Key Predistribution Systems に基づく 新しい暗号鍵管理方式の設計

松 本 勉

横浜国立大学大学院工学研究科人工環境システム学専攻
〒240 横浜市 保土ヶ谷区 常盤台 79-5 横浜国立大学工学部電子情報工学棟
電子メイル: tsutomu@mlab.dnj.ynu.ac.jp ファクシミリ: 045-338-1157

Designing Cryptographic Key Management System Based on Key Predistribution Systems

Tsutomu Matsumoto

Division of Artificial Environment Systems &
Division of Electrical and Computer Engineering
Yokohama National University
79-5, Tokiwadai, Hodogaya, Yokohama 240, Japan
Email: tsutomu@mlab.dnj.ynu.ac.jp Fax: +81-45-338-1157

あらまし:

共通鍵方式の暗号技術を多数のエンティティを含むネットワークで活用するには, 通信相手との間で簡便に鍵共有を行える鍵管理システムの確立が重要である.

複数の鍵共有サブシステムを使用することにより, 一つの鍵共有サブシステムだけを用いる場合に比べて, エンティティあたりのメモリ量を増やさずに, 鍵管理システム全体の安全性を高められる. ここで, 安全性はシステム全体を破るにはいくつの耐タンパー (内部の秘密を物理的・論理的に取り出すことが難しい; リバースエンジニアリングがしにくい) モジュールから秘密を取り出さなければならないか, で計量している.

本稿では, Key Predistribution Systems (KPS) をサブシステムとして複数個用い, 上記のアイディアに基づき開発中の鍵管理システムについて, 次の内容を中心に紹介する.

(1) 複数のサブシステムのエンティティへの配置:
ネットワークは v エンティティ を含むとし, b 個の鍵共有サブシステムを考える. 各エンティティには, r 個の耐タンパーサブモジュールを備えた 1 つのモジュールが割り当てられ, 各サブモジュールごとにそれぞれ 1 つの鍵共有サブシステムが実装されるものとする.

(2) 配置を表す結合行列の構成方法:
どの 2 つのエンティティをとっても, それらのモジュールには同じ鍵共有サブシステムを実装した耐タンパーサブモジュールが少くとも 1 つは共通にあるように鍵共有サブシステムを配置する仕方で, v が r の指数関数であり, かつ, ゲイン b/r のなるべく大きいものを, 組織的に構成する方法を紹介する.

(3) 複数のサブシステムのエンティティへの配置アルゴリズム:
エンティティの識別子が与えられたとき, そのエンティティに割り当てるべき鍵共有サブシステムの組を効率よく計算するアルゴリズムの概要を紹介する.

キーワード: 暗号, 鍵共有, 鍵管理, 結合構造, 識別子, 0-1 行列, 戦術的配置, 耐タンパーモジュール, Key Predistribution System, KPS

1 Introduction

Background. In large network systems, one of the prerequisites for efficiently utilizing confidential/ authenticated communications by common-key cryptography is to develop superior *key sharing* schemes which ensure conveniently sharing cryptographic keys among communication partners. The KPS (Key Predistribution Systems) [1] is a large class of such key sharing schemes covering several variants of concrete schemes in [2, 3, 4, 5] as well as the *linear schemes* [1] versions of which have been realized (eg.[6]). The remarkable property of KPS is that for the purpose of key sharing there are no need to send messages between the entities who will make a cryptographic communication using the obtained key. Each of the entity should just input the partner's identifier to its *tamper-resistant module* (TRM) containing a common algorithm and an individual data provided by one or plural system center(s).

Motivation. The total security of KPS would be based on lack of information, physical infeasibility, and computational infeasibility. By *completely broken* we mean a state that an attacker has gotten sufficient information to produce every individual data in the system. There exists a KPS which cannot be completely broken unless extracting m individual data from at least m TRMs. And this is proved optimal. More precisely, the ratio (minimum number of TRMs to attack)/(memory per key-bit per entity) is one. Can the ratio become larger if plural TRMs can be used by each user ? In other words, is a concurrent use of a set of key-sharing systems meaningful?

Methodology. Assigning several key-sharing systems to each entity can be interpreted as making an *incidence structure*, or equivalently, constructing a (0, 1)-matrix having certain desirable properties. Thus, combinatorics can serve a reasonable tool to apply.

