블록 암호 알고리즘을 사용하지 않는 인증 암호화 방법

(An Authenticated Encryption Scheme without Block Encryption Algorithms)

이 문 규 † 김 동 규 †† 박 근 수 †††

(Mun-kyu Lee) (Dong Kyue Kim) (Kunsoo Park)

요 약 본 논문에서는 블록 암호 알고리즘을 사용하지 않는 새로운 인증 암호화 방법을 제안한다. 이 방법은 Horster-Michels-Petersen 인증 암호화 방법에 기반하고 있으며, Bao-Deng 서명암호화에 이용된 기법을 적용함으로써 전송자의 서명을 수신자 이외의 임의의 제삼자가 검증할 수 있는 특성을 지닌다. 제안된 방법은 블록 암호 알고리즘을 이용하지 않으므로 구현시 코드 크기를 줄일 수 있는 장점을 가지며, 블록 암호 알고리즘을 이용하는 Bao-Deng 방법과 거의 같은 정도의 계산량 및 통신량만을 필요로 한다. 또한 제안된 방법은 기밀성, 인증성, 부인방지 등 안전성 요건들을 만족시킨다.

키워드 : 공개키 암호시스템, 인증 암호화, 서명암호화

Abstract We propose a new authenticated encryption scheme that does not require any block encryption algorithm. Our scheme is based on the Horster-Michels-Petersen authenticated encryption scheme, and it uses a technique in the Bao-Deng signcryption scheme so that the sender's signature can be verified by an arbitrary third party. Since our scheme does not use any block encryption algorithm, we can reduce the code size in its implementation. The computation and communication costs of the proposed scheme are almost the same as those of the Bao-Deng scheme that uses a block encryption algorithm. Our scheme also satisfies all the security properties such as confidentiality, authenticity and nonrepudiation.

Key words : public key cryptosystem, authenticated encryption, signcryption

1. Introduction

An *authenticated encryption scheme* is a message transmission scheme that sends messages in a secure and authentic way. Basically, an authenticated encryption scheme should satisfy the following properties [1, 2, 3]:

- *confidentiality* : it is computationally infeasible for an adaptive attacker to find out any secret information from a ciphertext.
- *authenticity* (*unforgeability*) : it is computationally infeasible for an adaptive attacker to masquerade as the sender in sending a message.
- *nonrepudiation* : it is computationally feasible for a third party to settle a dispute between the sender and the recipient in an event where the sender denies the fact that he is the originator of the message.

One way to implement such a scheme is first to sign a message and then to encrypt it. Nyberg and Rueppel suggested an authenticated encryption scheme of this type as an application of their message recovery signature scheme [4]. Other

· 이 논문은 2001년도 두뇌한국21사업에 의하여 지원되었음.
† 학생회원 : 서울대학교 컴퓨터공학부
mklee@theory.snu.ac.kr
†† 종신회원 : 부산대학교 전자전기정보컴퓨터공학부 교수
dkkim@islab.cs.pusan.ac.kr
††† 종신회원 : 서울대학교 컴퓨터공학부 교수
kpark@theory.snu.ac.kr
논문접수 : 2001년 5월 14일
심사완료 : 2002년 2월 25일

schemes try to reduce the computation and communication costs by combining encryption and signature. This type of schemes, which we call combined schemes, include the Horster-Michels-Petersen scheme [5] and the Lee-Chang scheme [6]. Recently proposed *signcryption schemes* [1, 2, 3, 7] achieve the same security properties as authenticated encryption schemes, by combining the ElGamal signature [8] and a block encryption scheme. The combined schemes, including signcryption schemes, either do not have nonrepudiation procedures [5, 6] or have inefficient nonrepudiation procedures which use zero-knowledge protocols [1, 2, 3, 7]. To overcome this problem, Bao and Deng [9] proposed a signcryption scheme with signature that can be publicly verified. In their scheme, once a transmitted message is unsigncrypted (i.e., decrypted and verified), anyone can verify the signature if he knows the sender's public key. Gamage, Leiwo and Zheng [10] gave a signcryption scheme based on the Bao-Deng scheme, in which public verification can be done without accessing the plaintext. The schemes of [9] and [10] also use block encryption algorithms.

