공개키 기반구조 하에서의 안전한 인터넷 전자 투표 프로토콜 설계

김진호, 김광조
국제정보보호기술연구소
한국정보통신대학원대학교

Design of Secure Internet Voting Protocol under PKI

Jinho Kim, Kwangjo Kim
International Research center for Information Security (IRIS)
Information and Communications University (ICU)
{kman, kkj}@icu.ac.kr

요 약

본 연구에서는 인터넷을 사용하여 누구나 투표에 참여할 수 있는 공개키 기반구조 하에서의 안전한 인터넷 전자 투표 프로토콜을 설계하였다. 설계된 전자 투표 프로토콜은 은닉 서명과 믹스 서버를 이용하여 투표자의 익명성을 제공하고 투표값의 기밀성은 이산대수 문제의 안전성에 기반 한 ElGamal 공개키 암호를 통해 달성되며 기타 전자 투표 프로토콜이 요구하는 기본 사항들을 만족시킨다. 투표자의 이중투표 방지는 공개키 기반구조에서 제공하는 인증서를 통해 이루어짐으로, 인터넷을 통해 인증서를 가진 사람은 누구나 투표를 할 수 있도록 되어있다.

I. 서론

인터넷을 사용한 전자 상거래, 교육, 개인간의 통신 등이 널리 사용되면서 인터넷을 이용한 전자 투표가 가능한가 라는 의문이 제기 되어 왔다. 만약 투표가 정확하게 이루어 질 수 있고, 사용하기가 편리하다면 인터넷 전자 투표 시스템은 전 세계적인 현상인 투표율 저하의 문제를 해결할 수 있는 대안이 될 것이다.

본 연구에서는 인터넷을 사용하여 누구나 투표에 참여할 수 있는 인터넷 전자 투표 프로토콜을 설계하였다. 설계된 전자 투표 시스템은 은닉 서명 방식을 이용한 전자 투표 프로토콜을 기반으로 공개키 기반구조와 인터넷을 접목시켰다. 이 프로토콜에서 익명성은 은닉 서명과 믹스 서버를 이용하여 이루어지며, 투표값의 기밀성은 이산대수 문제의 안전성에 기반한 ElGamal 공개키 암호를 통해 달성된다. 투표자의 이중투표 방지는 공개키 기반구조에서 제공하는 인증서를 통해 이루어지며, 인터넷을 통해 인증서를 가진 사람은 누구나 투표를 할 수 있다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 전반적인 개념 소개로 전자 투표 프로토콜의 요구 사항을 다룰 것이며, 3장에서는 전자 선거 프로토콜에서 사용되는 암호 기법들과 공개키 기반 구조에 대해서 설명한다. 4장에서는 설계한 프로토콜을 자세히 기술하며, 5장에서는 설계한 프로토콜의 안전성에 대해서 알아본다. 마지막으로 6장에서 결론을 맺는다.

II. 전자 투표 요구 사항

전국 규모로 시행되는 현행 선거에서의 일반적인 구성요소로는 선거 기관, 개표자, 투표자들이 있다. 선거 기관은 통상 신뢰할 수 있는 공공 기관이라 할 수 있으며 유권자를 등록하고 선거 시행 시 유권자를 확인하고 비밀이 보장되는 장소를 제공하여 선거를 하게 한다. 투표가 완료된 다음에는 투표지를 투표함에 담아 봉인 표기를 하여 취합하고 유효한 투표를 집계한 후 결과를 발표하게 된다. 이러한 투표 과정을 대략 살펴보면 유권자의 명부를 작성하는 유권자 등록단계, 유권자가 본인임을 확인 받는 단계, 투표하는 단계, 각 투표지를 취합하는 수집 단계, 투표 내용을 공개하는 개표 단계, 후보자에 대한 기표별로 더하는 계수단계와 결과의 발표단계로 구분할 수 있다. 전자 선거는 이러한 일련의 과정이 온라인(On-line) 상에서 공정하고 안전하게 유지되도록 하여야 하고 더불어 투표자의 비밀을 보장하여야 한다. FOO92[8]는 이러한 속성을 분류하여 7가지로 정의하였다.

