

# Polsterless Remote Electronic Voting

Tim Storer and Ishbel Duncan\*

19th December, 2003

## Abstract

Remote electronic voting is currently being piloted in the UK as a means of increasing the convenience of casting a ballot, which it is hoped will be reflected in an increased participation in elections. Most proposed electronic voting schemes envisage the use of cryptography in order to model the features of democratic elections, which, informally, include notions such as the secret ballot and a verifiable tallying system. This approach requires the use of a software artifact, or *polster*, which casts a ballot on the elector's behalf. A consequence of this approach requires the elector to trust software supplied by the election authority, as well as limiting the range of devices on which the ballot may be cast. An alternative to the use of cryptography employs a *polsterless* electronic voting system. Here, a proposed polsterless system for UK elections is considered and the flaws identified. A revised scheme is then proposed that provides verifiability and improved resistance to abuse, without requiring too much additional participation from the elector.

**keywords** remote electronic voting, verifi ability, polster.

---

\*University of St. Andrews. {tws, ishbel}@dcs.st-and.ac.uk. This work is supported by Microsoft Research.

# 1 Introduction

A commonly cited obstacle to the implementation of electronic voting in the UK is the so-called ‘digital divide’ [Pratchett, 2002]. The problem identified by this term describes the differentiation in access to new digital technologies, typically termed Information and Communication Technologies (ICTs) by government and service vendors, within the general population. For example, there is an assumption amongst the general public that REV will be conducted via the Internet [Communications, 2002]. However, growth in the number of households with home access to the Internet has slowed dramatically in recent years, stalling at around 51% of homes [MORI, 2003]. The technology barrier is exasperated by the differentiation in access between specific groups within the general population. A number of studies of e-democracy have shown that the economic groups who are least likely to vote (the group the Government is most keen to engage in the democratic process) are also the group least likely to have access to the Internet [Pratchett, 2002]. Although such technology is now available in public places such as libraries, fears have been raised about the heightened threat of coercion in these environments. Curiously, the already recognised practice of ‘family voting’ in the UK has not raised similar concerns about voting from home [Pratchett, 2002].

Whilst the causes of reduced turnout for elections is complex, the convenience of casting a ballot is certainly significant. Several surveys have identified non-electors who cited inconvenience as the main reason for not casting a ballot. This fact may well have surprised the government, since postal ballots have been made increasingly easy to obtain, and a record number were cast at the last election [House of Commons, 2000, The Electoral Commission, 2001]. The failure of postal voting to halt the decline in turnout at the last general election may partly be due to lack of awareness of this option on the behalf of electors [The Electoral Commission, 2001]. Encouragingly, the use of postal voting is believed to be responsible for a far more significant increase in turnout during pilots at local elections in 2002 and 2003 [The Electoral Commission, 2002,

The Electoral Commission, 2003].

## 1.1 Electronic Voting in the UK

Therefore, despite the difficulties, the UK Government has expressed a preference to conduct an e-enabled General Election sometime after June 2006 (the latest date for the next election). Remote Electronic Voting (REV) has conventionally been considered an application of cryptography because of the perceived need to protect ballots during transmission [Rjašková, 2003]. Various cryptographic constructs have also been proposed to model the anonymising effect of a conventional ballot box, the use of mix-networks, for example, where some mathematical proof is established that the output tally corresponds to the input ballots [Sako and Kilian, 1995]. As noted, a difficulty of implementing this approach is the less than universal access to technology capable of performing the necessary computations. Further, the elector is required to trust a software artifact, or *polster*, supplied by an election authority to correctly cast a ballot on their behalf [Malkhi et al., 2002].

In contrast to other ICTs, 75% of the population in the UK own a mobile phone, a figure which is relatively consistent across social groups [MORI, 2003]<sup>1</sup>. Further, some 87% of the UK population between the ages of 15 and 24 possess a mobile phone [OFTEL, 2003]. Given that the turnout for citizens aged between 18 and 24 at the 2001 General Election was believed to be just 39%, against the figure of 59.4% for the general population [The Electoral Commission, 2001], the use of mobile phones for casting a vote may well be worth investigating.

The market penetration of mobile phones then, suggests a suitable technology with which to overcome the barriers to implementing REV. A *polsterless* system exploiting access to mobile phone telephony would permit an elector to verify their ballot was correctly cast and tabulated, independently from the tabulating authority. This paper provides a brief overview of a *polsterless* system proposed by a UK government agency [CESG, 2002a], and outlines a number of flaws that

---

<sup>1</sup>Mobile phone ownership ranges from 68–83% vs, for example Internet access which ranges from 25%–76%.

were identified. The system chosen, whilst having the advantage of requiring little input and no computation on behalf of the elector, relies strongly upon a central election authority behaving correctly. To counter this requirement, this paper proposes a re-design of the original system in order to allow each elector to independently verify that their ballot was tabulated correctly. A further advantage of the redesign permits any external observers to verify the ballots collected by the tabulating authority against the final tally, without violating the anonymity of the elector.

## 1.2 Related Work

Relatively recent work on electronic voting has been conducted on removing the need for a client side cryptography, for example, the use of *advanced check vectors* to provide verifiability [Malkhi et al., 2002], where the importance of *polsterless REV* was identified. In summary, a dealer (the election registrar) delivers sets of vectors of values to intermediaries (the elector), along with a corresponding secret  $s$  for a group of the vectors (the candidates). In order to cast a vote, the intermediary sends a Vector  $V$  to a receiver (the tallier) who returns a check vector  $B$  to the intermediary. The elector then confirms  $VB = s$  in order to obtain a receipt for the vote. The system requires the prior-establishment of *secure channels* between electors and the election authorities.

The system proposed is interesting, because of the novel approach to implementing a trusted voting client (by eliminating it). Unfortunately, from the system design, it is clear that the electors would still need to perform a considerable amount of additive computation, whilst a *polster* would still be required to communicate with enough election authorities to prevent cheating. This feature makes the approach proposed in-practical for general use, since the system could not reasonably be described as convenient. Nevertheless, the recognition that electronic voting should not require complex client-side computation was a significant step forward.

## 2 The CESG e-Voting System

A recent security study published by the Communications and Electronic Security Group proposed a Remote Electronic Voting (REV) system that could be adopted by the UK Government for the purpose of conducting elections [CESG, 2002a]. The design of the system was attractive because the necessity for client side computation was eliminated, allowing a variety of devices to be used. Further, since the system did not rely upon a polster the elector did not need to trust that a cryptographic program had correctly cast a vote on their behalf.

After publishing the study in 2002, CESG received official responses from a variety of interested parties. The responses included concerns about the general conclusions of the study [Fairweather, 2002, Kitcat, 2002, Excelsior Consultancy, 2002], but also alluded to flaws in the proposed system itself [Mercuri, 2002]. The level of criticism received by CESG may have influenced the decision not to include the proposed electronic voting mechanism in their subsequent recommendations on the security requirements for REV [CESG, 2002b]. This decision was unfortunate, since the basic premise of the protocol, allowing ballot casting over virtually any medium, was worthy of further investigation. Whilst, it is undisputed here that the current form of the system is unsatisfactory, it is believed that the design could be revised to incorporate more acceptable properties of REV. Such an improved system (incorporating features such as verifiability and anonymity) could prove beneficial for those electors who find (for whatever reason) attending a polling station too inconvenient.

