효율적인 회의용 키분배시스템을 위한 Block Design의 응용

이 태 훈† · 정 일 용††

요 약

회의용 키분배 시스템은 회의용 키를 생성하여 키를 회의에 참석하고 있는 사람에게만 전달하여 서로간에 안전하게 통신하도록 한다. 본 논문에서는 Block Design의 한 분류인 symmetric balanced incomplete block design(SBIBD)를 적용한 효율적인 회의용 키분배시스템을 제안한다. 회의용 키를 생성하고 개인식별 정보를 근거로 하여 인증을 수행하는 기능을 이용하여 통신 프로토콜이 설계된다. 제안된 프로토콜은 회의용 키를 생성하는 메시지의 복잡도를 최소화시키는데, SBIBD의 특별한 분류에서는 참석자의 수 v에 따라 메시지 복잡도는 $O(v\sqrt{v})$가 된다. 보안시스템의 구현에서 중요한 요소인 프로토콜의 안전성은 factoring과 discrete logarithm을 계산할 정도로 난해하여 충분히 보장됨을 증명할 수 있다.

Application of Block Design for an Efficient Conference Key Distribution System

Tae-Hoon Lee† · Il-Yong Chung††

ABSTRACT

A conference key distribution system is a scheme to generate a conference key, and then to distribute this key to only paticipants attending at the conference in order to communicate with each other securely. In this paper, an efficient conference key distribution system is presented by employing a symmetric balanced incomplete block design (SBIBD), one class of block designs. Through techniques for creating a conference key and for performing authentication based on identification information, the communication protocol is designed. The protocol presented minimizes the message complexity for generating a conference key. In a special class of SBIBD the message complexity is $O(v\sqrt{v})$, where v is the number of participants. The security of the protocol, which is a significant factor in the construction of secure system, can be proved as computationally difficult to calculate as factoring and discrete logarithms.

키워드 : 회의용 키분배 시스템(conference key distribution system), 블록 디자인(block design), 인증(authentication)

1. Introduction

A conference key distribution system (CKDS) [1] is a scheme to generate a common secret key, called a conference key, for two or more users. In this paper, we present an efficient conference key distribution system. In this paper, identity-based conference key distribution system (CKDS) is presented, in which messages among users are authenticated using each user's identification information. To do authentication [2] is the most important of the security services, because all other security services depend upon it. It is the means of gaining confidence that people or things are who or what they claim to be.

An important CKDS system considering authentication was proposed by Shamir [3], where he utilizes ID-based public key system. User's public key contains user's name and address. Shamir and Fiat [4] suggested an authentication mechanism employing discrete logarithm. Okamoto [5] proposed identity-based key distribution system. Ingemarssory, Tang and Wang [6] presented a CKDS on ring network. Koyama and Ohta [7] proposed Identity-based CKDS (ICKDS) on ring network, complete graph and star network. Shimbo

※ 이 논문은 정보통신부 대학정보통신 연구센터 지원사업의 지원 및 한국소프트웨어진흥원의 관리로 수행되었음.
† 종신회원 : 광주대학교 컴퓨터전자통신공학부 교수
†† 종신회원 : 조선대학교 컴퓨터공학부 교수
논문접수 : 2001년 4월 16일, 심사완료 : 2001년 6월 2일

and Kawamura [8] analyzed several CKDS's.

In case that ICKDS is performed on complete graph. In order for all participants (users) to communicate mutually, a conference key should be generated. We assume that each user has his own key and a conference key is designed by using these keys. One possible manner in which this generation may be carried out is by requiring each user to send its own key to every other user. The relevant computation may then be performed at every site. This method requires $v\times(v-1)$ messages [6] (where v is the number of users in the network) to be sent and one round of message exchange. The conference key is computed as $r_1\times r_2\times\cdots\times r_v$, where r_i is user i's secret key. However, as v increases, the message overhead requires $O(v^2)$ and it causes the conference to be delayed.

In this paper, we present efficient conference key distribution system. To accomplish this, (v, k, λ)-configuration method, one class of block designs [9], is applied for generating the conference key and then this key is distributed to participants. Through this technique for creating a conference key and mutual authentications performed based on identification information, the communication protocol is designed.