In this paper, we propose an authenticated encryption scheme with public verifiability that does not require any block encryption algorithm. Our scheme is based on the Horster-Michels-Petersen scheme and it uses a technique due to Bao and Deng [9]. Our scheme has the following properties:

- After a transmitted ciphertext is decrypted and verified by the recipient, the decrypted signature can be verified publicly. Therefore, nonrepudiation is easily achieved.
- Since our scheme does not use any block encryption algorithm, we can reduce the code size. Therefore, our scheme can be used in applications where the memory sizes are restricted.
- The number of modular exponentiations of our scheme is the same as that of [9].
- The length of a message sent from the sender to the recipient is the same as that of [9], and a message for public verification is slightly longer than that of [9].

We also propose a modification of our scheme in which a message from the sender to the recipient is longer than a message for public verification. This modification can be used for a special case where message transmissions from the recipient to third parties are more frequent than those from the sender to the recipient.

This paper is organized as follows. In Section 2, we describe the Horster-Michels-Petersen scheme and the Bao-Deng scheme. Section 3 introduces our new authenticated encryption scheme with public verifiability. In Section 4, we analyze the security and efficiency of our scheme, and compare the efficiency with those of the previous schemes. In Section 5, we describe a modification of our scheme for a special case. We conclude in Section 6.

2. Related Work

2.1 Horster-Michels-Petersen Scheme

We first describe the Horster-Michels-Petersen authenticated encryption scheme [5], which is based on the Nyberg-Rueppel message recovery signature scheme [4]. From now on, *Alice* is the sender and *Bob* is the recipient.

- Initial Setting
  - p : a large prime
  - q : a large prime such that $q \mid p-1$
  - $g \in Z_p^*$: an element of Z_p^* of order q
  - $x_A \in Z_q$: Alice's private key
  - $y_A = g^{x_A} \bmod p$: Alice's public key
  - $x_B \in Z_q$: Bob's private key
  - $y_B = g^{x_B} \bmod p$: Bob's public key
  - $hash$: a one-way hash function
- Alice : to send a message $m \in Z_p$
  - choose a random $k \in Z_q$.
  - compute key $K = hash(y_B^k \bmod p)$.
  - compute ciphertext $r = \frac{m}{K} \bmod p$.
  - compute signature $s = k - x_A \cdot r \bmod q$.
  - send (r, s) to Bob.
- Bob : to recover m from (r, s)

- recover key $K = hash(y_B^s \cdot y_A^{x_B \cdot r \bmod q} \bmod p)$.
- recover message $m = K \cdot r \bmod p$.
- check if m satisfies a predefined redundancy scheme.

In this scheme, the recovery and verification of Alice's message need Bob's private key x_B. Therefore, public verification is impossible. The computation of signature s is different from that of the original ElGamal signature scheme [8]. This is an instance of the generalized ElGamal signature [11, 12].

2.2 Bao-Deng Scheme

We describe the Bao-Deng signcryption scheme [9], which is based on Zheng's signcryption scheme [1, 2]. The initial setting and notations are similar to those of the Horster-Michels-Petersen scheme. Additionally it needs a block encryption algorithm such as DES [13] and IDEA [14]. We will use E_K and D_K to denote the block encryption and decryption algorithms using key K. Two keys $K_1 = g^k \bmod p$ and $K_2 = hash(y_B^k \bmod p)$ are used instead of the single key $K = hash(y_B^k \bmod p)$ to provide public verifiability. We now describe the scheme:

- Alice : to send a message m
  - choose a random $k \in Z_q$.
  - compute key $K_1 = g^k \bmod p$.
  - compute key $K_2 = hash(y_B^k \bmod p)$.
  - compute ciphertext $c = E_{K_2}(m)$.
  - compute commitment $r = hash(m \| K_1)$.
  - compute signature $s = k/(r + x_A) \bmod q$.
  - send (c, r, s) to Bob.
- Bob : to recover m from (c, r, s)
  - recover key $K_1 = (y_A \cdot g^r)^s \bmod p$.
  - recover key $K_2 = hash(K_1^{x_B} \bmod p)$.
  - recover message $m = D_{K_2}(c)$.
  - verify $r = hash(m \| K_1)$.
  - for public verification, forward (m, r, s) to an arbitrary third party.
- Third Party : to verify (m, r, s)
  - recover key $K_1 = (y_A \cdot g^r)^s \bmod p$.
  - verify $r = hash(m \| K_1)$.

In this scheme, public verification is possible. The computation of signature s is another instance of the generalized ElGamal signature.

