Unconditionally Secure Hierarchical Key Assignment Schemes

Alfredo De Santis, Anna Lisa Ferrara, and Barbara Masucci

Dipartimento di Informatica ed Applicazioni
Università di Salerno, 84081 Baronissi (SA), Italy

E-mail: {ads, ferrara, masucci}@dia.unisa.it

Abstract

We consider the access control problem in a scenario where a group of users is divided into a number of disjoint classes located at different privilege levels, in such a way that any class can access the private data of any of its descendant lower level classes, but the opposite is not allowed. This multilevel security problem arises in organizations where a hierarchical structure exists. Government, diplomacy and the military are examples of such hierarchies.

The access control problem in a hierarchy can be solved by using a *hierarchical key assignment scheme*, where a trusted central authority assigns an encryption key and some private information to each class. For each class, the encryption key is used to protect private data by means of a symmetric cryptosystem, while the private information is used to compute the key assigned to each class lower down in the hierarchy.

In this paper we propose an *information-theoretic approach* to hierarchical key assignment schemes. We consider schemes which are unconditionally secure against attacks carried out by a coalition of classes of a certain size. We show a lower bound on the amount of the private information that each class has to store and propose an optimal construction for unconditionally secure hierarchical key assignment schemes.

1 Introduction

The *access control problem* deals with the specification of users' access permission and is a fundamental issue in any system that manages distributed resources, such as e-newspaper, pay-TV subscription services, etc. The *hierarchical access control problem* is defined in a scenario where the users of a computer system are organized in a hierarchy formed by a certain number of disjoint classes, called *security classes*. A hierarchy arises from the fact that some users have more access rights than others. For example, there are several situations where supervisors have all the privileges to control the tasks of their subordinates, while the subordinates have no privileges at all to access the supervisors' tasks. Similar situations abound in other areas, particularly in the government and military.

The hierarchical access control problem can be solved by using a *hierarchical key assignment scheme*, that is, a method to assign an encryption key and some private information to each class. The encryption key will be used by each class to protect its data by means of a symmetric cryptosystem. The private information will be used by each class to compute the

keys assigned to all classes lower down in the hierarchy. This assignment is carried out by a central authority, the CA, which is active only at the distribution phase.

In a *perfectly secure* hierarchical key assignment scheme, the key assigned to each class C_i is secure against a coalition of all the classes which are not entitled to access C_i's secret data, i.e., even pooling together their private information, they cannot compute anything about that key. The basic and straightforward perfectly secure hierarchical key assignment scheme requires each class to memorize the encryption keys assigned to all classes lower down in the hierarchy. The disadvantage of this solution is that it penalizes users in high level classes requiring them to handle more information than users in low level classes. Given the high complexity of such a scheme, a natural step is to trade complexity for security. We may still require that the key assigned to each class is unconditionally secure, but only with respect to an adversary controlling a coalition of classes of a limited size.

In this paper we design and analyze hierarchical key assignment schemes secure against attacks carried out by coalitions of classes of a certain size. We prove a lower bound on the size of the private information held by each class in such schemes. For perfectly secure hierarchical key assignment schemes, we show that the basic straightforward scheme is optimal with respect to the private information distributed to each class. We also show an optimal construction for hierarchical key assignment schemes secure against attacks carried out by a single class in a rooted tree hierarchy.

1.1 Related Work

The problem of reducing the inherent complexity of the basic straightforward hierarchical key assignment scheme was first considered by Akl and Taylor [1], who proposed an elegant solution for the general problem where the hierarchy on security classes is an arbitrary partial order. In their scheme, each class is assigned a key that can be used, along with some public parameters generated by the CA, to compute the key assigned to any class lower down in the hierarchy. Subsequently, many researchers have proposed schemes that either have better performances or allow inserting and deleting classes in the hierarchy.

The most used approach to hierarchical key assignment schemes (different from the one proposed in this paper) is based on unproven specific assumptions (e.g., [1, 2, 5, 6, 7, 8, 9, 10, 11]). We remark that our approach is *information theoretic* and indeed differs from the above computational approach since it does not depend on any unproven assumption. Yet, our bounds serve as foundations for the hierarchical access control problem in general, and in particular allow us to formally prove the optimality of the basic and straightforward scheme if we allow sets of classes of any size to collude against a single class.