### 2.1 Outline of the Voting Mechanism

The scope of the protocol is limited to voting and tallying - registration of electors is assumed to have taken place prior to the dissolution of parliament (and subsequent issue of writ in each constituency). Thus, an election authority is assumed to have an electoral roll detailing the  $m$  electors in the constituency with an electoral roll number, name and address. To initiate the

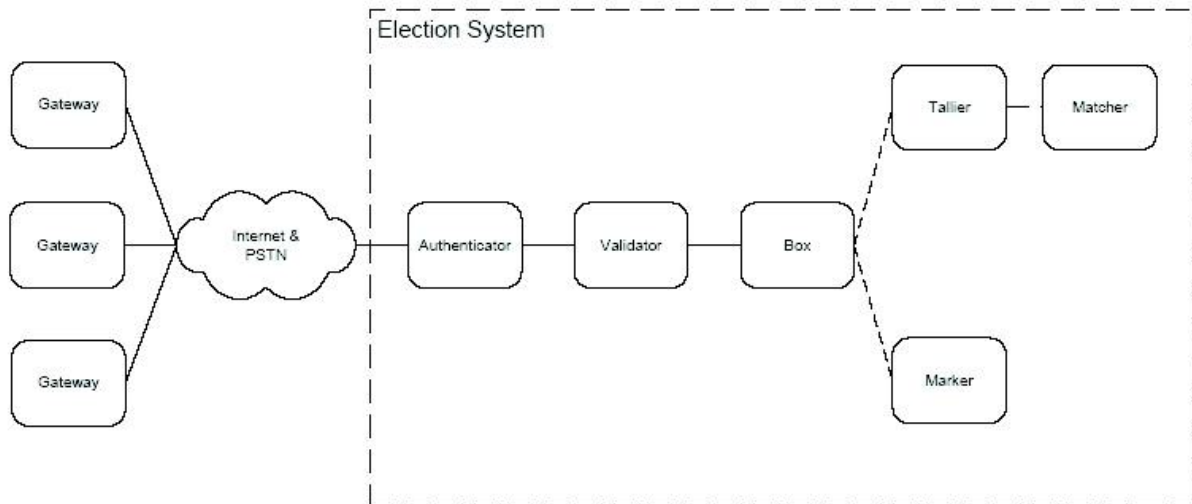
Name:	Joe Bloggs	
Address:	NoNumber, NoStreet, NoTown.	
VoterID:	4545 2321 6742 1209	
Candidates	PCIN	RID
M. Howard	7890	109223
T. Blair	4312	341789
C. Kennedy	2125	841739

**Figure 1:** Proposed layout for the CESG voting card. The security study suggested printing the voting card on pay-roll stationary in order to reduce the threat of personation. The delivery of the voting card to the voter would incur minimal cost, since this would essentially be an extension of the ballot card already sent to voters in the UK notifying them of an upcoming election.

election, each elector receives a set of voting credentials, illustrated in figure 1. The system envisages the use of a one-way secure channel to deliver the credentials and suggests the postal system as a suitable approximation to this requirement.

The electronic credentials are essentially an extension of the ballot card already delivered to electors prior to an election, containing information on polling place locations and when to vote for example. The extra electronic credentials consist of an elector (voter) identification number (*vid*), a set of personal candidate identity numbers (*pcins*) and a corresponding set of return identity number (*rids*). Each pairing of *pcin* and *rid* values correspond to a single candidate on the ballot. The CESG study provides some suggestion of how these electronic credentials might be securely published, for example using payroll stationary. However, there is no explanation of how these credentials will be securely stored within the voting authority and then distributed prior to the initiation of the election.

A ballot is cast by sending the *vid* and a *pcin* of choice to a *gateway* chosen by the elector, such as sending a text message using an SMS enabled mobile phone. After some delay period they should receive back an *rid*, which they check to match against the *rid* corresponding to their choice on the ballot. Should the two numbers not match, the elector should contact the election's administrator.



**Figure 2:** Architecture of the e-voting system proposed by CESG for vote collection and tabulation.[CESG, 2002a] The diagram indicates the path of a vote from the elector, via a gateway to the voting authority domain.

## 2.2 SureVote

The system outlined above has not, to our knowledge been employed in a remote electronic election. However, it is understood that a similar system has been employed for polling stations equipped with Direct Recording Electronic (DRE) machines. The system developed by SureVote, was noted by one author because of possible usability issues [Bederson et al., 2003] and has been outlined in more detail elsewhere [Chaum, 2001]. In brief, a DRE machine is securely connected to an election server, presumably located away from the polling station. Electors are issued ballot papers as in the CESG system and then cast their ballots on the DRE machine’s console as described above. The DRE machine then transmits the vote to the server. The server then responds with a “sure code” (*rid* in CESG’s system), which the elector checks against the ballot paper they were issued.

## 2.3 Architecture

Figure 2 is taken from the CESG security study and illustrates the authors' concept for the voting authority design. The study proposed that communications between the distinct elements in the architecture will be through pre-arranged encryption, suggesting they will be organised in some autonomous fashion within the authority domain.

The range of gateways available at an election will likely be dependent upon cooperation between government and the private sector, mobile phone operators, Internet service providers etc. After the elector has sent a  $\{vid, pcin\}$  combination to a gateway it is then forwarded to the *Authenticator* which stores a list of valid *vids* and the key for computing a corresponding *rid*. Once the received *vid* is authenticated, an *rid* is computed on the  $\{vid, pcin\}$  combination and a  $\{vid, rid\}$  combination is sent to the *Validator*. The Validator stores a list of valid *rid* for each *vid*. The Validator may then confirm or deny the Authenticator the *rid* generated.

Once the ballot has been validated, the Authenticator forwards the received *rid* to the elector via the original gateway medium used. The remaining work performed by the architecture stores and then later tabulates the ballot. At this point in the scheme's description, the description becomes somewhat vague. From the study

“The ballot is sent to the ballot box with the corresponding RID”.

Unfortunately, the study does not describe how the ballot is sent to the ballot box. The diagram indicates that the ballot is sent to the ballot box from the Validator, although there is no corresponding description in the text that the  $\{vid, pcin\}$  combination is sent to the Validator prior to being sent to the ballot box. If the ballot is sent via the Validator, then it is not clear as to the purpose of separating the Authenticator from the Validator, since the Validator observes the *pcin* eventually anyway. For the purpose of the analysis, we assume that the ballot is sent to the ballot box via the Validator.

After the deadline for casting votes is reached (in the UK currently 10pm on election day)

the [ballot] Box store is passed to the Tallier, which matches the  $\{vid, pcin\}$  combination of each vote against an anonymous candidate identification (*cid*) value to produce a tally for each. Finally, these tallies are passed to the matcher, which pairs each of the tallies with a candidate, according to the corresponding *cid*.

## 2.4 Flaws of the CESG System

### 2.4.1 Threat Model

Informal analysis of the CESG voting system revealed a number of flaws. In order to categorise them, the following threat model is adopted. The following types of attacker were identified:

- A malicious election authority domain, intent on either denying an elector a vote, or observing how they voted. Such an attacker would be comparable to an abusive regime or government, intent upon thwarting the democratic process.
- A malicious elector intent upon causing disruption to the election process in which they are entitled to vote.
- A malicious external attacker intent upon undermining the democratic process in the United Kingdom. The CESG study identified such attackers as foreign espionage agents, criminal organisation, protests groups or even investigative journalists [CESG, 2002a, pp.22].

Such attackers would be intent upon undermining the democratic process. Their aim may be to commit some abuse which the system must be robust enough to resist, such as tampering with the result or discovering how an elector decided to cast their ballot. Alternatively, they would not necessarily need to actively commit abuse, if doubt may be created simply by claiming to have done so. Contests to election results are not uncommon in the United Kingdom. Observers have suggested that without being able to properly verify an election, the result would

simply have to be declared void and re-run, raising the prospect of the same contests being raised.[Jackson and Syddique, 1991]

Having established the likely attackers on the election, it is also possible to demonstrate several exploitable flaws. The analysis focuses upon the receipt of the *rid* by the elector after having cast a vote. The receipt allows the elector to ensure that their vote was collected by the participant that can generate *rid* values. However, it was realised that this operation was worthless for both the elector and the election authority if either attempts to cheat the other.

#### **2.4.2 Verification of the Result**

From the elector's perspective, it is not possible to establish that the  $\{vid, pcin\}$  combination was sent to the ballot box by the Authenticator after validation, or that the series of translations results in a vote correctly cast for the elector's choice. On receipt of the correct *rid*, the elector is entitled to believe the election authority received their vote, but not that it contributed to the tally for the candidate of their choice. Since the *rid* is already in their possession, the elector cannot prove to any third party that they received the *rid* from the election authority. The *rid* is therefore not a receipt of a elector's ballot, since the elector cannot use the return to prove to either a third party or the election authority that they cast a vote that was not counted.

#### **2.4.3 Un-deniability**

Similarly to above, the election authority cannot prove to a third party that it did not receive a ballot from an elector and chose to arbitrarily ignore it. Again, the difficulty stems from the lack of verifiability in the election system. For example, a malicious elector may complain that they cast a ballot, received the correct  $\{rid\}$  in response, but their name did not appear in the marked electoral roll after the poll has closed and the result has been announced. The election authority cannot then show that it did not receive the ballot, since a receipt is pre-issued in the voting credentials at the start of the election - the election authority cannot force the elector to

reveal whether a copy of that receipt was received during voting or not.

This vulnerability is similar to the current difficulty caused by procedures for handling *personation* at polling stations. In these circumstances, an elector whose identity appears on the marked roll<sup>2</sup> may deny having cast a ballot. Since the election authority (from the perspective of the elector) is in possession of the voting credentials of all electors, the election authority cannot be unable to show that it did not cast ballots on behalf of non-voting electors.

