

향상된 사용자 편의성을 갖는 안전한 전자 투표 영수증 발급 방식

A Secure Receipt Issuing Scheme for e-Voting with Improved Usability

요 약

세계적으로 암호 방식을 이용한 전자 투표에 대한 논의가 활발하지만 전자 투표 실시를 막는 가장 큰 문제점으로 많은 전문가들은 전자 투표기에 대한 신뢰성 문제를 들고 있다. 최근 이를 해결하기 위해 전자 투표기를 검증할 수 있도록 투표자에게 영수증을 발급하는 방안에 대한 연구가 활발히 진행되고 있다. 전자 투표기와 같은 암호화 기기에 대한 검증은 복호화를 통한 직접 검증과 복호화를 하지 않는 간접 검증으로 나눌 수 있다. 복호화를 통한 검증은 가장 확실한 검증 방법이지만 비밀 투표의 원칙을 훼손할 수 있기 때문에 전자 투표기의 검증 방법으로는 적절하지 않다. 따라서, 간접 검증 방법을 사용해야 하는데, 간접 검증 방법의 안전성은 전자 투표기에 대한 신뢰 확률로 나타낸다. 기존 전자 투표기에 대한 간접 검증 방식은 특수한 장비를 필요로 하거나 신뢰 확률이 높지 않으며 별도의 검증자를 신뢰해야 하는 등의 문제가 있었다. 본 논문에서는 특수한 장비나 검증자 없이 기존 방식 이상의 안전성을 갖는 전자 투표 영수증 발급 방식을 제안한다. 또한, 실제 구현 상의 문제점을 살펴보고 안전성은 유지하면서 사용자 편의성을 향상시킬 수 있는 개선 방안을 제시한다.

키워드 : 전자 투표, 영수증, 확률적 암호화, 믹스넷

I. 서 론

세계적으로 전자 투표 실시에 대해 관심이 고조된 지금, 우리나라에서도 전자 투표 실시를 적극 검토하고 있다[15]. 전자 투표를 실시할 경우 비용적인 측면에서 여러 장점이 있을 것으로 기대하고 있지만 한편으로는 안전성에 대한 의문이 제기되고 있는 실정이다[6]. 특히, 전자 투표기는 투표자의 선택을 기록, 저장하는 핵심적인 역할을 하고 있는 상황에서 전자 투표기를 신뢰할 수 있는지가 중요한 문제로 부각되고 있는데, 이는 전자 투표기의 동작을 검증하는 것이 현실적으로 매우 어렵기 때문이다. 즉, 투표 과정에서 투표기가 화면에 표시하는 내용과 실제로 기록하는 내용이 같지 않을 수 있

다는 근본적인 문제 제기로서 이를 해결하지 않으면 전자 투표를 실시하는 것이 불가능하다고 할 수 있다. 전자 투표기에 대한 신뢰성 문제를 제기한 R.Mercuri는 투표값에 대한 전자적인 기록과 병행하여 용지 등의 물리적인 매체에 암호화하지 않은 투표값을 기록하여 투표 기록 검증 및 재검표에 이용할 수 있도록 해야 한다고 주장하기도 하였다[8].

하지만, R.Mercuri의 방식은 현재의 종이 투표 방식과 차이가 없기 때문에 전자 투표 실시에 따른 장점을 기대하기 어렵다. 더욱이 빠른 개표를 위해 릴 형태의 종이 테이프에 암호화하지 않은 투표 결과를 순차적으로 기록한다면 이로 인해 투표의 익명성 원칙이 훼손될 수도 있다. 따라서, 투표 내용은 암호화된 형태로만 기록하고 암호화된 투표 내용이 정확함을 투표기가 증명하도록 하는 방안이 필요한데, 이를 해결하기 위해 연구되고 있는 것이 투표값 기록에 대한 영수증을 발급하는 것이다[9,10,12-14]. 물론 이 때 발급된 영수증을 통해 투표자는 자신의 투표가 정확하게 기록되었음을 높은 확률로 확신할 수 있지만 다른 사람에게 투표 내용을 증명하는 용도로는 사용될 수 없어야 한다.

2002년과 2004년 D.Chaum에 의해 제안된 영수증 발급 방식은 시각적 암호화(visual cryptography)[4]를 응용한 방식으로 투표자는 투표 후 직관적으로 기록된 내용을 검증할 수 있지만, 다른 사람에게 투표 내용을 증명하는 것은 불가능하도록 구성한 방식이다[12]. 즉, 전자 투표기는 특수한 프린터와 두 장의 투명한 용지를 이용하여 투표 결과를 출력하는데, 두 장이 겹쳐진 상태로는 투표 결과를 육안으로 확인할 수 있지만 용지를 서로 분리하면 각각의 출력물로는 투표 결과를 확인하는 것이 불가능하다. 투표자는 이 가운데 한 장을 임의로 선택하여 영수증으로 간직하고 나머지 한 장은 참관인이 보는 상태에서 폐기해야 한다. 이 방식의 경우 투표기가 투표자의 선택을 예측할 수 있다면 투표 기록을 조작하는 것이 가능하므로 안전성은 $\frac{1}{2}$ 이라고 할 수 있다. D.Chaum 방식의 단점은, 영수증 발급에 특수한 프린터와 용지를 필요로 한다는 점과 안전성이 $\frac{1}{2}$ 로 높지 않다는 점이다.

2003년 A.Neff에 의해 제안된 영수증 발급 방식은 코드북을 기반으로 하는 방식으로, 투표기는 투표자의 선택에 따라 사전에 저장된 코드북을 참조하여 암호화된 투표값을 기록한다[9]. 만약 코드북을 참조하여 기록하지 않고 임의로 다른 값을 기록할 경우 투표 기록을 조작할 수 있는데, 이를 막기 위해 동일한 코드북을 가지고 있는 감시자가 수시로 투표 과정을 거쳐 감시하도록 하고 있다. 이 방식의 안전성은 전체 투표자의 수를 l 이라고 하고, 감시 횟수를 c 라고 했을 때 $\frac{c}{l+c}$ 가 된다. A.Neff 방식의 단점은 투표기의 동작을 검증하기 위해 감시자를 신뢰해야 한다는 점과 감시자의 감시 횟수가 투표자의 수만큼 되었을 때 비로소 안전성이 $\frac{1}{2}$ 이 된다는 점 등이다.

