An Auction Protocol Which Hides Bids of Losers

Kazue Sako

C&C Research Laboratories, NEC Corporation
4-1-1 Miyazaki Miyamae Kawasaki 216-8555 Japan
sako@ccm.cl.nec.co.jp

Abstract. Many auction protocols using practical cryptographic means have successfully achieved capability of hiding the bids of each entity, but not the values of bids themselves. In this paper we describe an auction protocol which hides the bids of non-winners even from the bid-opening centers, and still makes it possible to publicly verify the validity of the winning bid, i.e. that it was the highest bid submitted. The first approach to such a protocol was made by Kikuchi et al in [KHT98]. However, several deficiencies have been pointed out regarding their protocol; for example, it is not well suited for handling tie bids.
We present an auction protocol in which a bid will not be successfully decrypted unless it is the highest bid, thus ensuring bid privacy. In addition, it enables participants to verify that the winning bid is indeed the highest. Also in contrast to the previous work, our protocol can identify all the winners who submitted the winning bid.
Our protocol allows for very compact representations for bids: a bid is represented by a single probabilistic encryption. In the protocol of [KHT98] a bid is represented by a vector of encryptions, of length linear in the number of possible bid values.
We present two practical schemes based on the ElGamal cryptosystem and the RSA cryptosystems, respectively.

Key words: auction, privacy, group decryption.

1 Introduction

Fairness, privacy and correctness have been considered to be three major security issues in auction protocols. By fairness, we mean that we want to ensure that neither the value of a bid itself nor any partial information, e.g. information that might give any bidder an unfair advantage, will be disclosed before the opening time. By privacy, we mean that we do not want to reveal which entity has bid at what value even after the opening. By correctness, we mean that we want the winning bid to be the highest (or lowest) among bids which were entered before opening time, and we want the winner to be the person who made that bid.

These goals have been successfully achieved as depicted in the work of Franklin and Reiter[FR96]. However, no practical protocol has succeeded in keeping the *secrecy of losing bids*, i.e. no protocol that makes winning bid public

H. Imai, Y. Zheng (Eds.): PKC 2000, LNCS 1751, pp. 422–432, 2000.

but not the losing ones has yet been satisfactorily developed. This is important, as disclosure of these values also reveals information on strategies of losing entities. Leakage of such strategies will strongly influence succeeding auctions with a similar group of entities. However, despite the needs for such a protocol, almost all practical protocols intentionally reveal all the actual values that were bid, in order to enable verification that the winning bid was indeed the highest or lowest among all those made.

In this paper, we describe a protocol that enhances privacy in auctions by keeping values that were bid *secret*, but still enables the fact that the winning value is indeed the highest of the submitted bids to be *publicly* verified.[1]

Kikuchi et al [KHT98] first addressed this problem. They try to achieve this property by having a bid represented in a vector of L values, where L is the number of possible values that can be bid. Bidders place 0 on the values they do not wish to bid. The winning bid is determined by adding all the submitted bid-vectors and finding the last non-zero element in the summed vector. The protocol achieves minimum round complexity for bidders, as once they have submitted their bids they do not need to participate in opening. Unfortunately, their scheme, together with an enhanced scheme in [HTK98], is not well suited for handling tie bids, i.e. cases when there are two or more entities who submit the same winning bid. If this happens, their scheme can not specify who the winners are, or even how many winners there are.

Concurrent to our work, Sakurai and Miyazaki proposed a publicly verifiable auction scheme [SM99]. Using the techniques of the convertible undeniable signature scheme[MS97], their scheme succeeds in hiding the losing bids without assuming any trusted centers. The drawback of their scheme is that all bidders must participate in the opening of the bids, by executing a disavowal protocol for each values they did not bid until one finds the highest bid. Thus both computational and round complexity of the bidders are high. Recently, in [KM99], Kobayashi and Morita proposed several auction schemes based on the techniques of the hash chains. It dramatically improves the computational complexity. However, their schemes either require high round complexity to bidders, or require the center *to know* the all bids in order to find the highest one, i.e., the scheme can not keep any bids secret from the center.

