ID-BASED KEY AGREEMENT FOR MULTIMEDIA ENCRYPTION

Xun Yi, Chik How Tan, Chee Kheong Siew and Mahbubur Rahman Syed

Abstract— **In [1], the authors proposed a fast encryption algorithm for multimedia data, called FEA-M. The premise of using FEA-M is that both the sender and the recipient share a common secret key matrix in advance. However, how to achieve the common secret key matrix has not been considered in [1]. In this paper, we come up with an ID-based key agreement scheme for FEA-M which is built on a public-key infrastructure. This key agreement scheme is able to stand up both the passive and active attacks.**

Keywords— **Multimedia encryption, ID-based key agreement, intruder-in-the-middle attack, perfect forward secrecy.**

I. INTRODUCTION

THE security of multimedia data is important for multimedia commerce [2][3][4]. For example, in video on demand and video conferencing applications, it is desirable that only those who have paid for the services can view their video or movies.

The challenges of multimedia data encryption come from two facts. Firstly, multimedia data size is usually very large. Secondly, multimedia data needs to be processed in real time.

For most multimedia applications, the information rate is very high, but the information value is very low. Attacking the encrypted multimedia data is not interesting to adversaries [5], because most multimedia data is different from military secrets or financial information. To break such encryption codes is much more expensive than to buy the services.

The encryption algorithms with high security, such as DES [6], IDEA [7] and AES [8], when applied to high throughput multimedia data, will put great burden on storage space demand and increase latency. Thus, they may not be suitable for multimedia communications.

In [1], the authors proposed a fast encryption algorithm for high throughput multimedia data, called FEA-M, based on properties of Boolean matrices. FEA-M operates on 64×64 Boolean matrices with 64×64 key matrix and its structure is chosen to provide confusion and diffusion and to facilitate both hardware and software implementation. Computation complexity comparisons among some existing encryption algorithms and FEA-M have shown that FEA-M is much faster than others. It needs only about 1.5 XOR operations to encrypt one bit plaintext.

Xun Yi, Chik How Tan and Chee Kheong Siew are with the Information and Communication Institute of Singapore, School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, e-mail: exyi@ntu.edu.sg.

Mahbubur Rahman Syed is with Department of Computer and Information Sciences, Wissink Hall 273, Minnesota State University, Mankato, MN 56001, USA

In order to perform FEA-M, the sender and the recipient must share a common secret 64×64 key matrix. How to establish the common secret key matrix is raised.

Key establishment is a process whereby a shared secret becomes available to two parties, for subsequent cryptographic use. Key establishment is either key distribution or key agreement. Key distribution is a mechanism whereby one party chooses a secret key and then transmits it to another party. Key agreement is a mechanism in which a shared secret is derived by two parties as a function of information contributed by, or associated with, each of these, (ideally) such that no party can predetermine the resulting value.

The well-known Kerberos system [9], based on secret-key cryptosystems, is an on-line key distribution scheme in which a trusted authority acts as a key server and shares a secret key with every user in the network. When a party wishes to communicate with another party, it requests a session key from the trusted authority. The trusted authority generates a session key and sends it in encrypted form for both parties to decrypt.

If it is impractical or undesirable to have an on-line trusted authority, a common approach is to use a key agreement scheme. The first and best well-known key agreement scheme is Diffie-Hellman key agreement scheme [10], which was introduced by Diffie and Hellman in 1976. Unfortunately, Diffie-Hellman scheme is vulnerable to an active adversary who uses an intruder-in-the-middle attack. There is an episode of *The Lucy Show* in which Vivian Vance is having dinner in a restaurant with a date, and Lucille Ball is hiding under the table. Vivian and her date decide to hold hands under the table. Lucy, trying to avoid detection, holds with each of them and they think they are holding hands with each other.

In view of existence of the intruder-in-the-middle attack, a key agreement scheme should itself authenticate two parties' identities at the same time as a secret key is being established. Such a scheme is called authenticated key agreement scheme.

In large-scale multimedia networks, ID-based cryptosystems, which was first introduced by Shamir in [11], are most suitable to develop authenticated key agreement schemes. ID-based cryptosystems, based on public-key infrastructures, adopt users' names, social security numbers, addresses, office phone numbers and so on as users' public keys instead of random integers. The corresponding secret keys are computed by a trusted key generation center and issued to users in the form of smart card when they first join multimedia networks. The smart card contains a microprocessor, an I/O port, a RAM, a ROM with the secret

Contributed Paper

Manuscript received February 22, 2002

key and programs.

