A Robust Time-Bound Hierarchical Key Assignment Scheme

Yu-Li Lin, Chien-Lung Hsu, and Yu-Hao Chuang

***Abstract*—Recently, Chien proposed a time-bound hierarchical key assignment scheme based on tamper-resistant devices, which improves the performance in terms of the computational efforts and the implementation costs. Later, Santis *et al*. demonstrated a collusion attack on Chien's scheme to show that collusive malicious insiders can cooperatively derive some encryption keys and further proposed an improvement to eliminate the security flaws inherent in Chien's scheme. In this paper, we will prove that Santis *et al*.'s key derivation is incorrect and their claimed security requirements cannot be achieved. On the other hand, we will further propose a new key derivation to improve the weakness in Santis *et al*., scheme.**

***Index Terms*—Access control, cryptography, hierarchical key assignment.**

I. Introduction

In the real world, there are several examples of hierarchies such as military, government units, business companies, and etc. All users in such a system form a user hierarchy and can be assigned into a number of disjoint sets of security classes, say $C=\{C_1,C_2,\cdots,C_n\}$, which is partially ordered by a binary relation "$\leq$". The symbol "$C_j \leq C_i$" means that the security level of class C_j is lower than or equal to that of the class C_i. It also implies that the users in the security class C_i can access to the information held by those in the class C_j, while the opposite is not allowed. For instance, a business company can be regarded as a partially ordered hierarchy. All users in a business company are assigned to the different security classes according to their responsibilities such as top management, divisions, and departments. There are several projects in a department. Members of a project team can have access rights to their project, but are unable to access information about other projects. A manager of a division is authorized to access information in all departments and projects within that division. Hence, it is important to determine the access privilege management for a partially ordered hierarchy to resolve above access control problems.

In 1983, Akl and Taylor [1] first proposed a cryptographic key assignment scheme in an arbitrary partial order set (POSET) hierarchy in order to solve the access control problems. MacKinnon *et al*., [2] presented an optimal algorithm, called the canonical algorithm, to reduce the number of public parameters as compared with Akl and Taylor's scheme [1]. Sandhu [3] addressed a novel cryptographic solution to access control problem for the special case of a rooted tree hierarchy. The keys for security classes are iteratively generated by using one-way functions. This approach is easier for implementation.

Harn and Lin [4] proposed a bottom-up key generation scheme, instead of using a top-down approach as in the Akl and Taylor's scheme. The results showed that Harn and Lin's scheme is not only more efficient in the memory utilization since it needs less space to keep public information, but also can handle new user's insertion without changing all keys. Some interesting applications have become more prevalent, for example, electronic paper subscription and digital TV broadcasting. In these applications, a user may be assigned to a certain class for only a period of time. The users need distinct keys to encrypt their data in different time periods. The implementation of conventional key assignment schemes [1]-[7] required users to handle a large number of keys, which are impractical and inefficient for implementation.

In 2002, Tzeng [8] proposed a time-bound cryptographic key assignment scheme in which the cryptographic keys of a class is distinct for each time period. Tzeng's scheme can be applied to broadcast encrypted data to authorized users through a broadcast channel and to construct a flexible cryptographic key backup system. However, Tzeng's scheme is insecure against the collusion attack demonstrated by Yi and Ye [9]. A coalition of classes with distinct time periods can cooperate to compute unauthorized keys that they should not be able to learn. To improve Tzeng's scheme, Chien [10] later proposed a new time-bound hierarchical key assignment scheme that employed a low-cost tamper-resistant device to perform simple arithmetic operations. Chien's scheme reduces the implementation costs and the computational loads. Santis, Ferrara, and Masucci [11] recently showed that Chien's scheme is still vulnerable to collusion attack which is contrary to his claimed security requirements. That is, some malicious users can collusively to extend the time period of their secret keys. Santis, Ferrara, and Masucci [12] further proposed an improvement to eliminate the security flaws inherent in Chien's scheme. This paper, however, will demonstrate the incorrectness of Santis *et al*.'s scheme. That is, the key derivation will fail in Santis *et al.*'s scheme, which implies their scheme cannot achieve the claimed security requirements. We finally will propose an improved scheme to correct Santis *et al.*, scheme.

