ELECTRONIC VOTING SYSTEMS

MAUSAM & AMOL PRAKASH

December 1, 1999

Abstract

Technological advancements in the whole field of cryptography have its effects in nearly every realm of human life. Be it be transfer of cash or real buying and selling of objects or just anything, cryptographic techniques are expanding itself everywhere. A very interesting expansion of cryptography lies in the field of elections. Elections are of various types, they may be elections for finding the "Man of the Millennium" on the net done by something like the Time magazine, or electing a new class convener for the 1997 batch of computer science, IIT Delhi or of course the election of the President, U S A. The electorate size in all these elections varies, the rules vary, probable the election is just a yes/no or voting for one's preferences or what not. But still, there are certain common features that all these elections share and so based on those, attempts have been made to automate the election procedure or at least computerize it. This brings forward a lot of cryptographic issues because most elections in real life are non transferable and secret ballots. This study deals with these cryptographic issues and attempts to find various known protocols trying to achieve different features in the election procedures.

0.1 Elections: Common Features

0.1.1 Election: Step By Step

The following are the various tasks one needs to fulfill in any general election:

Registration: This task involves compiling a list of people eligible for vote. This may be done manually if all the voters are easily accessible or various other techniques may be used if the voters cannot be accessed that easily. It may be required that a certain identifying feature for the voter can be decided upon so that he/she can be identified at the time of electronic voting.

Validation: This task of course involves checking the eligibility of those attempting to vote. For this stage only, an identifying feature is required eg. the public key.

Collection: This task involves collecting all the voted ballots.

Tallying: This task involves counting the votes.

0.1.2 Election: Necessary Characteristics of a protocol

If one goes for any election, the most important thing in it is the people's faith that the elections have been done without any sort of fraud or mischief by either the authorities or any of the eligible voters or by ineligible people. This faith is restored in the people by having certain desired characteristics in any protocol. So let us go through those characteristics:

Accuracy: Any election system is called accurate if

- if a vote casted cannot be altered in any way.
- if a valid is always counted correctly in the final tally.
- if no invalid vote is counted.

Democracy: Any election system is called democratic if

- if only eligible voters vote.
- if every eligible voter can vote only once.

Privacy: An election system ensures privacy if

- the correspondence between any vote and its voter can never be found out either by the authorities or by any body else.

- no voter can prove that he/she voted in a particular way.There is a disagreement in this because a lot of people say it is non democratic to not allow a voter to prove his vote. But this feature, if present will take care of those cases in which a person is forced to vote in a particular way. Although, it may be interesting to note that this feature can not be completely satisfied because a voter may cast his vote in somebody else's presence so it is trivially proven to the other person the way he/she voted.

Verifiability: This is an interesting feature which is usually not found in the physical voting schemes. A system is verifiable if every voter can independently verify that his vote has been counted and counted correctly.

0.1.3 Elections: Some desirable but not necessary features

Let us look at certain other characteristics of a protocol which if present add to the value of the protocol.

Convenience: Any election system is convenient if it allows voters to cast their votes quickly, in one session, and with minimal equipment and special skills.

Flexibility: Any election system is flexible if it allows a variety of question formats including open ended questions (which may be a requirement for surveys).

Mobility: Any election system is mobile if there are no restrictions (other than logistic ones) on the location from which a voter can cast his vote.

Robustness: This ensures that the system can recover from the faulty behaviour of any (reasonably sized) coalition of parties.

Voter Duplication: In this one voter can copy the vote of another person without knowing the actual vote.

0.2 Growth of schemes for electronic voting

Over the period there has been a usual transition from real simplistic voting protocols to quite complex ones. And more or less necessarily they are becoming safer and safer. To the extent, that now evoting is no more a dream, it has been implemented. Although there is still some time that these elections will be possible for President, USA but at least it is possible to elect the convener of IIT Delhi CS batch. So let us go a little back in history and see the major milestones from a really simplistic mechanism to a fairly workable solution.

0.2.1 The start: A Simple Scheme

A fairly basic mechanism to vote will be as follows:

- The voter submits to the electronic validator, an electronic ballot and his voter identification number (I.D.).
- The validator checks that voter is eligible, has not yet casted his vote. If found valid, the validator would strip off the I.D. and send the ballot to the electronic tallier.
- The tallier records the votes and adds them to the electronic tally.

