Security analysis on the security enhancement for anonymous secure e-voting over computer network

Fuw-Yi Yang Hung-Yumr Chen Cai-Ming Liao
Department of Computer Science and Information Engineering,
Chaoyang University of Technology
yangfy@cyut.edu.tw s9627617@cyut.edu.tw s9527645@cyut.edu.tw

摘要

電子投票系統能節省大量人力和時間，只需透過網路即可完成投票與計票；在 2005 年，有學者提出一個安全的電子投票系統，利用不同基底產生混淆位元附加在選票資訊上，因此驗證伺服器無法藉由已知資訊，回推得知投票者，進而改善投票者洩漏身份的問題；在本論文中，我們分析 Hwang et al. 電子投票協定，驗證伺服器依然可以回推投票者身份，導致投票者身份洩漏。

關鍵詞：電子投票系統、盲簽章、匿名。

Abstract

An electronic voting system has received a great deal of attention in recent decades. People can use digital devices such as PC, cell phone, notebook to perform electronic voting over computer network. In 2005, some cryptography researchers proposed an e-voting scheme. This scheme uses different primitive elements to generate confusion elements. Therefore, the authentication server can't identify any voter of public tickets. This paper demonstrates that the Hwang et al. scheme still allows the authentication server to identify the voters so that voters will lose their anonymity. Thus, the Hwang et al. scheme can't provide a secure voting environment for voters.

Keywords：Electronic voting system, Blind signature, Anonymity

1. 前言

近年來，伴隨著電腦的普及和網路的發達，許多使用者使用網路進行交易或請求服務，電子投票即是一個很好的例子。傳統投開票方法，需耗費大量時間和人力，投票者必須到特定地點投票，計票也需耗費大量人力和時間，才能得知投票結果；運用電子投票系統，投票者只需利用網路完成取票和投票，計票時使用計票伺服器即可快速計算出投票結果，因此使用電子投票系統，可節省相當多的時間和人力資源。

在 1988 年，Mu et al. 提出一個電子投票協定[1]，此系統可計算龐大數量的選票，然而協定執行時，伺服器無法防止重複投票的情況發生。在 2003 年，Lin et al. 提出一個改善的協定[2]，伺服器將每位領票人的身份儲存於資料庫中，解決重複投票的問題，但這意味著驗證伺服器可由資料庫內的身份資訊，找出投票者的身份；在 2005 年，Hwang et al. 提出一個改善協定[3]，使用不同的基底數計算混淆位元，使得驗證伺服器無法經由存在資料庫內的資訊，回推找出原本選票的投票者。在本篇論文中，我們將分析 Hwang et al. 提出的協定，仍然存在投票者身份洩漏的問題。

一個完善的投票系統需達到以下幾點特性:

(1)匿名性:

使用者投出的選票，任何人(包括提供投票服務的伺服器)皆無法經由投出的選票，回推出每位投票者身份。

(2)不可偽造性：

每一張選票由驗證伺服器所簽署，只有擁有伺服器的私鑰，才能產生合法選票。

(3)預防重複投票：

伺服器會記錄每次申請選票的投票人身份和相關驗證資訊，爾後伺服器只需由資料庫內的資料，即可確認選票是否已使用。

2.回顧 Hwang et al. 電子投票協定

Hwang et al. 提出的電子投票協定，協定中的參與者包含：投票者（V）、驗證伺服器（AS）、投票伺服器（VS）、計票伺服器（TCS），和憑證驗證中心（CA）。

協定參數定義如下：

- p：由驗證伺服器產生的大質數，並做為公開的參數使用。
- n_{AS}：由驗證伺服器產生的系統參數，n_{AS} 就是往後驗證伺服器計算的共同模數。
- n_v：由使用者產生的系統參數，n_v 為使用者往後計算的共同模數，且須符合（$p< n_{AS} < n_v$）。
- g：g 是 Z_p^* 的元素，且為有限群 $GF(p)$ 的原根。
- h：h 是 Z_p^* 的元素，且為有限群 $GF(p)$ 的原根($h \neq g$)。
- e_x, d_x：使用者 x 利用 RSA 加密演算法分別產生的公開金鑰及私密金鑰。
- $Cert_x$：由憑證驗證中心(CA)所簽發的憑證，憑證內含有使用者 x 的公開金鑰資訊。
- t：時間戳章。
- ||：位元串接符號。

