情報保護學會論文誌
第12券 第6號, 2002. 12

효율적인 sealed-bid 경매 프로토콜

신 상 욱*, 류 희 수*

An efficient sealed-bid auction protocol

Sang Uk Shin*, Heuisu Ryu**

요 약

본 논문에서는 두 서버 S와 A를 가진 효율적이고 안전한 sealed-bid 경매 프로토콜을 제안한다. 제안된 기법은 Crescenzo-Ostrovskey-Rajagopalan의 Conditional Oblivious Transfer 프로토콜에서 사용된 기술을 이용하며, 서버 A는 신뢰되는 제3자(third party)가 아니고 서버 S와 공모하지 않는다고 가정된다. 이러한 가정하에서 제안된 경매 프로토콜은 어느 개체도 경매에 관한 어떠한 정보도 얻지 못하고 경매의 결과는 정확하다는 것을 보장한다. 또한 제안된 기법은 기제안된 Naor-Pinkas-Sumner의 기법보다 훨씬 적은 계산량을 요구하며 비슷한 통신 오버헤드를 가진다.

ABSTRACT

In this paper, we propose an efficient and secure sealed-bid auction protocol with two servers, a seller S and a third party A. The proposed scheme uses the idea of the conditional oblivious transfer protocol proposed by Crescenzo-Ostrovskey-Rajagopalan. A server A is not a trusted third party, but is assumed not to collude with a server S. In the proposed auction protocol, barring collusion between A and S, neither party gains any information about the bids, and moreover, the outcome of the auction will be correct. And the computational complexity of our auction protocol is considerably efficient and the communication overhead is similar to that of Naor-Pinkas-Sumner's scheme.

Keyword : *secure auction protocol, sealed-bid, oblivious transfer, conditional oblivious transfer*

Ⅰ. 서 론

경매(auction)는 비즈니스 세계에서 많은 양의 거래를 협상하기 위한 규범들 중의 한가지이지만, 소비자 판매 또는 소량의 구매는 경매 이용으로 인한 오버헤드가 높기 때문에 보통 고정된 가격으로 거래된다. 하지만 인터넷상의 새로운 경제 형태는 소규모 거래에 경매의 적용을 가능하게 한다.

일반적으로 경매 형태에는 English open cry, Dutch open cry, sealed-bid, Vickrey 경매와 같은 다양한 형태의 기법들이 있으며, 이들 경매 기법들은 각기 다른 목적을 달성하기 위해 다양한 상황에서 적용되고 있다. 그 중에서 sealed-bid 경매 기법은 입찰자(bidder)와 경매인(auctioneer)의 공모(collusion)로부터 개별적인 입찰자들의 이익을 보호하기 위해 보편적으로 사용되는 경매 형태이다. sealed-bid 경매는 광고된 상품에 대해 비밀 입찰가(bid)가 제출되고, 입찰 기간이 종료되면 입찰가를 열어 공개된 경매 규칙에 의해 승리자가 결정되는 기법으로, sealed-bid 경매의 기본적인 안전성 요구 사항은 다음과 같다.

- 입찰가의 비밀성(secrecy of bids) : 승리자의 입찰가를 제외한 어떤 입찰가도 노출되지 말아야 한다.

* 한국전자통신연구원 정보보호연구본부({shinsu, hsryu}@etri.re.kr)

- 검증 가능성(verifiability) : 누구라도 경매의 결과를 검증할 수 있어야 한다.
- 부인 방지(non-repudiation) : 입찰자는 자신의 입찰을 부인할 수 없어야 한다.
- 익명성(anonymity) : 승리자를 제외한 입찰자의 신분이 노출되지 말아야 하며, 입찰자와 입찰가의 관계가 알려지지 말아야 한다.

경매 프로토콜에 관한 초기 연구 중의 하나는 Franklin-Reiter[6]의 프로토콜이다. 이 기법은 프로토콜이 종료될 때까지만 입찰의 기밀성이 보장된다는 점에서 완전한 프라이버시(privacy)를 제공하지 않는다. 그리고 Harkavy-Tygar-Kikuchi[8]의 기법은 완전한 프라이버시를 제공하지만 입찰자의 과도한 참여(intensive bidder involvement)를 수반하며 다른 경매 형태로 쉽게 적용될 수 없다. Cachin[2]의 프로토콜은 두 개의 서버를 가지며 입찰자들간의 통신을 요구하고, Sako[13]의 기법은 privacy-preserving Dutch-style 경매 프로토콜로 입찰자가 자신의 입찰가를 공개하는 형태이며 매우 높은 계산량을 가지고 다른 형태의 경매로 확장이 불가능하다. Jakobsson-Juels[9] 프로토콜은 비트 단위 조작을 수반하는 함수(대표적인 예 : 경매)에 대해 general secure multi-party computation을 효율적으로 하는 것을 목적으로 하고, Baudron-Stern[1]의 기법은 하나의 서버만을 가지며 서버와 경매인간에 공모가 없다는 조건 하에 안전성을 보장하지만 다소 많은 계산량을 가진다.

최근에 Naor-Pinkas-Sumner[11]에 의해 제안되고 구현된 경매 기법은 분산된 신뢰를 가진 실제적인 sealed-bid 경매 프로토콜 연구에 상당한 진전을 가져왔다. 이 기법은 경매 issuer라는 제3자(third party) A를 가지며, 판매자 S와 제3자 A가 공모하지 않는다면 불필요한 정보는 노출되지 않는다. 이 기법은 Yao[14]의 secure two-party computation 모델에 기반하며, A는 암호화된 서킷(garbled circuit)을 구성하여 S에게 전달하며, S가 서킷을 계산하는 것을 돕는다. 서킷은 모든 가능한 안전성 요구 사항을 만족하도록 구성될 수 있다. 이 기법의 목적은 두 서버로 신뢰를 분산시키는 것이다. 하지만 이 기법의 단점은 A와 S간의 서킷 전송이 많은 통신 복잡도를 함축하며 부정한 제3자가 cut-and-choose 기법에 의해서만 검출될 수 있다는 것으로 이것은 프로토콜에 심각한 오버헤드를 초래한다. 또다른 단점은 서킷이 입찰자의 최대 수에 의존하므로 판매자는 경매 수행 전에 비교적 정확하게 입찰자의 수를 추정해야 한다는 것이다. 최근에 발견된 Naor-Pinkas-Sumner 기법의 안전성에서의 심각한 취약점은 A가 입찰가의 임의의 비트를 수정할 수 있다는 것이고, 이 취약점을 개선한 기법이 Juels-Szydlo[10]에 의해 제안되어졌다.