① 완전성(Completeness) : 모든 유효 투표가 정확하게 집계되어야 한다는 것으로, 최종 집계에서 정당한 투표가 제거되는 일이 없어야 한다.
② 건전성(Soundness) : 부정한 투표자에 의해 선거가 방해되는 일이 없어야 한다는 것으로, 최종 집계에서 부정 투표가 집계되어 선거에 영향을 끼치지 않아야 한다.
③ 기밀성(Privacy) : 모든 투표가 비밀로 되어야 한다는 것으로, 특히 투표결과로부터 투표자를 구별할 수 없어야 한다.
④ 단일성 또는 이중투표불가성(Unreusability) : 정당한 투표자가 두 번 이상 투표할 수 없다는 것으로 단지 한 번만 투표할 수 있어야 한다.
⑤ 적임성 또는 선거권(Eligibility) : 투표 권한을 가진 자만이 투표할 수 있는 것으로 투표가 허락되지 않은 사람은 투표할 수 없어야 한다.
⑥ 공정성(Fairness) : 투표에 영향을 미치는 것이 없어야 한다는 것으로 투표 중에 일부

분 결과를 알게 되어 투표에 영향을 미치는 상황 등이 없어야 한다.

⑦ 검증성(Verifiability) : 선거 결과를 변경할 수 없도록 누구라도 투표 결과를 확인하여 검증해 볼 수 있어야 한다.

이 후에 FOO92에서는 거론되지 않았던 유권자의 매표방지(Receipt-freeness)에 관한 논문[4],[10],[11]이 발표되어 매표방지가 전자선거의 중요한 최근 요구사항으로 대두되었으며 물리적으로 안전한 스마트카드 또는 정직한 검증자를 가정하에 설계가 가능하나 실제 구현에는 상당한 계산력이 요구된다. 또한, FOO92에서 분류한 일곱 번째 속성인 검증성의 요구조건은 자신의 투표결과를 검증하는 개별 검증성(Local verifiability)과 누구나 투표 결과를 검증할 수 있는 전체 검증성(Universal verifiability)으로 구분할 수 있다. 또한, 투표 단계에서 의도하지 않은 네트워크의 절단 등과 같은 투표 처리 과정의 장애 발생 시 이를 처리할 수 있는 원자성(Atomicity)도 제공되어야 한다.

본 논문에서 제안하는 프로토콜은 FOO92 프로토콜을 기반으로 하는 OMAFO99[12] 프로토콜을 공개키 기반구조 하에서의 인터넷 전자 프로토콜로 확장한 것이므로 기본적으로 위의 7가지 조건을 모두 충족시킨다 할 수 있다. 또한, 인터넷 상에서 발생할 수 있는 투표 중단에 대한 원자성도 제공하도록 설계되었다.

III. 관련 연구

다음에서 전자 투표 프로토콜에서 사용되는 주요 암호 기법들을 기술하며, 제안한 인터넷 전자 투표 프로토콜의 기본이 되는 OMAFO99 프로토콜[12]에 대해서 알아본다.

1. 사용된 암호기법

(1) ElGamal 공개키 암호 시스템

이 방식은 이산 대수 문제(Discrete Logarithm Problem)의 어려움에 기초한 공개키 암호 시스템으로 다음과 같이 동작한다.

큰 소수 p와 p-1을 나누는 소수 q가 있을 때 G를 위수가 q인 Z_p^* 의 부분 군이라고 하고, g를 G의 생성자라고 하자. 즉 G=<g> 가 된다. 이때 ElGamal 암호 기법에서 $x \in Z_p^*$ 와 $y = g^x \bmod p$ 는 개인키와 공개키가 된다. 주어진 메시지 m에 대한 암호화와 복호화는 다음과 같다. 먼저 송신자는 난수 $a \in Z_q$를 선택한 후 암호문 (G,M) = (g^a, my^a) 을 만들어 수신자에게 보낸다. 수신자는 자신의 개인키를 이용하여 다음과 같이 $m = M/G^x \bmod p$ 복호화하여 평문을 얻어낸다.

