行政院國家科學委員會專題研究計畫　期中進度報告

基於忘卻式傳輸協定的安全計算(1/3)

計畫類別：個別型計畫
計畫編號：NSC94-2213-E-009-116-
執行期間：94 年 08 月 01 日至 95 年 07 月 31 日
執行單位：國立交通大學資訊科學學系(所)

計畫主持人：曾文貴

計畫參與人員：朱成康、胡智明、陳冠廷、張君偉、陳仕烽

報告類型：精簡報告
報告附件：出席國際會議研究心得報告及發表論文
處理方式：本計畫可公開查詢

中　華　民　國 95 年 6 月 6 日

行政院國家科學委員會補助專題研究計畫 □ 成 果 報 告
☑期中進度報告

基於忘卻式傳輸協定的安全計算（1/3）

Secure Multiparty Computation Based on Oblivious Transfer Protocols

計畫類別：☑ 個別型計畫　□整合型計畫
計畫編號：NSC　94－2213－E－009－116－
執行期間：　94 年　8　月　1　日至　95　年　7　月　31　日

計畫主持人：曾文貴 教授
計畫參與人員：　朱成康、胡智明、陳冠廷、張君偉、陳仕烽

成果報告類型(依經費核定清單規定繳交)：☑精簡報告　□完整報告

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外，得立即公開查詢
□涉及專利或其他智慧財產權，□一年□二年後可公開查詢

執行單位：國立交通大學 資訊科學系

中　華　民　國　95　年　5　月　31 日

中文摘要

忘卻式傳輸機制 (OT) 是密碼學上很重要的一個基本元件，許多密碼學的應用都會利用忘卻式傳輸來確保送方與收方都達到設定的安全條件，OT 是密碼學裡的完全性密碼元件，只要有安全的 OT 協定就可以達成任何密碼學上的多人安全計算問題。本計畫研究 OT 的相關問題。

關鍵詞：分散式門檻密碼、預防式密碼、安全模式。

英文摘要

Oblivious transfer (OT) is an important cryptographic primitive. Many cryptographic applications, such as private information retrieval, can be achieved by basing on the OT protocol. The OT protocol is complete in the sense that every multi-party secure computation of a polynomial-time computable function can be realized by using the OT protocol only. In this project, we study the related issues about OT.

Keywords: Oblivious transfer, bounded storage model, secure multiparty computation.

一、計畫緣起及目的

忘卻式傳輸（Oblivious Transfer，OT）機制包含了傳送者 Sender（S）和接收者 Receiver（R）兩方，S 擁有一些秘密資訊 m0, m1,…，R 想要透過與 S 交換一些訊息而得到其中的一個秘密資訊 mσ，OT 機制保證不會讓 S 知道 R 的選擇 σ，同時 R 也不會得到其他沒有選到的秘密資訊。OT 機制是密碼學上很重要的一個基本元件，許多密碼學的應用都會利用忘卻式傳輸來確保送方與收方都達到設定的安全條件，例如在電子商務的的應用當中，商務網站可藉由這樣的機制販售付費資訊，如 mp3 音樂的下載，使用者可以選擇其想要購買的歌曲，並且保有隱私性，不讓網站知道其所選擇購買的音樂為何，網站也可藉由此機制保障其他未付費的資訊不會洩漏給使用者。

OT 是密碼學裡的完全性密碼元件（complete cryptographic primitive），只要有安全的 OT 協定就可以達成任何密碼學上的多人安全計算（multi-party secure computation）問題。所謂多人安全計算是指多人參與的密碼安全協定，有一公開的函數 f，每一參與者 Pi 有一秘密值 xi，他們要透過公開交換訊息的方式函數值 y=f(x1,x2,…)，最終所有的參與者都得到 y 值；對任何參與者 Pi 而言，除了可以由 y 和 xi 計算出的資訊之外，Pi 得不到其他參與者 Pj 的秘密值 xj 的其他資訊。由多人安全計算的定義可以看出任何密碼協定都可以套用這個模式，例如，在安全電子投票系統裡，最後要計算出票數，但又不要洩漏個別投票者所投的票，假設是投贊成與反對，每一投票 Vi 者貢獻 xi∈{0,1}（0 表反對，1 表贊成），安全電子投票系統就是由所有的投票者安全地計算出 f(x1,x2,…)=x1+x2+…。又如在雙人相互身份認證的問題裡，Alice (P1) 的私密與公開資訊為 x1 與 y1，Bob (P2) 的私密與公開資訊為 x2 與 y2，Alice 與 Bob 的相互身份認證就是計算 fy1,y2(x1,x2)=1 if and only if (y1,x1) 與 (y2,x2) 分別為成對的公開與私密資訊。

本計畫的目的有下列幾項：（1）研究以 OT 來直接建構安全的計算函數，例如比較兩個數的大小，判斷一數 x 是否落於某一區間 [a, b] 等，我們希望能夠將 query language 裡所需的運算皆以 OT 直接實現；本計畫將實做我們研究出的成果，我們希望實做出基於 OT 的安全 query language，達到保障使用者與資料庫擁有者的隱私與安全。（2）我們打算研究 k-out-of-n OT 機制，我們認為目前的方法還不夠好，應該可以達到更佳的回合數及訊息數。（3）我們將研究攻擊者限制模式

下的 OT，目前已知協定的主要缺點是收方與送方皆須使用至少 $O(n^{1/2})$ 空間，還未達到可行的門檻，我們將盡力尋找只使用 O(log n) 空間的 OT 協定並嚴格證明之。

二、研究成果

本年度(第一年度)的研究成果如下：

1. 我們提出 k-out-of-n 的 OT 協定，並嚴格證明其安全性，這是到目前為止最有效率的 k-out-of-n OT 協定。這篇論文發表在『C.-K. Chu, W.-G. Tzeng. Efficient k-out-of-n oblivious transfer schemes with adaptive and non-adaptive queries. In *Proceedings of International Workshop on Practice and Theory in Public-Key Cryptography (PKC 05),* Lecture Notes in Computer Science 3386, pp.172-183, 2005』。論文請見附件。
2. 我們利用 OT 的精神設計出 password-based anonymous authentication 協定，並把它應用在安全的資料庫存取上，這篇論發表在『W.-G. Tzeng. A secure system for data access based on anonymous authentication and time-dependent hierarchical keys. In *Proceedings of ACM Symposium on Information, Computer and Communications Security 06 (ASIACCS 06),* ACM Press, 2006.』中。論文請見附件。
3. 我們還將 OT 推廣為 Conditional oblivious cast (COC)，這篇論文發表在『C.-K. Chu, W.-G. Tzeng. Conditional oblivious cast. In *Proceedings of International Workshop on Practice and Theory in Public-Key Cryptography (PKC 06),* Lecture Notes in Computer Science 3958, pp.443-457, 2006』中。論文請見附件。

三、計畫成果自評

我們的研究結果發表了三篇會議論文，兩篇在高水準的 PKC05 國際會議，另一篇在 ACM ASIACCM 上。目前還有論文在撰寫中，以成果來看，我們達成了本計畫的目的。

參考文獻

1. William Aiello, Yuval Ishai, Omer Reingold: Priced Oblivious Transfer: How to Sell Digital Goods. EUROCRYPT 2001: 119-135.
2. Donald Beaver: How to Break a "Secure" Oblivious Transfer Protocol. EUROCRYPT 1992: 285-296.
3. Ian F. Blake, Vladimir Kolesnikov: Strong Conditional Oblivious Transfer and Computing on Intervals. ASIACRYPT 2004: 515-529.
4. Mihir Bellare and Silvio Micali. Non-Interactive Oblivious Transfer and Applications. In Proceedings of Advances in Cryptology - CRYPTO '89, volume 435 of LNCS, pages 547-557. Springer-Verlag, 1989.
5. Gilles Brassard, Claude Crepeau and Jean-Marc Robert. All-or-Nothing Disclosure of Secrets. In Proceedings of Advances in Cryptology - CRYPTO '86, volume 263 of LNCS, pages 234-238. Springer-Verlag, 1986.
6. Christian Cachin, Claude Crépeau, Julien Marcil: Oblivious Transfer with a Memory-Bounded Receiver. FOCS 1998: 493-502.
7. Yan-Cheng Chang, Chi-Jen Lu: Oblivious Polynomial Evaluation and Oblivious Neural Learning. ASIACRYPT 2001: 369-384.
8. Claude Crépeau, Jeroen van de Graaf, Alain Tapp: Committed Oblivious Transfer and Private Multi-Party Computation. CRYPTO 1995: 110-123.
9. Yan Zong Ding: Oblivious Transfer in the Bounded Storage Model. CRYPTO 2001: 155-170.
10. Shimon Even, Oded Goldreich and Abraham Lempel. A Randomized Protocol for Signing Contracts. Communications of the ACM, 28(6):637-647, 1985.
11. J.A. Garay, P.D. MacKenzie, Concurrent oblivious transfer, FOCS 2000, pp.314-324, 2000.

12. Y. Gertner, Y. Ishai, E. Kushilevitz, T. Malkin, Protecting data privacy in private data retrieval schemes, STOC 98, pp.151-160, 1998.
13. O. Goldreich, S. Micali, A. Wigderson. Proofs that yield nothing but the validity of the assertion and a methodology of cryptographic protocol design. FOCS 86, pp.174-187, 1986.
14. O. Goldreich, S. Micali, A. Wigderson. How to play any mental game. SOTFC 87, pp.218-229, 1987.
15. Dowon Hong, Ku-Young Chang, Heuisu Ryu. Efficient Oblivious Transfer in the Bounded-Storage Model. ASIACRYPT 2002: 143-159.
16. Yuval Ishai, Joe Kilian, Kobbi Nissim, Erez Petrank: Extending Oblivious Transfers Efficiently. CRYPTO 2003: 145-161.
17. Joe Kilian: Founding Cryptography on Oblivious Transfer. STOC 1988: 20-31.
18. Yi Mu, Junqi Zhang and Vijay Varadharajan. m out of n Oblivious Transfer. In Proceedings of the 7th Australasian Conference on Information Security and Privacy (ACISP '02), volume 2384 of LNCS, pages 395-405. Springer-Verlag, 2002.
19. Ueli M. Maurer. Conditionally-Perfect Secrecy and a Provably-Secure Randomized Cipher. J. Cryptology 5(1): 53-66 (1992).
20. Moni Naor and Benny Pinkas. Oblivious Transfer with Adaptive Queries. In Proceedings of Advances in Cryptology - CRYPTO '99, volume 1666 of LNCS, pages 573-590. Springer-Verlag, 1999.
21. M. Naor, B. Pinkas. Oblivious transfer and polynomial evaluation, STOC 99, pp.145-254, 1999.
22. M. Naor, B.Pinkas. Distributed oblivious transfer, Asiacrypt 00, Lecture Notes in Computer Science 1976, pp.205-219, Springer-Verlag, 2000.
23. M. Naor, B. Pinkas. Efficient oblivious transfer protocols, In Proceedings of the 12th Annual Symposium on Discrete Algorithms, pp.448-457, 2001.
24. Wakaha Ogata and Kaoru Kurosawa. Oblivious Keyword Search. Cryptology ePrint Archive: Report 2002/182, IACR, 2002.
25. Michael O. Rabin. How to exchange secrets by oblivious transfer. Technical Report TR-81, Aiken Computation Laboratory, Harvard University, 1981.
26. J.P. Stern. A new and efficient all-or-nothing disclosure of secrets protocol, Asiacrypt 98, Lecture Notes in Computer Science 1514, pp.357-371, Springer-Verlag, 1998.
27. Wen-Guey Tzeng. Efficient 1-out-of-n oblivious transfer schemes with universally reusable parameters, IEEE Transactions on Computers 53(2), pp.232-240, 2004.
28. Qian-Hong Wu, Jian-Hong Zhang, Yu-Min Wang: Practical t-out-n Oblivious Transfer and Its Applications. ICICS 2003: 226-237.
29. A. Yao. How to generate and exchange secrets. FOCS 86, pp.162-167, 1986.

Efficient k-Out-of-n Oblivious Transfer Schemes with Adaptive and Non-adaptive Queries

Cheng-Kang Chu and Wen-Guey Tzeng

Department of Computer and Information Science,
National Chiao Tung University,
Hsinchu, Taiwan 30050
{ckchu,tzeng}@cis.nctu.edu.tw

Abstract. In this paper we propose efficient two-round k-out-of-n oblivious transfer schemes, in which R sends $O(k)$ messages to S, and S sends $O(n)$ messages back to R. The computation cost of R and S is reasonable. The choices of R are unconditionally secure. For the basic scheme, the secrecy of unchosen messages is guaranteed if the Decisional Diffie-Hellman problem is hard. When $k = 1$, our basic scheme is as efficient as the most efficient 1-out-of-n oblivious transfer scheme. Our schemes have the nice property of *universal parameters*, that is each pair of R and S need neither hold any secret key nor perform any prior setup (initialization). The system parameters can be used by all senders and receivers without any trapdoor specification. Our k-out-of-n oblivious transfer schemes are the most efficient ones in terms of the communication cost, in both rounds and the number of messages.
Moreover, one of our schemes can be extended in a straightforward way to an *adaptive* k-out-of-n oblivious transfer scheme, which allows the receiver R to choose the messages one by one adaptively. In our adaptive-query scheme, S sends $O(n)$ messages to R in one round in the commitment phase. For each query of R, only $O(1)$ messages are exchanged and $O(1)$ operations are performed. In fact, the number k of queries need not be pre-fixed or known beforehand. This makes our scheme highly flexible.

Keywords: k-out-of-n Oblivious Transfer, Adaptive Oblivious Transfer

1 Introduction

Oblivious transfer (OT) is an important primitive used in many cryptographic protocols [GV87,Kil88]. An oblivious transfer protocol involves two parties, the sender S and the receiver R. S has some messages and R wants to obtain some of them via interaction with S. The security requirement is that S wants R to obtain the message of his choice only and R does not want S to know what he chooses. The original OT was proposed by Rabin [Rab81], in which S sends a message to R, and R gets the message with probability 0.5. On the other hand, S does not know whether R gets the message or not. Even, et al. [EGL85] suggested a more general scheme, called 1-out-of-2 OT (OT_2^1). In this scheme, S

S. Vaudenay (Ed.): PKC 2005, LNCS 3386, pp. 172–183, 2005.

has two messages m_1 and m_2, and would like R to obtain exactly one of them. In addition, S remains oblivious to R's choice. Brassard, et al. [BCR86] further extended OT_2^1 to 1-out-of-n OT (OT_n^1) for the case of n messages.

Oblivious transfer has been studied extensively and in many flavors. Most of them consider the case that R chooses one message. In this paper we are concerned about the case that R chooses many messages at the same time. A k-out-of-n OT (OT_n^k) scheme is an OT scheme in which R chooses k messages at the same time, where $k < n$. A straightforward solution for OT_n^k is to run OT_n^1 k times independently. However, this needs k times the cost of OT_n^1. The communication cost is two-round, $O(k)$ messages from R to S, and $O(kn)$ messages from S to R even using the most efficient OT_n^1 schemes [NP01,Tze02].

Oblivious transfer with adaptive queries (Adpt-OT) allows R to query the messages one by one adaptively [NP99a]. For the setting, S first commits the messages to R in the commitment phase. Then, in the transfer phase, R makes queries of the messages one by one. The cost is considered for the commitment and transfer phases, respectively. It seems that the adaptive case implies the non-adaptive case. But, the non-adaptive one converted from an adaptive one usually needs more rounds (combining the commitment and transfer phases), for example, the scheme in [OK02]. Since our scheme needs no trapdoors, there is no entailed cost due to conversion. Adaptive OT_n^k is natural and has many applications, such as oblivious search, oblivious database queries, private information retrieval, etc.

In this paper we propose efficient two-round OT_n^k schemes, in which R sends $O(k)$ messages to S, and S sends $O(n)$ messages back to R. The computation cost of R and S is reasonable. The choices of R are unconditionally secure. For the basic scheme, the secrecy of unchosen messages is guaranteed if the Decisional Diffie-Hellman (DDH) problem is hard. When $k = 1$, our scheme is as efficient as the one in [Tze02]. Our schemes have the nice property of universal parameters, that is, each pair of R and S need neither hold any secret key nor perform any prior setup (initialization). The system parameters can be used by all senders and receivers without any trapdoor specification. Our OT_n^k schemes are the most efficient one in terms of the communication cost, either in rounds or the number of messages.

Moreover, one of our schemes can be extended in a straightforward way to an Adpt-OT_n^k scheme. In our adaptive-query scheme, S sends $O(n)$ messages to R in one round in the commitment phase. For each query of R, only $O(1)$ messages are exchanged and $O(1)$ operations are performed. In fact, the number k of queries need not be fixed or known beforehand. This makes our scheme highly flexible.

1.1 Previous Work and Comparison

Rabin [Rab81] introduced the notion of OT and presented an implementation to obliviously transfer one-bit message, based on quadratic roots modulo a composite. Even, Goldreich and Lempel [EGL85] proposed an extension of bit-OT_2^1, in which m_1 and m_2 are only one-bit. Brassard, Crépeau and Robert [BCR86]

proposed OT_n^1 soon after in the name "all-or-nothing disclosure of secrets" (ANDOS). After that, OT_n^1 has become an important research topic in cryptographic protocol design. Some OT_n^1 schemes are built by invoking basis OT_2^1 several times [BCR87,BCS96,NP99b], and the others are constructed directly from basic cryptographic techniques [SS90,NR94,Ste98,NP01,Tze02]. Some OT_n^1 schemes derived from computational private information retrieval (CPIR) have polylogarithmic communication cost [Lip04]. Nevertheless, the privacy of the receiver's choice is computationally secure. Besides, there are various oblivious transfer schemes developed in different models and applications, such as OT in the bounded storage model [CCM98,Din01], distributed OT [NP00,BDSS02], Quantum OT [BBCS91,CZ03], and so on. Lipmaa [Lip] provided a good collection of these works.

