Secure E-voting for Preferential Elections [*]

Riza Aditya, Colin Boyd, Ed Dawson, and Kapali Viswanathan
{r.aditya,c.boyd,e.dawson,k.viswanathan}@qut.edu.au

Information Security Research Centre,
Queensland University of Technology,
GPO BOX 2434, Brisbane, QLD 4001, Australia

Abstract. Electronic voting (e-voting) systems can greatly enhance the efficiency, and potentially, the transparency of national elections. However, the security of such systems is an area of on-going research. The literature for secure e-voting is predominantly concerned with 1-out-of-m voting strategies, where m is the number of candidates running for the elections. This paper presents a case study of cryptologic protocols for secure e-voting systems that use preferential voting strategies.

1 Introduction

Many types of voting strategies are employed around the world. Although there has been extensive research for cryptologic protocols for binary voting strategies (yes or no votes), the attention to preferential voting systems [2] is minimal. In such voting systems, each voter is required to rank, provide an order of preference for, the candidates. If no candidate receives a majority, more than half of first preference votes, the candidate with the lowest first preference vote is eliminated. Votes of the eliminated candidate are redistributed to the remaining candidates depending on the second preference. Repeatedly, more candidates are eliminated until one reaches a majority.

A notion of security for e-voting systems can be summarised as the confidentiality service for individual voter-vote relationships. This can be formalised as $Conf(ID, Vote)$, where $Conf(\cdots)$ is a confidentiality service, ID is a voter's identity, and $Vote$ is the vote of that voter. Such a security is achieved using one of two techniques, namely $(ID, Conf(Vote))$ or $(Conf(ID), Vote)$. The vote either remains confidential or anonymous even after the end of the elections. This paper will adapt a proposal from each of these two techniques to preferential voting systems and study the resulting efficiency.

2 Preferential Voting using Homomorphic Encryption

This section will discuss the use of homomorphic encryption in secure e-voting systems, which uses the $(ID, Conf(Vote))$ technique, to accommodate a preferential voting strategy.

[*] Project funded by ARC Discovery 2002, Grant No: DP0211390

2.1 Homomorphic Encryption

Assume that $E^k(v)$ is a public encryption function, where k is a random value chosen by the voter and v is a vote, and $D(\cdots)$ is the corresponding private decryption function known only to the election officials, such that they form a public-key encryption system. The encryption function is said to be homomorphic when $E^{k_1}(v_1) \odot E^{k_2}(v_2) = E^{k_3}(v_1 \oplus v_2)$, where $\odot$ and $\oplus$ are some binary operators. In the LHS of the previous equation, the ciphertexts of individual votes can be *combined* using a binary operator, $\odot$. The RHS of the equation suggests that such a combining operation will result in another ciphertext, the decryption of which will result in an *accumulation* of the individual votes, $v_1 \oplus v_2$. Thus, it will be possible to compute the accumulation of the individual votes without having to retrieve the individual votes.

The Paillier cryptosystem [5] contains a homomorphic encryption function, used for voting, which operates on the congruence class $\mod N^2$, where $N = pq$, and p and q are large prime integers such that the factorisation problem is intractable. The next section will use the above cryptosystem.

2.2 Preferential Voting and Paillier Cryptosystem

Baudron *et al.* [3] proposed a novel technique for the design of 1-out-of-m electronic voting systems using Paillier cryptosystem. The vote is structured in a special form to be combined using homomorphic encryption.

We propose the following *simple* adaptation of the message structure to design an electronic preferential voting system. In this system, the voter is expected to vote for a particular sequence of candidates rather than to vote for the candidates themselves, as was proposed in the original scheme.

Preferential vote: A system constant $M = 2^{\lceil \log_2 l \rceil}$ is chosen, where l is the maximum number of voters in a constituency. The officials assign a unique number, $i \in \{0, \cdots, m!\}$, to every possible sequence (permutation) of m candidates, and accommodate for empty votes. The voter must provide a rank for every candidate or submit an empty vote. The size of the vote in this cryptosystem is $\log_2 M^{m!}$ bits. That is each sequence is represented by a counter that can count up to M. The voter encrypts the vote, M^i, using a homomorphic encryption scheme for the election officials. Equation 1 presents a pictorial representation of the vote (M^i), which chooses the first sequence of candidates, namely, $i = 0$.

$$\overbrace{|\underbrace{\overbrace{00\cdots00}^{\lceil\log_2 l\rceil}}_{i=m!}|\underbrace{\overbrace{00\cdots00}^{\lceil\log_2 l\rceil}}_{i=m!-1}|\cdots|\underbrace{\overbrace{00\cdots00}^{\lceil\log_2 l\rceil}}_{i=2}|\underbrace{\overbrace{00\cdots01}^{\lceil\log_2 l\rceil}}_{i=1}|\underbrace{\overbrace{00\cdots00}^{\lceil\log_2 l\rceil}}_{i=0}|}^{M^i} \tag{1}$$

The public key for a Paillier cryptosystem must be generated such that the modulus $M^{m!} < N^2$ so that the entire vote can be encrypted in one block.

When $m = 20$ and $l = 1000$, the size of the modulus is: $|N| = 10 \times 20!$ bits. Clearly, such a size for a modulus is impractical as numerous exponentiation operations are required.