We have observed in [8] that the desirable incidence structures are represented by *normal*, *cohesive*, *constant-row-weight* matrices having exponential number of rows in terms of row-dimension and greater ratio (row-dimension)/(row-weight) > 1. (Each technical term will be explained in the body of the paper.) Then we have proposed a binary operation over the set of all binary matrices and show that it is very helpful to provide concrete incidence structures desirable for our application.

We develop an efficient algorithm to calculate the contents of the row corresponding to a given index for a type of desirable incidence matrices recursively defined by the binary operation.

Organization. Section 2 describes our method using terminology of incidence structures and what are desirable for the method. Section 3 redescribes a binary operation to produce new incidence structures with preserving desirable properties for our application. Section 4 proposes an algorithm that transforms a given index to the contents of the row. Section 5 concludes the paper.

2 A Way of Using Plural Key-Sharing Systems

2.1 Primitive Key-Sharing Systems

Environment. By an *entity* we denote what processes/sends/receives information, like a person/a group of persons, an equipment/a set of devices, or, a program/a set of programs. Let $\mathcal{E} = \{\text{entity}0, \cdots, \text{entity}v-1\}$ be the set of all entities to be considered. Let $\mathcal{I}$ be a set and assume that each entity, say, entity i, has its own *identifier* (ID) ID_i ($\in \mathcal{I}$) such that entity i and entity j coincide if $ID_i = ID_j$.

Model Key-Sharing System. See Figure 1. When entity i joins the system, by a single system center or a set of plural system centers, entity i is supplied a *tamper resistant module* (TRM) which stores (after some preparatory interaction with the system center(s)) an individual data x_i ($\in \{0,1\}^{h\times m}$) and a common algorithm alg, where h and m are positive integers. The hm-bit data x_i depends on ID_i and the secret of the system center(s). Algorithm alg acts as follows. On input x_i and $ID_j \in \mathcal{I}$, alg outputs $k_{ij} = alg(x_i, ID_j)$ ($\in \{0,1\}^h$), the master-key for the pair of entities i and j, where alg and every x_i, x_j are designed so that $alg(x_a, ID_b) = alg(x_b, ID_a)$ holds for any entities a, $b \in \mathcal{E}$. The TRM is designed so that entity i can input ID_j and execute alg inside the TRM and obtain the output k_{ij} but for anybody even entity i it is physically and logically very hard to read out or infer hidden x_i correctly from the TRM. This system is a typical implementation of Key Predistribution System (KPS). Note that for the

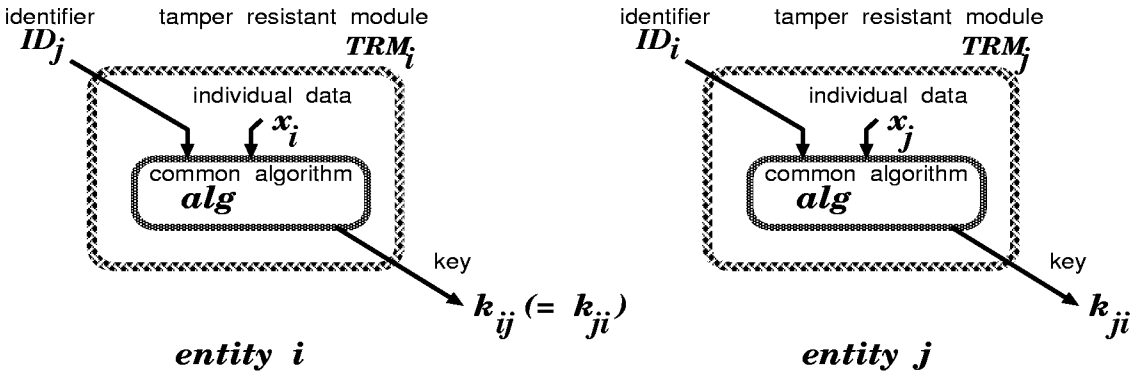

Figure 1: Primitive Key Predistibution System

purpose of key sharing there are no need to send messages between the entities. Note also that the above mentioned idea can be readily modified to cope with the key sharing among three or more entities.

Security. By *completely broken* we mean a state that an attacker has gotten sufficient information to produce every individual data in the system. And to attain such a state is called *complete breaking*. There exists such a KPS which cannot be completely broken unless extracting m individual data (x_i's) from at least m TRMs. And this is proved optimal.