For OT_n^k, Bellare and Micali [BM89] proposed an OT_n^{n-1} scheme. Naor and Pinkas [NP99b] proposed a non-trivial OT_n^k scheme. The scheme invokes a basis OT_2^1 scheme $O(wk \log n)$ times, where $w > \log \delta / \log(k^4/\sqrt{n})$ and δ is the probability that R can obtain more than k messages. The scheme works only for $k \leq n^{1/4}$. After then, they also took notice of adaptive queries and provided some Adpt-OT_n^k schemes [NP99a]. In one scheme (the two-dimensional one), each query needs invoke the basis $\mathrm{OT}_{\sqrt{n}}^1$ scheme twice, in which each invocation of $\mathrm{OT}_{\sqrt{n}}^1$ needs $O(\sqrt{n})$ initialization work. In another scheme, each adaptive query of messages need invoke the basis OT_1^2 protocol $\log n$ times. Mu, Zhang, and Varadharajan [MZV02] presented some efficient OT_n^k schemes[1]. These schemes are designed from cryptographic functions directly. The most efficient one is a non-interactive one. To be compared fairly, the setup phase of establishing shared key pairs of a public-key cryptosystem should be included. Thus, the scheme is two-round and R and S send each other $\mathrm{O}(n)$ messages. However, the choices of R cannot be made adaptive since R's choices are sent to S first and the message commitments are dependent on the choices. Recently, Ogata and Kurosawa [OK02] proposed an efficient adaptive OT scheme based on the RSA cryptosystem. Each S needs a trapdoor (the RSA modulus) specific to him. The scheme is as efficient as our Adpt-OT_n^k scheme. But, if the adaptive OT scheme is converted to a non-adaptive one, it needs 3 rounds (In the first round, S sends the modulus N to R).

Ishai, Kilian, Nissim and Petrank [IKNP03] proposed some efficient protocols for extending a small number of OT's to a large number of OT's. Chen and Zhu [CZ03] provided an OT_n^k in the quantum computation model. We won't compare these schemes with ours since they are in different categories.

In Table 1 we summarize the comparison of our, Mu, Zheng, and Varadharajan's, and Naor and Pinkas's OT_n^k schemes. In Table 2 we summarize the comparison of our and Naor and Pinkas's Adpt-OT_n^k schemes.

[1] Yao, Bao, and Deng [YBD03] pointed out some security issues in [MZV02].

Table 1. Comparison of OT_n^k schemes in communication cost.

	Ours (this paper)	Mu, et al. [MZV02]	Naor, et al. [NP99b]
rounds	2	2	$O(wk \log n)$
messages ($R \to S$)	$O(k)$	$O(n)$	$O(wk \log n))$
messages ($S \to R$)	$O(n)$	$O(n)$	$O(n + wk \log n)$
universal parameters	Yes	Yes	No (need setup)
made to adaptiveness	Yes (OT_n^k-II)	No	Yes

Table 2. Comparison of Adpt-OT_n^k schemes in communication cost.

		Ours (this paper)	2-dimensional one, Naor, et al. [NP99a]	OT_n^k, Ogata, et al.[OK02]
commitment phase	rounds	1	1	1
	messages	$O(n)$	$O(n)$	O(n)
transfer phase	rounds	2	3*	2
	messages	$O(1)$	$O(\sqrt{n})$**	O(1)

* Two invocations of $\mathrm{OT}_{\sqrt{n}}^1$ in parallel.

** Use the most round-efficient $\mathrm{OT}_{\sqrt{n}}^1$ scheme as the basis.

2 Preliminaries

Involved Parties. The involved parties of an OT scheme is the sender and receiver. Both are polynomial-time-bounded probabilistic Turing machines (PPTM). A party is semi-honest (or passive) if it does not deviate from the steps defined in the protocol, but tries to compute extra information from received messages. A party is malicious (or active) if it can deviate from the specified steps in any way in order to get extra information.

A malicious sender may cheat in order or content of his possessed messages. To prevent the cheat, we can require the sender to commit the messages in a bulletin board. When the sender sends the encrypted messages to the receiver during execution of an OT scheme, he need tag a zero-knowledge proof of showing equality of committed messages and encrypted messages. However, in most applications, the sender just follows the protocol faithfully. Therefore, we consider the semi-honest sender only and the semi-honest/malicious receiver.

Indistinguishability. Two probability ensembles $\{X_i\}$ and $\{Y_i\}$, indexed by i, are (computationally) indistinguishable if for any PPTM D, polynomial $p(n)$ and sufficiently large i, it holds that

$$|\Pr[D(X_i) = 1] - \Pr[D(Y_i) = 1]| \leq 1/p(i).$$

Correctness of a Protocol. An OT scheme is correct if the receiver obtains the messages of his choices when the sender with the messages and the receiver with the choices follow the steps of the scheme.

Security Model. Assume that S holds n messages $m_1, m_2, \ldots, m_n$ and R's k choices are $\sigma_1, \sigma_2, \ldots, \sigma_k$. Note that only semi-honest sender is considered. We say that two sets C and C' are different if there is x in C, but not in C', or vice versa. An OT_n^k scheme with security against a semi-honest receiver should meet following requirements:

1. Receiver's privacy – indistinguishability: for any two different sets of choices $C = \{\sigma_1, \sigma_2, \ldots, \sigma_k\}$ and $C' = \{\sigma'_1, \sigma'_2, \ldots, \sigma'_k\}$, the transcripts, corresponding to C and C', received by the sender are indistinguishable. If the received messages of S for C and C' are identically distributed, the choices of R are unconditionally secure.
2. Sender's security – indistinguishability: for any choice set $C=\{\sigma_1,\sigma_2,\ldots,\sigma_k\}$, the unchosen messages should be indistinguishable from the random ones.

An OT_n^k scheme with security against a malicious receiver should meet following requirements:

1. Receiver's privacy – indistinguishability: the same as the case of the semi-honest receiver.
2. Sender's security – compared with the Ideal model: in the Ideal model, the sender sends all messages and the receiver sends his choices to the trusted third party (TTP). TTP then sends the chosen messages to the receiver. This is the securest way to implement the OT_n^k scheme. The receiver R cannot obtain extra information from the sender S in the Ideal model. We say that the sender's security is achieved if for any receiver R in the real OT_n^k scheme, there is another PPTM R' (called simulator) in the Ideal model such that the outputs of R and R' are indistinguishable.

Computational Model. Let G_q be a subgroup of Z_p^* with prime order q, and $p = 2q+1$ is also prime. Let g be a generator of G_q. We usually denote $g^x \bmod p$ as g^x, where $x \in Z_q$. Let $x \in_R X$ denote that x is chosen uniformly and independently from the set X.

Security Assumptions. For our OT_n^k schemes against semi-honest and malicious receiver, we assume the hardness of Decisional Diffie-Hellman (DDH) problem and Chosen-Target Computational Diffie-Hellman (CT-CDH) problem, respectively.

Assumption 1 (Decisional Diffie-Hellman (DDH)). *Let $p = 2q + 1$ where p, q are two primes, and G_q be the subgroup of Z_p^* with order q. The following two distribution ensembles are computationally indistinguishable:*

- $Y_1 = \{(g, g^a, g^b, g^{ab})\}_{G_q}$, *where g is a generator of G_q, and $a, b \in_R Z_q$.*
- $Y_2 = \{(g, g^a, g^b, g^c)\}_{G_q}$, *where g is a generator of G_q, and $a, b, c \in_R Z_q$.*

For the scheme against malicious receiver, we use the assumption introduced by Boldyreva [Bol03], which is analogous to the chosen-target RSA inversion assumption defined by Bellare, et al. [BNPS01].

- System parameters: (g, h, G_q);
- S has messages: $m_1, m_2, \ldots, m_n$;
- R's choices: $\sigma_1, \sigma_2, \ldots, \sigma_k$;

1. R chooses two polynomials $f(x) = a_0 + a_1 x + \cdots + a_{k-1}x^{k-1} + x^k$ and $f'(x) = b_0 + b_1 x + \cdots + b_{k-1}x^{k-1} + x^k$ where $a_0, a_1, \ldots, a_{k-1} \in_R Z_q$ and $b_0 + b_1 x + \cdots + b_{k-1}x^{k-1} + x^k \equiv (x - \sigma_1)(x - \sigma_2)\cdots(x - \sigma_k) \bmod q$.
2. $R \longrightarrow S : A_0 = g^{a_0}h^{b_0}, A_1 = g^{a_1}h^{b_1}, \ldots, A_{k-1} = g^{a_{k-1}}h^{b_{k-1}}$.
3. S computes $c_i = (g^{k_i}, m_i B_i^{k_i})$ where $k_i \in_R Z_q^*$ and $B_i = g^{f(i)}h^{f'(i)} = A_0 A_1^i \cdots A_{k-1}^{i^{k-1}} (gh)^{i^k} \bmod p$, for $i = 1, 2, \ldots, n$.
4. $S \longrightarrow R$: $c_1, c_2, \ldots, c_n$.
5. Let $c_i = (U_i, V_i)$, R computes $m_{\sigma_i} = V_{\sigma_i}/U_{\sigma_i}^{f(\sigma_i)} \bmod p$ for each σ_i.

Fig. 1. OT_n^k-I: k-out-of-n OT against semi-honest receiver.

Assumption 2 (Chosen-Target Computational Diffie-Hellman (CT-CDH)). *Let G_q be a group of prime order q, g be a generator of G_q, $x \in_R Z_q^*$. Let $H_1 : \{0,1\}^* \to G_q$ be a cryptographic hash function. The adversary A is given input (q, g, g^x, H_1) and two oracles: target oracle $T_G(\cdot)$ that returns a random element $w_i \in G_q$ at the i-th query and helper oracle $H_G(\cdot)$ that returns $(\cdot)^x$. Let q_T and q_H be the number of queries A made to the target oracle and helper oracle respectively. The probability that A outputs k pairs $((v_1, j_1), (v_2, j_2), \ldots, (v_k, j_k))$, where $v_i = (w_{j_i})^x$ for $i \in \{1, 2, \ldots, k\}$, $q_H < k \leq q_T$, is negligible.*

3 k-Out-of-n OT Schemes

We first present a basic OT_n^k scheme for the semi-honest receiver in the standard model. Then, we modify the scheme to be secure against the malicious receiver in the random oracle model. Due to the random oracle model, the second scheme is more efficient in computation.

3.1 k-Out-of-n OT Against Semi-honest Receiver

The sender S has n secret messages $m_1, m_2, \ldots, m_n$. Without loss of generality, we assume that the message space is G_q, that is, all messages are in G_q. The semi-honest receiver R wants to get $m_{\sigma_1}, m_{\sigma_2}, \ldots, m_{\sigma_k}$. The protocol OT_n^k-I with security against the semi-honest receiver is depicted in Figure 1.

For system parameters, let g, h be two generators of G_q where $\log_g h$ is unknown to all, and G_q be the group with some descriptions. These parameters can be used repeatedly by all possible senders and receivers as long as the value $\log_g h$ is not revealed. Therefore, (g, h, G_q) are universal parameters.

The receiver R first constructs a k-degree polynomial $f'(x)$ such that $f'(i) = 0$ if and only if $i \in \{\sigma_1, \ldots, \sigma_k\}$. Then R chooses another random k-degree polynomial $f(x)$ to mask the chosen polynomial $f'(x)$. The masked choices $A_0, A_1, \ldots, A_{k-1}$ are sent to the sender S.

When S receives these queries, he first computes $B_i = g^{f(i)}h^{f'(i)}$ by computing $A_0 A_1^i \cdots A_{k-1}^{i^{k-1}} (gh)^{i^k} \bmod p$. Because of the random polynomial $f(x)$, S does not know which $f'(i)$ is equal to zero, for $i = 1, 2, \ldots, n$. Then S treats B_i as the public key and encrypts each message m_i by the ElGamal cryptosystem. The encrypted messages $c_1, c_2, \ldots, c_n$ are sent to R.

For each $c_i, i \in \{\sigma_1, \sigma_2, \ldots, \sigma_k\}$, since $B_i = g^{f(i)}h^{f'(i)} = g^{f(i)}h^0 = g^{f(i)}$, R can get these messages by the decryption of ElGamal cryptosystem with secret key $f(i)$. If $i \notin \{\sigma_1, \sigma_2, \ldots, \sigma_k\}$, since R can not compute $(g^{f(i)}h^{f'(i)})^{k_i}$ with the knowledge of g^{k_i} and $f(i), f'(i)$ only, the message m_i is unknown to R.

Correctness. Let $c_i = (U_i, V_i)$, we can check that the chosen messages m_{σ_i}, $i = 1, 2, \ldots, k$, are computed as

$$\begin{aligned} V_{\sigma_i}/U_{\sigma_i}^{f(\sigma_i)} &= m_{\sigma_i} \cdot (g^{f(\sigma_i)}h^{f'(\sigma_i)})^{k_{\sigma_i}}/g^{k_{\sigma_i}f(\sigma_i)} \\ &= m_{\sigma_i} \cdot (g^{f(\sigma_i)} \cdot 1)^{k_{\sigma_i}}/g^{k_{\sigma_i}f(\sigma_i)} \\ &= m_{\sigma_i}. \end{aligned}$$

Security Analysis. We now prove the security of OT_n^k-I.

Theorem 1. *For scheme OT_n^k-I, R's choices are unconditionally secure.*

Proof. For every tuple $(b'_0, b'_1, \ldots, b'_{k-1})$ representing the choices $\sigma'_1, \sigma'_2, \ldots, \sigma'_k$, there is a tuple $(a'_0, a'_1, \ldots, a'_{k-1})$ that satisfies $A_i = g^{a'_i}h^{b'_i}$ for $i = 0, 1, \ldots, k-1$. Thus, the receiver R's choices are unconditionally secure. □

Theorem 2. *Scheme OT_n^k-I meets the sender's security requirement. That is, by the DDH assumption, if R is semi-honest, he gets no information about messages m_i, $i \notin \{\sigma_1, \sigma_2, \ldots, \sigma_k\}$.*

Proof. We show that for all $i \notin \{\sigma_1, \sigma_2, \ldots, \sigma_k\}$, c_i's look random if the DDH assumption holds. First, we define the random variable for the unchosen messages

$$C = (g, h, (g^{k_{i_1}}, m_{i_1}(g^{f(i_1)}h^{f'(i_1)})^{k_{i_1}}), ..., (g^{k_{i_{n-k}}}, m_{i_{n-k}}(g^{f(i_{n-k})}h^{f'(i_{n-k})})^{k_{i_{n-k}}})),$$

where $k_{i_1}, k_{i_2}, \ldots, k_{i_{n-k}} \in_R Z_q^*$. Since the polynomial $f(x)$ and $f'(x)$ are chosen by the receiver, and $f'(i_1), \ldots, f'(i_{n-k}) \neq 0$, we can simplify C as

$$C' = (g, h, (g^{k_{i_1}}, h^{k_{i_1}}), \ldots, (g^{k_{i_{n-k}}}, h^{k_{i_{n-k}}}))$$

Since the indistinguishability is preserved under multiple samples, we just need to show that if the following two distributions

- $\tilde{C} = (g, h, g^r, h^r)$, where $h \neq 1, r \in_R Z_q^*$
- $\tilde{X} = (g, h, x_1, x_2)$, where $h \neq 1, x_1, x_2 \in_R G_q$

are distinguishable by a polynomial-time distinguisher $\mathcal{D}$, we can construct another polynomial-time machine $\mathcal{D}'$, which takes $\mathcal{D}$ as a sub-routine, to solve the DDH problem:

- System parameters: (g, H_1, H_2, G_q);
- S has messages: $m_1, m_2, \ldots, m_n$;
- R's choices: $\sigma_1, \sigma_2, \ldots, \sigma_k$;

1. R computes $w_{\sigma_j} = H_1(\sigma_j)$ and $A_j = w_{\sigma_j} g^{a_j}$, where $a_j \in_R Z_q^*$ and $j = 1, 2, \ldots, k$.
2. $R \longrightarrow S$: $A_1, A_2, \ldots, A_k$.
3. S computes $y = g^x$, $D_j = (A_j)^x$, $w_i = H_1(i)$, and $c_i = m_i \oplus H_2(w_i^x)$, where $x \in_R Z_q^*, i = 1, 2, \ldots, n$, and $j = 1, 2, \ldots, k$.
4. $S \longrightarrow R$: y, $D_1, D_2, \ldots, D_k$, $c_1, c_2, \ldots, c_n$
5. R computes $K_j = D_j / y^{a_j}$ and gets $m_{\sigma_j} = c_{\sigma_j} \oplus H_2(K_j)$ for $j = 1, 2, \ldots, k$.

Fig. 2. OT_n^k-II: k-out-of-n OT against malicious receiver.