본 논문에서는 두 서버 S와 A를 가진 안전한 sealed-bid 경매 프로토콜을 제안한다. 여기서 서버 S는 물품을 판매하기 위해 경매를 구성하는 개체이고, 서버 A는 다수의 S에 의해 수행되는 많은 경매를 제공할 수 있는 서비스 제공업자이다. 제안된 경매 프로토콜은 Crescenzo-Ostrovskey-Rajagopalan[4]의 Conditional Oblivious Transfer(COT) 프로토콜에 사용된 기술을 이용한다. COT 프로토콜은 추가적인 비밀 입력 d(전송자의 비밀)와 t(수신자의 비밀)에 대해 공유된 predicate가 만족되면 전송자가 수신자에게 비트 b를 전송하는 프로토콜로, 본 논문에서는 COT 프로토콜에 사용된 기술을 이용하여 기존의 경매 기법들에 비해 매우 낮은 계산 복잡도를 가지며 유사한 통신 오버헤드를 가지는 효율적인 경매 프로토콜을 구성한다. 제안된 프로토콜에서 서버 A는 신뢰되는 제3자가 아니고 서버 S와 공모하지 않는다고 가정한다. 이러한 가정하에서 제안된 프로토콜은 어느 개체들도 입찰가에 대한 어떠한 정보도 얻지 못하고 경매의 결과는 정확하다는 것을 보장한다. 그리고 제안된 기법의 통신 패턴은 일반적인 안전하지 않은 경매 프로토콜과 동일하다. 즉, 각 입찰자는 판매자 S에게만 메시지를 전송하고 서버 A 또는 다른 입찰자들과는 통신하지 않으며, 각 입찰자는 S에게 하나의 메시지만을 전송한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 제안된 경매 프로토콜에 사용되는 기본적인 구성 요소들에 대해 기술하고 3장에서 새로운 경매 프로토콜을 제안하고 제안된 기법의 특징을 분석한다. 또한 계산량과 통신량 관점에서 기제안된 Naor-Pinkas-Sumner의 기법과 비교 분석한다. 마지막으로 4장에서는 결론을 맺는다.

II. 구성 요소들

2.1 이차 잉여(Quadratic Residuosity) 가정과 Goldwasser-Micali 암호화 기법

Z_x^*를 x보다 작고 x와 서로 소인 양의 정수들의

곱셈 군(multiplicative group)이라고 하자. $y \in Z_x^*$가 법(modulo) x에 관한 이차 잉여(quadratic residue)일 필요충분조건은 $r^2 \equiv y \bmod x$을 만족하는 $r \in Z_x^*$이 존재하는 것이다. 그렇지 않다면 y를 법 x에 관한 이차 비잉여(quadratic non residue)라고 한다. 이차 잉여 predicate를 다음처럼 정의한다.

$$QR_x(y) = \begin{cases} y\text{가 법 } x\text{에 관한 이차 잉여이면, } 0 \\ \text{그렇지 않다면, } 1 \end{cases}$$

소수 p에 대해 이차 잉여를 결정하는 문제는 Legendre 기호(symbol)를 계산하는 문제와 동치이다. 소수 p와 $y \in Z_p^*$에 대해 $y \bmod p$의 Legendre 기호 $(y|p)$는 다음처럼 정의되고, 이것은 Euler's criterion을 사용하여 다항식 시간 안에 계산될 수 있다.

$$(y|p) = \begin{cases} y\text{가 법 } p\text{에 관한 이차 잉여이면, } 1 \\ \text{그렇지 않다면, } -1 \end{cases}$$

Euler's criterion과 다음의 사실에 의해 정수 x의 소인수 분해가 알려져 있는 경우 법 x에 관한 이차 잉여를 결정하는 효율적인 알고리즘이 제공된다.

[사실 1]

y가 법 x에 관한 이차 잉여가 될 필요충분조건은 y가 x의 소인수 각각의 법에 대해 이차 잉여이다.

소인수 분해가 알려져 있지 않는 경우 이차 잉여에 관한 힌트가 Jacobi 기호 $(y|x)$에 의해 주어질 수 있으며, Jacobi 기호는 x의 소인수 분해가 알려져 있지 않는 경우에도 다항식 시간 안에 계산될 수 있다. Jacobi 기호 $(y|x) = -1$ 이면 y는 법 x에 관한 이차 비잉여이지만, Jacobi 기호가 $+1$ 이면 이차 잉여를 결정하기 위한 효율적인 알고리즘은 알려져 있지 않다. 이 경우 가장 빠른 방법은 x를 소인수 분해한 후 $QR_x(y)$를 계산하는 것이다.

Z_x^{+1}은 $(y|x) = +1$인 정수 $y \in Z_x^*$의 집합이고, QR_x와 NQR_x는 각각 Jacobi 기호 $+1$을 가진 법 x에 관한 이차 잉여와 법 x에 관한 이차 비잉여의 집합이다.

[정의 1] 이차 잉여 가정

각각의 효율적인 nonuniform 알고리즘 $Q = \{Q_n\}$, 모든 상수 d, 모든 충분히 큰 n에 대해,

$$\Pr\left(x \leftarrow BL(n); y \leftarrow Z_x^{+1} : Q_n(x, y) = QR_x(y)\right) < 1/2 + n^{-d}.$$

즉, 어떤 효율적인 알고리즘도 랜덤 추측(random guessing)보다 좋은 확률로 이차 잉여 predicate의 값을 추측할 수 없다.

여기서 BL은 Blum 정수(integer)의 집합을 나타내고, $BL(n)$을 길이 n의 Blum 정수들의 부분 집합이라고 한다.

Goldwasser-Micali(GM)[7] 암호화 기법은 1984년에 제안된 확률적 암호화 기법으로 안전성이 정의 1의 이차 잉여 가정에 기반한다. 기본적인 GM 암호화 기법의 공개키는 같은 길이의 두 소수 p와 q의 곱인 랜덤한 정수 x와 Jacobi 기호 $+1$을 가진 법 x에 관한 이차 비잉여인 y로 구성된다. 개인 복호화 키는 두 소수 p와 q로 구성된다. 암호화는 비트 단위로 수행된다. 즉, '0' 비트는 법 x에 관한 이차 잉여로 암호화되고, '1' 비트는 Jacobi 기호 $+1$을 가진 법 x에 관한 이차 비잉여로 암호화된다. 암호문의 각 비트의 이차 잉여 여부가 평문의 비트 값을 나타낸다. p와 q의 지식은 Z_x에서 원소의 이차 잉여 여부에 대한 결정을 효율적으로 가능하게 한다. 정의 1의 이차 잉여 가정이 어렵다는 가정하에서 GM 암호화 기법은 의미론적 안전성(semantic security)를 제공한다.

$GM_E(x, y, m)$을 (x, y)를 사용하여 계산된 m의 암호문을 반환하는 알고리즘이라 하고, $GM_D(p, q, c)$를 x의 소인수 p와 q를 사용하여 계산된 암호문 c의 복호된 값을 반환하는 알고리즘이라 한다.

2.2 Conditional Oblivious Transfer

Oblivious Transfer(OT)는 1982년 Rabin[12]에 의해 처음 제안되어졌다. OT는 두 다항식 시간 개체인 Alice와 Bob 사이의 게임으로, Alice는 Bob이 1/2의 확률로 메시지를 수신하도록 메시지를 전송하기 원하고, Alice는 Bob이 메시지를 수신했는지 하지 않았는지를 알 수 없다.

