Robust Dynamic Access Control Scheme in a User Hierarchy Based on One-Way Hash Function

Chien-Lung Hsu, Pei-Ling Tsai, and Yen-Chun Chou
Department of Information Management, Chang Gung University
clhsu@mail.cgu.edu.tw, *m9344014@stmail.cgu.edu.tw*,
m9655005@stmail.cgu.edu.tw

Abstract- *In 2004, Yang and Li addressed an access control scheme based on one-way hash function that uses limited number of keys and some public hash functions to solve the dynamic access control problem. Our research finds that in their scheme users can overstep his authority to access unauthorized information. We further propose a robust dynamic access control scheme to eliminate such an attack.*

Keywords: Access control; One-way hash function; Cryptographic key assignment scheme; Dynamic access control.

1. Introduction

Evident in the widespread use of computer techniques and internet, computer system has gradually gained acceptance during people's life. In general, the security of electronic documents is protected by users' encryption techniques. However, considering the hierarchical property of an enterprise, how to share electronic documents confidentially has become an important issue. A primitive access control is to allow only the authorized personnel to access these electronic documents and block out any unauthorized access. Obviously, in an organization a manager must have the privilege to access employee data to protect the business digital information, while employees are assigned lower authority.

The access control in a computer communication system is usually formed as a user hierarchy called Partially Ordered Set (POSET), in which the users and their information are divided into *n* disjoined sets $\mathrm{SC}_1, \mathrm{SC}_2, ..., \mathrm{SC}_n$, called security clearance. Using "$\leq$" as a binary partially ordered relation, in this set $\mathrm{SC}_j \leq \mathrm{SC}_i$ denotes that SC_i has higher security clearance than SC_j -- that is, the users in SC_i have the authority to access data belonging to the users in SC_j but not the other way around.

The simplest way to address the access control problem is to allow users in each security clearance to hold all available secret keys in his direct or indirect security clearances. However, it is memory intensive for key retention, and as the hierarchy becomes larger and more complex, the key management becomes difficult. Akl and Taylor [AT83] introduced the cryptographic key assignment scheme that simplifies the key generation and derivation procedure to resolve the access control problem in the POSET. MacKinnon *et al*. [MTMA85] pointed out that

Akl and Taylor's scheme is vulnerable to the collusion attack and proposed a remedy. However, the dynamic access control problem remains unresolved; all secret keys and public parameters still have to be regenerated when adding, deleting, or changing nodes. Harn and Lin [HL90] proposed a bottom-up cryptographic key assignment scheme (compared to the top-down ones proposed by Akl and Talyor [AT83] and MacKinnon *et al.* [MTMA85]) to address such inefficiency. Nevertheless, the efficiency of dynamic access control problem has not yet been fully resolved.

Our comprehensive survey of solutions to dynamic access control problem [CHW92, WWH95, KSCL99, WC01] reveals that, although the number of parameters has been reduced the computation cost and time complexity are still high. Yang and Li [YL04] recently introduced an access control scheme based on one-way hash function that only uses a limited number of keys and some hash functions, and it also decreases the number of parameters needed to be altered. Their approach is a vast improvement in resolving the dynamic access control problem and the method requires little storage space. Unfortunately, we find that there exist some security flaws in Yang and Li's scheme [YL04]. Specifically, users can still exceed the pre-assigned rights and access unauthorized information when (1) a new node is added, or (2) a new node is added after another is deleted. In terms of security flaws, this paper will give the detailed discussion on these two situations and then propose improvements to eliminate the pointed attacks.

2. Review of Yang and Li's scheme

Yang and Li [YL04] proposed an access control scheme based on one way hash function, in which a trusted CA (central authority) first determines a set of public one way hash functions $H=\{H_1, H_2, ..., H_n\}$, where *n* is the degree of the hierarchy, that is, the maximum number of direct child nodes in the hierarchy. The degree in Figure 1 is 3, for example. In the key generation side, the CA assigns an arbitrary secret key for each security clearance first. Considering the property of access control in a hierarchy, it's necessary for a higher security clearance to have the capability of deriving the secret key of his direct or indirect child nodes from his own secret key to access the information. The way of key derivation can be divided into the following three situations:

(1) For root node (namely the node which has no direct parent nodes). The secret key of the node is arbitrarily assigned by the CA, and cannot be derived by anyone.