The protocol presented minimizes the message overhead for generating a conference key. Especially, in case of $\lambda=1$, the overhead is $O(v\sqrt{v})$, but needs two rounds of message exchange, where v is the number of participants. The security of the mechanism, which is a significant problem in the construction of secure system, can be proved as computationally difficult to calculate as factoring and discrete logarithms.

This paper is organized as follows. In the next section, we introduce a block design and state the theorems necessary for our presentation. The communication protocol that generates a conference key based on (v, k, λ) - configuration and distributes all the users is discussed in Section 3. The communication scheme considers the sites of distributed systems as constituting the blocks in a block design. This paper concludes with Section 4.

2. Block Design

In this paper, codewords are generated by employing a block design among methods of generation of error - correcting code. By a block design we mean a selection of the subsets of a given set such that some prescribed conditions are satisfied. In some designs, the elements in each of the subsets are also to be ordered in a certain way. A balanced incomplete block design (BIBD) is defined below.

Definition 1 : Let $X = \{x_1, x_2, \cdots, x_v\}$ be a set of v objects. A balanced incomplete block design of X is a collection of b k-subsets of X (the k-subsets denoted by $B_1, B_2, \cdots, B_b$) such that the following conditions are satisfied :

1. Each object appears in exactly r of the b blocks.
2. Every two objects appears simultaneously in exactly λ of the b blocks.
3. $k < v$.

For example, if $B_1=\{x_1,x_2,x_3\}$, $B_2=\{x_4,x_5,x_6\}$, $B_3=\{x_7, x_8,x_9\}$, $B_4=\{x_1,x_4,x_7\}$, $B_5=\{x_2,x_5,x_8\}$, $B_6=\{x_3,x_6,x_9\}$, $B_7=\{x_1,x_5,x_9\}$, $B_8=\{x_2,x_6,x_7\}$, $B_9=\{x_3,x_4,x_8\}$, $B_{10}=\{x_1,x_6,x_8\}$, $B_{11}=\{x_2,x_4,x_9\}$, $B_{12}=\{x_3,x_5,x_7\}$, then $X=\{x_1,x_2,\cdots,x_9\}$, $b=12$, $v=9$, $r=4$, $k=3$, $\lambda=1$. Since a BIBD is characterized by the five parameters b,v,r,k and λ, it is called a (b,v,r,k,λ) - configuration. It is clear that all five of the parameters are not independent. In other words, it is not true that there exists a BIBD for any arbitrary set of these parameters. However, there is no known sufficient condition on the existence of a certain (b,v,r,k,λ) - configuration. We shall show some relations among the parameters that are necessary conditions for the existence of a corresponding (b,v,r,k,λ) - configuration. The proof is shown in [9].

Theorem 1 : In a balanced incomplete block design, $bk = vr$, and $r(k-1)=\lambda(v-1)$.

Instead of a list of the k-subsets, a BIBD can be described by the incidence matrix Q, which is a (b X v) matrix with 0's and 1's as entries. The rows and columns of the matrix correspond to the blocks and the objects, respectively. The entry in the *i*th row and the *j*th column of Q is a1 if the block B_i contains the object x_j and is a 0 otherwise. The incidence matrix of the BIBD in previous example is described below.

In some case of a balanced incomplete block design, the number of blocks is the same as that of objects. A special class of BIBD is defined below.

Definition 2 : A balanced incomplete block design is said to be a symmetric balanced incomplete bl-

ock design(SBIBD) if b = v and r = k

$$Q = \begin{bmatrix} 1&1&1&0&0&0&0&0&0 \\ 0&0&0&1&1&1&0&0&0 \\ 0&0&0&0&0&0&1&1&1 \\ 1&0&0&1&0&0&1&0&0 \\ 0&1&0&0&1&0&0&1&0 \\ 0&0&1&0&0&1&0&0&1 \\ 1&0&0&0&1&0&0&0&1 \\ 0&1&0&0&0&1&1&0&0 \\ 0&0&1&1&0&0&0&1&0 \\ 1&0&0&0&0&1&0&1&0 \\ 0&1&0&1&0&0&0&0&1 \\ 0&0&1&0&1&0&1&0&0 \end{bmatrix}$$