We take a totally different approach from all these schemes to achieve bid secrecy and verifiability. Compared to the scheme proposed in [KHT98][2], we allow an efficient representation for bids at the cost of computational complexity at the authorities: We require a bidder to post a bid which is an encryption of *one* message, and the authorities to work on it multiple times, where as the [KHT98] scheme requires a bidder to post multiple messages and the authorities to work on it once. In contrast to the scheme in [SM99], we employ a multiple of centers, who we trust to perform a threshold decryption. This setting helps to keep the round and computational complexity low for the bidders. Unlike the

[1] With a slight modification, the protocol can be made to open the lowest bid.

[2] We note that Kikuchi et al also aimed at achieving bidder privacy, which is to allow anonymous participation.

one-move auction scheme in [KM99], our scheme can keep the secrecy of the losing bids even from the centers.

Our approach involves a novel usage of encryption and a set of keys. We express a bid as an encryption of a *known* message, with the *key* to encrypt it corresponding to the value bid. Thus, what we hide in ciphertext is not the message that is encrypted, but the key used to encrypt it. The bid itself can be identified by finding the corresponding decrypting key that successfully decrypts to a given message.

The protocol proceeds as follows: each bidder signs and posts an encryption of his bid. At the opening stage, authorities try to find the largest value, the decrypting key of which successfully decrypts one of the submitted bids. Authorities release information on decrypting keys for the winning bid and for the higher values. Then anyone can successfully identify all the winners with the winning bid, and ensure that no one has bid any value higher than the winning bid. Furthermore, the algorithm prevents even the authorities from learning the losing bids.

We present two practical schemes, one based on the ElGamal cryptosystem and one based on the RSA cryptosystem.

2 Previous Work

2.1 The Protocol of [KHT98]

In this section, we describe an abstracted scheme which incorporates the ideas given in [KHT98]. For simplicity, we assume the winners to be the one who has bid the highest value among a set of L possible bid values, $V = \{v_1, \cdots, v_L\}$.

We describe how a bidder i with his identity information ID_i would bid a value $v_{b_i} \in V$. The encoding of the value v_{b_i} is represented as a vector of L components, where the first b_i components are independently encrypted ID_i's and the rest 0. We will call this a bid-vector A_i of a bidder i.

$$A_i[j] = \begin{cases} f_j(ID_i) & \text{if } b_i < j, \\ 0 & \text{otherwise} \end{cases}$$

(Here, f_j is an encryption function for j-th component.) The idea of finding the highest bid is as follows. Given bid-vectors of all the bidders, each elements in the same component are added to generate what we will call a sum-vector T.

$$T[j] = \sum A_i[j]$$

If the last component $T[L]$ is 0, it means no one bid the value v_L. If we search for $j = L, L-1, \ldots, 1$ and find the first non-zero value $T[j]$ at j=t, then the winning bid is v_t and the winner w is identified by performing $f_t^{-1}(T[t]) = ID_w$. This sum-vector is published for verification.

In order to keep bids secret from authorities, each element $A_i[j]$ in a bid-vector is distributed among authorities using secret sharing techniques[Sha79], which addition of shares yield addition of secrets.

Further, in order to prevent faulty bidder who tries to bid using other entity's ID, the authors of the paper suggest to use ID's which are secretly signed by the authorities.

2.2 Weaknesses in [KHT98]

The protocol described above has the following weaknesses:

- **Protocol failure in a tie case.**
  If there were two or more entities who submit the same highest bid t, the value $T[t]$ is an addition of multiple IDs that have been encrypted by f_t. There is no way to decompose this sum to recover original IDs, and thus the protocol fails in identifying the winners.
- **Leaking the second highest bid to a winner.**
  A winner would be able to detect the second highest bid, by scanning beyond the t-th component of the sum-vector to find the very next component which is not the encryption of his ID. This contradicts to the aim of revealing no information on the bids of losers.
- **Inefficient bid representation.**
  Each bid is represented in a vector of L elements, where L is a number of possible values that can be bid. Further, this long bid-vector is distributed among authorities, so that the representation in shares are proportional to number of authorities.
- **Anonymous interference.**
  Anyone can anonymously disturb an auction by submitting a random number r which does not decrypt to his ID.

In the following section, we present a protocol robust against any disturbance from malicious bidders. It successfully identifies all the winners and hides other losing bids, with the same minimum round complexity. Although the computation cost increases at the authorities, the protocol allows a bid to be represented in one probabilistic encryption.