Machine $\mathcal{D}'$

Input: (g, u, v, w) (either from Y_1 or Y_2 in DDH)
Output: $\mathcal{D}(g, u, v, w)$

If $\mathcal{D}$ distinguishes $\tilde{C}$ and $\tilde{X}$ with non-negligible advantage ε (Should be $\epsilon(n, t)$, we omit the security parameter n and t here for simplicity, where t is the security parameter.), $\mathcal{D}'$ distinguishes Y_1, Y_2 in the DDH problem with at least non-negligible advantage $\varepsilon - 2/q$, where $dist(\tilde{C}, Y_1) = 1/q$ and $dist(\tilde{X}, Y_2) = 1/q$. □

Complexity. The scheme uses two rounds (steps 2 and 4), the first round sends $k + 1$ messages and the second round sends $2n$ messages. For computation, R computes $3k + 2$ and S computes $(k + 2)n$ modular exponentiations.

3.2 k-Out-of-n OT Against Malicious Receiver

A malicious player may not follow the protocol dutifully. For example, in scheme OT_n^k-I, a malicious R might send some special form of A_i's in step 2 such that he is able to get extra information, such as the linear combination of two messages (even though we don't know how to do such attack). So, we present another scheme OT_n^k-II that is provable secure against the malicious R. The scheme is depicted in Figure 2.

Let G_q be the subgroup of Z_p^* with prime order q, g be a generator of G_q, and $p = 2q + 1$ is also prime. Let $H_1 : \{0,1\}^* \to G_q, H_2 : G_q \to \{0,1\}^l$ be two collision-resistant hash functions. Let messages be of l-bit length. Assume that CT-CDH is hard under G_q.

Correctness. We can check that the chosen messages m_{σ_j}, $j = 1, 2, \ldots, k$, are computed as

$$\begin{aligned} c_{\sigma_j} \oplus H_2(K_j) &= m_{\sigma_j} \oplus H_2(w_{\sigma_j}^x) \oplus H_2(w_{\sigma_j}^x) \\ &= m_{\sigma_j}. \end{aligned}$$

Security Analysis. We need the random oracle model in this security analysis.

Theorem 3. *In OT_n^k-II, R's choice meets the receiver's privacy.*

Proof. For any $A_j = w_j g^{a_j}$ and w_l, $l \neq j$, there is an a'_l that satisfies $A_j = w_l g^{a'_l}$. For S, A_j can be a masked value of any index. Thus, the receiver's choices are unconditionally secure. □

Theorem 4. *Even if R is malicious, the scheme OT_n^k-II meets the requirement for the sender's security assuming hardness of the CT-CDH problem the random oracle model.*

Proof. Since we treat H_2 as a random oracle, the malicious R has to know $K_i = w_i^x$ in order to query the hash oracle to get $H_2(w_i^x)$. For each possible malicious R, we construct a simulator R^* in the Ideal model such that the outputs of R and R^* are indistinguishable.

R^* works as follows:

1. R^* simulates R to obtain $A_1^*, A_2^*, \ldots, A_k^*$. When R queries H_1 on index i, we return a random w_i^* (consistent with the previous queries.)
2. R^* simulates S (externally without knowing m_i's) on inputs A_1^*, $A_2^*, \ldots,$ A_k^* to obtain x^*, y^*, $D_1^*, D_2^*, \ldots, D_k^*$.
3. R^* randomly chooses $c_1^*, c_2^*, \ldots, c_n^*$.
4. R^* simulates R on input $(y^*, D_1^*, D_2^*, \ldots, D_k^*, c_1^*, c_2^*, \ldots, c_n^*)$ and monitors the queries closely. If R queries H_2 on some $v_j = (w_j^*)^{x^*}$, R^* sends j to the TTP T to obtain m_j and returns $c_j^* \oplus m_j$ as the hash value $H_2((w_j^*)^{x^*})$, otherwise, returns a random value (consistent with previous queries).
5. Output $(A_1^*, A_2^*, \ldots, A_k^*, y^*, D_1^*, D_2^*, \ldots, D_k^*, c_1^*, c_2^* \ldots, c_n^*)$.

If R obtains $k+1$ decryption keys, R^* does not know which k indices are really chosen by R. The simulation would fail. Therefore we show that R can obtain at most k decryption keys by assuming the hardness of chosen-target CDH problem: In the above simulation, if R queries H_1, we return a random value output by the target oracle. When R^* simulates S on input $A_1^*, A_2^*, \ldots, A_k^*$, we forward these queries to the helper oracle, and return the corresponding outputs. Finally, if R queries H_2 on legal v_{j_i} for all $1 \leq i \leq k+1$, we can output $k+1$ pairs (v_{j_i}, j_i), which contradicts to the CT-CDH assumption. Thus, R obtains at most k decryption keys.

Let $\sigma_1, \sigma_2, \ldots, \sigma_k$ be the k choices of R. For the queried legal v_{σ_j}'s, c_{σ_j} is consistent with the returned hash values, for $j = 1, 2, \ldots, k$. Since no other $(w_l^*)^{x^*}$, $l \neq \sigma_1, \sigma_2, \ldots, \sigma_k$, can be queried to the H_2 hash oracle, c_l has the right distribution (due to the random oracle model). Thus, the output distribution is indistinguishable from that of R. □

Complexity. OT_n^k-II has two rounds. The first round sends k messages and the second round sends $n+k+1$ messages. For computation, R computes $2k$, and S computes $n+k+1$ modular exponentiations.

- System parameters: (g, H_1, H_2, G_q);
- S has messages: $m_1, m_2, \ldots, m_n$;
- R's choices: $\sigma_1, \sigma_2, \ldots, \sigma_k$;

Commitment Phase

1. S computes $c_i = m_i \oplus H_2(w_i^x)$ for $i = 1, 2, \ldots, n$, and $y = g^x$ where $w_i = H_1(i)$, and $x \in_R Z_q^*$.
2. $S \longrightarrow R : y, c_1, c_2, \ldots, c_n$.

Transfer Phase

For each σ_j, $j = 1, 2, \ldots, k$, R and S execute the following steps:

1. R chooses a random $a_j \in Z_q^*$ and computes $w_{\sigma_j} = H_1(\sigma_j), A_j = w_{\sigma_j} g^{a_j}$.
2. $R \longrightarrow S : A_j$.
3. $S \longrightarrow R : D_j = (A_j)^x$.
4. R computes $K_j = D_j / y^{a_j}$ and gets $m_{\sigma_j} = c_{\sigma_j} \oplus H_2(K_j)$.

Fig. 3. Adpt-OT$_n^k$: Adaptive OT$_n^k$.

4 k-Out-of-n OT with Adaptive Queries

The queries of R in our schemes can be adaptive. In our schemes, the commitments c_i's of the messages m_i's of S to R are independent of the key masking. Therefore, our scheme is adaptive in nature. Our Adpt-OT$_n^k$ scheme, which rephrases the OT$_n^k$-II scheme, is depicted in Figure 3.

The protocol consists of two phases: the commitment phase and the transfer phase. The sender S first commits the messages in the commitment phase. In the transfer phase, for each query, R sends the query A_j to S and obtains the corresponding key to decrypt the commitment c_j.

Correctness of the scheme follows that of OT$_n^k$-II.

Security Analysis. The security proofs are almost the same as those for OT$_n^k$-II. We omit them here.

Complexity. In the commitment phase, S needs $n + 1$ modular exponentiations for computing the commitments c_i's and y. In the transfer phase, R needs 2 modular exponentiations for computing the query and the chosen message. S needs one modular exponentiation for answering each R's query. The commitment phase is one-round and the transfer phase is two-round for each adaptive query.

5 Conclusion

We have presented two very efficient OT$_n^k$ schemes against semi-honest receivers in the standard model and malicious receivers in the random oracle model. Our schemes possess other interesting features, such as, it can be non-interactive and needs no prior setup or trapdoor. We also proposed an efficient Adpt-OT$_n^k$ for

adaptive queries. The essential feature allowing this is the reversal of the orders of key commitment and message commitment. In most previous schemes (including OT_n^k-I), the key commitments (for encrypting the chosen messages) are sent to S first. The message commitments are dependent on the key commitments. Nevertheless, in our scheme OT_n^k-II the message commitments are independent of the key commitment. Thus, the message commitments can be sent to R first.

References

[BBCS91] Charles H. Bennett, Gilles Brassard, Claude Crépeau, and Marie-Hélène Skubiszewska. Practical quantum oblivious transfer. In *Proceedings of Advances in Cryptology - CRYPTO '91*, volume 576 of *LNCS*, pages 351–366. Springer-Verlag, 1991.

[BCR86] Gilles Brassard, Claude Crépeau, and Jean-Marc Robert. All-or-nothing disclosure of secrets. In *Proceedings of Advances in Cryptology - CRYPTO '86*, volume 263 of *LNCS*, pages 234–238. Springer-Verlag, 1986.

[BCR87] Gilles Brassard, Claude Crépeau, and Jean-Marc Robert. Information theoretic reductions among disclosure problems. In *Proceedings of 28th Annual Symposium on Foundations of Computer Science (FOCS '87)*, pages 427–437. IEEE, 1987.

[BCS96] Gilles Brassard, Claude Crépeau, and Miklós Sántha. Oblivious transfers and intersecting codes. *IEEE Transactions on Information Theory*, 42(6):1769–1780, 1996.

[BDSS02] Carlo Blundo, Paolo D'Arco, Alfredo De Santis, and Douglas R. Stinson. New results on unconditionally secure distributed oblivious transfer. In *Proceedings of Selected Areas in Cryptography - SAC '02*, volume 2595 of *LNCS*, pages 291–309. Springer-Verlag, 2002.

[BM89] Mihir Bellare and Silvio Micali. Non-interactive oblivious transfer and applications. In *Proceedings of Advances in Cryptology - CRYPTO '89*, volume 435 of *LNCS*, pages 547–557. Springer-Verlag, 1989.

[BNPS01] Mihir Bellare, Chanathip Namprempre, David Pointcheval, and Michael Semanko. Power of rsa inversion oracles and the security of Chaum's RSA-based blind signature scheme. In *Proceedings of Financial Cryptography (FC '01)*, pages 319–338. Springer-Verlag, 2001.

[Bol03] Alexandra Boldyreva. Threshold signatures, multisignatures and blind signatures based on the gap-diffie-hellman-group signature scheme. In *Proceedings of the Public-Key Cryptography (PKC '03)*, pages 31–46. Springer-Verlag, 2003.

[CCM98] Christian Cachin, Claude Crepeau, and Julien Marcil. Oblivious transfer with a memory-bounded receiver. In *Proceedings of 39th Annual Symposium on Foundations of Computer Science (FOCS '98)*, pages 493–502. IEEE, 1998.

[CZ03] Zhide Chen and Hong Zhu. Quantum m-out-of-n oblivious transfer. Technical report, arXiv:cs.CR/0311039, 2003.

[Din01] Yan Zong Ding. Oblivious transfer in the bounded storage model. In *Proceedings of Advances in Cryptology - CRYPTO '01*, volume 2139 of *LNCS*, pages 155–170. Springer-Verlag, 2001.

[EGL85] Shimon Even, Oded Goldreich, and Abraham Lempel. A randomized protocol for signing contracts. *Communications of the ACM*, 28(6):637–647, 1985.

[GV87] Oded Goldreich and Ronen Vainish. How to solve any protocol problem - an efficiency improvement. In *Proceedings of Advances in Cryptology - CRYPTO '87*, volume 293 of *LNCS*, pages 73–86. Springer-Verlag, 1987.

[IKNP03] Yuval Ishai, Joe Kilian, Kobbi Nissim, and Erez Petrank. Extending oblivious transfers efficiently. In *Proceedings of Advances in Cryptology - CRYPTO '03*, volume 2729 of *LNCS*, pages 145–161. Springer-Verlag, 2003.

[Kil88] Joe Kilian. Founding cryptography on oblivious transfer. In *Proceedings of the 20th Annual ACM Symposium on the Theory of Computing (STOC '88)*, pages 20–31. ACM, 1988.

[Lip] Helger Lipmaa. Oblivious transfer. http://www.tcs.hut.fi/~helger/crypto/link/protocols/oblivious.html.

[Lip04] Helger Lipmaa. An oblivious transfer protocol with log-squared communication. Technical report, Cryptology ePrint Archive: Report 2004/063, 2004.

[MZV02] Yi Mu, Junqi Zhang, and Vijay Varadharajan. m out of n oblivious transfer. In *Proceedings of the 7th Australasian Conference on Information Security and Privacy (ACISP '02)*, volume 2384 of *LNCS*, pages 395–405. Springer-Verlag, 2002.

[NP99a] Moni Naor and Benny Pinkas. Oblivious transfer and polynomial evaluation. In *Proceedings of the 31th Annual ACM Symposium on the Theory of Computing (STOC '99)*, pages 245–254. ACM, 1999.

[NP99b] Moni Naor and Benny Pinkas. Oblivious transfer with adaptive queries. In *Proceedings of Advances in Cryptology - CRYPTO '99*, volume 1666 of *LNCS*, pages 573–590. Springer-Verlag, 1999.

[NP00] Moni Naor and Benny Pinkas. Distributed oblivious transfer. In *Proceedings of Advances in Cryptology - ASIACRYPT '00*, volume 1976 of *LNCS*, pages 200–219. Springer-Verlag, 2000.

[NP01] Moni Naor and Benny Pinkas. Efficient oblivious transfer protocols. In *Proceedings of the 12th Annual Symposium on Discrete Algorithms (SODA '01)*, pages 448–457. ACM/SIAM, 2001.

[NR94] Valtteri Niemi and Ari Renvall. Cryptographic protocols and voting. In *Results and Trends in Theoretical Computer Science*, volume 812 of *LNCS*, pages 307–317. Springer-Verlag, 1994.

[OK02] Wakaha Ogata and Kaoru Kurosawa. Oblivious keyword search. *Journal of Complexity*, 20(2-3):356–371, 2004.

[Rab81] Michael O. Rabin. How to exchange secrets by oblivious transfer. Technical Report TR-81, Aiken Computation Laboratory, Harvard University, 1981.

[SS90] Arto Salomaa and Lila Santean. Secret selling of secrets with several buyers. *Bulletin of the European Association for Theoretical Computer Science (EATCS)*, 42:178–186, 1990.

[Ste98] Julien P. Stern. A new and efficient all or nothing disclosure of secrets protocol. In *Proceedings of Advances in Cryptology - ASIACRYPT '98*, volume 1514 of *LNCS*, pages 357–371. Springer-Verlag, 1998.

[Tze02] Wen-Guey Tzeng. Efficient 1-out-n oblivious transfer schemes. In *Proceedings of the Public-Key Cryptography (PKC '02)*, pages 159–171. Springer-Verlag, 2002.

[YBD03] Gang Yao, Feng Bao, and Robert Deng. Security analysis of three oblivious transfer protocols. Workshop on Coding, Cryptography and Combinatorics, Huangshan City, China, 2003.

A Secure System for Data Access Based on Anonymous Authentication and Time-Dependent Hierarchical Keys *

Wen-Guey Tzeng
Department of Computer Science
National Chiao Tung University
Hsinchu, Taiwan 30050
wgtzeng@cs.nctu.edu.tw

ABSTRACT

We consider the security problem for retrieving data from a web site (or a large database system) via Internet. Consider the situation that a user visits a web site to get information. He wants to retain anonymity of his identity, while the web site would like to authenticate the user's identity. We proposed an anonymous authentication scheme to provide a solution for these two seemingly conflicting requirements. Our anonymous authentication scheme is based on cryptographic techniques. Together with a novel time-dependent hierarchical key assignment scheme, we proposed a data access system that has the following distinct features simultaneously: (1) anonymous authentication, (2) cryptographic access control, (3) saving on-line encryption time, and (4) a flexible subscription system.

Keywords

anonymous authentication, entity security, hierarchical keys, trapdoor time-sharable sequence, world wide web security

1. INTRODUCTION

The world wide web (WWW) (or Internet-accessible database system) is a revolution for obtaining information. We can get information at an unprecedented speed. Nevertheless, we should be aware of its security problems as well. World wide web security issues are very broad. In this paper we are interested in the security problems with balanced views from web sites and users. That is, a web site wants only authorized users to access information and a user wants to protect his individual privacy.

It is a common practice that a web site (server) W requires a user U register his personal information, such as, name, email, affiliation, etc. When U wants to retrieve data from W, he sends his identity (user name) to W. W then authenticates U's identity and authorizes his access right. Therefore, W can gather U's visiting pattern (frequency, interested data, etc.) and uses it for other purposes. We have seen such intrusion to individual privacy occur in the credit card and telecommunication industries. In the era of Internet, users would like to enjoy convenience of the world wide web and to have protection against identity tracking, behavior analysis, etc. On the other hand, the web site, in particular, with subscription fee, would like to protect its rights by allowing only authorized users to access its information. These two requirements seem conflicting in the sense that the user would like his identity to be anonymous and the web site would like to know the user's identity for authentication and authorization.

To resolve this conflict, we propose an anonymous authentication scheme for data access, which provides balanced security mechanisms for both users and web sites, that is, a web site can authenticate a user's identity without knowing his identity. Our scheme is based on cryptographic techniques, such as witness-indistinguishable proof systems.

We also propose a new time-dependent hierarchical key assignment scheme in which a set of time-dependent class keys can be computed from a constant-sized key trapdoor. We combine the anonymous authentication and time-dependent hierarchical key assignment schemes to form a secure system for retrieving data from web sites. The system not only provides user anonymity and authentication, but also ensures communication security without doing on-line encryption. The distinct features of our system are summarized as follows.