(Fig. 1) (12 x 9) incidence matrix

From the definition, arbitrary two blocks in a SBIDB contains common λ elements. It can be also represented as (v,k, λ)-configuration and satisfies conditions described in Theorem 1. In case that $B_1 = \{x_1,x_2,x_4,x_7,x_{11}\}$, $B_2 = \{x_1,x_2,x_3,x_5,x_8\}$, $B_3 = \{x_2,x_3,x_4,x_6,x_9\}$, $B_4 = \{x_3,x_4,x_5,x_7,x_{10}\}$, $B_5 = \{x_4,x_5,x_6,x_8,x_{11}\}$, $B_6 = \{x_5,x_6,x_7,x_9,x_1\}$, $B_7 = \{x_6,x_7,x_8,x_{10},x_2\}$, $B_8 = \{x_7,x_8,x_9,x_{11},x_3\}$, $B_9 = \{x_8,x_9,x_{10},x_1,x_4\}$, $B_{10} = \{x_9,x_{10},x_{11},x_2,x_5\}$, $B_{11} = \{x_{10},x_{11},x_1,x_3,x_6\}$. Then it becomes (11,5,2)-configuration. A BIBD can be easily derived from the corresponding SBIBD through the intersection of two blocks (B_1,B_i) or the difference of two blocks (B_1,B_i).

Even if a symmetric balanced incomplete block design exists only for certain values of v, normalized Hadamard matrix is utilized for constructing this design of v = 4n-1. Especially, the protocol requires only $O(v\sqrt{v})$ messages based on finite projective planes, which leads to (k(k-1)+1, k,1) - configuration [12].

3. The design of a conference key distribution system based on symmetric balanced incomplete block design

3.1 Construction of a Conference Key

In order for v participants to communicate mutually, the conference key should be created by utilizing their own keys. The minimal message transmission overhead for this process must be guaranteed. In this paper, the ionic property of error-correcting code is applied and the minimal message overhead requisite to generate this key is maintained. The error-correcting coding method finds out a coset the codeword belongs to, and takes the original value even if a codeword has some errors that can be recoverable. We now apply this concept to the decentralized routing algorithm. Block i and object j correspond to participant i and key j, respectively and the number of blocks is the same as that of participants.

For example, seven users take part in conference and each has his own secret key. Each participant at conference computes a conference key based on (7,4,2) - configuration. (7 x 7) incidence matrix is now designed below.

$$Q = \begin{bmatrix} 1&1&0&1&0&0&1 \\ 1&1&1&0&1&0&0 \\ 0&1&1&1&0&1&0 \\ 0&0&1&1&1&0&1 \\ 1&0&0&1&1&1&0 \\ 0&1&0&0&1&1&1 \\ 1&0&1&0&0&1&1 \end{bmatrix}$$

(Fig. 2) (7 x 7) incidence matrix

In order to generate a conference key, each receives some keys from users chosen by employing the structure of this matrix. In this paper, two steps are required to calculate the key. User i receives key r_j from user j in case of $Q_{ij} = 1$. We now describe this process from the viewpoint of user 1. First, user 1 receives keys r_2, r_4, r_7 and then make $k_{11} = r_2 \times r_4 \times r_7$, $k_{12} = r_1 \times r_4 \times r_7$, $k_{14} = r_1 \times r_2 \times r_7$, $k_{17} = r_1 \times r_2 \times r_4$, where k_{1j} is the product of r_a's, $a \in \{1,2,4,7\}-\{j\}$. Simultaneously, other users do the same process. Next, user i receives k_{j1} from user j, if $Q_{j1} = 1$. User 1 receives k_{21}, k_{51}, k_{71} from users 2,5,7. Then the conference key K is calculated as $r_1^2 \times (k_{11} \times k_{21} \times k_{51} \times k_{71})$.