As can be readily verified that this scheme is flexible, mobile and convenient but surely this scheme does not possess most of the necessary conditions.

Loopholes:

1. A voter may vote for another voter using the other voter's I.D. (non-accurate)
2. If the validator wants he can see the vote. (non private)
3. Validator may change the vote. (non accurate)
4. The tallier may count incorrectly as there is no check on her. (non verifiable)

Some of the solutions for loopholes seem nearly obvious.Like for point 1,the voter I.D. may be suitably replaced with voter's digital signature and scheme becomes as secure as the discrete log problem.Moreover, we can prevent the validator from violating the voter's privacy by having voters encrypt their ballots with the tallier's public key. This idea of encryption is quite strong in the sense that all the transactions are done in this form only. But in this case, it does not seem to fit very well as the validator and tallier may collude to bust the privacy of the voter.

0.2.2 Two agency protocol : Nurmi, Salomaa, Santean (1991)

A major milestone in the history of electronic voting came when many of the above loopholes were accounted for by Nurmi, Salomaa and Santean. They proposed One agency and Two agency protocols. Let us discuss the Two Agency Protocol in detail:

- Electronic validator distributes secret identification tags to each voter just prior to elections.
- The validator then sends the list of all the identification tags, with the corresponding records of the voters.

- The voter sends the tallier his/her tag and an encrypted file containing the vote.
- Tallier after examining the validity of tag publishes the encrypted file that can be verified by the voter for its presence.
- The voter after checking for the file can send the key necessary to decrypt the vote.
- After the elections are over tallier publishes the list of votes corresponding to each encrypted file.

There are certain very important characteristics that this protocol possesses. Firstly, every voter can independently verify whether his vote was counted or not.Moreover if it was not, he can protest to the election commissioner on the pretext that it was earlier published by the validator. So, overall all the four mentioned loopholes of the simple scheme are rectified. But this still has certain loopholes that have not been taken care of.

Loopholes:

1. The tallier may cheat. He may cast votes for people who did not cast their votes. This he can do toward the end of the voting when the voting is more or less over. Interestingly, even if voters discover this violation, they cannot prove that they did not actually vote.
2. The validator and tallier can still collude and the voter's privacy may be busted.
3. The voter can prove it to any body that he/she voted in a particular way by showing the vote and the encryption key.

0.2.3 The Fujioka, Okamoto and Ohta Protocol (1993)

Most of the main problems of evoting got solved when this protocol was proposed.It has the following course of events:

- Voter prepares a voted ballot, encrypts it with a secret key, signs it and sends it to validator.
- Validator verifies the signature, signs the blinded ballot and sends it back to the voter.
- Voter removes the blinding encryption layer and sends it to tallier.
- Tallier verifies the signature of the validator, and places it on the list that is published after all voters vote.
- When list is published, the voter verifies his/her ballot on the list (i.e. the ballot is included) and sends the tallier the decryption key.

- Tallier decrypts the ballots and adds the vote to the tally. He also publishes the decryption keys along with encrypted ballots.

This idea removed most of the obvious loopholes and served as the base to any further advancement and actual implementation.

In the following section we will take up the various other advances and the attempts to improve upon this protocol. It will also talk about some possible loopholes of this protocol.

0.3 Advances in the FOO Protocol

Based on the FOO protocol a lot of people have independently tried to focus on different issues and suggested strategies. Some of them are as follows:

0.3.1 SENSUS: (Cranor and 2Cytron)

This protocol was proposed by Cranor and Cytron in 1996. The major important thing about it is that it was the first voting scheme to have actually been implemented. So it takes care of various implementational minute details also. It uses a pollster program which is authorised to vote on behalf of a voter. So let us first quickly look at the basic differences of SENSUS from the FOO protocol.

- SENSUS employs a pollster program that performs all cryptographic functions and transactions with the election programs on the voter's behalf. This is done because it was felt that in FOO, a voter has to perform a lot of operations of high specialisation, so automating it will increase convenience of the protocol.
- Rather than the tallier printing out all the encrypted votes, in SENSUS, the tallier responds to the encrypted ballot submitted by the voter by sending a receipt to the voter. The voter can now send the decryption key immediately. So the election process is complete in one session increasing the convenience of the system.

