of analysis did not consider the interactions systematically and hence, may not have uncovered all the possible threats. A more formal analysis of a voting protocol [17, 14] on the other hand may be more systematic, but is limited to the technical core of the protocol. Another approach is to develop a “catalogue of threats” [1], aside from [2] there is little work to date in this direction.

In [22], we proposed a model for analysis of threats in voting systems that is essentially “system-based”, but considerably more systematic than previous

similar work [13, 20]. While [2] has a largely technical focus and concentrates on DRE systems, our model operates at a higher level of abstraction and is not scheme-specific. In Appendix A we give a brief introduction to the model.

The structure of the paper is as follows. In Section 2 and 3 we apply the model to carry out a system-based analysis of the ThreeBallot voting scheme and Prêt à Voter, respectively. Finally, in Section 4 we discuss the main results of the analysis.

2 Threat Analysis Case Study: The ThreeBallot Voting System

2.1 Outline of the ThreeBallot Voting System

We now present an outline of the ThreeBallot voting system, for details see [18]. ThreeBallots is a verifiable paper-based voting system. Contrary to other verifiable voting systems, such as Prêt à Voter and PunchScan no encryption is used to provide voter verifiability, allowing a new level of transparency. In the following we describe the setup of the election. A voter votes using a multi-ballot, which consists of three individual ballots separated by perforated lines. Note that three single ballots together also could act as a multi-ballot. Each ballot lists the candidate choices together with a “bubble” for each candidate, and a ballot ID printed at the bottom of the ballot, see Figure 1.

Ballot		Ballot		Ballot	
Odin	<input type="radio"/>	Odin	<input type="radio"/>	Odin	<input type="radio"/>
Thor	<input type="radio"/>	Thor	<input type="radio"/>	Thor	<input type="radio"/>
Trym	<input type="radio"/>	Trym	<input type="radio"/>	Trym	<input type="radio"/>
Loki	<input type="radio"/>	Loki	<input type="radio"/>	Loki	<input type="radio"/>
30111979		15031983		45140852	

Fig. 1. An example of a multi-ballot

A multi-ballot must be marked according to specific rules defined by the ThreeBallot voting system. In the privacy of the voting booth the voter fills in exactly one bubble for each candidate, and fills in an extra bubble for the candidate(s) she prefers. After the voter has marked her multi-ballot, it is passed through a *checker machine* that optically scans the multi-ballot and verifies that the three ballots together have been correctly filled out according to the ThreeBallot voting system’s rules. There should be exactly one mark for each of the non-chosen candidates, while there should be exactly two marks for the

chosen candidate(s), if properly formed, the checker device prints a “red stripe” at the bottom of each ballot. The checker machine also lets the voter randomly choose a copy of one of the three ballots, to retain as a receipt. The receipt can later be used to verify that this part of the voter’s multi-ballot has been correctly registered at the public Web bulletin board (WBB). While the three ballots together prove the voter’s choice, the receipt does not give away any information about how the voter voted. After the checker machine has verified the multi-ballot, the voter casts her three ballots to the ballot box in the presence of a voter official. Each ballot is then scanned and posted to the WBB. Tallying is straightforward, the number of marks for each candidate are added up, and the result for each candidate is obtained by subtracting the number of voters from each candidate total (since each voter adds a mark for the candidates she votes against). Anyone can verify the final tally from the WBB.

2.2 Threat Analysis of ThreeBallots

Ballot form

- Identifiable information added by voter/official: A voter could mark three ballots in an identifying way, to later prove the triplet of ballots used to cast the vote. In [18] no method is described with respect to how the ballots are scanned to the WBB. If, for example only the ballot ID and the index values of the corresponding candidate marks are registered at the WBB, it would be very difficult for a voter to prove her vote by adding identifying marks to the ballots. On the other hand, only registering a representation of the original ballots may open up for more errors, although such errors should be detected e.g. by a helper organisation or the voters themselves with high probability.
- Voter identifiable from ballot form: The ThreeBallot voting system is vulnerable to the “Italian attack”, i.e. an attack where the coercer makes the voter vote for a selection of candidates that most likely will be unique. The three ballots can then be identified later at the WBB. A prerequisite for the attack is that there is a sufficient number of candidates to choose from, such that unique candidate selections can be made.

In [18] no method for separating the ballots is described. The patterns of tearing may reveal the three ballots forming a multi-ballot. However, it will be very time consuming for an election official to physically go through all the ballots to identify corresponding ballots.