2.2 Incidence Structures

Notation 1. For positive integers v and b, let $I(v,b)$ denote the set of all (0, 1)-matrices each of which has v rows and b columns.

Definition 1. A finite *incidence structure* is a triple $\mathbf{D} = (\mathcal{V}, \mathbf{B}, F)$ where $\mathcal{V}$ and $\mathbf{B}$ are any two disjoint finite sets and F is a binary relation between $\mathcal{V}$ and $\mathbf{B}$, i.e., $F \subset \mathcal{V} \times \mathbf{B}$ [7]. The elements of $\mathcal{V}$ are called *points*, those of $\mathbf{B}$ *blocks* and those of F *flags*. Let label the points as $p_0, \cdots, p_{v-1}$ and the blocks $B_0, \cdots, B_{b-1}$. Then the matrix $D = [D_{ij}]_{0 \le i < v, 0 \le j < b} \in I(v,b)$ defined by

$$D_{ij} = \begin{cases} 1 & \text{if } (p_i, B_j) \in F, \\ 0 & \text{otherwise,} \end{cases}$$

is called an *incidence matrix* for $\mathbf{D}$. D depends on the labeling used, but it is unique up to column and row permutations. Conversely, every (0, 1)-matrix determines an incidence structure.

Definition 2. Let $D = [D_{ij}] \in I(v,b)$ and $\mathbf{D}$ be the incidence structure determined by D. The i^{th} row's weight of D, or the i^{th} point's degree of $\mathbf{D}$, is defined as $R_i(D) = \sum_{j=0}^{b-1} D_{ij}$. The j^{th} column's weight of D, or j^{th} block's degree, of $\mathbf{D}$ is defined as $K_j(D) = \sum_{i=0}^{v-1} D_{ij}$. The meeting number of the i^{th} row and the j^{th} row of D, or the meeting number of the i^{th} point and the j^{th} point of $\mathbf{D}$, is defined as $\Lambda_{ij}(D) = \sum_{f=0}^{b-1} D_{if} D_{jf}$.

1) D and $\mathbf{D}$ are called *normal* if $R_i(D) < b$ and $K_j(D) < v$ for every i and j.
2) D and $\mathbf{D}$ are called *cohesive* if $\Lambda_{ij}(D) \ge 1$ for every i and j.
3) D is called *constant-row-weight* with $R(D)$ and $\mathbf{D}$ is called *constant-point-degree* with $R(D)$ if $R_i(D) = R(D)$ for every i.
4) D is called *constant-column-weight* with $K(D)$ and $\mathbf{D}$ is called *constant-block-degree* with $K(D)$ if $K_j(D) = K(D)$ for every j.
5) D and $\mathbf{D}$ are called *tactical* with $R(D)$ and $K(D)$ if $R_i(D) = R(D)$ and $K_j(D) = K(D)$ for every i and j.

Notation 2. If $D \in I(v,b)$ is tactical with $R(D) = r$ and $K(D) = k$,
we denote this fact by $D \in T(v,b,r,k)$.

Example 1. All of the following matrices are normal, cohesive, and tactical.

$$P = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \in T(3,3,2,2), \qquad Q = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \in T(4,4,3,3),$$

$$R = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix} \in T(10,5,3,6), \quad S = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix} \in T(7,7,3,3).$$

Observation 1. Given a normal cohesive tactical matrix $C \in T(v,b,r,k)$, for any integer $V \leq v$ we can construct a normal cohesive constant-row-weight matrix $D \in I(V,b)$ with $R(D) = r$ by selecting any V rows from C.

2.3 Assigning Plural Systems Using Incidence Structures

Environment. Assume a set $\mathcal{E}$ of v entities each of which (say entity i) has unique identifier ($ID_i \in \mathcal{I}$) as before and a set of b key-sharing systems $\mathbf{B} = \{B_0, \cdots, B_{b-1}\}$. Let $\mathcal{V} = \{p_0, \cdots, p_{v-1}\}$ be a set of v elements and $\mathbf{H}$ an one-way function from $\mathcal{I}$ to $\mathcal{V}$.