Theorem 2 : For user i, the conference key based on (v,k, λ)- configuration is computed as below.

$$K = r_i^{\lambda} \times \left(\prod_{Q_{ji}=1} k_{ji} \right)$$

Proof : According to the definition of (v,k, λ)-configuration, each row of (v x v) incidence matrix consists olf k 1's, as does each column. In order for all users to communicate mutually, the conference key should be composed of these secret keys $r_1, r_2, \cdots, r_v$. This key can be obtained by performing the following two steps. On the first step, user i receives (k-1) keys and computes k products, each of which is composed of (k-1) distinguished keys. On the second step, user i receives (k-1) products consisting of (k-1) keys again and collects k_{ji}. Then, the number of keys containing in collected products is k(k-1). Applying $k(k-1) = \lambda(v-1)$ described in Theorem 1, k(k-1) keys are composed of λ r_j's except his own secret key r_i. Therefore, user i can obtain a confer-

ence key by multiplying the product of (k^2-k) keys by r_i^{λ}.

The sequence of processes for calculating the conference key based on (7,4,2)-configuration is shown (Fig. 3).

User ID	Step 1	Step 2
1	$k_{11}=r_2\times r_4\times r_7$, $k_{12}=r_1\times r_4\times r_7$, $k_{14}=r_1\times r_2\times r_7$, $k_{17}=r_1\times r_2\times r_4$	$r_1^2\times(k_{11}\times k_{21}\times k_{31}\times k_{71})$
2	$k_{22}=r_1\times r_3\times r_5$, $k_{21}=r_2\times r_3\times r_5$, $k_{23}=r_2\times r_5\times r_1$, $k_{25}=r_2\times r_3\times r_1$	$r_2^2\times(k_{22}\times k_{12}\times k_{32}\times k_{62})$
3	$k_{33}=r_2\times r_4\times r_6$, $k_{34}=r_2\times r_3\times r_6$, $k_{36}=r_2\times r_4\times r_3$, $k_{32}=r_3\times r_4\times r_6$	$r_3^2\times(k_{33}\times k_{23}\times k_{43}\times k_{73})$
4	$k_{44}=r_3\times r_5\times r_7$, $k_{45}=r_3\times r_4\times r_7$, $k_{47}=r_3\times r_5\times r_4$, $k_{43}=r_4\times r_5\times r_7$	$r_4^2\times(k_{44}\times k_{14}\times k_{34}\times k_{54})$
5	$k_{55}=r_4\times r_6\times r_1$, $k_{56}=r_4\times r_5\times r_1$, $k_{51}=r_4\times r_6\times r_5$, $k_{54}=r_5\times r_6\times r_1$	$r_5^2\times(k_{55}\times k_{25}\times k_{45}\times k_{65})$
6	$k_{66}=r_2\times r_5\times r_7$, $k_{67}=r_2\times r_5\times r_6$, $k_{62}=r_6\times r_5\times r_7$, $k_{65}=r_2\times r_6\times r_7$	$r_6^2\times(k_{66}\times k_{36}\times k_{56}\times k_{76})$
7	$k_{77}=r_1\times r_3\times r_6$, $k_{71}=r_7\times r_3\times r_6$, $k_{73}=r_1\times r_7\times r_6$, $k_{76}=r_1\times r_3\times r_7$,	$r_7^2\times(k_{77}\times k_{17}\times k_{47}\times k_{67})$

(Fig. 3) Two steps for designing a conference key based on (7,4,2)-configuration

Theorem 3 : The communication complexity of computing a conference key based on (v,k,λ)-configuration is $O(v\sqrt{v})$, if $\lambda=1$.

Proof : According to Step 1 in (Fig. 3), each user i receives (k-1) keys to generate intermediate keys k_{ij}, $j\in(Q_{ij}=1)$. Then the complexity is $v\times(k-1)$. The process to compute a conference key in the second step is the same as that of Step1. Therefore, the total communication complexity of computing a conference key is $O(v\sqrt{v})$ in case of $\lambda=1$.

3.2 The design of a conference key distribution system providing authentication service

Even a conference key is constructed, we can not guarantee whether the key received from other user is right, which is needed for generating a conference key. To solve this problem, we utilizes user's identity information for authentication. Then a system in the network performs the following steps for creating a secret information.

① A system chooses p,q and computes $n=p\times q$, where p,q are primes and approximately 100 digits each.

