Elections with Unconditionally-Secret Ballots and Disruption Equivalent to Breaking RSA

David Chaum
Centre for Mathematics and Computer Science
Kruislaan 413 1098 SJ Amsterdam

Introduction

An election protocol is presented that has the following properties:

- A voter's privacy can be violated only by cooperation of all other voters.
- Voters can ensure that their ballots can be counted.
- Voters wishing to disrupt an election can cause only a limited delay before being disenfranchised, unless RSA is broken.

It is assumed, for simplicity, that a single organization z is empowered to decide who can register and that z acts faithfully to complete elections. (This assumption is relaxed somewhat in the final section.) Nevertheless, even if z were endowed with infinite computational power, z could not learn who votes which way or falsely convince voters that their votes are counted.

The remaining sections may be summarized as follows: (1) previous work on voting protocols and some related protocols underlying the present proposal are surveyed; (2) the ballot issuing protocol and its properties are presented separately, being the heart of the present contribution; (3) the model and overall voting protocol are presented based on the ballot issuing protocol; (4) some simple ways to apply the techniques to payment and credential systems are mentioned; and (5) the assumptions and several further points related to the protocols are discussed.

1. Relation to Previous Work

The first multi-party secure election protocol in the literature [Chaum 81] could not prevent someone able to break RSA from tracing ballots back to particular voters, although some properties about it could be proved under reasonable assumptions [Merritt 83]. A subsequent proposal did not at all protect the confidentiality of ballots from those conducting elections [Cohen & Fischer 85]. An extension [Cohen 86], similar in nature to

C.G. Guenther (Ed.): Advances in Cryptology - EUROCRYPT '88, LNCS 330, pp. 177-182, 1988.

the original [Chaum 81] proposal, divides the "government" into parts, in such a way that all parts must cooperate to violate participants' privacy. Using such a protocol to obtain the optimal privacy protection obtained here, however, would allow any single participant to disrupt the entire election. Also, it has security against cheating that is only linear in the effort required of each participant, in contrast to the exponential security proved here.

The present work draws on two previous basic results. One is a "sender untraceability" system detailed in [Chaum 88b]. It provides unconditional security against tracing the senders of messages and limits the disruption that can be caused by participants. The second is the notion of "blind signatures," which serves as a basis for untraceable payments and credentials, as introduced in [Chaum 85] and detailed in [Chaum 88c] and [Chaum & Evertse 87].

2. Ballot Issuing Protocol

The protocol defined in this section in essence allows an applicant y to give very high certainty to z that the ballot provided by y is of a form that allows y only to cast a single vote.

Consider the following protocol between an applicant y and organization z:

(1) Once, and for all applicants, z broadcasts: a small integer security parameter s; a second integer parameter n; an RSA modulus N; a prime $d>N$; and n distinct random units of the ring of residue classes modulo N (called units modulo N for short), denoted v_j, where $j \in \{1, ..., n\}$ throughout. (In this protocol "random" is used to mean uniformly distributed and independent of everything else.)

(2) $y \rightarrow z$: (read "y sends to z") $M=(m_{i,j})$, $m_{i,j} \equiv v_{\pi_i(j)} r_{i,j}^d \pmod N$, where $i \in \{1, ..., s\}$, with π_i random permutations of $\{1, ..., n\}$, and with $r_{i,j}$ random units modulo N.

(3) $z \rightarrow y$: C, a random nonempty proper subset of $\{1, ..., s\}$.

(4) $y \rightarrow z$: $k \in \{1, ..., s\} - C$; $P=(p_{i,j})$, $p_{i,j}=\pi_i(j)$, for $i \in C$; $p_{i,j}=\pi_k^{-1}(\pi_i(j))$, for $i \notin C$; $Q=(q_{i,j})$, $q_{i,j} \equiv r_{i,j} \pmod N$, for $i \in C$; and $q_{i,j} \equiv r_{k,\pi_k^{-1}(\pi_i(j))} r_{i,j}^{-1} \pmod N$, for $i \notin C$.

(5) z verifies that every row of P is a permutation of $\{1, ..., n\}$; that $m_{i,j} \equiv v_{p_{i,j}} q_{i,j}^d \pmod N$, for $i \in C$; and that $q_{i,j}^d \equiv m_{k,p_{i,j}} m_{i,j}^{-1} \pmod N$, for $i \notin C$.

