An International Journal
computers & mathematics with applications

PERGAMON Computers and Mathematics with Applications 46 (2003) 207–211
www.elsevier.com/locate/camwa

Improved Authenticated Multiple-Key Agreement Protocol

HER-TYAN YEH, HUNG-MIN SUN AND TZONELIH HWANG
Department of Computer Science and Information Engineering
National Cheng Kung University
Tainan, Taiwan 701, R.O.C.
hmsun@mail.ncku.edu.tw

(Received October 2000; revised and accepted October 2002)

Abstract—Recently, Yen and Joye showed that Harn and Lin's authenticated multiple-key agreement protocol is insecure against forgery and consequently proposed a revised protocol to repair it. Later, Wu *et al.* showed that Yen and Joye's revision is also insecure and therefore an improved protocol was proposed. However, Wu *et al.*'s protocol violates the original requirement in which no one-way hash function is needed. On the other hand, in order to overcome Yen and Joye's and Wu *et al.*'s attacks, Harn and Lin proposed a modified version by modifying the signature signing equation. But the modified version increases one exponentiation in the verification equation. In this paper, we first show that Wu *et al.*'s protocol still suffers the forgery problem, and then we propose an improved scheme that is secure against forgery and does not involve any one-way hash function. Compared with Harn and Lin's modified version, our scheme is efficient in the verification equation. © 2003 Elsevier Science Ltd. All rights reserved.

Keywords—Authenticated, Multiple-key, Key agreement.

1. INTRODUCTION

In 1998, Harn and Lin [1] proposed an authenticated multiple-key agreement protocol that enabled two parties to authenticate each other, and share multiple secret keys without using any one-way hash function. Recently, Yen and Joye [2] showed that Harn and Lin's authenticated multiple-key agreement protocol is insecure against forgery and consequently proposed a revised protocol to repair it. Later, Wu *et al.* [3] showed that Yen and Joye's protocol is also insecure against forgery, and therefore, an improved protocol was proposed. However, Wu *et al.*'s protocol violates the original requirement in which no one-way hash function is needed. In order to overcome Yen and Joye's and Wu *et al.*'s attacks, Harn and Lin [4] proposed a modified version by modifying the signature signing equation. But the modified version increases one exponentiation in the verification equation. In this paper, we first show that Wu *et al.*'s protocol still suffers the forgery problem, and then we propose an improved scheme that is secure against forgery and does not involve any one-way hash function. Compared with Harn and Lin's modified version, our scheme is efficient in the verification equation.

0898-1221/03/$ - see front matter © 2003 Elsevier Science Ltd. All rights reserved. Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX
PII: S0898-1221(03)00210-4

2. THE FIRST HARN AND LIN PROTOCOL

We briefly review the first Harn and Lin protocol in the following. Note that Harn and Lin focused on how to design a secure authenticated multiple-key agreement protocol without using a one-way hash function.

Let A and B be two users who want to share multiple secret keys. Initially, the system has a large prime P and a primitive element α in $GF(P)$. Users A and B have long-term secret key x_A and x_B and the corresponding public key $y_A = \alpha^{x_A} \bmod P$, $y_B = \alpha^{x_B} \bmod P$, respectively. Here we only describe A's part because B's part is the same as A's part. In the authentication phase, A randomly selects two short-term secret numbers k_{A_1} and k_{A_2} and computes $r_{A_1} = \alpha^{k_{A_1}} \bmod P$ and $r_{A_2} = \alpha^{k_{A_2}} \bmod P$. A then computes the signature as: $s_A = x_A - r_A k_A \bmod (P-1)$, where $k_A = k_{A_1} + k_{A_2} \bmod (P-1)$ and $r_A = \alpha^{r_{A_1} r_{A_2}} \bmod P$. Finally, A sends $\{r_{A_1}, r_{A_2}, s_A, \operatorname{cert}(y_A)\}$ to B, where $\operatorname{cert}(y_A)$ is the certification of A's public key.