3. The Proposed Scheme

In this section we propose an authenticated encryption scheme that does not require any block encryption algorithm. The initial setting of the proposed scheme is the same as that of the Horster-Michels-Petersen scheme. But we use two keys $K_1 = hash(g^k \bmod p)$ and $K_2 = hash(y_B^k \bmod p)$, which is a similar technique to that of the Bao-Deng scheme. We concatenate a hash value to message m instead of using a redundancy scheme. We now describe our scheme:

- Alice : to send a message m
  - choose a random $k \in Z_q$.
  - compute key $K_1 = hash(g^k \bmod p)$.
  - compute key $K_2 = hash(y_B^k \bmod p)$.
  - compute ciphertext
    $$r = (m \| hash(m \| K_2)) \cdot K_1 \cdot K_2 \bmod p. \quad (1)$$
  - compute signature $s = k - x_A \cdot r \bmod q$.
  - send (r, s) to Bob.
- Bob : to recover m from (r, s)
  - compute
    $$t = g^s \cdot y_A^{r \bmod q} \bmod p. \quad (2)$$
  - recover key $K_1 = hash(t)$.
  - recover key K_2 by
    $$K_2 = hash(t^{x_B} \bmod p). \quad (3)$$
  - recover message $m' = \frac{r}{K_1 K_2} \bmod p$.
  - partition m' into two parts m_1' and m_2' and verify $m_2' = hash(m_1' \| K_2)$. (The length of each part is predefined.) If it is verified, set $m \leftarrow m_1'$.
  - for public verification, forward (K_2, r, s) to an arbitrary third party.
- Third Party : to verify (K_2, r, s)
  - compute $t = g^s \cdot y_A^{r \bmod q} \bmod p$.
  - recover key $K_1 = hash(t)$.

- recover message $m' = \frac{r}{K_1 K_2} \bmod p$.
- partition m' into two parts m_1' and m_2' and verify $m_2' = hash(m_1' \| K_2)$.

If Alice and Bob follow the protocol properly, then t will satisfy $t = g^k \bmod p$ in (2). Therefore K_1 will be recovered correctly. The recovery of K_2 can also be done since $t^{x_B} \equiv (g^k)^{x_B} \equiv (g^{x_B})^k \equiv y_B^k \bmod p$. Then Bob can recover and verify the message. The situation is the same for a third party.

Our scheme can be used best for small message transmission. Since $(m \| hash(m \| K_2)) \in Z_p$, the size of message m should satisfy $|m| \le |p| - l$, where $|x|$ denotes the number of bits in x, and l is the length of the output of *hash*. But it can be adapted for the case of a long message as follows. Alice partitions message m into $|p|$-bit blocks $m_1, m_2, \ldots, m_t$ (use padding if necessary) and a final block m_{t+1} of size $|p| - l$, and she computes ciphertext blocks $r_1, r_2, \ldots, r_t$ by $r_i = m_i \cdot K_1 \cdot K_2 \bmod p$. The last ciphertext block r_{t+1} is computed by $r_{t+1} = (m_{t+1} \| hash(m_1 \| m_2 \| \cdots \| m_{t+1} \| K_2)) \cdot K_1 \cdot K_2 \bmod p$, and signature is computed by $s = k - x_A \cdot r_1 \cdot r_2 \cdots r_{t-1} \bmod q$. Then Alice sends $(r_1, r_2, \ldots, r_{t+1}, s)$ to Bob. The rest of the scheme can be modified correspondingly. In this adaptation, however, knowledge of one plaintext block m_i enables an intruder to compute other blocks, since we use the same random value k, and thus the same keys K_1 and K_2 for sending more than one message block.

4. Analysis

4.1 Security

The security of our scheme depends on the difficulty of the discrete logarithm problem and the one-wayness of hash functions. Now we analyze the security properties of our scheme.

Confidentiality : Since solving a discrete logarithm problem is infeasible, one cannot extract x_B from y_B. Thus an attacker cannot recover key K_2 by (3) without knowledge of x_B. However, because $(m \| hash(m \| K_2)) \cdot K_1 \cdot K_2 = np + r$ for some n by (1), an attacker could try to recover K_2 by guessing n and factoring $np + r$ as follows. (Note that K_1 can be computed by anyone.)

- Attack against confidentiality
  0. $n \leftarrow -1$.
  1. $n \leftarrow n + 1$.
  2. If $K_1 \nmid np + r$, goto Step 1.
  3. If some x_1 and x_2 satisfy
     $$(np + r) / K_1 = (x_1 \| hash(x_1 \| x_2)) \cdot x_2, \quad (4)$$
     then set $m \leftarrow x_1$, and $K_2 \leftarrow x_2$. Otherwise, goto Step 1.
     (Note that the probability that x_1 and x_2 such that $x_1 \neq m$ or $x_2 \neq K_2$ satisfy (4) is negligible.)