A respondent to the original consultation suggested that the identities of electors who cast electronic ballots should not be published in order to improve security[Fairweather, 2002]. This would actually worsen the flaw, since a malicious election authority would be able to cast additional ballots without external challenge.

#### **2.4.4 Anonymity for the Elector**

The external anonymity of a ballot is well protected, since there is no external connection between the electors' identities, the voting credentials transmitted to the election authority and the final tally. However, from the perspective of the elector, the election authority knows both the *vids* and the *cids* that correspond to their voting credentials. A malicious election authority would be able to monitor ballots as they were cast, determine who they were cast by and even modify them if required.

### **3 Revising the CESG Approach**

Having considered the short analysis given above it may seem that the CESG voting system is virtually beyond repair, given the serious verifiability vulnerabilities demonstrated. The initial responses to the security study raised such concerns in a general sense about possible flaws in the voting system. To quote one of the responses,

---

<sup>2</sup>a list produced after an election of the identities of those electors who cast a ballot

“The system described in the report is typical of the proposals that have been deemed flawed by noted cryptographers such as Bruce Schneier and David Chaum, and security expert Peter Neumann, as well as many other computer scientists and researchers who have been commenting on this subject for the past decade.”[Mercuri, 2002]

The responses to the survey were perhaps helpful to CESC in order to indicate the opposition of computer scientists to remote electronic voting. However, it is proposed that the above consideration of the weaknesses in the CESC voting system clarify that the underlying concept is sound - the importance of polsterless vote casting for the purpose of verifi ability. Rather than discarding the general technique more careful design may yield a polsterless, elector verifi able remote electronic voting system. The remaining sections formalisms the CESC system as a cryptographic protocol, with the necessary participants and functions specifi ed. The protocol is then revised in order to introduce verifi ability and to a lesser degree, elector anonymity.

### **3.1 Formalisation as a Protocol**

#### **3.1.1 Election Setup**

The system responsible for collecting votes is modeled as a set of discrete processes, the set of which is denoted *ElectionAuthority*. In the model a single process is responsible for initiating the election, denoted *ElectionSetup*, which distributes credentials to electors. ElectionAuthority also provides the other participants in the domain with the relevant information in order to process ballots. For initiation, the ElectionSetup process requires the following initiation parameters.

- $m$  candidates each with a unique `candName:String`
- $n$  electors each with a unique `electorName:String`
- secret government elector identification number generator key  $K_{VID}$

- secret government candidate identification number generator key  $K_{CID}$
- secret government personal candidate identification number generator key  $K_{PCIN}$
- secret government return identity generator key  $K_{RID}$
- the number of digits  $len_{VID}$  of a  $vid$  value
- the number of digits  $len_{PCIN}$  of a  $pcin$  value
- the number of digits  $len_{RID}$  of a  $rid$  value
- the number of digits  $len_{CID}$  of a  $cid$  value

**Functions** The operation of ElectionSetup may now be specified in more detail. The following four arrays are defined as the stores of voting credentials for electors. For each array, a corresponding function is defined specifying the required parameters.

- let  $cid_{1 \leq j \leq m} = \mathbf{CID}$ .

**genCIDs**( candNames:String[], len<sub>VID</sub>:int, K<sub>CID</sub>:byte[]):int[]

*Computes a unique candidate identity number for each String of candidateNames.*

- let  $vid_{1 \leq i \leq n} = \mathbf{VID}$ .

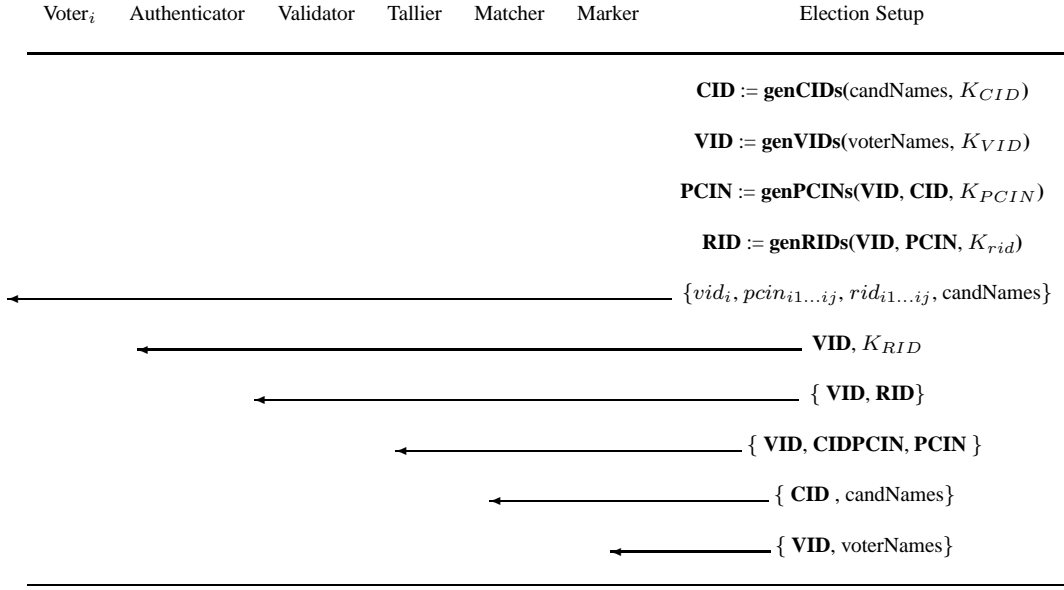
**genVIDs**( voterNames:String[], len<sub>VID</sub>:int, K<sub>VID</sub>:byte[]):int[]

*Computes a unique voter identity number for each String of voterNames.*

- let  $pcin_{1 \leq ij \leq nm} = \mathbf{PCIN}$ .

**genPCINs**( VID:int[n], CID:int[m], len<sub>PCIN</sub>:int, K<sub>PCIN</sub>:byte[]):int[][]

*Computes the non-unique personal candidate identity numbers between a candidate and a voter.*



**Figure 3:** Initiation of the protocol between the processes of the ElectionAuthority domain and a single voter. ElectionSetup uses four functions to generate arrays of hashed message authentication codes (HMACs) based on initiation parameters. These arrays are then distributed amongst the remaining processes of the domain in order for them to perform their specified tasks (authentication, validation) etc. Voter<sub>i</sub> receives values  $vid_i$ ,  $pcin_{i1...ij}$  and  $rid_{i1...ij}$  from the arrays **VID**, **PCIN** and **RID** respectively, together with candNames to provide a set of voting credentials.

- let  $rid_{1 \leq ij \leq nm} = \mathbf{RID}$ .

**genRIDs**( VID:int[n], PCIN:int[n][m],  $len_{RID}$ :int,  $K_{RID}$ :byte[] ):int[][]

*Computes the non-unique return identity numbers between candidates and voters.*

In order to generate the values for the four arrays, the e-voting study proposed the use of cryptographic Message Authentication Codes (HMACs). The sub-function genHMAC is defined below for the purpose of generating a single value. Each of the above functions executes genHMAC once for each value of the array they generate.

- **genHMAC**( input:byte[],  $len_{output}$ :int,  $K_{input}$ :byte[] ):int

*Computes the non-unique return identity numbers between candidates and voters.*

Figure 3 summarises the construction of the four arrays by ElectionSetup and the distribution of these arrays and other values to the remaining components of the election authority. In summary, Authenticator is sent the **VID** in order to authenticate a  $\{vid, pcin\}$  combination when

received from the gateway.  $K_{RID}$  is used to compute the corresponding  $rid$  for the Validator. Validator is sent the **VID** and **RID** arrays in order to validate the  $vid, pcin$  received by  $rid$ . Tallyer is sent the **VID** array and an extra construction **CIDPCIN**. This is a  $n \times m$  array of  $cid$  HMACs denoted

$$cidpcin_{1 \leq ij \leq nm} = \mathbf{CIDPCIN}$$

This construction indexes  $cids$  per elector against the  $pcins$  with which they will cast their ballot, such that