2005년 D.Chaum은 현재의 종이 투표 방식과 유사한 영수증 발급 방식을 제안하였다[14]. 이 방식을 이용할 경우 투표 과정이 현재의 종이 투표 방식과 유사하기 때문에 투표자에 대한 별도의 교육이 필요 없고, 투표기도 투표값을 알지 못하는 장점이 있다. 하지만 암호문이 기록된 투표 용지를 사전에 투표자 수보다 훨씬 많이 제작해야 하고,

투표자는 투표 용지에 기록된 암호문이 정당한지를 투표하기 전 검증해야 하는데, 투표자가 직접 투표 용지를 검증하는 것은 불가능하기 때문에 투표 용지 검증을 담당하는 검증자를 신뢰해야 하는 문제가 있다. 또한, D.Chaum의 이전 방식과 마찬가지로 투표 용지를 분리하여 후보자 목록 부분을 폐기해야 하는데, 폐기 과정에서 비밀 투표의 원칙이 훼손될 수 있는 문제가 있다.

전자 투표 시스템에 대한 요구 사항은 크게 다음 두가지로 정리할 수 있다.

Req. 1) *cast-as-intended* 모든 투표자는 자신의 투표가 의도한 대로 기록되었음을 확인할 수 있어야 한다.

Req. 2) *counted-as-cast* 모든 투표자는 기록된 투표 결과로부터 개표 결과가 나왔음을 확인할 수 있어야 한다.

여기서 두 번째 조건의 경우는 암호화된 투표값을 저장한 이후의 개표 과정에 해당되는데, 믹스넷 등을 이용하여 누구나 검증할 수 있는 여러 방식[1,7,11]이 이미 제안되어 있으며 본 논문의 범위를 벗어난다. 이에 비해 첫 번째 조건은 전자 투표기에 대한 신뢰성의 문제로서 아직 해결해야 할 과제가 많은 상태이다.

본 논문에서는 첫번째 조건과 관련하여 전자 투표 시스템이 만족시켜야 할 추가적인 다음 두가지 요구 사항을 제시한다.

Req. 3) 투표소 내에서 투표자로 하여금 투표기를 포함한 어떠한 기기나 사람에 대해 신뢰를 요구해서는 안된다.

Req. 4) 개표 과정 이외에는 투표자의 투표값을 유추할 수 있는 어떠한 형태의 복호화도 불가능해야 한다.

투표기를 신뢰하지 못하는 상황에서 다른 기기나 사람에 대한 신뢰를 요구하는 것이 바람직하지 않음은 자명하기 때문에 세 번째 조건은 반드시 필요하다. 전자 투표 시스템 상에서 복호화 키는 안전성을 위해 하나 이상의 관리 기관만 소유하도록 되어 있고, 개표 과정에서의 복호화도 비밀 투표 원칙을 위해 투표 결과에 익명성을 부여한 후 이루어지는 것이 일반적이다. 만약 투표 기간 중 복호화가 가능하다면 매표나 중간 결과 노출로 인해 선거에 부정적인 영향을 미칠 수 있기 때문에 네 번째 조건 역시 반드시 필요하다.

2003년 제안된 A.Neff의 방식은 감시자를 신뢰해야 하는 문제가 있기 때문에 세 번째 조건을 만족시키지 못하며, 2005년 제안된 D.Chaum의 방식은 투표 용지의 검증을 위해 검증자를 신뢰해야 하고, 이 과정에서 투표값을 유추할 수 있는 연산이 사용되기 때문에 세번째와 네번째 조건을 만족시키지 못한다.

본 논문에서는 두 번째 조건이외에 첫 번째, 세 번째 그리고 네 번째 조건을 만족시킬 수 있는 전자 투표 영수증 발급 방식을 제안한다. 제안한 방식은 안전성이 최소 $\frac{1}{2}$ 이상으로 기존 방식 이상의 안전성을 가지며, 투표자가 직접 발급된 영수증을 검증할 수 있는 방식이다. 또한, 널리 알려진 cut-and-choose 또는 divide-and-choose 기법을 응용한 것으로서, 구현이 용이하며 D.Chaum의 방식과는 달리 특수한 프린터/용지를 필요로

하지 않고, 영수증의 일부를 안전하게 폐기해야 할 필요도 없으며, A.Neff의 방식처럼 감시자를 둘 필요도 없다. 제안한 방식과 기존 방식의 장단점을 요약하면 다음 (표 1)과 같다.

(표 1) 장단점 비교

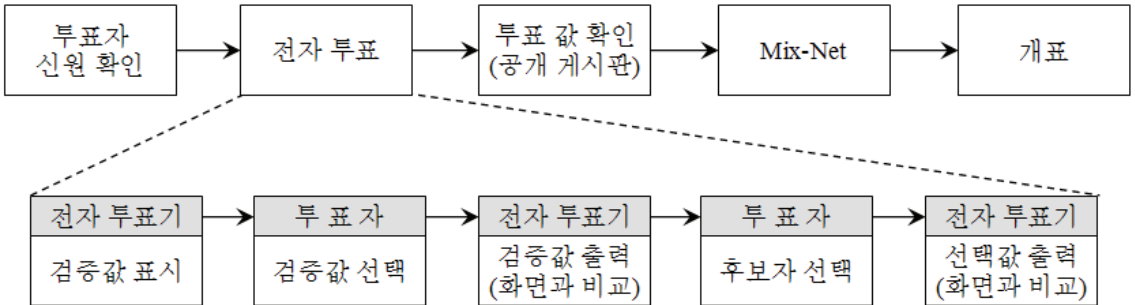
비교항목	A.Neff (2003)	D.Chaum(2004)	D.Chaum(2005)	제안한 방식
안전성 ¹⁾	$\frac{c}{l+c}$ ²⁾	$\frac{1}{2}$	$\frac{b-1}{b}$ ³⁾	$1 - \frac{1}{2^{n-1}}$ ⁴⁾
사전 준비	코드북 생성	없음	암호화된 투표 용지 제작	없음
추가 장비	없음	특수 프린터 및 용지	없음	없음
추가적인 신뢰 가정	검증자에 대한 신뢰 필요	없음	검증자에 대한 신뢰 필요	없음
투표자 본인이 직접 검증	불가능	가능	불가능	가능

- 1) 투표자 한 명이 전자 투표기를 신뢰할 수 있는 확률.
2) c 는 검증자의 검증 횟수, l 은 전체 투표자 수.
3) 투표 용지의 사전, 사후 검증을 제외하고 투표자 한 명이 b 장의 투표 용지에서 한 장을 선택할 경우.
4) n 은 후보자 수.

하지만 제안한 방식은 사용자에게 여러 번의 무작위 선택을 요구하며, 전자 투표기 화면과 영수증에 기록된 비교적 긴 문자열을 육안으로 비교해야 하는 문제가 있다. 본 논문에서는 이러한 문제를 해결하기 위해 무작위 선택에 대한 인터페이스 개선 방안과 안전성은 유지하면서 비교해야 할 문자열의 길이를 획기적으로 줄일 수 있는 방안을 함께 제시한다.

II. Cut-and-choose 기반 영수증 발급 방식

전자 투표의 전체 흐름과 그 가운데 제안하는 방식을 그림으로 나타내면 다음과 같다.