Since it was the protocol which was implemented let us quickly look at the exact implementation so that finer details of the protocol can be studied and understood. It uses the standard public key RSA type encryption scheme for its message transfers and verification is done by checking usual RSA signatures.

The notations are as follows:

- P: pollster
- Vd: validator
- T: tallier
- ve,vd,vn: validator's key pairs and modulus.

- RVL: list of registered voters with corresponding voter ID's public keys, and indication as to whether or not they have voted till now.
- V: voted ballot
- ID: P's ID number.
- se,sd: ballot seal key pair
- m=V^{se},digested.
- k: large random number
- ie,id,in: voter's key pair and modulus.
- te,td,tn: tallier's key pair and modulus.
- b=m*k^{ve}(mod vn): blinded ballot digest.
- RL: list of recipients issued with corresponding sealed ballots, decryption keys and receipt numbers.

The protocol is as follows:

- P sends $E_{ve}(b, ID, b^{id})$ to Vd.
- Vd decrypts with vd.
- Vd verifies that $(b^{id})^{ie} = b$.
- If RVL indicates validation not yet issued Vd updates RVL and signs blinded digest b^{vd}.
- Vd sends $E_{vd}(b^{vd})$ to P.
- P decrypts with id.
- P unblinds b^{vd} by computing $m^{vd} = b^{vd}/k(mod\, vn)$.
- P verifies that $(m^{vd})^{ve} = m$.
- P sends $E_{te}(m^{vd}, V^{se})$ to T.
- T decrypts with td.
- T verifies that $(m^{vd})^{ve} = V^{se}$.
- T signs V^{se} as $(V^{se})^{td}$.
- T updates RL and assigns a receipt no.
- T sends $(V^{se})^{td}$ and receipt no. to P.
- P verifies that $((V^{se})^{td})^{te}$ =V^{se}.
- P sends receipt number and sd to T.
- T decrypts V^{se} with sd.
- T updates RL and tally.

0.3.2 Anonymous Remailing

One of the major drawbacks of SENSUS are that the final dialogue between the Pollster and the tallier does not grant privacy. When voter sends his key for decryption purposes, it will not be that difficult to trace the message. One of the suggested solutions is the the scheme of Anonymous remailing. In this all communications are done through a chain of remailers, the more in number the more difficult, it is to bust the privacy.

Problems: There are two major issues associated with it. Firstly, to prevent traffic analysis,very smart remailers would have to be designed which would therefore be of multiple layers.The problem id that it will consume a lot of cryptographic computations. But the major issue is that it is hard to develop practically an architecture of anonymous remailers that grants security and fault tolerance. Lets say one makes a chain of large number of anonymous remailers and one of them crashes or gets corrupted then the entire chain should be removed.

0.3.3 MULTI-AUTHORITY SECRET-BALLOT ELECTIONS : Cramer, Franklin, Schoenmakers, Yung (1996)

As one of the possible solutions to aforementioned problem comes this multi-authority protocol. The intuition is like this that instead of using blinding signatures to hide vote the idea should be to split vote into shares and send theses shares to different tallying authorities. Each tallier now has only a piece of vote so they can't get to know the complete result.Their partial results may be added finally to give the final results. The only major problem with this scheme is to design a ballot paper which supports this "split and merge" process.

This protocol is highly efficient as it reduces the work required by the voter or an authority to a linear number of cryptographic operations in a population size. Also this protocol has extensions in batched elections, where the same ballot is used several times. So this way parallel elections and multiway elections (selecting from large number of choices) can be conducted.

Some details:

Here the parties are modelled are polynomial time processes. They can communicate through:

1. Bulletin Board : This is read by all parties. Each party controls their section of the bill board, where they can only post messages but these messages cannot be erased or overwritten.
2. Private Channels : These support private communication between voters and authorities. Any public-key encryption may be used, using bulletin board to post corresponding encryptions.