After receiving $\{r_{A_1}, r_{A_2}, s_A, \operatorname{cert}(y_A)\}$, B first computes $r_A = \alpha^{r_{A_1} r_{A_2}} \bmod P$ and verifies the authenticity of $\{r_{A_1}, r_{A_2}\}$ via the following equation: $y_A ? = (r_{A_1} \cdot r_{A_2})^{r_A} \cdot \alpha^{s_A} \pmod P$.

If it holds, B processes the key generation phase and derives the common keys as follows:

$$\begin{aligned} k_1 &= r_{A_1}^{k_{B_1}} \bmod P, \\ k_2 &= r_{A_2}^{k_{B_1}} \bmod P, \\ k_3 &= r_{A_1}^{k_{B_2}} \bmod P, \\ k_4 &= r_{A_2}^{k_{B_2}} \bmod P. \end{aligned}$$

Note that only three out of these four keys can be used in order to provide perfect forward secrecy [5].

3. YEN AND JOYE'S PROTOCOL

It can be seen that in Harn and Lin's protocol, given a valid four-tuple $\{r_{A_1}, r_{A_2}, s_A, \operatorname{cert}(y_A)\}$, if an attacker can find integers $r'_{A_1}, r'_{A_2} \in Z_P$ satisfying $r'_{A_1} \cdot r'_{A_2} = r_{A_1} \cdot r_{A_2}$, then he can convince B that he is A. This is because the verification equation $y_A ? = (r'_{A_1} \cdot r'_{A_2})^{r_A} \cdot \alpha^{s_A} \pmod P$ still holds. So, letting q be a small factor of r_{A_1}, the attacker can set $r'_{A_1} = r_{A_1}/q$ and $r'_{A_2} = r_{A_2} \cdot q$. For a small factor q (e.g., 2), there is a nonnegligible probability such that $r'_{A_2} < P$. Thus, the Harn and Lin protocol is insecure against forgery.

In order to overcome this problem, Yen and Joye added a constraint on the computed result of r_{A_1} and r_{A_2}. The short-term secret key k_{A_1} and k_{A_2} should be chosen such that the resulting r_{A_1} and r_{A_2} satisfying $\lceil P/2 \rceil \leqq r_{A_1}, r_{A_2} \leqq P-1$. Furthermore, the signature generation equation is replaced by: $s_A = x_A - (r_{A_1} \cdot r_{A_2}) \cdot k_A \bmod (P-1)$.

Hence, the verification equation becomes: $y_A ? = (r_{A_1} \cdot r_{A_2})^{r_{A_1} r_{A_2}} \cdot \alpha^{s_A} \pmod P$.

With the modification, the attacker cannot generate another pair of $\{r'_{A_1}, r'_{A_2}\}$ by replacing $r'_{A_1} = r_{A_1}/q$ and $r'_{A_2} = r_{A_2} \cdot q$ because the constraint: $\lceil P/2 \rceil \leqq r_{A_1}, r_{A_2} \leqq P-1$ is applied.

4. WU *ET AL.*'S PROTOCOL

Wu *et al.* showed that in Yen and Joye's revision, even if both r_{A_1} and r_{A_2} are located in the range $[\lceil P/2 \rceil, P-1]$, an attacker can still find such r'_{A_1} and r'_{A_2} satisfying $r'_{A_1} \cdot r'_{A_2} = r_{A_1} \cdot r_{A_2}$ with a nonnegligible probability. We refer the reader to [3] for the details.

To eliminate this weakness, Wu *et al.* employed a one-way hash function h in the signature generation and verification equation. They replaced the signature generation by: $s_A = x_A - h(r_{A_1} \cdot r_{A_2}) \cdot k_A \bmod (P-1)$, and hence, the verification equation becomes

$$y_A ? = (r_{A_1} \cdot r_{A_2})^{h(r_{A_1} \cdot r_{A_2})} \cdot \alpha^{s_A} \pmod P.$$

Although this modification tries to provide higher security strength for mutual authentication, it also violates the original requirement that no one-way hash function is needed for designing an authenticated multiple-key agreement protocol as in Harn and Lin's scheme.