(그림 1) 전자 투표 전체 과정

1. 영수증 발급 과정

투표 과정에서 사용되는 파라미터 및 표기법은 다음과 같다.

- n : 후보자의 수
- $E(m, r)$: 평문 m 과 임의의 난수 r 에 대한 ElGamal 암호화 연산
- $E^{-1}(c)$: 암호문 c 에 대한 복호화 연산 ($E^{-1}(E(m, r)) = m$)

전자 투표기에서 사용하는 암호화 방식은 ElGamal 암호 방식과 같은 확률적 암호화 (probabilistic encryption)[2,3]가 필수적이다. 왜냐하면, 전자 투표의 경우 평문 공간의 크기가 극히 작아서 난수를 이용하는 확률적 암호화를 사용하지 않으면 특정 암호문에 대한 평문을 찾는 것이 매우 쉽기 때문이다.

이제 신원 확인 과정을 마친 투표자 i 가 n 명의 후보자에 대해 실시하는 전자 투표 과정을 자세히 설명하면 다음과 같다.

1. 투표기는 $j=1, \dots, n$ 를 암호화한 값 e'_j 와 e''_j 를 계산한다. (즉, $E^{-1}(e'_j) = E^{-1}(e''_j)$)
($e'_j, e''_j = (E(j, w'_j), E(j, w''_j))$) (w'_j, w''_j 는 투표기가 생성한 임의의 난수)
2. 투표자는 $j=1, \dots, n$ 에 대해 검증값으로 사용할 $e_j \in \{e'_j, e''_j\}$ 를 임의로 선택한다.
3. 투표기는 투표자가 선택한 n 개의 e_j 와 함께, e_j 를 생성할 때 사용한 난수 $w_j \in \{w'_j, w''_j\}$ 를 영수증에 출력한다.
4. 투표자는 출력된 e_j 가 투표기 화면에 표시된 것과 동일한지 검증한다.
5. 투표자는 검증을 마친 후, 원하는 후보자 v_i 를 선택한다. (단, $1 \leq v_i \leq n$)
6. 투표기는 e'_{v_i} 와 e''_{v_i} 가운데 2 단계에서 검증값으로 선택되지 않은 e'_{v_i} 또는 e''_{v_i} 을 투표값 $e^*_{v_i}$ 로 하여 영수증에 출력한다. ($e^*_{v_i} \in \{e'_{v_i}, e''_{v_i}\}$)
7. 투표자는 출력된 $e^*_{v_i}$ 가 투표기 화면에 표시된 것과 동일한지 검증한다.
8. 투표기는 투표자의 검증이 완료되면 $e^*_{v_i}$ 를 투표자 i 에 대한 투표 결과로 공개 게시판에 등록한다.

위의 과정을 마치면, 투표자는 n 개의 검증값 e_j 와 1개의 투표값 $e^*_{v_i}$ 가 표시된 영수증을 받게 되는데, 공개 게시판에 등록되어 있는 투표값과 영수증에 표시된 투표값이 일치하는지 검증한다. 그리고, 다음과 같이 n 개의 검증값 e_j 를 검증하여 $e^*_{v_i}$ 의 유효성을 확인한다.

$j=1, \dots, n$ 에 대해 $E(j, w_j) = e_j$ 인지 검증

n 개의 검증값 모두에 대해 위의 검증 과정을 확인하였으면 $e^*_{v_i}$ 가 유효함을 $1 - \frac{1}{2^{n-1}}$ 의 확률로 확신할 수 있다. 투표를 마치면 $e^*_{v_i}$ 는 투표자의 ID와 함께 읽기 전용 공개 게시판에 공개되기 때문에 투표자는 자신의 영수증에 기록된 $e^*_{v_i}$ 와 같은 값인지 비교해야 한다. 공개 게시판에 등록된 투표값에 대한 암호문은 개별 투표자의 검증을 마친 후 믹스넷을 거쳐 익명성을 만족시킨 후 개표에 사용된다. 믹스넷을 이용한 개표는 이

미 많은 연구를 통해 누구나 검증할 수 있는 안전한 방법이 제안되어 있다[1,7,11].

2. 투표자 관점에서 본 문제점

제안한 방식의 경우 투표자의 관점에서 본 문제점은 크게 두가지이다. 하나는 무작위 선택의 횟수가 후보자의 수인 n 회로서 매우 많다는 점이고, 다른 하나는 육안으로 비교해야 할 투표기 화면의 내용과 영수증에 출력된 내용이 너무 길다는 것이다.

일반적으로 사람이 무작위 선택에 취약하다는 것은 이미 알려진 사실이다. 따라서, 무작위 선택은 가능한 최소로 줄이는 것이 바람직하다. 시각적 암호화를 이용한 D.Chaum의 영수증 발급 방식의 경우 투표자의 무작위 선택은 1 회이다. 따라서, n 회를 1 회로 줄이는 방안이 필요하다.

영수증에 출력되는 검증값 비교는 보다 심각한 문제를 안고 있다. 만약 ElGamal 암호 방식에서 소수 p 가 1,024 비트라면, 암호문은 2,048 비트의 길이를 갖게 된다. 사람이 육안으로 짧은 시간 내에 n 개의 2,048 비트 문자열 쌍을 비교한다는 것은 사실상 불가능하다. 암호학적으로 안전한 해쉬 함수를 적용하여 해쉬값을 비교하는 방법을 생각할 수 있지만, 해쉬 함수의 출력도 160 비트 이상으로서 Base64 인코딩을 적용하여 사람이 인지하기 쉬운 알파벳으로 인코딩한다고 가정해도 28 자 이상의 알파벳을 비교해야 하기 때문에 무리가 있다. 아래 그림은 후보자가 4 명인 경우 검증값을 출력한 예로서, ElGamal 암호화 결과값에 SHA-1 해쉬 함수와 Base64 인코딩을 적용하였다. 이렇게 하여도 각 검증값은 28 자의 알파벳으로 구성되어 있어 짧은 시간 내에 정확하게 검증하는 것이 사실상 불가능하다. 아래 (그림 2)는 제안한 방식을 구현한 결과인데, (a)의 투표기 화면에서 모두 4 번의 무작위 이진 선택을 하였으며, 그 결과 출력된 (b)의 영수증과 (a) 화면을 비교하여 값이 일치하는지 검증해야 한다.

검증용 난수 선택

영수증 발급을 위해, 두개의 난수 "가"와 "나" 중에서 각각 하나씩 선택하세요.