Hwang et al. 的系統運作共分為三個階段：1.投票者申請選票階段。2.投票階段及投票伺服器收集選票階段。3.選票計票階段。

投票者申請選票階段

步驟 2-1-1

申請選票時，選票申請人選取盲因子(b_1, b_2)及亂數(k_1,r)，利用伺服器的公開金鑰計算(w_1, w_1', w_2, w_2')，計算方法如下：

$$w_1 = g^r b_1^{e_{AS}} \bmod n_{AS}$$
$$w_1' = h^r b_1^{e_{AS}} \bmod n_{AS}$$
$$w_2 = g^{k_1} b_2^{e_{AS}} \bmod n_{AS}$$
$$w_2' = h^{k_1} b_2^{e_{AS}} \bmod n_{AS}$$

接著，使用者傳送申請資訊{V, AS, $Cert_V$, t, w_1, w_1', w_2, w_2', $(w_1 \| w_1' \| w_2 \| w_2' \| t)^{d_v} \bmod n_V$ }給驗證伺服器。

步驟 2-1-2

驗證伺服器收到使用者申請資訊後，驗證時間戳章是否在允許傳輸延遲範圍內、和憑證 $Cert_V$ 的真實性，驗證簽章資訊 $(w_1 \| w_1' \| w_2 \| w_2' \| t)^{d_v} \bmod n_V$ 是否正確。假如使用者的申請資訊為合法資訊，驗證伺服器選取一個亂數 k_2，與時戳進行加密計算 w_3。驗證伺服器利用私鑰對使用者的盲訊息(w_1, w_1', w_2, w_2')進行簽章，計算方法如下：

$$w_3 = (k_2 \| t)^{e_v} \bmod n_{AS}$$
$$w_4 = (w_1 \times AS)^{d_{AS}} \bmod n_{AS}$$
$$= (a_1 \times AS)^{d_{AS}} \times b_1 \bmod n_{AS}$$
$$w_5 = (w_1' \times AS)^{d_{AS}} \bmod n_{AS}$$
$$= (a_2 \times AS)^{d_{AS}} \times b_1 \bmod n_{AS}$$
$$w_6 = (w_2 \times g^{k_2} \times AS)^{d_{AS}} \bmod n_{AS}$$
$$= (y_1 \times AS)^{d_{AS}} \times b_2 \bmod n_{AS}$$

$$w_7 = (w_2'^2 \times h^{k_2} \times AS)^{d_{AS}} \bmod n_{AS}$$
$$= (y_2 \times AS)^{d_{AS}} \times b_2^2 \bmod n_{AS}$$

其中(
$a_1 = g^r \bmod p, a_2 = h^r \bmod p,$
$y_1 = g^{k_1+k_2} \bmod p, y_2 = h^{2k_1+k_2} \bmod p$
)，之後驗證伺服器會回傳訊息{V, AS, w_3, $(w_4 \| w_5 \| w_6 \| w_7' \| t)^{d_V} \bmod n_V$ }給使用者，同時，驗證伺服器儲存亂數 k_2 和申請投票者的身份資訊與資料庫內，以供爾後伺服器查證申請人是否有重複申請選票的情形。