2 The Model

In this section we present the hierarchical access control problem. Consider a set of users divided into a number of disjoint classes, $C_1, \ldots, C_\ell$, called *security classes*. A security class can represent a person, a department, or a user group in an organization. In accordance with authority, position, or power, there is a binary relation $\leq$ that partially orders the set of classes $\mathcal{C} = \{C_1, \ldots, C_\ell\}$. The poset $(\mathcal{C}, \leq)$ is called a *partially ordered hierarchy*. For any

two distinct classes C_i and C_j, the notation $C_i \leq C_j$ is used to indicate that the users in C_j can access C_i's data. In the real world there are several examples of hierarchies where an access control is required. Applications exist in business and in other areas of the private sector, for example in the management of databases containing sensitive information or in the protection of industrial secrets. Similar situations abound in other areas, particularly in the government and military.

The partially ordered hierarchy $(\mathcal{C}, \leq)$ can be represented by a directed acyclic graph, where each class corresponds to a vertex in the graph and there is an edge from class C_j to class C_i if and only if $C_i \leq C_j$. Further, this graph can be simplified by eliminating all the edges which can be implied by the property of the transitive closure. Figure 1 shows an example of a partially ordered hierarchy.

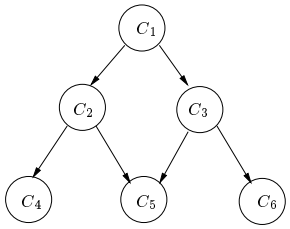

Figure 1: An example of a partially ordered hierarchy.

For any $i = 1, \ldots, \ell$, we define the *accessible set* of C_i as the set of indices corresponding to all descendant classes of C_i, including C_i itself, i.e., $A_i = \{j : C_j \leq C_i\} \cup \{i\}$. A class C_i such that $A_i = \{i\}$ is called a *leaf class*. Leaf classes correspond to the lowest security level in the hierarchy. For any $i = 1, \ldots, \ell$, we also define the *forbidden set* of C_i as the set of indices corresponding to all classes that cannot access class C_i, i.e., $F_i = \{j : C_i \not\leq C_j\}$, where the notation $C_i \not\leq C_j$ is used to indicate that the users in C_j have no access to C_i's data. For example, consider the partially ordered hierarchy of Figure 1. The accessible and forbidden sets of each class are the following:

$$\begin{array}{ll} A_1 = \{1,2,3,4,5,6\} & F_1 = \{2,3,4,5,6\} \\ A_2 = \{2,4,5\} & F_2 = \{3,4,5,6\} \\ A_3 = \{3,5,6\} & F_3 = \{2,4,5,6\} \\ A_4 = \{4\} & F_4 = \{3,5,6\} \\ A_5 = \{5\} & F_5 = \{4,6\} \\ A_6 = \{6\} & F_6 = \{2,4,5\} \end{array}$$

The hierarchical access control problem can be solved by using a *hierarchical key assignment scheme*, where a trusted third party, called the *central authority* (CA), has the task to assign a key and some private information to each class in the hierarchy. For any class C_i, we denote by p_i the private information sent by the CA to users in class C_i and by k_i the key assigned to class C_i. Moreover, we denote by P_i and K_i the sets of all possible values that p_i and k_i can assume, respectively. Given a set $X = \{i_1, i_2, \ldots, i_v\} \subseteq \{1, \ldots, \ell\}$, where $i_1 < i_2 < \ldots < i_v$, we denote by P_X and K_X the set $P_{i_1} \times \cdots \times P_{i_v}$ and $K_{i_1} \times \cdots \times K_{i_v}$, respectively.

In this paper, with a boldface capital letter, say $\mathbf{Y}$, we denote a random variable taking values on a set, denoted with the corresponding capital letter Y, according to some probability distribution $\{Pr_{\mathbf{Y}}(y)\}_{y \in Y}$. The values such a random variable can take are denoted by the corresponding lower case letter. Given a random variable $\mathbf{Y}$ we denote by $H(\mathbf{Y})$ the Shannon entropy of $\{Pr_{\mathbf{Y}}(y)\}_{y \in Y}$ (we refer the reader to [3] for a complete treatment of Information Theory).