$$\text{genHMAC}(vid_i + cidpcin_{ij}, len_{pcin}, K_{PCIN}) = pcin_{ij}$$

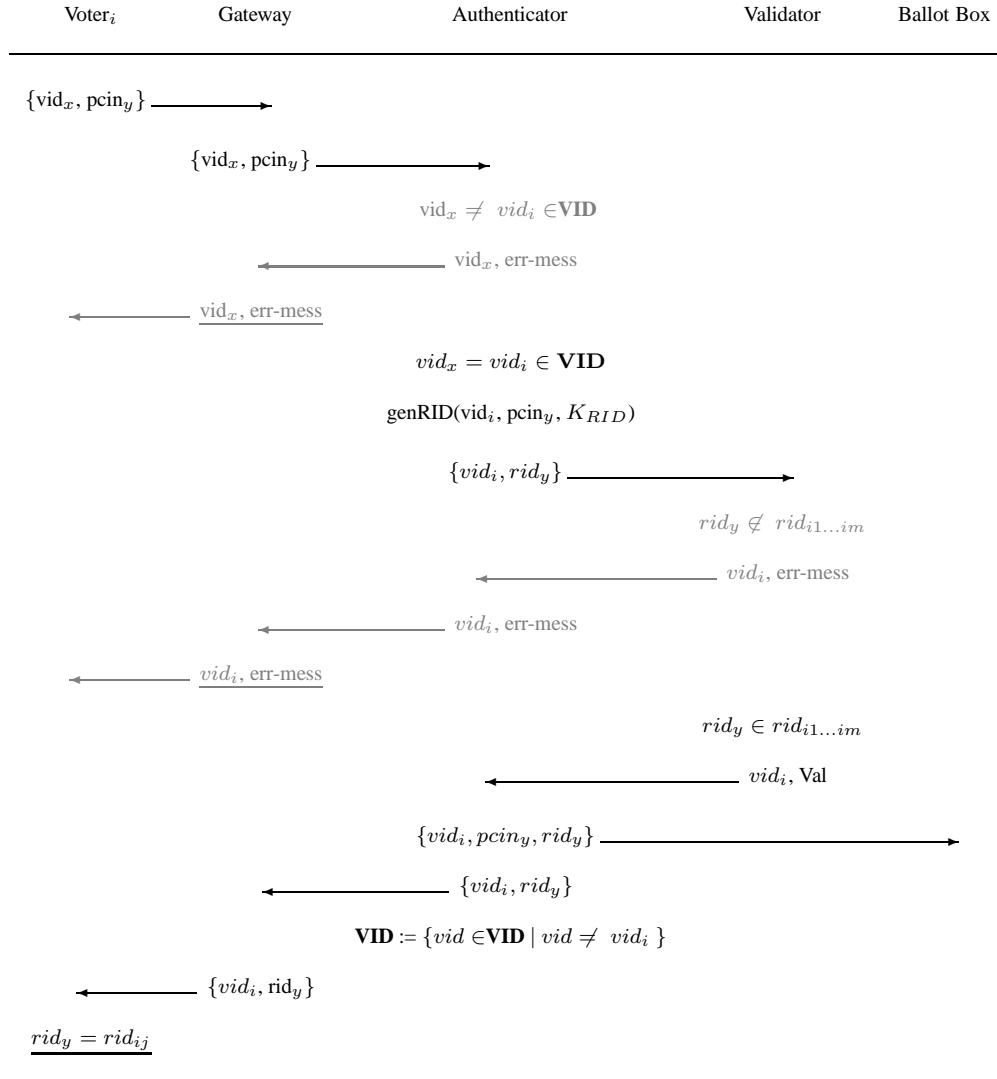
Finally, the Matcher receives the arrays **CID** and  $candName_{1...m}$  such that

$$\text{genHMAC}(candName_j, len_{cid}, K_{PCIN}) = cid_j$$

This arrangement allows the Matcher to output the tally obtained by each candidate at the end of the protocol.

### 3.1.2 Ballot Casting

Figure 4 summarises the ballot casting sub-protocol between a single elector and the processes of the ElectionAuthority domain. The sub-protocol has three legal termination points, all of which are the result of messages being received by the elector. The protocol terminates when the elector receives a message via the gateway, stating either that the  $vid$  has not been recognised, the  $\{vid, pcin\}$  combination has not generated a valid  $rid$  for that  $pcin$ , or that the  $\{vid, pcin\}$  combination has been validated successfully. Under the first two conditions, the elector may re-initiate the voting protocol, since no valid ballot has been acknowledged by the election authority.



**Figure 4:** Successful execution of the ballot casting protocol, reaching termination and a vote successfully collected. Faint messages in the protocol indicate early protocol termination sequences if the  $\{vid, pcin\}$  combination does not contain a legal  $vid$ , or does not compute a legal  $rid$ .

Under the third condition, the elector is expected to assume the ballot has been successfully stored in the ballot box and further votes will be ignored. Note that the elector is not actually prevented from re-initiating the protocol once a valid ballot has been received. Instead, the elector's ID is removed from those acceptable to the Authenticator, and new instances of the protocol will be terminated under the first condition.

In order to compute an *rid* for each vote received, the following function is specified for the Authenticator:

- **genRID**( *vid*:int, *pcin*:int ):int

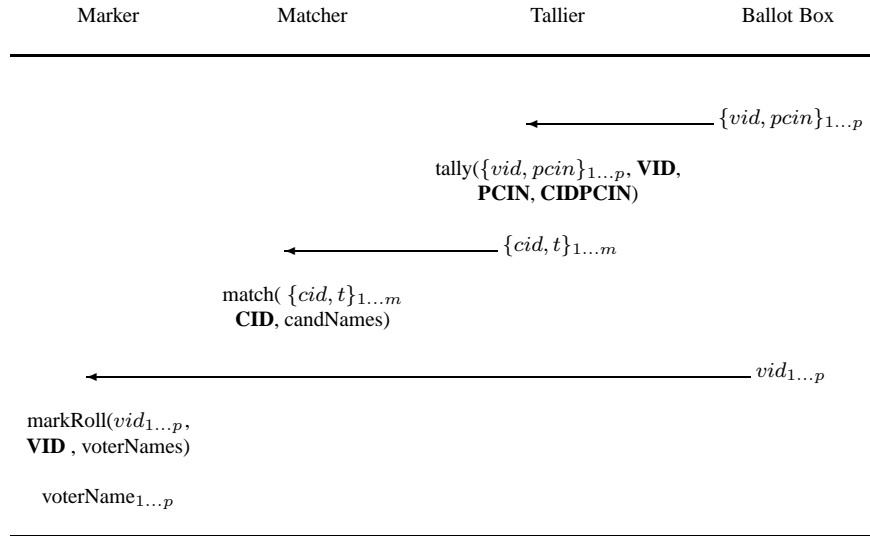
*Computes a non-unique return identity number between a candidate and a voter.*

Given the previously specified **genHMAC**() function, it may be noted that **genRID**() may be implemented as:

- **genHMAC**( *vid<sub>i</sub> + pcin<sub>ij</sub>*, *len<sub>rid</sub>*, *K<sub>RID</sub>* ):int

### 3.1.3 Tallying

Once voting has been completed, the tallying protocol may be initiated. During voting, it is expected that some  $p'$  voters will attempt to cast a vote, of whom  $p$  (the turnout) will do so successfully. Figure 5 summarises the transfer of HMAC values between components of the voting authority in order to obtain a tally for the  $m$  candidates. Initially, ballots are transferred to the tallier in the form  $\{vid, pcin\}$ . The tallier then obtains the *cid* for each ballot. Using the **VID** and **PCIN** arrays, the tallier obtains an index into the **CIDPCIN** array. The tally  $t$  for that *cid* is then incremented. Once all the ballots have been processed, the tallies, together with their corresponding *cid* are passed to the matcher. Using the **CID** and **candNames** arrays, each tally is then matched to a candidate to produce the result of the election.



**Figure 5:** Sequence of messages and computation during the tallying and marking processes at the end of the election. The output of the tallying protocol is a list of candidate names with a corresponding tally (the election returns) and a list of voter names (the marked roll).

**Functions** A function is specified for each of the processes of the ElectionAuthority domain in order to compute the result of the election. Most of the parameters for the functions are supplied by ElectionSetup during the initiation of the protocol.