	가	나
Num 1.	ixK32CE9h4CNbdh1YMBY5i6becA=	xvxQSNZT19/jN7oENvpSfyNu1p4=
Num 2.	G0hwvXh+rKgHDX8YG7XFH61ecU0=	LXk4UN+4FSXc00DwD51uVmLTu+Y=
Num 3.	oCbLtv1hKYTBGoRPPNueuhI8qrM=	74NnP13GyET3Q3BQzwfKJQY0efU=
Num 4.	QD9UnK2HBrboQMvYdVUBZJKEkM=	Fz++nZ/Lrm5aY0kVrY/FdNerEHY=

(a) 투표기 화면

비교값(암호화->해쉬값)		난수
Num 1.	ixK32CE9h4CNbdh1YMBY5i6becA=	pUa+VoijckwMNIXzyaBGRYRsUJhmk9+u9n5YcBVHa9GHwt83pPcty f5uPR6Mykk3BKoy2KhLKI8sucwW47NbM+yHk8X+szUP6DTyO2wHF3W Ktntwt2axOUwtJdf0w1uyHXHFZMgjEdtiSMU8AGZXULZRpkfeyjFmm 9rKUYgcUE=
Num 2.	LXk4UN+4FSXcO0DWD51uVmlTu+Y=	w5kwXoHuTyuw7XUHpy3qjUWPdhRDQY8whqpyAzGuC3y318Y1HadnEJ meUnXkUPLNP8612H32QaCidvfn7ESIhRGyC6CiSb+Hm83O54c+HdN2 LcPmOVRgchF5s8LJCqwtU4+iNoKmaV+8NAKjH3wEQF4yuqSofX3zcn SbHG7nEm0=
Num 3.	74NnP13GyET3Q3BQZwfkJQYOefU=	jnjK9H2EfYqNKA1hJQiEBESjsPimCCSbUbA31Cqq6LQf4VRsezGnAG UDATYKiCi6XGhJDBKVMARTV20kIcbqrDKhV6o7WNvuIHvu1IYqujnA 58mrTZsoXioFfzDwqwodFWBZ2DNOK9pCUIL615H4Z63DwDP22zxLA3 VYGPah+lk=
Num 4.	QD9UnK2H8rboQMvYdVUBZJKEnkM=	xrIXaAowtCg+1Jm6TqfFkPQXOnPRrt8f1s10+wwUNYKFSCYZ4fKRwy kgFO5LAMEmcBxUd87EjpiRsk+Q7XFqgXBR50y32bCRw8w+0j/kTppg eN2EqLsFI+zGLPM152j2vRb0hxCi7+q0wTzHtfaZrxuwY8h5Kky+QQ HKKtXkrak=

(b) 출력된 영수증

(그림 2) 후보자가 4 명인 경우 검증값을 선택한 화면과
그에 따라 출력된 영수증

III. 안전성을 저해하지 않는 사용자 편의성 향상 기법

1. 비교 문자열의 길이 최소화

투표기 화면과 영수증 출력 내용에서 비교해야 할 문자열의 길이가 길다는 것은 매우 심각한 문제이다. 이를 해결할 수 있는 방안으로 보다 작은 길이의 출력을 갖는 축약 함수를 이용하는 것을 제안한다. 즉, 160 비트의 길이를 갖는 SHA-1의 해쉬값을 8~10 비트로 다시 축약하는 것이다. 다만, 이 경우 축약된 값이 서로 충돌할 가능성이 매우 높기 때문에 안전성에 문제가 생길 수 있다.

본 논문에서는 이 문제를 해결하기 위해 투표기 화면에 검증값을 표시하기 전, 전체 검증값을 무작위 순서로 영수증에 먼저 출력하도록 하였다. 무작위 순서로 출력하지 않으면 투표자가 자신의 투표값을 증명할 수 있기 때문에 반드시 무작위 순서로 출력해야 한다. 무작위로 출력했음을 확인하는 가장 간단한 방법은 검증값을 크기 순서대로 정렬하여 출력하는 것이다.

소수 p 가 1,024 비트인 ElGamal 암호 방식을 이용하는 수정된 투표 과정은 다음과 같다. 단, $H(\cdot)$ 는 축약 함수로서 사전에 공개된다.

1. 투표기는 $j=1, \dots, n$ 를 ElGamal 암호화한 값 e'_j 와 e''_j 를 계산한다.
 $(e'_j, e''_j) = (E(j, w'_j), E(j, w''_j))$ (w'_j, w''_j 는 투표기가 생성한 임의의 난수)
2. 투표기는 모든 e'_{j1} 과 e''_{j1} 을 크기 순서대로 정렬하여 영수증에 출력한다.
3. 투표기는 축약 함수 $H(\cdot)$ 를 이용하여 2,048 비트 문자열인 e'_j 와 e''_j 를 각각 8 비트로 축약하여 화면에 표시한다.

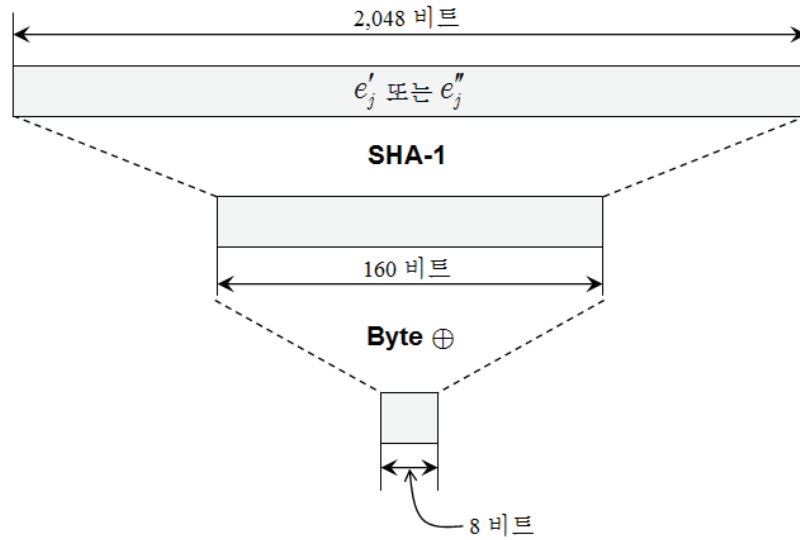
$$\overline{e_j'} = H(e_j'), \overline{e_j''} = H(e_j'')$$

4. 투표자는 $j=1, \dots, n$ 에 대해 검증값으로 사용할 $\overline{e_j} \in \{\overline{e_j'}, \overline{e_j''}\}$ 를 임의로 선택한다.
5. 투표기는 투표자가 선택한 n 개의 $\overline{e_j}$ 와 그에 대응되는 e_j 를 생성할 때 사용한 난수 $w_j \in \{w_j', w_j''\}$ 를 영수증에 출력한다.
6. 투표자는 출력된 $\overline{e_j}$ 가 투표기 화면에 표시된 것과 동일한지 검증한다.
7. 투표자는 검증을 마친 후, 원하는 후보자 v_i 를 선택한다. (단, $1 \leq v_i \leq n$)
8. 투표기는 $\overline{e_{v_i}'}$ 와 $\overline{e_{v_i}''}$ 가운데 2 단계에서 검증값으로 선택되지 않은 $\overline{e_{v_i}'}$ 또는 $\overline{e_{v_i}''}$ 를 투표값 $\overline{e_{v_i}^*}$ 로 하여 영수증에 출력한다. ($\overline{e_{v_i}^*} \in \{\overline{e_{v_i}'}, \overline{e_{v_i}''}\}$)
9. 투표자는 출력된 $\overline{e_{v_i}^*}$ 가 투표기 화면에 표시된 것과 동일한지 검증한다.
10. 투표기는 투표자의 검증이 완료되면 $\overline{e_{v_i}^*}$ 와 $e_{v_i}^*$ 를 투표자 i 에 대한 투표 결과로 공개 게시판에 등록한다.

