

An innovative revocation scheme for one-to-many E-services

Ren-Hung Lin^{a,*}, Jinn-Ke Jan^{b,*}

^a *Institute of Applied Mathematics, National Chung Hsing University, Taichung 402, Taiwan, ROC*

^b *Institute of Computer Science, National Chung Hsing University, Taichung 402, Taiwan, ROC*

Received 13 June 2005; received in revised form 25 December 2006; accepted 8 February 2007

Available online 14 February 2007

Abstract

E-service providers usually encrypt each E-service message to prevent unauthorized receivers from reading the content and they change the encryption keys periodically for security reasons. In some one-to-many E-services, like global positioning systems (GPS) or news feeds systems, it is not reasonable to ask customers to stay on-line all the time and save the changes of keys to the system. If registered customers are not on-line, they will miss the re-keying messages and will not be able to decrypt any E-service content. We propose a practical revocation scheme for one-to-many E-services with stateless and service diversity properties. Our scheme reduces the key-storage requirement of service providers to a constant size and customers are not required to be on-line constantly (stateless). Also, service providers can provide various E-services for their customers at the same time (service diversity) and customers can request any E-service or cancel it whenever needed.

© 2007 Elsevier B.V. All rights reserved.

Keywords: One-to-many services; Security; Performance; Encryption

1. Introduction

In recent Internet applications, one-to-many E-services have become the focus of research and applications development [2,10,12]. Using minimal resources, a service provider (SP) can employ broadcast mechanisms to transmit data to all customers simultaneously. One major challenge for one-to-many E-services is to provide efficient methods for controlling authorized access. Generally speaking, the SP encrypts E-service messages to prevent illegal receivers accessing them.

For performance consideration, it is more efficient to employ a symmetric encryption method during transmissions. A symmetric encryption uses transposition and substitution skills to process the original message (plaintext) to a confused output (ciphertext). It is much faster than an asymmetric encryption method like RSA [11] or Elgamal [3] algorithm because RSA-like cryptosystem takes too

much computing time to perform exponentiation operations. For each transmission, if we can change the session key, we will get more security. The SP may share an individual master key with every customer and encrypt the session key with related master keys. When the number of customers is growing, the SP needs a large space and bandwidth to manage $O(n)$ master keys and $O(n)$ re-keying messages. Therefore, it is critical to manage these keys efficiently.

Wallner et al. [14] and Wong et al. [16] almost simultaneously first proposed the tree-based key management for one-to-many communication. Their scheme reduces the complexity of re-keying message to $O(\log n)$ but the SP needs $O(n)$ spaces to store all related keys. Naor et al. [9] proposed a stateless scheme in which it is not necessary for customers to be on-line constantly to receive each re-keying message. However, the SP still needs $O(n)$ spaces to store all related keys in their scheme. Mihaljevic [8] proposed a reconfigurable key management scheme to obtain an appropriate communications-storage-processing trade-off. Employing Mihaljevic's scheme, a small increase of communication overload causes a large reduction of the

* Corresponding authors.

E-mail addresses: renhung@amath.nchu.edu.tw (R.-H. Lin), jkjan@cs.nchu.edu.tw (J.-K. Jan).

storage and processing overload. But in the optimal case, the SP still needs $O(n)$ spaces to store all related keys. Tseng [13] reduces the key storage of the SP to a constant complexity by employing pseudo random function (PRF) concept. Employing Tseng’s PRF concept, a minimized key storage management scheme is designed for the SP.

We propose a practical one-to-many E-services scheme with stateless and service diversity properties. By employing our scheme, the SP can provide customers various one-to-many E-services at the same time. Customers can request any E-service or cancel it at any time. In addition, customers do not need to stay on-line all the time to save the changes to the system. In the first section, we give a formal definition of PRF. Then we illustrate our one-to-many E-services scheme through three phases and give an example of our scheme in action. Finally, there are performance and security analysis in the last section.