In [12], Okamoto proposed an efficient ID-based key agreement based on Diffie-Hellman scheme. Combining Okamoto scheme with particular needs for FEA-M, we come up with a new ID-based key agreement scheme for FEA-M in this paper. This scheme is secure in the sense that both the passive and active attacks can be prevented.

The remaining sections are arranged as follows: Section II introduces Diffie-Hellman key agreement scheme and the intruder-in-the-middle attack. Section III presents our ID-based key agreement scheme. Section VI integrates this scheme with FEA-M. Section V and Section VI analyzes security and performance of this scheme. Conclusion is drawn in the last section.

II. DIFFIE-HELLMAN SCHEME AND INTRUDER-IN-THE-MIDDLE ATTACK

If we do not want to use an online key server, such as Kerberos [9], in multimedia networks, we are forced to use a key agreement scheme to exchange secret key. The first and best known key agreement scheme is Diffie-Hellman scheme [10] as follows.

Assumes that p is a large prime, g is a primitive element of $GF(p)$ (i.e., the set $\{0, 1, 2, \cdots, p-1\}$), and (p, g) are public known. A common secret key k is achieved by the sender A and the recipient B as follows.

Step 1 A randomly chooses an integer a such that $0 < a \leq p-2$ and computes $g^a \ (mod\ p)$.

Step 2 B also randomly chooses an integer b such that $0 < b \leq p-2$ and computes $g^b \ (mod\ p)$.

Step 3 A and B exchange $g^a (mod\ p)$ and $g^b (mod\ p)$.

Step 4 A computes the common secret key k in the following way

$$k = (g^b)^a = g^{ab} \ (mod\ p) \tag{1}$$

B computes k in the following way

$$k = (g^a)^b = g^{ab} \ (mod\ p) \tag{2}$$

At the end of the scheme, A and B agree upon a shared secret key k, which is unavailable to eavesdroppers. This secret may then be converted into cryptographic keying material for a secret-key cryptosystem.

Diffie-Hellman scheme can be illustrated in figure 1.

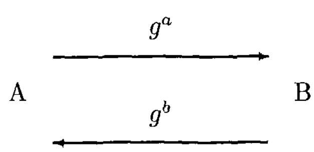


Fig. 1. Diffie-Hellman key agreement scheme

Unfortunately, this scheme is vulnerable to an active adversary who uses the intruder-in-the-middle attack, in which the intruder C intercepts message between A and B and substitutes his own messages, as indicated in figure 2.

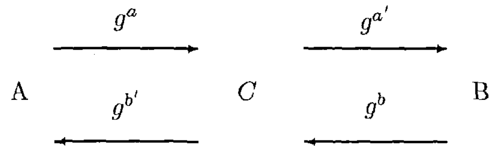


Fig. 2. Intruder-in-the-middle attack to Diffie-Hellman scheme

In figure 2, g^a and g^b are substituted with $g^{a'}$ and $g^{b'}$ respectively by the intruder C with knowing a' and b'.

At the end of the scheme, A has actually established the secret key $g^{ab'}$ with C, and B has established a secret key $g^{a'b}$ with C. When A sends an encrypted message to B, C will be able to decrypt it but B will not. The same situation appears if B sends an encrypted message to A. The intruder-in-the-middle attack is actually undetectable in Diffie-Hellman key agreement scheme.

III. OUR ID-BASED KEY AGREEMENT FOR FEA-M

Our scheme involves a trusted key generation center (TC) which computes and issues secret keys for authorized users. It can be described in three phases, i.e., set-up phase, key generation phase and key agreement phase as follows.

A. Set-up

Same as key generation of RSA [13], TC chooses two distinct big primes p and q such that $p = q = 3 \ (mod\ 4)$ and computes $n = p \cdot q$ and $\phi(n) = (p-1)(q-1)$. It also chooses an integer g which is the primitive element of both $GF(p)$ and $GF(q)$. Next, TC determines a pair of public and private keys (e, d) such that $e \cdot d = 1 \ (mod\ \phi(n))$ where e is randomly chosen between 1 and $\phi(n)$ and $gcd(e, \phi(n)) = 1$. Then TC publishes (n, g, e) but keeps (p, q, d) secret.