1. Anonymous authentication: anonymity of users and authentication right of web sites are guaranteed simultaneously.

2. Cryptographic access control: since data are encrypted, web sites and their mirror sites can use a weaker access control system without jeopardizing security of information in the databases.

3. Saving the computation cost of on-line encryption: since stored data are encrypted, it is not necessary to do on-line encryption for secure communication.

4. A flexible subscription system: with the time-dependent hierarchical key assignment scheme, web sites can es-

*Research supported in part by National Science Council grant NSC 94-2213-E-009-116, Taiwan.

tablish a flexible subscription system, in which users pay different premiums for different access rights.

1.1 Related work

Hierarchical key assignment schemes for access control are first studied by Akl and Taylor [1]. Many researchers followed to propose improvements [3, 9, 16]. Tzeng [22] proposed a time-dependent hierarchical key assignment scheme, which is an extension of them with an additional dimension on time periods. The scheme has some interesting applications, for example, secure broadcasting and key backup. The scheme cannot withstand the collusion attack.[1] In this paper we propose a new time-dependent hierarchical key assignment scheme that is secure against the collusion attack of multiple adversaries.

Entity authentication has been studied extensively (see [8, 18, 20]). ISO/IEC has a series of standards for entity authentication [10, 11, 12, 13, 14]. But most work assumes that the user's identity is not anonymous.

Kilian and Petrank [15] proposed a user identification scheme in which the system can authenticate a user's identity without knowing his identity. Their scheme has a third party to generate a certificate for a user so that the user can use the certificate to authenticate himself without revealing his identity. In their scheme, it is hard to revoke a user's certificate (membership). Thus, we need to employ another delicate and time-consuming cryptographic scheme to verify the status of a certificate. The whole system becomes very complicated.

Chaum [4] introduced the pseudonym system, in which a user has a different pseudonym for each organization. The goal is to prevent organizations from inferring the user's information by combining their data of the user's visiting patterns. Chaum and Evertse [5] proposed a RSA-based pseudonym system that is secure against cooperation of all organizations. Lysanysanskaya, etc., proposed a pseudonym system that discourages a user from sharing his master key with other users. The master key is used to derive certificates for organizations. We can see that the goal of pseudonym systems is different from ours.

2. ENTITY SECURITY AND ANONYMOUS AUTHENTICATION

In the entity security model [18], user identification, authentication and authorization are three steps by which a system provides services to a user. For user identification, the user sends his identity to the system and the system checks whether the identity is legal for the system. For user authentication, the system authenticates the user's identity by verifying whether the user possesses some secret information (password, secret key, etc.) about the identity whom the user claims to be. After authenticating the user's identity, the system checks whether the user's request is within his access right. This is user authorization. After checking authorization, the system processes the user's legal requests.

[1]In the paper [22], the scheme is only proven to be secure against the attacks from a single adversary. It is neither claimed nor proven to be secure against the collusion attack of multiple adversaries.

We can see that in the entity security model user identification is the main step that the system identifies who the user is. The system uses the user's identity to retrieve the user's authentication and authorization data. Then, the system uses the data to authenticate the user and authorizes the user's access right to resources. In some applications, such as data retrieval from web sites, there may be only one authorization level, that is, all users are allowed to use all resources (files) of the system. In some other applications, there are multiple authorization levels. For example, in a subscribed database system, each user may pay some premium for accessing data of a specific category, such as sports, business, etc. The system assigns the user to the authorization level of the category which he pays for.

A straightforward solution for anonymous authentication is to have a third party R to help. A user U registers his identity α to R and gets a pseudo identity (pseudonym) $\overline{\alpha}$ [17]. R gives U's pseudo information to the system W and hides U's real identity from W. Every time U visits the system, he uses his pseudo identity $\overline{\alpha}$ for authentication. By this information, W authorizes U to access data in its database. For the system to function properly, R need be trusted by both the system and users. To implement an appropriate R is not an easy task. Even though R is used, the visiting pattern of $\overline{\alpha}$ is exactly that of α. It can still be used in some way. For example, it is possible to infer the real identity α from the visiting pattern.

We solve the anonymous authentication problem by the cryptographic techniques of public-key certificates and witness-indistinguishable proof systems, which are shown in Section 5.

3. PRELIMINARIES

Efficient computation. By efficient computation we mean that the computing time is a polynomial function of the input size. Note that the size of a number x is its bit length $\lfloor \log_2 x \rfloor + 1$. Modular multiplication and exponentiation are efficiently computable.

Modular exponentiation over a composite number. Let $p' = 2p + 1$ and $q' = 2q + 1$ be two large primes, typically 512-bit long, the composite number $N = p'q'$ and the Euler's totient function $\phi(N) = (p' - 1)(q' - 1) = 4pq$. We consider the order-$pq$ subgroup $G_{pq} = \{a^4 \bmod n | a \in Z_n^*\}$ of Z_n^*. We say that g is a generator for G_{pq} if $\{g^i | 0 \leq i \leq pq - 1\} = G_{pq}$. The modular exponentiation over N is to compute $x^a \bmod N$ for given x, a and N. We can compute $x^a \bmod N$ by the square-multiply method, which takes $1.5\lfloor \log_2 a \rfloor$ modular multiplications in average. A modern computer can easily afford this computation.

Common modulus property. The common modulus property is that given a, b, $x^a \bmod N$ and $x^b \bmod N$, we can compute $x^{\gcd(a,b)} \bmod N$ as follows. We first use the extended Euclidean algorithm to find integers a' and b' such that $a'a + b'b = \gcd(a, b)$ and then compute

$$\begin{aligned} &(x^a \bmod N)^{a'} (x^b \bmod N)^{b'} \bmod N \\ &= x^{a'a+b'b} \bmod N \\ &= x^{\gcd(a,b)} \bmod N. \end{aligned}$$

Computing the eth roots modulo a composite. The problem of computing the eth roots modulo a composite is, for given e, y and N, to compute x such that $x^e \equiv y \pmod{N}$. Solving this problem is equivalent to breaking the famous RSA public-key cryptosystem [19]. We assume that it is computationally infeasible to compute the eth roots modulo N for any given e, $2 \leq e \leq \phi(N) - 1$.

Computing the discrete logarithm modulo a prime. The problem of computing discrete logarithm modulo a prime p is, for given a generator g and an integer y, to compute integer x such that $y \equiv g^x \pmod{p}$. Solving this problem efficiently is considered very hard.

Partially ordered hierarchy. A partially ordered hierarchy is a directed graph $G = (V, E)$ without cycles. We call the nodes without out-edges as the *base* nodes and the others as the *super* nodes. A path from node v_i to node v_j is a sequence $(v_{r_0}, v_{r_1}, \ldots, v_{r_l})$ of nodes with $v_{r_0} = v_i$, $v_{r_l} = v_j$, and $(v_{r_k}, v_{r_{k+1}}) \in E$ for all k, $1 \leq k \leq l-1$. A node v_j is an immediate descendant of another node v_i if $(i, j) \in E$. See Figure 1 for a 8-node partially ordered hierarchy.

Hierarchical key assignment. Let C_i, $1 \leq i \leq m$, be classes (nodes) that form a partially ordered hierarchy. Let $C_j < C_i$ denote that C_j is lower than C_i, that is, there is a path from C_i to C_j. Let $C_j \leq C_i$ denote that $C_j < C_i$ or $C_j = C_i$. A hierarchical key assignment scheme is to assign each class C_i a (cryptographic) class key K_i so that given K_i and public parameters, we can compute the class key K_j if and only if $C_j \leq C_i$, $1 \leq j \leq m$. The security requirement is that for any set S of classes, we cannot use their class keys to compute the key of the class that is not lower than any class in S. The challenge is to design such a scheme such that the size of each class key is independent of the number of classes in the hierarchy.

Time-dependent hierarchical key assignment. It is like hierarchical key assignment except that a class key also depends on the factor of time t. A class key of C_i at time t is denoted as $K_{i,t}$. A time-dependent hierarchical key assignment is to assign key trapdoors $K_{[i,t_1,t_2]}$ such that we can use $K_{[i,t_1,t_2]}$ to compute $K_{j,t}$ if and only if $C_j \leq C_i$ and $t \in \{t_1, \ldots, t_2\}$.

Witness-indistinguishable proof system. Assume that there are n public keys pk_i, $1 \leq i \leq n$, which each has a corresponding private key sk_i. A witness-indistinguishable proof system for the public keys is that the prover convinces the verifier that it knows one of the private keys, but does not reveal which one.

Notation. Let $x \in_R X$ denote that x is chosen from the set X uniformly and independently.

4. TIME DEPENDENT HIERARCHICAL KEY ASSIGNMENT SCHEME

Our time-dependent hierarchical key assignment scheme is based on hardness of computing the eth roots modulo a composite and the common-modulus property of modular exponentiations. We first present the basic concepts and then adapt it to the structure of the time-dependent hierarchical key assignment scheme.

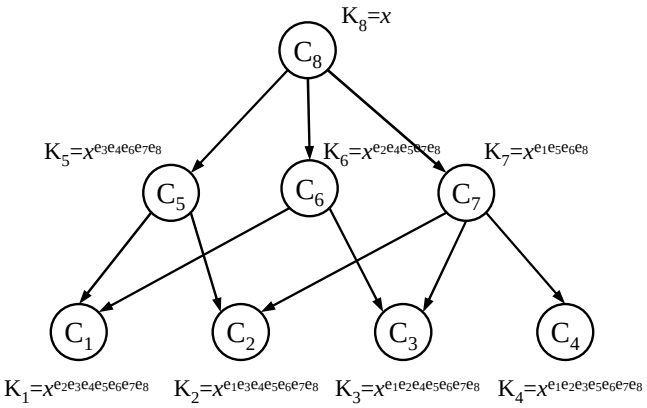

Figure 1: A hierarchical key assignment.

4.1 Hierarchical key assignment scheme

For a partially ordered hierarchy of m classes C_i, $1 \leq i \leq m$, we associate each class C_i a cover set of classes

$$E_i = \{\, j \mid C_j \leq C_i, 1 \leq j \leq m\}.$$

The cover set of the base class C_i (with no out-edges) is $\{i\}$ and the cover set of a super class C_i can be computed as $\{i\} \cup E_{i_1} \cup E_{i_2} \cup \cdots \cup E_{i_t}$ recursively, where $C_{i_1}, C_{i_2}, \ldots, C_{i_t}$ are immediate descendants of C_i. We can see that if $C_j \leq C_i$, then $E_j \subseteq E_i$.

The hierarchical key assignment scheme of Akl and Taylor [1]is as follows.

1. **System setup**.
   (a) Let N and G_{pq} be the ones defined in Section 3.
   (b) Let $e_1, e_2, \ldots, e_m$ be the first m primes and $\theta = e_1 e_2 \cdots e_m$.[2]
   (c) Choose a generator $x \in_R G_{pq}$.
2. **Public parameters**. $(N, e_1, e_2, \ldots, e_m)$. These parameters are publicly known. We let the method of computing e_i's be known so that the entire e_i's need not be stored.
3. **Class key**. The class key of C_i with cover set $E_i = \{i_1, i_2, \ldots, i_t\}$ is
$$K_i = x^{\theta/(e_{i_1} e_{i_2} \cdots e_{i_t})} \bmod N.$$
4. **Key derivation**. Given the class key K_i and the public parameters, we can compute the class key K_j of C_j for any $C_j \leq C_i$ as follows. Let $E_i - E_j = \{d_1, d_2, \ldots, d_s\}$ and, then,
$$K_j = K_i^{e_{d_1} e_{d_2} \cdots e_{d_s}} \bmod N.$$

Let us see an example in Figure 1 for this construction. The partially ordered hierarchy has 8 classes. Classes C_1,

[2] e_i's are not necessarily prime. We only require them to be pariwisely relatively prime.

$C_2, \ldots, C_4$ are base classes and the others are super classes. The cover sets are $E_1 = \{1\}$, E_2=$\{2\}$, $E_3 = \{3\}$, $E_4 = \{4\}$, E_5=$\{1, 2, 5\}$, $E_6 = \{1, 3, 6\}$, $E_7 = \{2, 3, 4, 7\}$, and $E_8 = \{1, 2, 3, 4, 5, 6, 7, 8\}$. The class keys are

$K_1 = x^{e_2e_3e_4e_5e_6e_7e_8} \bmod N$, $K_2 = x^{e_1e_3e_4e_5e_6e_7e_8} \bmod N$,
$K_3 = x^{e_1e_2e_4e_5e_6e_7e_8} \bmod N$, $K_4 = x^{e_1e_2e_3e_5e_6e_7e_8} \bmod N$,
$K_5 = x^{e_3e_4e_6e_7e_8} \bmod N$, $K_6 = x^{e_2e_4e_5e_7e_8} \bmod N$,
$K_7 = x^{e_1e_5e_6e_8} \bmod N$, $K_8 = x$.

We see how to use K_7 to derive K_3. Since $E_7 - E_3 = \{2, 4, 7\}$, we have

$$K_3 = K_7^{e_2e_4e_7} \bmod N = x^{e_1e_2e_4e_5e_6e_7e_8} \bmod N.$$

Security. Given any set S of class keys, one cannot compute the unauthorized class keys unless one can compute the eth root modular a composite.

THEOREM 4.1. *[1] Given a set of class keys K_{i_1}, $K_{i_2}, \ldots, K_{i_t}$, one cannot compute the class key K_j of class C_j that is not a descendent of any C_{i_l}, $1 \le l \le t$, unless one can solve the problem of computing the eth root modulo a composite for some $e \ge 2$.*

4.2 New time-dependent hierarchical key assignment scheme

Let C_i, $1 \le i \le m$, be the classes that are partially ordered. Let time be divided into time periods $1, 2, \ldots, z$, where z is the maximum time period. This maximum time period should not be considered as limitation of the system. For example, if each time period represents a week, $z = 5200$ denotes roughly 100 years. For a time-dependent hierarchical key assignment scheme, class C_j at time period t has time-dependent class key $K_{j,t}$. We require that, given the key trapdoor $K_{[i,t_1,t_2]}$, one can compute $K_{j,t}$ if and only if $C_j \le C_i$ and $t_1 \le t \le t_2$. Furthermore, the size of $K_{[i,t_1,t_2]}$ should be independent of the number of classes in the hierarchy and the length of time periods.

Our construction revolutionizes the idea of Akl and Taylor's construction. We put all original classes of the hierarchy as base classes C'_i and add super classes $C'_{[i]}$, $1 \le i \le m$. Each $C'_{[i]}$ covers the base class set

$$\{C'_j | C_j \le C_i, 1 \le j \le m\}.$$

Then, we assign class keys K_j to C'_j, $1 \le j \le m$, and trapdoors $K_{[i]}$ to $C'_{[i]}$ such that we can use trapdoor $K_{[i]}$ to compute all class keys K_j if and only if $C_j \le C_i$. Now, the security requirement of our scheme is different from that of Akl and Taylor's. We only require that given any set of key trapdoors $K_{[i_1,r_1,s_1]}$, $K_{[i_2,r_2,s_2]}, \ldots, K_{[i_l,r_l,s_l]}$, one cannot compute the class key $K_{j,t}$ with $C_j \not\le C_{i_k}$, $t < r_k$ or $t > s_k$ for all k, $1 \le k \le l$.

Based on the above idea, our time-dependent hierarchical key assignment scheme has mz base classes $C'_{j,t}, 1 \le j \le m, 1 \le t \le z$, and $mz(z+1)/2$ super classes $C'_{[i,t_1,t_2]}$, $1 \le i \le z$, $1 \le t_1 \le t_2 \le z$. The base class $C'_{j,t}$ denote the class C_j at time period t. Though we have total $mz(z+1)/2 + mz$ classes, we use only mz exponents (e's) for the scheme. The key trapdoor $K_{[i,t_1,t_2]}$ corresponds to the time-dependent class key of the super class $C'_{[i,t_1,t_2]}$ that covers the base classes $C'_{j,t}$, for any $C_j \le C_i$ and $t_1 \le t \le t_2$. We associate each super class $C'_{[i,t_1,t_2]}$ a cover set of base classes

$$E'_{[i,t_1,t_2]} = \{ (j,t) \,|C'_{j,t}, C_j \le C_i, t_1 \le t \le t_2\}.$$

The formal description is as follows.

1. **System setup**.
   (a) Let N and G_{pq} be the ones defined in Section 3.
   (b) Let $e_{j,t}$, $1 \le j \le m$, $1 \le t \le z$, be the first mz primes and
   $$\theta = \prod_{1 \le j \le m, 1 \le t \le z} e_{j,t}.$$
   (c) Choose a generator $x \in_R G_{pq}$.
2. **Public parameters**. $(N, e_{1,1}, \ldots, e_{m,z})$, Again, $e_{i,t}$'s need not be stored.
3. **Time-dependent class key**. The time-dependent class key $K_{j,t}$ of the base class $C'_{j,t}$ is
   $$K_{j,t} = x^{\theta/e_{j,t}} \bmod N.$$
4. **Time-dependent key trapdoor**. The key trapdoor $K_{[i,t_1,t_2]}$ of the super class $C'_{[i,t_1,t_2]}$, $1 \le i \le m$, $1 \le t_1 \le t_2 \le z$, is
   $$K_{[i,t_1,t_2]} = x^{\theta/(e_{j_1,r_1}e_{j_2,r_2}\cdots e_{j_l,r_l})} \bmod N,$$
   where $E'_{[i,t_1,t_2]} = \{(j_1,r_1),(j_2,r_2),\ldots,(j_l,r_l)\}$.
5. **Time-dependent key derivation**. Given $K_{[i,t_1,t_2]}$ of the super class $C'_{[i,t_1,t_2]}$ and the public parameters, we can compute $K_{j,t}$ of the base class $C'_{j,t}$ if $C_j \le C_i$ and $t_1 \le t \le t_2$ as
   $$K_{j,t} = (K_{[i,t_1,t_2]})^{e_{j_1,d_1}e_{j_2,d_2}\cdots e_{j_s,d_s}} \bmod N,$$
   where $E'_{[i,t_1,t_2]} - E'_{[j,t,t]} = \{(j_1,d_1),(j_2,d_2),\ldots,(j_s,d_s)\}$.