The Method. We propose a way of assigning each entity a subset of $\mathbf{B}$ so that any two entities have at least one common key-sharing system with which they can share an h-bit common key. This is nothing but determining a cohesive incidence structure with point set $\mathcal{V}$ and block set $\mathbf{B}$. We denote its incidence matrix by

$$D = \begin{bmatrix} D(0) \\ \vdots \\ D(v-1) \end{bmatrix} \in I(v,b)$$

with $D(i) \in I(1,b)$. We assume that there is an efficient algorithm A_D to compute a row in D given an index for the row.

Procedure. Let entity a and entity b be the entities of concern. Firstly entity a and b independently compute $i = \mathbf{H}(ID_a)$ and $j = \mathbf{H}(ID_b)$ and find $D(i) = A_D(i)$ and $D(j) = A_D(j)$. Of course entity a and entity b can record $D(i)$ and $D(j)$, respectively. Secondary, they calculate the common key-sharing system(s) indicated by both $D(i)$ and $D(j)$. See Figure 2.

Memory per Entity. If row $D(i)$ shows a subset $\{S_0, \cdots, S_{r-1}\} \subset \mathbf{B}$ then by the system centers for $S_0 \cdots, S_{r-1}$, respectively, entity i is supplied r tamper-resistant modules (TRMs) containing data depending on the entity's identifier, ID_i, and the secret information for the corresponding key-sharing system. As described in **2.1** since there is a key-sharing system which cannot be completely broken unless at least m TRMs are successfully attacked to infer m pieces of $h \times m$-bit data inside them, assume that every system in $\mathbf{B}$ has such a property. Then if D is constant-row-weight with $R(D) = r$ then each entity has r TRMs which contain $r \times h \times m$-bit data in total, while the minimum number of TRMs to attack for complete breaking is $b \times m$. Thus, the ratio (minimum number of TRMs to attack)/(total memory per key-bit per entity) is b/r. Generally we adopt the following definition.

Definition 3. For matrix $D \in I(v,b)$ and its corresponding incidence structure $\mathbf{D}$, the *gain* $\Gamma(D)$ of D and $\mathbf{D}$ is defined as