OT의 변형인 Conditional Oblivious Transfer (COT)는 1999년 Crescenzo-Ostrovskey-Rajagopalan[4]에 의해 제안되었다. COT에서 Bob과 Alice는 각각 비밀 입력을 가지며, 비밀 입력으로 계산되고 다항식 시간안에 계산 가능한 공개된

predicate를 공유한다. Alice로부터 Bob으로의 비트 b의 COT는 다음의 요구 사항을 가진다. predicate가 성립하면, Bob은 성공적으로 비트 b를 수신하며, predicate가 성립하지 않으면, Bob은 비트 b에 관한 어떤 정보도 얻지 못한다. 또한 프로토콜을 수행하는 동안 어떤 효율적인 전략도 Alice가 predicate의 실제 값을 계산하는데 도움을 주지 못한다.

k 비트의 두 sequence, $t_1, \cdots, t_k$와 $d_1, \cdots, d_k$가 주어지면, predicate GE는 다음처럼 정의된다: $GE(t_1, \cdots, t_k, d_1, \cdots, d_k) = 1$일 필요 충분 조건은 $(t_1 \circ \cdots \circ t_k) \geq (d_1 \circ \cdots \circ d_k)$이다. 여기서 $\circ$은 스트링의 연결을 나타내고 스트링 $t_1 \circ \cdots \circ t_k$와 $d_1 \circ \cdots \circ d_k$는 정수로 해석된다.

[정의 2] Conditional Oblivious Transfer

Alice와 Bob은 시큐리티 파라메터(security parameter) n에서 다항식 시간으로 동작하는 두 개의 확률적인 튜링 머신(probabilistic Turing machine)이다. x_A와 x_B는 각각 Alice와 Bob의 비밀 입력이다. b는 Alice가 Bob에게 전달하기 원하는 비밀 비트이다. $q(\cdot, \cdot)$는 다항식 시간안에 계산 가능한 predicate이다. 다음을 만족하는 상수 c가 존재하면 (Alice, Bob)은 predicate q에 대한 conditional oblivious transfer 프로토콜이다.

i. 전송 검증 : $q(x_A, x_B) = 1$이면, 각 $b \in \{0,1\}$에 대해, 다음이 성립한다.

$$\mathrm{Prob}[\sigma \leftarrow \{0,1\}^{n^c};\ tr \leftarrow (\mathrm{Alice}(x_A, b), \mathrm{Bob}(x_B))(\sigma) : \mathrm{Bob}(\sigma, x_B, tr) = b] = 1$$

ii. Bob에 대한 안전성 : $q(x_A, x_B) = 0$이면, 임의의 Bob'에 대해 확률 변수 X_0와 X_1은 equally distributed이다. $b \in \{0,1\}$에 대해,

$$X_b = [\sigma \leftarrow \{0,1\}^{n^c};\ tr \leftarrow (\mathrm{Alice}(x_A, b), \mathrm{Bob}'(x_B))(\sigma) : (\sigma, tr)]$$

iii. Alice에 대한 안전성 : 임의의 Alice'에 대해, 다음을 만족하는 효율적인 시뮬레이터 M이 존재한다. 임의의 상수 c와 임의의 충분히 큰 n에 대해, $|p_0 - p_1| \leq n^{-c}$가 성립한다.

$$p_0 = \mathrm{Prob}[\sigma \leftarrow \{0,1\}^{n^c};\ tr \leftarrow (\mathrm{Alice}'(x_A), \mathrm{Bob}(x_B))(\sigma) : \mathrm{Alice}'(\sigma, x_A, tr) = q(x_A, x_B)]$$

$$p_1 = \mathrm{Prob}[\sigma \leftarrow \{0,1\}^{n^c} : \mathrm{M}(\sigma, x_A) = q(x_A, x_B)]$$

predicate GE에 대한 COT 프로토콜은 다음과 같이 동작한다: 프로토콜 시작시에 Alice는 비밀키로 k-비트 스트링 $t = (t_1, \cdots, t_k)$를 가지고, Bob은 비밀키로 k-비트 스트링 $d = (d_1, \cdots, d_k)$를 가진다. b는 Alice가 Bob에게 전송하기 원하는 비트이다. 먼저 Bob은 자신의 공개키로 Blum 정수 x와 k쌍 (tuple) $(D_1, \cdots, D_k)$를 계산한다. 여기서 $D_i = r_i^2 y^{d_i} \bmod x$이고, r_i는 Z_x^*에서 랜덤하게 선택된 값이고 d_i는 Bob의 비밀키의 비트이다. 유사하게 Alice는 자신의 공개키로 정수 $T_1, \cdots, T_k \in Z_x^{+1}$을 계산한다. 여기서 T_i가 이차 비잉여일 필요충분조건은 $t_i = 1$이다. COT 프로토콜 (A, B)에서 정수 y의 역할을 하기 위해 적절히 계산된 T_i와 D_i의 곱을 사용한다(부록 참고). 여기서 곱은 비트 스트링 상의 predicate GE를 표현하는 부울 표현(boolean expression)에 의해 계산된다. 이것의 구현은 $\Theta(k^2)$ 모듈러 곱셈 (modular multiplication)을 요구한다.

COT 프로토콜의 상세한 사항은 부록에 기술된다. 이 COT 프로토콜은 정의 2에서의 3가지 성질, 즉 전송 검증, Bob과 Alice에 대한 안전성을 만족한다. 3절에서 Crescenzo-Ostrovskey-Rajagopalan의 COT에 사용된 기법을 적용한 효율적이면서 안전한 sealed-bid 경매 프로토콜을 제안한다.

Ⅲ. 두 서버를 가진 효율적인 sealed-bid 경매 프로토콜

이 장에서는 앞에서 기술된 COT를 응용한 효율적인 경매 프로토콜을 제안한다. 제안된 경매 프로토콜은 판매자 S와 제3자 A의 두 서버를 가지는 sealed-bid 경매 프로토콜로써, 2절에서 언급한 COT에서 사용한 기술을 입찰가 비교 과정에 이용한다. 제안된 프로토콜에서 두 서버 A와 S는 각각 COT에서 Alice와 Bob의 역할을 수행하며, COT에서 predicate GE가 만족되는지 판단하는 과정에 사용된 기술을 응용하여 원래의 입찰가를 노출하지 않고 암호화된 두 입찰가를 비교한다. 제안된 프로토콜은 두 서버가 공모하지 않는다면 어떤 개체도 입찰가에 대한 어떠

한 정보도 얻을 수 없으며, 경매의 결과는 정확하다는 것이 보장된다.

제안된 프로토콜에서는 암호화된 두 입찰가를 비교하여 원래의 입찰가가 노출되는 것을 방지한다. 이를 위해 원래의 입찰가에 GM 암호화 기법을 적용하여 입찰가를 암호화한다.

3.1 제안된 경매 프로토콜

입찰자 B_i의 입찰가 b는 k비트 값, $b=(b_1,\cdots,b_k)$이다. (NM_G, NM_E, NM_D)는 non-malleable 암호화 기법[5]이고 NM_G에 의해 암호화를 위한 비밀키와 공개키 쌍 (NM_sk, NM_pk)이 생성된다. (Gen, Sig, Ver)는 안전한 전자 서명 기법이고 Gen에 의해 서명 생성키와 서명 검증키 쌍 ($s-sk$, $s-pk$)이 생성된다.

3.1.1 경매 공고 및 초기화

서버 S는 판매할 물품, 서버 A의 ID, 경매 규칙 등을 공고한다. 각 입찰자 B_i는 Gen을 수행하여 서명 생성키와 검증키 쌍 ($Bs-sk_i$, $Bs-pk_i$)를 생성하여 $Bs-pk_i$를 공개한다. 유사하게 서버 A는 알고리즘 NM_G를 수행하여 (NM_pk, NM_sk) 쌍을 생성하고, Gen을 수행하여 서명 생성키와 검증키 쌍 ($As-sk$, $As-pk$)를 생성하여 NM_pk와 $As-pk$를 공개한다. 서버 S는 $p\equiv q\equiv 3 \bmod 4$인 두 개의 $\frac{n}{2}$ 비트 소수 p와 q를 랜덤하게 선택하여 $x=pq$을 계산하고, Gen을 수행하여 서명 생성키와 검증키 쌍 ($Ss-sk$, $Ss-pk$)를 생성하여 x와 $Ss-pk$를 공개한다.