A voter could prove her vote by writing down, memorizing or taking photos of the ballot IDs of the ballots forming the valid triple. If in addition the voter makes the ballot IDs known to the coercer before the WBB phase, the coercer will be quite certain of how the voter voted. A countermeasure mentioned in [18] is to use ballot IDs that are hard to memorize. Another possible mitigation is the “Shamos checker”, which prevents the voter from learning the ballot IDs of the two ballots not chosen as receipts. A brief description of the Shamos checker is as follows:

- If the multi-ballot is valid, 3 random ballot IDs are generated, without being shown to the voter. The voter selects one of the ballots to retain as a receipt.
- The ballots that are not selected are put into a holding bin, while the machine produces the ballot selected as the receipt.
- A voter can then verify that her receipt is identical to the selected ballot. If this is not the case, the “I got a bad receipt” button could be pressed, and the ballots in the holding bin will be put into a spoiled ballot bucket. Otherwise, the ballots in the holding bin will be put into the ballot box.

Although the voter is not able to remember the ballot IDs of the two other ballots, new threats may be introduced. The machine could learn the correspondences between ballot IDs and ballots or could choose ballot IDs such that they can be easily correlated later. Another possible weakness is protection of the spoiled ballot box. Spoiled ballots could for instance be substituted with the ballot the checker produces to the voter. The voter could also refuse to put the last ballot into the ballot box, in order to steal votes from the other candidates.

- Authority knowledge: Election officials may learn the correspondences between ballot IDs of the three ballots forming the multi-ballot during ballot construction. Printing of separate ballot forms is one possible countermeasure. Another possibility mentioned in [18] is to let the voters pick ballot IDs from a bucket of stickers. A dishonest election official could learn the correspondence of ballot IDs when ensuring that the voter casts exactly three ballots. This could be envisaged as a “social engineering” type attack, in which a voter who has not understood the importance of not revealing the ballot IDs, could be tricked into revealing them. Voter education is important to ensure the voters’ understanding of the system.
- Voter’s choice incorrectly represented: A dishonest voter or election official could try to add or remove marks, after the ballots have been verified by the checker machine. This may or may not be detected during verification of the final tally, depending on the extent of the manipulation. It would, for instance, be quite obvious if the total number of votes for a particular candidate was more than the number of voters, or if a candidate has a negative number of votes. However, the attack cannot be traced without violating voter privacy. Checksums calculated over the original marked ballot is mentioned as a possible countermeasure. On the other hand, this would require a more complex checker device, which may be more vulnerable to tampering.
- Ballot form spoiled: An election official could mark a ballot in order to invalidate it (if a checksum scheme is used), or physically destroy a ballot form. If one of the three ballots belonging to a voter is modified, the voter will detect it with a probability of $1/3$, provided she checks her receipt against the WBB. Thus, it would be difficult for a dishonest voting official to carry out any substantial ballot spoiling attacks without detection, given that voters are diligent in checking their receipts against the WBB. However, tracing

the attacks to the dishonest election official may be difficult, so this could be a way of launching a DoS attack.

- Ballot form faked: An interesting point is the level of authentication provided by the “red stripe” printed by the checker machine to prove that the ballots have been correctly formed. Given a set of valid ballots, a dishonest voter/election official may be able to fake the stripe and cast illegally formed multi-ballots. In the ThreeBallots scheme the names of all the voters and all the ballots ($3n$ if there are n voters) are posted to the WBB. Therefore, ballot faking attacks may be detected, but not necessarily traced. Although this approach allows public scrutiny of who voted, it may make forced abstention attacks easier to carry out.

Voting booth

- Voter’s activity monitored: As for all schemes that require a polling station, a camera in the booth is a threat. Shoulder surfing may be more difficult than for other schemes, since the representation of the voter’s choices is more complex. The candidates the voter votes *against* are marked once, while the voter’s candidate choice has exactly two marks, a quick glance may therefore not immediately reveal how the voter voted.
- Voter records own choice: The voter could use a camera phone to record her vote, for example.
- Voter’s choice influenced: There could be a subliminal message in the booth to persuade the voter to vote for one of the candidates, for instance. Note that the above threats to the voting booth component would be present in almost any scheme, but should nevertheless be evaluated in an analysis.
- Voter smuggles out unmarked ballot form: A chain voting attack could be initiated if a voter smuggles out an unmarked ballot form. The coercer can then confirm how the voter votes by checking the ballot IDs at the WBB. Note that if the ballot ID stickers approach is used, the stickers could also be smuggled out, so the coercer could control the voters’ behaviour.

Ballot storage

- Ballot stuffing: It is assumed that a voter casts exactly three properly formed ballots. A voter violating this rule could, for instance, cast only two ballots where only the marks for the desired candidate are included and discard the last ballot which contains the mandatory marks for the other candidates. The scheme does not specify a method to ensure that the voter casts exactly three ballots. An enforcement of this rule should preserve voter privacy as well.