3 The Basic Protocol

3.1 Outline

The basic idea behind our proposed protocol is to present a probabilistic encryption of bid v in such a way that it will not be decrypted unless v is the winning bid. For simplicity, we assume winners to be those who have bid the highest value among a set of L possible bid values, $V = \{v_1, \cdots, v_L\}$. We assume multiple authorities open the bids. If it is not necessary to keep bids secret from the authorities, then a setting with a single authority suffices. Note that this case remains nontrivial, since the authority must still prove that the highest bid is indeed the highest.

In order to achieve our goal, we employ a set of encryption functions $\{E_v\}$ and a set of decryption functions $\{D_v\}$ for $v \in V$. The ciphertext of a bid v will

be an encryption $E_v(M_v)$ for a predetermined value M_v. All encrypted bids from each bidder are signed and posted, and authorities will perform decryption to open *only* the highest bid. The opening procedure is as follows: the authorities first take the largest $v_L \in V$ and try to decode all the encypted bids one by one using D_{v_L}. If any of these decodes to a predetermined value M_{v_L}, it is an indication that the ciphertext was encrypted using E_{v_L}, and thus the bidder is a winner with a winning bid of v_L. If not, then the authorities take the next largest value v_{L-1} and continue until they find the largest v_t for which at least one of the encrypted bids indeed decodes to M_{v_t}.

3.2 Sets of Encryption and Decryption Functions

We require the following properties on the function sets $\{E_v\}$ and $\{D_v\}$ and a set of values $\{M_v\}$:

Property 1 *Indistinguishability*
Given any $E_v(M_v)$ and $E_{v'}(M_{v'})$ for $v, v' \in V$, a polynomial turing machine can not distinguish whether or not $v = v'$.

Property 2 *Incompatible decryption*
Given any $E_v(M_v)$ and $v' \in V$ that is equal to or larger than v, $D_{v'}(E_v(M_v)) = M_{v'}$ if and only if $v = v'$.

Property 3 *Independent decryption*
Given any $E_v(M_v)$ and $v' \in V$ that is strictly larger than v, $D_{v'}(E_v(M_v))$ does not give any information on v, except that $v \neq v'$.

Property 4 *Group decryption*
Each decryption function D_v is distributed among authorities, such that decryption is possible only through a collaboration of authorities forming a quorum. (This property is not necessary if keeping bids secret from the authorities is not required.)

Property 5 *Verifiable decryption*
Given $E(M)$ and M' and E_v, the authorities can supply a proof that proves that M' is a correct decryption of $E(M)$ under decryption function D_v.

Property 6 *Verifiable generation*
We require a means to verify that the sets $\{E_v\}, \{D_v\}$ and $\{M_v\}$ have been chosen to achieve the above properties.

3.3 Bidding and Opening

Given the set of functions satisfying the above properties, the auction proceeds as follows:

1. [Set-up] The authorities set up $\{E_v\}, \{D_v\}$ and $\{M_v\}$, where $\{E_v\}$ and $\{M_v\}$ are posted in a way that anyone can confirm their validity(Property 6).
2. [Bidding] Each bidder b with a bid v_b posts $C_b = E_{v_b}(M_{v_b})$ with his signature.
3. [Opening] The authorities

(a) set $k = L$ and decrypt $\{C_b\}_b$ using D_{v_k}.
(b) While $D_{v_k}(C_b) \neq M_{v_k}$ for all b, set $k = k - 1$.
(c) Publish $t = k$ as the winning bid and list all b s.t. $D_{v_t}(C_b) = M_{v_t}$ as winners $\{w_i\}$.
(d) Publish for all $j \geq t$, $D_{v_j}(C_b)$ with its proof of being correct(Property 5).

4. [Verification] Anyone may verifiy the following:
(a) $D_{v_j}(C_b)$ is the correct decryption regarding each D_{v_j}, for $j \geq t$.
(b) For all $v_j \neq v_t$, $D_{v_j}(C_b) \neq M_{v_j}$.
(c) For each of the winners w_i, $D_{v_t}(C_{w_i}) = M_{v_t}$ holds.

3.4 Discussions

We claim that the following properties are achieved in this protocol.

