強指定驗證者簽章的改善

楊伏夷
朝陽科技大學資訊工程系
yangfy@cyut.edu.tw

梁菱仁
朝陽科技大學資訊工程系
s9827625@cyut.edu.tw

摘要

指定驗證者簽章協定，適用於電子投票與電子投標等應用場合，簽章收受者可以確認文件的內容，但又無法對第三者證明簽署者的身分。本篇文章回顧以前研究學者們提議的協定，並分析這些協定面臨不同攻擊之情況下，攻擊者可以揭露簽署者的身分，因而，提出一個指定驗證者簽章協定，不僅能提供指定驗證者簽章應具備之安全需求，在簽署文件和驗證簽章方面，具有較高的計算效率，簽章的字串編碼也較精簡。

關鍵詞：指定驗證者簽章、可否認性驗證、不可偽造性、正確性、簽章模擬。

Abstract

Designated verifier signature schemes are applied in situations such as electronic voting and electronic auctions. Signature recipients can ascertain the contents of the document, but cannot verify the identity of the signer to a third party. This study reviewed the protocols of past research and analyzed whether these protocols would disclose signer identities under various types of attack. We also proposed a designated verifier signature scheme that meets security requirements and possesses more efficient calculation as well as simpler string encoding for signers.

Keywords: designated verifier signature, deniable authentication, unforgeability, correctness, signature simulation.

1. 前言

數位簽章能提供簽署電子文件(或訊息)的技術，文件收受者藉由數位簽章確認文件簽署者的身分(真實性 Authenticity)、文件簽署者不可否認曾經簽署過文件(不可否認性 Non-repudiation)及文件未曾被竄改(完整性 Integrity)，因此，許多現有的電子商務系統加入數位簽章，希望增加信賴度與避免糾紛，例如：電子財務轉帳、電子合約、商務交易等。

傳統的數位簽章技術[4, 8]，簽署者使用自己的私密金鑰(Private Key)簽署電子文件，產生數位簽章，任何人可使用簽署者的公開金鑰(Public Key)驗證簽章的有效性，也就是說確定簽署者的身分和文件的完整。但是任何人皆可驗證簽章的有效性，有些情況可能不適合，例如：賣方簽署給買主的報價文件，可能變成買主要求其他賣方提供更低報價的憑據；投標者簽署的投標文件，可能包含某些價格或其他敏感資訊，一旦文件被公開之後，這些資訊也跟著洩露；電子投票的應用場合，投票者簽署選票，一旦選票被公開驗證之後，個人的政黨喜好與政治傾向也無法保密。

為了解決上述問題，密碼學者提出了指定驗證者簽章技術(Designated Verifier Signatures)[5]，指定驗證者簽章是使用指定者的公開金鑰及簽署者自己的私密金鑰來簽署文件，因此只有指定驗證者才可驗證簽章的有效性。與傳統簽章不同的是，指定驗證者簽章無法達到不可否認性，但能提供可否認性驗證(Deniable Authentication)，可否認性驗證是由 Deng 等人在 2001 年所提出的技術[2]，具有以下兩個特性：1. 只有指定驗證者能夠驗證簽章的真實來源；2. 指定驗證者無法向第三者證明簽章的真實來源。指定驗證者簽章技術，對於同一電子文件，簽署者和指定驗證者皆可以產生無法區別的數位簽章，因此能達到以上兩個特性。

指定驗證者簽章通常應用在電子拍賣或者電子投票[7]的場合，以電子拍賣的例子來說：投標者(傳送者)簽署標單文件，拍賣商(指定接收者)接收標單文件，投標者一旦將選定價錢的標單投出，拍賣商只能驗證標單的真實來源及完整性，卻無法在拍賣過程中，向其他投標者證明標單的真實來源，因此可以防止拍賣商藉此炒作商品價格。在 1996 年，Jakobsson 等人首先提出指定驗證者簽章技術及應用[5]，但是，倘若攻擊者在指定接收者尚未收到簽章之前攔截簽章，攻擊者便可確定簽署者的身分，因為攻擊者使用簽署者和指定接收者的公開金鑰便可驗證簽章。在 2004 年，Saeednia 等人提出強指定驗證者簽章技術(Strong Designated