A voter or dishonest election official may be able to add extra votes to a multi-ballot if the checker malfunctions. Another threat is a voter who verifies two multi-ballots through the checker, and combines these to one “badly” formed ballot. It should be very difficult to pass more than one ballot through the checker, e.g. with election officials closely observing the process,

or authentication mechanisms implemented in the device. Two voters could, however, bypass this by colluding to cast one illegal vote. This could proceed in the following way: each voter gets a properly formed multi-ballot verified by the checker, the first voter smuggles out the multi-ballot, while the second voter combines the two legal ballots to form an illegal ballot. This will be more effective than casting two legitimate votes, as they can construct a vote that, for example, gives three marks for their candidate and none for the others. The net effect is therefore that they lose one mark for their desired candidate, but steal two marks from the other candidates. Figure 2 illustrates how two legally formed ballots could be combined to form an illegal multi-ballot; for purpose of illustration we have chosen some easily recognisable ballot IDs.

Ballot	Ballot	Ballot
Odin ○	Odin ●	Odin ●
Thor ●	Thor ○	Thor ○
Trym ●	Trym ○	Trym ○
Loki ●	Loki ○	Loki ○
11111	22222	33333

Ballot	Ballot	Ballot
Odin ●	Odin ●	Odin ○
Thor ○	Thor ○	Thor ●
Trym ○	Trym ○	Trym ●
Loki ●	Loki ○	Loki ○
44444	55555	66666

Ballot
Odin ●
Thor ○
Trym ○
Loki ○
22222

Ballot
Odin ●
Thor ○
Trym ○
Loki ○
33333

Ballot
Odin ●
Thor ○
Trym ○
Loki ○
55555

Fig. 2. Creating an illegal multi-ballot

A possible countermeasure, could involve the checker machine printing three equal images or identifiers on the back of the ballots (chosen randomly from a large set) and adding a perforated line above the images. The voter then proceeds to an election authority who verifies that the images are equal to each other and tears them off. The voter can then cast her ballots to the ballot box. This procedure would also ensure that the voter casts exactly

three ballots. However, it may introduce more threats by adding an extra layer of complexity to the scheme.

- Ballot spoiling: The ballot box could e.g. be destroyed or replaced with one containing fake ballots, but voters will detect such attacks if they verify their receipts against the WBB. In [18] the use of “helper organisations” (e.g. the League of Women Voters) is envisaged as an additional help to verify WBB integrity.

Election results

- Early publishing: A threat to early publishing in ThreeBallots is the fact that by simply tallying the marks for each candidate on the voters’ receipts one can get an indication of who is winning the election at that particular polling place. This threat was pointed out to us through personal correspondence with Roberto Samarone Araújo, Ricardo Felipe Custódio and Jeroen van de Graaf. A prerequisite for the attack is that voters are willing to show their receipts to some organisation that is awaiting people at the polling station. Given that voters mark their candidate choice in a random pattern and randomly choose one of the ballots to copy, to retain as a receipt, there is a risk that a statistical analysis will reveal which candidate is winning the race. This has also been confirmed through simulations carried out by Araújo et al.
- Absence of verifiability: Vulnerabilities covered under WBB and encrypted receipts.
- False/erroneous counts: This is not a significant threat due to the transparency of the scheme. All ballots are posted to the WBB, so that anyone can calculate and verify the final tally. This is a huge advantage compared to traditional paper based schemes and the verification is easier to understand for the average voter, than the mix net or homomorphic approach used in encrypted receipt schemes.

Voting device

- Identifiable information added: A checker could encode information about which ballots belong with each other, e.g. in the way the red lines are printed onto the ballots. Testing that the checker device is properly calibrated both before and during the election is a possible countermeasure.
- Voter’s activity monitored: A malicious program inserted into the device could register information about the ballots and record sequence numbers. Another threat is a wireless component in the device which communicates to the outside, so the voter using the device could be linked with the information passed through the checker, i.e. a voter’s choices.
- Faulty authorisation: Not a threat, as the voter’s credentials are not authorised by the checker machine. However, ensuring that the voter only gets to check one multiple-ballot through the checker is a possible mitigation against ballot stuffing attacks.

- Voter’s choice incorrectly/not recorded: A malicious checker could, for example add extra marks to the ballots on the chance that the voter does not notice. A malicious checker could also allow improperly formed ballot forms. However, this would require prior knowledge about the checker or the possibility that a voter could modify the checker machine.
- Denial of Service: A machine breakdown is not a threat to this scheme, as it is to DREs, where the votes are stored in the machines memory. There is less risk of losing votes, but voters should be prevented from voting until a new electronic device is found, to ensure that only properly formed ballot forms are cast.