In this, each voter constructs a ballot as an encryption of the desired vote and then post it at at the bulletin board. Here a proof of validity is also required that convinces all parties that it is a valid vote without revealing it. But, the authorities are able to decrypt it. This situation can be alleviated by having multiple authorities instead of one. The encrypted vote is distributed over the authorities such that fewer than some number of authorities remain ignorant about the vote. In the end, the final tally is published together with some auxiliary information to enable universal verifiability. Thus any party can accumulate the encrypted votes and check the final tally, by holding it against this accumulation and the auxiliary information. Here a homomorphic encryption algorithm based on discrete logarithm problem is used. Also the proof of validity is done through a more efficient 3-move protocol instead of zero-knowledge proofs. All this makes it highly efficient and in net linear to the size of population as far as the effort on the part of the voter is concerned.

0.3.4 Supporting a variety of ballot papers

In certain elections there are different types of ballot papers and each voter is eligible to vote in only some of them. The authority should take care that no voter votes on ballot papers not meant for him. This can be achieved in the following way:

- At the time of validation, the validator sends the right ballot paper signed by him.So that the voter has to use only that ballot paper which has been signed by the validator so he uses the ballot paper for only the election, he is eligible for.
- But now, two voters from different ballot papers classes may collaborate. One solution to this is to have a check on each ballot separately such that no two exactly signed ballot papers be counted twice but it is not an efficient implementation.
- So rather a better solution is to have the ballot paper locally available on the net itself but different keys should be used while signing the encrypted vote by the validator. Thus the signature of validator on the final vote discloses to the tallier the voters's eligibility class.

0.3.5 Multiple talliers

Vote dropping is another possibility that is a tallier may refuse to count a vote by modifying the signature data before publishing the list. Only the voter can verify it.If the voter checks, he/she would file a complaint and this would thus bring out the possible uncertainty in the election system and may reduce the faith in this process. Moreover it has always been stressed that the elections be over in a single session.

An obvious solution to this is to have multiple talliers who count independently (probably from different interest groups) who count the votes independently and then the tallies are compared. This would reduce the chances of miscalculation unless all the talliers decide to collude, the safety of which would increase as the number increases still it is not a fool proof method.May be if not all, but large number of them collude then, it might also happen that the really fair tallier is found cheating!

0.3.6 Multiple Administrator Protocol : Brandon William DuRette (1999)

Even after all the above advancements, one basic loophole still remains unplugged.That is, that the validator himself may cast votes towards the end of the election on behalf of those who have not cast their votes.

What happens if the some of the voters who come to vote in the very end. The validator can be smart enough to do any one of the following

- claim that the election is closed.
- respond "server is too busy to accept any connections".
- keep a safety factor vote for the late comers.

Although the errors are not undetectable . Each voter can independently check their votes. But, expecting people who are unwilling to vote in the original election to check that their vote was not cast is unreasonable. Vote forgery must be prevented without additional burden to the voters and no burden to to the non-voters.

To prevent a single validator from forging votes we simply require more than one validator to verify the votes. Thus, no single administrator has the power to cast a vote on somebody's behalf. Only if many of the administrators collude is this possible again they should be best chosen from different interest groups.

Some details:

There are two basic methods to enforce multiple validators. The most straight forward is to require each validator to sign all votes independently.To be valid a vote must contain several signatures. Another method would be to use a threshold signature protocol. In such protocols, a single signature is produced using the keys of several validators. While threshold signatures schemes have theoretical elegance, for this type of problem implementation of such a protocol was not chosen due to complexity of implementation.

Another interesting discussion involves the new level of complexity created by multiple validator implementation. Let us see the following insecure protocols:

- Simple Protocol: It is a straight forward implementation of the signature scheme. The scheme comprises of n validators of which t signatures are required for $t<=n$. In this scheme, double voting is achievable if $t<=n/2$. If we do not want $t>n/2$ for scalability reasons or any other reasons, then some following protocols have been suggested none of which proves secure.

- Communicating validators: The voters might be prevented from double voting if the validators could communicate what vote they have signed. Validators would agree to only sign votes that contain the same data as votes signed by the first validator to sign a particular voter's vote.The fallacy in this method is that there is no way for a validator to know what the vote is that is being signed. This information is blinded from the validator.The blinding for each validator produces different data, so even votes which contain the same data appear different to the validators.