– *Correctness*
The winning bid is indeed the highest among all the bids. If there exists a k that is larger than the winning bid t, then the authorities should have stopped when scanning the bids with D_k(Property 2). Winners are those who submit the winning bid, as correctness of the opening is also guaranteed in Property 2.
– *Verifiability of the result*
Due to the Property 5, all entities can verify that the authorities performed the correct decryptions for $k \geq t$. Therefore, everyone can confirm that there is no bid higher than t and that the announced winners are the only ones who submitted the winning bid t.
– *Fairness*
In order to achieve fairness, we require our encryption to be non-malleable [DDN91]. Informally, this property ensures that seeing one bidder's encryption does not give another bidder an unfair advantage, say by generating an encryption of a bid that is one dollar more than the previous bid. We note that we can add nonmalleability by reencryption: If an encryption function E_v does not by itself achieve non-malleability, we can always reencrypt using some arbitrary non-malleable encryption. This encryption can be removed prior to decrypting via E_v.
We note that a nonmalleable encryption is still vulnerable to a replay attack, posting the same bid. Some fixes to this problems are: 1. Do not accept the second same bid, or 2. Encrypt $M_{v_b}||b$, where b is the identity of the bidder, or 3. Provide proof of ownership of the ciphertext that the bidder himself has generated it.
– *Non-repudiation*
The winners can not deny they submitted the winning bid, as there is a digital signature given to their encrypted bid which indeed decrypts properly.
– *Privacy of losing bid*
Due to Property 1, the value of bids is hidden in the encryption. Due to Property 4, the bids of losers will not be revealed in the course of opening,

that is, when the decryption functions are performed on them. Once the winning bid is found, then no further decryption trial will be performed on the losing bids, so they are never decrypted.

– *Robustness*
Even if an invalid bidder submits a meaningless encryption, the auction proceedings will be unaffected; the invalid bids are simply ignored.
– *Efficiency*
Bidders need to encrypt only once and submit only a single encryption. On the other hand, authorities need to perform as many as $(L - t + 1)B$ decryptions, where t denotes the winning bid v_t and B is the number of bidders. The authorities further need to publish proofs for decryptions, the cost of which varies depending on the implementation.[3]

4 Schemes Based on Practical Cryptosystems

In the following, we give two examples of the set of functions achieving the property discussed in Sect. 3.2. They are based on the ElGamal cryptosystem and the RSA cryptosystem, respectively.

ElGamal based ones require a list of public keys $\{E_v\}$ for each $v \in V$. On the other hand, RSA based ones allow bidders to generate $\{E_v\}$ from the value v, so this long list is not necessary. However, in the aspect of information necessary verify the decryption, the ElGamal based schemes require authorities to reveal only the corresponding secret keys where as the RSA based schemes require them to publish each decryption results. Further, the ElGamal based schemes are suited to prove that they achieve required properties such as indistinguishability and incompatible decryption property, where as we can only heuristically claim such in RSA based schemes. Procedures to distributedly generate keys among authorities are more complicated in the RSA scheme than the ElGamal scheme.

4.1 ElGamal Based Scheme

In this section, we use a set of encryptions based on the ElGamal cryptosystem [E84]. We assume a large prime p where $p-1$ has a large prime factor q is given by the authorities together with a generator $g \in Z_p^*$ over a subgroup of order q. Further, z_v is independently and disjointly generated for each $v \in V$, which will be secretly held by the authorities. For simplicity let $M_v = M$ for all v. We define E_v by

$$E_v(M) = (g^\alpha \bmod p, M \cdot h_v^\alpha \bmod p)$$

where $h_v = g^{z_v} \bmod p$ is a public parameter for value v and α is a non-zero random number in Z_q.

[3] For example, the ElGamal-based scheme in Subsect. 4.1 requires $L-t$ keys to be published, where as the RSA-based one in Subsect. 4.2 requires $(L-t+1)B$ decryption results.

Given $E(M) = (x, y)$, we define D_v by

$$D_v(E(M)) = y/x^{z_v} \bmod p$$

We will show that the above sets of functions fulfill our requirements.[4]