B. Key generation

For an authorized user A whose identification information is ID_a, TC calculates

$$s_a = ID_a^{-d} \ (mod\ n) \tag{3}$$

where the size of ID_a is less than that of n.

Then TA stores (n, g, e, ID_a, s_a) into a smart card and issues it to the user A.

It is obvious that if s_a is computed according to formula (3), then

$$s_a^e = ID_a^{-1} \ (mod\ n) \tag{4}$$

C. Key agreement

Assume that both the sender (A) and the recipient (B) have smart cards containing their private keys issued by TC. A common secret key k is established by A and B as follows.

Step 1: A and B insert their smart cards into multimedia encryptor and decryptor respectively.

Step 2: A randomly chooses an integer r_a and computes

$$t_a = g^{r_a + ID_b} s_a \ (mod\ n) \tag{5}$$

while B randomly selects an integer r_b and calculates

$$t_b = g^{r_b+ID_a} s_b \ (mod\ n) \tag{6}$$

Step 3: A and B exchange (ID_a, t_a) and (ID_b, t_b).
Step 4: A computes the common secret key k in the following way:

$$\begin{aligned} k &= ((g^{-ID_a} t_b)^e ID_b)^{r_a} \qquad (7) \\ &= ((g^{r_b} s_b)^e ID_b)^{r_a} \\ &= (g^{r_b e} s_b^e ID_b)^{r_a} \\ &= g^{e r_b r_a} \ (mod\ n) \end{aligned}$$

while B calculates the common secret key in the following way:

$$\begin{aligned} k &= ((g^{-ID_b} t_a)^e ID_a)^{r_b} \\ &= ((g^{r_a} s_a)^e ID_a)^{r_b} \\ &= (g^{r_a e} s_a^e ID_a)^{r_b} \\ &= g^{e r_a r_b} \ (mod\ n) \end{aligned}$$

After the above four steps, A and B reach a common secret key k. k is in the size of the RSA modulus n.

According to FEA-M, a 64×64 key matrix $\mathbf{K}$ and a 64×64 initial matrix $\mathbf{V}_0$ are required to perform encryption. Based on the common secret key k, $\mathbf{K}$ and $\mathbf{V}_0$ are generated respectively as follows.

Suppose that ℓ is the largest integer such that $2^\ell < n$ and m is the smallest integer such that $m \cdot \ell \geq 4096$. With the common secret key k, both A and B compute

$$k_i = k^{2^i} (mod\ n) = (k_{i1}, k_{i2}, \cdots, k_{i\ell}) \tag{8}$$

where $k_{ij} = 0$ or 1 and $i = 1, 2, \cdots, m$ and then concatenate them to form

$$X = k_1 || k_2 || \cdots || k_m = (x_1, x_2, \cdots, x_{m\ell}) \tag{9}$$

where $x_i = 0$ or 1 and $||$ stands for the concatenation of two ℓ-bit blocks.

Then they construct the same initial 64×64 matrix $\mathbf{V}_0$ as follows

$$\mathbf{V}_0 = \begin{bmatrix} x_1 & x_2 & \cdots & x_{63} & x_{64} \\ x_{65} & x_{66} & \cdots & x_{127} & x_{128} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{4033} & x_{4044} & \cdots & x_{4095} & x_{4096} \end{bmatrix}_{64\times 64}$$

Next, A determines the key matrix $\mathbf{K}$ in the following way:

Step 1: Let $i = 1$ and $\mathbf{K} = \mathbf{I}$ where $\mathbf{I}$ is the 64×64 identical matrix. 64 rows of $\mathbf{K}$ are denoted as $R_1, R_2, \cdots, R_{64}$ respectively.
Step 2: Compute

$$R_i = R_i \oplus \bigoplus_{1\leq j \leq 64, j\neq i} x_{64(i-1)+j} \cdot R_j \tag{10}$$

while $\oplus$ stands for bit-by-bit XOR of 64-bit blocks.

Step 3: Let $i = i + 1$. If $i \leq 64$, then go to Step 2.
Step 4: Output 64×64 key matrix $\mathbf{K}$.