2. Pseudo random functions

A pseudo random function expands a key with a seed to a pseudo random output. A PRF is sometimes called a Key Derivation Function (KDF) when it is used to derive subsequent keys. We give the formal definition of PRF as follows:

2.1. Definition of PRF

A PRF is a deterministic function $f: \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}^n$ which is computable in polynomial time and takes two inputs $x, s \in \{0, 1\}^n$. Consider x to be a variable and let s be a hidden random seed and function index, $f(x, s) = f_s(x)$. The function $x \rightarrow f_s(x)$ is considered a good PRF if it looks like a random function. Fig. 1 illustrates a PRF that takes two input values and outputs a random number, and Table 1 presents some practical algorithms for PRF constructions [19].

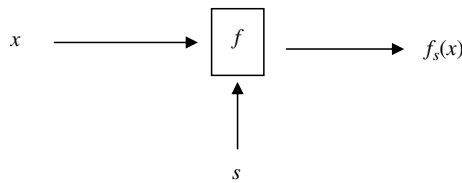


Fig. 1. PRF takes two input values and outputs a random number.

Table 1
Practical algorithms for PRF constructions

Algorithm	Designers	Published
HMAC-PRF	David Hopwood	2000
KDF2	P1363 Working Group	2001
SSL3-PRF	Netscape Communications Corp.	2000
TLS-PRF	IETF Transport Layer Security Working Group	1999

3. Our one-to-many E-services scheme

In this section, we illustrate our one-to-many E-Services scheme phase by phase and use a simplified example to employ our scheme.

3.1. Initiation phase

The SP analyzes the group size of customers in an E-commerce market in advance. Then the SP constructs a logical full binary key tree of appropriate size, as Fig. 2 illustrates, to manage key-encryption-keys (KEKs) for customers. Each customer is linked to a unique leaf node and assigned a subset of KEKs. A session key K must be encrypted with KEKs and sent to valid receivers in each E-service transmission. Note that in our scheme, the SP encrypted each message with a randomly chosen session key. Additionally, the SP maintains a privilege table to manage which E-service customers have paid for it (see Table 2).

For a node j in level i , the SP generates corresponding node key $L_{i,j}$ by employing a PRF $f_s(x)$.

$$L_{i,j} = f_s(x) = f_s(g(i, j)) \tag{1}$$

The value s is a secure random seed that only the SP holds and the function g maps (i, j) to an integer in a polynomial time. The SP assigns $\log N + 1$ keys associated with the nodes along the path from the root to leaf m to customer C_m .

Let \mathcal{N} be the set of all customers, $|\mathcal{N}| = N$, and $R_i \in \mathcal{N}$ be a group of $|R_i| = r_i$ customers whose decryption privi-

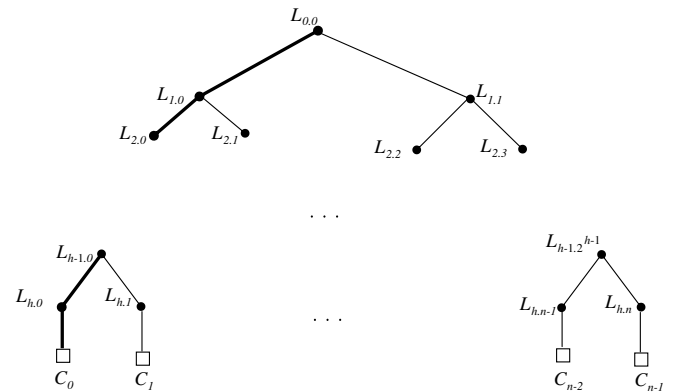


Fig. 2. A key tree with height h for managing n customers. Customer C_0 is assigned $\{L_{0,0}, L_{1,0}, \dots, L_{h-1,0}, L_{h,0}\}$ as his subset of KEKs.

Table 2
A privilege table to manage each customer’s access right

E-services	C_0	C_1	...	C_{n-1}
S_1	✓	✓		
S_2		✓	✓	
...		✓		✓
S_k	✓		✓	✓

leges should be revoked in the E-services S_i . $\text{Min} T(R_i)$ is the minimal subtree of the full binary tree that connects all the leaves in R_i .