(2) 은닉서명

은닉 서명(Blind Signature)은 A가 B에게 메시지의 내용을 모르게 하면서 메시지에 대한 서명을 받는 방식으로 메시지에 대한 익명성을 보장하기 위하여 사용된다. 이 프로토콜

에서 사용된 Schnorr 방식 은닉서명은 다음과 같다.

o Schnorr 방식 은닉서명[14],[15]

- 비밀 메시지 m에 대한 서명을 얻어내기 위해서 사용자 A는 서명자에게 서명을 위한 통신을 요청한다.
- 서명자 : 서명자는 난수 $k \in Z_q$를 선택한 후 $r = g^k \bmod p$를 계산해서 A에게 r 을 보낸다.
- 사용자 A : 난수 $a, b \in Z_q$ 를 선택한 후, $r' = rg^a y^b \bmod p$와 e'=Hash(m,r')을 계산하여 서명자에게 e = e'- b mod q를 보낸다.
- 서명자 : 서명자는 s=k-e*x mod q를 계산해서 A에게 보낸다.
- 사용자 A : s'=s+a mod q를 계산하여 메시지 m에 대한 은닉 서명 (e',s')을 얻는다.

o 서명 검증 : 서명에 대한 검증은 e' = Hash(m, $g^{s'} y^{e'} \bmod p$)가 같은지를 확인하는 것으로 이루어진다.

(3) 전자 서명

공개키 암호 시스템의 개인키를 이용하여 메시지를 암호화하여 보내는 방법으로 상대는 서명자의 공개키로 수신된 메시지를 검증해 봄으로써 서명자를 유일하게 확인할 수 있다. 전자 투표 시스템에서는 Schnorr 방식 전자 서명[15]을 사용한다.

서명 과정 : 평문 m

- 난수 $k \in Z_q$를 선택한 후 r= g^k mod p를 계산한다.
- e=Hash(m,r), s=k-e*x mod q를 계산한다.
- 문서 m에 대한 서명 : (e,s)

검증 과정 : 수신자는 e = Hash(m,$g^s y^e$ mod p)가 같은지 검증한다.

(4) 문턱 암호

문턱암호(Threshold Cryptosystem)는 Shamir에 의해서 1879년 처음 제안되었다.[17] Shamir에 의해서 제안된 (k,n) 문턱 암호는 비밀 정보 D를 n개의 조각 D1,....,Dn 으로 나누어서 n명이 나누어 가지게 한 후, 그중 k명 이상이 모여 비밀 정보를 복원하는 방식이다. 이 방식에서 k개 미만의 정보로는 비밀정보 D를 복원하는 것이 계산량적으로 매우 힘들어야 한다. Shamir가 제안한 방식은 다항식 보간법을 이용한다.

(5) 익명 통신로

익명 통신로(Anonymous Channel)란 송신자와 송신자가 보낸 메시지간의 관계를 숨길 수 있는 통신로를 말한다. 이러한 익명 통신로를 만들기 위해 사용되는 방법으로 Mix-net[7]이 있다. Chaum이 처음 제안한 이 방식은 복수의 믹스 서버를 이용한다. 각각의 믹스 서버는 입력 값의 모양을 바꾼 후 랜덤 한 순서로 출력해서 외부에 입력 값과 출력 값의 관계를 숨기는 방식이다. 그러나 일부 서버가 입력 데이터를 부정한 방법으로 개조하는 경우 이용자가 처음부터 잘못된 값을 보냈는지, 서버가 부정행위를 했는지 알 수가 없다. 따라서 각각의 서버가 정확히 계산을 정확히 했는지를 검증할 수 있는 다양한 방식[1],[2] 을

제시한 연구가 제안되었으나, 효율성에 있어서 많은 문제점을 가지고 있다. 또한 전체 검증성을 보장하지 않으나 크기가 큰 데이터를 효과적으로 처리 할 수 있도록 공개키 방식과 대칭키 방식을 혼합한 방식[13]도 제안되었다. 제안한 시스템에서는 혼합한 방식을 사용한다.

(6) 공개키 기반 구조

공개키 암호 시스템은 각종 정보보호 시스템에서 전자 문서에 인증성, 전자 서명, 무결성 등을 포함하는 안전한 서비스를 제공하는 핵심 요소이다. 공개키 암호 시스템에서 사용자는 개인키와 수학적으로 연결되어져 있는 공개키를 갖는다. 여기서 공개키는 누구나 알 수 있도록 공개되어지며, 개인키는 소유자만이 알 수 있는 비밀 장소에 보관된다. 공개키 시스템의 문제점은 공개되어진 키가 원래 소유자의 공개키인지를 어떻게 보장할 수 있느냐 하는 것이고, 이러한 문제를 해결하기 위한 방안으로 제시된 것이 공개키 기반 구조(PKI : Public Key Infrastructure)이다. 공개키 기반 구조에서는 신용할 수 있는 인증기관(CA)이 존재해서 소유자와 공개키를 묶어주는 인증서를 발행한다. 인증서는 인터넷 상에서 사용자의 실체를 확인하는 개인 신분증과 같은 역할을 하며, 인증서에 포함된 공개키를 이용해서 공개키 암호시스템이 제공하는 여러 서비스를 수행할 수 있다.