Verifier Signatures)[9]來改善上述弱點，該技術結合Schnorr簽章協定[10]及Zheng的簽章加密協定(Signcryption Scheme)[12]，必需使用指定驗證者的私密金鑰才可驗證簽章，因此可防止攻擊者攔截到訊息，進而從中察覺出簽署者的身分。但是 Lee 和 Chang [6]指出 Saeednia 等人的協定仍然存在著安全性上的漏洞，由於使用簽署者的私密金鑰也可驗證簽章，假使簽署者的私密金鑰不小心洩露，並且被攻擊者取得，那麼攻擊者便可半途攔截並驗證簽章，察覺簽署者的身分。因此，Lee 和 Chang 提出新的強指定驗證者簽章協定[6]，他們的協定是建立在Schnorr 簽章協定及Wang 等人的驗證加密協定(Authenticated Encryption Scheme) [11]上，能抵擋上面所述的攻擊，換言之，攻擊者縱然得到簽署者的私密金鑰，驗證簽章仍然必須面臨解離散對數問題(Discrete Logarithm Problem)的難題。

在本篇文章，我們首先回顧 Lee 和 Chang 的強指定驗證者簽章協定，指出 Lee 和 Chang 的協定仍然存在著安全性上的漏洞，若指定驗證者的私密金鑰洩露，攻擊者可半途攔截並驗證簽章，察覺簽署者的身分。在 2.4 節說明上述的攻擊似乎無法阻擋，也討論協定[6]對於簽署者造成不公平與困擾；接著我們提出新的指定驗證者簽章協定，來改善協定[9]的計算與通訊效率，並提供可否認性驗證以及強指定驗證者簽章所需俱備之安全需求。

2. 回顧 Lee 和 Chang 的協定

Lee 和 Chang 的強指定驗證者簽章協定如圖 1，本章節先說明系統參數及符號，接著敘述簽章的產生，驗證簽章，指定驗證者模擬產生簽章，以及分析可否認性驗證無法達成的原因。

系統參數：

p,q：系統產生兩個大質數，滿足 $q \mid p-1$

k,k'：由 Alice 及 Bob 所選擇的隨機亂數

x_A, y_A：Alice 的私密金鑰及公開金鑰，並且 $y_A = g^{x_A} \bmod p$

x_B, y_B：Bob 的私密金鑰及公開金鑰，並且 $y_B = g^{x_B} \bmod p$

$H(\cdot)$：單向雜湊函數，$H(\cdot):\{0,1\}^* \rightarrow \{0,1\}^l$，$l$ 是安全等級參數，例如 $l = 160$

m：一個文件(訊息)

2.1 簽章產生階段

文件簽署者 Alice 選擇隨機亂數 $k \in Z_q^*$，並且計算 r,s,t，計算式如下：

$$r = g^k \bmod p$$

$$s = k + x_A \cdot r \bmod q$$

$$t = H(m, y_B^s \bmod p)$$

指定驗證者簽章為 $\sigma = (r,t)$，完成文件簽署之後，Alice 將文件 m 及簽章 σ 傳送給指定驗證者 Bob。

2.2 簽章驗證階段

Bob 收到文件及簽章後，便可透過以下的驗證式確認簽署者的身分及簽章是否正確，驗證式如下：

$$t \stackrel{?}{=} H(m, (r \cdot y_A^r)^{x_B} \bmod p)$$

設若正確，Bob 相信此文件是由 Alice 所簽署之有效文件。

2.3 簽章模擬階段

在此階段，Bob 可以透過以下的計算式，模擬出簽章 σ'，並且此簽章也能通過上述的驗證式；首先 Bob 先選擇隨機亂數 $k' \in Z_q^*$，並且計算 r', t'，計算方式如下：