$$\Gamma(D) = \frac{b}{\frac{1}{v}\sum_i R_i(D)} = \frac{vb}{\sum_i R_i(D)}.$$

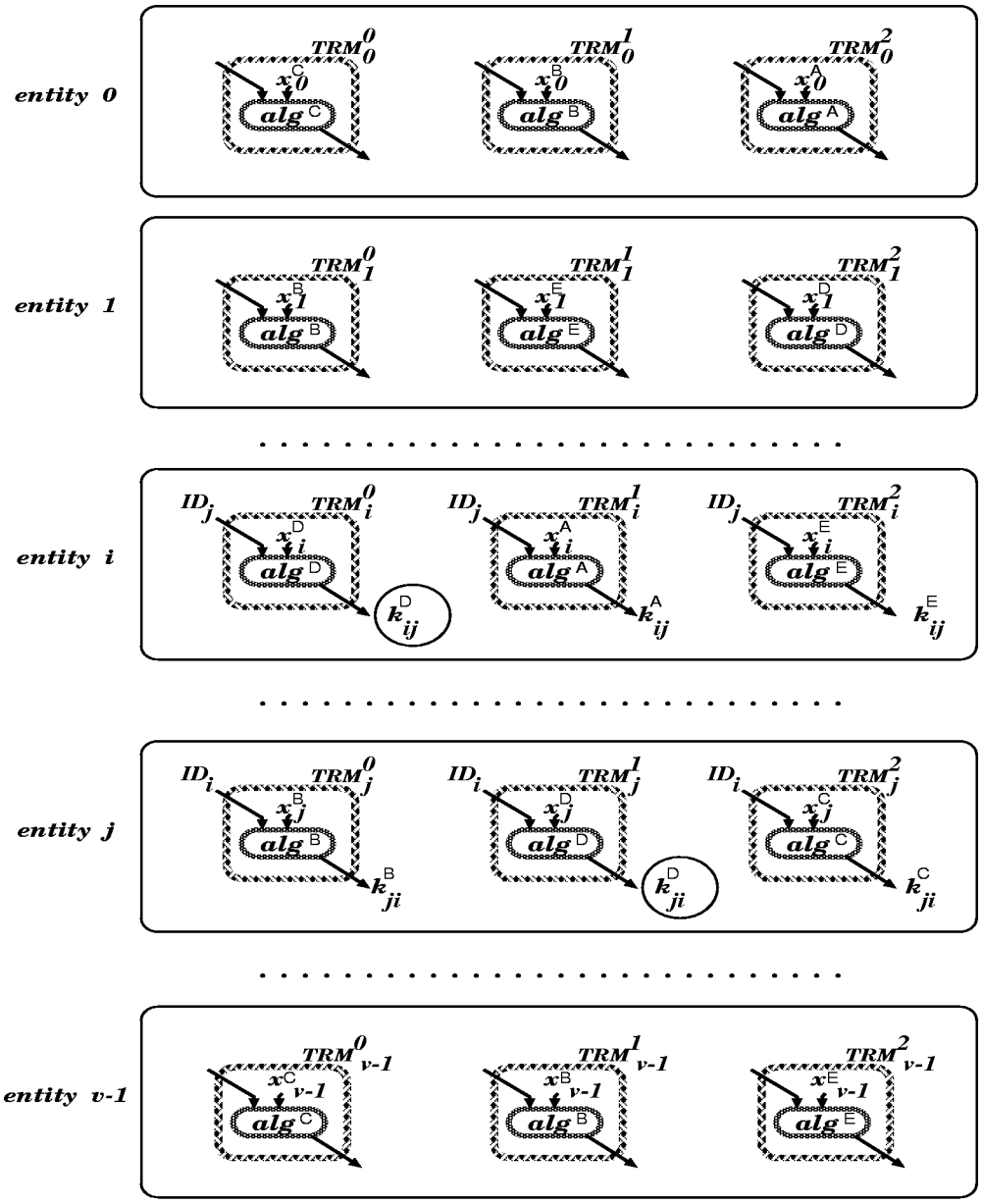

Figure 2: Using Plural Subsysems

Proposition 1. If D is constant-row-weight with $R(D) = r$ we have $\Gamma(D) = b/r$.

Observation 2. The conventional method using a single key-sharing system corresponds to matrix $J_1 \in I(v, 1)$ each of which entry is 1. Thus its gain is $\Gamma(J_1) = 1$. The same gain is attained by the method using $r = b$ TRMs per entity with b key-sharing systems. This is corresponding to matrix $J_b \in I(v, b)$ each of which entry is 1. The proposed assignment using a larger gain will save memory per entity to yield the same security level measured by the tamper-resistance required for complete breaking. For example, matrices R and S in Example 1 yield $\Gamma(R) = 5/3$ and $\Gamma(S) = 7/3$, respectively.

Desirable Incidence Structures. Let D be a cohesive incidence matrix defining our assignment. If there is an all-one column in D then the key-sharing system corresponding to the column can be always used. This is somewhat redundant, so we prefer to let D be normal. And the situation where every entity uses the same number of TRMs is easy to treat in theory and in practice, we prefer to let D be constant-row-weight. Another important condition to D is that it must have a lot of rows so that the method can be applied to large networks containing a lot of entities. Summing up all the conditions we

have the following criterion:

Matrix $D \in I(v,b)$

- should be normal, cohesive, constant-row-weight,
- should yield a large gain, $\Gamma(D)$,
- should have v which is exponential in r,
- should have an efficient algorithm A_D to compute a row from its index.

3 Constructing Desirable Incidence Structures

Most of the contents of Section 3 have appeared in [8] although some notations are changed.

3.1 Examination of All Constant-Weight Vectors of Fixed Dimension

Fact 1. For any positive integer n, a matrix W_{2n+1} which has $\binom{2n+1}{n+1}$ rows of dimension $2n+1$ and weight $n+1$ is normal, cohesive, and tactical with $W_{2n+1} \in T(\binom{2n+1}{n+1}, 2n+1, n+1, \binom{2n}{n})$. Similarly, for any integer $n \geq 2$, a matrix W_{2n} which has $\binom{2n}{n+1}$ rows of dimension $2n$ and weight $n+1$ is normal, cohesive, and tactical with $W_{2n} \in T(\binom{2n}{n+1}, 2n, n+1, \binom{2n-1}{n})$.

Example 2. Three matrices in Example 1 can be interpreted as $P = W_3$, $Q = W_4$, and $R = W_5$.