3.1.2 입찰가 제출

[단계 2.1]

입찰자 B_i는 k비트 입찰가 $b=(b_1,\cdots,b_k)$를 S의 공개키로 GM 암호화 알고리즘을 이용하여 $D_i=(d_1,\cdots,d_k)$로 다음처럼 암호화한다.

$$\begin{aligned}&\text{for } i=1,\cdots,k \ \{\\ &\quad r_i \in_R Z_x^*\\ &\quad d_i = r_i^2(-1)^{b_i} \bmod x\\ &\}\end{aligned}$$

[단계 2.2]

D_i를 A의 공개키로 암호화하고, D_i에 대한 서명을 생성한다.

$$C_i = NM\_E(NM\_pk, D_i)$$
$$Sd_i = Sig(Bs-sk_i, D_i)$$

[단계 2.3]

B_i는 (C_i, Sd_i)를 S에게 전송한다.

3.1.3 입찰가 비교

[단계 3.1]

S는 수신된 모든 C_i들을 랜덤한 순서로 서버 A에게 전달하고 C_i의 리스트와 자신의 서명 $Sig(Ss-sk, (C_1\|C_2\|\cdots\|C_m))$을 공개한다. 여기서 m은 입찰자의 수를 나타낸다.

[단계 3.2]

A는 수신된 C_i들에 대한 서명 $Sig(As-sk, (C_1\|C_2\|\cdots\|C_m))$을 공개한다. 각 C_i들을 복호화한다.

$$D_i = NM\_D(NM\_sk, C_i)$$

수신된 C_i들의 순서를 랜덤하게 치환시킨 후 치환된 순서를 비밀로 유지한다. A는 토너먼트 방식으로 입찰가들을 비교한다. m명의 입찰자들에 대해 $\lceil \log_2 m \rceil$번의 입찰가 비교 라운드를 수행한다. 첫 번째 라운드에서 처음 제시된 입찰가 중에서 두 개씩 선택하여 비교한다. 두 번째 라운드에서는 첫 번째 라운드에서의 승리자에 대해 비교 과정을 수행한다. 이 과정을 $\lceil \log_2 m \rceil$번 수행한다. 라운드 회수 R을 1로 설정한다.

[단계 3.3]

R번째 라운드에서 A는 i번째 입찰가 $D=(d_1,\cdots,d_k)$와 j번째 입찰가 $T=(t_1,\cdots,t_k)$를 선택한다. 여기서 $i,j\in\{1,\cdots,\lceil m/2^{R-1}\rceil\}$, $i\neq j$이다. 각 라운드에서 i와 j는 한번만 선택된다. l을 1로 설정한다.

[단계 3.4]

A는 D와 T의 l번째 비트 비교를 위한 M_{ij}^l을 계산하여 S에게 전송한다.

$$M_{ij}^l = (-d_l t_l) \bmod x$$

[단계 3.5]

S는 A에게 비교 결과인 *flag*를 전송한다.

if $M_{ij}^{l} \in NQR_x$ then *flag* = 'Equal'
otherwise *flag* = 'Not Equal'

[단계 3.6]

A는 수신된 *flag*에 따라 다음처럼 동작한다.

```
if l equal to k {
   if flag equal to 'Equal'
         goto 단계 3.8
   else {
         M_ij^l = (-(t_l)^2 d_l) mod x
         send M_ij^l to S
         goto 단계 3.7
   }
}
else {
   if flag equal to 'Equal' {
         l = l+1
         M_ij^l = (-d_l t_l) mod x
         send M_ij^l to S
         goto 단계 3.5
   }
   else {
         M_ij^l = (-(t_l)^2 d_l) mod x
         send M_ij^l to S
         goto 단계 3.7
   }
}
```

[단계 3.7]

S는 A에게 비교 결과인 *flag*를 전송한다.

if $M_{ij}^{l} \in NQR_x$ then *flag* = 'GT'
otherwise *flag* = 'LT'

[단계 3.8]

A는 수신된 *flag*를 검사한다. 만약 *flag*가 'GT'이면, j번째 입찰가 T가 i번째 입찰가 D보다 크다. 만약 *flag*가 'LT'이면, i번째 입찰가 D가 j번째 입찰가 T보다 크다. *flag*가 'Equal'이면, 두 입찰가는 같다.

[단계 3.9]

A는 입찰가의 모든 (i, j) 쌍에 대해 단계 3.3부터 반복 수행한다.

[단계 3.10]

R번째 라운드에서 승리한 입찰가들의 인덱스와 인덱스에 대한 서명을 S에게 전송한다. 여기서 입찰가들의 인덱스는 단계 3.2에서 A가 수신한 입찰가들의 순서에서 인덱스이다. A는 $R=R+1$로 설정한 후 최종 승리자가 결정될 때까지 승리한 입찰가들에 대해 단계 3.3을 계속 수행한다.

3.1.4 승리자의 입찰가 공개

[단계 4.1]

A는 최종 승리자의 인덱스를 S에게 전송한다.

[단계 4.2]

S는 승리자 B_{win}의 ID_{win}과 암호화된 입찰가의 서명 Sd_{win}을 A에게 전송한다.

[단계 4.3]

A는 D_{win}의 서명 Sd_{win}을 검증한 후 승리자의 암호화된 입찰가 D_{win}과 자신의 서명 $Sig(As-sk, (ID_{win} \| D_{win}))$을 S에게 전송한다.

[단계 4.4]

S는 $Sig(As-sk, (ID_{win} \| D_{win}))$와 Sd_{win}을 검증한 후에 D_{win}을 복호화하여 $b_{win}=(b_1, \cdots, b_k)$을 계산한다. S ID_{win}, b_{win}, D_{win}, Sd_{win}, $Sig(As-sk, (ID_{win} \| D_{win}))$, $Sig(Ss-sk, (ID_{win} \| b_{win}))$}를 공개한다.

두 개의 암호화된 입찰가 $D=(d_1, \cdots, d_k)$와 $T=(t_1, \cdots, t_k)$의 비교 과정에서, $-d_l t_l \bmod x$가 법 x에 관한 이차 비잉여이면 두 암호화된 입찰가의 l번째 비트는 같다. 그리고 $-(t_l)^2 d_l \bmod x$가 법 x에 관한 이차 비잉여이면 T의 l번째 비트가 D보다 크다. 그러므로 A와 S는 원래의 입찰가를 노출하지 않고 암호화된 입찰가를 비교할 수 있다.

3.2 제안된 기법 분석

제안된 sealed-bid 경매 프로토콜은 다음과 같

은 특징을 가진다.