- **tally**( ballots: {int, int}[] VID: int[], PCIN: int[][], CIDPCIN: int[][] ): {int, int}[]

*For each {vid, pcin} combination received, the corresponding cid is obtained from cidpcin array and its tally incremented. The function outputs the tally for each cid.*

- **match**( tallies: {int, int}[], CID: int[], candNames: String[] ): {String, int}[]

*Replaces each cid in the tally with the corresponding candName, revealing the result of the election.*

- **mark**( VIDs: int[], register: {VID: int, voterName: String}[] ): String[]

*Outputs a list of voterNames corresponding to the received list of vids that were used to cast votes.*

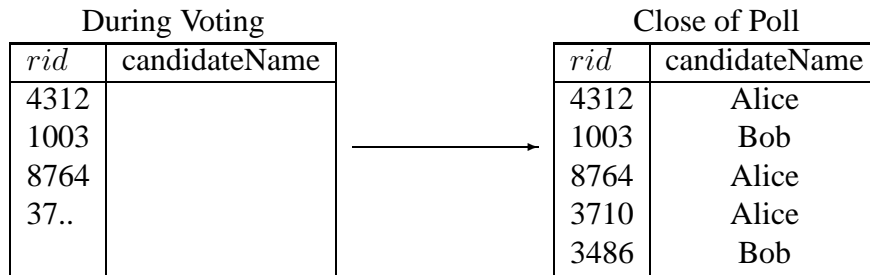
In addition to generating the tallies, Tallier also transfers the array of elector identity numbers  $vid_{1..p}$  collected from received ballots to Marker. Using **VID** and `electorNames` arrays, Marker outputs an array of the names of electors who cast a ballot (known as the marked roll).

### 3.2 Verifiability and Un-deniability

As noted in section 2.4, the CESG system is neither verifiable (from the perspective of electors or external observers) or undeniable (from the perspective of the ElectionAuthority. Although the system attempts to furnish the elector with a receipt with which they may verify the result, its effect is only to assure the elector that the ballot has been correctly received by the ElectionAuthority. The elector is unable to determine whether the ballot is subsequently tallied correctly. A further problem from the perspective of the ElectionAuthority is that it cannot prove that it did not receive a ballot from an elector, since all receipts for ballots are pre-issued with the voting credentials.

So rather than privately acknowledging to the elector having received a ballot, a more suitable approach would be to acknowledge the ballot in public, such that it may not be denied later on. Similarly, should the elector not demand their ballot be acknowledged prior to the close of poll, the ElectionAuthority should be able to legitimately deny having received the ballot. Crucially with this approach, there is the requirement that the secrecy of the ballot should be preserved.

In order to implement the public acknowledgment system, a new process is required, denoted *Publisher* that operates in a similar manner to an electronic bulletin board, a common cryptographic construct. *Publisher* is implemented as universally readable and ElectionAuthority may also append *rid* values to the bulletin board as they are generated from  $\{vid, pcin\}$  combinations. Thus, the process of casting a ballot is identical to the original CESG e-voting system. However, after having cast their ballot, the elector then proceeds to check that the corresponding *rid* value on their voting credentials appears on the bulletin board. If the value does not appear after some period of time then the elector must assume the system has not collected their ballot and either



**Figure 6:** The Publisher bulletin board during ballot casting and after the close of poll. During ballot casting, the *rid* values are appended to the left hand column of the construct by the ElectionAuthority. After the close of poll the corresponding candidateNames are added to the right hand column. Since the *rid* value is a secret shared between the elector and the ElectionAuthority, ballot secrecy is preserved, whilst allowing verifiability. External observers are able to compute a tally based upon the unchallenged ballots published.

re-try, or contact the election administrator.

Having published the *rid* values during voting does not in itself produce a verifiable result, since the final tally of the election must be shown to correspond to the ballots cast. To achieve this, after the election, the ElectionAuthority publishes the identities of the candidates that corresponds to each of the *rid* values on the Publisher. Each elector may now confirm that the candidate associated with the *rid* value on the bulletin board corresponds with that on their voting credentials. Note that the *rid* value is a unique secret shared between the elector and the ElectionAuthority, such that an external attacker cannot tell which value published belongs to which elector. The *rid* values must be unique since if two ballots had the same *rid* value and represented the same candidate, the ElectionAuthority could use one of the ballots to ‘prove’ to two electors that their ballot were counted correctly, whilst using the second to cast a different ballot. Figure 6 demonstrates the sequence of publication for the *rid* values and candidate names. If the values do not match then the elector has the ability to challenge the ElectionAuthority on the basis of the supplied credentials. Further, since the candidates corresponding to the *rid* values are publicly available, the tally for the election is *universally verifiable*.

### 3.3 Ballot Anonymity

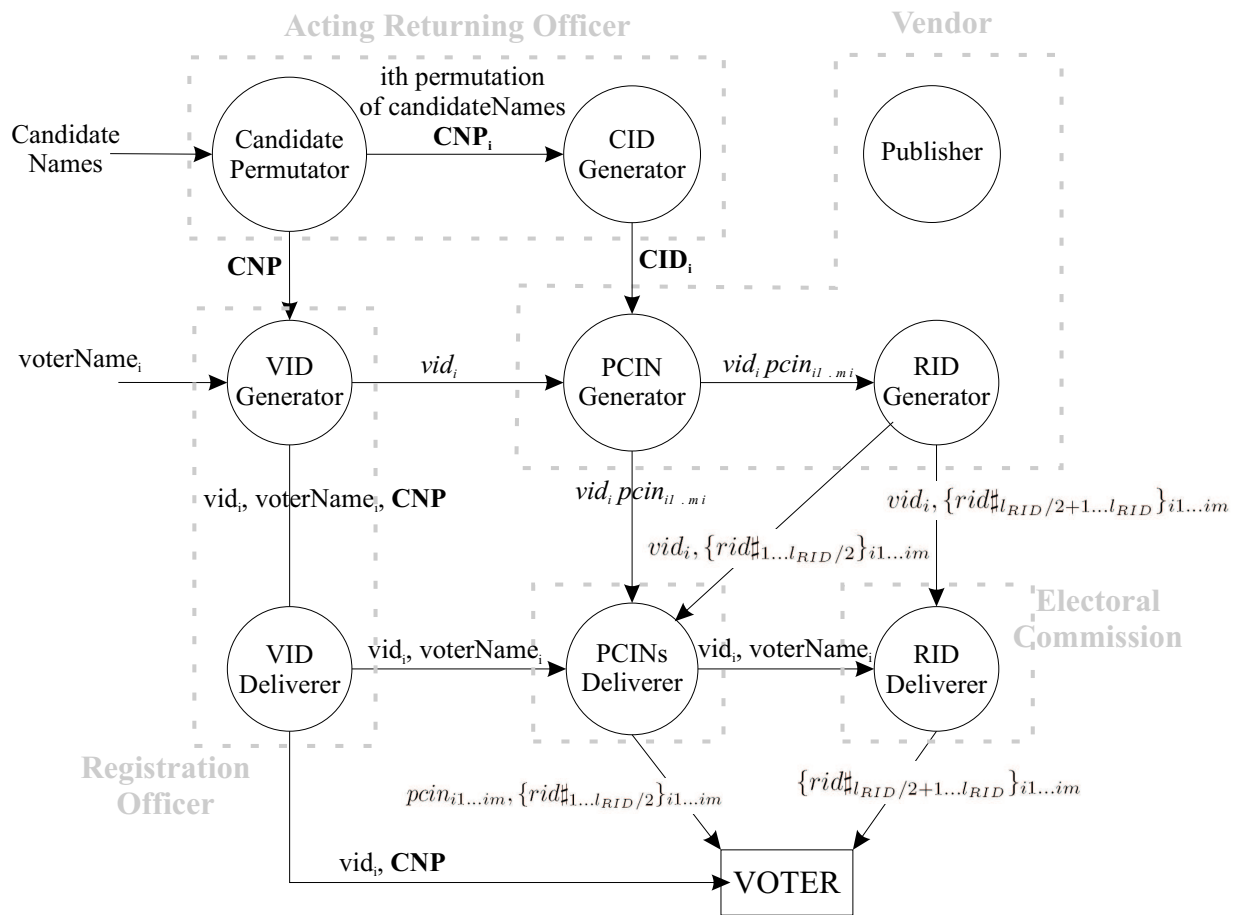
Although the modified system described above is now both verifiable and undeniable, a single process, ElectionSetup, is still used to generate and distribute the credentials for casting and processing of ballots. Such a design is vulnerable because (a) the process represents a single point of external attack and (b) electors must trust a single process not to behave maliciously with the anonymity of their ballot.

In addition, the use of the *rid* value as a unique secret shared between the ElectionAuthority and the elector, requires that the association between an *rid* value and an elector should not be known by anyone process of the ElectionAuthority domain. This is not a trivial task to accomplish, since if the *rid* value is to be delivered to the elector, then the delivering process must know the identity of the elector whilst at the same time possessing *rid* value material.

Figure 7 describes the redesign ElectionAuthority as a data-flow. The design provides for increased protection for anonymity than the single ElectionSetup process in the original CESC voting system. In the new design the ElectionAuthority domain is divided into several domains, each under the control of an independent organisation.