Observation 3. The above construction can achieve large values of v and k with keeping b and r small. However, there is a limitation; $b/r < 2$. Namely, the incidence structures defined by the all r-out-of-b vectors can be used for our application, but the gain b/r attained is always upper-bounded by two.

3.2 Examination of Finite Projective Planes

Fact 2. For any prime power q, there exists a projective plane $PG(2,q)$ of order q [7]. In fact, by letting W be the vector space of dimension 3 over $GF(q)$, all 1-dimensional subspaces of W as the points, and all 2-dimensional subspaces of W as the blocks, constitute an incidence structure $PG(2,q)$, which is normal, cohesive, and tactical with incidence matrix $G_q \in T(q^2+q+1, q^2+q+1, q+1, q+1)$.

Example 3. Matrix S in Example 1 determines $PG(2,2)$.

Observation 4. If a normal cohesive tactical matrix $D \in T(v,b,r,k)$ shows another regularity that $\Lambda_{ij}(D) = 1$ for every pair of different rows of D, then D defines an incidence structure called a *block design* [7]. Projective planes are examples of block designs. For any incidence matrix for any block design, we have a famous restriction on the parameters; $v \leq b$, known as Fisher's inequality [7]. For this reason, the block designs cannot serve as a good source of incidence structures desirable for our application. Nevertheless, since the gain $b/r = (q^2+q+1)/(q+1) > q$ can be large, projective planes could be used as seeds with which generate desirable incidence structures.

3.3 A Binary Operation

Definition 4a. Let v, b, B be positive integers and $C \in I(v,b)$, $D \in I(1,B)$ be matrices such that

$$C = \begin{bmatrix} C(0) \\ \vdots \\ C(v-1) \end{bmatrix}, \quad C(i) \in I(1,b), \qquad D = [D_0, \cdots, D_{B-1}], \quad D_j \in \{0,1\}.$$

Let $w_0(D) = 0$ and $w_j(D) = \sum_{g=0}^{j-1} D_g$ for $j = 1, \cdots, B$.
If $v > 1$ let $i(f)$ $(\in \{0, \cdots, v-1\})$ denote the f^{th} $(\in \{0, \cdots, w_B(D)-1\})$ digit of the radix-v expression of integer i $(\in \{0, \cdots, v^{w_B(D)}-1\})$.
Then $C\%D$ is defined as the matrix

$$E = [E_{ij}]_{0 \leq i < v^{w_B(D)}, 0 \leq j < B} \in I(v^{w_B(D)}, b \cdot B)$$

such that
1) when $v = 1$,

$$E_{ij} = \begin{cases} [0, \cdots, 0] \in I(1, b) & \text{if } D_j = 0, \\ C(0) \in I(1, b) & \text{if } D_j = 1, \end{cases}$$

2) when $v > 1$,

$$E_{ij} = \begin{cases} [0, \cdots, 0] \in I(1, b) & \text{if } D_j = 0, \\ C(i(w_j(D))) \in I(1, b) & \text{if } D_j = 1. \end{cases}$$

Remark. In Definition 4B, 'the f^{th} digit ' is somewhat ambiguous. We define $i(f)$ as one satisfying

$$i = \sum_{f=0}^{w_B(D)-1} v^{w_B(D)-1-f} i(f).$$

Definition 4b. Let v, b, V, B be positive integers and $C \in I(v, b)$, $D \in I(V, B)$ be matrices such that

$$D = \begin{bmatrix} D(0) \\ \vdots \\ D(V-1) \end{bmatrix}, \quad D(j) \in I(1, B).$$

Then $C\%D$ is defined by

$$C\%D = \begin{bmatrix} C\%D(0) \\ \vdots \\ C\%D(V-1) \end{bmatrix}.$$

Proposition 2. For matrices $C \in I(v, b)$ and $D \in I(V, B)$ we have $C\%D \in I(\sum_{i=0}^{V-1} v^{R_i(D)}, b \cdot B)$. In particular, if D is constant-row-weight with $R(D)$, we have $C\%D \in I(v^{R(D)} \cdot V, b \cdot B)$.

Proposition 3. Operation % is associative, *i.e.*, for matrices C, D, E we have $(C\%D)\%E = C\%(D\%E)$.

Example 4.