투표자는 투표소를 나와서 $\overline{e_{v_i}^*}$ 의 유효성을 검증해야 하는데, 이 때 별도의 공인 검증기관을 이용하거나, 투표자가 직접 소프트웨어 등을 개발하여 검증할 수 있다. 위의 과정 2에서 출력된 $2n$ 개의 검증값 집합을 S 라고 할 때 검증 과정은 다음과 같다.

1. 모든 $j=1, \dots, n$ 에 대해, 영수증에 출력된 w_j 를 이용하여 $e_j = E(j, w_j)$ 를 계산하고, $H(e_j) = \overline{e_j}$ 인지 검증하고, $e_j \in S$ 인지 검증한다.
2. 공개 게시판에 등록된 $\overline{e_{v_i}^*}$ 와 $e_{v_i}^*$ 를 이용하여 $\overline{e_{v_i}^*} = H(e_{v_i}^*)$ 인지 검증하고, $\overline{e_{v_i}^*}$ 가 영수증에 출력된 것과 일치하는지 검증하며, $e_{v_i}^* \in S$ 인지 검증한다.

2,048 비트 입력을 받아 8 비트 출력을 내는 축약 함수 $H(\cdot)$ 는 (그림 3)과 같이 구성하였다. 여기서 Byte \oplus 연산은 전체 바이트에 대한 X-OR 연산을 의미한다. 예를 들어, SHA-1 해쉬 출력 결과가 “64E0D79D7AC8DD90CF7FDC6BFA0B4F0978C7B933”이었다면 이에 대한 한 바이트 축약 결과는 $64 \oplus E0 \oplus \dots \oplus 33 = B4$ 가 된다.



(그림 3) 축약 함수 $H(\cdot)$

축약된 8 비트는 4 비트씩 분리하여 보다 비교하기 쉽도록 다음 표에 의해 두 글자로 대체된다. 예를 들어, B4(10110100)는 “타마”로 대체하여 화면과 영수증에 출력한다.

(표 2) 4 비트의 문자 대체표

비트열	대체 문자	비트열	대체 문자
0 0 0 0	가	1 0 0 0	자
0 0 0 1	나	1 0 0 1	차
0 0 1 0	다	1 0 1 0	카
0 0 1 1	라	1 0 1 1	타
0 1 0 0	마	1 1 0 0	파
0 1 0 1	바	1 1 0 1	하
0 1 1 0	사	1 1 1 0	갑
0 1 1 1	아	1 1 1 1	을

만약, 영문자로 대체한다면 SHA-1의 출력을 10 비트씩 분리한 후 전체를 X-OR 연산하여 10 비트로 축약하고 (표 3)의 대체표를 이용하면 된다.

(표 3) 5 비트의 문자 대체표

비트열	대체 문자	비트열	대체 문자	비트열	대체 문자	비트열	대체 문자
00000	A	01000	I	10000	Q	11000	Y
00001	B	01001	J	10001	R	11001	Z
00010	C	01010	K	10010	S	11010	1
00011	D	01011	L	10011	T	11011	2
00100	E	01100	M	10100	U	11100	3
00101	F	01101	N	10101	V	11101	4
00110	G	01110	O	10110	W	11110	5
00111	H	01111	P	10111	X	11111	6

다음 그림은 위에서 설명한 축약 함수를 적용하여 구현한 예로서, 왼쪽은 투표기 화면에 표시된 내용이고, 오른쪽은 이에 대응되는 영수증 출력 결과이다.

<투표기 화면>

선택값 : 11	
바타	가하
을마	갑라
차갑	하차
카라	마파

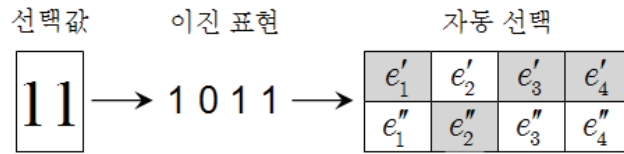
<출력된 영수증>

선택값 : 11		난수
	가하	mX3GxvLphC+bpBhLd5705H14/Iiv/wdTF4xi7rQ+c a/qAMztEC2FkrmqneWd18/nQ1Gdxmn2jEzt+ksBfp O6nDcdaMbzzj/heLjwu0LMxWHS+IzVuMgB/5idYN9 X8B04bsfqvMYjwRd09kuSBAbfuiQ1j0S5FXxLPjDI Uq00UeU=
을마		351pjE7euQp/7WTSron8hWOUXiSJmBdxJXKiTfst WQa12cMur5yhQhIyXr73ftthBspF7k3/xUR4wg6M+ tFwsdkJKH5ZulFpubV15tt+ZVe1A7wxYWyAVrNCO4 hfLxQ8X6P+eWDgJkF5zuYr7PtK39aFaYy+jDgdfIL HJSqB1C=
	하차	566/95cZVAnv/xzJmZbWg9kEmShdYIzN50wKp+HiJ umsnxYt7ndJUTjZTuh81lyGAqrUAp19lw/GeSvTSy nj5PqobNenBYQ/M75x/urD/12Y+BLnZSF7BL+RDPn FKeWPCFG/FmwMarAYkOHmlPPKRVDip8ThstIjH2if ldhbK0U=
	마파	cRodJ4v9OCnn4suRDy8WtuWYEHPh61SnUa5S1D9Pq RpPcXVAolm7dxd2F75nsZZoPRVq+LGr6nawYbxqF N8yxKU4Draqd8DRRYBULcZhSpOkgwKAP5GfOCajV/ cvCbntS/aOQhQNFjxxDXd/wSyASk1qMRcaQkQ9/SS 107q100=