The initiation of the election thus proceeds as indicated in the figure. The key component of the process is the separation of the computation and delivery functions between different domains, preventing the domain that generates the *rid* values from knowing the identity of the elector to whom they will be delivered. This protects the anonymity of the elector based upon the assumption of non-collusion across domains.

The *rid* values are also divided into two parts and sent to two different domains for delivery to the elector. This division of values is denoted on figure 7 as  $\mathbf{RID}\#_{1\dots len_{RID}/2}$  and  $\mathbf{RID}\#_{len_{RID}/2+1\dots len_{RID}}$ . That is the first  $len_{RID}/2$  digits of each *rid* value are sent to the PCIN-Deliver, whilst the remaining digits of each *rid* value are sent to the Electoral Commission for delivery. Given that the *rid* values describe a unique correspondence between an elector and a candidate, delivers must not be allowed to know a complete *rid* value and the identity of the



**Figure 7:** Dataflow of an anonymous setup architecture for the CESG voting System

elector to whom they will be sent. If this condition is violated then when the *rid* values are published with their corresponding candidate, the deliver would be able to determine how the elector's voted.

An additional benefit of the re-design is that the voting system is resistant to *intelligent ballot stuffing* by the ElectionAuthority, again under the assumption of non-conclusion. In order to perform ballot stuffing, a process needs to know a  $\{vid\ pcin\}$  combination, whilst to be intelligent, the process must know which candidate corresponds to which *pcin* value. Although it is notable that the PCINDeliver or Vendor domains may commit *blind ballot stuffing* they cannot by themselves commit intelligent ballot stuffing.

The specification of **genCID()** also must be modified to ensure that a different set of *cid* values are generated for every elector. This is combined with a new function for returning officer, which permutes the order of candidates:

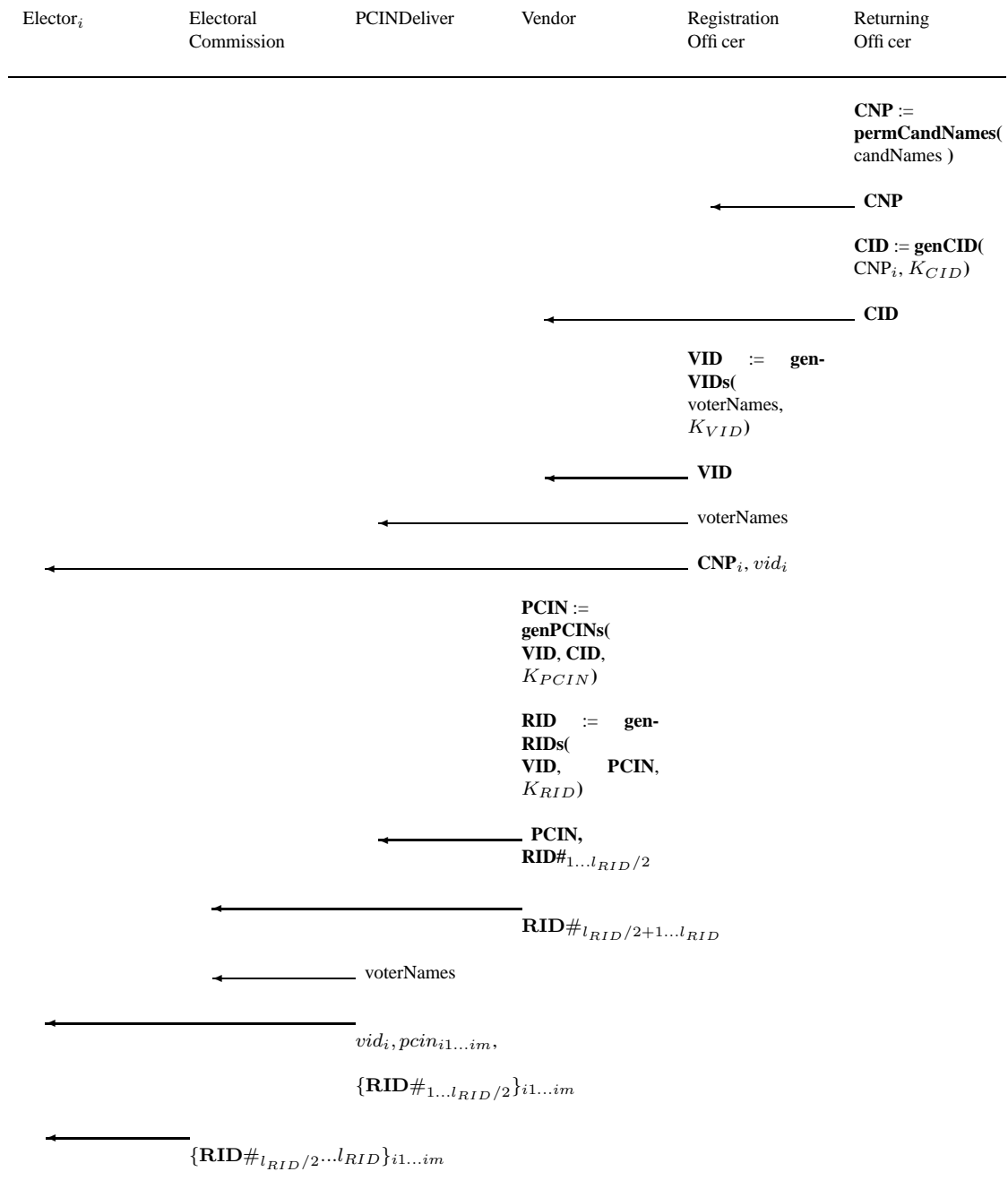
- **permCandNames( candNames:String[] ): String[]**

*Generates a random permutation of the candNames array.*

The effect of these modifications to the ReturningOfficer domain is to prevent a single elector from colluding with the Vendor to commit intelligent ballot stuffing. Without the modification, a single set of voting credentials could be used to match candidate names with the corresponding *cid* values.

### 3.4 The Complete Redesign as a Protocol

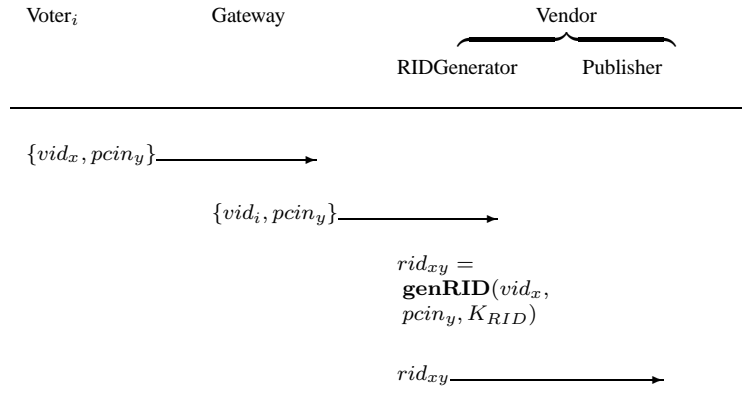
For ease of comparison with the original system, the complete re-design is now summarised as a cryptographic protocol. The summary demonstrates the re-use of the atomic components of the original ElectionSetup. Rather than placing the initiation parameters all in a single ElectionSetup, they are divided amongst the new domains. The domain are as follows:



**Figure 8:** The initiation of the re-designed polsterless e-voting system, see also figure 7. The protocol is similar to the original system, except that the initiation parameters begin distributed amongst independent domains. The domains distribute the minimum values to other domains in order for them to complete the initiation, but without allowing the domains to escalate their capabilities. The design allows for ballot anonymity, resistance to intelligent ballot stuffing by the authority (since no one domain knows enough of the voting credential) and also for the elector to verify that their ballot was tabulated correctly and that the overall result was tallied correctly from the ballots.

- RegistrationOfficer is modeled upon the same role in UK constituencies, maintaining the list of eligible electors.
  - Parameters:  $electorNames$ ,  $len_{VID}$  and  $K_{VID}$ .
  - Functions: **genVID()**.
- ReturningOfficer is modeled upon the same role in UK constituencies, accepting nominations for candidates.
  - Parameters:  $candNames$ ,  $len_{CID}$  and  $K_{CID}$ .
  - Functions: **permCandNames()** and **genCID()**.
- Vendor is the domain controlled by the organisation chosen to run the e-election.
  - Parameters:  $len_{PCIN}$ ,  $len_{RID}$ ,  $K_{PCIN}$  and  $K_{RID}$
  - Functions: **genPCIN()** and **genRID()**.
- ElectoralCommission Modeled upon the independent organisation that observes and reports upon elections in the United Kingdom. The Electoral Commission adopts the role of delivering  $rid$  values.
- PCINDeliver An organisation for the delivery of  $pcin$  and part of the  $rid$  values.