Theorem: *For* y *following the protocol,* π_k *is statistically independent of the messages transmitted.*

Proof: (sketch) Without loss of generality, fix k. The tuple (P, Q, M) defines the messages transmitted in an instance of the protocol, and A denotes the set of all possible such tuples. Similarly, B is the set of all possible tuples $(\pi_l, r_{i,j})$ with $l \neq k$, $1 \leqslant i \leqslant s$ and

$1 \leqslant j \leqslant n$. It follows easily from the protocol that each π_k defines a one-to-one correspondence between A and B. Moreover, by the mutual independence and uniformity of all the π_i and $r_{i,j}$, the conditional probability distribution of B given π_k is uniform for each instance of the protocol. Therefore the conditional probability distribution of A given π_k is always uniform and hence independent of π_k.□

Theorem: *Assuming* y *cannot form* d*th roots of random units modulo* N, *then when* z *reveals* d*th roots modulo* N *of* h *distinct* $\mathrm{m}_{\mathrm{k,j}}$, *with* k *fixed and* $1 \leqslant \mathrm{j} \leqslant \mathrm{n}$, *the probability of allowing* y *to learn* d*th roots of other than exactly* h *of the* v_i *does not exceed* $1/(2^s-2)$.

Proof: (Sketch) It is sufficient to show that, with probability $\geqslant 1-1/(2^s-2)$, there exists exactly one permutation π such that for each j, $1 \leqslant j \leqslant n$, y knows an r_j such that $m_{k,j} = v_{\pi(j)} r_j^d$. With probability $\geqslant 1/(2^s-2)$ there exists at least one permutation π' such that y can express each entry $m_{k,j}$ as $m_{k,j} \equiv v_{\pi'(j)} r'^d_j \pmod N$, since otherwise only one C allows y to succeed. (Notice that for y to successfully cheat, the $m_{i,j}$'s must be properly constructed for each $i \in C$ and improperly constructed for each $i \notin C$. But this implies that only one C allows y to cheat.) It remains to be shown that there cannot be two permutations π' and π'' such that y knows $r'_{k,j}$ and $r''_{k,j}$, with $m_{k,j} \equiv v_{\pi'(j)} r'^d_{k,j} \equiv v_{\pi''(j)} r''^d_{k,j} (\mathrm{mod}\ N)$ for $j \in \{1, ..., n\}$. If there were two such permutations, then y would have been able to learn the dth root of a quotient $v_{\pi'(j)} v^{-1}_{\pi''(j)}$ for some j with $\pi'(j) \neq \pi''(j)$. But it is easy to see that the ability to compute roots on random quotients is polynomial time reducible to the ability to compute roots on random units.□

3. Overall Voting Protocol

Elections are in three phases:

Preliminary: In the preliminary phase, z broadcasts those things mentioned in the first step of the ballot issuing protocol above. This is done only once for the entire election. Additionally, z broadcasts an assignment of an outcome to each v_i, thus partitioning the v_i into fixed, disjoint equivalence classes, such that each class corresponds with a distinct outcome. For example, assuming the election allows each voter to cast a single vote (as is assumed throughout) for at most one of two candidates, then the v_i are partitioned into two outcome classes, one for each candidate.

Registration: During the registration phase, each applicant communicates with z. If z agrees to allow a particular applicant to register, then the applicant and z conduct an instance of the ballot issuing protocol of the previous section. The result of this is a tuple of n elements, $m_{k,i}$, one element of which is selected by the applicant. This selected element is denoted b_l for the lth registered voter. (It is now assumed that $n \gg m$). The final result of the registration phase, which is broadcast by z, is the set of b_l, for $1 \leqslant l \leqslant m$, where m is the number of registered voters. It will still be possible for disputes regarding the b's to be resolved at this point without revealing anything about the votes.

Voting: The voting phase is begun by z broadcasting the dth roots of all of the b_l. (Naturally, if this is not carried out properly, everyone will know.) Then, the lth voter recovers the dth root on a v_i, simply by dividing the dth root of b_l by the corresponding $r_{h,j}$. Each voter then broadcasts, under the sender untraceability protocol mentioned above, the root of the single v_i recovered. Finally, each voter can verify that the root of the v_i sent by that voter was in fact available from the broadcast channel. The number of votes for a particular outcome is just the number of distinct dth roots of v_i's corresponding to that outcome.

4. Payments and Credentials

The election protocol can be used to directly realize untraceable payments: each v_i stands for, say, one dollar; registration is withdrawal from a bank account; payment is made by providing a shop with a dth rood on a v_i that has not yet been accepted for deposit by the bank.

A variation on the election protocol can also be used to implement a "credential mechanism" [Chaum 85 and Chaum & Evertse 87]. The v_i serve as unique personal identifiers, one selected by each individual. Let d_i be distinct primes, with $d_k|d$ and $(d_k,\phi(N))=1$, for suitably many k's. Each individual participates in an instance of the election protocol with each organization, using a d_k unique to that organization. (See [Shamir 83] for why such use of the d_i is secure.) If not all m votes are cast in any organization's "election," at least one participant is cheating. In this case, people reveal all their $r_{k,i}$ and π_k, and those who are unable to show that their b_l corresponds to a v_i that was broadcast are revealed as cheaters and excluded from the protocol. This is repeated with different v_i until no cheating is detected.