(2) For the node which has only one direct parent node. Suppose node r_i is the only direct parent of node r_j ($r_j \le r_i$), and r_j is the $l_{i,j}$ th child node of r_i (from left to right). It can be shown that the secret key of r_i is K_i, and the secret key of r_j can be derived from $K_j = H_{l_{i,j}}(K_i)$.

(3) For the node which has more than one direct parent nodes. Suppose node r_j has *m* direct parent nodes $(r_j^1, r_j^2, ..., r_j^m)$, where $r_j \le r_j^i$, for *i* = 1, 2, …, *m*, and r_j is the $l_{i,j}$ th child node of r_j^i. The secret keys of $(r_j^1, r_j^2, ..., r_j^m)$ are $(K_1, K_2, ..., K_m)$, respectively. The parent node r_j^i must

share the other parameters $H_{l_t}(K_t)$, for t = 1, 2, …, m and $t \neq i$, to derive the secret key of r_j by calculating

$$H_{l_i}(H_{l_1}(K_1), H_{l_2}(K_2),..., H_{l_m}(K_m)).$$

Given a hierarchy with eight nodes (A,B,…,H) in the Figure 1, the secret key of each node is assigned by the CA first to be $(K_A, K_B,..., K_H)$, and

(1) The secret key of node A is K_A.

(2) Node B can derive the secret key of node E by computing $H_2(K_B)$, such that $K_E = H_2(K_B)$. Hence, $K_B = H_1(K_A)$ and $K_C = H_2(K_A)$ on this account.

(3) Node B or node C can derive the secret key of node F K_F by computing $K_F = H_3(H_3(K_B), H_1(K_C))$, if nodes B and C know $H_1(K_C)$ and $H_3(K_B)$, respectively.

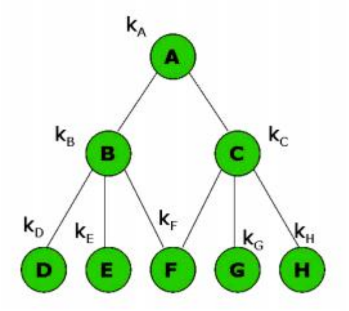

Figure 1 Key derivation in the hierarchy

3. Attacks and Improvements on Yang and Li's scheme

Yang and Li [YL04] claimed that their scheme can resolve the dynamic access control problem when adding, deleting, or changing the relationship between nodes. Only parts of nodes instead of all nodes need to renew secret keys. Unfortunately, some flaws are found in the situation that a node has more than one direct parent nodes. We point out that someone may overstep his authority to access the unauthorized information in the two cases of adding a node, or adding a new node after deleting another, below.

(1) ***In the case of adding a node.*** Node F in Figure 1 has two direct parent nodes B and C, hence the information of $H_1(K_C)$ and $H_3(K_B)$ is held by node B and node C, respectively. When adding a new node Q, as shown in Figure 2, the secret keys of node Q, D, E, and F are $K_Q = H_1(K_B)$, $K_D = H_2(K_B)$, $K_E = H_3(K_B)$, and $K_F = H_4(K_B)$, respectively. At this time, node C will hold $H_4(K_B)$ and $H_3(K_B)$. He can derive the secret key of node E by the information of $H_3(K_B)$. The security of node E is threatened.

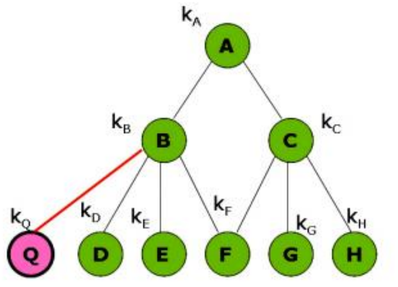

Figure 2 Adding a new node Q

(2) **In the case of adding a new node after deleting another.** Node C in Figure 3 originally holds $H_3(K_B)$. However, after deleting node E, node C will hold $H_2(K_B)$ and $H_3(K_B)$, and the secret key of node F will be $K_F = H_2(H_2(K_B), H_1(K_C))$. After that, if a new node Q is added, node C will be able to derive K_Q form $H_2(K_B)$ to access the unauthorized information.

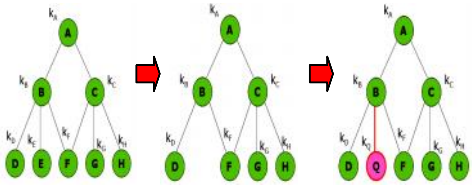

Figure 3 Adding a new node Q after deleting a node E

According to the above weakness in Yang and Li's scheme, some improvements are given as follows. Suppose ID_i and K_i be the identity and the secret key of each node, respectively, for *i*= {A, B, …, H} as shown in Figure 1. Comparing with Yang and Li's scheme, only an own-way hash function H is used here. Three scenarios of key generation ways are modified as follow.