(그림 4) 축약 함수를 적용하여 구현한 예

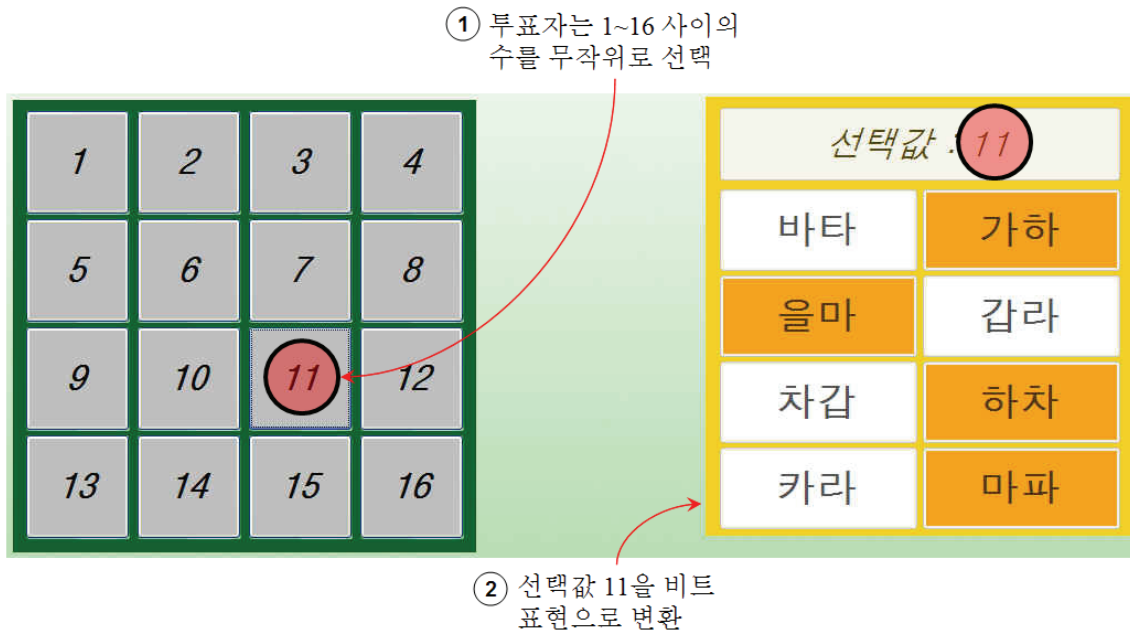
2. 무작위 선택 횟수 최소화

제안한 방식에서의 무작위 선택은 모두 이진(binary) 선택이다. 따라서, n 번의 이진 선택은 $[0, 2^n - 1]$ 사이의 수를 한 번 선택하는 것과 같다. 다음 그림은 후보자가 4 명인 경우 한 번의 난수 선택을 4 번의 이진 선택으로 변환하는 예이다.



(그림 5) 4 번의 무작위 이진 선택을 [1,16] 사이의 난수 선택으로 변환한 예

따라서, (그림 2-a)의 투표기 화면에 다음의 (그림 6)과 같은 인터페이스를 추가하여 한 번의 난수 선택으로 대체할 수 있다. 예를 들면, 아래와 같이 1~16 사이의 수를 표시하고 투표자는 이 가운데 하나를 임의로 선택하도록 한다. 투표자가 11을 선택했다면, 11에 대한 비트 표현인 1011을 이용하여 검증값을 자동 선택하도록 할 수 있다 (단, 16은 0을 선택한 것과 같다).

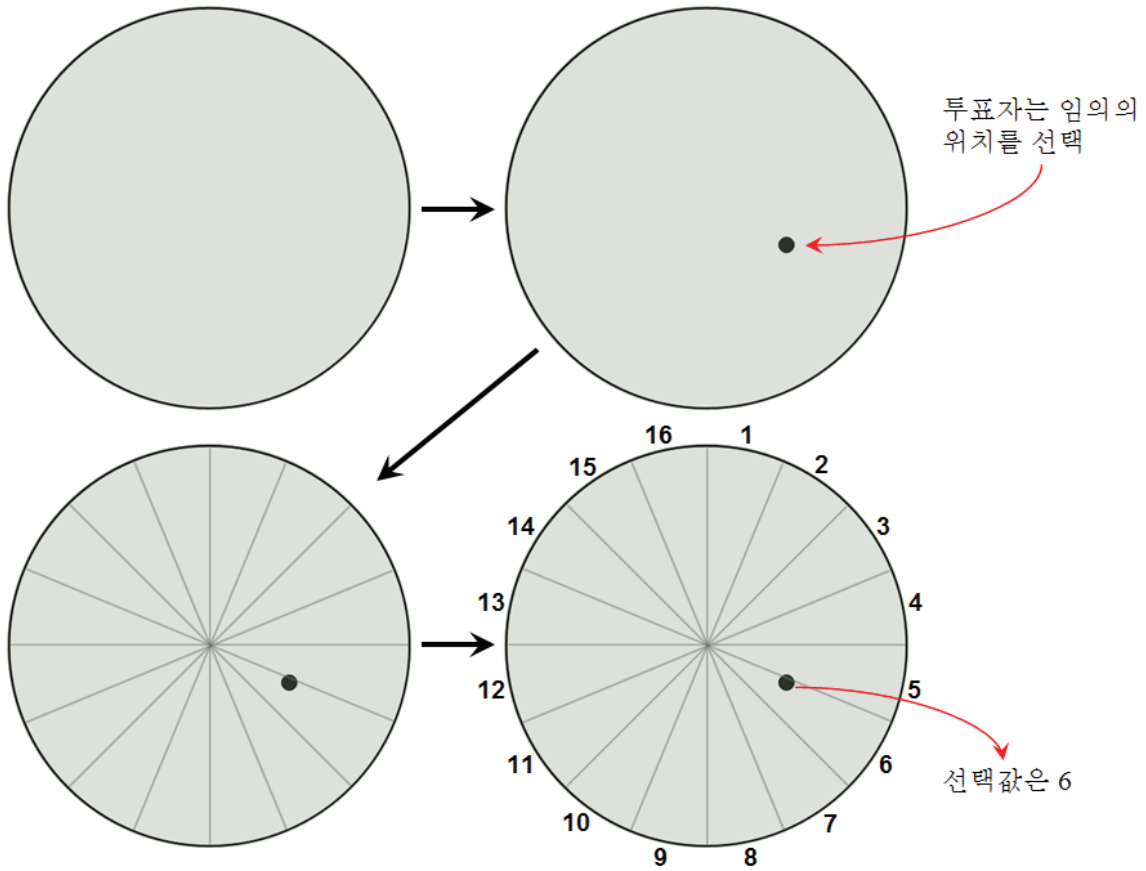


(그림 6) 4 번의 이진 선택을 한 번의 숫자 선택으로 변환한 인터페이스의 예

이 경우 문제가 될 수 있는 것은 1~16 사이에서 임의로 숫자를 선택하도록 하는 것이 난수 발생기를 통해 1~16 사이의 난수를 발생한 것과 동일한 분포를 갖는가 하는 점이다. 왜냐하면 사람들이 위의 (그림 6)과 같은 인터페이스를 대했을 때 심리학적으로 선호하는 위치나 숫자가 거의 비슷하다면 이는 전자 투표기의 부정 확률을 높일 수 있기 때문이다.