Figure 8 provides a formalisation of the distributed setup which is comparable to the operation of ElectionSetup in the original CESG system. Although for clarity, functions are associated with particular processes, it is not anticipated or considered necessary for each of these domains to keep the implementation of these functions secret. Rather, a sub-set of initiation parameters, notably the four cryptographic keys  $K_{CID}$ ,  $K_{VID}$ ,  $K_{PCIN}$  and  $K_{RID}$  belonging to three of the ElectionAuthority domains.



**Figure 9:** The voting and publication mechanism of the revised e-voting protocol. Note that the mechanism for voting is exactly the same as the CESG e-voting system, except that the elector does not expect to receive back a *rid* value from the Gateway. Instead the elector may view the Publisher’s bulletin board in order to verify that the correct unique *rid* value is published. The elector is entitled to challenge the administration of the election prior to tallying if the *rid* is not published within some finite period of time, or if another *rid* value from their voting credentials appear on the board.

As indicated by the figure, rather than receiving a single set of credentials from one deliverer, as in the original protocol. The elector instead receives three different sets of credentials. These are the *vid* and **CNP** values from the Registration Officer. The *pcin* and half of each of the *rid* values from the PCINDeliver, and the remaining half of each of the *rid* values from the Electoral Commission.

An immediate concern of this design is that the two domains, PCINDeliver and Electoral Commission might collaborate in order to determine how an elector voted. An improvement upon this system would be to distribute the digits of the *rid* values amongst  $len_{RID}$  delivery domains. This approach would optimise the robustness of the delivery mechanism against collusion amongst delivery authorities, since all  $len_{RID}$  authorities would need to collude in order to determine for certain how the electors voted. Unfortunately, this mechanism would also require that the  $len_{RID}$  deliveries and the elector to re-compile the *rid* values on receipt, rather than recompiling the two parts of the *rid* values as presented in the system above. A consideration of the risks of collaboration across domains is made in section 4.1.

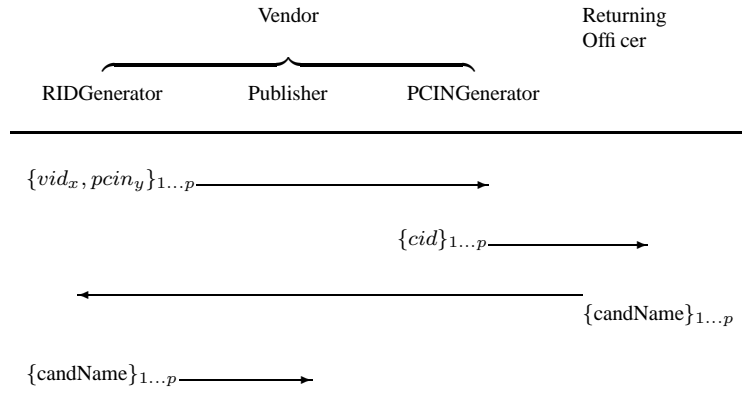
Rather than using a separate structure to collect the ballots sent via the gateways (Authenticator, Validator etc), the same structure that initiated the election may also be used to collect,

process and later, tally the ballots to produce a result. The re-use of these structures has the advantage of retaining the robustness of the initiation, without introducing any extra complexity. Figure 9 demonstrates the sequence of messages for ballot casting.

Despite the redesign, the voting mechanism for the elector is the same as the original CESC evoting protocol, requiring the elector to send their *vid* and a single *pcin* value of their choice from their voting credentials. The modified system, although now more robust, requires the voter to participate in the verification process in order to prevent the Vendor from casting blind ballots on their behalf. As stated before, it is expected that  $p$  electors of the total electorate will successfully cast a ballot and that a further subset  $r$  will in addition verify that their *rid* value has been correctly published on the bulletin board, such that  $r \leq p \leq n$ . Since the Vendor does not know which electors have verified their ballots, it is restricted to casting  $(n - p)$  ballots blindly at the end of the election with confidence that they will not be subject to verification by the electors.

Blind ballot casting would not (assuming perfect randomness of ballots) affect the actual outcome of simple election such as simple plurality, or first-past-the-post (FPTP), although the winning share of the total vote would be affected - arguably an important psychological factor. More complex elections, such as those conducted using single-transferable-vote (STV) or other more complex proportional systems would be more unpredictable under this flaw, since the large number of possible combinations on the ballots (an ordered subset of candidates) make perfect randomness a less accurate model.

The re-design of the CESC system has made use of the bulletin board construct, effectively a trusted store and publisher of the *rid* values of ballots as they are cast. Figure 7 indicates that the Publisher is placed within the Vendor domain, such that as  $\{vid, pcin\}$  combinations are received, the corresponding *rid* value may be generated and published for verification by the elector. A flaw with this scheme is the trusted nature of the bulletin-board construct. Although, the Vendor is described as having append only capabilities to the Publisher, possession within provides the Vendor with arbitrary access and capabilities. A malicious Vendor would be to



**Figure 10:** The tallying protocol conducted at the end of the revised e-voting protocol. Values are passed back through the domains that generated them in order to obtain the final candidateNames that correspond to *rid* values published on the bulletin board Publisher. The final tally may be computed from the publication on the board.

publish the correct *rid* value of a ballot, delay a short period of time before replacing this with another *rid* value.

To prevent this attack, independent external organisations may be employed to monitor the bulletin board throughout the election in order to enforce the append-only policy. For this purpose, the political parties, or the candidates themselves may be involved in the verification of the election. Each of the candidates would constantly monitor the Publisher and then re-publish the set of *rid* values as they are added. In the event of an *rid* value being removed from the Publisher by the Vendor, the candidates would be alerted to the discrepancy. For robustness, each of the candidates would monitor each other’s published *rid* values, such that cheating candidates who add *rid* values to their own lists are discovered. This approach models solutions to the Byzantine Generals problem, by which agreement is reached even if a restricted but unknown number of components are faulty [Lamport et al., 1982, Merritt, 1984].

## 4 Further Work

In this paper, the CESG e-voting system has been formalised and then revised in order to introduce useful properties for electronic voting applications. However, a difficulty of implementing

any such e-voting system has been that no one solution incorporates all the ideal features. Some of the limitations of the revised system are discussed below with a view to future revisions.

## 4.1 Robustness

Although the election system described is robust under the assumption of non-collusion across domains, most e-voting systems consider the possibility of multiple collaborations between independent authorities responsible for tabulating ballots [Benaloh and Yung, 1986]. Such schemes assume the collaboration of a maximum subset of authorities, with that maximum subset providing a security parameter for the protocol. Whilst the system above is believed to be verifiable even in the presence of sustained collusion, it is acknowledged that the anonymity of the system is reliant upon the non-collaboration across independent domains.

Table 1 illustrates the capabilities obtained by any two collaborating domains. An immediate threat to the revised system is the ability of the Vendor to commit blind ballot stuffing. Although such a capability has only a limited usefulness to the Vendor, since they little influence over the election, for a practical election, such a flaw is clearly unacceptable. Even more seriously, should the Vendor collaborate with the Returning Officer, then intelligent ballot stuffing is possible, whilst collaboration with the Registration Officer allows an elector's identity may be discovered.

One particular feature of the election system is the ability of the Registration Officer to collaborate with the vendor in order to trace the ballots of electors (in the event of an accusation of voting credential theft, for example). This feature is similar to UK practice with paper ballots, whereby the booklets of counter-foils, each labeled with an electoral roll and ballot serial number, are required in order to trace who cast a particular ballot. Whether such collaboration would be labeled an attack or feature, it would seem, depends upon the context of the collaboration taking place!

An ideal revision of the Election Authority design would remove these flaws by reducing

	Registration Offi cer	Vendor	$\{pcin\}$ Deliverer
Acting Returning Offi cer		Intelligent ballot box stuffi ng.	Intelligent ballot box stuffi ng.
$\{rid\}$ Deliverer		Link ballot to elector.	Link ballot to elector.
$\{pcin\}$ Deliverer	Inteligent ballot box stuffi ng. Link ballot to elector.	Link ballot to elector.	
Generator	Inteligent ballot box stuffi ng. Link ballot to elector.	Blind ballot box stuffi ng.	

