Improvement of Signature Scheme Based on Factoring and Discrete Logarithms *

Li-Hua Li[‡] Shiang-Feng Tzeng[‡] Min-Shiang Hwang[†]

Graduate Institute of Networking and Communication Engineering[†]
Chaoyang University of Technology
168 Gifeng E. Rd., Wufeng,
Taichung County, Taiwan 413, R.O.C.
Email: mshwang@cyut.edu.tw
Fax: 886-4-23742337

Department of Information Management[‡]
Chaoyang University of Technology
168 Gifeng E. Rd., Wufeng,
Taichung County, Taiwan 413, R.O.C.

December 13, 2003

*This research was partially supported by the National Science Council, Taiwan, R.O.C., under contract no.: NSC91-2213-E-324-003.

[†]Responsible for correspondence: Prof. Min-Shiang hwang

Improvement of Signature Scheme Based on Factoring and Discrete Logarithms

Abstract

Laih and Kuo proposed two efficient signature schemes based on discrete logarithms and factorization. However, their schemes require many keys for a signing document. In this article, we shall propose an improvement of Laih and Kuo's signature schemes. The improved scheme will outperform their schemes in the number of keys.

Keywords: Cryptography, digital signature, discrete logarithms, factoring.

1 Introduction

On common feature among all digital signature schemes are that the security is based on a single cryptographic assumption, i.e., discrete logarithms (DL) [2, 3, 5, 9] or factoring (FAC) a large composite number problem [1, 6, 8]. Although schemes based on one of the above cryptographic assumptions appear secure today, they may be exploded in the future.

Recently, Laih and Kuo [7] presented two new signature schemes which are based on both DL and FAC. Their schemes are not only secure but also efficient in computational complexity. However, their schemes require many keys for a signing document. To reduce the number of keys, we shall propose an improvement of Laih-Kuo's signature schemes in this article.

2 Review of Laih-Kuo's Schemes

We will review briefly Laih-Kuo's schemes [7] in the following two sub-sections.

2.1 Laih-Kuo's Scheme-1

Each user owns $2t+1$ secret keys and public keys such that t-pair (u_i, k_i) satisfying $k_i u_i^2 \equiv 1 \bmod n$, a pair (u, z) satisfying $ku^2 \equiv -1 \bmod n$ and $z \equiv g^k \bmod p$, and t-pair (s_i, z_i) satisfying $z_i \equiv g^{s_i} \bmod p$ for $i = 1, 2, \cdots, t$.

To create a signature for a message M, the signer executes the following steps:

1. Randomly select two numbers r and R such that $GCD(r, n) = 1$ and calculate $K \equiv g^R \bmod p$, where p is a large prime; g is a primitive element in $GF(p)$; n is a factor of $(p-1)$ that is the product of two large primes.

2. Calculate $h(M, K) = d = (d_1, d_2, \cdots, d_t)$, where $d_i \in \{0, 1\}$ and $h(\cdot)$ denotes a one-way hash function with t-bit length.

3. Compute $M' \equiv \sum_{i=1}^{t} s_i d_i \bmod n$ and $x \equiv \frac{1}{2}(r + \frac{M'+RK^2}{r}) \bmod n$.

4. Calculate $h(x) = e = (e_1, e_2, \cdots, e_t)$, where $e_i \in \{0, 1\}$.

5. Compute $y \equiv \frac{1}{2} u \prod_{i=1}^{t} u_i^{e_i} (r - \frac{M'+RK^2}{r}) \bmod n$.

Then, the signature of M is (K, x, y). After receiving the signature (K, x, y), the verifier calculates $h(x) = e = (e_1, e_2, \cdots, e_t)$ and $h(M, K) = d = (d_1, d_2, \cdots, d_t)$. The verifier validates the signature by checking the following congruent equality: $g^{x^2} z^{y^2 \prod_{i=1}^{t} k_i^{e_i}} \equiv \prod_{i=1}^{t} z_i^{d_i} K^{K^2} \bmod p$.

2.2 Laih-Kuo's Scheme-2

The purpose of this scheme is to reduce the number of keys of the Laih-Kuo's scheme-1. Each user owns $t+1$ secret keys and public keys such that t-pair (u_i, k_i) satisfying $k_i u_i^2 \equiv 1 \bmod n$ and a pair (u, z) satisfying $ku^2 \equiv -1 \bmod n$ and $z \equiv g^k \bmod p$ for $i = 1, 2, \cdots, t$.