We consider hierarchical key assignment schemes where the key assigned to each class is unconditionally secure with respect to an adversary controlling a coalition of classes of a limited size. Our schemes are characterized by a security parameter r, the size of the adversary coalition. The maximum value that the security parameter r can assume is equal to $\max_{i=1,\ldots,\ell} |F_i|$, since any adversary coalition for class C_i can contain at most $|F_i|$ classes. We formally define r-secure hierarchical key assignment schemes by using the entropy function, mainly because this leads to a compact and simple description of the schemes and because the entropy approach takes into account all probability distributions on the keys assigned to the classes. An r-secure hierarchical key assignment scheme is defined as follows.

Definition 2.1 *Let $(\mathcal{C}, \leq)$ be a partially ordered hierarchy and let $1 \leq r \leq \max_{i=1,\ldots,\ell} |F_i|$. An r-secure hierarchical key assignment scheme for $(\mathcal{C}, \leq)$ is a method to assign a key to each class in such a way that the following two properties are satisfied:*

1. Any class allowed to access another class can compute the key assigned to that class. *Formally, for any $i = 1, \ldots, \ell$, and any $j \in A_i$, it holds that*
$$H(\mathbf{K}_j | \mathbf{P}_i) = 0.$$

2. Any coalition of at most r classes not allowed to access another class have absolutely no information about the key assigned to that class.
Formally, for any $j = 1, \ldots, \ell$ and any $X \subseteq F_j$ such that $|X| \leq r$, it holds that
$$H(\mathbf{K}_j | \mathbf{P}_X) = H(\mathbf{K}_j).$$

In Section 3.1 we will consider hierarchical key assignment schemes where each key is secure against any coalition of classes having size at most $r = \max_{i=1,\ldots,\ell} |F_i|$. In the following, these schemes will be called *perfectly secure* hierarchical key assignment schemes. In Section 3.2 we will restrict our attention to 1-secure hierarchical key assignment schemes, i.e., schemes such that each key is secure against a single class trying to compute it.

3 Lower Bounds

In this section we prove lower bounds on the size of the private information held by each class in any r-secure hierarchical key assignment scheme. In order to prove our results we need the next definition.

Definition 3.1 *Let $(\mathcal{C}, \leq)$ be a partially ordered hierarchy. In any r-secure hierarchical key assignment scheme for $(\mathcal{C}, \leq)$, for any $i = 1, \ldots, \ell$, a sequence of classes $C_1 \ldots C_m$ is called* r-independent *for C_i if the following properties are satisfied:*

1. $\{1, \ldots, m\} \subseteq A_i$*;*

2. For any $j = 2, \ldots, m$*, there exists a set* $X_j \subseteq F_j$ *such that*

(a) $|X_j| \leq r$*,*

(b) $\{1, \ldots, j-1\} \subseteq \cup_{h \in X_j} A_h$*.*

For example, consider the hierarchy shown in Figure 1. It is easy to see that $C_4C_2C_1$, $C_5C_2C_1$, $C_5C_3C_1$, and $C_6C_3C_1$ are 1-independent sequences for C_1, whereas, $C_4C_5C_6C_2C_3C_1$ is a 5-independent sequence for C_1. Moreover, C_4C_2 and C_5C_2 are 1-independent sequences for C_2, whereas, $C_4C_5C_2$ is a 2-independent sequence for C_2.

The next theorem states a lower bound on the size of the private information distributed to each class in any r-secure hierarchical key assignment scheme.

Theorem 3.2 *Let* $(\mathcal{C}, \leq)$ *be a partially ordered hierarchy. In any* r*-secure hierarchical key assignment scheme for* $(\mathcal{C}, \leq)$*, for any* $i = 1, \ldots, \ell$*, if there exists an* r*-independent sequence of classes* $C_1 \ldots C_m$ *for* C_i*, then it holds that*

$$H(\mathbf{P}_i) \geq \sum_{j=1}^{m} H(\mathbf{K}_j).$$

Proof. Let C_i be a class and let $C_1 \ldots C_m$ be an r-independent sequence for C_i. From *1.* of Definition 3.1 and *1.* of Definition 2.1 it follows that $H(\mathbf{K}_j|\mathbf{P}_i) = 0$, for any $j = 1, \ldots, m$. Hence, we have that

$$H(\mathbf{K}_1 \ldots \mathbf{K}_m|\mathbf{P}_i) \leq \sum_{j=1}^{m} H(\mathbf{K}_j|\mathbf{P}_i) = 0. \tag{1}$$