Now we consider the efficiency of this attack.

Lemma 1. The number of times that Step 3 is executed to recover K_2 is approximately 2^l.

Proof. Since $(m \| hash(m \| K_2)) \in Z_p$ and K_1, K_2 are the outputs of *hash*, $np + r$ satisfies $0 \le np + r < p2^{2l}$. Therefore, the number of n's one should try is approximately 2^{2l}. The probability that $np + r$ is divided by K_1 is approximately $1/2^l$, since $K_1 \approx 2^l$ with high probability. Hence the number of n's that reach Step 3 is approximately 2^l. □

Lemma 2. The number of trials in the exhaustive search of K_2 is 2^l.

Proof. It is straightforward since $|K_2| = l$. □

By Lemmas 1 and 2, the number of times that Step 3 is executed is approximately the same as the number of trials in the exhaustive search. On the other hand, each trial of the exhaustive search seems to require much less work than that of a single execution of Step 3, since Step 3 needs a factoring of $(np + r) / K_1$ as well as some additional works. Thus, the attack considered above seems less efficient than even the exhaustive search. (This is the same condition as in the original ElGamal encryption scheme [8] and the

Horster-Michels-Petersen scheme [5].) Also note that the multiplication operation in (1) can be replaced by any other invertible operation.

The one-way property of the hash function in computing K_2 is necessary. If the hash function is not one-way, the preimage (or at least, some partial information for the preimage) of the function can be reconstructed. In this case, an attacker may get the Diffie-Hellman key $K_{DH} = g^{x_A \cdot x_B} \bmod p$ using only one public signature (K_2, r, s) as follows:

$$K_{DH} = \left(hash^{-1}(K_2) \cdot y_B^{-s}\right)^{r^{-1} \bmod q} \bmod p.$$

Then for every further message m_{new} with ciphertext (r_{new}, s_{new}), key $K_{2,new}$ can be computed by

$$K_{2,new} = hash(y_B^{s_{new}} \cdot K_{DH}^{r_{new} \bmod q} \bmod p),$$

and confidentiality is lost.

Authenticity : Because one can extract neither k from $g^k \bmod p$ nor x_A from y_A, only Alice can generate legitimate signatures. Of course, a substitutional attack is possible, but we prevent this by using a one-way hash function in the computation of r [8].

It is necessary to include K_2 in $hash(m \| K_2)$ when r is computed. Otherwise, Bob can forge a message. Suppose we use $r = (m \| hash(m)) \cdot K_1 \cdot K_2 \bmod p$ instead of $r = (m \| hash(m \| K_2)) \cdot K_1 \cdot K_2 \bmod p$. When Bob receives (r, s) from Alice, Bob can generate an arbitrary message m_{new} and compute

$$K_{2,new} = \frac{r}{(m_{new} \| hash(m_{new})) \cdot K_1} \bmod p.$$

Then Bob sends $(K_{2,new}, r, s)$ to a third party, and this is verified.

Nonrepudiation : Once Bob decrypts and verifies (r, s), anyone can verify (K_2, r, s). Therefore, it is computationally feasible for any third party to settle a dispute between Alice and Bob, without Bob's private key or zero-knowledge protocol.

4.2 Efficiency

In this section, we consider the computation and communication costs of the proposed scheme, and compare them with those of the previous schemes.

Computation : We can assume that modular exponentiation is the most time-consuming operation. Our scheme needs 2 modular exponentiations for Alice, 3 for Bob, and 2 for public verification. In fact, we can reduce computation time using a few precomputation techniques. For example, when Alice computes $g^k \bmod p$, she can use the BGMW method [15] or its improvements [16, 17], since g is fixed. The same technique can be applied when Bob computes $g^s \bmod p$. When Bob computes $t^{x_B} \bmod p$, he can use addition-chain methods [18, 19], since x_B is fixed.

Table 1 Number of modular exponentiations

	Alice	Bob	Public Verification
sign-and-encrypt [4]	3	3	2
Horster-Michels-Petersen [5]	1	2	·
Bao-Deng [9]	2	3	2
Proposed scheme	2	3	2

In Table 1, we compare computation costs of the ElGamal-type authenticated encryption schemes. The number of exponentiations of our scheme is smaller than that of the sign-and-encrypt approach [4], and the same as that of Bao-Deng [9]. The Horster-Michels-Petersen scheme [5] uses less computation, but it does not have the property of public verification.