2. OMAFO99 프로토콜

FOO92에서 제안된 은닉 서명을 이용한 전자 투표 프로토콜은 투표자가 개표 시 다시 투표에 참여 해야한다는 문제점을 가지고 있다. 이런 약점을 문턱암호를 이용해서 개선한 프로토콜이 OMAFO99이며 다음과 같다.

- 기호 정의

V_i : 투표자 i , C : 개표자 , A : 선거관리자 , M : 믹스서버 , BB : 공개 게시판

EC() / DC() : 개표자의 공개키/개인키를 이용한 암호화/복호화 함수

EM() / DM() : 믹스 서버의 공개키/개인키를 이용한 암호화/복호화 함수

$Sign_i()$: 투표자 V_i 의 서명생성 함수 , Sign_A() : 선거관리자 A 의 서명생성함수

BLIND() : 은닉 함수 , UNBLIND() : 비은닉 함수

ID_i : V_i 의 식별정보 , v_i : V_i 의 투표값

가. 등록단계(Registering stage)

Step 1 :

1) 투표자 V_i는 투표값 v_i를 선택한다.

2) 개표자 C의 공개키를 이용한 암호화한다. : x_i = EC (v_i)

3) x_i를 난수 r_i를 이용해서 은닉한다. : e_i = BLIND (x_i, r_i)

4) e_i에 대한 서명을 생성한다. : s_i = $Sign_i$ (e_i)

5) 선거관리자 A 에 $\langle ID_i, e_i, s_i \rangle$를 전송한다.

Step 2 :

1) A 는 V_i가 투표권한이 있는지를 검사한다.

- 만약 권한이 없다면 A는 인증을 거부한다.

2) A 는 V_i가 이미 투표를 했는지 검사한다.

- 만약 했다면 A는 인증을 거부한다.

3) A 는 V_i의 서명 s_i를 검증한다.

- 만약 서명이 유효하지 않다면 인증을 거부한다.

4) 서명이 유효하다면 A는 e_i에 서명한다. : d_i = Sign_A (e_i)

5) A 는 d_i를 V_i에게 전송한다.

Step 3 :

1) V_i는 x_i에 대한 서명 y_i를 추출한다. : y_i = UNBLIND (d_i, r_i)

2) y_i가 x_i에 대한 A의 서명인지를 검증한다.

3) 만약 검증이 실패하면, V_i 는 〈 x_i, y_i〉가 유효하지 않다고 주장한다.

나. 투표단계(Voting stage)

1) 투표자는 〈 x_i, y_i〉를 믹스서버 M 의 공개키로 암호화한다. : c_i = EM(〈 x_i, y_i〉).

2) c_i에 대한 서명을 생성한다. : s_i = Sign$_i$ (c_i).

3) 공개 게시판 BB에 〈 c_i, s_i〉전송한다.

다. 개표단계(Counting stage)

Step1 :

1) M은 BB의 c_i를 복호화 하여 무작위 순서로〈 x_i, y_i〉를 출력한다.

Step2 :

1) C 는 x_i에 대해서 y_i가 정당한 A의 서명인지 검증한다.

2) 검증에 실패하면, 〈 x_i, y_i〉를 공개한다.

3) 만약 개표자중 정직한 t명 이상이 〈 x_i, y_i〉와 같은 값이 이미 존재한다고 주장하면 M은 영지식 증명 기법을 이용해서 〈 x_i, y_i〉가 c_i를 복호화한 값이라는 것을 증명한다. 각각의 개표자는 증명의 정당성을 검증한다.

- 검증이 실패하면 M의 부정 행위임으로 해당 M을 믹싱 과정에서 제외시킨다.
- 검증이 성공하면 M은 정직하나, 투표자가 유효하지 않은 정보를 보낸 것으로 되므로 투표값을 개표에서 제외시킨다.