Consider the mutual information $I(\mathbf{P}_i; \mathbf{K}_1 \ldots \mathbf{K}_m)$. It holds that

$$H(\mathbf{P}_i) - H(\mathbf{P}_i|\mathbf{K}_1 \ldots \mathbf{K}_m) = H(\mathbf{K}_1 \ldots \mathbf{K}_m) - H(\mathbf{K}_1 \ldots \mathbf{K}_m|\mathbf{P}_i). \tag{2}$$

Since $H(\mathbf{P}_i|\mathbf{K}_1 \ldots \mathbf{K}_m) \geq 0$, from (1) and (2) it follows that

$$H(\mathbf{P}_i) \geq H(\mathbf{K}_1 \ldots \mathbf{K}_m). \tag{3}$$

Since $C_1 \ldots C_m$ is an r-independent sequence for C_i, from *2.* of Definition 3.1 we have that, for any $j = 2, \ldots, m$, there exists a set $X_j \subseteq F_j$ such that $|X_j| \leq r$ and $\{1, \ldots, j-1\} \subseteq \cup_{h \in X_j} A_h$. Therefore, from *2.* of Definition 2.1 it holds that

$$H(\mathbf{K}_j|\mathbf{P}_{X_j}) = H(\mathbf{K}_j). \tag{4}$$

Moreover, from *1.* of Definition 2.1 we have that

$$H(\mathbf{K}_1 \ldots \mathbf{K}_{j-1}|\mathbf{P}_{X_j}) \leq \sum_{s=1}^{j-1} H(\mathbf{K}_s|\mathbf{P}_{X_j}) = 0.$$

Hence, it follows that

$$H(\mathbf{K}_1 \ldots \mathbf{K}_{j-1}|\mathbf{K}_j\mathbf{P}_{X_j}) \leq H(\mathbf{K}_1 \ldots \mathbf{K}_{j-1}|\mathbf{P}_{X_j}) = 0. \tag{5}$$

Consider the mutual information $I(\mathbf{K}_j; \mathbf{K}_1 \dots \mathbf{K}_{j-1} | \mathbf{P}_{X_j})$. It holds that

$$H(\mathbf{K}_j|\mathbf{P}_{X_j}) - H(\mathbf{K}_j|\mathbf{K}_1 \dots \mathbf{K}_{j-1}\mathbf{P}_{X_j}) = H(\mathbf{K}_1 \dots \mathbf{K}_{j-1}|\mathbf{P}_{X_j}) - H(\mathbf{K}_1 \dots \mathbf{K}_{j-1}|\mathbf{K}_j\mathbf{P}_{X_j}). \quad (6)$$

Hence, from (5) and (6) it follows that

$$H(\mathbf{K}_j|\mathbf{K}_1 \dots \mathbf{K}_{j-1}\mathbf{P}_{X_j}) = H(\mathbf{K}_j|\mathbf{P}_{X_j}). \quad (7)$$

Therefore, it holds that

$$\begin{aligned} H(\mathbf{K}_1 \dots \mathbf{K}_m) &= H(\mathbf{K}_1) + \sum_{j=2}^{m} H(\mathbf{K}_j|\mathbf{K}_1 \dots \mathbf{K}_{j-1}) \\ &\geq H(\mathbf{K}_1) + \sum_{j=2}^{m} H(\mathbf{K}_j|\mathbf{K}_1 \dots \mathbf{K}_{j-1}\mathbf{P}_{X_j}) \\ &= H(\mathbf{K}_1) + \sum_{j=2}^{m} H(\mathbf{K}_j|\mathbf{P}_{X_j}) \text{ (from (7))} \\ &= \sum_{j=1}^{m} H(\mathbf{K}_j) \text{(from (4).} \quad (8) \end{aligned}$$

Hence, the lemma follows from equations (3) and (8). □

In Definition 2.1 we did not make any assumption on the entropies of random variables $\mathbf{K}_i$ and $\mathbf{K}_j$, for different classes C_i and C_j. For example, we could have either $H(\mathbf{K}_i) > H(\mathbf{K}_j)$ or $H(\mathbf{K}_i) \leq H(\mathbf{K}_j)$. Our results apply to the general case of arbitrary entropies of keys, but for clarity we state the next result for the simpler case that all entropies of keys are equal, i.e. $H(\mathbf{K}_i) = H(\mathbf{K}_j)$ for all $i, j \in \{1, \dots, \ell\}$. We denote this common entropy by $H(\mathbf{K})$.