Communication : Here we consider only a single message block, but the analysis is similar when there are more message blocks. In our scheme, the length of a message from Alice to Bob is $|r| + |s| = |p| + |q|$, and the length of a message from Bob to a third party is $|K_2| + |r| + |s| = l + |p| + |q|$. Since r includes message m and its hash value, these are equivalent to $|m| + l + |q|$ and $|m| + 2 \cdot l + |q|$, respectively. Note that the length of a message from Alice to Bob is shorter than that from Bob to a third party, which is a desirable property because public verifications are relatively less frequent than message transmissions from Alice to Bob.

Table 2 Communication Costs

	Alice to Bob	Bob to a third party
sign and-encrypt [4]	$\|m\|+\|redundancy\|+\|p\|+\|q\|$	$\|m\|+\|redundancy\|+\|q\|$
Horster-Michels-Petersen [5]	$\|m\|+\|redundancy\|+\|q\|$	·
Bao-Deng [9]	$\|m\|+l+\|q\|$	$\|m\|+l+\|q\|$
Proposed scheme	$\|m\|+l+\|q\|$	$\|m\|+2\cdot l+\|q\|$

In Table 2, we compare communication costs of the ElGamal-type authenticated encryption schemes. The length of a message from Alice to Bob in our scheme is the same as that in the Bao-Deng scheme [9], and the length of a message from Bob to a third party is slightly longer than that of [9]. But our scheme does not use any block cipher.

5. Modification for a Special Case

It is reasonable to assume that message transmissions from Alice to Bob are more frequent than transmissions from Bob to third parties for public verification. In some cases, however, message transmissions from Bob to third parties can be more frequent than those from Alice to Bob. For example, consider the following scenario. Bob is a member of a group B, and there are secure channels among the members of B. The members of B want to receive a common message from Alice. In this case, Alice sends a message to Bob as an encrypted and signed form through some insecure channel, Bob decrypts and verifies the message, and then he sends the signature including the message to the members of B through secure channels. For this case, we can modify our scheme as follows:

• Initial Setting : the same as that of the Horster-Michels-Petersen scheme.

• Alice : to send a message m
  - choose a random $k \in Z_q$.
  - compute $K_1 = g^k \bmod p$.
  - compute $K_2 = hash(y_B^k \bmod p)$.
  - compute $r = (m \| hash(m \| K_1)) \cdot K_1 \bmod p$.
  - compute $s = k - x_A \cdot r \bmod q$.
  - compute $c = r \cdot K_2 \bmod p$.
  - send (K_1, c, s) to Bob.

• Bob : to recover m from (K_1, c, s)
  - recover $K_2 = hash(K_1^{x_B} \bmod p)$.
  - compute $r = \frac{c}{K_2} \bmod p$.
  - recover $m' = \frac{r}{K_1} \bmod p$.
  - verify $K_1 = g^s \cdot y_A^{r \bmod q} \bmod p$.
  - partition m' into two parts m_1' and m_2' and verify $m_2' = hash(m_1' \| K_1)$. If it is verified, set $m \leftarrow m_1'$.
  - send (r, s) to other members of B.

• Other members of B : to verify (r, s)
  - recover $K_1 = g^s \cdot y_A^{r \bmod q} \bmod p$.
  - recover $m' = \frac{r}{K_1} \bmod p$.
  - partition m' into two parts m_1' and m_2' and verify $m_2' = hash(m_1' \| K_1)$.

Our analysis of the security and computation cost of this scheme is similar to that of Section 4. But the communication cost is different. The length of a message from Alice to Bob is $|K_1|+|c|+|s| = |p|+|m|+l+|q|$, and the length of a message from Bob to others is $|r|+|s| = |m|+l+|q|$. Note that the messages from Bob to others are used not only for nonrepudiation, but also for broadcasting m.

6. Conclusion

In this paper, we proposed an authenticated encryption scheme that does not require any block encryption algorithm. In our scheme, once Bob decrypts and verifies the ciphertext, Alice's signature can be verified by anyone. Our scheme needs the same number of modular exponentiations and almost the same communication cost as those of the Bao-Deng scheme. Since our scheme does not use any additional block encryption algorithm, it has a reduced code size. Therefore, our scheme

can be used in applications where the memory sizes are restricted. We also proposed a modification of our scheme that can be used when message transmissions from Bob to third parties are more frequent than those from Alice to Bob.