Rest of this paper is sketched as follows. In Section II, we will review Santis *et al*., scheme [12]. In Section III, we will demonstrate the incorrectness of Santis *et al*., scheme and

Manuscript received January 10, 2014; revised March 21, 2014.

Yu-Li Lin is with Ministry of Justice Investigation Bureau, New Taipei City 231, Taiwan (e-mail: andrina_lin@yahoo.com.tw).

Chien-Lung Hsu and Yu-Hao Chuang is with Department of Information Management, Chang-Gung University, Taiwan (e-mail: clhsu@mail.cgu.edu.tw, yuhao0512@gmail.com).

propose an improvement. Finally, we give conclusions in Section IV

II. Review of Santis et al.'s Scheme

Santis *et al*.'s scheme [12] consists of the initialization, the user registration, the key generation, and the key derivation phases. In the initialization phase, a trusted agent (TA) determines all system parameters. In the user registration phase, a user can join the system and be assigned a secret key according to his security class. In the key generation phase, TA will determine a secret key for encrypting data belonging to each class at each time period. In the key derivation phase, the legal user can use his own secret key to derive the encrypting key for his class or lower class at some time period. Suppose the system has *n* disjoint classes, $\{C_1, C_2, \cdots, C_n\}$, and the time is divided into *z* periods, starting at time period 1. Let ID_i be the identity of the class C_i. A user belonging to class C_i from time period t_1 to t_2 is able to derive the secret key $K_{j,t}$ of the intended class C_j at time period *t*, where $C_j \le C_i$ and $t_1 \le t \le t_2$. Detailed descriptions of these phases are given below.

Initalization. Initially, the trusted agent TA determines his own secret key *X* and a secure one-way hash function *h*, and randomly chooses two secret values, *a* and *b*, as the time-bound seeds for time periods. TA determines a secret key k_i for each class C_i (for *i* = 1, 2, …, *n*) and a public value

$$r_{ij} = h(X \| ID_i \| ID_j \| k_i) \oplus h(X \| k_j) \quad (1)$$

For the relationship between C_i and C_j where $C_j \le C_i$ and there is no class C_l such that $C_j \le C_l \le C_i$. Note that the symbol "‖" denotes the string concatenation and "$\oplus$" denotes the bit-wise XOR operation. Finally, TA publishes *h*, ID_i's, and all r_{ij}'s on an authenticated public board.

User registration. When a user joins the system and is assigned to a class C_i in the time interval $[t_1, t_2]$, TA transmits C_i's secret key k_i to him via a secure channel and gives him a tamper-resistant device containing ID_i, TA's secret key *X*, and two hash values ($h^{t_1}(a), h^{z-t_2}(b)$). Note that no one can access the tamper-resistant device to obtain the stored information.

Key generation. Assume a user U_i belonging to the class C_i during the time period *t* ($t_1 \le t \le t_2$) can encrypt the data by the key $K_{i,t}$. The key $K_{i,t}$ is defined as $K_{i,t} = h(h(X \| k_i) \oplus h^t(a) \oplus h^{z-t}(b))$.

Key derivation. When a user U_i belonging to class C_i in time interval $[t_1, t_2]$ wants to decrypt the encrypted data of the class C_j at time period *t*, where $C_j \le C_i$, there is no class C_l such that $C_j \le C_l \le C_i$, and $t_1 \le t \le t_2$. The user U_i inputs the public value r_{ij}, the identity ID_i, and the secret key k_i to the temper-resistant device. The device performs the following steps to derive the decryption key $K_{j,t}$:

Step 1. Use the public information r_{ij} and C_i's secret key k_i to compute C_j's secret information $h(X \| k_j)$ by

$$h(X \| k_j) = r_{ij} \oplus h(X \| ID_i \| ID_j \| k_i) \quad (2)$$

Step 2. Compute $h^t(a)$ and $h^{z-t}(b)$ as $h^t(a) = h^{t-t_1}(h^{t_1}(a))$ and $h^{z-t}(b) = h^{t_2-t}(h^{z-t_2}(b))$.

Step 3. Derive C_j's secret key $K_{j,t}$ by

$$K_{j,t} = h(h(X \| k_j) \oplus h^t(a) \oplus h^{z-t}(b)).$$

If there exists some class(es) between C_i and C_j, the user U_i can use above method to iteratively derive the secret key(s) of C_l's, where $C_j \le C_l \le C_i$ in time interval $[t_1, t_2]$. All above steps will be performed iteratively by the same way.