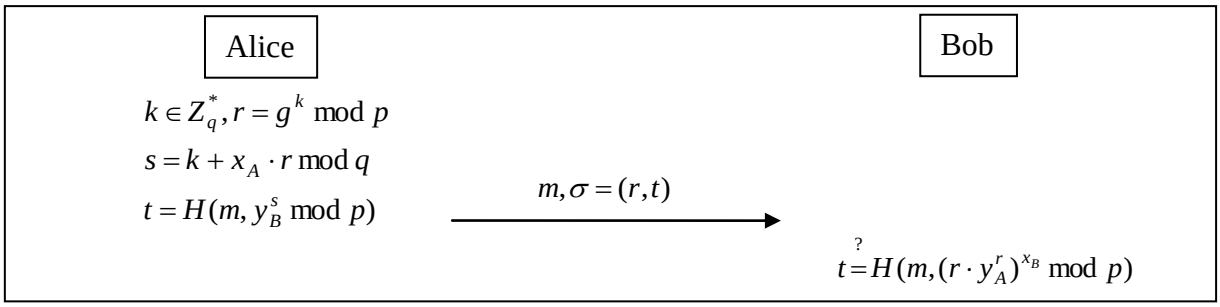

圖 1. Lee 和 Chang 的強指定驗證者簽章協定

$r' = g^{k'} \bmod p$

$t' = H(m, (r' \cdot y_A^{r'})^{x_B} \bmod p)$

透過上面計算式，Bob 模擬出的簽章為 $\sigma' = (r', t')$，此簽章同樣能驗證成功。

2.4 Lee 和 Chang 的協定無法達到可否認性驗證

我們發現 Lee 和 Chang 的協定，在正常的情況下，一般人是無法從簽章中確認簽署者的身分，因為由 Alice 所簽署的 σ 和 Bob 所模擬產出的簽章 σ'，兩者皆隨機分佈在相同的空間 $(Z_p^* \times \{0,1\}^l)$，因此無法分辨。但是，假如指定驗證者 Bob 的私密金鑰不小心洩露，並且被攻擊者取得，那麼攻擊者也可以半途攔截得到簽章，透過驗證簽章，察覺簽署者的身分，就如同 Lee 和 Chang [6]對於 Saeednia 等人協定[9]的攻擊一樣，只是將攻擊情境由洩露簽署者的私密金鑰換成洩露指定驗證者的私密金鑰。綜合上述情況，攻擊者得到簽署者或指定驗證者的私密金鑰，對於協定[9]的可否認性驗證是個威脅；得到指定驗證者的私密金鑰，對於協定[6]的可否認性驗證也構成威脅，似乎暫時無解決之道。

Lee 和 Chang [6]的協定限制指定驗證者才能驗證簽章，導致文件簽署者無法確認簽章的有效性，似乎不甚合理，例如，簽署者傳遞簽章文件(m, r, t)，因為傳輸雜訊或受到攻擊等因素，指定驗證者卻收到(m', r, t)，假設文件 m' 對指定驗證者有利，可能未經驗證就直接宣稱驗證成功，將造成文件簽署者的困擾。

揭露簽署者或指定驗證者的私密金鑰，導致攻擊者得以確定文件簽署者的身分，似乎有些匪夷所思，但其概念相當於金鑰交換技術(Key Agreement)所要求的完全順向安全特性[1](Full Perfect Forward Secrecy)，也就是說交換會議金鑰(Session Key)的雙方，傳送者和接收者，使用會議金鑰來加密傳遞的訊息，若雙方的私密金鑰洩露，利用會議金鑰加密的密文，仍然不至於被解密。這個概念在指定驗證者簽章的意義是類似的，縱然簽署者或指定驗證者的私密金鑰暴露，仍然保有簽署者身分的模糊性質。

3. 我們的協定

經由上面的敘述與分析，協定[6]對於簽署者並不公平，因此本章節提出一個新的方法，在縮短簽章長度與降低計算需求量的條件之下，達成協定[9]的功能，也就是說，提供可否認性驗證，和達到指定驗證者簽章應俱備的安全需求。

3.1 簽章產生階段

首先，簽署者 Alice 計算簽署者與指定驗證者的 Diffie-Hellman 金鑰[3]，$y_{AB} = (y_B)^{x_A} \bmod p$；接著選擇隨機亂數 $k \in Z_q^*$，並且計算 e，指定驗證者簽章為 $\sigma = (e, k)$，計算式如下：