Verifiable receipt

- Voter identifiable from receipt: Randomisation attacks are not a particular threat, since the voter can fill out the receipt according to the coercer’s wishes, but still trick the coercer in the way the two corresponding ballots are filled out.
- Authority knowledge: Kleptographic attacks are not a particular threat, as the scheme does not make any use of cryptography. One possibility could be to encode information into the ballot ID number, e.g. a specific hash function computed over a ballot ID might reveal which ballots correspond to each other. Printing of single ballots, where the voter picks three single ballots randomly would counter this threat. An interesting vote buying attack is described in [6], in which the voter can effectively sell parts of her ability to verify her vote. The coercer pays or intimidates voters to construct receipts that only contain a mark for one candidate, e.g. Thor. If a sufficient number of people are coerced, the coercer will then have a higher probability of getting away with changing votes from say Odin to Thor, as these ballots do not constitute receipts.
- Receipt discarded/surrendered: May indicate receipts that will not be checked.
- Invalid signature: A possible threat if the checker does not sign the receipt properly, or the signature verifier does not work properly. The voter would then be unable to convince the system that she has been disenfranchised.
- Faked receipt: The voter could claim to be disenfranchised by falsely claiming a faked receipt.

WBB

- Monitoring access to the WBB: A coercer with access to the the web server log at the public bulletin board web site could register which ballot receipt IDs are checked, and use this information to detect whether voters are trying to cheat by presenting fake triplets. The web interface used to access the WBB should therefore be implemented carefully to not reveal ballot IDs of receipts. Similarly, a dishonest employee of a helper organisation could sell receipts IDs. Information about which ballot IDs constitute receipts, together with the scheme’s logics for filling out multi-ballots, could also be used to

match ballots from the WBB in order to re-construct valid multi-ballots. In [23] several such successful attacks are described with various simulated election races.

- Voter presented with fake WBB: A false WBB could be shown to mislead a voter into believing that her receipt has been correctly recorded when in fact it has not.
- WBB modified: The WBB should be a write-only medium, but this seems hard to enforce in practice. There is a risk that the WBB could be modified after the voter has checked her receipt at the WBB.
- Denial of Service: DoS attacks are a possible threat.

3 Threat Analysis Case Study: Prêt à Voter'05

3.1 Outline of the Prêt à Voter Scheme'05

We now present a brief outline of the Prêt à Voter'05 scheme, for full details see [8]. Once registered in the polling station, voters select a ballot form, sealed in an envelope, at random. A typical example is shown below.

Odin	
Thor	
Trym	
Loki	
	<i>7rJ94K</i>

In the isolation of the booth, the voter makes her selection by placing a cross in the right-hand (RH) column against the candidate of choice. The left-hand (LH) column that carries the candidate list is discarded, leaving the ballot receipt. In this case, voting for Thor, the receipt would appear as follows:

X
<i>7rJ94K</i>

The voter then leaves the booth and casts her vote in the presence of an official: the receipt is placed under an optical reader, or similar device, to record the cryptographic value at the bottom of the strip, and the numerical representation of the cell into which the cross has been entered. The voter retains a digitally signed, hard copy of the right hand strip (RHS) as her receipt.

The candidate list on the ballot forms is randomised. Thus, with the left hand strip (LHS) removed and without knowledge of the appropriate cryptographic

keys, the RHS does not indicate which way the vote was cast. A nice side-effect of using a randomised candidate list is that a random order does not favour any of the candidates, whereas a fixed candidate list may influence voters to vote for candidates that are listed early.

The cryptographic value printed on the bottom of the receipt, the “onion”, is the key to extraction of the vote. Buried cryptographically in this value, is the seed information needed to reconstruct the candidate list. Thus, only a threshold subset of tellers holding the appropriate keys are able to reconstruct the candidate order and so interpret the vote value encoded on the receipt.

Once the election has closed, the receipts are transmitted to a central tabulation server which posts them to a secure Web bulletin board (WBB). This is an append-only, publicly visible facility. Only the tabulation server can write to this and, once written, anything posted to it will (in theory) remain unchanged. A voter can visit the WBB and confirm that her receipt appears correctly.

After a suitable period to allow voters to check their receipts, the tellers perform a robust, anonymising, decryption mix on the batch of posted receipts. The paper receipt allows voters to prove the absence or corruption of their receipt in the event that it fails to appear correctly on the WBB.

Various mechanisms are deployed to detect and deter any corruption in the construction of the ballot forms. The approach suggested in [8] is to perform a random pre-audit of the ballot forms.