To create a signature for a message M, the signer executes the following steps:

1. Select a random number r such that $GCD(r, n) = 1$.

2. Choose two random numbers R_1 and R_2 such that $K_1 = g^{R_1} \bmod p$ and $K_2 = g^{R_2} \bmod p$, and calculate $x \equiv \frac{1}{2}(r + \frac{h^2(M)R_1K_1^2 + R_2K_2^2}{r}) \bmod n$.

3. Calculate $h(x) = e = (e_1, e_2, \cdots, e_t)$, where $e_i \in \{0, 1\}$.

4. Compute $y \equiv \frac{1}{2}u \prod_{i=1}^{t} u_i^{e_i}(r - \frac{h^2(M)R_1K_1^2 + R_2K_2^2}{r}) \bmod n$.

The signer sends the signature (x, y, K_1, K_2) associated with the message M to the verifier. After receiving the signature (x, y, K_1, K_2), the verifier calculates $h(x) = e = (e_1, e_2, \cdots, e_t)$ and checks the validity through the following equation: $g^{x^2} z^{y^2 \prod_{i=1}^{t} k_i^{e_i}} \equiv K_1^{h^2(M)K_1^2} K_2^{K_2^2} \bmod p$.

3 The Improved Signature Scheme

The system parameters of the proposed scheme are the same as those of Laih-Kuo's schemes as described previously. In our scheme, each user has only two pairs of secret key and public key. One of the pairs is $(u,\ z)$ satisfying $ku^2 \equiv -1 \bmod n$ and $z \equiv g^k \bmod p$ and the other pair is $(u_1,\ k_1)$ satisfying $k_1u_1^2 \equiv 1 \bmod n$. The signer executes the following steps:

1. Select a random number r such that $GCD(r, n) = 1$.

2. Randomly select a number R such that $K = g^R \bmod p$ and calculate $x \equiv \frac{1}{2}(r + \frac{RK^2}{r}) \bmod n$.

3. Calculate $y \equiv \frac{1}{2}uu_1^{h(M,x)}(r - \frac{RK^2}{r}) \bmod n$.

Then, (K, x, y) is a signature of M signed by the signer.

After receiving the signature (K, x, y) for M, the verifier verifies the validity of the signature through the following equation:

$$g^{x^2} z^{y^2 k_1^{h(M,x)}} \equiv K^{K^2} \bmod p. \tag{1}$$

If it holds, the message M is authenticated and the signature $(K,\ x,\ y)$ is valid.

Further, like Laih-Kuo's schemes in [7], the improved scheme can also be applied to the identification environment. The major feature of the identification scheme is that the security is based on discrete logarithms and factorization.

4 Security Analysis and Performance Evaluation

The security analysis of our improved scheme is similar to that of Laih-Kuo's schemes. Some possible attacks by which an adversary may try to take down the improved scheme will be analyzed as follows.

Assume an adversary attempts to obtain the secret key from the public key of any user. In this attack, the adversary must solve the DL problem [2, 3, 5, 9] to derive k from public key z. In addition, the adversary needs to solve the FAC a composite number n [4, 6, 8] to recover u from $ku^2 \equiv -1 \bmod n$. Similarly, the adversary must solve the FAC of n to obtain u_1 from $k_1 u_1^2 \equiv 1 \bmod n$.

Assume an adversary attempts to forge a valid signature (K, x, y) for a given message M. She/he knows no secret key and valid signature of the signer. The adversary sets two variables to be fixed integers and finds the solution to the other variable from Equation (1). Then the adversary has to randomly select $(K,\ x)$ or $(x,\ y)$ and find y or K to satisfy Equation (1), which is as difficult as solving DL and FAC simultaneously. In another similar approach, given $(K,\ y)$, finding x to satisfy Equation (1) is also as difficult as solving DL, FAC and one-way hash function, simultaneously.

An adversary may also try collecting l valid signatures $(K_j,\ x_j,\ y_j)$ on message M_j signed by signer, $j = 1, 2, \cdots, l$. She/he attempts to obtain the secret key u and u_1 from the following equations:

$$\begin{aligned}
x_1^2 + y_1^2 k k_1^{h(M_1, x_1)} &= R_1 K_1^2 \bmod n, \\
x_2^2 + y_2^2 k k_1^{h(M_2, x_2)} &= R_2 K_2^2 \bmod n, \\
&\vdots
\end{aligned}$$

$$x_l^2 + y_l^2 k k_1^{h(M_l,x_l)} \quad = \quad R_l K_l^2 \bmod n.$$

In the above l equations, there are $(l+1)$ variables, i.e., k and R_j, $j = 1, 2, \cdots l$, which are not known by the adversary. On the other hand, the adversary first solves DL to obtain k and R_j, then she/he still has to face the problem of FAC to find out u and u_1.