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \% \left[\begin{array}{ccc} 1 & 1 & 0 \\ \hline 1 & 0 & 1 \\ \hline 0 & 1 & 1 \end{array}\right] = \left[\begin{array}{ccc|ccc|ccc} 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ \hline 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ \hline 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{array}\right].$$

3.4 The Operation Preserves Useful Properties

Theorem 1. For matrices $C \in I(v,b)$ and $D \in I(V,B)$, we have the following properties.

1) $C\%D$ is non-zero if C and D are non-zero.
2) $C\%D$ is normal if C and D are normal.
3) $C\%D$ is cohesive if C and D are cohesive.
4) $C\%D$ is constant-row-weight if C and D are constant-row-weight. Moreover, $R(C\%D) = R(C) \cdot R(D)$.
5) $C\%D$ is tactical if C and D are tactical. Moreover, $K(C\%D) = v^{R(D)-1} \cdot K(C) \cdot K(D)$.

3.5 Recursively Constructing Desirable Structures

Definition 5. For a positive integer n and a binary matrix D, we define

$$D^{(n)} = \begin{cases} D & \text{if } n = 1, \\ D\%D^{(n-1)} & \text{if } n \geq 2. \end{cases}$$

Remark. In [8] we used $D^{(n-1)}\%D$. But this difference does not affect the defined object since Proposision 3 holds.

Theorem 2. For $C \in T(v,b,r,k)$ and $D \in T(V,B,R,K)$, we have

$$C\%D \in T(v^R \cdot V, b \cdot B, r \cdot R, v^{R-1} \cdot k \cdot K),$$

$$C^{(n)} \in T(v^{\frac{r^n-1}{r-1}}, b^n, r^n, v^{\frac{r^n-1}{r-1}-n} \cdot k^n).$$

Observation 5. Using the technique developed in this section we can see that many examples almost meet the criterion for desirable incidence structures. As stated in Observation 1, from these we can derive a lot of desirable normal cohesive constant-row-weight matrices. The remaining task is to devise a good row calculating algorithm.

4 Calculating a Row from Its Index

Definition 6. Let $F = [f_0, \cdots, f_{b-1}]$ be a binary b-tuple of (Hamming) weight r and $[G_0, \cdots, G_{r-1}]$ be a r-tuple of binary b-tuples. We define the #-product of F and $[G_0, \cdots, G_{r-1}]$ by

$$F\#[G_0, \cdots, G_{r-1}] = H_0 \| \cdots \| H_{b-1},$$

where for $j = 0, \cdots, b-1$, H_j is the binary b-tuple determined by

$$H_j = \begin{cases} [0, \cdots, 0] \in I(1,b) & \text{if } f_j = 0, \\ G_{f_0+\cdots+f_j-1} & \text{if } f_j = 1, \end{cases}$$

and $\|$ denotes the operation of concatenating two tuples.

Definition 7. Let $D = \begin{bmatrix} D(0) \\ \vdots \\ D(v-1) \end{bmatrix}$ with $D(i) \in I(1,b)$ be a normal cohesive tactical matrix and $D \in T(v,b,r,k)$. Let $i = [i_0, i_1, \cdots, i_{(r^n-r)/(r-1)}] \in \{0, \cdots, v-1\}^{(r^n-1)/(r-1)}$ be a v-ary $(r^n-1)/(r-1)$-tuple. For matrix D and positive integer n we define the algorithm $A_{D^{(n)}}$ that transforms i into a binary tuple

$$\begin{aligned} A_{D^{(n)}}(i) = (\cdots(\quad & (\quad D(i_0) \\ & \# \quad [D(i_1), \cdots, D(i_r)]) \\ & \# \quad [D(i_{r+1}), \cdots, D(i_{r+r^2})])\cdots) \\ & \# \quad [D(i_{(r^{n-1}-1)/(r-1)}), \cdots, D(i_{(r^n-r)/(r-1)})]. \end{aligned}$$

Theorem 3. Let $e_{v,r}(i) = \sum_{j=0}^{(r^n-r)/(r-1)} v^j i_j$. Then $A_{D^{(n)}}(i)$ is the $e_{v,r}(i)$ th row of matrix $D^{(n)}$. In particular, for any $i \in \{0,\cdots,v-1\}^{(r^n-1)/(r-1)}$, $A_{D^{(n)}}(i)$ is a binary b^n-tuple of weight r^n. And for any i and $j \in \{0,\cdots,v-1\}^{(r^n-1)/(r-1)}$, the weight of $A_{D^{(n)}}(i) \wedge A_{D^{(n)}}(j)$ is positive.