- 통신 패턴 : 제안된 프로토콜의 통신 패턴은 안전하지 않은 일반적인 경매와 동일하다. 즉 입찰자는 경매인 S와만 통신하고 A 또는 다른 입찰자와 통신하지 않는다. 각 입찰자는 S에게 하나의 메시지만을 전송한다.
- 다른 형태의 경매로 적응성(auction adaptability) : 제안된 경매 프로토콜은 작은 오버헤드로 Vickrey 경매와 같은 다른 형태의 경매로 적용가능하다.
- 프라이버시 : 제안된 프로토콜은 두 서버 중 한 서버와 임의의 수의 입찰자를 제어할 수 있는 공격자에 대해 프라이버시를 제공한다. 그러한 공격자에게 노출되는 정보는 경매의 결과이다. 입찰자는 자신의 입찰가를 $GM_E()$와 $NM_E()$로 이중 암호화함으로써 입찰가를 숨긴다. S는 $NM_E()$로 암호화된 입찰가를 수신하고 A는 $GM_E()$로 암호화된 것을 수신하기 때문에, S와 A는 입찰가를 알지 못한다. 제안된 프로토콜에서 입찰가의 프라이버시는 GM 암호화 기법과 non-malleable 암호화 기법의 안전성에 의존한다. 그렇지만 S와 A는 입찰가의 비교 과정에서 입찰가에 대한 부분 정보를 얻을 수 있다. 단계 3.5에서 $-d_l t_l \bmod x$가 법 x에 관한 이차 비잉여이면 두 암호화된 입찰가의 l번째 비트가 같다는 것을 안다. 그러나 l번째 비트가 '1' 또는 '0'인지는 알지 못한다. 또한 단계 3.7에서 $-(t_l)^2 d_l \bmod x$가 법 x에 관한 이차 비잉여이면 t_l이 d_l보다 크다는 것을 알 수 있다 (이 경우 t_l은 '1'이고 d_l은 '0'이다). 그렇지만 단계 3.7은 두 입찰가 비교 과정에서 단지 한번만 수행되므로 l번째 비트를 제외한 나머지 비트에 관한 정보를 얻을 수 없다. 더욱이 단계 3.1에서 입찰가들이 랜덤한 순서로 A에게 전송되고 A 또한 랜덤하게 치환된 순서로 두 입찰가를 비교하므로, S와 A는 실제 입찰자와 부분 정보를 관련시킬 수 없다.
- 부인 방지 : 입찰자 B_i는 자신의 서명 Sd_i로 인해 입찰을 부인할 수 없다.
- 검증 가능성 : 단계 3.1에서 S는 C_i의 리스트와 자신의 서명을 공개하고 A 또한 그 리스트의 서명을 공개한다. 그러므로 각 입찰자는 자신의 입찰가가 리스트에 포함되어 있는지를 검사할 수 있다. 더욱이 각 입찰자는 승리자의 입찰가와 자신의 입찰가를 비교함으로써 경매의 결과를 검증할 수 있다. 만약 자신의 입찰가가 더 높다면, 자신의 입찰가 $(b_1, \cdots, b_k)$와 이것의 암호화된 값 $(d_1, \cdots, d_k)$ 계산에 사용된 $(r_1, \cdots, r_k)$, Sd_i를 공개하여 자신이 승리자라고 주장할 수 있다. S와 A는 단계 4.4에서 경매의 결과에 대해 서명을 하기 때문에 분쟁이 발생하면 자신들의 잘못을 부인할 수 없다.
- 익명성 : 제안된 프로토콜은 입찰자의 익명성을 제공하지 않는다. 하지만 입찰자가 경매 사이트에 등록한 후 경매에 참여하는 인터넷 상의 경매를 고려해보면, 제안된 프로토콜에서도 입찰자는 경매에 참여하기 전에 서버 A에 등록한 후 경매에 참여해야 한다. 이 등록 과정에서 입찰자는 실제 ID 대신 임시 ID를 얻어 경매에 참여할 수 있다. 이 경우 판매인 S는 입찰자에 관한 정보를 얻을 수 없게 된다. 하지만 이것은 서버 A에 대한 추가적인 신뢰를 요구하게 된다. 서버 A는 암호화된 입찰가를 가지므로 입찰자와 실제 입찰가를 관련시킬 수 없다.
- 암호화 기법의 안전성 : 입찰자가 non-malleable 암호화 기법을 사용하여 S에게 입찰가를 제출하는 것은 중요하다. 그러한 암호화 기법은 S가 암호문을 수정하여 평문에서 의미있는 변화를 유발하는 것을 방지한다[5].
- 재연 공격 방지 : 이전에 수행된 경매로부터의 입찰가를 새로운 경매에 그대로 제출하는 재연 공격을 방지해야 한다. 이것은 경매에 판매자 S의 ID, 날짜와 같은 경매에 유일한 식별자을 추가함으로써 쉽게 처리될 수 있다. C_i 계산시에 이 값을 추가하여 암호화한다. non-malleability 성질이 경매의 식별자가 수정될 수 없다는 것을 보장한다.
- 부정한 서버 검출 : S 또는 A가 입찰자와 공모하면, S 또는 A는 특정 입찰자를 항상 승리자로 만들거나 입찰가 비교 과정에서 부정확한 값을 전달함으로써 계산을 방해할 수 있다. 제안된 프로토콜에서 S는 M_{ij}^l이 어떤 입찰자에 관련되는지를 식별할 수 없기 때문에 S는 특정 입찰자를 승리자로 만들 수 없다. 또한 A는 S가 정확히 동작하고 있다는 것을 검증할 수 있다. 이를 위해 A는 랜덤하게 두 값 $u=(u_1, \cdots, u_k)$와 $v=(v_1, \cdots, v_k)$를 선택하여 비교 단계를 수행한다. A는 이 과정을 비교 단계에서 랜덤하게 적절한 회수로 수행한다. 한번이라도 부정이 검출된 S는 신뢰를 상실하게

되고 그 이후에 경매 참가가 제한된다. A가 특정 입찰자 B_i를 승리자로 만들기 위해서는 단계 3.8에서의 *flag*에 무관하게 B_i를 승리자로 결정한다. 그렇지만 단계 3.10에서 A가 R 라운드에서의 승리자의 인덱스를 S에게 전달하기 때문에 이것이 검출된다. S는 자신은 승리자를 알 수 있게 랜덤하게 C_i들을 생성하여 A의 동작을 검증할 수 있다. S는 원래의 C_i들와 랜덤하게 선택된 C_i들 중에서 랜덤하게 선택하여 단계 3.1을 수행한다. 이것을 부정한 A가 검출되지 않은 확률이 충분히 작도록 적절한 회수만큼 수행한다. 한번이라도 부정이 검출된 A는 신뢰를 상실하게 되며 따라서 이러한 부정에 대한 A의 위험에 대한 이익 비율(risk-to-benefit ratio)이 매우 작기 때문에 두 번의 수행만으로도 충분한 것으로 보인다.

3.3 제안된 기법의 계산 복잡도와 통신 복잡도

제안된 기법의 계산 복잡도와 통신 복잡도를 분석한 후 Naor-Pinkas-Sumner[11]의 기법과 비교한다.