3.2. Transmission phase

For a given set R_i of revoked customers in E-service S_i , let $(L_{i_1:j_1}, L_{i_2:j_2}, \dots, L_{i_m:j_m})$ be all the root node keys of the subtrees whose roots are adjacent to nodes of outdegree 1 in $\text{Min} T(R_i)$, but they are not in $\text{Min} T(R_i)$. Note that the set R_i can be obtained by checking the S_i tuple from the privilege table and the SP does not need to reconstruct $\text{Min} T(R_i)$ for each transmission until some customers want to stop paying for E-service S_i . Then SP encrypts session key K with KEKs and broadcasts the message

$$\langle (i_1, j_1), (i_2, j_2), \dots, (i_m, j_m), [E_{L_{i_1:j_1}}(K), E_{L_{i_2:j_2}}(K), \dots, E_{L_{i_m:j_m}}(K)], E'_K(M) \rangle \quad (2)$$

There are m indices, $\{(i_1, j_1), (i_2, j_2), \dots, (i_m, j_m)\}$, to indicate which customers can decrypt this message correctly. $E_{L_{i_m:j_m}}(K)$ means using node key $L_{i_m:j_m}$ to encrypt session key K . $E'_K(M)$ means encrypting the E-service message M with session key K in this transmission. Function E and E' are symmetric encryption algorithms, e.g., DES or AES, etc.

3.3. Decoding phase

Upon receiving an E-service message, the customer finds that one index of his node keys belongs to $\{(i_1, j_1), (i_2, j_2), \dots, (i_m, j_m)\}$. Note that in case a customer $C_k \in R_i$, $0 \leq k \leq n - 1$, the result is null. The authorized customer then extracts the corresponding key $L_{i_w:j_w}$, $w \in \{1, 2, \dots, m\}$, and computes $D_{L_{i_w:j_w}}(E_{L_{i_w:j_w}}(K))$ to obtain K . Finally, the legal customer can compute $D'_K(E'_K(M))$ to obtain message M . Function D and D' is the corresponding decryption algorithm (reversing algorithm) for function E and E' , respectively.

3.4. A simplified example

Suppose there are eight customers in the E-commerce market. The SP constructs a logical full binary key tree with height 3 to manage key-encryption-keys (KEKs) for customers. As illustrated in Fig. 3, each customer C_i , $0 \leq i \leq 7$, is assigned a subset of KEKs defined in the initiation phase. For example, customer C_5 is assigned $\{L_{0,0}, L_{1,1}, L_{2,2}, L_{3,5}\}$ as his subset of KEKs.

After customers choose their own E-services, the SP constructs a privilege table as depicted in Table 3 to manage which E-service customers have paid for it. From the privilege table, the SP can find customers C_3 and C_6 should be revoked when transmitting messages for E-service S_1 and obtain $R_1 = \{C_3, C_6\}$ and $\text{Min} T(R_1)$. The subtree with bold edges as depicted in Fig. 4 is the $\text{Min} T(R_1)$ for E-service S_1 . $L_{2,0}, L_{2,2}, L_{3,2}$, and $L_{3,7}$ are all the root node keys

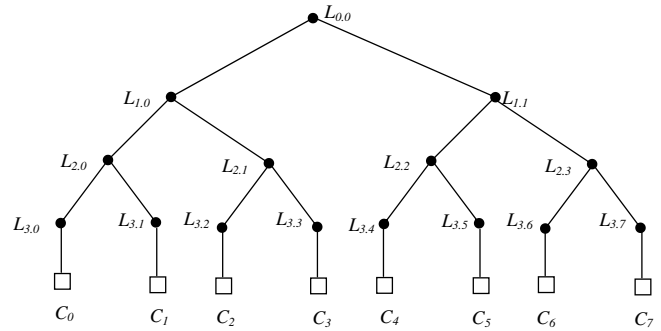


Fig. 3. A key tree to manage eight customers.