- [Property 1] *Indistinguishability*
  Given any $E_v(M) = (x, y)$ and $E_{v'}(M) = (x', y')$ for $v, v' \in V$, finding $v = v'$ or not is equivalent to determining whether quadruples $(x, x', y/M, y'/M)$ are random or $log_x(y/M) = log_{x'}(y'/M)$ holds. This is as difficult as the Diffie-Hellman Decision Problem(c.f. [CS98]).
- [Property 2] *Incompatible decryption*
  Given any $E_v(M)$ and $v' \in V$, $D_{v'}(E_v(M_v)) = M \cdot g^{\alpha \cdot (z_v - z_{v'})}$. This equals M only if or $z_v = z_{v'}$, since $\alpha \neq 0 \bmod q$. Since the z_v's are disjointly generated, it follows that $v = v'$.
- [Property 3] *Independent decryption*
  Given any $E_v(M)$ and $v' \in V$, $D_{v'}(E_v(M_v)) = g^{\alpha \cdot (z_v - z_{v'})}$ indeed does not reveal any information on v, except that $v \neq v'$, as long as z_v's are generated randomly.
- [Property 4] *Group decryption*
  The key pair (z, h) is constructed in a way that each authority receives a share z_i and is publicly committed to this share by $h_i = g^{z_i}$ [Ped91].
- [Property 5] *Verifiable decryption*
  Given $E(M) = (x, y)$ and h_v, authorities can publish the secret key z_v. Then one can perform decryption by themselves for any $E(M)$ by $M = y/(x^{z_v})$.
- [Property 6] *Verifiable generation*
  By an appropriate use of pseudo-random generators, we can confirm that the set $\{E_v\}, \{D_v\}$ and $\{M_v\}$ are chosen independently, and thus suffices the above properties.

4.2 RSA Based Schemes

In this subsection, we use a set of encryptions based on the RSA cryptosystem [RSA]. In contrast to the protocol described in the previous subsection where we need to publish a list of public parameters for all possible bid v, the protocol below allows bidders to generate public parameters for themselves.

We require the authorities to generate two large primes p and q where $p-1$ and $q-1$ can be represented as $2^k p'$ and $2^\ell q'$ respectively for large prime factors p' and q'. The product $N = p \cdot q$ is published, together with a cryptographically secure hash function H[5].

[4] We note it does not achieve non-malleable property as it is. A simple fix is to apply an encryption that is known to achieve non-malleability, on top of the encrypted bids. Another heuristic approach is to include bidder's ID in the message, e.g., $M|ID|hash(M|ID)$.

[5] For the defined encryption scheme to be secure against adpative chosen message attack, we require the hash function H to be division intractable[GHR99].

We define M_v to be

$$M_v = v|R|H(v|R)$$

where R is a random number of predermined bit length and | is a concatenation.[6] The encryption E_v is given by

$$E_v(M_v) = (v|R|H(v|R))^{H(v)|1} \bmod N.$$

The decryption algorithm D_v proceeds as follows:

1. First, compute an odd exponent $e_v = H(v)|1$. This exponent is almost certainly relatively prime to $LCM(p-1, q-1)$.
2. Compute $d_v = {e_v}^{-1} \bmod LCM(p-1, q-1)$.
3. Decrypt $E(M)$ by computing $\{E(M)\}^{d_v} \bmod N$. One can conclude $E(M)$ to be an encryption for the bid v if the decrypted output is conformant with $v|R|H(v|R)$.

We will informally argue that the above sets of functions and parameters should fulfill our requirements.

- [Property 1] *Indistinguishability*
  Given any $E_v(M_v)$ and $E_{v'}(M_{v'})$ for $v, v' \in V$, we know of no efficient way of determining whether or not $v = v'$ holds.
- [Property 2] *Incompatible decryption*
  Given any $E_v(M_v)$ and $v' \in V$, $D_{v'}(E_v(M_v))$ is not likely to yield a decrypted message M with prefix v' unless $v = v'$.
- [Property 3] *Independent decryption*
  Given any $E_v(M_v)$ and $v' \in V$, $D_{v'}(E_v(M_v))$ would not provide a useful information to reveal v, except that $v \neq v'$.
- [Property 4] *Group decryption*
  The methods to generate and share RSA keys in a distributed manner are studied in [FMY98,MS99].
- [Property 5] *Verifiable decryption*
  For each $E(M)$ and E_v, the authorities can reveal $M_v = D_v(E(M))$. It is easy to verify that M_v is indeed a decryption of $E(M)$ under D_v by simply checking if $(M_v)^{H(v)|1} = E(M)$, and M_v conforms to $v|R|H(v|R)$. Note that on contrast to the ElGamal based schemes, we can not give away d_v which is a decryption exponent. A pair d_v and e_v gives prime factors of N which makes all encryption decrypted.
- [Property 6] *Verifiable generation*
  The sets $\{E_v\}, \{D_v\}$ and $\{M_v\}$ suffice above properties, if we can verify that the RSA keys are properly generated in a distributed manner.