또 다른 인터페이스 구성으로 생각할 수 있는 것이 아래 (그림 7)과 같은 원형 선택이다. 투표자는 원형 선택 내에서 임의의 위치를 선택하면 되고, 선택된 지점은 계산을 거쳐 숫자로 변환된다. (그림 7)은 투표자의 선택에 따라 1~16 사이의 숫자로 변환하는

과정을 나타낸다.



(그림 7) 원형 인터페이스를 구성한 예

원형 인터페이스가 (그림 6)의 인터페이스에 비해 갖는 장점은 보다 많은 후보자를 수용할 수 있다는 것이다. 이와 함께 투표자의 선택에 따라 생성된 숫자열은 난수열과 유사한 확률 분포를 가질 것으로 기대되지만, 이는 많은 사람이 심리학적으로 원형 선택판의 가운데를 선호할 것이라는 가정이 있어야 한다. 따라서, 무작위 선택을 위한 효율적인 인터페이스는 심리학적인 면까지 고려하여 선택의 결과가 최대한 난수와 유사하도록 구성해야 한다.

IV. 안전성 분석 및 효율성 비교

1. 안전성 분석

제 2 장에서 제안한 방식의 안전성은 후보자의 수에 의해 결정되며, 후보자가 n 명이라고 했을 때 $1 - 2^{-(n-1)}$ 의 안전성을 갖는다. 왜냐하면, 투표기가 발각되지 않고 투표값을 조작하기 위해서는 투표자의 이전 선택을 예측해야 하기 때문이다. 투표기가 n 개의 이전 선택을 예측할 확률은 2^{-n} 이지만, 투표자의 선택과 무관하게 투표값을 조작하

더라도 조작된 결과는 $[1, n]$ 범위에 있으므로 $n-1$ 개의 이진 선택만 예측하면 된다. 따라서, 안전성은 $1-2^{-n}$ 이 아닌 $1-2^{-(n-1)}$ 이며, 최소값은 2^{-1} 이다. 예를 들어, $n=4$ 이며 투표자의 선택과는 무관하게 투표값을 1 번으로 조작하는 경우를 보자(그림 8. 참조). 만약 투표자의 검증값 선택을 1011로 예측했다면 해당 위치의 값만 정확하게 1부터 4를 암호화하고 나머지는 모두 1을 암호화하면 된다. 하지만, 조작하고자 하는 1 번의 경우는 어느 쪽을 선택하더라도 1을 암호화하기 때문에 예측해야 하는 이진 선택은 4 개가 아니라 3 개가 되며, 따라서 예측 확률은 2^{-4} 이 아닌 2^{-3} 이다.

$e''_1 = E(1, w_1)$	$e'_1 = E(1, w_1)$
$e''_2 = E(2, w_2)$	$e'_1 = E(1, w_2)$
$e''_3 = E(1, w_3)$	$e'_3 = E(3, w_3)$
$e'_4 = E(1, w_4)$	$e'_4 = E(4, w_4)$

(그림 8) 투표자의 선택을 1011로 예측한 경우 화면 표시

일반적으로 전자 투표 영수증 발급 방식의 안전성은 투표자 한 명이 전자 투표기를 신뢰할 수 있는 확률로 나타내는데, 기존 방식과 제안한 방식의 안전성을 비교하면 아래 (표 3)과 같다.

(표 3) 안전성 비교

A.Neff (2003) ¹⁾	D.Chaum (2004)	D.Chaum (2005) ²⁾	제안한 방식 ³⁾
$\frac{c}{l+c}$	$\frac{1}{2}$	$\frac{b-1}{b}$	$1 - \frac{1}{2^{n-1}}$

1) c 는 검증자의 검증 횟수, l 은 전체 투표자 수.

2) 투표 용지의 사전, 사후 검증을 제외하고 투표자 한 명이 b 장의 투표 용지에서 한 장을 선택한 경우.

3) n 은 후보자 수.

2004년 D.Chaum이 제안한 방식의 안전성은 $\frac{1}{2}$ 이며, A.Neff 방식은 검증 횟수와 투표자 수에 따라 결정되는데, 안전성이 $\frac{1}{2}$ 이 되기 위해서는 검증 횟수가 투표자 수만큼 많아야 하기 때문에 비효율적이다. D.Chaum의 2005년 방식의 안전성은 투표 용지의 사전, 사후 검증을 제외하고 투표자가 b 장의 투표 용지에서 한 장을 선택하여 투표한다고 가정했을 때 $\frac{b-1}{b}$ 이 된다. 즉, 두 장의 용지에서 하나를 선택한다면 $\frac{1}{2}$ 이 되는데, 사전에 제작해야 하는 투표 용지의 수가 전체유권자수 $\times b$ 장 이상이 되어야 하기 때문에 b 를 크게 하는데는 한계가 있다.

이에 비해 제안한 방식의 안전성은 후보자의 수에 의해 결정되기 때문에 안전성은 최소한 $\frac{1}{2}$ 이상이다. 하지만, $n=2$ 인 경우에도 후보자 한 명에 대해 생성하는 암호문을 2 개에서 t 개로 확장한다면 추가적인 부담없이 $1-t^{-1}$ 으로 신뢰 확률을 높일 수 있다.

또한 A.Neff 방식과 2005년 D.Chaum 방식의 경우 투표기 검증이나 투표 용지 검증을

위해 별도의 검증자나 검증기를 신뢰해야 하는 문제가 있다.

이제 제안한 방식에 축약 함수 $H(\cdot)$ 를 적용한 경우의 안전성에 대해 설명하도록 한다. 2,048 비트에 이르는 n 개의 문자열을 짧은 시간 내에 육안으로 비교하는 것은 사실상 불가능하기 때문에 축약 함수를 사용하는 것은 불가피하다. 축약 함수를 사용할 경우 고려해야 할 것은 서로 다른 j, i 가 주어졌을 때 투표기가 다음 식 (1)을 만족하는 w, w' 을 효율적으로 찾는 것이 불가능해야 한다는 것이다.

$$H(E(j, w)) = H(E(i, w')) \quad (j \neq i) \quad (1)$$

해쉬 함수의 경우는 160 비트 이상의 충분히 긴 출력과 충돌 회피성으로 인해 축약 함수로 사용할 수 있지만, 8 비트로 축약한다면 안전성에 문제가 발생할 수 있다. 하지만 제 3 장에서 제안한 방식과 같이 투표자가 선택하기 전 모든 암호문을 영수증에 출력하도록 하면 이러한 문제를 해결할 수 있다. 단, 영수증에 $2n$ 개의 검증값을 순서대로 출력한다면 투표자는 자신의 투표값을 증명할 수 있게 된다. 따라서, $2n$ 개의 검증값은 무작위 순서로 출력해야 한다.