Step3 :

1) 모든 개표자는 문턱 암호를 이용해서 x_i를 복호한다. : v_i = DC (x_i)

2) v_i를 공개 게시판에 공개한다.

3) 검증자는 투표수와 투표자 수가 일치하고, 개표자가 정확히 개표했는지 검사한다.

4) 검증이 실패하면 무효를 주장한다.

IV. 공개키 기반구조 하에서의 전자 투표 프로토콜

이번 장에서는 공개키 기반구조 하에서의 인터넷 전자 투표 프로토콜에 대해서 설명한다. 제안한 프로토콜은 OMAFO99 프로토콜을 기반으로 하며 공개키 기반구조 하에서의 인증서를 이용한다.

1. 사용된 기호

아래 표기는 투표 프로토콜 설계에 사용된 기호와 그의 정의들이다.

- 투표 참여자

투표자, 선거 관리자, 믹스 서버, 개표 서버, 공개 게시판, 인증기관

- 연산

* : 곱셈 , ^ : 지수승 , % : 나머지 연산 , || : 연접

- 기호

Cert	: 투표자의 인증서
ID	: 투표자의 식별정보
PubKeyVoter/PrvKeyVoter	: 투표자의 공개키/개인키
PubKeyAdm/PrvKeyAdm	: 선거관리자의 공개키/개인키
PubKeyTallier /PrvKeyTallier	: 개표자의 공개키/개인키
PubKeyTallier$_i$ /PrvKeyTallier$_i$	: 개표자 i의 공개키/개인키
PubKeyMixer/PrvKeyMixer	: 믹스 서버의 공개키/개인키
EncPubElg()/DecPubElg()	: ElGamal 암/복호 함수
ElgFactGM = (G,M)	: EncPubElg 으로 생성된 값
ElgFactG/ElgFactM	: ElgFactGM 의 G/M
EncSym()/DecSym()	: 대칭키 암/복호 함수
SesKey	: EncSym/DecSym 에 사용된 대칭키
Hash()	: 해쉬 함수
Sign()/Verify()	: 서명/검증 함수
SigVoter	: 투표자의 서명값
SigAmin	: 선거관리자의 서명값
SigMidxer	: 믹스 서버의 서명값
SigTallier	: 개표자의 서명값
BlindCommitment()	: 은닉 서명에 사용될 난수 생성 함수

tildeA	: BlindCommitment 의 결과
Blinding()/Unblinding()	: 은닉 함수/비은닉 함수
BlindSign()	: 은닉 서명 함수
BldSig	: 은닉 서명값

2. 인터넷 전자 투표 프로토콜

1) 등록 및 인증서 발급 단계

ㅇ 투표자

- 선거관리자 웹사이트에 들려서 등록 양식을 받는다.
- 등록 및 인증서 발급에 필요한 정보를 입력한다.
  * ID, 이름, 패스워드, e-mail, 주소 등
  * ID의 이중성과 e-mail 을 검사한다.
- 입력정보를 암호화하여 선거관리자에게 보낸다.

ㅇ 선거관리자

- 투표자 정보를 복호화한다.
- 투표자 정보를 선거 관리자의 DB에 저장한다.

ㅇ 투표자

- 키 생성 프로그램을 받는다.
- 키생성 프로그램에 ID와 패스워드를 넣고 ElGamal 키쌍을 생성한다.
- 개인키(PrvKeyVoter)는 투표자 PC에 저장하고, 공개키는 인증서 생성을 위해서 선거관리자에게 보낸다.

ㅇ 선거관리자

- 등록된 정보(ID, 이름, e-mail)와 투표자의 개인키(PubKeyVoter)를 가지고 인증기관에 인증서 발급 절차 수행을 요청한다.
- 인증기관에게서 받은 인증서를 선거관리자 DB에 저장한다.

ㅇ 투표자

- 인증서가 발급됐다는 화면을 본다.

2) 인증 단계

인터넷을 이용한 전자 투표이므로 투표 전 반드시 투표자는 인터넷 상에서 자신을 인증하는 절차가 필요하며, 선거 관리자는 투표자의 신분과 이중 투표 여부를 확인한 후 투표할 수 있는 권한을 부여한다.