- Managed validators: It involves addition of another type of server, the manager, whose signature was necessarily involved to cast a vote. Theoretically the manager would sign only once per voter hereby preventing the voter from double voting. But it does not stop the manager himself from colluding with the voters and casting double votes.

No protocol has been developed so far which seems fool proof as far as multiple validator ($t<=n/2$) protocols are concerned.

But this development is still a remarkable step towards actualization of evoting. This protocol (managed validator) has been developed at MIT under the name of EVOX.

0.4 On-line voting

Till now we had been studying various off-line voting schemes. Here various communications eg. sending the tags to the voter, collection of votes etc. is done through e-mail. The actual voting is done off-line.

There are schemes which implement On-line voting also. Here, validation, collection and tallying all are done on-line. This leads to the need for a much stronger and much more secure system to implement this.

0.4.1 Security Issues

Internet is expanding by dimensions day by day. This makes it more and more probable for attacks by adversaries. Thus any voting system to be implemented which is secure from all types of attacks and cheat, it should be very robust and strong. The various things that could be used are :

- Personal Identification Number (PIN) : This ensures that a voter's ballot can be used only by that person. Thus a voting system based on PIN promises as much security as the PIN promises.

- Various Encryption Protocols : Cryptographic protocols can be used to protect digital ballots and passwords as they travel from one person to the other through the Internet.

- Digital Signatures : These could be used so that election authorities can authenticate a digital ballot by signing it. Also, voters can sign a digital ballot envelope containing their vote, thereby authenticating it.
- Biometric Identifiers eg. voice recognition, fingerprint recognition. System based on this has the special security that voters will not be able to give their passwords to other people, thus this will prevent selling and buying of votes.

0.4.2 Advantages of on-line system

1. On-line voting promises ease of use and decreasing equipment costs. Since every communication is digital, the infrastructure requirements are minimal. The system can be basically menu driven and interactive, thus making it very easy to use on the part of the voter.
2. It provides a perfect mechanism to promote new election policies such as proxy voting and proportional representation eg. if the system allows a person voting on the behalf of some other person then the on-line voting scheme implements this very easily. One just needs to authenticate the person through which he wants to vote. This can be done through by just giving him the password etc. Also, if proportional representation is to be implemented then this can be simply done through ensuring that only an "upper-bound" number of each proportion votes.
3. One of the most promising characteristics that on-line voting possesses is that it provides a whole new crop of election vendors, many of whom have no prior experience in public elections. Thus if anyone sitting in any corner of the world wants to initiate a poll (may be on any topic which could even be done through discussion), he or she can do so by simply asking the administrators of the various voting systems. Also this will bring lots of lots of administrators for different kinds of elections. Thus this system can redefine democracy where each person can initiate an election as well as vote.
4. Finally, Internet's impact on business can't be disputed. It has brought in a whole new dimension to the business through e-commerce. Although the usage of computers and Internet has still not penetrated deep inside, but still the potential that Internet has in future can't be ignored.

0.5 Tail piece

This study is grossly incomplete. To say that the total work in elections has been just this much in past twenty years would be extremely unfair for the brilliant and dedicated researchers of the computer science field. We here have presented most of the successful work applicable in the overall context. There are a good number of historical developments which we have left out either because the process of development was too slow. That is the quality of improvement was minimal and similar features were also incorporated in the next development we presented. The aim is surely neither to undermine somebody's contribution to the subject nor exaggerate somebody's and surely not to hurt anyone's feelings.If unknowingly we have not mentioned some important landmark in the field then we apologize for it. There are anyhow certain schemes which are not applicable to a general election procedure. Some schemes work on very small number of voters model, some work assuming only a yes/no type of ballot paper model etc. which we have not included not for the quality of idea they reflect but for the non-generalism they inherit. They may surely be still the best way to go about in their models.

So on the whole we have tried to focus on the general structure of electronic elections with their possible implementations and advancements. We have not dealt with the existence of a complete fool proof protocol because till date there does not exist one. But the authors feel that, the ideas presented here with certain more thinking should yield a much improved version of the solution which might be real time implementable and may become quite popular in the days to come may be with some more on-line voting systems developed.