B determines a matrix $\mathbf{K}^*$ in the following way:

Step 1: Let $i = 1$ and $\mathbf{K}^* = \mathbf{I}$. 64 columns of $\mathbf{K}^*$ are denoted as $C_1, C_2, \cdots, C_{64}$ respectively.
Step 2: Compute

$$C_i = C_i \oplus \bigoplus_{1\leq j \leq 64, j\neq i} x_{64(i-1)+j} \cdot C_j \tag{11}$$

Step 3: Let $i = i + 1$. If $i \leq 64$, then go to Step 2.
Step 4: Output 64×64 key matrix $\mathbf{K}^*$.

According to linear algebra theory, we know $\mathbf{K} \cdot \mathbf{K}^* = \mathbf{1}$, in other word, $\mathbf{K}^*$ is the inverse of $\mathbf{K}$.

Our scheme can be depicted in figure 3.

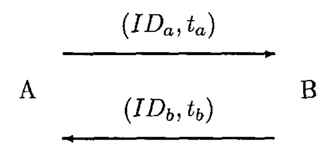


Fig. 3. Our ID-based key agreement scheme

IV. INTEGRATION OF FEA-M WITH OUR ID-BASED KEY AGREEMENT

A. *Encryption and decryption*

After A and B reach a common 64×64 key matrix $\mathbf{K}$ and a common 64×64 initial matrix $\mathbf{V}_0$, FEA-M can be performed as follows.

At first, A divides the plaintext message into blocks $P_1, P_2, \cdots$ with same length 4096 bits. Each 4096-bit block is arranged to a 64×64 matrix which is encrypted into ciphertext matrix in the following way:

$$C_i \ = \ K^i \cdot (P_i + C_{i-1}) \cdot K^{i+1} + P_{i-1} \tag{12}$$

Each ciphertext matrix is decrypted into plaintext matrix in the following way:

$$P_i \ = \ K^{-i} \cdot (C_i + P_{i-1}) \cdot K^{-(i+1)} + C_{i-1} \tag{13}$$

where $i = 1, 2, \cdots$, $+$ and $\cdot$ stand for the matrix addition and multiplication over $GF(2)$ respectively and

$$\mathbf{P}_0 = \mathbf{C}_0 = \mathbf{V}_0$$

Note: In fact, the above encryption algorithm is an improvement of the original FEA-M in [1]. The current FEA-M keeps the same computation complexity as before. In particular, it satisfies the diffusion design criterion because each ciphertext bit in equation (12) depends on each plaintext bit and each key bit. It can be seen from

$$c_{gh}^{(i)} = (\bigoplus_{l=1}^{64} k_{gl}^{(i)} \cdot \bigoplus_{m=1}^{64} ((p_{lm}^{(i)} \oplus c_{lm}^{(i-1)}) \cdot k_{mh}^{(i+1)})) \oplus p_{gh}^{(i-1)}$$

$$= (\bigoplus_{l=1}^{64} \bigoplus_{m=1}^{64} \bigoplus_{n=1}^{64} k_{gl}^{(i)} \cdot (p_{lm}^{(i)} \oplus c_{lm}^{(i-1)}) \cdot (k_{mn}^{(i)} \cdot k_{nh})) \oplus p_{gh}^{(i-1)}$$

where

$$
\begin{aligned}
K &= (k_{nh}^{(1)})_{64\times 64}\\
K^i &= (k_{mn}^{(i)})_{64\times 64}\\
k_{mh}^{(i+1)} &= \bigoplus_{n=1}^{64} k_{mn}^{(i)} \cdot k_{nh}\\
P_i &= (p_{lm}^{(i)})_{64\times 64}\\
C_i &= (c_{lm}^{(i)})_{64\times 64}
\end{aligned}
$$

where $p_{lm}^{(i)}, c_{lm}^{(i)}, k_{mn}^{(i)} \in GF(2)$, $l, m, n = 1, 2, \cdots, 64$.

B. Update of key matrix and initial matrix

Based on the common secret key k currently used by A and B, the updated key matrix and the updated initial matrix are determined by A and B as follows.

Let

$$
\begin{aligned}
\delta &= k^k \ (mod\ n) &\quad (14)\\
k &= \delta &\quad (15)
\end{aligned}
$$

According to formulae (8) and (9), both A and B determine the updated k_i and X and further construct the updated initial matrix $\mathbf{V}_0$, the updated key matrix $\mathbf{K}$ and its inverse $\mathbf{K}^{-1}$ as described in Section III.

