Perfectly-Secure Key Distribution for Dynamic Conferences

Carlo Blundo[1,*], Alfredo De Santis[1,*], Amir Herzberg[2], Shay Kutten[2], Ugo Vaccaro[1,*], Moti Yung[2]

[1] Dipartimento di Informatica ed Applicazioni, Università di Salerno, 84081 Baronissi (SA), Italy.

[2] IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA.

Abstract. A key distribution scheme for dynamic conferences is a method by which initially an (off-line) trusted server distributes private individual pieces of information to a set of users. Later any group of users of a given size (a dynamic conference) is able to compute a common secure key. In this paper we study the theory and applications of such perfectly secure systems. In this setting, *any* group of t users can compute a common key by each user computing using only his private piece of information and the *identities* of the other $t-1$ group users. Keys are secure against coalitions of up to k users, that is, even if k users pool together their pieces they cannot compute anything about a key of any t-size conference comprised of other users.

First we consider a non-interactive model where users compute the common key without any interaction. We prove a lower bound on the size of the user's piece of information of $\binom{k+t-1}{t-1}$ times the size of the common key. We then establish the optimality of this bound, by describing and analyzing a scheme which *exactly* meets this limitation (the construction extends the one in [2]). Then, we consider the model where interaction is allowed in the common key computation phase, and show a *gap* between the models by exhibiting an interactive scheme in which the user's information is only $k+t-1$ times the size of the common key. We further show various applications and useful modifications of our basic scheme. Finally, we present its adaptation to network topologies with neighborhood constraints.

1 Introduction

Key distribution is a central problem in cryptographic systems, and is a major component of the security subsystem of distributed systems, communication systems, and data networks. The increase in bandwidth, size, usage, and applications of such systems is likely to pose new challenges and to require novel

[*] Partially supported by Italian Ministry of University and Research (M.U.R.S.T.) and by National Council for Research (C.N.R.) under grant 91.02326.CT12.

E.F. Brickell (Ed.): Advances in Cryptology - CRYPTO '92, LNCS 740, pp. 471-486, 1993.

ideas. A growing application area in networking is "conferencing" a group of entities (or network locations) collaborate privately in an interactive procedure (such as: board meeting, scientific discussion, a task-force, a classroom, or an bulletin-board). In this work we consider perfectly-secure key distribution for conferences. (Note that key distribution for two-party communication (session-keys) is a special case of conferences of size two).

If users of a group (a conference) wish to communicate in a network using symmetric encryption, they must share a common key. A key distribution scheme (denoted KDS for short) is a method to distribute initial private pieces of information among a set of users, such that each group of a given size (or up to a given size) can compute a common key for secure conference. This information is generated and distributed by a trusted server which is active only at the distribution phase.

Various key distribution schemes have been proposed so far, mainly to pairs of users (session keys). A basic and straightforward perfectly-secure scheme (which is useful in small systems) consists of distributing initial keys to users in such a way that each potential group of users shares a common key. In the case of session keys, if n is the number of users, the server has to generate $n(n-1)/2$ keys and each user holds $n-1$ keys, one for each possible communication. When n gets large it becomes problematic or even impossible to manage all keys. This is known as the n^2 problem. For conferences, when we allow all possible subsets of a given size to join together (what we call the dynamic conference setting), the number of keys becomes prohibitively large.

Given the high complexity of such a distribution mechanism, a natural step is to trade complexity for security. We may still require that keys are perfectly secure, but only with respect to an adversary controlling coalitions of a limited size. This novel approach was initiated by Blom [2] for the case of session keys (other related schemes are given in [10, 14]). We are motivated by Blom's (somewhat forgotten) pioneering work. We consider key-distribution for dynamic conferences and study the theory and applications of such systems. Our scheme has two parameters: t, the size of the conference (group), and k, the size of adversary coalitions. Another characteristic of such schemes is whether they are interactive (users discuss during common-key establishment phase) or non-interactive.

1.1 The results

We give a precise model of our setting and then we analyze and design perfectly-secure key distribution schemes for dynamic conferences. We show the following:

1. **Lower bound**: We consider the non-interactive model and prove that the size of the piece of a user's information is at least $\binom{k+t-1}{t-1}$ times the size of the common key.
2. **Matching upper bound**: We propose a concrete scheme and show that it indeed gives pieces of this size, thus establishing the optimality of the bound.
3. **Gap**: We compare the interactive to the non-interactive settings. We show an interactive scheme where the user's information is only $k+t-1$ times the

size of the common key, proving a separation between the interactive and the non-interactive cases.
4. **Constrained Conferencing**: In Section 7 we present modifications of the schemes to systems in which conferences are generated according to neighborhood constraints (of the network communication graph).
5. **Applications**: We then extend the ideas to show numerous applications and uses of the scheme, such as: hierarchical key distribution schemes, asymmetric user-population, access-control validation, partial key revocation, etc.