3.2 Threat Analysis of Prêt à Voter’05

Ballot form

- Identifiable information added by voter/official: Not a threat as the LHS is detached and the RHS only states the onion and the voter’s mark.
- Voter identifiable from ballot form: Only the numerical value of the voter’s mark and the corresponding onion are recorded to the WBB, so unless the correspondence between onion and candidate order is leaked, the voter will not be able to prove how she voted. A voter could e.g. prove her vote by retaining the LHS. Possible mitigations are:
 - Enforcing destruction of the LHS in front of an official. However, the official may learn the correspondence between onion and candidate order and give away the information.
 - Mechanical destruction of the LHS, though this could be difficult to carry out in practice.
 - Having decoy LHS freely available in the booth. However, an adversary could mark decoy strips in a subtle way. A coercer may also be able to arrange that only “dummy” strips with a particular candidate ordering are available.
- Authority knowledge: Information could be leaked during storage and distribution, or later, once receipts have been posted to the WBB. Distributed generation of ballot forms has been proposed as a countermeasure in [21]. However, onion and candidate list correspondences could still be revealed by tellers acting in collusion.

- Voter’s choice incorrectly represented: A possible threat if the onion is not a true encryption of the candidate order. Suggested mitigations are as follows:
 - Voter casts dummy votes. Given the RHS and associated onion, return the candidate she selected.
 - Return the seed and run a checking algorithm for well-formedness.

The problem with the first method is that it is only a partial check of the ballot form construction. In addition tellers working in collusion, could return a fake candidate ordering. The second method is more thorough, but is only available to auditors. It is important to ensure that ballot forms are not re-used once they have been used for checking. Another possibility is to use an offline auditing mechanism, where audit information is posted on the ballot forms but concealed with a scratch strip [5].

A better solution might be for the voter to verify the construction of the ballot she actually uses to vote. One possibility is the use of two-sided ballot forms, where the voter can verify correct construction of one of the sides, while using the other ballot side to vote. This adds a “cut-and-choose step”: since the voter can check an arbitrary side, there is greater assurance that the other side is also correct [21].

Another possible threat is invalid decryption of receipts. However, this will, with a high probability, be caught during randomised partial checking (RPC) [12].

- Ballot form spoiled: A ballot form could be spoiled by a dishonest election official, e.g. by adding additional marks to a ballot form or physically destroying a ballot form. Another threat is that the onion could be modified during scanning to the WBB. All of these attacks would be detected if voters check their receipts, but a dishonest election official could initiate a DoS attack in this way.
- Ballot form faked: Fake ballot forms can be constructed with knowledge of the tellers’ public keys. A badly constructed fake ballot could also be used to initiate a DoS attack, as this would be caught during RPC of the mixing/decryption phase with high probability. However, ballot stuffing attacks should be difficult to carry out as the casting of ballots is supervised, so in principle, a voter is only able to cast one vote.

Voting booth

- Voter’s activity monitored: The voter could be monitored by a hidden camera in the booth.
- Voter records own choice: A camera phone could be used to record the voter’s choice.
- Voter choice influenced: There could be a subliminal message in the booth.
- Voter smuggles out unmarked ballot: Chain voting attacks are a threat as the coercer can control how the voter votes by checking the WBB for the corresponding onion. A countermeasure proposed in [20] is to cover the onion with a scratch strip and not reveal the onion before the tallying phase.

Ballot storage This has not been specified in Prêt à Voter, but the device would presumably record receipts, e.g. by writing to a disk and transmitting immediately to the WBB.

- Ballot stuffing: Extra votes could be recorded by a faulty/malicious device.
- Ballot spoiling: Recorded data could be lost or corrupted. In addition, the disks could be substituted by a malicious party. However, if voters verify their receipts at the WBB, in combination with a VEPAT mechanism such errors will be detected.

Election results

- Early publishing: Tallying and publishing of final results to the WBB should be synchronised.
- Absence of verifiability: Not a threat, unless there is a DoS from the WBB.
- False/erroneous count: The risk of an erroneous count should be minimal as there are various mechanisms to verify decryption and tallying of votes.

Voting device

- Identifiable information added by device: Not a threat, as the device only scans the receipt.
- Voter's activity monitored: A possible threat (e.g. via wireless connection), but as long as the crypto primitives used in the ballot form construction remain secret, the voter's choice cannot be learned from the RHS scanned by the device.
- Faulty authorisation: Not a threat, as the device does not authorise the voter.
- Voter choice incorrectly/not recorded: A possible threat, but would be discovered if voters are diligent in checking their receipts on the WBB. The use of a VEPAT mechanism or helper organisations are countermeasures as well.
- Denial of Service: Device failure is a threat, but the voter does not face the possibility of losing her vote if unable to scan her receipt, as may be the case with some electronic schemes.

Verifiable receipt

- Voter identifiable from receipt: Randomisation attacks are a threat. An attacker could, for example, require that the first candidate is marked, regardless of which candidate ordering is used. The level of threat is determined by the extent a voter can pick a ballot of her own choosing and the number of candidates in an election. In the case of few candidates, it might be easy for the voter to pick a ballot where she can vote as she wishes while satisfying the coercer. A randomisation attack may benefit the low key candidates as the votes will be spread evenly across the candidates.