(1) For a node which has no direct parent nodes, the secret key is assigned by the CA (eg. The secret key of node A is K_A assigned by the CA).

(2) For a node r_j which has only one direct parent node r_i $(r_j \le r_i)$, r_i can derive the secret key of r_j by computing $K_{r_j} = H(ID_{r_j}, K_{r_i}, ID_{r_i})$. For instance, node A can derive the secret key of node B by computing $H(ID_B, K_A, ID_A)$, such that $H(ID_B, K_A, ID_A)$.

(3) For a node r_j which has more than one direct parent nodes $(r_j{}^1, r_j{}^2, ..., r_j{}^m)$, where $r_j \le r_j{}^i$, for *i* = 1, 2, …, *m*, each parent node $r_j{}^i$ can derive the secret key of node r_j by calculating $K_{r_j} = H(H(ID_{r_j}, K_{r_j^1}, ID_{r_j^1}, \ldots, ID_{r_j^m}), H(ID_{r_j}, K_{r_j^2}, ID_{r_j^1}, \ldots, ID_{r_j^m}), ..., H(ID_{r_j}, K_{r_j^m}, ID_{r_j^1}, \ldots, ID_{r_j^m}))$ with sharing $H(ID_{r_j}, K_{r_j^i}, ID_{r_j^1}, \ldots, ID_{r_j^m})$, where i = 1, 2, …, *m*, and $\forall i \neq j$. For example, node B or C can derive the K_F by computing $H(H(ID_F, K_B, ID_B, ID_C),$ $H(ID_F, K_C, ID_B, ID_C))$.

4. Security analysis

In this section, we will give some security analysis for our proposed improved scheme.

(1) The access control problem can be resolved by using a one-way hash function. That is, the parent node can derive the secret keys of his direct or indirect child nodes, while the child node cannot derive the secret key of his parent from the hash value.

(2) Consider the scenario of that a node has more than one direct parent nodes, the information each parent nodes hold are only the hash values of the child's identity and another parent's secret key. For any parent node, it's infeasible to derive the secret key of other parent nodes from the shared information.

(3) The secret key of each child node is the hash value of his own identity and the direct parent's secret key. Therefore, it's impossible for a parent node to derive the same secret keys of different child nodes. In the scenario of that a node has more than one direct parent nodes, even if adding a node or adding a new node after deleting a node, it won't violate the objective of access control.

5. Conclusions

This paper pointed out that Yang and Li's scheme violates the requirements of dynamic access control problem in the situation of that a node has more than one direct parent nodes. We further give improvements to eliminate the pointed out attacks by only using a one way hash function and bounding the child node's identity into the derived key.

References

[AT83] S. G. Akl and P. D. Taylor, "Cryptographic solution to a problem of access control in a hierarchy", ACM Transactions on Computer System, Vol. 1, No. 3, 1983, pp. 239-248.

[CHW92] C. C. Chang, R. J. Hwang, and T. C. Wu, "Cryptographic key assignment scheme for access control in a hierarchy", Information Systems, Vol. 17, No. 3, 1992, pp. 243-247.

[HL90] L. Harn and H. Y. Lin, "Cryptographic key generation scheme for multilevel data security", Computers and Security, Vol. 9, No. 6, 1990, pp. 539-546.

[KSCL99] F. H. Kuo, V. R. L. Shen, T. S. Chen, and F. Lai, "Cryptographic key assignment scheme for dynamic access control in a user hierarchy", IEE Proceedings – Computers and Digital Techniques, Vol. 146, No.5, 1999, pp. 235-240.

[MTMA85]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 Transactions on Computers, Vol. C-34, No. 9, 1985, pp. 797-802.

[WC01] T. C. Wu and C. C. Chang, "Cryptographic key assignment scheme for hierarchical access control", Computer Systems Science and Engineering, Vol. 16, No. 1, 2001, pp. 25-28.

[WWH95] T. C. Wu, T. S. Wu, and W. H. He, "Dynamic access control scheme based on the Chinese remainder theorem", Computer Systems Science and Engineering, Vol. 10, No. 2, 1995, pp. 92-99.

[YL04] C. Yang and C. Li, "Access control in a hierarchy using one-way hash functions", Computers and Security, Vol. 23, No. 8, 2004, pp. 659-664.