References

[1] Zheng, Y., *"Digital signcryption or how to achieve cost(signature & encryption) << cost(signature) + cost(encryption),"* *CRYPTO '97, LNCS* Vol.1294, pp.165-179, Springer-Verlag, 1997.

[2] Zheng, Y., *"Signcryption and its applications in efficient public key solutions,"* *Information Security Workshop (ISW '97), LNCS* Vol.1397, pp.291-312, Springer-Verlag, 1998.

[3] Petersen, H. and Michels, M., *"Cryptanalysis and improvement of signcryption schemes,"* *IEE Proceedings - Computers and Digital Techniques*, Vol.145, No.2, pp.149-151, 1998.

[4] Nyberg, K. and Rueppel, R. A., *"Message recovery for signature schemes based on the discrete logarithm problem,"* *Eurocrypt '94, LNCS* Vol.950, pp.182-193, Springer-Verlag, 1995.

[5] Horster, P., Michels, M. and Petersen, H., *"Authenticated encryption schemes with low communication costs,"* *Electronics Letters*, Vol.30, No.15, pp.1212-1213, 1994.

[6] Lee, W.-B. and Chang, C.-C., *"Authenticated encryption scheme without using a one way function,"* *Electronics Letters*, Vol.31, No.19, pp.1656-1657, 1995.

[7] He, W.-H. and Wu, T.-C., *"Cryptanalysis and improvement of Petersen-Michels signcryption scheme,"* *IEE Proceedings - Computers and Digital Techniques*, Vol.146, No.2, pp.123-124, 1999.

[8] ElGamal, T., *"A public key cryptosystem and a signature scheme based on discrete logarithms,"* *IEEE Transactions on Information Theory*, Vol.IT-31, No.4, pp.469-472, 1985.

[9] Bao, F. and Deng, R. H., *"A signcryption scheme with signature directly verifiable by public key,"* *PKC '98, LNCS* Vol.1431, pp.55-59, Springer-Verlag, 1998.

[10] Gamage, C., Leiwo, J. and Zheng, Y., *"Encrypted message authentication by firewalls,"* *PKC '99, LNCS* Vol.1560, pp.69-81, Springer-Verlag, 1999.

[11] Horster, P., Petersen, H. and Michels, M., *"Meta-ElGamal signature schemes,"* *Proceedings of the second ACM conference on computer and communications security*, pp.96-107, 1994.

[12] Horster, P., Michels, M. and Petersen, H., *"Meta-message recovery and meta-blind signature schemes based on the discrete logarithm problem and their applications,"* *Asiacrypt '94, LNCS* Vol.917, pp.224-237, Springer-Verlag, 1995.

[13] National Bureau of Standards, *"Data Encryption Standard,"* *Federal Information Processing Standards Publication FIPS PUB 46*, 1977.

[14] Lai, X., Massey, J. and Murphy, S., *"Markov ciphers and differential cryptanalysis,"* *Eurocrypt '91, LNCS* Vol.547, pp.17-38, Springer-Verlag, 1991.

[15] Brickell, E. F., Gordon, D. M., McCurley, K. S. and Wilson, D. B., *"Fast exponentiation with precomputation,"* *Eurocrypt '92, LNCS* Vol.658, pp.200-207, Springer-Verlag, 1993.

[16] de Rooij, P., *"Efficient exponentiation using precomputation and vector addition chains,"* *Eurocrypt '94, LNCS* Vol.950, pp.389-399, Springer-Verlag, 1995.

[17] Lim, C. H. and Lee, P. J., *"More flexible exponentiation with precomputation,"* *CRYPTO '94, LNCS* Vol.839, pp.95-107, Springer-Verlag, 1994.

[18] Knuth, D. E., *Seminumerical Algorithms*, 2nd Ed., *The Art of Computer Programming*, Vol.2, Addison-Wesley, Reading, Massachusetts, 1981.

[19] Bos, J. and Coster, M., *"Addition chain heuristics,"* *CRYPTO '89, LNCS* Vol.435, pp.400-407, Springer-Verlag, 1990.

이 문 규

1996년 서울대학교 컴퓨터공학과 학사. 1998년 서울대학교 컴퓨터공학과 석사. 1998년 ~ 현재 서울대학교 컴퓨터공학부 박사과정. 관심분야는 컴퓨터이론, 암호학

김 동 규

정보과학회논문지 : 시스템 및 이론 제 29 권 제 1 호 참조

박 근 수

정보과학회논문지 : 시스템 및 이론 제 29 권 제 1 호 참조