3 Preferential Voting using Mix-Networks

This section will discuss the use of mix-networks in secure e-voting system. It presents an implementation, using the $(\mathit{Conf}(\mathit{ID}), \mathit{Vote})$ technique, of a robust mix-network for a simple and relatively secure preferential e-voting system.

3.1 Mix-Networks

Let D be a decryption algorithm computable only by the mix-network. Let $\phi : \mathbb{Z}_n \rightarrow \mathbb{Z}_n$ be a randomly chosen secret permutation function. The operation of the mix-network can be described by the following operation: $m_{\phi(i)} = D(c_i)$, where $i \in \mathbb{Z}_n$. The LHS of the previous equation is a random sorted set of plaintexts, $m_{\phi(i)}$, corresponding to the RHS, decryption, D, of a set of input ciphertexts, c_i.

The input to the mix-network could be of the form $(\mathit{ID}, \mathit{Conf}(\mathit{Vote}))$, and the output would be of the form $(\mathit{Conf}(\mathit{ID}), \mathit{Vote})$. Thus, a mix-network can be viewed as a confidentiality translation service translating the confidentiality service from the vote (or data) to the identity.

Abe [1] proposed a mix-network using ElGamal encryption algorithm in re-encrypting the set of input messages, $I_v = \{c_i | i \in \mathbb{Z}_n\}$, to produce a randomly permuted set of output messages, $O_v = \{m_{\phi(i)} | i \in \mathbb{Z}_n\}$. The next section will use the mix-network by Abe.

3.2 Preferential Voting and Mix-Networks

The mix-network by Abe can be used to construct electronic secret ballot voting schemes. In such schemes, the vote need not be in a special form as the tabulation phase is conducted using the plaintext vote anonymised. In contrast to schemes based upon homomorphic encryption, the size of input messages does not directly affect the efficiency of the mix-network.

In electronic preferential voting employing mix-network, the vote can be of the form an integer ranging from 1 to $m!$, where m is the number of candidates running for the election. Thus, the message size is $\lceil \log_2(m!) \rceil = 62$ bits, where $m = 20$. Let $j \in \mathbb{Z}_l$ be an index into a list of voters, so that l is the number of voters.

We propose a generic e-voting scheme utilising the above mix-network as follows:

Set-up: The election officials publish the parameters for a threshold decryption cryptosystem [4], (E, D), such that $E^k(\cdots)$ is the public encryption function, where k is a random value chosen by the voter, and $D(\cdots)$ is a t-out-of-n decryption function.

Vote submission: Each voter j:

1. selects a sequence, $i_j \in \{1, \cdots, m!\}$, to represent his/her preference;
2. encrypts the selection, e.g: $E^{k_j}(i_j)$;
3. identifies itself to the vote collecting official to establish its identity, ID; and,
4. communicates the encrypted vote, $E(i_j)$, to the vote collecting official.

The voting official verifies the identity of every voter and forwards the set of votes from valid voters, $U = \{E^{k_j}(i_j)\}$, to the mix-network.

Vote mixing: The mix-network permutes and decrypts the objects from the set U and outputs a set of plaintext votes V, such that the correspondence between the objects in U and the objects in V are a secret.

Vote tabulation: The election official electronically processes the plaintext vote in V by using a program for counting the preferences and calculating the elected candidate.

The processing cost is mainly contributed by the mixing operation. The mix-network is composed of a number of gates, and the computational cost for each gate is 23.6 modular exponentiations [1]. Each voter generates exactly one input message to the mix-network. Providing $2 \log_2 l - 1$ delay [1] per input, where $l = 1000$ is the number of inputs to the mix-network, total computational cost of the mix-network is $23.6(2 \log_2 l - 1)t = 448400$ modular exponentiations.

4 Conclusion

The size of the electronic vote for a preferential voting system is inherently larger than a 1-out-of-m voting system, when the number of candidates, m, increases. In preferential voting system, the size of the vote is at least $\log_2(m!)$ bits. Therefore the voting systems using some form of homomorphic encryption [3] tend to be inefficient or impractical.

Voting systems that employ mix-networks, on the other hand, do not require a special form for the electronic vote. The computational complexity is not adversely affected by the number of candidates. Therefore, mix-network based voting systems are ideally suited for preferential voting systems. Future research will be directed towards the design of more efficient robust mix-networks.

References

1. Masayuki Abe. Mix-networks on permutations networks. In *Advances in Cryptology—ASIACRYPT 99*, pages 258–273, 1999.
2. Australian Electoral Commission. *Australian Electoral Commission*, 2002. Available from http://www.aec.gov.au, last accessed 17 February 2003.
3. Olivier Baudron, Pierre-Alain Fouque, David Pointcheval, Jacques Stern, and Guillaume Poupard. Practical multi-candidate election system. In *ACM symposium on Principles of distributed computing*, pages 274–283, 2001.
4. Yvo Desmedt and Yair Frankel. Threshold cryptosystems. In *Advances in Cryptology—CRYPTO 89*, pages 307–315, 1989.
5. Pascal Paillier. Public key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology—EUROCRYPT 99*, pages 223–238, 1999.