Security analysis. The security of this scheme is equivalent to computing the eth roots modulo a composite.

THEOREM 4.2. *Given any set of key trapdoors $K_{[i_1,r_1,s_1]}$, $K_{[i_2,r_2,s_2]}, \ldots, K_{[i_l,r_l,s_l]}$, one cannot compute the class key $K_{j,t}$ with $C_j \not\le C_{i_k}$, $t < r_k$ or $t > s_k$ for all k, $1 \le k \le l$, unless one can solve the problem of computing the eth roots modulo a composite for some $e \ge 2$.*

PROOF. Let

$$E' = E'_{[i_1,r_1,s_1]} \cup E'_{[i_2,r_2,s_2]} \cup \cdots \cup E'_{[i_l,r_l,s_l]}.$$

Since $C_{j,t}$ cannot be computed from any $C_{[i_k,r_k,s_k]}$, $1 \le k \le l$, we have $(j,t) \notin E'$. The key trapdoor $K_{[i_k,r_k,s_k]}$ has the form $x^{v_k e_{j,t}} \bmod N$ for some v_k. By the common modulus property, one can compute $x^{e_{j,t} \cdot \gcd(v_1,\ldots,v_l)} \bmod N$. However, the class key $K_{j,t}$ has the form $x^b \bmod N$ for some b with $\gcd(b, e_{j,t}) = 1$. To compute $K_{j,t}$ from the given key trapdoors, one has to remove the power of $e_{j,t}$

from $K_{[i_1,r_1,s_1]}$, $K_{[i_1,r_2,s_2]}, \ldots,$ $K_{[i_l,r_l,s_l]}$. This is equivalent to computing the eth roots modulus a composite as follows.

Let (y, e, N) be given. Set $e_{j,t} = e$ and $x = y^{1/e_{j,t}} \bmod N$. It does not matter that we don't know x. We selects $e_{i,t'}$, $1 \le i \le m$, $1 \le t' \ne t \le z$ such that they are relatively prime. Then,

$$K_{j,t} = x^{\Pi_{1\le i\le m, 1\le t'\ne t\le z}\ e_{i,t'}} \bmod N$$

and

$$K_{[i_k,r_k,s_k]} = y^{v_k} \bmod N, 1 \le k \le l.$$

Assume that one can compute $K_{j,t}$ from $K_{[i_k,r_k,s_k]}$, $1 \le k \le l$. We can compute x by the common modulus property of $y = x^{e_{j,t}} \bmod N$ and $K_{j,t}$ since

$$\gcd(e_{j,t}, \Pi_{1\le i\le m, 1\le t'\ne t\le z}\ e_{i,t'}) = 1.$$

This is a contradiction. Thus, our time-dependent hierarchical key assignment scheme is secure against any collusion attack from multiple adversaries. □

5. ANONYMOUS AUTHENTICATION SCHEME

Our anonymous authentication scheme is based on certificates and witness-indistinguishable proof systems. The idea is to let each user possess a certificate (secret). When the user requests services, he engages a witness-indistinguishable proof system with the system server W using his certificate. If the user does not own a valid certificate, he cannot pass the test from W. Since the certificate that the user uses for authentication cannot be distinguished by W, the user's identity is anonymous. Therefore, W can authenticate the user without knowing its identity. We first describe system setup, system parameters and the user registration procedure.

1. **System setup**.
   (a) W chooses a large prime $p = 2p' + 1$ and a generator g for Z_p^*, where p' is also prime. Typically, p is 1024-bit long.
   (b) W maintains an *authentication list* L, which consists of the public authentication keys of its members. L is empty initially.
2. **System parameters**. (g, p, L).
3. **User registration**. When U registers to W for the first time, W gives an identity α to U. U selects a value $s \in_R Z_{p-1}$ with $\gcd(s, p-1) = 1$ and gives $(\alpha, g^s \bmod p)$ to W. W adds U's *authentication key*
   $$(\alpha, v) = (\alpha, g^s \bmod p)$$
   to L. U keeps its *certificate* (α, v, s).
4. **User revocation**. W simply removes U's authentication key (α, v) from L.

Assume that $L = \{(\alpha_1, v_1), (\alpha_2, v_2), \ldots, (\alpha_m, v_m)\}$ and U's certificate is (α_j, v_j, s_j) for some $1 \le j \le m$. Our anonymous authentication scheme is as follows.

1. U selects w_1, $w_2, \ldots,$ w_m, c_1, $c_2, \ldots,$ c_{j-1}, $c_{j+1}, \ldots,$ $c_m \in_R Z_{p-1}$ and computes $a_i = g^{w_i} v_i^{c_i} \bmod p$ for $1 \le i \ne j \le m$ and $a_j = g^{w_j} \bmod p$. Then, U sends $(a_1, a_2, \ldots, a_m)$ to W.
2. W selects $c \in_R Z_{p-1}$ and sends it to U.
3. U computes
   $$c_j = c - (c_1 + c_2 + \cdots + c_{j-1} + c_{j+1} + \cdots + c_m) \bmod p - 1$$
   and sets $r_i = w_i$ for $1 \le i \ne j \le m$ and $r_j = w_j - c_j s_j \bmod p - 1$. Then, U sends (c_i, r_i), $1 \le i \le m$, to W.
4. W verifies whether $a_i = g^{r_i} v_i^{c_i} \bmod p$ for all $1 \le i \le m$. If it is so, W accepts U as a legal member; otherwise, W rejects U as a legal member.

In the scheme, U commits his certificate (α_j, v_j, s_j) on $a_j = g^{w_j} \bmod p$, which he can answer any challenge c_j from W correctly with his certificate. The other $a_i = g^{w_i} v_i^{c_i} \bmod p, i \ne j$, are simulated by selecting the challenge c_i first. For W's challenge c, U has to fix the challenges $c_i, i \ne j$ first. Otherwise, he cannot give a correct response r_i. Then, $c_j = (c - \sum_{i\ne j} c_i) \bmod p - 1$ is fixed. With his certificate, U gives a correct response $r_j = w_j - c_j s_j \bmod p - 1$.

Correctness. If U has (α_j, v_j, s_j) with $v_j = g^{s_j} \bmod p$, he can compute $r_j = w_j - c_j s_j \bmod p - 1$ such that

$$g^{r_j} v_j^{c_j} \bmod p = g^{w_j - c_j s_j} g^{s_j c_j} \bmod p = g^{w_j} \bmod p = a_j.$$

For other $i \ne j$, we have $g^{r_i} v_i^{c_i} \bmod p = g^{w_i} v_i^{c_i} \bmod p = a_i$.

5.1 Security analysis

For the above scheme, we show two things. The first is that a non-member cannot pass the anonymous authentication scheme except with a negligible probability.[3] And, the second is that the system W cannot know the identity of U even with an unlimited computing power.

THEOREM 5.1 (UNFORGABILITY). *A non-member cannot pass authentication by the anonymous authentication scheme with a non-negligible probability unless he can solve the discrete logarithm modulo a prime with an overwhelming probability.*

PROOF. If a non-member A can pass the authentication scheme with a non-negligible probability ϵ, by the triangular inequality there is j, $1 \le j \le n$, such that A can impersonate as U_j with probability ϵ/n, which is non-negligible also. For a fixed selection j, the scheme is a zero-knowledge proof of knowledge of the discrete logarithm $\log_g v_j$. Since the success probability of impersonation is non-negligible ϵ/n, by the standard argument in the cryptographic field one can compute $s_j = \log_g v_j$ with an overwhelming probability using A as a subroutine. Thus, one can solve the discrete logarithm modulo a prime with an overwhelming probability. □

[3] A probability is negligible if it is bounded by a negligible function $\epsilon(k)$, which is smaller than any $1/Q(k)$ asymptotically, where $Q(k)$ is a polynomial and k is the security parameter.

THEOREM 5.2 (ANONYMITY). *The system server W can not know U's identity even if its computing power is unlimited.*

PROOF. Assume that U_j has the certificate (α_j, v_j, s_j) and $U_{j'}$ has the certificate $(\alpha_{j'}, v_{j'}, s_{j'})$. We show that the distribution of the exchanged messages between U_j and W is the same as that between $U_{j'}$ and W, $j' \neq j$.

For the distribution of the exchanged message between U_j and W, since w_i, $1 \leq i \leq n$, are randomly chosen over Z_{p-1}, a_i's are totally independent and each is uniformly distributed over Z_p^*. The values c_i's are of $(n-1)$-degree freedom under the constraint $c = c_1 + c_2 + \cdots + c_n \bmod p - 1$, where c is selected randomly by W. The value r_i is totally dependent on a_i and c_i, $1 \leq i \leq n$. This argument is the same for the distribution of the exchanged messages between $U_{j'}$ and W. Since the two distributions are the same, W cannot distinguish whom interacts with it. Therefore, user's identity is unknown to W even if W's computing power is unlimited. □

6. DATA ACCESS SYSTEM

Figure 2 shows a conventional model for data retrieval from a web site with authentication, authorization and communication security. U first sends his identity α to W. W gets the authentication data of α and authenticates α by an authentication protocol. If U passes the authentication, W and U execute a key-exchange protocol to establish a communication session key k. U then sends his data retrieval command ω to W. W checks whether ω is authorized. If it is so, W retrieves data D from its database system, encrypts D with k as $C = E(k, D)$, and sends the encrypted data C to U, where E is a symmetric encryption method, such as DES. Finally, U uses k to decrypt C to get D. Since W knows α, U is not anonymous to W.

For secure communication, W has to encrypt data D on-line. If there are requests for retrieving data in a short period of time, the on-line computation load of W would be heavy so that the system performance is lowered.

Figure 3 shows our proposed model for data retrieval from a web site. The data in the database of the web site is encrypted with the class keys of the time-dependent hierarchical key assignment scheme. The system authenticates a user's identity anonymously by an anonymous authentication scheme. The authorization is controlled by the key trapdoor that a user possesses. In this model, on-line encryption of communication is not necessary.

6.1 The system

W sets a partially ordered hierarchy with classes $C_i, 1 \leq i \leq m$, and assigns time-dependent class keys $C_{i,t}$, as described in Section 4.2. W also chooses an anonymous authentication scheme, as described in Section 5. W can perform the following operations.

1. **Storing new data**. When W decides to assign the new data D to class C_i at time t, it uses the time-dependent class key $K_{i,t}$ to encrypt D as $E(K_{i,t}, D)$ and put it into the database of the web site.
2. **User registration**. When a new user U registers to the system, W verifies its identity and then issues a certificate (α, v, s) to U. W adds U's authentication key (α, v) into the authentication list L. Then, W decides which class U should be in, say C_i, and what data are authorized to U, say between time periods t_1 and t_2. W issues the key trapdoor $K_{[i,t_1,t_2]}$ to U.
3. **Membership revocation**. When W need revoke U's membership, it simply removes U's authentication key from its authentication list L. Thus, U can no longer pass anonymous authentication with W.
4. **Anonymous authentication**. When U need retrieve data from the system, W performs anonymous authentication with U. If U passes the anonymous authentication, W starts to process the command.
5. **Command processing**. After passing anonymous authentication, U sends a data retrieval command ω to W. W simply sends the command to the database system for processing. Suppose that the database system returns data $D_{j,t} = E(K_{j,t}, D)$. W sends $D_{j,t}$ to U. If U has the appropriate key trapdoor $K_{[i,t_1,t_2]}$, he can decrypt $D_{j,t}$ to obtain D. Otherwise, U is not authorized to obtain D.

6.2 The user

After registering to W, U has two private parameters. One is the certificate (α, v, s) for anonymous authentication and the other is the key trapdoor $K_{[i,t_1,t_2]}$ for decrypting authorized data. A user U can perform the following operations.

1. **Anonymous authentication**. When U need retrieve data from the system, he uses his certificate (α, v, s) to execute the anonymous authentication scheme with W. If U passes the authentication, he sends his request command to W.
2. **Data decryption**. Assume that $D_{j,t}$ is returned by W. If the requested data is authorized, that is, $C_j \leq C_i$ and $t_1 \leq t \leq t_2$, U uses his key trapdoor $K_{[i,t_1,t_2]}$ to derive the time-dependent class key $K_{j,t}$. U decrypts $D_{j,t}$ with key $K_{j,t}$ to obtain D.

6.3 A subscription system

W can establish a flexible subscription system by the time-dependent hierarchical key assignment scheme. W classifies data into classes by various criteria, such as, categories, sensitivity, etc. We assume that the higher the class is, the more valuable the data in the class is. Each data is also tagged with time t. For example, a news is tagged with the time period it was reported. Then, the data D classified into class C_i of time period t is encrypted with key $K_{i,t}$ and stored into the database.

W places a price tag for class C_i and time periods $[t_1, t_2]$. If a user U pays for the data in class C_i between time periods t_1 and t_2, W gives him the key trapdoor $K_{[i,t_1,t_2]}$ so that he can decrypt the authorized data.

The above subscription system has some distinct features. Firstly, W can put its database on mirror sites for better

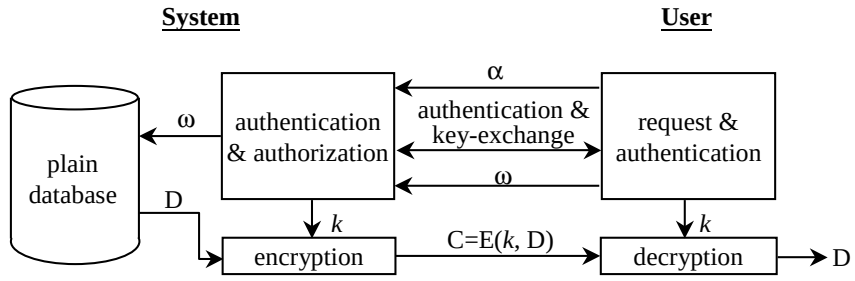

Figure 2: Conventional model for data retrieval with authentication, authorization and communication security.

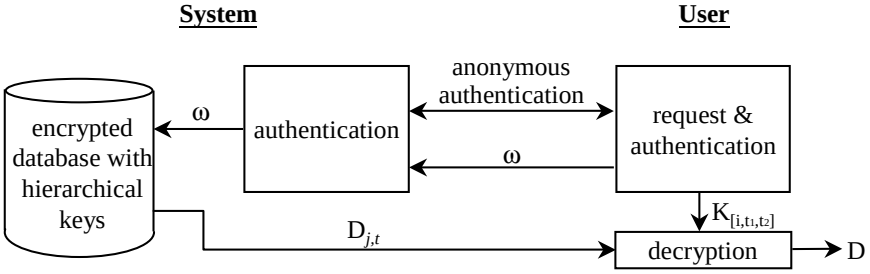

Figure 3: New model for data retrieval with anonymous authentication, key-controlled authorization and communication security.

services, such as faster access. Since mirror sites are less trusted, W does not want to put valuable information on them. By our system, W need not give user information to the mirror sites. The mirror sites cannot obtain the information in the database since the data are encrypted. Secondly, U need not rush to get all purchased information out of W. For conventional subscription systems, a user who pays for the data is allowed to access the database for a period of time. After the expiration date, the user can no longer access the database. Therefore, the user may want to get all data out of the database before his membership expires, no matter whether the data is useful to him or not. This sometimes causes severe traffic and system load. By our subscription system, the user can access the database as long as W does not revoke his membership. On the other hand, he can get only the data that he paid for.

6.4 Efficiency analysis

There are two efficiency problems in anonymous authentication. The first is that on-line computation for modular exponentiation is indeed necessary. Nevertheless, authentication is executed once for each visit, the computation load should not be a big problem for modern computers. The second one is that if the system has a large number of members, anonymous authentication is not efficient. For this case, we can sacrifice a little anonymity for efficiency by grouping members. Each group consists of a reasonable number of members. Each member belongs to a group. The system has group authentication lists $L_1, L_2, \ldots, L_r$. When a member visits the system, he first provides its group name to W. W then uses the group authentication list to authenticate the user anonymously. Although the system knows which group the user is in, it cannot know who the user is.

6.5 Discussion

A system may discard anonymous authentication and leaves access control to time-dependent class keys entirely. This shall save computation cost of anonymous authentication. We have discussed that the system's database may be put into mirror sites for faster access. Since mirror sites are less trusted, user authentication may not achieve its goal. Without authenticating users, the mirror sites can provide faster access and entail no serious security problems.

Most web site systems use only one-level hierarchy for data. In those systems, the access control depends solely on time periods. This reduces cost for computing class keys.