- Authority knowledge: Kleptographic channels [11] are a threat, i.e. crypto variables chosen in such a way as to leak information to a colluding party. In Prêt à Voter’05, this is possible by choosing a seed value such that a keyed hash of the onion value reveals the candidate order. However, this would require a great deal of searching. In Prêt à Voter’06 [21] distributed creation of ballot forms is suggested as a possible countermeasure against kleptographic attacks.
An important advantage with the Prêt à Voter scheme is that the voter does not need to communicate their choice to any device, and as such subliminal or semantic channels are not threats.
- Discarded receipts/surrendered receipts: May indicate receipts that will not be checked and hence could be altered without detection. A possible countermeasure is a verifiable encrypted paper audit trail (VEPAT) mechanism [19].
- Invalid signature: A possible threat if the mechanism for digitally signing receipts is malicious/fails; this also applies to the mechanism for checking the signature on the receipt. The voter is then unable to prove an incorrectly recorded receipt.
- Faked receipt: A voter could falsely claim to be disenfranchised with a fake receipt. This could be mitigated by using signatures as proof of authenticity of the receipts.

WBB

- Monitoring access to the WBB: There is a risk that the WBB could be modified after the voter has checked her receipt and prior to the randomising mix phase. Although specified as a write-only medium, this is difficult to enforce in practice. Fraud will be detected if voters verify their receipts more than once, but voters may be reluctant to do so. As mentioned before a helper organisation in combination with a VEPAT may help ensure the integrity of the WBB.
- Voter presented with fake WBB: The voter could be presented with a fake WBB, e.g. in a spoofing attack, and be misled into believing her vote has been recorded correctly when in reality it has been changed.
- WBB modified: The integrity of the scheme is dependent on a certain percentage of voters verifying their receipts at the WBB. According to Carl Ellison [9]: “if there is a human step that is optional, then one can assume the human will not perform it. Some will and some won’t, but for the purpose of security analysis, one must assume the worst case.” A suggested mitigation is to have a VEPAT mechanism [19] in place, and for independent authorities to check the correspondence between the receipts and the contents of the WBB.
- Denial of service: A possible threat, e.g. due to network overload or power failure. DoS may also be an issue if a decryption mix net is used, e.g. if the tellers keys are corrupted/deleted. As discussed in [21], the advantage of a re-encryption mix is that faulty tellers can be removed if necessary.

4 Final Remarks

In the following we discuss the main results of applying the model defined in [22] to analyse the ThreeBallot voting system and Prêt à Voter.

4.1 The ThreeBallot Voting System

To ensure confidentiality in the ThreeBallots scheme it is important that once the multi-ballot has been split into three ballots in the voting booth, it should not be possible to link them at the WBB later. This is to avoid vote buying or coercion. However, as the analysis shows, it may be possible to link corresponding ballots in several ways, e.g. by remembering the ballot IDs, marking the candidate choices in a special way (the “Italian attack”), adding identifiable marks to the ballots, or by trying to reconstruct multi-ballots by matching ballots posted to the WBB. Another threat to confidentiality is registering which ballot IDs constitute receipts. This information can be used to verify voter behaviour, e.g. making it harder for a voter to present a “fake” triplet of ballots to the coercer.

Threats to the integrity of the scheme include all threats that may violate the principle that the final count should accurately reflect the true intention of the voters. Examples include ballot modifications, ballot faking, or tampering with the election results. New marks could be added or deleted from a multi-ballot after it has been verified, using checksums is a possible countermeasure. Another threat is combining two verified multi-ballots to one “badly” formed multi-ballot. An important requirement to ensure integrity is that the voter should cast exactly three ballots.

Threats to availability include DoS attacks against the checker device, ballot spoiling attacks and denying access to the WBB. While the scheme advocates transparency, e.g. by posting the names of all who voted and all the ballots to the WBB, it may make it easier to launch DoS attacks. A dishonest election official could add ballots such that the number of ballots does not correspond with the number of voters. Another possible approach is to deliberately spoil receipts to get a number of voters to complain about incorrectly recorded receipts.

We identify some interesting trade-offs in the scheme. A countermeasure against the remembering of ballot IDs is to make them harder to remember, e.g. by using a bar code or mixing fixed noise with the ballot ID. Although, these approaches make it more difficult for the voter to remember the IDs of the ballots, it also makes the verification against the WBB harder. Another interesting trade-off relates to the simplicity of the checker machine. Ideally, the checker machine should be a stateless memoryless machine that only checks if a multi-ballot has been correctly formed or not. A more complex machine may thwart some of the attacks above, e.g. by authorising the voters or adding checksums to the ballots. On the other hand, a more complex checker is more vulnerable to tampering, since malicious software could be used to learn and communicate the ballot IDs of three ballots.