Table 3
A privilege table to manage each customer's access right

E-services	C_0	C_1	C_2	C_3	C_4	C_5	C_6	C_7
S_1	✓	✓	✓		✓	✓		✓
S_2				✓		✓	✓	
S_3	✓	✓	✓	✓	✓	✓		✓

of the subtrees whose roots are adjacent to nodes of outdegree 1 in $\text{Min} T(R_1)$, but they are not in $\text{Min} T(R_1)$. The SP encrypts the session key K_1 with $L_{2,0}, L_{2,2}, L_{3,2}, L_{3,7}$ individually and broadcasts the following message for E-service S_1 to customers:

$$\langle (2, 0), (2, 2), (3, 2), (3, 7), [E_{L_{2,0}}(K_1), E_{L_{2,2}}(K_1), E_{L_{3,2}}(K_1), E_{L_{3,7}}(K_1)], E'_{K_1}(M) \rangle \quad (3)$$

Upon receiving an E-service message, customer C_5 finds that one index of his node keys belongs to $\{(2, 0), (2, 2), (3, 2), (3, 7)\}$ (in case $C_k \in R_1$, $0 \leq k \leq 7$, the result is null). C_5 then extracts the corresponding key $L_{2,2}$ and computes $D_{L_{2,2}}(E_{L_{2,2}}(K_1))$ to obtain K_1 . Finally, C_5 obtains and outputs M by computing $D'_{K_1}(E'_{K_1}(M))$.

Similarly, the SP constructs $\text{Min} T(R_2)$ and $\text{Min} T(R_3)$ by employing the same processing process mentioned above. Then the SP broadcasts the following messages(4) and messages(5) for E-service S_2 and E-service S_3 , respectively to customer C_i , $0 \leq i \leq 7$.

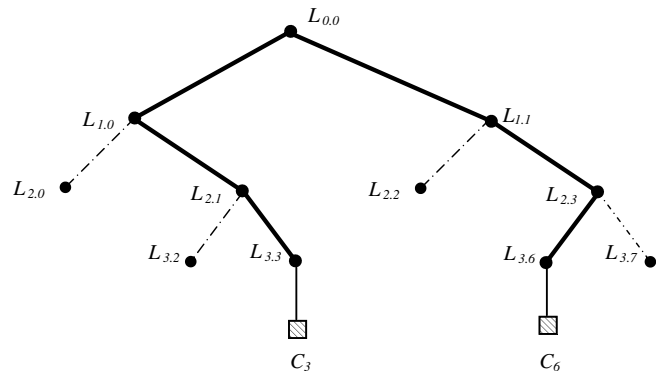


Fig. 4. Subtree $\text{Min} T(R_1)$ and all the root node keys of the subtrees whose roots are adjacent to nodes of outdegree 1 in $\text{Min} T(R_1)$, but they are not in $\text{Min} T(R_1)$.

$$\langle (3, 3), (3, 5), (3, 6), [E_{L_{3,3}}(K_2), E_{L_{3,5}}(K_2), E_{L_{3,6}}(K_2)], E'_{K_2}(M) \rangle \quad (4)$$

$$\langle (1, 0), (2, 2), (3, 7), [E_{L_{1,0}}(K_3), E_{L_{2,2}}(K_3), E_{L_{3,7}}(K_3)], E'_{K_3}(M) \rangle \quad (5)$$

Only the valid customers can decode these messages correctly.

4. Performance analysis

Since the key space is a critical issue in secure one-to-many E-service communications, we firstly examine the key space requirement in our scheme. By employing the PRF technique, the SP holds only one hidden random seed (secret key) s in the initiation phase. The value s is independent to the number of registered customers. This means the SP does not worry about the growing number of registered customers causing the growth of its key space. The SP just uses its hidden seed s to generate appropriate KEKs and share them with specific customers, respectively.

Customers in our scheme are assigned to $\log N + 1$ keys as their subsets of KEKs. The number N is the estimated number of customers in an E-service market. Consider a large one-to-many E-service market with $2^{28} \approx 256,000,000$ customers [5]. These customers can be represented by a logical full binary tree with 28 levels. Each customer has to store 29 KEKs in their receiving devices. If each KEK is a 128-bit AES key [18], they are all fit into less than one half of a kilobyte of memory.