**Table 1:** Malicious capabilities of 2-way collaborations of domains of the election authority. The table summarises the instances where collaboration allows a malicious attacker to (a) determine how an elector cast their ballot after the results of the election are announced, or (b) intelligently stuff the ballot box using non-voting elector’s credentials to cast a ballot for a known candidate. Note that collaborations capable of

the usefulness of collaboration to the individual domains, notably the Vendor. Such a re-design would need to retain the overall simplicity of the election system from the perspective of the elector, particularly the polsterless voting mechanism.

## 4.2 Receipt-freeness

A difficulty recognised during the re-design of the CESG system has been the incorporation of both verifiability and receipt-freeness [Benaloh and Tuinstra, 1994], or the stronger notion of coercion resistance. In general terms, receipt-freeness describes the inability of an elector to prove to an external party (such as a candidate) how they cast their ballot. The stronger notion of coercion resistance allows an elector to cast a ballot without the knowledge of an external attacker, in order to thwart attacks such as forced abstention [Juels and Jakobson, 2002].

Since the 1872 (Ballot) Act, elections in the UK have been conducted using *traceable* secret ballot papers [House of Commons, 1872]. This mechanism provides for a secret ballot under normal circumstances, but also allows personated ballot papers to be removed from the count in the event of an election petition. The recent change to postal voting on demand has, however, demonstrated the difficulty of providing receipt-freeness in remote elections, regardless of medium. For example, the elector may make a photo-copy of their ballot paper prior to submission, in order to receive some reward for the vote from a particular candidate.

No instance of such activity has yet been reported in a UK election and it would appear that whilst the possibility of such electoral fraud does not become more serious, the government intends to focus upon increasing participation through more convenient voting mechanisms [Office of the Deputy Prime Minister, 2003]. Clearly, the design of the election system provided above has focused upon verifiability rather than receipt-freeness. However, the threat of a return to coercion as a common electoral practice should not be ignored. Whilst it is likely that vote-buying (i.e. the willing participation of the elector in an attack prior to voting) cannot be prevented through technological means for remote voting, the presence of a permanent record

(the bulletin board) of how an elector cast their ballot may be unacceptable. An improvement upon the system proposed above would perhaps restrict the range of observers to whom the *rid* value can be verified to or by. For example, an external attacker would be excluded from the set of participants who can use an elector's credentials to determine how their vote was cast on the bulletin board after the election.

## 5 Conclusions

The revised election system proposed above demonstrates the worth of polsterless electronic voting in terms of the verifiability of the resulting tally from the perspective of a non-technical user. This facility builds upon the advantage of using non-cryptographically capable devices for facilitating electronic voting, as proposed by the original CESG study. Further, the re-design demonstrates how the architecture employed for collecting ballots may be made more robust against such practices as ballot stuffing (for example by the vendor) and also how the anonymity of the ballot casting process may also be protected.

The range of criteria for a satisfactory electronic voting scheme in the UK makes the design of such systems extremely complex. Critics of electronic voting have suggested that such complexity and range of requirements make the electronic medium entirely unsuited to the conduct of elections, particularly when they are remote [Mercuri, 2001]. Whilst it is evident that no completely satisfactory scheme yet exists for the UK, the redesign above suggests that the combination of existing constructs and requirements may eventually result in a solution. The possible advantages of convenience, and indeed greater accuracy of electronic voting schemes should not be dismissed because of the initial complexity of the problem.

## References

- [Bederson et al., 2003] Bederson, B. B., Lee, B., Sherman, R. M., Herrnson, P. S., and Niemi, R. G. (2003). Electronic voting system usability issues. In *Proceedings of the conference on Human factors in computing systems*, pages 145–152, Ft. Lauderdale, Florida, USA. ACM Press.
- [Benaloh and Tuinstra, 1994] Benaloh, J. and Tuinstra, D. (1994). Receipt-free secret-ballot elections (extended abstract). In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 544–553. ACM Press.
- [Benaloh and Yung, 1986] Benaloh, J. and Yung, M. (1986). Distributing the power of a government to enhance the privacy of voters. In *5th ACM Symposium on Principles of Distributed Computing (PODC '86)*, pages 52–62.
- [CESG, 2002a] CESG (2002a). Communications and electronic security group. e-voting security study. <http://www.edemocracy.gov.uk/library/papers/study.pdf>.
- [CESG, 2002b] CESG (2002b). e-voting technical and security requirements. e-voting technical and security requirements. <http://www.edemocracy.gov.uk/library/papers/evoting.pdf>.
- [Chaum, 2001] Chaum, D. (2001). Surevote technical overview(slides). <http://www.vote.caltech.edu/wote01/pdfs/surevote.pdf>.
- [Communications, 2002] Communications, C. (2002). Report to the office of the envoy: e-democracy report of research findings. [http://www.edemocracy.gov.uk/downloads/Full\\_Report.pdf](http://www.edemocracy.gov.uk/downloads/Full_Report.pdf).
- [Excelsior Consultancy, 2002] Excelsior Consultancy (2002). Security warning for e-democracy.

- [Fairweather, 2002] Fairweather, D. N. B. (2002). CESG report on evoting security - response of the centre for computing and social responsibility. <http://www.ccsr.cms.dmu.ac.uk/resources/general/responses/ppera2000.print.html>.
- [House of Commons, 1872] House of Commons (1872). Parliamentary and Municipal Elections (Ballot) Act, 1872 Ch. 33. HM Stationary Office.
- [House of Commons, 2000] House of Commons (2000). Representation of the People Act, 2000 Ch. 2. HM Stationary Office.
- [Jackson and Syddique, 1991] Jackson, P. and Syddique, E. (1991). Ballot secrecy. Electoral Reform Society, 6, Chancel Street, Blackfriars, London SE1 0UU.
- [Juels and Jakobson, 2002] Juels, A. and Jakobson, M. (2002). Coercion resistant electronic elections. [eprint.iacr.org/2002/165.pdf](http://eprint.iacr.org/2002/165.pdf).
- [Kitcat, 2002] Kitcat, J. (2002). e-voting security study response: FREE e-democracy project.
- [Lamport et al., 1982] Lamport, L., Shostack, R., and Pease, M. (1982). The byzantine generals' problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401.
- [Malkhi et al., 2002] Malkhi, D., Margo, O., and Pavlov, E. (2002). E-voting without 'cryptography'.
- [Mercuri, 2001] Mercuri, R. (2001). *Electronic Vote Tabulation: Checks and Balances*. PhD thesis, University of Pennsylvania.
- [Mercuri, 2002] Mercuri, R. (2002). Response to formal request for comment by the CESG (UK) on internet voting.
- [Merritt, 1984] Merritt, M. (1984). Elections in the presence of faults. In *Proceedings of the third annual ACM symposium on Principles of distributed computing*, pages 134–142.

- [MORI, 2003] MORI (2003). Market and opinion research international technology tracker.  
<http://www.mori.com/emori/tracker.shtml>.
- [Office of the Deputy Prime Minister, 2003] Office of the Deputy Prime Minister (2003). Electoral pilots at the C.E.P. and local elections - consultation paper.  
[http://www.odpm.gov.uk/stellent/groups/odpm/\\_localgov/documents/page/odpm\\_lo%cgov\\_024004.hcsp](http://www.odpm.gov.uk/stellent/groups/odpm/_localgov/documents/page/odpm_lo%cgov_024004.hcsp).
- [OFTEL, 2003] OFTEL (2003). Office of telecommunications. consumers' use of mobile phone telephony. Oftel, <http://www.oftel.gov.uk/publications/research/2003/q11mobr0103.pdf>, 50 Ludgate Hill, London, EC4M 7JJ.
- [Pratchett, 2002] Pratchett, L. (2002). The implementation of electronic voting in the uk. De Montfort University.
- [Rjašková, 2003] Rjašková, Z. (2003). Electronic voting schemes. Master's thesis, Comenius University, Bratislava.
- [Sako and Kilian, 1995] Sako, K. and Kilian, J. (1995). Receipt free mix-type voting scheme - a solution to the implementation of a voting booth. In Guillou, L. and Quisquater, J.-J., editors, *EUROCRYPT '95*, pages 393–403. Springer Verlag. LNCS no. 921.
- [The Electoral Commission, 2001] The Electoral Commission (2001). Election 2001: the official results.
- [The Electoral Commission, 2002] The Electoral Commission (2002). Modernising elections, a strategic evaluation of the 2002 electoral pilot schemes.
- [The Electoral Commission, 2003] The Electoral Commission (2003). The shape of elections to come: A strategic evaluation of the 2003 electoral pilot schemes.