ThreeBallots achieves voter verifiability and unconditional privacy, i.e. privacy that relies neither on trusted third parties, nor on computational intractabil-

ity assumptions (e.g. hardness of factoring). The scheme may not be practical as it stands, but is of immense theoretical importance as it demonstrates that it is possible to design a verifiable scheme with unconditional privacy without use of cryptography.

4.2 Prêt à Voter

In Prêt à Voter the confidentiality of the scheme is dependent on keeping the candidate order and onion correspondence secret. Threats to confidentiality include authority knowledge, chain voting, the voter retaining the LHS and kleptographic attacks. However, several mitigations have been suggested against most of these threats: distributed generation of ballot forms to counter authority knowledge and kleptographic attacks; scratch strips to mitigate chain voting attacks; and “dummy” LHS strips available in the booth to make it more difficult for the voter to prove how she voted. Prêt à Voter is vulnerable to randomisation attacks, but the impact on the election results may be limited, since the coercer cannot directly choose which candidate the voter votes for. However, low key candidates may benefit from the attack, if the votes are spread more evenly across the candidates.

Threats to the integrity of the scheme are largely related to the onion not corresponding to the candidate list. This means that the voters’ choices are not correctly recorded. Voters can however verify the correctness of ballot forms by casting dummy votes, or using a two-sided ballot form approach, where the voter verifies a random side and uses the other side to vote. Other threats to integrity include fake WBBs and the fact that the WBB could be modified after the voter has verified her receipt.

The availability of the scheme may be violated through ballot spoiling or faking attacks, or deletion of mix administrators’ keys. The latter threat is especially true for the “classic” version of Prêt à Voter which uses decryption mixes. A ballot spoiling attack could be a method to launch a DoS attack, since this would be discovered during randomised partial checking of the mix net.

Prêt à Voter is robust against most threats when considering the various countermeasures proposed for the scheme. A possible trade-off by introducing those countermeasures is a higher level of complexity that may weaken the voters’ understanding of the system.

References

1. Workshop on developing an analysis of threats to voting systems, 2005. <http://vote.nist.gov/threats/index.html>.
2. The machinery of democracy: Protecting elections in an electronic world (full report), 2006. Brennan Centre for Justice, NYU School of Law, <http://www.brennancenter.org>
3. Punchscan, 2006. <http://www.punchscan.org>.
4. VoteHere, 2006. <http://www.votehere.net/default.php>.

5. B. Adida and R.L. Rivest. Scratch & vote: Self-contained paper-based cryptographic voting. In *Workshop on Privacy in the Electronic Society*, 2006.
6. A. W. Appel. How to defeat Rivest's threeballot voting system. <http://www.cs.princeton.edu/~appel/papers/DefeatingThreeBallot.pdf>, 2006.
7. D. Chaum. Secret-ballot receipts: True voter-verifiable elections. *IEEE Security and Privacy*, 2(1):38–47, January-February 2004.
8. D. Chaum, P. Y. A. Ryan, and S. Schneider. A practical, voter-verifiable election scheme. In *European Symposium on Research in Computer Security*, number 3679 in Lecture Notes in Computer Science. Springer-Verlag, 2005.
9. C. Ellison. Upnp security ceremonies design document, 2003.
10. A. Fujioka, T. Okamoto, and K. Ohta. A practical secret voting scheme for large scale elections. In *Workshop on the Theory and Application of Cryptographic Techniques: Advances in Cryptology*, pages 244–251. ACM, 1992.
11. M. Gogolewski, M. Klonowski, P. Kubiak, M. Kutylowski, A. Lauks, and F. Zagorski. Kleptographic attacks on e-election schemes with receipts. In *International Conference on Emerging Trends in Information and Communication Security*, Lecture Notes in Computer Science. Springer-Verlag, 2006.
12. M. Jakobsson, A. Juels, and R. Rivest. Making mix nets robust for electronic voting by randomized partial checking. In *USENIX Security Symposium*, pages 339–353, 2002.
13. C. Karlof, N. Sastry, and D. Wagner. Cryptographic voting protocols: A systems perspective. In *USENIX Security Symposium*, 2005.
14. S. Kremer and M. Ryan. Analysis of an electronic voting protocol in the applied pi-calculus. In *European Symposium on Programming*, number 3444 in Lecture Notes in Computer Science, pages 186–200. Springer-Verlag, 2005.
15. A. Neff. A verifiable secret shuffle and its application to e-voting. In *Conference on Computer and Communications Security*, pages 116–125. ACM, 2001.
16. A. Neff. Practical high certainty intent verification for encrypted votes, 2004. <http://www.votehere.net/documentation/vhti>.
17. T. Peacock. *Guess My Vote: a Study of Opacity and Information Flow in Voting Systems*. PhD thesis, School of Computing Science, Newcastle University, 2006.
18. R. L. Rivest. The ThreeBallot voting system. Unpublished draft, <http://theory.lcs.mit.edu/~rivest/Rivest-TheThreeBallotVotingSystem.pdf>, 2006.
19. P. Y. A. Ryan. Verified encrypted paper audit trails. Technical Report CS-TR-966, University of Newcastle upon Tyne, 2006.
20. P. Y. A. Ryan and T. Peacock. Prêt à voter: a systems perspective. Technical Report CS-TR-929, University of Newcastle upon Tyne, 2005.
21. P. Y. A. Ryan and S. A. Schneider. Prêt à voter with re-encryption mixes. In *ESORICS*, number 4189 in Lecture Notes in Computer Science, pages 313–326. Springer-Verlag, 2006.
22. T. Tjøstheim, T. Peacock, and P. Y. A. Ryan. A model for system-based analysis of voting systems. In *Fifteenth International Workshop on Security Protocols*, 2007.
23. C. E. M. Strauss. A critical review of the triple ballot voting system, part2: Cracking the triple ballot encryption. Unpublished draft, <http://cems.browndogs.org/pub/voting/tripletrouble.pdf>, 2006.