$e = H(m, y_B^{k \cdot x_A} \bmod p)$

Alice 再將文件 m 及簽章 σ 傳送給 Bob。

3.2 簽章驗證階段

Bob 收到文件及簽章後，便透過以下的驗證式確認簽章是否為 Alice 簽署，驗證式如下：

$e \stackrel{?}{=} H(m, y_A^{k \cdot x_B} \bmod p)$

設若正確，Bob 相信此文件是由 Alice 所簽署。

3.3 簽章模擬階段

在此階段，Bob 先選擇隨機亂數 $k' \in Z_q^*$，並且計算 e'，模擬出簽章 σ'，計算方式如下：

$e' = H(m, y_A^{k' \cdot x_B} \bmod p)$

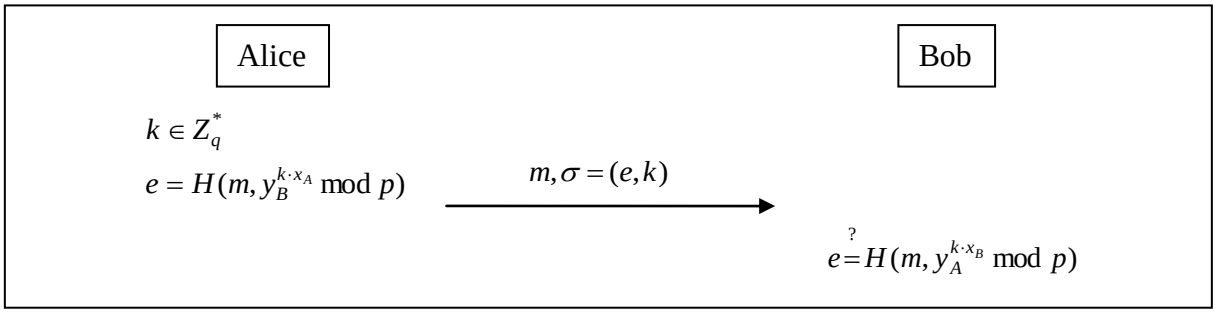

圖 2. 提議的指定驗證者簽章協定

Bob 模擬出的簽章為$\sigma' = (e', k')$，此簽章能通過上述的驗證式，除了 Alice 跟 Bob 之外，別人無法驗證此簽章的真實性，提議的指定驗證者簽章協定如圖 2。

Lee 和 Chang[6]的協定(圖 1)，簽署文件和驗證簽章皆需要兩個指數運算；Saeednia[9]的協定，簽署文件需要一個指數運算，驗證簽章需要兩個指數運算；我們的協定(圖 2)，簽署文件和驗證簽章皆只需要一個指數運算。假設單向雜湊函數 $H(\cdot)$輸出 160 位元的雜湊字串，模 p 的位元長度 1024 位元，模 q 的位元長度 160 位元，Lee 和 Chang[6]的協定，簽章σ的位元長度為 1184 位元；Saeednia[9]的協定，簽章σ的位元長度為 480 位元；我們的協定，簽章σ的位元長度僅為 320 位元。由此可知，提議的指定驗證者簽章協定在簽署文件和驗證簽章的計算需求量，都優於協定[6]；簽章σ的位元長度，則是最精簡，表格 1 摘要以上數據。

4. 安全分析

在本章節，我們將討論提議的指定驗證者簽章協定能正確的驗證簽章(正確性 Correctness)；除了簽署者與指定驗證者之外，其他的第三者無法在多項式時間內產生簽章(不可偽造性 Unforgeability)；第三者無法確定簽署者的身分(簽署者身分不明確性 Signer Ambiguity)；除了指定驗證者之外，其他的第三者無法驗證簽章(強指定驗證者簽章 Strong Designated Verifier Signatures)；簽署者或指定驗證者的私密金鑰被揭露，仍然無法確定簽署者的身分(可抵擋私密金鑰洩露攻擊 Resistant to Key Compromise Attack)。

4.1 正確性

由 Alice 所簽署的簽章文件為(m, σ)，Bob 可以透過驗證式 $e \stackrel{?}{=} H(m, (y_A)^{k \cdot x_B} \bmod p)$ 驗證簽章的正確性。

4.2 不可偽造性

攻擊者想要偽造指定驗證者簽章(m, σ)，首先必須要計算出 Diffie-Hellman 金鑰 y_{AB}，但是攻擊者在不知道 x_A 和 x_B 情況下是無法計算出 y_{AB}，也無法計算$(y_{AB})^k$來偽造簽章。若攻擊者想從先前竊取到的簽章，使用離線猜測攻擊，企圖猜測出 y_{AB}^k，進而推算出 Alice 及 Bob 的 Diffie-Hellman 金鑰 y_{AB}，他必須先破解單向雜湊函數 $H(\cdot)$，得到數據 $Q = y_{AB}^k$，才能計算出 $y_{AB} = (Q)^{1/k} \bmod p$。

4.3 簽署者身分不明確

從 Alice 傳送給 Bob 的簽章中隨機選擇 $(\overline{m}, \overline{\sigma})$，機率 $\Pr[(m,\sigma) = (\overline{m}, \overline{\sigma})]$ 是 $1/(q-1)$，因為 $\overline{\sigma} = (e, k)$ 中的 k 是由 Z_q^* 所選擇出來的；同樣的，由 Bob 所模擬出的簽章中隨機選擇 $(\overline{m}, \overline{\sigma})$，$\Pr[(m', \sigma') = (\overline{m}, \overline{\sigma})]$ 也是 $1/(q-1)$，因此，任何人是很難從 Bob 模擬出的簽章與 Alice 的簽章區別出真實的簽署者。