Assume an adversary is able to solve the DL problem, she/he can obtain an integer k from $z = g^k \bmod p$. In this attack, the adversary tries to forge the signature for any message. The adversary can reduce Equation (1) as $x^2 + y^2 k k_1^{h(M,x)} = RK^2 \bmod n$. Given R and K, it is difficult to determine x and y from above equation because of the difficulty of breaking FAC problem. Therefore, the improved scheme can successfully withstand those possible attacks.

Next, we compare the improved scheme and Laih-Kuo's schemes in terms of the number of keys, computational complexity and communication cost. We denote the following notation to analyze the comparison: sk is the number of secret keys; pk is the number of public keys; T_{exp} is the time for modular exponentiation; T_{mul} is the time for modular multiplication; T_{inv} is the time for a modular inverse computation; T_f is the time for performing a one-way hash function $h(\cdot)$; $|x|$ denotes the bit length of x. We ignore the time for performing modular addition computation. Suppose $f(\cdot)$ outputs a t-bit length. The probability of the bit being 0 or 1 is about fifty percent.

The comparisons of the number of keys, computational complexity and communication cost between the improved scheme and Laih-Kuo's schemes are listed in Table 1. From Table 1, we can see that the number of keys of the improved scheme is better than Laih-Kuo's schemes. The computational complexity of the improved scheme is lower than Laih-Kuo's scheme-2. As for the communication cost, the improved scheme performs better than Laih-Kuo's scheme-2 and is the same as Laih-Kuo's scheme-1.

Table 1: The comparison among our scheme and Laih-Kuo's schemes

	Laih-Kuo's Scheme-1	Laih-Kuo's Scheme-2	Our Scheme
The number of keys	sk: $2t+1$ pk: $2t+1$	sk: $t+1$ pk: $t+1$	sk: 2 pk: 2
Computational Complexity	Sign: $T_{exp} + (t/2+5)T_{mul}$ $+T_{inv} + 2T_h$ Verify: $3T_{exp} + (t+4)T_{mul}$ $+2T_h$	Sign: $2T_{exp} + (t/2+9)T_{mul}$ $+T_{inv} + 2T_h$ Verify: $4T_{exp} + (t/2+8)T_{mul}$ $+2T_h$	Sign: $2T_{exp} + 6T_{mul}$ $+T_{inv} + T_h$ Verify: $4T_{exp} + 5T_{mul}$ $+T_h$
Communication Cost	$2\|n\| + \|p\|$	$2\|n\| + 2\|p\|$	$2\|n\| + \|p\|$

5 Conclusions

In this article, we have proposed an improved digital signature scheme. The security of our improved scheme is equivalent to those of Laih-Kuo's schemes based on discrete logarithms and factorization. Some possible attacks have also been considered. Furthermore, we have also demonstrated that the number of keys of the improved scheme is lower compared to Laih-Kuo's schemes.

References

[1] Chin-Chen Chang and Min-Shiang Hwang, "Parallel computation of the generating keys for RSA cryptosystems," *IEE Electronics Letters*, vol. 32, no. 15, pp. 1365–1366, 1996.

[2] T. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. IT-31, pp. 469–472, July 1985.

[3] Min-Shiang Hwang, Chin-Chen Chang, and Kuo-Feng Hwang, "An ElGamal-like cryptosystem for enciphering large messages," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 445–446, 2002.

[4] Min-Shiang Hwang, Cheng-Chi Lee, and Yan-Chi Lai, "Traceability on RSA-based partially signature with low computation," *Applied Mathematics and Computation*, 2002.

[5] Min-Shiang Hwang, Cheng-Chi Lee, and Eric Jui-Lin Lu, "Cryptanalysis of the batch verifying multiple DSA-type digital signatures," *Pakistan Journal of Applied Sciences*, vol. 1, no. 3, pp. 287–288, 2001.

[6] Min-Shiang Hwang, Iuon-Chung Lin, and Kuo-Feng Hwang, "Cryptanalysis of the batch verifying multiple RSA digital signatures," *Informatica*, vol. 11, no. 1, pp. 15–19, 2000.

[7] Chi-Sung Laih and Wen-Chung Kuo, "New signature scheme based on factoring and discrete logarithms," *IEICE Transactions on Fundamentals on Cryptography and Information Security*, vol. E80-A, no. 1, pp. 46–53, 1997.

[8] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Communications of the ACM*, vol. 21, pp. 120–126, Feb. 1978.

[9] Yuan-Liang Tang, Min-Shiang Hwang, and Yan-Chi Lai, "Cryptanalysis of a blind signature scheme based on ElGamal signature," *to appear in International Journal of Pure and Applied Mathematics*, 2002.