V. SECURITY OF OUR ID-BASED KEY AGREEMENT SCHEME

A. Security of our scheme against passive and active attacks

Threats to key agreement schemes mainly come from the passive and active attacks. A passive attack involves an adversary who attempts to determine the secret key by simply recording data and thereafter analyzing it. An active attack involves an adversary who attempts to masquerade as a legal user by altering or replaying messages.

In our scheme, suppose that a passive adversary intercepts t_a and t_b being transmitted over the public channel and attempts to determine the secret key k. Without knowing r_a and r_b, the passive adversary cannot compute k according to formula (7).

Although the passive adversary can infer both g^{er_a} and g^{er_b} by computing

$$
\begin{aligned}
(t_a \cdot g^{-ID_b})^e ID_a = g^{er_a} \ (mod\ n) &\quad (16)\\
(t_b \cdot g^{-ID_a})^e ID_b = g^{er_b} \ (mod\ n) &\quad (17)
\end{aligned}
$$

it is intractable for him to determine $k = g^{er_ar_b}$ with g^{er_a} and g^{er_b}.

In addition, without knowledge of the private key d of the trusted key generation center (TC), it is difficult for the passive adversary to solve g^{r_a} from g^{er_a} and further determine s_a from t_a (or solve g^{r_b} from g^{er_b} and further determine s_b from t_b). This difficulty is same as that of solving the intractable discrete logarithm problem [14][15].

Therefore, our scheme is able to withstand the passive attack.

Next, we exam the security of our scheme against the active attack shown in figure 4.

$$
\begin{array}{ccccc}
 & \xrightarrow{(ID_a,\, g^{r_a+ID_b}s_a)} & & \xrightarrow{(ID_a,\, g^{r'_a+ID_b})} & \\
A & \xleftarrow[(ID_b,\, g^{r'_b+ID_a})]{} & C & \xleftarrow[(ID_b,\, g^{r_b+ID_a}s_b)]{} & B
\end{array}
$$

Fig. 4. Intruder-in-the-middle attack to our ID-based key agreement scheme

In figure 4, $g^{r_a+ID_b}s_a$ and $g^{r_b+ID_a}s_b$ are substituted with $g^{r'_a+ID_b}$ and $g^{r'_b+ID_a}$ respectively by the intruder C with knowing r'_a and r'_b.

After A receives $g^{r'_b+ID_a}$, A follows our scheme to compute

$$
\begin{aligned}
k_a &= ((g^{-ID_a}g^{r'_b+ID_a})^e ID_b)^{r_a}\\
&= g^{er_ar'_b} ID_b^{r_a}
\end{aligned}
$$

After B receives $g^{r'_a+ID_b}$, B follows our scheme to compute

$$
\begin{aligned}
k_b &= ((g^{-ID_b}g^{r'_a+ID_b})^e ID_a)^{r_b}\\
&= g^{er'_ar_b} ID_a^{r_b}
\end{aligned}
$$

Although the intruder C can obtain g^{er_a} and g^{er_b} with formulae (16) and (17) and further computes $g^{er_ar'_b}$ and $g^{er'_ar_b}$ with knowledge of r'_a and r'_b, C can know neither k_a nor k_b without knowledge of r_a and r_b.

Once A and B detect that they can decrypt a ciphertext into a significant plaintext, they will restart the key agreement scheme again. As a consequence, the intruder-in-the-middle attack to our scheme cannot succeed.

B. Security of key matrix and initial matrix update

In our scheme, the updated key matrix $\mathbf{K}$ and the updated initial matrix $\mathbf{V}_0$ are determined on the basis of the current secret key k shared by A and B as described in Section IV. k is updated according to formulae (14) and (15).

If $\mathbf{K}$ and $\mathbf{V}_0$ are compromised, k^{2^i} $(i = 1, 2, \cdots)$ are almost compromised according to the structure of $\mathbf{V}_0$. However, since $p = q = 3(mod\ 4)$ and factors of n is not available to any attacker, determining k with k^{2^i} $(i = 1, 2, \cdots)$ is intractable according to Fiat-Shamir identification scheme [16]. Without knowledge of k, it is hard to know the updated secret key k^k and further the updated key matrix $\mathbf{K}$ and the updated initial matrix $\mathbf{V}_0$.