Naor-Pinkas-Sumner의 기법은 다음과 같이 동작한다. 경매 issuer A는 경매 프로토콜을 나타내는 암호화된 서킷(garbled circuit)을 구성하고, 경매인 S는 암호화된 입찰가를 사용하여 경매를 위한 서킷을 평가한다. 주어진 입찰가에서 비트 b에 대한 암호화된 입력을 얻기 위해 Proxy OT(Oblivious Transfer) 프로토콜을 수행한다. Proxy OT 프로토콜에서 입찰자는 선택자(Chooser), S는 대리인(Proxy), A는 전송자(Sender)로 동작한다. 전송자는 입찰가에서 '0' 비트와 '1' 비트에 해당하는 서킷에 대해 암호화된 입력 (t_0, t_1)을 전송한다. 선택자는 자신의 입찰가에서 비트 b에 기반하여 t_b를 선택하여 전송 프로토콜을 통해 그것을 대리인에게 전송한다. 모든 입찰가의 모든 비트에 대해 이 과정을 수행한 후, 대리인은 입력 입찰가에 관해 서킷을 평가하여 경매의 결과를 결정할 수 있다. Proxy OT의 프라이버시 성질은 대리인이 임의의 비트에 대한 b 또는 t_{1-b}를 알지 못한다는 것을 보장한다. 그러므로 대리인은 입찰 정보에 관해 알지 못하며, 유사하게 전송자 역시 입찰가에 관해 알지 못한다. 대리인과 전송자가 공모하는 경우에만 입찰가에 대한 프라이버시가 손상된다.

제안된 기법의 계산량과 통신 오버헤드 분석을 위해 경매에 참여한 입찰자의 수를 m이라 하고, n은 제안된 기법에 사용된 공개키 암호(GM 암호화 기법, non-malleable 암호화 기법, 전자 서명)의 시큐리티 파라메터라고 하자. k는 입찰의 비트 길이다.

3.3.1 계산 복잡도

〔표 1〕은 제안된 기법과 Naor-Pinkas-Sumner의 기법의 계산 복잡도를 보여준다. 먼저 제안된 경매 프로토콜의 계산 복잡도는 다음과 같다. 제안된 경매 프로토콜에 m 명의 입찰자가 참가했을 경우 S와 A간의 입찰가 비교 과정은 토너먼트 방법을 사용하기 때문에 총 $(m-1)$번 수행된다.

제안된 프로토콜에서 입찰자 B_i의 계산 복잡도를 살펴보면, B_i는 단계 2.1에서 $D_i=(d_1,\cdots,d_k)$ 계산을 위해 k번의 모듈러 제곱(modular square: MOD SQR) 연산을 수행해야 하고, 단계 2.2에서 C_i 계산을 위해 k번의 NM_E 암호화 연산이 필요로 하고 Sd_i 계산을 위해 한번의 서명 생성 연산을 필요로 한다.

서버 A는 단계 3.2에서 수신된 입찰가의 복호화를 위해 mk번의 NM_D 복호화 연산을 수행해야 한다. 그리고 S와 두 개의 입찰가를 비교하는 단계 3.4에서 단계 3.8의 과정에서 최악의 경우 $(k+1)$ 번의 모듈러 곱셈(modular multiplication: MOD MUL) 연산과 1번의 MOD SQR 연산을 요구한다. 토너먼트 방식에 의한 입찰가 비교 단계(단계 3.2에서 단계 3.9)가 $(m-1)$번 수행되므로 최종 승리자 결정을 위해 총 $(m-1)\times(k+1)$ 번의 MOD MUL 연산과 $(m-1)$ 번의 MOD SQR 연산을 요구한다. 마지막으로 단계 3.2에서 모든 C_i들을 수신했을 때 한번의 서명 생성 연산이 필요하고 승리자

〔표 1〕 제안된 기법과 Naor-Pinkas-Sumner 기법의 계산량 비교

	제안된 기법	Naor-Pinkas-Sumner의 기법
입찰자 B_i	k MOD SQR + k NM_E +1 $Sig(\cdot)$	k MOD EXP+ k MOD DIV+2 k NM_E
서버 A	mk NM_D +(m-1)(k+1) MOD MUL+(m-1) MOD SQR+2 $Sig(\cdot)$ +1 $Ver(\cdot)$	2σ $F(\cdot)$ + mk MOD DIV+ 2 mk ElGamal_E+ mk NM_D
서버 S	{(m-1)(k+1)+ k) GM_D +2 $Ver(\cdot)$ +2 $Sig(\cdot)$	2 mk ElGamal_D+ mk NM_D+ σ $F(\cdot)$

의 입찰가 공개 과정(단계 4.3)에서 한번의 서명 생성과 한번의 서명 검증 연산이 요구된다.

제안된 프로토콜에서 서버 S는 먼저 단계 3.1에서 한번의 서명 생성 연산을 수행한다. 그리고 A와 두 개 입찰가 비교를 위해 단계 3.4에서 단계 3.8의 과정에서 최악의 경우 $(k+1)$번의 GM_D 연산을 필요로 하므로, 최종 승리자 결정을 위해 총 $(m-1)\times(k+1)$번의 GM_D 연산을 수행한다. 승리자의 입찰가 공개를 위해 단계 4.4에서 2번의 서명 검증 연산과 k번의 GM_D 복호화 연산 그리고 한번의 서명 생성 연산을 필요로 한다.

Naor-Pinkas-Sumner의 기법에서 계산 복잡도는 다음과 같다. 먼저 각 입찰자 B_i는 입찰가 제출을 위해 k번의 Proxy OT 프로토콜을 수행하므로, 이를 위해 k번의 모듈러 지수승(modular exponentiation: MOD EXP) 연산과 k번의 모듈러 나눗셈(modular division: MOD DIV) 연산을 필요로 하며 $2k$번의 NM_E 암호화 연산을 수행해야 한다.

σ개의 게이트(gate)를 가진 서킷으로 가정할 때 경매 issuer A는 경매를 위한 서킷 준비를 위해 2σ번의 유사랜덤 함수 $F(\cdot)$ 계산을 요구한다. 그리고 Proxy OT 프로토콜에서 전송자로 동작하므로 각 입찰자에 대해 k번의 MOD DIV 연산, $2k$번의 ElGamal_E 공개키 암호화 연산, k번의 NM_D 복호화 연산을 필요로 한다.

마지막으로 판매자 S는 Proxy OT 프로토콜에서 대리인으로 동작하며, 이때 각 입찰자에 대해 $2k$번의 ElGamal_D 공개키 복호화 연산, k번의 NM_D 복호화 연산을 수행해야 한다. 그리고 서킷 평가를 위해 σ번의 유사랜덤 함수 계산을 필요로 한다.

비교를 위해 두 프로토콜에 사용된 기본 연산의 복잡도를 살펴보면, MOD MUL 연산과 MOD DIV 연산의 비트 복잡도는 약 $O(n^2)$이고 MOD EXP 연산은 약 $O(n^3)$ 비트 연산의 수행 시간을 가진다. 그리고 GM_D 계산은 Jacobi 기호 계산과 같고 이것은 약 $O(n^2)$의 비트 복잡도를 가진다. 또한 ElGamal_E 암호화는 2번의 MOD EXP 연산을 요구하고 ElGamal_D 복호화는 한번의 MOD EXP 연산, 한번의 MOD DIV 연산, 한번의 MOD MUL 연산을 필요로 한다. 따라서 ElGamal_D 복호화의 비트 복잡도는 약 $O(n^3)$이다. MOD SQR 연산은 MOD MUL 연산보다 작기 때문에 k번의 MOD SQR 연산은 한번의 MOD EXP 연산보다 작은 복잡도를 가진다.