Our analysis applies information-theory and its basic notions of entropy and mutual information, as well as their conditional versions. In Section 2 we review these notions and present basic equations to be used in the analysis.

1.2 Related work

The two common approaches to key distribution, taken in order to reduce the inherent complexity of the basic straightforward scheme are schemes based on public-key cryptography [5] or on an authentication server [19]. Numerous suggestions for key distribution schemes based on computational assumptions are known, as well as a number of suggestions for conference keys. We note that "Merkle's puzzles" [17] is also a pioneering key generation scheme which is computational, for a *seemingly negative* result concerning such methods see [11]. The interactive model is related to (but different from) the recent models basing perfectly-secure common key generation on an initial card deal [6, 7]. Blom's innovative method (and thus our setting) is a key distribution which is ID-based that predated the formal definition of this notion by Shamir [21]; his technical tool was MDS linear codes. Later, Matsumoto and Imai [16] extended the work of [2] to general symmetric functions, and systematically defined key distribution schemes based on such general function; our scheme can actually be viewed as a special case of their general system. (Another related recent work is in [23]). Fiat and Naor have suggested recently a key distribution scheme which is not algebraic, and Alon has given a lower bound for their scheme [18]. Remark: finally we note that various suggestions for computational key distribution in different settings (e.g., [15, 20, 25, 24, 8]) and conferencing (e.g., [12, 3, 22]) have appeared in the last years, (mainly in the Crypto and Eurocrypt conferences proceedings series).

Organization: In Section 2 we recall the definition of the entropy and some of its property. In Section 3 we formally describe the model of a KDS in terms of the entropy. In Section 4 we prove the lower bound on the entropy of each user in a k-secure t-conference KDS. In Section 5 we then describe and analyze the actual schemes for k-secure t-conference KDS. In Section 6 we show how interaction can be used to dramatically decrease the amount of information held by each user. In Section 7 we present another result: a protocol to realize a conference KDS when not all of pairs of users are able to communicate. In Section 8 we present applications, in particular the scheme can be combined with authentication procedures, as the ID of the owner and other meaning attached to a key owner can be naturally supported by such a system.

2 Background

In this part we review the information theoretic concepts we are going to use. For a complete treatment of the subject the reader is advised to consult [4] and [9].

Given a probability distribution $\{p(x)\}_{x \epsilon X}$ on a set X, we define the *entropy* of X, $H(X)$, as

$$H(X) = -\sum_{x \epsilon X} p(x) \log p(x)^2.$$

The entropy $H(X)$ is a measure of the average information content of the elements in X or, equivalently, a measure of the average uncertainty one has about which element of the set X has been chosen when the choices of the elements from X are made according to the probability distribution $\{p(x)\}_{x \epsilon X}$. It is well known that $H(X)$ is a good approximation to the average number of bits needed to faithfully represent the elements of X. The following property of $H(X)$ can somehow illustrate the soundness of our first claim:

$$0 \leq H(X) \leq \log |X|, \qquad (1)$$

where $H(X) = 0$ if and only if there exists $x_0 \in X$ such that $p(x_0) = 1$; $H(X) = \log |X|$ if and only if $p(x) = 1/|X|$, $\forall x \in X$.

Given two sets X and Y and a joint probability distribution $\{p(x, y)\}_{x \epsilon X, y \epsilon Y}$ on their cartesian product, the *conditional entropy* $H(X|Y)$, also called the equivocation of X given Y, is defined as

$$H(X|Y) = -\sum_{y \epsilon Y} \sum_{x \epsilon X} p(y) p(x|y) \log p(x|y).$$

The conditional entropy can be written as $H(X|Y) = \sum_{y \epsilon Y} p(y) H(X|Y = y)$ where $H(X|Y = y) = -\sum_{x \epsilon X} p(x|y) \log p(x|y)$ can be interpreted as the average uncertainty one has about which element of X has been chosen when the choices are made according to the probability distribution $p(x|y)_{x \epsilon X}$, that is, when it is known that the value chosen ¿from the set Y is y. From the definition of conditional entropy it is easy to see that

$$H(X|Y) \geq 0. \qquad (2)$$

If we have $n + 1$ sets $X_1, \ldots, X_n, Y$ the entropy of $X_1 \ldots X_n$ given Y can be written as

$$H(X_1 \ldots X_n|Y) = H(X_1|Y) + H(X_2|X_1 Y) + \cdots + H(X_n|X_1 \ldots X_{n-1} Y) \qquad (3)$$