III. Incorrectness and Improvement of Santis Et al., Scheme

In this section, we will show the incorrectness of Santis *et al*., scheme and then propose the improvement to correct Santis *et al*., scheme.

A. Incorrectness of Santis et al.'s scheme

According to Santis *et al*.'s scheme, each class C_i has the secret key k_i and the tamper-resistant device after the initialization phase. Consider the situation that a user U_i belonging to C_i wants to access the information held by some user U_j of C_j, where $C_j \le C_i$ and there is no class C_l such that $C_j \le C_l \le C_i$, U_i can input public information (r_{ij}, ID_i, ID_j) and his secret key k_i into his tamper-resistant device to obtain $h(X \| k_j)$. The device uses $h(X \| k_j)$ to derive U_j's secret key $K_{j,t}$ at time period *t*. Finally, U_i can decrypt U_j's information with the derived secret key $K_{j,t}$.

Consider the another situation that a user U_i in C_i wants to decrypt the encrypted data held by some user in C_j from the path C_i to C_j in a user hierarchy, where $C_j \le C_{l_i} \le \cdots \le C_{l_2} \le C_{l_1} \le C_i$, U_i has to perform the computations iteratively by the following equations:

$$h(X \| k_{l_1}) = r_{il_1} \oplus h(X \| ID_i \| ID_{l_1} \| k_i) \quad (3)$$

$$h(X \| k_{l_2}) = r_{l_1 l_2} \oplus h(X \| ID_{l_1} \| ID_{l_2} \| k_{l_1}), \cdots, \quad (4)$$

$$h(X \| k_j) = r_{l_i j} \oplus h(X \| ID_{l_i} \| ID_j \| k_{l_i})$$

From Eq. (3), we can precisely know that $h(X \| k_{l_1})$

consists of TA's secret key X and U_{l_1} 's secret key k_{l_1} . If we want to carry out Eq. (4) to derive the secret value $h(X \| k_{l_2})$, we must first obtain secret key k_{l_1} from $h(X \| k_{l_1})$ and then feed it to the right-hand side of Eq. (4). Unfortunately, U_i cannot derive the next secret value $h(X \| k_{l_2}) = r_{l_1 l_2} \oplus h(X \| ID_{l_1} \| ID_{l_2} \| k_{l_1})$ since k_{l_1} is protected by the one-way hash function. Based on the intractability of reversing the one-way hash function, it can be seen that the user U_i cannot derive the secret keys, $k_{l_1}, k_{l_2}, \cdots, k_{l_i}$, from $h(X \| k_{l_1}), h(X \| k_{l_2}), \cdots, h(X \| k_{l_i})$.

For example, the set of classes is organized as a user hierarchy such as Fig. 1. The users belonging to C_1, C_2, C_3 , and C_4 are associated with time intervals $[t_1, t_6]$, $[t_2, t_5]$, $[t_3, t_4]$, and $[t_2, t_3]$, respectively, where $1 \le t_1 < t_2 < t_3 < t_4 < t_5 < t_6 \le z$. As shown in Fig. 1, the users belonging to C_1 have access right to the information held by those belonging to C_4 . When the users in C_1 wants to derive C_4 's secret key $K_{4,t}$ in the time period t ($t_2 \le t \le t_5$), they have to input the public value r_{12} , the identity ID_1 , and the secret key k_1 into his tamper-resistant device to compute $h(X \| k_2) = r_{12} \oplus h(X \| ID_1 \| ID_2 \| k_1)$. If the users belonging to C_1 has the ability to reverse the one-way hash function h, he can derive k_2 from the derived $h(X \| k_2)$. Then, they can further derive the secret key by the following equations:

$$h(X \| k_4) = r_{24} \oplus h(X \| ID_2 \| ID_4 \| k_2)$$

$$h^t(a) = h^{t-t_2}(h^{t_2}(a)) \text{ , } h^{z-t}(b) = h^{t_5-t}(h^{z-t_5}(b))$$

$$K_{4,t} = h(h(X \| k_4) \oplus h^t(a) \oplus h^{z-t}(b))$$

Since the security of Santis *et al.*'s scheme is primarily assumed based on the intractability of reversing the one-way hash function h, the users in C_1 cannot derive k_2 from $h(X \| k_2)$.