② A relatively large integer e is selected so that e is relatively prime to $(p-1)\times(q-1)$ and d is calculated below

$$e\times d \equiv 1 \bmod (p-1)\times(q-1)$$

③ Obtain g, which belongs to GF(p) and GF(q).

④ Compute secret information S_i by employing user i's information ID_i.

$$S_i = ID_i^d$$

A system distributes e_i,g,n and user i keeps d_i,S_i secret. In order to authenticate user entity and to generate a conference key, we define some notations. " $i \rightarrow j : M$" indicates that user i transmits information M to user j. " $i :$ " describes that user i stays at his site and does something like verification or computation. We now present the communication protocol below.

1. $i \rightarrow j : (ID_i, (X_i)^{e_j}, Y_i, t_i)$

$$X_i = g^{e_i \times r_i} \bmod n,\quad Y_i = S_i \times g^{C_{i_1}\times r_i} \bmod n,$$

where $C_{i_1} = h(X_i, t_i)$ and $j \in B_i$

User i belonging to block j creates two information X_i and Y_i for authentication, encrypts X_i with e_j and send $(ID_i, (X_i)^{e_j}, Y_i, t_i)$ to user j, where h is a hashing function all the users take in common.

2. j : $X_i = ((X_i)^{e_j})^{d_j}$, $ID_i = Y_i^e / X_i^{C_{i_2}}$,

where $C_{i_2} = h(X_i, t_i)$

By employing a hashing function and information received from user i, user j authenticates counterpart's entity, if $ID_i = Y_i^e / X_i^{C_{i_2}}$, then the claim is legitimate.

3. $j \rightarrow p : (ID_j, (X_{jp})^{e_p}, Y_{jp}, t_j)$

$$X_{jp} = X_{p_1} \times X_{p_2} \cdots \times X_{p_{(k-1)}},\ \textit{where}\ p_i \in B_j - p$$

$$Y_{jp} = S_j \times g^{C_{j_1}\times r_j} \bmod n,\ \textit{where}\ C_{j_1} = h(X_{jp}, t_j)$$

User j collects information transmitted from users belonging to block j, computes X_{jp} and Y_{jp}, and send $(ID_j, (X_{jp})^{e_p}, Y_{jp}, t_j)$ to user p.

4. p : $X_{jp} = ((X_{jp})^{e_p})^{d_p})$, $ID_j = Y_{jp}^e / X_{jp}^{C_{j_2}}$,

where $C_{j_2} = h(X_{jp}, t_j)$

User p authenticates user j's entity by using information obtained from user j, if $ID_j = Y_{jp}^e / X_{jp}^{C_{j_2}}$, authentication process is succeeded.

Theorem 3 : If $ID_i = Y_i^e / X_i^{C_{i_1}}$, then user j gains confidence that information for generating a conference key is transmitted from user i.

Proof : $Y_i^e / X_i^{C_{i_1}} = (S_i \times g^{C_{i_1} \times r_i})^e / (g^{e \times r_i})^{C_{i_1}}$

$= S_i^e$, if $C_{i_1} = C_{i_2}$. Since $S_i = ID_i^d$, $(ID_i^d)^e$ is ID_i by Euler's Theorem.

In order to compute a conference key, user p utilizes his own secret key and X_{jp}'s transmitted from the users in block p. Since each secret key and e appear λ times and $\lambda(v-1)$ times in X_{jp}'s, respectively. Then, user p calculates a conference key below.

$$K = (X_{jp_1} \times X_{jp_2} \cdots \times X_{jp_{(k-1)}}) \times g^{e_p^d \times r_p}$$

3.3 Analysis of the proposed conference key distribution system

The communication protocol based on (v,k,λ)- configuration is now analyzed. Since the first and second steps require $v \times (k-1)$ messages each, the complexity is $O(v \times k)$ by Theorem 2. According to Theorem 1, k is determined by the values of v and λ. In case of $\lambda = 1$, k becomes approximately $\sqrt{v}$. So, the complexity is $O(v\sqrt{v})$.