We analyze the computational complexity of our scheme as follows. From message (2) mentioned above, we can clearly observe that valid customers firstly decrypt the received message to obtain the session key K and then process another decryption with session K to recover the original message M . All we need is two decryption computations for each received message so the time complexity of decryption is $O(1)$.

It is a little complicated to examine the order of encryption complexity. In the transmission phase, the number of encryption computations is equal to the number of nodes of outdegree 1 in $\text{Min } T(R_i)$. Referring to [20], the full version of [9], we employ a similar mathematical induction technique to show that our encryption complexity is $O(r \log_2(N/r))$. Assume it is true for trees of depth i , i.e. there is at most $r \cdot (\log_2(2^i/r)) = r \cdot (i - \log_2 r)$ nodes of outdegree 1 in a subtree with r leaves. From this assumption, let us derive the maximum number of nodes of outdegree 1 in a tree with depth $i + 1$. If all the leaves are linked in one subtree of depth i , obviously the total number of nodes of outdegree 1 is at most

$$r \cdot (i - \log_2 r) + 1 \leq r \cdot (i - \log_2 r) + r = r \cdot (i + 1 - \log_2 r). \quad (6)$$

Otherwise, the number of nodes of outdegree 1 is the sum of the numbers of nodes of outdegree 1 both in the left and the right subtree. Suppose there are $r_1, r_1 \geq 1$, and r_2 ,

Table 4

The key-storage requirement and the computation complexity in our scheme

Key storage at service provider	Key storage at customer	Order of encryption	Order of decryption
$O(1)$	$O(\log_2 N)$	$O(r \log_2(N/r))$	$O(1)$

Table 5

Comparisons with other one-to-many communication schemes.

	Wong et al.'s scheme	Mihaljevic's SKT-A scheme	Our proposed scheme
Key storage requirement of the SP	$O(n)$	$O(n)$	$O(1)$
Key storage requirement of each customer	$O(\log_2 n)$	$O(H_0^{1.5} - H_0 - \log_2 n), H_0 < \log_2 n$	$O(\log_2 n)$
Stateless	No	Yes	Yes

$r_2 \geq 1$, leaves in the left and the right subtree respectively, then we obtain $r = r_1 + r_2$. The number of nodes of outdegree 1 is at most

$$r_1 \cdot (i - \log_2 r_1) + r_2 \cdot (i - \log_2 r_2) = r \cdot i - (r_1 \log_2 r_1 + r_2 \log_2 r_2). \quad (7)$$

Since $(r_1 \log_2 r_1 + r_2 \log_2 r_2) \geq r(\log_2 r - 1)$, we apply this inequality to (7) and get

$$r_1 \cdot (i - \log_2 r_1) + r_2 \cdot (i - \log_2 r_2) \leq r \cdot i - r \cdot (\log_2 r - 1) = r \cdot (i + 1 - \log_2 r). \quad (8)$$

From (6) and (8), it means $O(r \log_2(2^{i+1}/r))$ holds. Therefore, the order of encryption complexity, $O(r \log_2(N/r))$, is true for all N . We summarize our analysis of the key-storage requirement and the computation complexity in Table 4.

We compare the proposed scheme with the previously proposed binary-tree structure schemes, which include Wong et al.'s logical tree-based scheme and Mihaljevic's scheme. Table 5 represents a comparison of the above schemes in terms of the key-storage requirement of the SP, the key-storage requirement of customers, and the stateless property. The newly proposed scheme shows less storage requirements than the two previously proposed schemes.