A Brief overview of the model

The model presented in [22] defines a set of components and associated threat categories to those components. The model was developed in a stepwise manner,

where we first introduced a base model for a simple manual voting system, such as the one currently used in the U.K. We then extended the base model to include various features, such as a voting device, paper audit trail, verifiable receipts, etc. The threat categories to each component were determined by looking at the direct threats that could violate the purpose and requirements of a component. When deciding whether a threat category applies or not, it is important to consider the details of the particular scheme and how a threat may be manifested.

When using the model, the main components of a scheme such as the ballot form, voting booth, etc. are identified and the possible threats to each component at each phase of the protocol are considered in turn. In this way, it provides a guideline for evaluation of the system with the detail of a protocol-level analysis, but at the same time taking interactions between the various components directly into consideration. An advantage of the model apart from offering a more systematic approach to analysis, is that the components can be selected as appropriate and thus tailored to the scheme being analysed. In addition, by working through the threat categories in the model, and at the same time applying reasoning as appropriate to the scheme, the analyst is arguably better able to identify new threats than if using a catalogue of threats.

The model was designed to be as general as possible, so that it can be used for a range of different systems: from manual, paper-based voting, such as the current UK system, to more sophisticated systems that make use of, e.g. voting devices and verifiable receipts.

The possible threats associated with each component are given in Figures 3 - 10. Note that for all components in the model, the property violated is listed alongside each threat. Here, we consider the main properties required of secure systems, i.e., confidentiality, integrity and availability, rather than the traditional requirements of voting systems such as ballot secrecy, accuracy, verifiability, etc [14, 10].

Note that the model does not include certain threats such as forced abstention due to shortage of election equipment, complex registration, etc., as these are generally due to forces outside the system and need to be addressed by means other than improvements in the protocol.

Threat	Property violated
Identifiable information added by voter/official	Confidentiality
Voter identifiable from ballot form	
Authority knowledge	
Voter's choice incorrectly represented	Integrity
Ballot form spoiled	
Ballot form faked	

Fig. 3. Ballot form

Threat	Property violated
Voter's activity monitored	Confidentiality
Voter records own choice	
Voter's choice influenced	Integrity
Voter smuggles out unmarked ballot form	

Fig. 4. Voting booth

Threat	Property violated
Ballot stuffing	Integrity
Ballot spoiling	

Fig. 5. Ballot storage

Threat	Property violated
Early publishing	Integrity
Absence of verifiability	
False/erroneous count	

Fig. 6. Election results

Threat	Property Violated
Identifiable information added	Confidentiality
Voter's activity monitored	
Faulty authorisation	Integrity
Voter's choice incorrectly/not recorded	
Denial of service	Availability

Fig. 7. Voting device

Threat	Property violated
Voter identifiable from receipt	Confidentiality
Voter's choice noted by official	
Mismatch between voter's choice and paper copy	Integrity

Fig. 8. Paper audit trail

Threat	Potential threat
Voter identifiable from receipt	Confidentiality
Authority knowledge	
Receipt discarded/ surrendered	Integrity
Invalid signature	
Faked receipt	

Fig. 9. Verifiable receipt

Threat	Potential threats
Monitoring access to the Web bulletin board (WBB)	Confidentiality
Voter presented with fake WBB	Integrity
WBB modified	
Denial of service	Availability

Fig. 10. WBB