5. THE SECOND HARN AND LIN PROTOCOL

In order to overcome Yen and Joye's and Wu *et al.*'s attacks, Harn and Lin proposed a modified version by modifying the signature signing equation. The signature generation equation is replaced by

$$s_A = x_A - r_{A_1}k_{A_1} + r_{A_2} \cdot k_{A_2} \bmod(P-1),$$

and the verification equation becomes

$$y_A? = r_{A_1}^{r_{A_1}} \cdot r_{A_2}^{r_{A_2}} \cdot \alpha^{s_A} \pmod P.$$

Although the modified version successfully overcomes Yen and Joye's and Wu *et al.*'s attacks, comparing with the first Harn and Lin protocol, there is one additional exponentiation in the verification equation.

6. CRYPTANALYSIS OF WU *ET AL.*'S PROTOCOL

Although Wu *et al.*'s modification employed a one-way hash function to provide higher security strength for mutual authentication, their protocol still suffers the forgery problem. The reason is that if $r'_{A_1} \cdot r'_{A_2} = r_{A_1} \cdot r_{A_2}$, then their hashed values: $h(r'_{A_1} \cdot r'_{A_2})$ and $h(r_{A_1} \cdot r_{A_2})$ are equal. If an attacker can find integers $r'_{A_1}, r'_{A_2} \in Z_P$ satisfying $r'_{A_1} \cdot r'_{A_2} = r_{A_1} \cdot r_{A_2}$, then he can convince B that he is A. This is because the verification equation $y_A? = (r_{A_1} \cdot r_{A_2})^{h(r_{A_1} \cdot r_{A_2})} \cdot \alpha^{s_A} \pmod P$ still holds. Thus, the forgery still exists.

7. OUR IMPROVED PROTOCOL

In Wu *et al.*'s paper, although they have proposed a method to enhance the security of their protocol, however, it still suffers the forgery problem and violates the original expectation in Harn and Lin's protocol in which no one-way hash function is needed. In Harn and Lin's modified protocol, although it successfully overcomes Yen and Joye's and Wu *et al.*'s attacks, it increases the computations of exponentiation complexity in the verification equation. In the following, we try to design a secure and efficient authenticated multiple-key agreement protocol without using any one- way hash function. One straightforward modification is to replace r_A as $r_A = r_{A_1} + r_{A_2} \pmod{P-1}$. Thus, the signature generation equation is replaced by: $s_A = x_A - (r_{A_1} + r_{A_2}) \cdot k_A \bmod(P-1)$, and the verification equation becomes: $y_A? = (r_{A_1} \cdot r_{A_2})^{(r_{A_1}+r_{A_2})} \cdot \alpha^{s_A} \pmod P$.

Note that in this case if $\{r_{A_1}, r_{A_2}, s_A, \text{cert}\,(y_A)\}$ satisfies the verification equation, then so does $\{r_{A_2}, r_{A_1}, s_A, \text{cert}\,(y_A)\}$. For avoiding ambiguity, we make the constraint: $0 < r_{A_1} \leqq r_{A_2} < P-1$.

After this modification, if an attacker wants to forge a valid four-tuple $\{r'_{A_1}, r'_{A_2}, s_A, \text{cert}\,(y_A)\}$ from a past four-tuple $\{r_{A_1}, r_{A_2}, s_A, \text{cert}\,(y_A)\}$, he must make $r'_{A_1} \cdot r'_{A_2} = r_{A_1} \cdot r_{A_2} \pmod P$ and $r'_{A_1} + r'_{A_2} = r_{A_1} + r_{A_2} \pmod{P-1}$ hold simultaneously. That is, the attacker needs to solve

$$r'_{A_1} \cdot r'_{A_2} = r_{A_1} \cdot r_{A_2} \pmod P, \tag{1}$$

$$r'_{A_1} + r'_{A_2} = r_{A_1} + r_{A_2} \pmod{P-1}. \tag{2}$$

From equation (2), we have $r'_{A_1} = r_{A_1} + r_{A_2} - r'_{A_2} + k(P-1)$ and substitute it into equation (1). We can obtain: $(r_{A_1} + r_{A_2} - r'_{A_2} + k(P-1)) \cdot r'_{A_2} = r_{A_1} \cdot r_{A_2} \pmod P$.