Security of the protocol is now considered. In order to reveal secret information S_i, given e and n, d can not be computed since no polynomial algorithm has been found for solving factorization problem. The secret key r_i should be protected. Given X_i, to get r_i is a difficult problem because finding discrete logarithm is generally a hard problem. Therefore, security of the communication protocol is computationally secure.

4. Conclusion

An efficient identity-based conference key distribution system is developed for group communication service, on which only participants in group communicate each other. To accomplish this, (v,k,λ)- configuration method is applied for generating a conference key and then this key is distributed to participants through authentication technique. The communication protocol requires two rounds of message exchange and $O(v\sqrt{v})$ messages in case of $\lambda = 1$, compared with $O(v^2)$ messages needed for one round of message exchange.

The security of the protocol is a significant problem in the construction of secure system. In this paper, it can be proved as computationally difficult to calculate as factoring and discrete logarithms.

Reference

[1] C. Chang, T. Wu and C. Chen, "The Design of a Conference Key Distribution System," Proc. of ASIACRYPT'92, pp. 11.1-11.6, 1992.

[2] J. Seberry and J. Pieprzyk, Cryptography : An Introduction of Computer Security. Prentice-Hall, NewYork, 1988.

[3] A. Shamir "Identity-based cryptosystems and signature schemes," Proc. of Crypto'84, Lecture Notes in Computer Science No.196, Springer-Verlag, pp.47-53, 1985.

[4] A. Fiat and A. Shamir, "How to prove yourself : Practical solutions to identification and signature schemes," Proc. of Crypto'86, Lecture Notes in Computer Science No.263, Springer-Verlag, pp.186-194, 1987.

[5] T. Okamoto, "Proposal for identity-based key distribution system," Electron. Lett., No.22, pp.1283-1284, 1986.

[6] I. Ingemarsson, D. T. Tang and C. K. Wong, "A Conference Key Distribution system," IEEE Trans. on Info. Theory Vol.IT-28, pp.714-720, 1982.

[7] K. Koyama and K. Ohta, "Identity-Based Conference Key Distribution System," Proc. of Crypto'87, Lecture Notes In Computer Science, 1987.

[8] A. Shimbo and S. Kawamura, "Cryptoanalysis of several Conference Key Distribution Schemes," Proc. of ASIACRYPT'91, pp.155-160, 1991.

[9] C. Liu, "Block Designs" in Introduction to Combinatorial Mathematics, McGraw-Hill, New York, 1968.

[10] M. Rhee, error Correcting Coding Theory, McGraw-Hill, New York, 1989.

[11] D. Welsh, Codes and Cryptography, Oxford Science Pub., Oxford, 1988.

[12] J. Ryou, "A Load Balancing Algorithm in Distributed Computing Systems," J. of Korea Info. Sci. Soc., Vol.20, No.3, pp.430-441, 1993.

[13] T. Lee, I. Chung, "The Design of Authentication Mechanism Employing the Block Design for Information Security in CORBA Environment," J. Korean Inst. of Commun. Sci., Vol.24, No.3B, pp.330-337, Mar, 1999.

이 태 훈

e-mail : thlee@hosim.kwangju.ac.kr

1982년 한국항공대학교 전자공학과 (공학사)

1984년 아주대학교 대학원 전자공학과 (공학석사)

1999년 아주대학교 대학원 전자공학과 (공학박사)

1984년~1993년 한국전자통신연구원 선임연구원

1993년~현재 광주대학교 컴퓨터전자통신공학부 부교수

2000년~현재 광주대학교 테크노파크지원센터 소장

관심분야 : 네트워크 보안, 전자상거래, 멀티미디어통신

정 일 용

e-mail : iyc@mail.chosun.ac.kr

1983년 한양대학교 공과대학 졸업 (공학사)

1987년 City University of New York 전산학과(전산학석사)

1991년 City University of New York 전산학과(전산학박사)

1991년~1994년 한국전자통신연구소 선임연구원

1994년~현재 조선대학교 컴퓨터공학부 부교수

2001년~현재 조선대학교 정보통신보안시스템 연구센터 소장

관심분야 : 네트워크 보안, 전자상거래, 분산시스템 관리, 코딩 이론, 병렬 알고리즘