ㅇ 투표자

- ID와 패스워드를 입력해서 개인키(PrvKeyVoter)를 찾는다.
- 난수 r을 Z_q^*에서 생성한다.
- ID||r||SigVoter를 선거관리자에 전송한다. : SigVoter=Sign(ID||r)

ㅇ 선거관리자

- 투표자의 ID등록 여부 확인, 미등록 시 인증을 거부한다.
- 투표자의 서명을 확인한다. : Verify(SigVoter)
- 인증서 DB에서 해당하는 인증서를 가져와서 서명을 검증한다.
- 서명이 틀리면 인증을 거부한다.
- 투표자의 이중투표 여부를 확인한다. 이중투표 시, 인증을 거부한다.

3) 투표 단계

ㅇ 투표자

- 투표자는 투표프로그램을 선거관리자로부터 받는다.
- ID와 패스워드를 입력해서 PrvKeyVoter를 찾는다.
- 후보리스트에서 원하는 후보를 선택한다. : 선택값 v_i
- 난수 I를 생성.
- m을 다음과 같이 만든다. : m = v_i || I
- 개표자의 공개키(PubKeyTallier)를 이용한 암호화한다.: ElgFactGM= EncPubElg(m)
- 선거관리자에 tildeA를 요청한다.

ㅇ 선거관리자

- BlindCommitment()를 실행한다.

  랜덤값 Omega를 Z_q^*에서 생성한다.

  tildeA를 다음과 같이 계산한다. : tildeA=(g^Omega) % p

  선거관리자 DB에 tildeA, Omega를 저장한다.
- tildeA를 투표자에게 전송한다.

ㅇ 투표자

- tildeA를 받는다.
- Blinding()함수를 실행한다.

  은닉암호에 필요한 난수 a,b를 Z_q^*에서 생성한다.

  W를 계산한다. : W=tildeA*(g^a)*(PubKeyAdmin^b) % p

  C = Hash(ElgFactGM,W) 계산한다.

  은닉 암호값 tildeC를 계산한다. : tildeC= (C - b) % q
- 은닉 암호값에 대한 서명을 생성한다. : SigVoter = Sign(tildeC)
- 선거관리자에게 evsig = ID||tildeC||SigVoter를 전송한다.

ㅇ 선거관리자

- 투표자로부터 evsig를 수신한다. : ID||tildeC||SigVoter = evsig
- 인증서를 이용해서 투표자의 서명을 검증한다. : VerifySign(SigVoter)
- 선거관리자 DB에 tildeC와 SigVoter를 저장한다.

- BlindSign을 실행한다. : ezc=(Omega - tildeC*PrvKeyAdmin) % q
- 투표자에 ezc를 전송한다.

o 투표자
- 선거관리자로부터 ezc를 받는다.
- Unblinding() 함수를 실행한다.

  Z = (ezc + a) % q를 계산한다.

  C 와 Hash(ElgFactGM,(g^Z)*(PubKeyAdmin^C) % p)값이 같은지 검증한다.

  Schnorr 은닉 서명 BldSig = (Z||C)을 만든다.
- 믹스 서버의 공개키를 이용한 공용키를 생성한다.

  난수 r을 Z_q^*에서 생성한 후, G'= g^r % p을 계산한다.

  공유 비밀키를 생성한다. : SesKey = Hash(PubKeyMixer^r % p)
- 공유 비밀키(SesKey)를 이용한 대칭키 암호화를 수행한다.

  esev= ElgFactGM||BldSig

  M'= EncSym(esev)
- (G'||M')에 서명한다. : SigVoter=Sign(G'||M')
- 공개 게시판 에 emgzc를 전송한다. : emgzc = ID||G'||M'||SigVoter

o 공개 게시판
- 투표자로부터 emgzc를 받는다. : ID||G'||M'||SigVoter = emgzc
- 인증서 DB에서 ID에 해당하는 인증서를 가져와서 투표자의 서명을 검증한다.
- 이중 투표 여부를 확인한다. 이미 투표했다면 투표거부 화면을 전송한다.
- 투표자의 투표 여부 정보를 미 투표에서 투표로 수정한다.
- 투표 성공화면을 투표자에게 전송한다.
- (ID,G',M',SigVoter)를 투표 DB에 저장한다.
- 공개 게시판에 투표값을 공개한다.