[6] R can contain the bidder's ID for non-malleability purpose.

5 Conclusion

In this paper we proposed an auction protocol which hides the bids of non-winners, and still makes it possible to publicly verify that the winning bid was indeed the highest submitted. Our protocol enjoys minimum round complexity for bidders and does not suffer from any problems in case of a tie bid, which was the case in previous work. Moreover, our protocol is efficient for bidders in that it requires only a single encryption for each bid, whereas in previous work the length of encoded bid was required to be linear to the number of possible bid values. The uniqueness of our construction is that we hide the index of keys used in encryption rather than the message.

We gave two concrete examples of schemes, one based on the ElGamal cryptosystem, and the other on the RSA cryptosystem.

Acknowledgements

The author would like to thank Joe Kilian for his valuable comments to this work, including the suggestion of the RSA-variant of the proposed scheme. She also thanks Hiroaki Kikuchi and Shingo Miyazaki for the discussions on this topic. Her thanks also goes to the anonymous referees for the invaluable comments on the paper.

References

[CS98] Ronald Cramer and Victor Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *Advances in Cryptology — CRYPTO '98*, volume 1462 of *Lecture Notes in Computer Science*, pages 13–25, 1998. 429

[DDN91] D. Dolev, C. Dwork, and M. Naor. Non-malleable cryptography. In *STOC '91*. 427

[E84] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In *Advances in Cryptology — CRYPTO '84*, volume 196 of *Lecture Notes in Computer Science*, pages 10–18, 1984. 428

[FMY98] Y. Frankel, P.MacKenzie and M. Yung, *Robust efficient distributed RSA-key generation.* in *STOC 98*, pp.663-672,1998. 430

[FR96] Matthew Franklin and Michael Reiter. *The design and implementation of a secure auction service.* In *IEEE Transactions on Software Engineering* , No.22, Vol.5, pages 302-312, 1996. 422

[GHR99] Rosario Gennaro, Shai Halevi and Tal Rabin. *Secure hash-and-sign signatures without the random oracle.* In *Eurocrypt 99*, 1999. 429

[HTK98] Michael Harkavy, Doug Tyger, Hiroaki Kikuchi. *Electronic auctions with private bids.* In *Third USENIX Workshop on Electronic Commerce*, 1998. 423

[KHT98] Hiroaki Kikuchi, Michael Harkavy, Doug Tyger. *Multi-round anonymous auction protocols.* In *IEEE Workshop on Dependable and Real-Time E-Commerce System* 1998. 422, 423, 424, 425

[KM99] Kunio Kobayashi and Hikaru Morita. *Efficient sealed-bid auction with quantitative competition using one-way functions.* In *Technical Report of IEICE, ISEC* May 1999. 423, 424

[MS97] M. Michels and M. Stadler. *Efficient convertible undeniable signature schemes.* In *Proc. 4th Annual Workshop on Selected Areas in Cryptograph, SAC'97*,1997. 423

[MS99] Shingo Miyazaki and Kouichi Sakurai. *Notes on threshold schemes in the distributed RSA Cryptosystem.* In *the 1999 Symposium on Cryptography and Information Security*, pp.451–456, 1998. 430

[Ped91] Torben P. Pedersen. A threshold cryptosystem without a trusted party (extended abstract). In *Advances in Cryptology — EUROCRYPT '91*, volume 547 of *Lecture Notes in Computer Science*, pages 522–526, 1991. 429

[RSA] Ronald Rivest, Adi Shamir and Len Adleman. *A method for obtaining digital signature and public-key cryptosystems.* In *Communications of the ACM* , Vol.21, No.2,pages 120–126. 1978. 429

[SM99] Kouichi Sakurai and Shingo Miyazaki. *A bulletin-board based digital auction scheme with bidding down strategy –towards anonymous electronic bidding without anonymous channels nor trusted centers* in *Cryptographic Techniques and E-Commerce, Proceedings of the 1999 International Workshop on Cryptographic Techniques and E-Commerce(CryTEC '99)* 1999. 423

[Sha79] Adi Shamir. How to share a secret. *Communications of the ACM*, 22:612–613, 1979. 424