Corollary 3.3 *Let $(\mathcal{C}, \leq)$ be a partially ordered hierarchy. In any r-secure hierarchical key assignment scheme for $(\mathcal{C}, \leq)$, for any $i = 1, \dots, \ell$, if there exists an r-independent sequence for C_i having length m, then it holds that*

$$H(\mathbf{P}_i) \geq m \cdot H(\mathbf{K}).$$

3.1 Perfectly Secure Hierarchical Key Assignment Schemes for Partially Ordered Hierarchies

In this section we consider hierarchical key assignment schemes where each key is secure against any coalition of classes having size at most $r = \max_{i=1,\dots,\ell} |F_i|$. These schemes are called *perfectly secure hierarchical key assignment schemes*. We need the next definition.

Definition 3.4 *Let $(\mathcal{C}, \leq)$ be a partially ordered hierarchy. In any perfectly secure hierarchical key assignment scheme for $(\mathcal{C}, \leq)$, for any $i = 1, \dots, \ell$, any r-independent sequence of classes for C_i is called an* independent sequence *for C_i.*

For example, consider the hierarchy shown in Figure 1. It is easy to see that $C_4C_5C_6C_2C_3C_1$ is an independent sequence for C_1. The next lemma shows how to construct an independent sequence for each class C_i.

Lemma 3.5 *Let $(\mathcal{C}, \leq)$ be a partially ordered hierarchy. In any perfectly secure hierarchical key assignment scheme for $(\mathcal{C}, \leq)$, for any $i = 1, \ldots, \ell$, there exists an independent sequence of classes for C_i, whose length is $|A_i|$.*

Proof. Let C_i be a class. We show how to construct an r-independent sequence of classes for C_i having length $|A_i|$, where $r = \max_{i=1,\ldots,\ell} |F_i|$. Let G be the direct acyclic graph corresponding to classes whose indices belong to A_i and let $C_{|A_i|}C_{|A_i|-1} \cdots C_2 C_1$ be the sequence of classes output by the topological sorting on G. This sequence has the property that for each edge (C_s, C_t) in G, i.e., such that $C_t \leq C_s$, the class C_s appears before than C_t in the ordering. It is easy to see that $C_1 C_2 \ldots C_{|A_i|}$ is an independent sequence for C_i. Indeed, $\{1, \ldots, |A_i|\} = A_i$ and for any $j = 2, \ldots, |A_i|$, the set $X_j = F_j$ satisfies Property *2.* of Definition 3.1. □

The next theorem easily follows from the previous lemma and from Corollary 3.3.

Theorem 3.6 *Let $(\mathcal{C}, \leq)$ be a partially ordered hierarchy. In any perfectly secure hierarchical key assignment scheme for $(\mathcal{C}, \leq)$, for any $i = 1, \ldots, \ell$, it holds that*

$$H(\mathbf{P}_i) \geq |A_i| \cdot H(\mathbf{K}).$$

Hence, each class C_i has to store a private information p_i whose size is lower bounded by the sum of the sizes of the keys assigned to all classes whose indices belong to the accessible set A_i. This bound is tight. Indeed, the following basic and straightforward perfectly secure hierarchical key assignment scheme meets it with equality.

Initialization phase

- The CA chooses a large prime number q.

Key generation phase

- For each class C_j, the CA randomly chooses a key $k_j \in Z_q$.

Information distribution phase

- The CA sends the key k_j to any class C_i such that $j \in A_i$, over a private channel.

Figure 2: A perfectly secure hierarchical key assignment scheme.

For example, consider the hierarchy of Figure 1. The basic hierarchical key assignment scheme distributes information as follows:

C_1 gets $(k_1, k_2, k_3, k_4, k_5, k_6)$	C_2 gets (k_2, k_4, k_5)	C_3 gets (k_3, k_5, k_6)
C_4 gets (k_4)	C_5 gets (k_5)	C_6 gets (k_6)

It is easy to see that the scheme of Figure 2 satisfies Definition 2.1. Indeed, each class C_i gets the key k_j assigned to any class C_j such that $j \in A_i$, so Property *1.* is satisfied. As for Property *2.*, since all keys are independently chosen by the CA, any $|F_i|$ keys do not have any information about the key k_i assigned to class C_i. Finally, the scheme of Figure 2 meets the bound of Theorem 3.6. Indeed, the users in class C_i receive exactly $|A_i|$ keys, and the total amount of this information is equal to $|A_i| \log q$. Hence, the protocol is optimal with respect to the size of the information distributed to users.