C. Perfect forward secrecy

A scheme is said to have perfect forward secrecy if compromise of long-term keys does not compromise past session keys. In our scheme, the long-term key of an user A is s_a which is just employed to certify the authenticity of t_a sent by A. r_a is randomly chosen and thus independent of the long-term key s_a. Even if both A's long-term key s_a and B's long-term key s_b are compromised, their past random integers r_a and r_b are not revealed to any attacker. In this case, the attacker still cannot determine the previous secret key k used by A and B with formula (7). Therefore, our scheme has perfect forward secrecy property.

VI. PERFORMANCE OF OUR ID-BASED KEY AGREEMENT SCHEME

In our scheme, all users adopt the uniform RSA modulus n generated by the trusted key generation center TC and thereby they are free from the trouble of generating large primes to determine their RSA modulus.

Most of authenticated key agreement schemes use certificates to ensure the authenticity of public keys of users. So both parties in key agreement need to authenticate certificates issued by a certification authority at first. It is no doubt that the verification brings more computation into authenticated key agreement schemes.

Our scheme is based on identities of users. The public key of an user in our scheme is just a meaningful information such as his name, social security number, address, phone number and so on instead of random number. Therefore, one party in our scheme does not need verify whether the public key of another party is authentic or not.

From formula (5), we know that A is required to compute one exponentiation and one multiplication modulo n. From formula (7), we can see that A is required to compute two exponentiations and two multiplications modulo n if g^{-ID_a} is precomputed.

Assume that the length of n is ℓ bits and m is the smallest integer such that $m \cdot \ell \geq 4096$, then A needs to compute m multiplications modulo n in formula (8).

In order to determine the 64×64 key matrix based on X, A needs perform $64 \cdot 64$ XOR of 64-bit blocks.

Suppose modular multiplications of ℓ-bit integers are performed in ℓ^2 times, r_a and r_b are chosen as any λ-bit integers where $\lambda << \ell$, and pseudo-random generators run in λ^2 times, then computation complexity of our scheme is

$$2\lambda^2 + 3\lambda\ell^2 + 3\ell^2 + m\ell^2 + 64 \cdot 64 \cdot 64 \tag{18}$$

Usually, ℓ is about 1024 and λ is about 160. In this case, the computation complexity of our scheme is $O(2^{30})$ bit operations (about three modular exponentiations).

The computation complexity for B to run our scheme is the same as that for A because B performs the same types of computation as A.

VII. CONCLUSION

Key agreement is an important issue for multimedia encryption. In this paper, we have come up with an ID-based key agreement scheme for the fast multimedia encryption algorithm FEA-M.

Security analysis has shown that our scheme is able to withstand both the passive attack and the active attack. In addition, our scheme has perfect forward secrecy property.

Performance analysis has shown that only three modular exponentiations are required to compute in our scheme when the RSA modulus n has 1024 bit and all random integers are chosen as 160-bit integers.

REFERENCES

[1] X. Yi, C. K. Tan, C. K. Siew and M. R. Syed, "Fast encryption for multimedia", *IEEE Transactions on Consumer Electronics*, vol. 47, no. 1, pp. 101-107, Feb 2001.

[2] F. Andres, "Multimedia and security", *IEEE Multimedia*, vol. 8, no. 3, pp. 20-21, Jul-Sept 2001.

[3] H. H. Yu, D. Kundur and C. Y. Lin, "Spies, thieves, and lies: the battle for multimedia in the digital era", *IEEE Multimedia*, vol. 8, no. 3, pp. 8-12, Jul-Sept 2001.

[4] J. Dittman, P. Wohlmacher and K. Nahrstedt, "Using cryptographic and watermarking algorithms", *IEEE Multimedia*, vol. 8, no. 3, pp. 54-65, Jul-Sept 2001.

[5] B. M. Macq and J. J. Quisquater, "Cryptology for digital TV broadcasting", *Proceedings of the IEEE*, vol. 83, no. 6, pp. 944-957, Jun 1995.

[6] FIPS PUB 46, "Data encryption standard", *Federal Information Processing Standards Publications*, U. S. Department of Commerce/National Bureau of Standards, Jan 1977.

[7] X. Lai and J. Massey, "A proposal for a new block encryption standard", In *Proceedings of EUROCRYPT'90*, Aarhus, Denmark, May 1990, pp. 389-404.

[8] FIPS PUB 197, "Advanced encryption standard", *Federal Information Processing Standards Publications*, U. S. Department of Commerce/N.I.S.T., National Technical Information Service, Nov 2001.