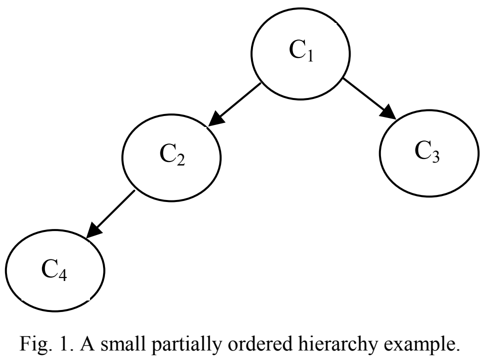

Fig. 1. A small partially ordered hierarchy example.

B. The Proposed Improvement

The incorrectness of Santis *et al*.'s scheme is caused by the fact that the secret key k_i cannot be derived from the secret value $h(X \| k_i)$. To eliminate this incorrectness, we can replace Eqs. (1) and (2) with Eqs. (1*) and (2*), respectively:

$$r_{ij} = h(ID_i \| ID_j \| h(X \| k_i)) \oplus h(X \| k_j)$$

$$h(X \| k_j) = r_{ij} \oplus h(ID_i \| ID_j \| h(X \| k_i))$$

When a user U_i belonging to class C_i in time interval $[t_1, t_2]$ wants to decrypt the encrypted data of the class C_j at time period t, where $C_j \le C_i$, there is no class C_l such that $C_j \le C_l \le C_i$, and $t_1 \le t \le t_2$. The user U_i inputs the public value r_{ij} , the identity ID_i , and his own secret key k_i to the temper-resistant device. The device performs the following steps to derive the decryption key $K_{j,t}$:

Step 1. Use the public information r_{ij} and C_i 's secret key k_i to compute C_j 's secret information $h(X \| k_j)$ by $h(X \| k_j) = r_{ij} \oplus h(ID_i \| ID_j \| h(X \| k_i))$.

Step 2. Compute $h^t(a)$ and $h^{z-t}(b)$ as

$$h^t(a) = h^{t-t_1}(h^{t_1}(a))$$

$$h^{z-t}(b) = h^{t_2-t}(h^{z-t_2}(b))$$

Step 3. Derive C_j 's secret key $K_{j,t}$ by $K_{j,t} = h(h(X \| k_j) \oplus h^t(a) \oplus h^{z-t}(b))$.

If there exists some class(es) between C_i and C_j , the user U_i can use above method to iteratively derive the secret key(s) of C_l 's, where $C_j \le C_l \le C_i$ in time interval $[t_1, t_2]$. All above steps will be performed iteratively by the same way.

From Eqs. (1*) and (2*), we can see that the value $h(X \| k_j)$ can be easily derived by $h(X \| k_j) = r_{ij} \oplus h(ID_i \| ID_j \| h(X \| k_i))$ with the knowledge of $h(X \| k_i)$. Hence, incorrectness of Santis *et al.*'s scheme is corrected in the proposed improvement. The security analysis of our proposed scheme is similar to that of Santis *et al.*'s scheme based on the same cryptographic assumptions. The interested readers are encouraged to refer [12].

IV. Conclusions

We have demonstrated the incorrectness of Santis *et al.*'s scheme, which implies their scheme cannot achieve the claimed security requirements. That is, Santis *et al.*'s key derivation only allows the users to derive the secret key for the direct successor of the intended class. The secret keys for all immediate successors of the intended class however cannot be derived, which is contrary to their claimed security requirements. We finally proposed an improvement to fix the pointed out problem.

Acknowledgment

We would like to thank anonymous referees for their valuable suggestions. We thank Healthy Aging Research Center (HARC) of Chang Gung University for excellent technical assistance. This work was supported in part by the Healthy Aging Research Center Grant CMRPD1B0331, in part by the Chang Gung Memorial Hospital Grant CMRPD3D0041, and in part by National Science Council under the grant NSC 100-2628-H-182-001-MY3.

REFERENCES

[1] S. G. Akl and P. D. Taylor, "Cryptographic solution to a problem of access control in a hierarchy," *ACM Trans. Comput. Syst.*, vol. 1, no. 3, pp. 239-248, August, 1983.