4.4 抵擋私密金鑰洩露攻擊

在我們所提出的協定中，倘若簽署者或指定驗證者的私密金鑰不小心洩露，攻擊者在取得私密金鑰的情況下，可以將公開的簽章進行驗證，驗證式如下：

$$e \stackrel{?}{=} H(m, y_A^{k \cdot x_B} \bmod p)$$

$$e \stackrel{?}{=} H(m, y_B^{k \cdot x_A} \bmod p)$$

然而，透過上述驗證式，攻擊者還是無法從驗證式中確認簽章者的身分，因為使用簽署者或指定驗證者的私密金鑰來驗證簽章，它們驗證的方式是相同的，而且 Alice 或 Bob 皆可產生 Diffie-Hellman 金鑰 y_{AB}，故攻擊者無法確定簽章者的身分。

5. 結論

在本研究中，我們發現協定[6, 9]在一般的情況下，可以提供可否認性驗證的特性，但若攻擊者取得指定驗證者的私密金鑰，並且攔截與

表 1. 計算需求量與簽章長度

	Lee 和 Chang[6]	Saeednia[9]	提議的協定
簽署文件(指數運算次數)	2	1	1
驗證簽章(指數運算次數)	2	2	1
簽章的位元長度(位元)	1184	480	320

驗證簽章，對於協定[6, 9]的可否認性驗證都構成威脅，也似乎無解決之道；協定[6]限制指定驗證者才能驗證簽章，對文件簽署者形成不公平，因此，本研究不採納協定[6]的方案，而是在協定[9]的基礎上，提出一個新的指定驗證者簽章協定，簽章長度節省 33%，驗證簽章計算需求量節省 50%，也達到指定驗證者簽章所需之安全需求。

參考文獻

[1] Blake-Wilson, S. and Menezes, A., "Authenticated Diffie-Hellman key agreement protocols," ***The 5th Annual Workshop on Selected Areas in Cryptography*** (SAC 98), LNCS 1556, pp. 339-361, 1999.

[2] Deng, X., Lee, C. H. and Zhu, H., "Deniable Authentication Protocols," ***IEE Proceedings of Computers and Digital Techniques***, Vol. 148, pp. 101-104, March, 2001.

[3] Diffie, W. and Hellman, M., "New directions in cryptography," ***IEEE Transactions on Information Theory***, Vol. 22, Issue 6, pp.644-654, 1976.

[4] ElGamal, T., "A public key cryptosystem and a signature scheme based on discrete logarithms," ***IEEE Trans. Inform. Theory***, IT-31, (4), pp. 469-472, 1985.

[5] Jakobsson, M., Sako, K. and Impagliazzo, R., "Designated verifier proofs and their applications," ***Advances in Cryptology—EUROCRYPT '96***, Vol. 1070, pp. 143-154, Springer-Verlag, 1996.

[6] Lee, J. S. and Chang, J. H., "Comment on Saeednia et al.'s strong designated verifier signature scheme," ***Computer Standards & Interfaces***, Vol. 31, Issue 1, pp. 258-260, January 2009.

[7] Li, C. T., Hwang, M. S. and Liu, C. Y., "An electronic voting protocol with deniable authentication for mobile ad hoc networks," ***Computer Communications***, Vol. 31, Issue 10, pp. 2534-2540, June 2008.

[8] Rivest, R., Shamir, A. and Adleman, L., "A method for obtaining digital signatures and public key cryptosystems," ***Communications of the ACM***, Vol. 21, No. 2, pp. 120-126, 1978.

[9] Saeednia, S., Kremer, S. and Markowitch, O., "An efficient strong designated verifier signature scheme," ***Information Security and Cryptology***, Vol. 2971, pp. 40–54, Springer-Verlag, 2004.

[10] Schnorr, C. P., "Efficient signature generation for smart cards," ***Journal of Cryptology 3***, pp. 161-174, 1991.

[11] Wang, G., Bao, F., Ma, C. and Chen, K., "Efficient authenticated encryption schemes with public verifiability," ***IEEE Computer Society***, Vol. 5, pp. 3258-3261, 2004.

[12] Zheng, Y., "Digital signcryption or how to achieve cost(signature & encryption) bb cost(signature)+cost(encryption)," ***Advances in Cryptology — Crypto '97***, Lecture Notes in Computer Science, Vol. 1294, pp. 165–179, Springer-Verlag, 1997.