[9] J. Kohl and C. Neuman, "The Kerberos network authentication service", *Network Working Group Request for Comments 1510*, Sept 1993.

[10] W. Diffie and M. E. Hellman, "New directions in cryptography", *IEEE Transactions on Information Theory*, vol. IT-22, no. 6, pp. 644-654, 1976.

[11] A. Shamir, "Identity-based cryptosystems and signature scheme", *Proceedings of CRYPTO'84*, Santa Barbara, California, USA, Aug 1984, pp. 47-53.

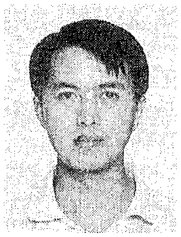
[12] E. Okamoto and K. Tanaka, "Key distribution system based on identification information", *IEEE Selected Areas in Communications*, vol. 7, no. 4, pp. 481-485, May 1989.

[13] R. L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", *Communications of ACM*, vol. 21 , no. 2, pp. 120-126, Feb 1978.

[14] V. Shoup, "Lower bounds for discrete logarithms and related problems", *Proceedings of EUROCRYPT'97*, Konstanz, Germany, May 1997, pp. 256-266.

[15] A. M. Odlyzko, "Discrete logarithms: the past and the future", *Designs, Codes and Cryptography*, vol. 19, no. 2-3, pp. 129-145, 2000.

[16] A. Fiat and A. Shamir, "How to prove yourself: practical solutions to identification and signature problems", in *Proceedings of CRYPTO'86*, Santa Barbara, California, USA, Aug 1986, pp. 186-194.



Xun Yi is currently a teaching fellow with School of Electrical and Electronic Engineering, Nanyang Technology University (NTU), Singapore. He obtained Ph.D. degree from Xidian University, P. R. China in 1995. From 1995 to 1997, he was a postdoctoral research fellow at National Mobile Communication Lab, Southeast University, P. R. China. From 1997-1998, he was a research fellow in School of Computing, National University of Singapore (NUS). From 1998-1999, he worked as an assistant professor in School of Information Science, Japan Advanced Institute of Science and Technology (JAIST), Japan. His research areas include cryptography, multimedia network security, wireless communications, electronic commerce and mobile agent technology.

Chik How Tan is currently an assistant professor with School of Electrical and Electronic Engineering, Nanyang Technology University (NTU), Singapore. He received a B.Sc. (Honours) in Mathematics from National University of Singapore in 1984 and M.A. and Ph.D. in Mathematics from University of Wisconsin, USA in 1990 and 1992, respectively. From 1992 to 2000, he worked at DSO National Laboratories, Singapore. Then he joined Nanyang Technological University in 2000. His research interests include cryptography, coding theory, information security, network and Internet security.

Chee Kheong Siew, David is currently the Director of Information Communication Institute of Singapore (ICIS), School of EEE, Nanyang Technological University, Singapore. He obtained his B. Eng. in Electrical Engineering from University of Singapore in 1979 and MSc. in Communication Engineering, Imperial College in 1987. After a six and a half years stint in the industry, he joined NTU as a Lecturer in 1986 and was appointed Associate Professor in 1999. He was seconded to National Computer Board (NCB) as the Deputy Director, ICIS in August 1995 and managed the transfer of ICIS from NCB to NTU in 1996. In January 1997, he was appointed as the Director of this Institute. His research interests include e-commerce, information security, traffic shaping, neural networks and network performance.

Mahbubur Rahman Syed is currently a professor in Department of Computer and Information Sciences Minnesota State University. He obtained his M.Sc degree in June 1978 and Ph.D degree in November 1980 both in the field of Computer Technology from the Faculty of Electrical Engineering at Budapest Technical University in Hungary. Dr. Syed has worked for several institutions in Hungary, Bangladesh, Australia and in USA. He has about 20 years of experience in teaching, in industry, in research and in academic leadership in the field of computer science and engineering. His research interests include multimedia computing and communications, multimedia document retrieval and processing, educational multimedia, video over ATM, microprocessor based designs, VHDL design, pattern recognition, neural network and artificial intelligence, fuzzy logic, electronic commerce and security issues. He has about 100 refereed publications. Dr. Syed is the co-editor-in-chief of the Tamkang Journal of Science and Engineering - an International Journal.