[2] S. J. MacKinnon, P. D. Taylor, H. Meijer, and S. G. Akl, "An optimal algorithm for assigning cryptographic keys to control access in a hierarchy," *IEEE Trans. Comput.*, vol. 34, no. 9, pp. 797-802, September, 1985.

[3] R. S. Sandhu, "Cryptographic implementation of a tree hierarchy for access control," *Inf. Process. Lett.*, vol. 27, no. 2, pp. 95-98, February, 1988.

[4] L. Harn and H. Y. Lin, "A cryptographic key generation scheme for multilevel data security," *Comput. Secur.,* vol. 9, no. 6, pp. 539-546, October, 1990.

[5] G. Ateniese, A. De Santis, A. L. Ferrara, and B. Masucci, "A note on time-bound hierarchical key assignment schemes," *Inf. Process. Lett.*, vol. 113, no. 5–6, pp. 151-155, 2013.

[6] A. D. Santis, A. L. Ferrara, and B. Masucci, "New constructions for provably-secure time-bound hierarchical key assignment schemes," *Theor. Comput. Sci.*, vol. 407, no. 1–3, pp. 213-230, 2008.

[7] J. H. Yeh, "A secure time-bound hierarchical key assignment scheme based on RSA public key cryptosystem," *Inf. Process. Lett.,* vol. 105, no. 4, pp. 117-120, February, 2008.

[8] W. G. Tzeng, "A time-bound cryptographic key assignment scheme for access control in a hierarchy," *IEEE Trans. Knowl. Data Eng.,* vol. 14, no. 1, pp. 182-188, January, 2002.

[9] X. Yi and Y. Ye, "Security of Tzeng's time-bound key assignment scheme for access control in a hierarchy," *IEEE Trans. Knowl. Data Eng.*, vol. 15, no. 4, pp. 1054-1055, July, 2003.

[10] H. Y. Chien, "Efficient time-bound hierarchical key assignment scheme," *IEEE Trans. Knowl. Data Eng.,* vol. 16, no. 10, pp. 1301-1304, October, 2004.

[11] A. Santis, A. Ferrara, and B. Masucci, "On the insecurity of a time-bound hierarchical key assignment scheme," Tech. Report, Dept. of Math., University of Waterloo*,* 2005.

[12] A. Santis, A. Ferrara, and B. Masucci, "Enforcing the security of a time-bound hierarchical key assignment scheme," *Inf. Sci.,* vol. 176, no. 12, pp. 1684-1694, June, 2006.

Yu-Li Lin received the B.S. degree in 1998. She received her M.S. degree and the Ph.D. degree from National Taiwan University. All in Information Management, Taiwan. Her research interests include cryptography, information security, key management, network security and digital forensics.

Chien-Lung Hsu received a B.S. degree in business administration from the National Taiwan University of Science and Technology, Taiwan in 1995; He received his M.S. degree in information management from the National Taiwan University of Science and Technology, Taiwan in 1997. And a Ph.D. degree in information management from the National Taiwan University of Science and Technology, Taiwan in 2002. Respectively. He was an assistant professor and an associate professor in the Department of Information Management, Chang Gung University (CGU), and Taiwan from 2004 to 2007 and from 2007 to 2011. Currently, he is a professor in the Department of Information Management, Chang Gung University since 2011. He is also the leader of the Ubiquitous Security and Applications Lab, the director of Chinese Cryptology Information Security Association (CCISA, Taiwan), the chair of Education Promotion Committee of CCISA, the member of Academia-Industry Cooperation Committee of CCISA, the chair of Program of RFID Applications in Logistics Supply Chain Management of CGU, the chair of Program of Information Security with Medical Applications of CGU, the director of Division of Instructional Support of Computer Center of CGU, the researcher of Healthy Aging Research Center (HARC) of CGU, the researcher of Elder Industry Development and Research Center (EIDRC) of CGU, and the senior researcher of Taiwan Information Security Center (TWISC). His current research includes cryptography, information security, wireless sensor network, mobile commerce, digital forensics, vehicular system security, healthcare system and user acceptance, smart home system, and etc.

Yu-Hao Chuang received a bachelor's degree from Chinese Culture University in 2004. He received his master's degree in information management from Chang Gung University in 2006. And a Ph.D. degree in information management from National Central University in 2012, respectively. His current research topics include information security, information systems, social technology, innovation, strategic information systems, and mobile commerce.