Example 6. Let

$$D = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \in T(3,3,2,2)$$

and $n = 3$.

For $i = [0,1,2,1,0,2,1]$, $j = [1,0,2,0,1,1,2]$, and $l = [2,0,1,0,2,2,1]$, we have

$$\begin{aligned} A_{D^{(3)}}(i) &= ([110]\#[[101],[011]])\#[[101],[110],[011],[101]] \\ &= [101011000]\#[[101],[110],[011],[101]] \\ &= [101000110000011101000000000], \end{aligned}$$

$$\begin{aligned} A_{D^{(3)}}(j) &= ([101]\#[[110],[011]])\#[[110],[101],[101],[011]] \\ &= [110000011]\#[[110],[101],[101],[011]] \\ &= [110101000000000000000101011], \end{aligned}$$

$$\begin{aligned} A_{D^{(3)}}(l) &= ([011]\#[[110],[101]])\#[[110],[011],[011],[101]] \\ &= [000110101]\#[[110],[011],[011],[101]] \\ &= [000000000110011000011000101]. \end{aligned}$$

Thus we have

$$\begin{aligned} A_{D^{(3)}}(i) \wedge A_{D^{(3)}}(j) &= [100000000000000000000000000], \\ A_{D^{(3)}}(i) \wedge A_{D^{(3)}}(l) &= [000000000000011000000000000], \\ A_{D^{(3)}}(j) \wedge A_{D^{(3)}}(l) &= [000000000000000000000000001]. \end{aligned}$$

Observation 6. We can derive an efficient meeting-point calculating algorithm by modifying the above row calculating algorithm. For details consult ref [9].

5 Discussion and Conclusion

1) For the key sharing problem, we have shown a way of using plural tamper-resistant modules per entity to achieve the same level of security with reduced memory.

2) We have set the criterion for the desirable incidence structure for this purpose and constructed several nice candidates.

3) We have devised an efficient row calculating algorithm for the obtained incidence structures. The property of operation % is effectively used to construct the algorithm.

4) The newly introduced approach for assigning key-sharing systems is promising and worth while to further develop theoretically and practically.

5) The present author is leading a project to implement and evaluate the approach by adopting the KPSs.

References

[1] T. Matsumoto and H. Imai, "On the Key Predistribution System: A practical solution to the key distribution problem," Advances in Cryptology: Proceedings of CRYPTO'87, Lecture Notes in Computer Science No. 293, pp. 185-193, Springer-Verlag, 1987.

[2] R. Blom, "An optimal class of symmetric key generation systems," Advances in Cryptology: Proceedings of EUROCRYPT'84, Lecture Notes in Computer Science No. 209, pp. 335-338, Springer-Verlag, 1985.

[3] L. Gong and D. J. Wheeler, "A matrix key-distribution scheme," Journal of Cryptology, Vol. 2, pp. 51-59, Springer-Verlag, 1990.

[4] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, "Perfectly-secure key distribution for dynamic conferences," Advances in Cryptology: Proceedings of CRYPTO'92, Lecture Notes in Computer Science No. 740, pp. 471-486, Springer-Verlag, 1993.

[5] W.-A. Jackson, K. M. Martin, and C. M. O'Keefe, "Multisecret threshold schemes," Advances in Cryptology: Proceedings of CRYPTO'93, Lecture Notes in Computer Science No. 773, pp. 126-135, Springer-Verlag, 1994.

[6] T. Matsumoto, "A novel IC card for KPS-based cryptography," IFIP WG10.5 Workshop on Secure Design and Test of Crypto-Chips, Abstract, Gmunden, Austria, 1991.

[7] T. Beth, D. Jungnickel, and H. Lenz, Design Theory, B.I.-Wissenschaftsverlag, 1985.

[8] T. Matsumoto, "Incidence structures for key sharing," Advances in Cryptology: Proceedings of ASIACRYPT'94, Lecture Notes in Computer Science No. 917, pp. 342-353, Springer-Verlag, 1995.

[9] T. Matsumoto, S. Inoue, and R. Uesaki, "Design and evaluation of algorithms for concurrent use of plural tamper resistant modules," SCIS96-6C, 1996 Symposium on Cryptography and Information Security, IEICE, January 29-31, 1996.