3.2 Security against a Single Class for Partially Ordered Hierarchies

In the previous section we have shown the optimality of the basic and straightforward perfectly secure hierarchical key assignment scheme. Given the high complexity of such an assignment mechanism, a natural step is to trade complexity for security. Hence, in this section we consider the lowest level of security, i.e., we restrict our attention to hierarchical assignment schemes where each key is secure only against a single class in its forbidden set.

In the following, a sequence of m classes $C_1C_2 \dots C_{m-1}C_m$ such that $C_m \leq C_{m-1} \leq \dots \leq C_2 \leq C_1$ will be called a *path* of length m. Given a class C_i, we will denote by h_i the *height* of C_i, i.e., the number of classes on the longest path from C_i to a leaf class in the hierarchy, including the class C_i itself. For example, in the hierarchy shown in Figure 1, the sequence of classes $C_1C_2C_4$ is a path of length 3. The next lemma shows how to construct a 1-independent sequence for each class C_i.

Lemma 3.7 *Let $(\mathcal{C}, \leq)$ be a partially ordered hierarchy. In any 1-secure hierarchical key assignment scheme for $(\mathcal{C}, \leq)$, for any $i = 1, \dots, \ell$, there exists a 1-independent sequence of classes for C_i whose length is h_i.*

Proof. Let C_i be a class at height h_i. Hence, there exists a path of length h_i starting in C_i. W.l.o.g, let $C_iC_{i+1} \dots C_{i+h_i-1}$ be such a path. It is easy to see that $C_{i+h_i-1} \dots C_{i+1}C_i$ is a 1-independent sequence for C_i. □

The next theorem easily follows from the previous lemma and from Corollary 3.3.

Theorem 3.8 *Let $(\mathcal{C}, \leq)$ be a partially ordered hierarchy. In any 1-secure hierarchical key assignment scheme for $(\mathcal{C}, \leq)$, for any $i = 1, \dots, \ell$, it holds that*

$$H(\mathbf{P}_i) \geq h_i \cdot H(\mathbf{K}).$$

Hence, each class has to store a private information whose size is lower bounded by the sum of the sizes of the keys assigned to the classes in the longest path from that class to a leaf class. This bound is tight. Indeed, in the following we show a hierarchical key assignment scheme for a particular kind of partially ordered hierarchy (the tree hierarchy) that meets it with equality.

3.2.1 An Optimal Protocol for Tree Hierarchies

In this section we consider an important kind of partially ordered hierarchy: the rooted tree hierarchy. The case of a rooted tree hierarchy was also considered by Sandhu [10] in the computationally secure setting. In particular, Sandhu proposed a key assignment scheme where each user holds exactly one key corresponding to its class and all users hold keys having the same size. The key held by a class can be used to derive the keys of all descendant classes. The bound of Theorem 3.8 shows that in the unconditionally secure setting we need to distribute more information to each class, even if we require security only against a single class.

In Figure 3 we show a 1-secure key assignment scheme for a rooted tree hierarchy with maximum degree g and height h. In our scheme all keys assigned to classes have the same

size, while the amount of private information distributed to each class depends on its height. Our scheme satisfies Properties *1.* and *2.* of Definition 2.1 (due to space constraints, the proof is omitted and can be found in [4]). Moreover, the scheme is optimal with respect to the private information held by each class. Indeed, each class C_i, at height h_i, receives the $h_i - 1$ values $(x_1, \ldots, x_{h_i-1})$ and its key k_i from the CA. Hence, the size of the information distributed to class C_i is equal to $h_i \log q$.

Initialization phase

- The CA chooses a large prime number $q > h$.
- The CA randomly chooses g pairs of integers $(a_i, b_i) \in Z_q^* \times Z_q^*$, for $i = 1, \ldots, g$, such that the corresponding vectors are linearly independent. These pairs of integers are made public.
- Then, the CA randomly chooses h integers $(x_1, x_2, \ldots, x_h)$ in Z_q.