5. Applying our scheme to pay-tv systems

With the technology of digital television (DTV) broadcasting, the SP may deliver different types of multimedia content to many customers simultaneously. In the Pay-TV system, the SP charges for one-to-many E-services according to what content each customer chooses. The SP encrypts (scrambles) each multimedia content and then broadcasts it to customers. Customers can recover specific

multimedia content based on their payment. By employing Conditional Access System (CAS) [6,7,17], the SP can prevent unauthorized access to content provided for legal subscribers. Generally, the CAS operates in a three-level hierarchy for key distribution. The SP chooses a variable random control word (CW) as a random seed of a PRF to generate a pseudo random sequence and encrypts the content by the sequence. Then the SP encrypts CW by authorization key (AK) to form Entitlement Control Message (ECM). The AK and other entitlement information will be encrypted together by Master Private Key (MPK) to form Entitlement Management Message (EMM). The SP transmits ECM, EMM, and encrypted content to subscribers in the broadcasting channel. The Set-Top-Box (STB) at the receivers' end decrypts EMM with MPK stored in the smartcard to recover AK. After decrypting the CW with the AK, the CW will be input into the PRF to get the same pseudo random sequence to decrypt the received content.

Fig. 5 represents the basic components of a typical Pay-TV system. The MPK is initially written to a smartcard and the SP delivers the smartcard to the subscriber when he/she completes the registration to the SP. By changing the MPK stored in the smartcard on-line, the SP can remove or add the access right to a protected one-to-many E-service. If the device at the receiver end is off-line while the SP is changing the MPK, the subscriber can not recover the subsequent encrypted contents for a period of time. Moreover, with a three-level hierarchy in CAS, receivers must perform three deciphering operations to recover multimedia content broadcasted to them. In comparison with CAS, our proposal achieves two merits. With our revocation scheme, it is not necessary to ask the customers to stay on-line constantly and save the changes of the keys to the system. Devices at the customers' end may not work for a period of time because of a power failure or a system crash. After the system has recovered, end devices can decrypt the subsequent multimedia content immediately since our proposed scheme incorporates the stateless property. Customers do not need to worry about the SP transmitting re-keying messages during which time their devices will not

work properly. Additionally, our proposed scheme reduces the number of deciphering operations from three to two at customers' devices. They only perform one deciphering operation to recover the session key and another operation to recover the multimedia content with the decrypted session key.

As the technical analysis mentioned above shows, our scheme ensures that each customer can receive his/her subscribed content properly when the SP transmits E-Services through the broadcasting channel. From the users' point of view, the SP provides a trustworthy and efficient solution for them because of the stateless property and because of the decreased time needed for deciphering. However, in a real E-commerce application, the SP should additionally design friendly Web site interfaces to enhance the obvious trustworthiness to customers. In addition to a secure technical mechanism [15], a well-designed Web site is another essential to increase customers' trust [1,4].

6. Security analysis

In our proposed protocol, the SP uses PRF to manage the KEKs and assigns the appropriate subset keys to each customer. Given a number generated by the PRF, attackers are unable to predict either the next number or the previous number. If the space of a random seed s is large, our scheme can resist brute-force attacks easily. Therefore, attackers are unable to compute each KEK ($L_{i,j}$) or to derive the secret random seed s as mentioned in (1).

Each one-to-many E-service message is encrypted with a session key before transmitting to customers. The session key is randomly chosen by the SP for each E-service message encryption. If the SP uses 128-bit AES key as a session key, it is computationally infeasible for attackers to guess the correct value. There are 2^{128} different possibilities in every session key.

7. Conclusions

The proposed scheme in this article reduces the key-storage requirements of a service provider to a constant size

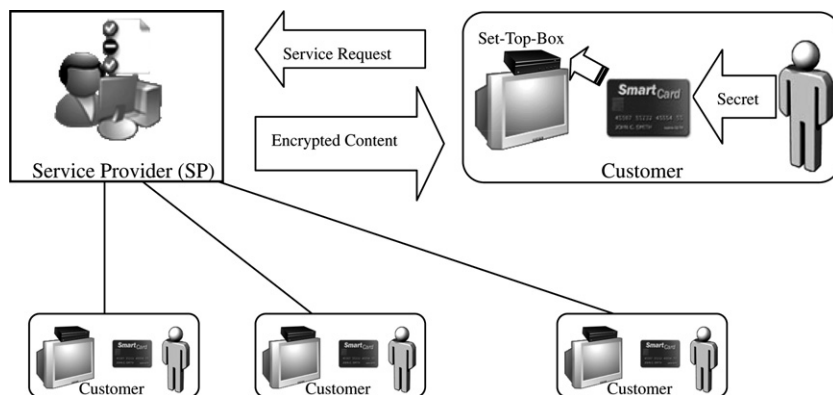


Fig. 5. Basic components of a Pay-TV system.

and it is practical in one-to-many E-services communications. A service provider can efficiently manage each customer's service request and provide various E-services at the same time. Also, customers are not required to be on-line constantly to receive the re-keying message since our scheme fits the stateless requirement.