* 모든 투표 종료후

4) 믹싱 단계

o 믹스 서버
- 투표 DB에서 투표값 E_i = (G_i',M_i')를 꺼낸다.
- 전체 L개 투표값에 대한 리스트를 만든다. : E = (E_1,...., E_L)
- 믹스 서버의 개인키로 각각의 투표값을 복호한다.

  for j=1 to L

  Let (G',M') = E_j

  SesKey = Hash(G'^PubKeyMixer % p)

  $esev_j$ = DecSym(M') using SesKey
- 복호한 투표값을 랜덤한 순서로 재배열한다.

D = (D_1,...., D_L) = Shuffle($esev_1$,..., $esev_L$)

- 리스트 D에 서명한다. : SigMixer = Sign(D)
- 리스트와 서명을 개표자에게 보낸다. : D||SigMixer

5) 개표 단계

o 개표자

- D||SigMixer를 받는다. : (D_1,..., D_L) = D, D_i=ElgFactGM||BldSig
- 믹스 서버의 공개키(PubKeyMixer)를 이용해서 서명을 검증한다. : Verify(SigBB)
- D_i 에서 BldSig가 선거관리자의 서명인지 검증한다. OMAFO99에서는 영지식 증명기법으로 믹스 서버의 믹싱 작업을 검사하나, 영지식 증명방식은 쌍방향 통신로가 필요하고 계산력 소요가 커서 실제 구현에는 어려움이 있으므로, 여기서는 선거 관리자의 서명에 대한 검증만으로 대신한다.

  ElgFactGM||BldSig = D_i, (Z||C) = BldSig

  C 가 Hash(ElgFactGM,(g^Z)*(PubKeyAdmin^C) % p)와 같은지 검사한다.
- 개표자는 각각의 개인키로 부분 복호화를 수행한다.

  for j=1 to L in D

  ElgFactG||ElgFactM = ElgFactGM

  G_j = ElgFactG^PrvKeyTallier % p
- F = (G_1,....., G_L)
- 개표자는 G_j 에 대한 문턱암호 복호화를 수행한다.

  for j=1 to L in F

  v_j || I = (M_j/G_j)%p
- 투표 DB에 개표 결과를 저장한다.
- 개표자는 v_i 에 대한 결과 집계한다.
- 개표 결과와 집계 결과를 공개 게시판에 계시한다.

V. 시스템 안전성

제안한 공개키 기반구조 하에서의 인터넷 전자 투표 프로토콜은 OMAFO99의 프로토콜을 기반으로 하고 있으면, OMAFO99 프로토콜은 FOO92의 모든 특성을 가지고 있다. FOO92와의 중요한 차이라면 FOO92의 중요한 약점인 투표자가 개표 시 참여해야한다는 것(No walk-away)인데, 이는 투표 프로토콜을 비현실적으로 만든 점이다. OMAFO99에서는 이런 약점을 문턱암호를 이용해 보완하였다. 또한 OMAFO99는 전자 투표 프로토콜의 7가지 요구조건을 모두 충족시키므로, 이를 적용한 설계 시스템에서도 그대로 유효하다. 먼저 전자 투표 프로토콜의 익명성은 은닉 서명과 믹스 서버를 이용하여 이루어지며, 투표값의 기밀성은 이산대수 문제의 안전성에 기반 한 ElGamal 방식 암호를 통해 달성된다. 공개키 기반구조에서 제공하는 인증서와 은닉 서명을 사용해서 투표자의 이중투표 방지와 선거권이

있는 유권자만 투표할 수 있는 서비스를 제공한다. 인터넷을 통해 인증서를 가진 사람은 누구나 투표를 할 수 있다. 제안한 프로토콜에서 완전성, 공정성, 건전성은 문턱 암호를 사용해서 달성된다. 즉, n 명의 개표자중 t명 이상이 정직하다면 개표 결과에 대한 3가지 조건을 모두 충족시킨다. 또한, 최종 집계 결과를 기록하는 공개 게시판이 투표값을 받아야 투표 여부가 결정되므로 중간에 투표가 중단된다하더라도 투표자는 투표를 처음부터 다시 시작할 수 있다. 즉, 원자성이 보장된다. 하지만, FOO92가 만족시키지 못하는 매표방지와 전체 검증성은 이 프로토콜에서도 제공하지 못한다.