It is possible that two users team up to cheat as follows. An authorized user downloads data for another one who is not authorized to get. This is the problem for all systems that provide content information. It should be resolved by the legal system.

7. CONCLUSIONS

We have proposed a secure system for data access. The system provides an authentication mechanism so that the user's identity is anonymous. The system uses the time-dependent hierarchial key assignment scheme to control authorization. The system provides an integrated view for authentication, authorization and communication security. It would be interesting to have an implementation to check its feasibility and practicability.

8. REFERENCES

[1] S.G. Akl and P.D. Taylor. Cryptographic solution to a problem of access control in a hierarchy. *ACM Transactions on Computer Systems* 1(3), pp.239-248, 1983.

[2] D. Boneh,M. Franklin. Anonymous authentication with subset queries. *In Proceedings of The 6th ACM Conference on Computer and Communications Security*, ACM Press, 1999.

[3] C.C. Chang, R.J. Hwang, and T.C. Wu. Cryptographic key assignment scheme for access control in a hierarchy. *Information Systems* 17(3), pp.243-247, 1992.

[4] D. Chaum. Security without identification: transaction systems to make big brother obsolete. *Communications of the ACM* 28(10), pp.1030-1044, 1985.

[5] D. Chaum, J.-H. Evertse. A secure and privacy-protecting protocol for transmitting personal information between orgainzation. *In Proceedings of Advances in Cryptology - Crypto 86*, Lecture Notes in Computer Science 263, pp.118-167, 1986.

[6] D.E. Denning, D.K. Branstad. A taxonomy for key escrow encryption systems. *Communications of the ACM* 39(3), pp.34-40, 1996.

[7] U. Feige, A. Shamir. Witness indistinguishable and witness hiding protocols. *In Proceedings of The 22nd ACM Symposium on Theory of Computing*, pp.416-426, ACM Press, 1990.

[8] W. Ford. *Computer Communications Security: Principles, Standard Protocols and Techniques*. Prentice Hall, Englewood Cliffs, New Jersey, 1994.

[9] L. Harn and H.Y. Lin. A cryptographic key generation scheme for multilevel data security. *Computers & Security* 9(6), pp.539–546, 1990.

[10] ISO/IEC 9798-1. Information technology - Security techniques - Entity authentication mechanisms - Part 1: General model. ISO, Geneva, Switzerland, 1991 (first edition).

[11] ISO/IEC 9798-2. Information technology - Security techniques - Entity authentication - Part 2: Mechanisms using symmetric encipherment algorithms. ISO, Geneva, Switzerland, 1994 (first edition).

[12] ISO/IEC 9798-3. Information technology - Security techniques - Entity authentication mechanisms - Part 3: Entity authentication using a public-key algorithm. ISO, Geneva, Switzerland, 1993 (first edition).

[13] ISO/IEC 9798-4. Information technology - Security techniques - Entity authentication - Part 4: Mechanisms using a cryptographic check function. ISO, Geneva, Switzerland, 1995 (first edition).

[14] ISO/IEC 9798-5. Information technology - Security techniques - Entity authentication - Part 5: Mechanisms using zero knowledge techniques. ISO, Geneva, Switzerland, 1996 (draft).

[15] J. Kilian, E. Petrank. Identity escrow. *In Proceedings of Advances in Cryptology - Crypto 98*, Lecture Notes in Computer Science 1462, pp.169-185, Springer-Verlag, 1998.

[16] S.J. Mackinnon, P.D. Taylor, H. Meijer, and S.G. Akl. An optimal algorithm for assigning cryptographic keys to control access in a hierarchy. *IEEE Transactions on Computers* 34(9), pp.797-802, 1985.

[17] A. Lysyanskaya, R. Rivest, A. Sahai, S. Wolf. Pseudonym systems. *The 6th Annual Workshop on Selected Areas in Cryptography*, Lecture Notes in Computer Science 1758, Springer-Verlag, 1999.

[18] S. Muftic. *Security Mechanisms For Computer Networks*. Ellis Horwood, Chichester, England, 1989.

[19] R. Rivest, A. Shamir, L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM* 21(2), pp.120-126, 1978.

[20] B. Schneier. *Applied cryptography: protocol, algorithms, and source code in C, 2nd Edition*. John Wiley & Sons, New York, 1996.

[21] W.-G. Tzeng. Common modulus and chosen message attacks on public-key schemes with linear recurrence relations. *Information Processing Letters* 70, pp.153-156, 1999.

[22] W.-G. Tzeng. A time-bound cryptographic key assignment scheme for access control in a hierarchy. *IEEE Transactions on Knowledge and Data Engineering* 14(1), pp.182-188, 2002.

[23] W.-G. Tzeng, C.-M. Hu. Inter-protocol interleaving attacks on some authentication and key distribution protocols. *Information Processing Letters* 69(6), pp.297-302, 1999.

Conditional Oblivious Cast*

Cheng-Kang Chu and Wen-Guey Tzeng

Department of Computer Science,
National Chiao Tung University,
Hsinchu, Taiwan 30050
{ckchu, tzeng}@cis.nctu.edu.tw

Abstract. We introduce a new notion of *conditional oblivious cast* (COC), which involves three parties: a sender S and two receivers A and B. Receivers A and B own their secrets x and y, respectively, and the sender S holds the message m. In a COC scheme for the predicate Q (Q-COC), A and B send x and y in a masked form to S, and then S sends m to A and B such that they get m if and only if $Q(x, y) = 1$. Besides, the secrets x and y can not be revealed to another receiver nor the sender. We also extend COC to 1-out-of-2 COC (COC_2^1) in which S holds two messages m_0 and m_1, and A and B get m_1 if $Q(x, y) = 1$ and m_0 otherwise. We give the definitions for COC and COC_2^1, and propose several COC and COC_2^1 schemes for "equality", "inequality", and "greater than" predicates. These are fundamental schemes that are useful in constructing more complex secure interactive protocols. Our schemes are efficiently constructed via homomorphic encryption schemes and proved secure under the security of these encryption schemes.

Keywords: oblivious cast, conditional oblivious transfer, secure computation.

1 Introduction

Oblivious transfer (OT) is an important cryptographic primitive proposed by Rabin [18]. It involves two parties: the sender S and the receiver R, where S sends a bit of which R gets it with probability $\frac{1}{2}$. After Rabin's work, OT was developed in several types, such as 1-out-of-2 OT [11], 1-out-of-n OT [5, 16, 21], k-out-of-n OT [8, 14, 15], conditional OT (COT) [3, 10], etc. In Q-COT, S owns a secret x and a message m, and R owns a secret y such that R gets m from S if and only if the condition $Q(x, y)$ is evaluated as true.

Oblivious cast (OC) [12] is a generalization of OT to the three-party case: one sender S and two receivers A and B. The bit is received by exactly one of A and B, each with probability $\frac{1}{2}$. We generalize OC and introduce a new notion of *conditional oblivious cast* (COC), where A and B own their secrets x and y, respectively, and the sender S holds the message m. In a COC scheme for the predicate Q (Q-COC), A and B send x and y in a masked form to S, and

* Research supported in part by National Science Council grants NSC-94-2213-E-009-116, Taiwan, ROC.

M. Yung et al. (Eds.): PKC 2006, LNCS 3958, pp. 443–457, 2006.

then S sends m to A and B such that they get m if and only if $Q(x, y) = 1$. Furthermore, the secrets x and y can not be revealed to another receiver nor the sender. We also extend COC to 1-out-of-2 COC (COC_2^1) in which S holds two messages m_0 and m_1, and A and B get m_1 if $Q(x, y) = 1$ and m_0 otherwise.

There are two cases for the message receiving: A and B both get m, or only one of them gets m. The schemes we propose in this paper are all designed for the first case. However, in some applications only one receiver, determined by the condition, is allowed to get the message, and S can not know who gets the message. We have a general transformation of our COC_2^1 schemes to suit this kind of model (Section 4.3).

In this paper, we give the definitions for COC and COC_2^1, and propose several COC and COC_2^1 schemes for "equality", "inequality", and "greater than" predicates. These are fundamental schemes that are useful in constructing more complex secure interactive protocols. Our schemes are efficiently constructed via homomorphic encryption schemes and proved secure.

COC not only covers all functionalities of COT, but also broadens the range of its applications. We provide three examples:

- *Priced oblivious transfer*: Aiello et al. [1] introduced the notion of "priced oblivious transfer", which protects the privacy of a customer's purchase from a vendor. In their setting, the buyer needs to deposit an amount in each vendor. This is not very practical if a buyer wants to purchase various goods from many vendors. By using our COC schemes, we can construct a generalized priced OT such that the buyer can deposit the money in one bank only. When the buyer wants to buy an item from a vendor, he sends the corresponding price and the bank sends the buyer's current balance in the encryption form to the vendor. The vendor then sends the item such that the buyer can get it if the price does not exceed his balance.
- *Oblivious two-bidder system*: A party S has a secret for selling, and A and B are two bidders. The winner can obtain the secret from S directly. At the end, S has no idea who the winner is. This system can be constructed from COC for the "greater than" predicate (in the second message-receiving case) immediately.
- *Oblivious authenticated information retrieval*: A can get some information from S if he passes the authentication procedure provided by B. For instance, consider a mobile news subscription service provided by an independent agent. We assume that a mobile phone has no extra memory to store the subscription information but only an IMSI (International Mobile Subscriber Identity) in the SIM card. Users can pay the subscription fee to their mobile phone company, and the company provides an encrypted subscription list of IMSIs to the news provider. When a user wants to read news on the bus, his mobile phone sends the encrypted IMSI to the news provider. The news provider then sends news to the user if the IMSI is in the subscription list. In this case, the user's identity (IMSI) is anonymous to the news provider. The scheme can be constructed by COC for the "membership" predicate discussed in Section 5.2.

Related works. COT was first proposed by Di Crescenzo et al. [10]. In their definition of COT, the focus is to provide "all-or-nothing" transfer of the message from S to R by the condition. Blake et al. [3] strengthened COT to strong COT (SCOT), which provides "1-out-of-2" message transfer from S to R by the condition and adds more security requirements for S.

The notion of our COC is to separate the role of the secret holder from S. The main difference in design techniques is that, in COT and SCOT, the secure computation is done by S with a masked input and a plain input, whereas the secure computation in our COC and COC_2^1 is done by S with two masked inputs. A COC scheme that meets the requirements of our definitions can be easily transferred to a COT or SCOT scheme.

2 Definitions and Preliminaries

In this section we give formal definitions for COC and COC_2^1 and introduce useful tools and notations.

2.1 Conditional Oblivious Cast

Informally speaking, a COC scheme for predicate Q (Q-COC) has the following three properties:

- Correctness: both of A and B get m from S if $Q(x, y) = 1$.
- Sender's security: A and B cannot get any information about m if $Q(x, y) = 0$.
- Receiver's security: after running the protocol, x is kept secret from B and S, and y is kept secret from A and S.

The definition for Q-COC is as follows:

Definition 1 (Q-COC). *Let k be the security parameter, and A, B and S be all polynomial-time probabilistic Turing machines (PPTMs). Let $\langle A, B, S\rangle(\cdot)$ denote the communication transcript. We say that a three-party interactive system $\Pi = (A, B, S)$ is a secure Q-COC scheme if it satisfies the following requirements for some constant c:*

1. *Correctness: For any $x, y, m \in \{0,1\}^{k^c}$ with $Q(x, y) = 1$,*
   $\Pr[\mu \leftarrow \{0,1\}^{k^c}; tr \leftarrow \langle A(x), B(y), S(m)\rangle(\mu) :$
   $\text{“}A(x, \mu, tr) = m\text{”} \wedge \text{“}B(y, \mu, tr) = m\text{”}] = 1.$
2. *Sender's security: For any PPTM A', B' and any $x, y, m, m' \in \{0,1\}^{k^c}$ with $Q(x, y) = 0$, A' and B' cannot distinguish the following probability ensembles with non-negligible advantage, respectively:*
   - $V^{\Pi}_{A'B'} = (x, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B'(y), S(m)\rangle(\mu))$,
   - $R^{\Pi}_{A'B'} = (x, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B'(y), S(m')\rangle(\mu))$,

   and
   - $V^{\Pi}_{B'A'} = (y, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B'(y), S(m)\rangle(\mu))$,
   - $R^{\Pi}_{B'A'} = (y, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B'(y), S(m')\rangle(\mu))$.

3. *Receiver's security:*
   (a) *For any PPTM A', B', S' and any $x, x', y, y', m \in \{0,1\}^{k^c}$ with $Q(x,y) = Q(x,y') = Q(x',y)$, S' cannot distinguish the following probability ensembles with non-negligible advantage:*
   - $V^{\Pi}_{S'A'} = (m, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B(y), S'(m)\rangle(\mu))$,
   - $S^{\Pi}_{S'A'} = (m, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B(y'), S'(m)\rangle(\mu))$,

   and
   - $V^{\Pi}_{S'B'} = (m, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A(x), B'(y), S'(m)\rangle(\mu))$,
   - $S^{\Pi}_{S'B'} = (m, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A(x'), B'(y), S'(m)\rangle(\mu))$.

   (b) *For any PPTM A', B', S' and any $x, x', y, y', m \in \{0,1\}^{k^c}$ with $Q(x,y) = Q(x,y') = Q(x',y)$, A' and B' cannot distinguish the following probability ensembles with non-negligible advantage, respectively:*
   - $V^{\Pi}_{A'S'} = (x, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B(y), S'(m)\rangle(\mu))$,
   - $S^{\Pi}_{A'S'} = (x, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B(y'), S'(m)\rangle(\mu))$,

   and
   - $V^{\Pi}_{B'S'} = (y, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A(x), B'(y), S'(m)\rangle(\mu))$,
   - $S^{\Pi}_{B'S'} = (y, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A(x'), B'(y), S'(m)\rangle(\mu))$.

2.2 1-Out-of-2 Conditional Oblivious Cast

In COC_2^1, the message sender S holds two messages m_0 and m_1. A Q-COC_2^1 scheme must satisfy the following three properties:

- Correctness: both of A and B get m_1 from S if $Q(x,y) = 1$, and m_0 if $Q(x,y) = 0$.
- Sender's security: A and B get exactly one message from S.
- Receiver's security: after running the protocol, x is kept secret from B and S, and y is kept secret from A and S.

The definition for Q-COC_2^1 is as follows.

Definition 2 (Q-COC_2^1). *Let k be the security parameter, and A, B and S be all PPTMs. Let $\langle A, B, S\rangle(\cdot)$ denote the communication transcript. We say that a three-party interactive system $\Pi = (A, B, S)$ is a secure Q-COC_2^1 scheme if it satisfies the following requirements for some constant c:*

1. *Correctness:*
   (a) *For any $x, y, m_0, m_1 \in \{0,1\}^{k^c}$ with $Q(x,y) = 0$,*
   $$\Pr[\mu \leftarrow \{0,1\}^{k^c}; tr \leftarrow \langle A(x), B(y), S(m_0, m_1)\rangle(\mu) : \text{“}A(x,\mu,tr) = m_0\text{”} \wedge \text{“}B(y,\mu,tr) = m_0\text{”}] = 1.$$
   (b) *For any $x, y, m_0, m_1 \in \{0,1\}^{k^c}$ with $Q(x,y) = 1$,*
   $$\Pr[\mu \leftarrow \{0,1\}^{k^c}; tr \leftarrow \langle A(x), B(y), S(m_0, m_1)\rangle(\mu) : \text{“}A(x,\mu,tr) = m_1\text{”} \wedge \text{“}B(y,\mu,tr) = m_1\text{”}] = 1.$$
2. *Sender's security: For any PPTM A', B' and any $x, y, m_0, m_1, m_1' \in \{0,1\}^{k^c}$ with $Q(x,y) = 0$, A' and B' cannot distinguish the following probability ensembles with non-negligible advantage, respectively:*
   - $V^{\Pi}_{A'B'} = (x, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B'(y), S(m_0, m_1)\rangle(\mu))$,
   - $R^{\Pi}_{A'B'} = (x, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B'(y), S(m_0, m_1')\rangle(\mu))$,

and
- $V^{\Pi}_{B'A'} = (y, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B'(y), S(m_0, m_1)\rangle(\mu))$,
- $R^{\Pi}_{B'A'} = (y, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B'(y), S(m_0, m_1')\rangle(\mu))$.

The similar requirements is met $Q(x, y) = 1$.

3. *Receiver's security:*
   (a) *For any PPTM* A', B', S' *and any* $x, x', y, y', m_0, m_1 \in \{0,1\}^{k^c}$ *with* $Q(x, y) = Q(x, y') = Q(x', y)$, S' *cannot distinguish the following probability ensembles with non-negligible advantage:*
   - $V^{\Pi}_{S'A'} = (m_0, m_1, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B(y), S'(m_0, m_1)\rangle(\mu))$,
   - $S^{\Pi}_{S'A'} = (m_0, m_1, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B(y'), S'(m_0, m_1)\rangle(\mu))$,

   and
   - $V^{\Pi}_{S'B'} = (m_0, m_1, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A(x), B'(y), S'(m_0, m_1)\rangle(\mu))$,
   - $S^{\Pi}_{S'B'} = (m_0, m_1, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A(x'), B'(y), S'(m_0, m_1)\rangle(\mu))$.