References

- [1] F. Akhter, D. Hobbs, Z. Maamar, Determining the factors which engender customer trust in business-to-consumer (B2C) electronic commerce, in: Proceedings of the IEEE International Conference on E-Commerce Technology, 2004, pp. 291–294.
- [2] R. Canetti, et al., Multicast Security: a taxonomy and some efficient constructions, in: INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies, 2, 1999, pp. 708–716.
- [3] T. Elgamal, A public key cryptosystem and a signature scheme based on discrete logarithms, *IEEE Transactions on Information Theory* 31 (4) (1985) 473–481.
- [4] U. Greveler, How Pay-TV becomes E-Commerce, in: Proceedings of the Seventh IEEE International Conference on E-Commerce Technology, 2005, pp. 508–511.
- [5] D. Halevy, A. Shamir, The LSD broadcast encryption scheme, in: CRYPTO 2002, LNCS 2442, 2002, pp. 47–60.
- [6] Y.-L. Huang, S.-P. Shieh, F.-S. Ho, J.-C. Wang, Efficient key distribution schemes for secure media delivery in Pay-TV systems, *IEEE Transactions on Multimedia* 6 (5) (2004) 760–769.
- [7] B.-F. Liu, W.-J. Zhang, T.-P. Jiang, A scalable key distribution scheme for conditional access system in digital Pay-TV system, *IEEE Transactions on Consumer Electronics* 50 (2) (2004) 632–637.
- [8] M. Mihaljevic, Key management schemes for stateless receivers based on time varying heterogeneous logical key hierarchy, *Advances in Cryptology – ASIACRYPT 2003*, Springer, Heidelberg, 2003, pp. 137–154.
- [9] D. Naor, M. Naor, J. Lotspiech, Revocation and tracing schemes for stateless receivers, CRYPTO 2001: LNCS, 2139, 2001, pp. 41–62.
- [10] S. Rafaei, D. Hutchison, A survey of key management for secure group communication, *ACM Computer Survey* 35.3 (2003) 309–329.
- [11] R. Rivest, A. Shamir, L. Adleman, A method for obtaining digital signatures and public key cryptosystems, *Communications of the ACM* (February) (1978).
- [12] A.E. Sayed, V. Roca, L. Mathy, A survey of proposals for an alternative group communication service, *IEEE on Networks* 17.1 (2003) 46–51.
- [13] Y.-M. Tseng, A scalable key-management scheme with minimizing key storage for secure group communications, *International Journal of Network Management* 13 (2003) 419–425.
- [14] D. Wallner, E. Harder, R. Agee, 1999. Key Management for Multicast: Issues and Architectures. RFC 2627.
- [15] Y. Wang, T.-Y. Li, LITESET/A++: a new agent-assisted secure payment protocol, in: Proceedings of the IEEE International Conference on E-Commerce Technology, 2004, pp. 244–251.
- [16] C.-K. Wong, M. Gouda, S.S. Lam, Secure group communications using key graphs, *IEEE/ACM Transactions on Networking* 8.1 (2000) 16–30.
- [17] Q. Xie, S.-B. Zeng, X.-J. Yu, A smart-card-based conditional access subsystem separation scheme for digital TV broadcasting, *IEEE Transactions on Consumer Electronics* 51 (3) (2005) 925–932.
- [18] Announcing the Advanced Encryption Standard (AES), Federal Information Processing Standards Publication 197, November 26, 2001.
- [19] www.users.zetnet.co.uk/hopwood/crypto/scan/prf.html.
- [20] www.wisdom.weizmann.ac.il/~naor/PAPERS/2nl.pdf.