VI. 결론

본 연구에서는 인터넷을 사용하여 누구나 투표에 참여할 수 있는 공개키 기반구조를 이용한 인터넷 전자 투표 프로토콜을 설계하였다. 설계된 전자 투표 프로토콜은 은닉 서명 방식을 이용한 전자 투표 프로토콜을 기반으로 공개키 기반구조와 인터넷을 접목 시켰다. 현재까지 나와있는 전자 투표 시스템은 대부분 은닉 서명방식을 이용[5],[8],[9] 하거나 암호 프로토콜의 준동형(homomorphism)[3],[6] 성질을 이용해서 설계되었으나, 공개키 기반구조 하에서의 인터넷 전자 투표에 대해서는 고려하지 않았다. 따라서 현재 우리가 설계한 프로토콜은 공개키 기반구조와 인터넷을 사용한 유일한 방식으로 추후에 구현 결과를 보고할 예정이다.

[참고문헌]

[1] M. Abe, Universally Verifiable Mix-net with Verification Work Independent of the Number of Mix-servers , Advances in Cryptology-Eurocrypt 98, LNCS Vol. 1403, pp.437-447, Springer-Verlag, 1998

[2] M. Abe, "Mix-network on Permutation Networks", Advances in Cryptology-Asiacrypt"99, LNCS Vol. 1716, pp.258-273, Springer-Verlag, 1999

[3] J. M. Adler, W. Dai, R. L. Green, C. A. Neff, "Computational Details of the VoteHere Homomorphic Election System", VoteHere Inc., 2000

[4] J. C. Benaloh and D. Tuinstra, "Receipt-free Secret Ballot Elections", Proc. of 26th ACM STOC, pp. 544-553, 1994

[5] L. F. Cranor and R. K. Cytron, "Design and Implementation of a Practical Security-Conscious Electronic Polling System", WUCS-96-02, Dept. of CS, Washington University, St. Louls, 1996

[6] R. Cramer, R. Gennaro, and B. Schoenmakers, "A Secure and Optimally Efficient Multi-Authority Election Scheme", Advances in Cryptology-Eurocrypt"97, LNCS Vol. 1233, pp.103-118, Springer-Verlag, 1997

[7] David L. Cham, "Untracable Electronic Mail, Return Addresses, and Digital Pseudonyms", Comm. ACM 24, pp.84-88, 1981
[8] A. Fujioka, T. Okamoto and K. Ohta, " A Practical Secret Voting Scheme for Large Scale Election", Advances in Cryptology-Auscrypt92, LNCS Vol. 718, pp. 248-259, Springer-Verlag, 1993
[9] M. A. Herschberg, "Secure Electronic Voting Over the World Wide Web", Master Thesis, Dept. of EE and CS, MIT, 1997
[10] B. Lee, and K. Kim, "Receipt-free Electronic Voting through Collaboration of Voter and Honest Verifier", Proceeding of JW-ISC2000, pp.101-108, Jan. 25-26, 2000, Okinawa, Japan.
[11] V. Niemi and A. Renvall, "How to Prevent Buying of Voters in Computer Elections", Advances in Cryptology-Asiacrypt"94, LNCS Vol. 917, pp. 164-170, Springer-Verlag, 1994
[12] M. Ohkubo, F. Miura, M. Abe, A. Fujioka and T. Okamoto, "An Improvement on a Practical Secret Voting Scheme", Information Security'99, LNCS Vol.1729, pp.225-234, Springer-Verlag, 1999.
[13] M. Ohkubo and M. Abe, " A Length-invariant Hybrid Mix", Advances in Cryptology-Asiacrypt 2000, LNCS Vol. 1976, pp. 192-204, Springer-Verlag, 2000
[14] D. Pointcheval and J. Stern, "Security Arguments for Digital Signatures and Blind Signatures", Journal of Cryptology, LNCS Vol. 13, pp. 361-396, Springer-Verlag, 2000
[15] C. P. Schnorr, "Efficient Identification and Signatures for Smart Cards", Advances in Cryptology-Crypto"89, LNCS Vol. 435, pp.239-251, Springer-Verlag, 1990
[16] B. Schoenmakers, "A Simple Publicly Verifiable Secret Sharing Scheme and its Application to Electronic Voting", Advances in Cryptology-Crypto"99, LNCS Vol. 1166, pp. 148-164, Springer-Verlag, 1999
[17] A. Shamir, "How to Share a Secret", Comm. ACM 22, pp.612-613, 1979