By selecting a proper k, we can solve r'_{A_2} from the above quadratic equation in a finite field $GF(P)$, and hence, obtain the corresponding r'_{A_1}. We show this as follows.

DEFINITION 1. *(See [6].) If m is a positive integer, we say that the integer a is a quadratic residue of m if $(a, m) = 1$ and the congruence $x^2 = a \pmod{m}$ has a solution. If the congruence $x^2 = a \pmod{m}$ has no solution, we say that a is a quadratic nonresidue of m.*

THEOREM 1. *(See [6].) Let P be an odd prime and a be an integer not divisible by P. Then, the congruence $x^2 \equiv a \pmod{P}$ has either no solutions or exactly two incongruent solutions modulo P.*

PROOF. This is a well-known theorem. For simplicity, we omit the proof and refer the reader to [6]. In addition, in [7] is provided a simple and fast probabilistic algorithm to find adaptive solution of x. ∎

THEOREM 2. *(See [6].) If p is an odd prime, then there are exactly $(p-1)/2$ quadratic residue of p and $(p-1)/2$ quadratic nonresidue of p among the integers $1, 2, \ldots, p-1$.*

PROOF. We omit the proof and refer the reader to [6]. ∎

LEMMA 1. *For an arbitrarily selected a, the probability that the resulting equation has solutions for $x^2 \equiv a (\bmod P)$ is $1/2$.*

LEMMA 2. *Consider the quadratic congruence $ax^2 + bx + c = 0 \pmod{P}$, where P is prime and a, b, and c are integers with $P \nmid a$. Letting $d = b^2 - 4ac$, the congruence $ax^2 + bx + c = 0 \pmod{P}$ is equivalent to the congruence $y^2 \equiv d \pmod{P}$, where $y = 2ax + b$. If $d \equiv 0 \pmod{P}$, then there is exactly one solution x modulo P; if d is a quadratic residue of P, then there are two incongruent solutions, and if d is a quadratic nonresidue of P, then there are no solutions.*

By Lemmas 1 and 2, we can easily find adaptive solution (r'_{A_1}, r'_{A_2}) by selecting a proper k, where $(r_{A_1} + r_{A_2} - r'_{A_2} + k(P-1)) \cdot r'_{A_2} = r_{A_1} \cdot r_{A_2} \pmod{P}$. The solution (r'_{A_1}, r'_{A_2}) will be different from (r_{A_1}, r_{A_2}). Thus, an attacker can forge a valid four-tuple $\{r'_{A_1}, r'_{A_2}, s_A, \operatorname{cert}(y_A)\}$. In the following example, we demonstrate that the forgery is possible.

EXAMPLE. Let $r_{A_1} = 8$, $r_{A_2} = 11$, and $P = 13$. Then an attacker can forge $r'_{A_1} = 2$ and $r'_{A_2} = 5$ such that both $r'_{A_1} \cdot r'_{A_2} = r_{A_1} \cdot r_{A_2} \pmod{P}$ and $r'_{A_1} + r'_{A_2} = r_{A_1} + r_{A_2} \pmod{P-1}$ hold simultaneously. This is because $8 \cdot 11 \pmod{13} = 2 \cdot 5 \pmod{13} = 10$ and $8 + 11 \pmod{12} = 2 + 5 \pmod{12} = 7$.

So we improve it by replacing r_A as $r_A = r_{A_1} \oplus r_{A_2} \pmod{P-1}$, where $\oplus$ denotes the bit-wise addition. Thus, the signature generation equation is replaced by: $s_A = x_A - (r_{A_1} \oplus r_{A_2}) \cdot k \bmod (P-1)$, and the verification equation becomes: $y_A ? = (r_{A_1} \cdot r_{A_2})^{(r_{A_1} \oplus r_{A_2})} \cdot \alpha^{s_A} \pmod{P}$.