   (b) *For any PPTM* A', B', S' *and any* $x, x', y, y', m_0, m_1 \in \{0,1\}^{k^c}$ *with* $Q(x, y) = Q(x, y') = Q(x', y)$, A' *and* B' *cannot distinguish the following probability ensembles with non-negligible advantage, respectively:*
   - $V^{\Pi}_{A'S'} = (x, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B(y), S'(m_0, m_1)\rangle(\mu))$,
   - $S^{\Pi}_{A'S'} = (x, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A'(x), B(y'), S'(m_0, m_1)\rangle(\mu))$,

   and
   - $V^{\Pi}_{B'S'} = (y, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A(x), B'(y), S'(m_0, m_1)\rangle(\mu))$,
   - $S^{\Pi}_{B'S'} = (y, \mu \leftarrow \{0,1\}^{k^c}, tr \leftarrow \langle A(x'), B'(y), S'(m_0, m_1)\rangle(\mu))$.

Remark. For clarity and simplicity, we will first assume that all parties in our COC and COC_2^1 schemes are semi-honest (honest-but-curious), that is, they follow the procedure step by step, but try to get extra information about the secrets or messages by extra computation. We also assume that A, B and S operates independently. No two parties will collude against the third one. Then we provide some techniques to transform the schemes into ones that are secure against malicious parties and their collusion in Section 5.1.

2.3 Homomorphic Encryption Schemes

Multiplicatively homomorphic encryption scheme. An encryption scheme (G, E, D) is multiplicatively homomorphic if for any m_0 and m_1, $D(E(m_0) \otimes E(m_1)) = D(E(m_0 \cdot m_1))$, where $\otimes$ is an operation defined on the image of E.

The ElGamal encryption scheme as follows is multiplicatively homomorphic.

- $G(1^k) = (p, q, g, \alpha, \beta)$, where p is a k-bit prime, and $q = \frac{p-1}{2}$ is also a prime, $\mathbb{G}_q$ is the subgroup of $\mathbb{Z}_p^*$ with order q, g is a generator of $\mathbb{G}_q$, and $\beta = g^\alpha \bmod p$ for $\alpha \in \mathbb{G}_q$. Let $PK = (p, q, g, \beta), SK = (p, q, g, \alpha)$. All relevant computations are under group $\mathbb{G}_q$.
- $E(m) = (g^r, m\beta^r)$, where $m \in \mathbb{G}_q, r \in_R \mathbb{Z}_q$.
- $D(c) = c_2/c_1^\alpha$, where $c = (c_1, c_2)$.

For $E(m_0) = (g^{r_0}, m_0\beta^{r_0})$ and $E(m_1) = (g^{r_1}, m_1\beta^{r_1})$, the operation $E(m_0) \times E(m_1) = (g^{r_0} \cdot g^{r_1}, m_0\beta^{r_0} \cdot m_1\beta^{r_1})$ is multiplicatively homomorphic since

$$\begin{aligned} D(E(m_0) \times E(m_1)) &= D(g^{r_0} \cdot g^{r_1}, m_0\beta^{r_0} \cdot m_1\beta^{r_1}) \\ &= D(g^{r_0+r_1}, m_0 m_1 \beta^{r_0+r_1}) \\ &= D(E(m_0 \cdot m_1)). \end{aligned}$$

We can compute $E(m^c)$ from $E(m)$ via repeated multiplication for a constant c.

Additively homomorphic encryption scheme. An encryption scheme (G, E, D) is additively homomorphic if for any m_0 and m_1, $D(E(m_0) \oplus E(m_1)) = D(E(m_0 + m_1))$, where $\oplus$ is an operation defined on the image of E.

The Paillier encryption scheme [17] as follows is additively homomorphic.

- $G(1^k) = (p, q, N, \alpha, g)$, where $N = pq$ is a k-bit number, p and q are two large primes, g is an integer of order $\alpha N \bmod N^2$ for some integer α. Let $PK = (g, N), SK = \lambda(N) = \mathrm{lcm}(p-1, q-1)$.
- $E(m) = g^m r^N \bmod N^2$, where $m \in \mathbb{Z}_N, r \in_R \mathbb{Z}_N$.
- $D(c) = \frac{L(c^{\lambda(N)} \bmod N^2, N)}{L(g^{\lambda(N)} \bmod N^2, N)} \bmod N$, where $L(u, N) = \frac{u-1}{N}$.

For $E(m_0) = g^{m_0} r_0^N \bmod N^2, E(m_1) = g^{m_1} r_1^N \bmod N^2$, the operation $E(m_0) \cdot E(m_1) = (g^{m_0} r_0^N) \cdot (g^{m_1} r_1^N)$ is additively homomorphic since

$$\begin{aligned} D(E(m_0) \cdot E(m_1)) &= D((g^{m_0} r_0^N) \cdot (g^{m_1} r_1^N)) \\ &= D((g^{m_0+m_1}(r_0 r_1)^N)) \\ &= D(E(m_0 + m_1)). \end{aligned}$$

We can compute $E(cm)$ from $E(m)$ via repeated addition for a constant c.

Note that ElGamal and Paillier encryption schemes are proved semantically secure if and only if the Decisional Diffie-Hellman and the Computational Composite Residuosity assumptions hold, respectively [20, 17].

2.4 0-Encoding and 1-Encoding

In our COC scheme for "greater than" predicate, we use two types of encoding to reduce the "greater than" problem to the set intersection problem [13]. Let $s = s_n s_{n-1} \dots s_1 \in \{0,1\}^n$ be a binary string of length n. The 0-encoding of s is

$$\hat{S}_s^0 = \{s_n s_{n-1} \dots s_{i+1} 1 | s_i = 0, 1 \le i \le n\}.$$

and 1-coding of s is

$$\hat{S}_s^1 = \{s_n s_{n-1} \dots s_i | s_i = 1, 1 \le i \le n\}.$$

For two binary strings x, y of the same length, we have that $x > y$ if and only if there is exact one common element in $\hat{S}_x^1$ and $\hat{S}_y^0$.

If we compare strings in $\hat{S}_x^1$ and $\hat{S}_y^0$ one against one, it would be quite inefficient since we need $O(n^2)$ comparisons. Because each element in $\hat{S}_s^0$ (or $\hat{S}_s^1$) has a different length, we compare the elements of the same length in the two sets only. We define the *ordered* sets for $b \in \{0,1\}, 1 \le i \le n$:

$$S_s^b[i] = \begin{cases} z_i & \text{if } \exists z_i \in \hat{S}_s^b \text{ and } |z_i| = i; \\ r_i^b & \text{otherwise,} \end{cases}$$

where $S_s^b[i]$ denotes the i-th element in S_s^b, and r_i^b is an arbitrary binary string with length $i+1+b$. Therefore, because of different lengths, r_i^b must not be equal to the string $S_s^{1-b}[i]$. Thus we just need to test if $S_x^1[i] = S_y^0[i]$ for each $i \in \{1, 2, \dots, n\}$.

2.5 Setup and Notations

In the setup phase of our schemes for semi-honest adversary, A and B need to agree on a public/secret key pair (PK, SK) of the homomorphic encryption scheme privately. There are several ways to accomplish this work. For example, if A and B have their own public/secret key pairs, one party generates (PK, SK) first, and securely sends it to the other party. This common key pair allows S to compute the predicate on their secrets by the homomorphic encryption scheme. Also, S need choose a key pair (PK_S, SK_S) (for any semantically secure public key encryption scheme) such that A and B can send their secrets to S privately (against the other party).

Let $\mathbb{G}_q$ be the group of the multiplicatively homomorphic encryption scheme and $\mathbb{Z}_N$ be the group of the additively homomorphic encryption scheme. For key pair (PK, SK), E_{PK} and D_{SK} represent encryption and decryption for the underlying encryption scheme.

We use x_i to denote the i-th bit of the value $x = x_n x_{n-1} \cdots x_1$. Let $X[i]$ denote the i-th element of the ordered set X. Let $x \in_R X$ mean that x is chosen from X uniformly and independently. Let $|x|$ be the length (in bits) of x. To encrypt a vector $v = \langle v_1, v_2, \ldots, v_n \rangle$, we write $E(v) = \langle E(v_1), E(v_2), \ldots, E(v_n) \rangle$.

In some schemes, A and B need to "identify" the correct message from a set of decrypted ciphertexts. This can be achieved by some padding technique (e.g. OAEP [2]) such that receivers can check the integrity of a message. If a decryption contains the valid padding, it is the correct message with overwhelming probability.

3 Conditional Oblivious Cast

We provide COC schemes for three basic predicates: "equality", "inequality", and "greater than".

3.1 COC for "Equality" Predicate

To determine if $x = y$, we compute x/y via the multiplicatively homomorphic encryption scheme. If $x/y = 1$, A and B get the message m; otherwise, they get nothing. The scheme EQ-COC is described in Figure 1.

Theorem 1. *The EQ-COC scheme has the correctness property, unconditional sender's security, and computational receiver's security if the underlying homomorphic encryption scheme has semantic security.*

Proof. For correctness, if $x = y$, A and B compute m by

$$\begin{aligned} D_{SK}(e) &= D_{SK}(E_{PK}(m) \otimes (E_{PK}(x) \otimes E_{PK}(y)^{-1})^r) \\ &= D_{SK}(E_{PK}(m) \otimes (E_{PK}(1)^r)) \\ &= D_{SK}(E_{PK}(m)) \\ &= m. \end{aligned}$$

- System parameters: (p, q, g).
- Message sender S has a message m and a key pair (PK_S, SK_S).
- Receiver A has a secret x, and receiver B has a secret y, where $x, y \in \mathbb{G}_q$.
- Receiver A and B have a common key pair (PK, SK)

1. A and B send $E_{PK_S}(E_{PK}(x))$ and $E_{PK_S}(E_{PK}(y))$ to S respectively.
2. S decrypts the received messages to get $E_{PK}(x)$ and $E_{PK}(y)$. S computes

$$e = E_{PK}(m) \otimes (E_{PK}(x) \otimes E_{PK}(y)^{-1})^r$$

   and sends it to A and B, where $r \in_R \mathbb{Z}_q$.
3. A and B compute $\hat{m} = D_{SK}(e)$ and identify whether $\hat{m}$ is valid.

Fig. 1. COC scheme for "Equality" predicate: EQ-COC

For sender's security, we show that if $x \neq y$, m is unconditionally secure to A and B. Since $e = E_{PK}(m) \otimes (E_{PK}(x) \otimes E_{PK}(y)^{-1})^r) = E_{PK}(m \cdot (x/y)^r), r \in_R \mathbb{Z}_q$, for any possible m', there is another $r' \in \mathbb{Z}_q$ such that $e = E_{PK}(m' \cdot (x/y)^{r'})$. As long as $x \neq y$, e can be decrypted to any possible message in $\mathbb{G}_q$. This ensures unconditional security of S's message m.

For receiver's security, it is easy to see that S gets no information about x and y due to semantic security of the encryption scheme. Since A and B are symmetric, we only prove the security of B against A. We construct a simulator S_A for A's real view

$$V_A(PK, SK, PK_S, x) = (PK, SK, PK_S, x, E_{PK_S}(E_{PK}(x)), E_{PK_S}(E_{PK}(y)), e).$$

The simulator S_A on input $(PK, SK, PK_S, x, \hat{m})$ is as follows, where $\hat{m}$ (may be a valid message or a random value) is the output of a real execution:

1. Choose a random value $y^* \in \mathbb{G}_q$.
2. Compute $e^* = E_{PK}(\hat{m})$.
3. Output $(PK, SK, PK_S, x, E_{PK_S}(E_{PK}(x)), E_{PK_S}(E_{PK}(y^*)), e^*)$.

By semantic security of the encryption scheme, A cannot distinguish the ciphertexts $E_{PK_S}(E_{PK}(y^*))$ and $E_{PK_S}(E_{PK}(y))$. Furthermore, since e^* is identically distributed as e, the output of S_A is indistinguishable from V_A. Therefore, A gets no information about y except those computed from x and $\hat{m}$. □

In the scheme, we assume $x, y \in \mathbb{G}_q$. If the length of x (or y) is longer than $|p|$, A and B compare $h(x)$ and $h(y)$, where h is a collision-resistant hash function. This technique is applied to later schemes whenever necessary.

3.2 COC for "Inequality" Predicate

COC for the "inequality" predicate is more complicated than that for the "equality" predicate. A and B need to send the ciphertexts of their secrets bit by bit. We use additively homomorphic encryption schemes in this scheme, which is depicted in Figure 2.

- System parameters: n.
- Message sender S has a message m and a key pair (PK_S, SK_S).
- Receiver A has a secret x, and receiver B has a secret y, where $|x| = |y| = n$.
- Receiver A and B have a common key pair (PK, SK), where $PK = (g, N)$.

1. A and B send $E_{PK_S}(E_{PK}(x_i))$ and $E_{PK_S}(E_{PK}(y_i))$ to S respectively, $1 \leq i \leq n$.
2. For each $i \in \{1, 2, \ldots, n\}$, S decrypts the received messages to get $E_{PK}(x_i)$ and $E_{PK}(y_i)$, and computes the following values via homomorphic encryption:
   (a) $d_i = x_i - y_i$, $d'_i = x_i + y_i - 1$.
   (b) $e_i = 2e_{i+1} + d_i$, where $e_{n+1} = 0$.
   (c) $c_i = m + r_i(e_i - d_i + d'_i)$, where $r_i \in_R \mathbb{Z}_N$
3. S sends $E_{PK}(c)$ in a random order to A and B, where $c = \langle c_1, c_2, \ldots, c_n \rangle$.
4. A and B decrypt the received messages and identify the correct message if existent.

Fig. 2. COC scheme for "Inequality" predicate: INE-COC

In the scheme, $d_i = x_i - y_i$ and $d'_i = x_i - \bar{y}_i$ are 0, 1 or -1. If $x_i = y_i$, $d_i = 0$; otherwise, $d'_i = 0$. Let l be the leftmost different bit between x and y, i.e. the largest i such that $d_i \neq 0$. We have $e_i = 0$ if $i > l$, $e_i \neq 0$ if $i < l$, and $e_i = d_i$ if $i = l$.

If $x \neq y$, the message m is embedded into the index i at which x_i and y_i are distinct. However, we have to avoid leaking information of the number of distinct bits. So S masks m with random values on all indices except the index l. It leaves only one copy of m in c_i's:

- For $i = l$, since $e_l = d_l$ and $d'_l = x_l - \bar{y}_l = 0$, $(e_l - d_l + d'_l) = 0$. Therefore, $c_l = m$.
- For $1 \leq i < l$, c_i would be a random value because $e_i - d_i + d'_i = 2e_{i+1} + d'_i \neq 0$ and $r_i \in_R \mathbb{Z}_N$.
- For $l < i \leq n$, c_i is also a random value because $e_i = d_i = 0$, $d'_i \neq 0$ and $r_i \in_R \mathbb{Z}_N$.

Theorem 2. *The INE-COC scheme has the correctness property, unconditional sender's security, and computational receiver's security if the underlying homomorphic encryption scheme has semantic security.*

Proof. (sketch) Let l be the index of the first different bit of x and y (from the most significant bit). We see that $d_l = e_l = x_l - y_l = 1$ or -1, and $d'_l = x_j - \bar{y}_j = 0$. Therefore, $c_l = m + r_l(e_l - d_l + d'_l) = m + r_l \cdot 0 = m$. Thus, A and B get m from the permutation of the encryptions.

For sender's security, we see that if $x = y$, all d_i's and e_i's are 0, and all d'_i's are not 0 (in fact, $+1$ or -1). Thus, for each index i, $c_i = m + r_i(0 \pm 1) = m \pm r_i$. Since for any possible $\tilde{m}$, there exists an $\tilde{r}_i$ such that $c_i = \tilde{m} + \tilde{r}_i$, m is unconditionally secure to A and B.

For receiver's security, S gets no information about x and y by the semantic security of the encryption scheme. As in the proof of EQ-COC, for each of A and B, we can construct a simulator such that the adversary cannot distinguish the real view and the simulated view. Therefore the receiver's security holds. □

- System parameters: (p, q, g).
- Message sender S has a message m and a key pair (PK_S, SK_S).
- Receiver A has a secret x, and receiver B has a secret y, where $x, y \in \mathbb{G}_q, |x| = |y| = n$.
- Receiver A and B have a common key pair (PK, SK)

1. A encodes x as S_x^1, and sends $E_{PK_S}(E_{PK}(S_x^1[i]))$ to S, $1 \leq i \leq n$.
2. B encodes y as S_y^0, and sends $E_{PK_S}(E_{PK}(S_y^0[i]))$ to S, $1 \leq i \leq n$.
3. S decrypts the received messages and computes

$$e_i = E_{PK}(m) \otimes (E_{PK}(S_x^1[i]) \otimes E_{PK}(S_y^0[i])^{-1})^{r_i},$$

   where $r_i \in_R \mathbb{G}_q$, $1 \leq i \leq n$. S sends e_i's to A and B in a random order.
4. A and B search $\hat{m}_i = D_{SK}(e_i)$, $1 \leq i \leq n$, to identify the correct m if existent.

Fig. 3. COC scheme for "Greater Than" predicate: GT-COC

3.3 COC for "Greater Than" Predicate

For the "greater than" predicate, we use the encoding methods mentioned in Section 2.4. A encodes x via 1-encoding and B encodes y via 0-encoding. The problem is then reduced to the "equality" problem immediately. When S receives encrypted S_x^1 and S_y^0, he checks equality for corresponding strings. The scheme is presented in Figure 3. The security argument is the same as the proof of the EQ-COC scheme. This method is more efficient than the GT-COC$_2^1$ scheme (in the next section, by setting m_0 as a random number).