2. 효율성 비교

(표 4)는 2003년 제안된 A.Neff 방식, 2004년 및 2005년 제안된 D.Chaum 방식과 제안한 방식의 효율성을 비교한 것이다.

(표 4) 효율성 비교

비교항목	A.Neff (2003)	D.Chaum(2004)	D.Chaum(2005)	제안한 방식
사전 준비	코드북 생성	없음	암호화된 투표 용지	없음
특수 장비	없음	프린터 및 용지	없음 ¹⁾	없음
검증자 ²⁾	필요	없음	필요	없음
투표 절차 친숙도 ³⁾	낮음	중간	높음	낮음
투표소 외부에서 검증	불가능	불가능	불가능	가능
대리자를 통한 검증 ⁴⁾	예	아니오	예	아니오

- 1) 투표 용지를 읽기 위한 스캐너가 필요하지만, 스캐너가 전자 투표기를 대신한다.
- 2) 투표소 내에서 전자 투표기를 제외하고 투표자가 신뢰해야 하는 기기 또는 사람을 말한다.
- 3) 2005년 D.Chaum 방식은 현재의 종이 투표 방식과 같은 형태의 투표 용지에 물리적으로 기표하는 방식이다. 따라서, 투표자 친숙도가 가장 높고 투표 방법에 대한 교육의 필요성이 가장 낮다.
- 4) 투표소 밖이나 내부에서 투표자 본인이 아닌 대리인 또는 기기를 통해 검증해야 하는 것을 말하며, 이러한 대리인이나 기기는 신뢰할 수 있다는 가정이 필요하다.

효율성 비교 결과를 보면, 제안한 방식은 사전 준비, 특수 장비, 검증자, 투표소 외부 검증 및 대리자 검증에서 장점을 갖는 반면, 2005년 D.Chaum 방식은 투표 절차 친숙도에서, 2004년 D.Chaum 방식은 대리자 검증 항목에서 장점을 갖는다. 따라서, 제안한 방식이 안전성 및 효율성에서 기존 방식에 비해 우수하다고 할 수 있다. 다만, 투표 절차 친숙도 측면에서는 기존 방식보다 낮은 것으로 평가할 수 있는데, 이는 실제로 구현하여 일반인에게 시연한 결과와 크게 다르지 않다. 전자 투표기 시제품을 제작하여 일반

인을 대상으로 시연한 결과 검증값을 왜 선택해야 하는지 모르는 경우가 많았고, 검증값 선택과 투표값 선택을 혼동하는 경우도 많았다. 물론 새로운 시스템에 대한 교육이 불가피한 측면이 있지만, 향후 보다 직관적인 사용자 인터페이스 구성에 관한 연구도 함께 진행되어야 할 것이다.

V. 결 론

전세계적으로 전자 투표 실시에 따른 장단점 분석과 안전성에 대한 우려가 높은 것이 현실이다. 하지만, 안전성 문제만 해결된다면 전자 투표 실시가 현재의 종이 투표 방식이 갖는 한계를 극복할 것이라는 점에는 대체로 공감하고 있다. 안전성 문제의 핵심은 전자 투표기에 대한 신뢰성 문제이고, 영수증 발급은 이를 해결할 수 있는 최선의 방안이다.

본 논문에서는 기존 영수증 발급 방식이 갖는 문제점과 한계를 해결할 수 있는 새로운 영수증 발급 방식을 제안하였다. 잘 알려진 cut-and-choose 방법을 응용한 제안 방식은 투표소 내의 어떠한 기기나 사람에 대한 신뢰도 요구하지 않는다는 장점이 있으며, 안전성 측면에서도 기존 방식보다 뛰어나다고 할 수 있다. 또한, 본 논문에서는 제안한 방식의 구현상 문제점을 살펴보고 이를 해결하여 보다 사용자 편의성을 높인 개선 방식에 대해 설명하였다. 향후에는 효율성 비교에서 지적했듯이 투표 절차를 보다 직관적이고 알기 쉽게 표현할 수 있는 사용자 인터페이스에 대한 연구가 진행되어야 할 것이다.

참 고 문 헌

- [1] D.Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms", *Comm. of the ACM*, vol. 24, no. 2, pages 84–88, Feb. 1981.
- [2] S.Goldwasser and S.Micali, "Probabilistic Encryption", *Journal of Computer System Sciences(JCSS)*, vol. 28, no. 2, pages 270–299, Apr. 1984.
- [3] T.ElGamal, "A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", *IEEE Trans. on Information Theory*, vol. IT-31, no. 4, pages 469–472, 1985.
- [4] M.Naor and A.Shamir, "Visual Cryptography", *Proc. of Advances in Cryptology (Eurocrypt'94)*, LNCS 950, pages 1–12, 1995.
- [5] P.Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes", *Proc. of Advances in Cryptology(Eurocrypt'99)*, LNCS 1592, pages 223–238, 1999.
- [6] R.Mercuri, "Rebecca Mercuri's Statement on Electronic Voting," <http://www.notablessoftware.com/RMstatement.html>, 2001.

- [7] C.A.Neff, “A Verifiable Secret Shuffle and Its Application to E-Voting”, *Proc. of the 8th ACM Conference on Computers and Communications Security(CCS-8)*, pages 116–125, 2001.
- [8] R.Mercuri, “A Better Ballot Box?”, *IEEE Spectrum Online*, pages 46–50, Oct. 2002.
- [9] C.A.Neff and J.Adler, “Verifiable e-Voting: Indisputable Electronic Elections at Polling Places”, http://www.votehere.net/vhti/documentation/VH_VHTi_WhitePaper.pdf, VoteHere Inc., 2003.
- [10] D.Chaum, P.Y.A.Ryan, and S.Schneider, “A Practical, Voter-Verifiable Election Scheme”, *Technical Report CS-TR-880*, University of Newcastle upon Tyne, 2004.
- [11] P.Golle, M.Jakobsson, A.Juels, and P.Syverson, “Universal Re-encryption for Mixnets”, *CT-RSA 2004*, LNCS 2964, pages 163–178, 2004.
- [12] D.Chaum, “Secret-Ballot Receipts: True Voter-Verifiable Elections”, *IEEE Security and Privacy Magazine*, vol. 2, no. 1, pages 38–47, Jan. 2004.
- [13] D.Evans and N.paul, “Election Security: Perception and Reality”, *IEEE Security and Privacy Magazine*, vol. 2, no. 1, pages 24–31, Jan. 2004.
- [14] D.Chaum, P.Y.A.Ryan, and S.Schneider, “A Practical Voter-Verifiable Election Scheme”, *Proc. of 10th European Symposium on Research in Computer Security(ESORICS2005)*, LNCS 3679, pages 118–139, 2005.
- [15] “전자선거추진협의회 공식 출범”, <http://www.nec.go.kr/notice/report/>, 중앙선거관리위원회, 2006.