따라서, [표 1]과 위의 기본 연산의 복잡도에 의해 제안된 기법이 입찰자의 계산 오버헤드 관점에서 Naor-Pinkas-Sumner의 기법보다 훨씬 효율적임을 알 수 있다. 그리고 서버 A의 계산량을 비교해보면 Naor-Pinkas-Sumner 기법은 제안된 기법보다 대략 $2mk$번의 ElGamal_E 암호화 연산을 더 필요로 한다. 또한 서버 S의 계산량을 비교해보면, 제안된 기법에서 $(mk+m)$번의 GM_D 계산($O(n^2)$)은 Naor-Pinkas-Sumner 기법에서 $2mk$번의 ElGamal_D 복호화 연산($O(n^3)$)보다 훨씬 적은 비용이고, Naor-Pinkas-Sumner 기법은 추가적으로 mk번의 NM_D 연산을 더 요구한다. 따라서, 제안된 경매 프로토콜이 계산량적인 관점에서 Naor-Pinkas-Sumner 기법보다 매우 효율적임을 알 수 있다.

3.3.2 통신 복잡도

먼저 제안된 기법의 통신 복잡도를 살펴보면, 제안된 프로토콜에서 각 입찰자 B_i는 입찰가 제출 과정에서만 S에게 하나의 메시지를 전달하고 A와는 통신을 하지 않으며, 그 이후의 과정에는 더 이상 참여하지 않는다. 이 과정에서 각 입찰자 B_i는 약 $(k+2)n$ 비트를 S에게 전달해야 한다.

그리고 S와 A간의 통신 복잡도는 다음과 같다. 먼저 입찰가 비교를 위해 S는 nmk 비트를 A에게 전달하고, 최종 승리자를 결정하기까지 S와 A는 $(m-1)$번의 통신 라운드를 수행한다. 두 개의 입찰가 비교를 위해 A는 S에게 최악의 경우 $(k+1)$번의 M_{ij}^{l} 메시지를 전달하고, 각 M_{ij}^{l} 메시지는 n 비트 메시지이므로 $(k+1)n$ 비트가 전송된다. 또한 S는 최악의 경우 $(k+1)$번의 *flag* 메시지를 A에게 전송한다. 따라서 $(m-1)$번의 입찰가 비교 과정을 수행하기 위해 S와 A간에 전송되는 총 비트 수는 $(m-1)\times(k+1)\times(n+g)$ 비트이다. 여기서 g는 *flag*의 비트 길이다. 그리고 최종 승리자 공개 과정에서 두 서버간에 발생하는 통신 오버헤드는 $(k+2)n+\lceil \log_2 m \rceil + I$ 비트이다. 여기서 I는 입찰자 ID의 비트 길이다.

Naor-Pinkas-Sumner의 기법에서 통신 복잡도는 다음과 같다. 먼저 각 입찰자 B_i는 S에게 $2kn$ 비트 메시지를 전송한다. 그리고 A와 S간에 온라인(online)으로 각 입찰자에 대해 $(2k+1)n$ 비트 메시지 전송과 서킷 전송을 위해 오프라인(offline)으로 약 $2400km$ 비트의 전송을 필요로 한다.

〔표 2〕 제안된 기법과 Naor-Pinkas-Sumner 기법의 통신량 비교

	제안된 기법	Naor-Pinkas-Sumner의 기법
$B_i \rightarrow S$	$(k+1)n$ 비트	$2kn$ 비트
$S \leftrightarrow A$	$mkn+(m-1)(k+1)(n+g)+(k+2)n$ $+\lceil \log_2 m \rceil + I$ 비트	$(2k+1)nm$ 비트(online) + 2400 km 비트(offline)

〔표 2〕은 두 기법의 통신 복잡도를 비교한다. 입찰자의 통신 오버헤드 관점에서 Naor-Pinkas-Sumner의 기법은 제안된 기법에 비해 $(k-1)n$ 비트를 추가적으로 더 전송해야 한다. 그리고, S와 A 사이의 통신량을 비교해보면 제안된 기법이 Naor-Pinkas-Sumner 기법보다 $(gm(k+1)+n)$ 비트를 추가적으로 더 전송해야 하지만, g가 작은 값(2 또는 4비트)이기 때문에 이 추가적인 전송량은 A와 S간에 발생하는 전체 전송량에서 작은 비율을 차지한다. 또한 Naor-Pinkas-Sumner의 기법은 오프라인으로 2400km 비트 전송을 요구한다.

좀더 구체적으로 두 기법의 통신 오버헤드를 비교하기 위해 $m=1000$ ($\log_2 m \approx 10$), $k=10$, $n=1024$, $g=4$라고 하자. 그러면 Naor-Pinkas-Sumner 기법에서 입찰자의 통신 오버헤드는 20,480 비트이고 제안된 기법은 11,264 비트이므로, 제안된 기법이 약 2배정도 효율적이다. 그리고 A와 S간의 온라인 통신 오버헤드는 제안된 기법이 약 21,548,990 비트이고 Naor-Pinkas-Sumner 기법은 약 21,540,000 비트이다. 그렇지만 Naor-Pinkas-Sumner 기법은 오프라인으로 약 24Mbit를 더 전송해야 한다. 그러므로 제안된 기법은 온라인 통신량에서 Naor-Pinkas-Sumner 기법과 비슷하고 오프라인 통신을 요구하지 않기 때문에 통신 오버헤드 관점에서 효율적이다.

Ⅳ. 결 론

본 논문에서는 Crescenzo-Ostrovskey-Rajagopalan에 의해 제안된 Conditional Oblivious Transfer 프로토콜에 사용된 기술을 이용하여 두 개의 서버를 가진 효율적이고 안전한 sealed-bid 경매 프로토콜을 제안하였다. 제안된 프로토콜에서 제3자 A와 경매인 S가 공모하지 않는다면 어떤 개체들도 입찰에 대한 어떠한 정보도 얻을 수 없으며, 경매의 결과는 정확하다는 것이 보장된다. 또한 제안된 프로토콜은 일반적인 경매 프로토콜과 동일한 통신 패턴을 가진다. 즉, 각 입찰자는 판매자 S에게만 메시지를 전송하고, 서버 A 또는 다른 입찰자들과는 통신하지 않는다. 그리고 제안된 기법은 최근까지 제안된 기법들 중에서 안전하면서 매우 효율적인 것으로 알려진 Naor-Pinkas-Sumner의 기법보다 훨씬 적은 계산량을 가지며 유사한 통신 오버헤드를 가진다.

참 고 문 헌

〔1〕 O. Baudron, J. Stern, "Non-interactive private auctions", *Financial Cryptography'01*, pp.303~313, 2001.

〔2〕 C. Cachin, "Efficient private bidding and auctions with an oblivious third party", *ACM CCS'99*, pp.120~127, 1999.

〔3〕 G. De Santis, G. Di Crescenzo, G. Persiano, "Zero-Knowledge arguments and public-key cryptography", *Information and Computation*, Vol. 121, pp.23~40, 1995.

〔4〕 G. Di Crescenzo, R. Ostrovsky, S. Rajagopalan, "Conditional oblivious transfer and timed-released encryption", *Advances in Cryptography - Eurocrypt'99*, LNCS 1592, pp.74~89, 1999.

〔5〕 D. Dolev, D. Dwork, M. Naor, "Non-malleable cryptography", *ACM Symp. on Theory of Computing*, 1991.

〔6〕 M. Franklin, M. Reiter, "The design and implementation of a secure auction server", *IEEE Trans. on Information Theory*, Vol. 22, No. 5, pp.302~312, 1996.