4 1-Out-of-2 Conditional Oblivious Cast

In this section, we present COC$_2^1$ schemes for the "equality" ("inequality") and "greater than" predicates.

4.1 COC$_2^1$ for "Equality" Predicate

Our COC$_2^1$ scheme for the equality predicate is naturally extended from the EQ-COC and INE-COC schemes. Intuitively, if $x = y$, A and B get m_1 by the EQ-COC scheme and, otherwise, they get m_0 by the INE-COC scheme. For better integration, we modify the EQ-COC scheme to use additively homomorphic encryption schemes. The scheme is shown in Figure 4. It is almost the same as the INE-COC scheme except that S sends an extra ciphertext c_{eq} to A and B.

Theorem 3. *The EQ-COC$_2^1$ scheme has the correctness property, unconditional sender's security, and computational receiver's security if the underlying homomorphic encryption scheme has semantic security.*

Proof. (sketch) We see that if $x = y$, all d_i's are equal to 0, and c_{eq} is equal to m_1. The opposite case holds by the same arguments in the proof of Theorem 2. This ensures the correctness property.

- System parameters: n.
- Message sender S has messages: (m_0, m_1) and a key pair (PK_S, SK_S).
- Receiver A has a secret x, and receiver B has a secret y, where $|x| = |y| = n$.
- Receiver A and B have a common key pair (PK, SK), where $PK = (g, N)$.

1. A and B send $E_{PK_S}(E_{PK}(x_i))$ and $E_{PK_S}(E_{PK}(y_i))$ to S respectively, $1 \leq i \leq n$.
2. For each $i \in \{1, 2, \ldots, n\}$, S decrypts the received messages to get $E_{PK}(x_i)$ and $E_{PK}(y_i)$, and computes the following values via homomorphic encryption:
   (a) $d_i = x_i - y_i, d'_i = x_i + y_i - 1$.
   (b) $e_i = 2e_{i+1} + d_i$, where $e_{n+1} = 0$.
   (c) $c_{eq} = m_1 + \sum_{i=1}^{n} r_i d_i$, $c'_i = m_0 + r'_i(e_i - d_i + d'_i)$, where $r_i, r'_i \in_R \mathbb{Z}_N$
3. S sends $E_{PK}(c_{eq}), E_{PK}(c')$ to A and B in a random order, where $c' = \langle c'_1, c'_2, \ldots, c'_n \rangle$.
4. A and B decrypt the received messages and identify the correct message

Fig. 4. 1-out-of-2 COC scheme for "Equality" predicate: EQ-COC$_2^1$

For sender's security, let $r = \sum_{i=1}^{n} r_i d_i$. Since $r_i \in_R \mathbb{Z}_N$, if $x \neq y$, there is a $d_i \neq 0$ such that r is uniformly distributed, and thus m_1 is unconditionally secure to A and B. If $x = y$, by the proof of Theorem 2, m_0 is unconditionally secure to A and B.

For receiver's security, S gets no information about x and y by the semantic security of the encryption scheme. For each of A and B, we can construct a simulator such that the adversary cannot distinguish the real view and the simulated view. The receiver's security holds. □

4.2 COC$_2^1$ for "Greater Than" Predicate

It is obvious that we can apply the GT-COC scheme twice to achieve a GT-COC$_2^1$ scheme. One invocation is for testing $x > y$ and the other one is for testing $x \leq y$. But, this approach costs twice as much as the GT-COC scheme. Our scheme for GT-COC$_2^1$ in Figure 5 is more efficient. It costs an extra ciphertext (for the case $x = y$) from S to A and B only.

Let l be the leftmost different bit between x and y. For $i < l$ and $i > l$, e_i and e'_i would be random values in $\mathbb{Z}_N$, respectively. When $i = l$, we have $e_i = d_i$ and $e'_i = 0$. Therefore, f_i is a random value when $i \neq l$ and $f_l = d_l$. If $x > y$, $f_l = 1$ and thus $c_l = m_1$; if $x < y$, $f_l = -1$ and thus $c_l = m_0$. For the case $x = y$, we use an extra value c_{eq} to embed m_0 like scheme EQ-COC$_2^1$.

Theorem 4. *The GT-COC$_2^1$ scheme has the correctness property, unconditional sender's security, and computational receiver's security if the underlying homomorphic encryption scheme has semantic security.*

Proof. (sketch) For correctness, consider the following three cases:

- $x > y$: let l be the index of the first different bit of x and y (from the most significant bit), we have $e_l = d_l = 1, e'_l = d'_l = 0$, and thus $f_l = e_l + e'_l = 1$. Therefore $c_l = \frac{m_1 - m_0}{2} \cdot 1 + \frac{m_1 + m_0}{2} = m_1$.

- System parameters: n.
- Message sender S has messages: (m_0, m_1) and a key pair (PK_S, SK_S).
- Receiver A has a secret x, and receiver B has a secret y, where $|x| = |y| = n$.
- Receiver A and B have a common key pair (PK, SK), where $PK = (g, N)$.

1. A and B send $E_{PK_S}(E_{PK}(x_i))$ and $E_{PK_S}(E_{PK}(y_i))$ to S respectively, $1 \leq i \leq n$.
2. For each $i \in \{1, 2, \ldots, n\}$, S decrypts the received messages to get $E_{PK}(x_i)$ and $E_{PK}(y_i)$, and computes the following values via homomorphic encryption:
   (a) $d_i = x_i - y_i$, $d'_i = x_i + y_i - 1$
   (b) $e_i = r_i e_{i+1} + d_i$, $e'_i = r'_i d'_i$, where $e_{n+1} = 0$, $r_i, r'_i \in_R \mathbb{Z}_N$
   (c) $f_i = e_i + e'_i$
   (d) $c_i = \frac{m_1 - m_0}{2} f_i + \frac{m_1 + m_0}{2}$, $c_{eq} = m_0 + \sum_{i=1}^{n} r''_i d_i$, where $r''_i \in_R \mathbb{Z}_N$.
3. S sends $E_{PK}(c), E_{PK}(c_{eq})$ in a random order to A and B, where $c = \langle c_1, c_2, \ldots, c_n \rangle$.
4. A and B decrypt the received messages and identify the correct message.

Fig. 5. 1-out-of-2 COC scheme for "Greater Than" predicate: GT-COC_2^1

- $x < y$: similarly, since $f_l = e_l = d_l = -1$ in this case, we have $c_l = \frac{m_1 - m_0}{2} \cdot (-1) + \frac{m_1 + m_0}{2} = m_0$.
- $x = y$: by the same argument in the proof of Theorem 3, A and B get m_0 from c_{eq}.

For sender's security, we see that if $x \neq y$, then for all $i \neq l$, f_i is uniformly distributed in $\mathbb{Z}_N$. That is, all c_i's except c_l are uniformly distributed in $\mathbb{Z}_N$. For index l, according to the above argument, $c_l = m_0$ if $x < y$ and $c_l = m_1$ if $x > y$. Moreover, by the proof of Theorem 3, $c_{eq} = m_0$ if $x = y$, and c_{eq} is uniformly distributed if $x \neq y$. Therefore, m_0 is unconditionally secure to A and B if $x > y$, and m_1 is unconditionally secure to A and B if $x \leq y$.

For receiver's security, S gets no information about x and y by the semantic security of the encryption scheme. As in the previous proofs, for each of A and B, we can construct a simulator such that the adversary cannot distinguish the real view and the simulated view. Therefore, the receiver's security holds. □

4.3 A General Transformation

We provide a general transformation from COC_2^1 to the second case mentioned in Section 1 for COC. We use the GT-COC_2^1 scheme as an example. The alternative model for COC is that when $x > y$, only A gets the message m and when $x \leq y$, only B gets the message. We modify our GT-COC_2^1 scheme to meet this requirement. In the beginning, A and B choose their own public/secret key pairs, namely, (PK_A, SK_A) and (PK_B, SK_B). Then S lets $m_1 = E_{PK_A}(m)$ and $m_0 = E_{PK_B}(m)$, and performs the scheme as usual. We see that if $x > y$, both A and B get $m_1 = E_{PK_A}(m)$. But, only A can decrypt it to get the message m. Similarly, if $x \leq y$, only B gets the message.

5 Extensions

In this section we introduce how to modify our COC schemes against malicious parties and collusion. We also discuss the construction of other predicates. The details of these modifications and extensions are left to the full version of this paper.

5.1 Schemes Secure Against Malicious Parties and Collusion

We can make our COC schemes secure against malicious parties and their collusion by using the threshold version of homomorphic cryptosystems. At the initial stage, each party gets a secret key share (from a dealer or a distributed key generation protocol). If the number of collusive parties does not exceed the threshold, they get nothing about the message. Since all parties (including the sender) exchange messages in encrypted form, all computation can be publicly verified. After the final result in encrypted form is obtained, all parties perform the threshold decryption for the result.

We need some non-interactive zero-knowledge proof systems for verification in the corresponding schemes (assuming PK is the common public key):

- Proof of plaintext knowledge. The prover proves that he knows the plaintext x for the encryption $E_{PK}(x)$ he created.
- Proof of one-bit plaintext. The prover proves that x is 0 or 1 for the encryption $E_{PK}(x)$ he created.
- Proof of correct exponentiation. Given (multiplicatively homomorphic) $E_{PK}(x)$, the prover outputs $E_{PK}(a)$ and $E_{PK}(x^a)$, and proves that $E_{PK}(x^a)$ is indeed the encryption of x^a.
- Proof of correct multiplication. Given (additively homomorphic) $E_{PK}(x)$, the prover outputs $E_{PK}(a)$ and $E_{PK}(ax)$, and proves that $E_{PK}(ax)$ is indeed the encryption of ax.

We can find such proof systems for the ElGamal and Paillier homomorphic encryption schemes [7, 19, 6, 9]. For the schemes INE-COC, EQ-COC$_2^1$ and GT-COC$_2^1$, the receivers need to prove that the encrypted messages they send are indeed the encryptions of 0 or 1. Boneh et al. [4] provide a verification gadget for this type of checking. Thus we can avoid using the proof system of one-bit plaintext.

5.2 Other Predicates

In addition to the basic predicates, we can design COC (COC$_2^1$) schemes for many other interesting predicates. For these predicates, the sender may need perform multiplication on two messages encrypted by an additively homomorphic encryption scheme. However, there is no known encryption scheme with both additive and multiplicative homomorphism properties. Fortunately, Boneh et al. [4] introduced an additively homomorphic encryption scheme which can perform multiplication on two ciphertexts one time. In the setting of using threshold

cryptosystem, the sender can even perform multiplication on two ciphertexts arbitrary times via some interactions [9].

In fact, our COC can be designed for any predicate based on the evaluation of bivariable polynomial $f(x, y)$. For example, to compute a public polynomial $f(x, y) = a_2x^2y^2 + a_1x^2y + a_0y$, the receivers send the encryptions of x, x^2 and y, y^2 to the sender respectively. The sender then computes the polynomial by the following steps.

1. Perform the multiplication on the encrypted messages [4] such that $z_2 = x^2y^2$ and $z_1 = x^2y$.
2. Perform the constant multiplication: a_2z_2, a_1z_1 and a_0y.
3. Perform $f(x, y) = a_2z_2 + a_1z_1 + a_0y$.

After computing $f(x, y)$, the sender can embed messages into the result.

Alternatively, we can assume that one receiver holds the polynomial f and the other holds the secret x, and the sender embeds messages into the result of $f(x)$. For example, for the "membership" predicate, one receiver first encodes his set of secrets as a k-degree polynomial such that $f(x) = 0$ iff x belongs to the set, and the other receiver computes $x, x^2, \ldots, x^k$ for his secret x. The sender then sends the message to the receivers such that they get it iff $f(x) = 0$. This "membership" predicate can be used in our oblivious authenticated information retrieval application described in Section 1.

6 Conclusion

We introduce a new notion of *conditional oblivious cast*, which extends conditional oblivious transfer to the three-party case. The definitions of this notion are given. We also provide some implementations for some basic predicates such as "equality", "inequality", and "greater than" predicates. We believe this new notion will be an useful primitive of cryptographic protocols.

References

1. William Aiello, Yuval Ishai, and Omer Reingold. Priced oblivious transfer: How to sell digital goods. In *Proceedings of Advances in Cryptology - EUROCRYPT '01*, volume 2045 of *LNCS*, pages 119–135. Springer-Verlag, 2001.
2. Mihir Bellare and Phillip Rogaway. Optimal asymmetric encryption. In *Proceedings of Advances in Cryptology - EUROCRYPT '94*, volume 950 of *LNCS*, pages 92–111. Springer-Verlag, 1994.
3. Ian F. Blake and Vladimir Kolesnikov. Strong conditional oblivious transfer and computing on intervals. In *Proceedings of Advances in Cryptology - ASIACRYPT '04*, volume 3329 of *LNCS*, pages 515–529. Springer-Verlag, 2004.
4. Dan Boneh, Eu-Jin Goh, and Kobbi Nissim. Evaluating 2-dnf formulas on ciphertexts. In *Proceedings of the 2nd Theory of Cryptography Conference (TCC 2005)*, volume 3378 of *LNCS*, pages 325–341. Springer-Verlag, 2005.
5. Gilles Brassard, Claude Crépeau, and Jean-Marc Robert. All-or-nothing disclosure of secrets. In *Proceedings of Advances in Cryptology - CRYPTO '86*, volume 263 of *LNCS*, pages 234–238. Springer-Verlag, 1986.

6. Jan Camenisch and Markus Stadler. Proof systems for general statements about discrete logarithms. Technical Report 260, Institute for Theoretical Computer Science, ETH Zurich, Mar 1997.
7. David Chaum, Jan-Hendrik Evertse, Jeroen van de Graaf, and Rene Peralta. Demonstrating possession of a discrete logarithm without revealing it. In *Proceedings of Advances in Cryptology - CRYPTO '86*, volume 263 of *LNCS*, pages 200–212. Springer-Verlag, 1986.
8. Cheng-Kang Chu and Wen-Guey Tzeng. Efficient k-out-of-n oblivious transfer schemes with adaptive and non-adaptive queries. In *Proceedings of the Public Key Cryptography (PKC '05)*, volume 3386 of *LNCS*, pages 172–183. Springer-Verlag, 2005.
9. Ronald Cramer, Ivan Damgård, and Jesper Buus Nielsen. Multiparty computation from threshold homomorphic encryption. In *Proceedings of Advances in Cryptology - EUROCRYPT '01*, volume 2045 of *LNCS*, pages 280–299. Springer-Verlag, 2001.
10. Giovanni Di Crescenzo, Rafail Ostrovsky, and Sivaramakrishnan Rajagopalan. Conditional oblivious transfer and timed-release encryption. In *Proceedings of Advances in Cryptology - EUROCRYPT '99*, volume 1592 of *LNCS*, pages 74–89. Springer-Verlag, 1999.
11. Shimon Even, Oded Goldreich, and Abraham Lempel. A randomized protocol for signing contracts. *Communications of the ACM*, 28(6):637–647, 1985.
12. Matthias Fitzi, Juan A. Garay, Ueli Maurer, and Rafail Ostrovsky. Minimal complete primitives for secure multi-party computation. In *Proceedings of Advances in Cryptology - CRYPTO '01*, volume 2139 of *LNCS*, pages 80–100. Springer-Verlag, 2001.
13. Hsiao-Ying Lin and Wen-Guey Tzeng. An efficient solution to the millionaires' problem based on homomorphic encryption. In *Proceedings of Applied Cryptography and Network Security 2005 (ACNS '05)*, volume 3531 of *LNCS*, pages 456–466. Springer-Verlag, 2005.
14. Yi Mu, Junqi Zhang, and Vijay Varadharajan. m out of n oblivious transfer. In *Proceedings of the 7th Australasian Conference on Information Security and Privacy (ACISP '02)*, volume 2384 of *LNCS*, pages 395–405. Springer-Verlag, 2002.
15. Moni Naor and Benny Pinkas. Oblivious transfer and polynomial evaluation. In *Proceedings of the 31st Annual ACM Symposium on the Theory of Computing (STOC '99)*, pages 245–254. ACM, 1999.
16. Moni Naor and Benny Pinkas. Efficient oblivious transfer protocols. In *Proceedings of the 12th Annual Symposium on Discrete Algorithms (SODA '01)*, pages 448–457. ACM/SIAM, 2001.
17. Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Proceedings of Advances in Cryptology - EUROCRYPT '99*, volume 1592 of *LNCS*, pages 223–238. Springer-Verlag, 1999.
18. Michael O. Rabin. How to exchange secrets by oblivious transfer. Technical Report TR-81, Aiken Computation Laboratory, Harvard University, 1981.
19. Claus Peter Schnorr. Efficient signature generation by smart cards. *Journal of Cryptology*, 4(3):161–174, 1991.
20. Yiannis Tsiounis and Moti Yung. On the security of ElGamal based encryption. In *Proceedings of the Public-Key Cryptography (PKC '98)*, volume 1431 of *LNCS*, pages 117–134. Springer-Verlag, 1998.
21. Wen-Guey Tzeng. Efficient 1-out-n oblivious transfer schemes. In *Proceedings of the Public-Key Cryptography (PKC '02)*, pages 159–171. Springer-Verlag, 2002.