步驟 2-1-3

申請人收到驗證伺服器傳送的的簽章訊息封包後，解密訊息 w_3，取得亂數 k_2，產生私密金鑰 $x_1 = k_1 + k_2 \bmod(p-1)$ 和 $x_2 = 2k_1 + k_2 \bmod(p-1)$，並計算相對的公開金鑰 $y_1 = g^{k_1+k_2} \bmod p$ 和 $y_2 = g^{2k_1+k_2} \bmod p$，接著，利用公鑰(y_1, y_2)進行去盲化，計算簽章($s_1 \sim s_4$)，計算方法如下：

$$s_1 = w_4 \times b_1^{-1} = (a_1 \times AS)^{d_{AS}} \bmod n_{AS}$$
$$s_2 = w_5 \times b_1^{-1} = (a_2 \times AS)^{d_{AS}} \bmod n_{AS}$$
$$s_3 = w_6 \times b_2^{-1} = (y_1 \times AS)^{d_{AS}} \bmod n_{AS}$$
$$s_4 = w_7 \times b_2^{-2} = (y_2 \times AS)^{d_{AS}} \bmod n_{AS}$$

步驟 2-1-4

申請者對投票資訊 m 進行簽章，產生兩組簽章訊息 (a_1, s_5) 和 (a_2, s_6)，而簽章產生方法如下：

$$s_5 = x_1^{-1}(ma_1 - r) \bmod p-1$$
$$s_6 = x_2^{-1}(ma_2 - r) \bmod p-1$$

此時，申請者取得合法選票 $T = \{s_1 \| s_2 \| s_3 \| s_4 \| s_5 \| s_6 \| a_1 \| a_2 \| y_1 \| y_2 \| m\}$，並進行之後的投票動作。

投票者投票及選票收集階段

步驟 2-2-1

投票者傳送附加結果 m 的選票給投票伺服器(VS)，投票伺服器驗證資訊 (a_1, a_2, y_1, y_2) 是否合法，驗證方法如下：

$$AS \times a_1 = s_1^{e_{AS}} \bmod n_{AS}$$
$$AS \times a_2 = s_2^{e_{AS}} \bmod n_{AS}$$
$$AS \times y_1 = s_3^{e_{AS}} \bmod n_{AS}$$
$$AS \times y_2 = s_4^{e_{AS}} \bmod n_{AS}$$

步驟 2-2-2

投票伺服器驗證簽章對 (a_1, s_5) 和 (a_2, s_6)，確認投票資訊 m 是否正確，驗證計算方法如下：

$$g^{ma_1} = y_1^{s_5} \times a_1 \bmod p$$
$$h^{ma_2} = y_2^{s_6} \times a_2 \bmod p$$

若為有效選票，投票伺服器將選票 T 存入資料庫中。之後等到此回合投票時間結束後，投票伺服器將把所有收集到的選票傳給計票伺服器（TCS），進行投票結果計算。

選票計算階段

步驟 2-3-1

計票伺服器（TCS）確認每張選票的附加資訊 (y_1, y_2, a_1, a_2) 是否重複使用，並計算隨機亂數 k_2，來查核是否有人使用相同的資訊申請投票，計算方法如下：

$$x_1 = \frac{m'a_1 - ma_1}{s_5' - s_5} \bmod p-1$$
$$x_2 = \frac{m'a_2 - ma_2}{s_6' - s_6} \bmod p-1$$
$$k_1 = x_2 - x_1 = (2k_1 + k_2) - (k_1 + k_2)$$
$$k_2 = x_1 - k_1$$

3.分析 Hwang et al. 電子投票協定

Hwang et al. 提出的協定，依然存在投票者身份洩漏的問題；投票伺服器公開全部選票後，驗證伺服器能夠從每一張選票的資訊回推並從資料庫中找出每位投票者身份。

投票伺服器公開全部選票後，驗證伺服器得知每張選票內的簽章 $s_3 = w_6 \times b_2^{-1} \bmod n_{AS} = (y_1 \times AS)^{d_{AS}} \bmod n_{AS}$，即可由下列計算跟伺服器內的儲存資料比對，標示出每位投票者的身份，步驟如下：