The remaining unused k's each correspond to a type of credential. An organization issues the kth credential to a person by providing the d_kth root of the person's selected element, b_l; then and only then can the d_kth root of the person's selected element with any other organization be shown.

5. Discussion

It has been assumed that n was large enough to make the possibility of the same v_i being chosen accidentally by two voters acceptably small. This might require something like $n=100m^2$, which might be impractical for large m. Another approach allows $n=m$. It is based on the idea that voters will be able to reserve v_i's anonymously. One way to do this by is using the "slot reservation" protocol of [Chaum 84a], which has been improved by [den Boer 87]. A simple variation allows reservations to be made and confirmed one at a time, using any sender untraceability system. (Reducing from $2m$ to m could be accomplished by elections using one d_k for each type of vote.)

If less than m disjoint roots of v_i are broadcast, z could form and broadcast extra votes. Thus people who register and do not vote, in effect, allow z to steal their vote. Someone might entrap z, however, by allowing a vote to be stolen and latter broadcasting the real (different) vote, possibly untraceably.

The essential requirements of the communication channel are that z must not be able to provide inconsistent or incomplete messages to different voters, and that voters must be able to broadcast the messages required to untraceably submit votes. The first property could be achieved in some cases simply by z making digital signatures on all messages including some kind of hash or (even all previous messages) and a time stamp, since if inconsistent messages become known, z would be incriminated.

The requirement that d be prime and $>N$ ensures that $(d,\phi(N))=1$. To get certainty that a small d has this property seems difficult in general. It is easy, however, to modify the protocol presented to give exponential certainty that $(d,\phi(N))=1$ using the idea that y and z can "flip coins by telephone" [Blum 82] to develop t mutually trusted random units, after which z is required to reveal their dth roots. The probability that z can cheat is then $<2^{-t}$, assuming that z cannot cheat during the coin flipping. This can be ensured if, for example, z provides the modulus used in coin flipping and is then required to reveal its factorization afterwards.

A natural extension is to divide among several entities various functions of z, such as: creating the random v_i's; making the registration (withdrawal) decision; and signing the b_i's.

Summary and Conclusion

Election protocols embodying robustness, verifiability of returns by voters, and unconditional security for voters' privacy have been presented. The techniques also allow untraceable payments and credentials.

References

(1) Blum, M., "Coin flipping by telephone," *Proceedings of IEEE Compcon*, 1982, pp. 133-137.

(2) Boer, B. den, private communication.

(3) Chaum, D., "Untraceable electronic mail, return addresses and digital pseudonyms," *Comm. ACM* 24, 2 (February 1981), pp. 84-88.

(4) Chaum, D., "Security without identification: transaction systems to make big brother obsolete," *Comm. ACM* 28, 10 (October 1985), pp. 1030-1044.

(5) Chaum, D., Evertse, J.-H., "A secure and privacy-protecting protocol for transmitting personal information between organizations," *Advances in Cryptology: Proceedings of CRYPTO 86,* A.M. Odlyzko, Ed., Springer-Verlag, pp. 118-167, 1987.

(6) Chaum, D., "Blinding for unanticipated signatures," *Advances in Cryptology: Proceedings of Eurocrypt 87,* D. Chaum and W.L. Price, Eds., Springer-Verlag, pp. 227-233, 1988a.

(7) Chaum, D., "The dining cryptographers problem: unconditional sender and recipient untraceability," *Journal of Cryptology*, Vol. 1 No. 1, pp. 65-75, 1988b.

(8) Chaum, D., "Privacy protected payments: unconditional payer and / or payee untraceability," to appear in *Smart Card 2000,* North-Holland, 1988c.

(9) Cohen, J. and Fischer, M., "A robust and verifiable cryptographically secure election scheme," Proceedings 26th FOCS, 1985, pp. 372-382.

(10) Cohen, J.D., "Improving Privacy in Cryptograhpic Elections," Yale University Computer Science Department Technical Report YALEU / DCS / TR-454, February 1986.

(11) Merritt, M., *Cryptographic Protocols*, Ph.D. Thesis, Georgia Institute of Technology, GIT-ICS-83 / 06, 1983.

(12) Shamir, A., "On the generation of cryptographically strong pseudorandom sequences," *ACM Transactions on Computer Systems*, Vol. 1 No. 1, pp. 37-44, February 1983.