Thus, an attacker must solve

$$r'_{A_1} \cdot r'_{A_2} = r_{A_1} \cdot r_{A_2} \pmod{P}, \tag{3}$$

$$r'_{A_1} \oplus r'_{A_2} = r_{A_1} \oplus r_{A_2} \pmod{P-1}. \tag{4}$$

From equation (4), we obtain $r'_{A_1} = r_{A_1} \oplus r_{A_2} \oplus r'_{A_2} + k(P-1)$, and substituting it into equation (3), we can obtain

$$\left(r_{A_1} \oplus r_{A_2} \oplus r'_{A_2} + k(P-1)\right) \cdot r'_{A_2} = r_{A_1} \cdot r_{A_2} \pmod{P}. \tag{5}$$

Here we remind that the distributive law for the operations ($\oplus$ and $\cdot$) does not hold. That is $(a \oplus b) \cdot c \neq (a \cdot c) \oplus (b \cdot c)$, where $\oplus$ denotes the bit-wise addition and $\cdot$ denotes the multiplication operation in the congruence class modulo P. As an example, let $P = 7$, $a = 011_2$, $b = 110_2$, $c = 011_2$. It is clear that $(a \oplus b) \cdot c = (011_2 \oplus 110_2) \cdot 011_2 = 101_2 \cdot 011_2 = 5 \cdot 3 \bmod 7 = 1 = 001_2$. However, $(a \cdot c) \oplus (b \cdot c) = (011_2 \cdot 011_2) \oplus (110_2 \cdot 011_2) = (3 \cdot 3 \bmod 7) \oplus (6 \cdot 3 \bmod 7) = 010_2 \oplus 100_2 = 110_2$. Hence, $(a \oplus b) \cdot c \neq (a \cdot c) \oplus (b \cdot c)$.

This property of the operations suggests that equation (5) is unable to be reduced to a simpler form. The only method to find r'_{A_2} in equation (5) is to do an exhaustive search. When P is small, the attacker is able to find a solution for r'_{A_2}; however, when P is a large prime number, the attacker is unable to find the solution.

Compared with the first Harn and Lin protocol, our scheme does not increase any computations of exponentiation complexity. Comparing with Harn and Lin's modified protocol, although our scheme and Harn and Lin's modified protocol all can counter the forgery attacks showed in Yen and Joye's and Wu *et al.*'s methods, our scheme is more efficient in the verification equation.

8. CONCLUSIONS

In this paper, we have shown that Wu *et al.*'s protocol still suffers the forgery problem. Then, we have proposed an improved authenticated multiple-key agreement protocol that uses a digital signature to authenticate Diffie-Hellman public key. The proposed protocol is secure and efficient against forgery, and does not involve any one-way hash function.

REFERENCES

1. L. Harn and H.Y. Lin, An authenticated key agreement protocol without using one-way function, In *Proc. 8th National Conf. Information Security*, Kaohsiung, Taiwan, pp. 155–160, (1998).
2. S.M. Yen and M. Joye, Improved authenticated multiple-key agreement protocol, *Electron. Lett.*, 1738–1739, (1998).
3. T.S. Wu, W.H. He and C.L. Hsu, Security of authenticated multiple-key agreement protocols, *Electron. Lett.*, 391–392, (1999).
4. L. Harn and H.Y. Lin, Authenticated key agreement protocol without using one-way functions, *Electron. Lett.*, 629–630, (2001).
5. C.H. Lim and P.J. Lee, Security of interactive DSA batch verification, *Electron. Lett.*, 1592–1593, (1994).
6. K.H. Rosen, *Elementary Number Theory and Its Application*, Third Edition, (1992).
7. R.C. Peralta, A simple and fast probabilistic algorithm for computing square roots modulo a prime number, *IEEE Trans. on Information Theory* **32** (6), 846–847, (1986).