〔7〕 S. Goldwasser, S. Micali, "Probabilistic encryption", *J. Comp. Sys. Sci*, Vol. 28, No. 1, pp.270~299, 1984.

〔8〕 M. Harkavy, J. D. Tygar, H. Kikuchi, "Electronic auctions with private bids", *3rd USENIX Workshop on Electronic Commerce*, pp.61~73, 1999.

〔9〕 M. Jakobsson, A. Juels, "Mix and match:

Secure function evaluation via ciphertexts", *Advances in Cryptography - Asiacrypt'00*, LNCS 1976, pp.143~161, 2000.

[10] A. Juels, M. Szydlo, "A two-server, sealed-bid auction protocol", *Financial Cryptography'02*, to appear.

[11] M. Naor, B. Pinkas, R. Sumner, "Privacy preserving auctions and mechanism design", *ACM Conf. on Electronic Commerce*, pp.129~139, 1999.

[12] M. O. Rabin, "How to exchange secrets by oblivious transfer", *Tech Memo TR-81*, Aiken Computation Laboratory, 1981.

[13] K. Sako, "An auction protocol which hides bids of losers", *PKC'00*, LNCS 1751, pp.422~432, 2000.

[14] A. C. Yao, "Protocols for secure computations", *FOCS'82*, pp.160~164, 1982.

부록: COT 프로토콜

COT는 1995년 Santis-Crescenzo-Persiano[3]에 의해 제안된 OT의 간단한 변형인 서브 프로토콜 (A, B)를 사용한다. $NQR-COT-Send(b, x, y)$는 입력으로 비트 b와 (x, y)를 가지는 알고리즘 A로, $GM_E(x, y, b)$로 암호화된 값을 반환한다. x는 Blum 정수, $y \in Z_x^{+1}$이다. $NQR-COT-Receive(mes, (x, p, q, y))$는 (x, y)를 사용하여 A에 의해 전송된 메시지 mes를 $GM_D(p, q, mes)$로 복호화하기 위해 사용하는 알고리즘 B로, 복호화 결과는 b 또는 $\perp$이다($\perp$는 유효하지 않은 메시지를 나타낸다). 프로토콜 (A, B)에서 y가 이차 비잉여이면 알고리즘 B는 A에 의해 전송된 비트 b를 수신한다. 그렇지 않다면 정직하지만 호기심 강한(honest-but-curious) B에 관하여 b의 실제 값은 무조건적으로 안전하다(unconditionally secure). 또한 B가 b를 실제 수신했는지에 관하여 A가 추측할 수 있게 하는 효율적인 전략은 존재하지 않는다.

먼저 Alice는 비트 b를 비트 a와 비트 $a \oplus b$로 분할한다. a는 랜덤하게 선택된다. Alice는 (x, T_1)을 사용하여 비트 a를 전송하고 서브 프로토콜 (A, B)에 대한 입력으로 $(x, D_1 T_1 \bmod x)$을 사용하여 $a \oplus b$를 전송한다. Bob이 b를 수신할 필요충분조건은 $t_1 > d_1$이다. Alice는 $(x, -T_1 D_1 \bmod x)$를 사용하여 랜덤 비트 c를 전송한다. Bob이 c를 수신할 필요충분조건은 $t_1 = d_1$이다. $t < d$이면 Bob은 균일하고(uniform) b에 독립인 분포를 가진 비트들만을 계산하는 것이 가능하다.

[알고리즘 Alice]

입력 $b, t_1, \cdots, t_k \in \{0,1\}$에 관해 다음을 수행한다.

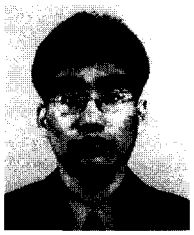
1. Receive : Bob으로부터 $x, D_1, \cdots, D_k$을 수신하고 $b_1 = b$로 설정한다.
2. For $i = 1, \cdots, k$ $\{a_i, c_i \in \{0,1\}$와 $r_i \in Z_x^*$를 랜덤하게 선택하고 $T_i = r_i^2(-1)^{t_i} \bmod x$를 계산한다.
   $i = k$이면, $c_i = b_i$로 설정한다.
   $mes_{i1} = NQR-COT-Send(a_i, (x, T_i))$를 계산한다.
   $mes_{i2} = NQR-COT-Send(a_i \oplus b_i, (x, D_i T_i \bmod x))$를 계산한다.
   $mes_{i3} = NQR-COT-Send(c_i, (x, -D_i T_i \bmod x))$를 계산한다.
   $b_{i+1} = b_i \oplus c_i$로 설정한다.
   }
   $p_A = (T_1, \cdots, T_k)$와
   $mes = ((mes_{11}, mes_{12}, mes_{13}), \cdots, (mes_{k1}, mes_{k2}, mes_{k3}))$으로 설정한다.
3. Send : (p_A, mes)를 Bob에게 전달한다.

[알고리즘 Bob]

충분히 긴 스트링 σ와 $d_1, \cdots, d_k \in \{0,1\}$를 입력으로 하여 다음을 수행한다.

1. $p \equiv q \equiv 3 \bmod 4$인 두 개의 n-비트 소수 p, q를 랜덤하게 선택한다. $x = pq$로 설정한다.
   for $i = 1, \cdots, k$,
   $r_i \in Z_x^*$를 랜덤하게 선택하고
   $D_i = r_i^2(-1)^{d_i} \bmod x$를 계산한다.
   $p_B = (x, D_1, \cdots, D_k)$로 설정한다.
   Send : p_B를 Alice에게 전달한다.
2. Receive : $((T_1, \cdots, T_k), (mes_{11}, mes_{12}, mes_{13}, \cdots, mes_{k1}, mes_{k2}, mes_{k3}))$을 Alice로부터 수신한다.
3. For $i = 1, \cdots, k$,
   $a_i = NQR-COT-Receive(mes_{i1}, (x, p, q, T_i))$
   를 계산한다.
   $e_i = NQR-COT-Receive(mes_{i2}, (x, p, q, D_i T_i \bmod x))$를 계산한다.
   $a_i \neq \perp$이고 $e_i \neq \perp$이면,
   Output : $a_i \oplus e_i \oplus c_{i-1} \oplus \cdots \oplus c_1$를 출력하고 중단한다.
   그렇지 않다면,
   $c_i = NQR-COT-Receive(mes_{i3}, (x, p, q, -D_i T_i \bmod x))$를 계산한다.
   $i = k$이고 $c_i \neq \perp$이면,
   Output : c_i를 출력하고 중단한다.
   Output : $\perp$를 출력한다.

〈著者紹介〉



신 상 욱 (Sang Uk Shin) 정회원
1995년 2월 : 부산수산대학교(현 부경대학교) 전자계산학과(학사)
1997년 2월 : 부경대학교 전자계산학과(석사)
2000년 2월 : 부경대학교 전자계산학과(박사)
2000년 4월~현재 : 한국전자통신연구원 선임연구원
〈관심분야〉 암호학, 이동통신 보안

류 희 수 (Heuisu Ryu) 정회원
1990년 2월 : 고려대학교 수학과(학사)
1992년 2월 : 고려대학교 수학과(석사)
1999년 5월 : Johns Hopkins University 수학과(박사)
2000년 7월~현재 : 한국전자통신연구원 선임연구원
〈관심분야〉 정보보호이론, 타원곡선암호