Key generation phase

- If C_i is the root class, then $k_i = x_h$.
- Let C_i be a class at height h_i and let k_i be the key assigned by the CA to class C_i. Assume that the class C_i has $g_i \leq g$ children. W.l.o.g., let $C_{i_1}, \ldots, C_{i_{g_i}}$ be its children. For any $t = 1, \ldots, g_i$, the key for class C_{i_t} is computed by the CA as follows:

$$k_{i_t} = a_t k_i + b_t x_{h_{i_t}} \bmod q.$$

Information distribution phase

- Let C_i be a class at height $h_i > 1$. The CA sends the $h_i - 1$ values $(x_1, \ldots, x_{h_i-1})$ to C_i over a private channel. These values will be used by class C_i to compute the keys for all classes in its accessible set. Moreover, the CA sends to C_i the key k_i, computed in the key generation phase.
- If C_i is a leaf class, i.e., $h_i = 1$, then the CA sends to C_i only its key k_i.

Figure 3: A 1-secure hierarchical key assignment scheme for a rooted tree hierarchy.

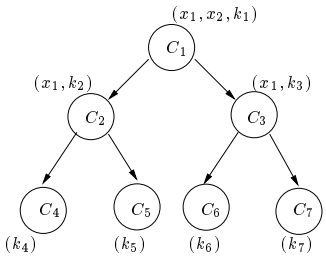

Figure 4: Information distributed by the 1-secure scheme of Figure 3.

Figure 4 shows an example of our scheme. The keys assigned to the classes in the rooted tree of Figure 4 are computed as follows:

$$k_1 = x_3 \qquad k_2 = a_1 k_1 + b_1 x_2 \bmod q \qquad k_3 = a_2 k_1 + b_2 x_2 \bmod q$$
$$k_4 = a_1 k_2 + b_1 x_1 \bmod q \qquad k_5 = a_2 k_2 + b_2 x_1 \bmod q \qquad k_6 = a_1 k_3 + b_1 x_1 \bmod q$$
$$k_7 = a_2 k_3 + b_2 x_1 \bmod q$$

Conclusions

In this paper we have proposed an *information-theoretic approach* to the access control problem in a hierarchy. Our approach does not depend on any specific unproven assumption. We have considered hierarchical key assignment schemes which are unconditionally secure against attacks carried out by a coalition of classes of a certain size. We have shown lower bounds on the size of the private information held by each class and have proposed some optimal constructions for unconditionally secure hierarchical key assignment schemes.

The same approach is extended in [4] to analyze key assignment schemes for any arbitrary access control policy (i.e., where the set of classes is not necessarily a poset). There we also show new bounds on the size of the information kept secret by each class and on the number of random bits needed to set up a key assignment scheme. Moreover, we propose some optimal constructions for such schemes.

References

[1] S. G. Akl and P. D. Taylor, *Cryptographic Solution to a Problem of Access Control in a Hierarchy*, ACM Trans. on Comp. Sys., 1(3): 239–248, 1983.

[2] C. C. Chang, R. J. Hwang, and T. C. Wu, *Cryptographic Key Assignment Scheme for Access Control in a Hierarchy*, Information Systems, 17(3): 243–247, 1992.

[3] T. M. Cover and J. A. Thomas, *"Elements of Information Theory"*, John Wiley & Sons, 1991.

[4] A. De Santis, A. L. Ferrara, and B. Masucci, *Unconditionally Secure Key Assignment Schemes for Any Access Control Policy*, preprint.

[5] L. Harn and H. Y. Lin, *A Cryptographic Key Generation Scheme for Multilevel Data Security*, Computers and Security, 9(6): 539–546, 1990.

[6] M. S. Hwang, *A Cryptographic Key Assignment Scheme in a Hierarchy for Access Control*, Math. and Comput. Modeling, 26(1): 27–31, 1997.

[7] H. T. Liaw, S. J. Wang, and C. L. Lei, *A Dynamic Cryptographic Key Assignment Scheme in a Tree Structure*, Comp. and Math. Appl., 25(6): 109–114, 1993.

[8] C. H. Lin, *Dynamic Key Management Schemes for Access Control in a Hierarchy*, Computer Communications, 20: 1381–1385, 1997.

[9] 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. on Computers, C-34(9): 797–802, 1985.

[10] R. S. Sandhu, *Cryptographic Implementation of a Tree Hierarchy for Access Control*, Information Processing Letters, 27: 95–98, 1988.

[11] V. R. L. Shen, T. S. Chen, and F. Lai, *Novel Cryptographic Key Assignment Scheme for Dynamic Access Control in a Hierarchy*, IEICE Trans. on Fundamentals, E80-A(10): 2035–2037, 1997.