步驟 3-1-1

驗證伺服器擁有每位申請者資料 $\{v, w_1, w_1', w_2, w_2', w_3, w_4, w_5, w_6, k_2\}$。

步驟 3-1-2

投票伺服器公告選票資訊 $\{s_1 \| s_2 \| s_3 \| s_4 \| s_5 \| s_6 \| a_1 \| a_2 \| y_1 \| y_2 \| m\}$。

步驟 3-1-3

驗證伺服器由上述資料計算得知 b_2：

$$b_2 = \frac{w_6}{s_3} = \frac{(y_1 \times AS)^{d_{AS}} \times b_2}{(y_1 \times AS)^{d_{AS}}} \bmod n_{AS}$$。

步驟 3-1-4

得知 b_2 後，驗證伺服器計算 g^{k_1} 和 h^{k_1}：

$$g^{k_1} = \frac{w_2}{b_2^{e_{AS}}} \bmod n_{AS} \quad , \quad h^{k_1} = \frac{w_2'}{b_2^{e_{AS}}} \bmod n_{AS}$$

步驟 3-1-5

得知 g^{k_1} 和 h^{k_1} 後，選擇資料庫內任一組申請選票人資料 (V', k_2')，確認 $y_1 \stackrel{?}{=} g^{k_1} \times g^{k_2'} \bmod p, y_2 \stackrel{?}{=} h^{2k_1} \times h^{k_2'} \bmod p$，重複步驟 3-1-5，驗證伺服器即可找出選票由哪位投票者投出。

經由上述步驟可證明，驗證伺服器儲存每位申請人的身份資訊，經由計算可回推出選票由哪位投票者投出，因此 Hwang et al. 提出的電子投票架構，依然無法提供投票者一個安全匿名的投票環境。

此外投票者投出的選票 $T = \{s_1 \| s_2 \| s_3 \| s_4 \| s_5 \| s_6 \| a_1 \| a_2 \| y_1 \| y_2 \| m\}$，擁有選舉結果 m，但卻以明文傳送投票結果 m，傳輸過程中並無任何密碼學機制保護，因此只要惡意攻擊者攔截每張傳送到投票伺服器的選票，即可輕易控制整體投票結果。

4. 結論

本篇文章指出 Hwang et al. 提出的電子投票協定，存在兩個問題：問題一、投票申請者身份洩漏，協定無法提供匿名投票的特性；問題二、選票內容由明文傳送，惡意攻擊者能輕易攔截選票得知選票內容，進而影響整體投票結果。因此 Hwang et al. 提出的電子投票協定依然無法提供給投票者一個安全且匿名的投票環境。

參考文獻

[1] Y. Mu, V. Varadharajan, Anonymous secure e-voting over a new work, ***Proceedings of the 14th Annual Computer Security Application Conference***, pp. 293- 299, 1998.

[2] I.C. Lin, M.S. Hwang, C.C. Chang, Security enhancement for anonymous secure e-voting over a network, ***Computer Standards and Interfaces***, pp. 131- 139, 2003.

[3] S.-Y. Hwang, H.-A. Wen, Tzonelih Hwang, On the security enhancement for anonymous secure e-voting over computer network, ***Computer Standards & Interfaces***, Vol. 27, pp. 163-168, January 2005.

[4] H.-Y. Chien, J.K. Jan, Y.M. Tseng, Cryptanalysis on Mu–Varadharajan's e-voting schemes, ***Applied Mathematics and Computation***, pp. 525-530, July 2003.

[5] J. Camenisch, J. Piveteau, M. Stadler, Blind signatures based on discrete logarithm problem, ***Advances in Cryptology***, ***EUROCRYPT'94***, Lecture Notes in Computer Science 950 (1994) , pp. 428- 432.

[6] R. Cramer, R. Gennaro, J. Borrell, A secure and optimally efficient multi-authority election scheme, ***Advances in Cryptology***, ***EUROCRYPT'97***, Lecture Notes in Computer Science 1233 (1997), pp. 103–117.