[2] All logarithms in this paper are of base 2

The *mutual information* between X and Y is defined by

$$I(X;Y) = H(X) - H(X|Y) \tag{4}$$

and enjoys the following properties:

$$I(X;Y) = I(Y;X), \tag{5}$$

and $I(X;Y) \geq 0$, from which one gets

$$H(X) \geq H(X|Y) \tag{6}$$

with equality if and only if X and Y are independent. Given sets X, Y, Z and a joint probability distribution on their cartesian product, the *conditional mutual information* between X and Y given Z can be written as

$$I(X;Y|Z) = H(X|Z) - H(X|Z\ Y). \tag{7}$$

Since a property of the conditional mutual information is $I(X;Y|Z) \geq 0$ we get

$$H(X|Z) \geq H(X|Z\ Y). \tag{8}$$

3 The Model

In this section we present the key distribution problem and model. A key distribution scheme (indicated by KDS for short) distributes some information among a set of users, so that any t of them can join and generate a secure key. We assume a trusted off-line server active only at initiation (unlike an on-line server approach put forth in [19] which we call server-based KDS). We say the system is k-secure if any k users, pooling together their pieces, have no information on keys they should not know. These schemes can be further classified into two categories: interactive (where users are engaged in a protocol, prior to usage of the common key), and non-interactive where keys are generated privately by the individuals. Next, we formally define non-interactive key distribution schemes. Our definition of security is based on the notion of entropy and is thus unconditional.

Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a set of users. The algorithm used by the server to generate the pieces of information, that will be distributed to the users, is randomized. The server generates the vector $(u_1, u_2, \ldots, u_n)$ according to some probability distribution on the cartesian product $U_1 \times \cdots \times U_n$. The piece u_i denotes the information given by the server to user U_i. In order to simplify notation we denote by U_i both the user U_i and the random variable induced by the value u_i, and by $S_{i_1,\ldots,i_t}$ we denote both the set of common keys among users $U_{i_1}, \ldots, U_{i_t}$ and the random variable induced by these common keys. Each user U_{i_j} can deterministically compute, on input only u_{i_j} and $i_1, \ldots, i_{j-1}, i_{j+1}, \ldots, i_t$, his common keys $s_{\sigma(i_1),\ldots,\sigma(i_t)}$, for all permutations $\sigma : \{i_1, i_2, \ldots, i_t\} \rightarrow \{i_1, i_2, \ldots, i_t\}$, to be used with users $U_{i_1}, \ldots, U_{i_{j-1}}, U_{i_{j+1}}, \ldots, U_{i_t}$. Each common key $s_{i_1,\ldots,i_t}$ is generated according to a probability distribution $\{p(s_{i_1,\ldots,i_t})\}_{i_1,\ldots,i_t}$, induced by the fact that each user calculates deterministically the common key by using the initial information received ¿from the server, which has been generated by a randomized algorithm. The probability $p(s_{i_1,\dots,i_t})$ denotes the *a priori* probability that the common key among users $U_{i_1},\dots,U_{i_t}$ is $s_{i_1,\dots,i_t}$.

The maximum value that the security parameter k can take in any t-conference KDS for n users is $n-t$ since any adversary coalition can contain at most $n-t$ users. Formally we define a k-secure t-conference key distribution scheme for n users as follows.

Definition 3.1 *Let $\mathcal{U}$ be a set of users and let k, $k \le |\mathcal{U}| - t$, be an integer. A non-interactive key distribution scheme for $\mathcal{U}$ is k-secure if*

1. *Each t-uple of users can non-interactively compute the common key.* For all $U_{i_1},\dots,U_{i_t} \in \mathcal{U}$, it holds $p(s_{i_1,\dots,i_t}|u_{i_1}) = \cdots = p(s_{i_1,\dots,i_t}|u_{i_t}) = 1$.
2. *Any group of k users have no information on a key they should not know.* For all $U_{i_1},\dots,U_{i_t},U_{j_1},\dots,U_{j_k} \in \mathcal{U}$ such that $j_1,\dots,j_k \notin \{i_1,\dots,i_t\}$, it holds $p(s_{i_1,\dots,i_t}|u_{j_1},\dots,u_{j_k}) = p(s_{i_1,\dots,i_t})$.

Property 1. means that given the value held by the user U_{i_l}, $l = 1,2,\dots,t$, a unique value of the common key exists. Property 2. states that the probability that the common key among users $U_{i_1},\dots,U_{i_t}$ is $s_{i_1,\dots,i_t}$ given the information held by users $U_{j_1},\dots,U_{j_k}$ is equal to the *a priori* probability that the common key is $s_{i_1,\dots,i_t}$. This means that random variables $S_{i_1,\dots,i_t}$ and $U_{j_1} \times \cdots \times U_{j_k}$ are statistically independent, so the values $u_{j_1},\dots,u_{j_k}$ reveal no information on the common key $s_{i_1,\dots,i_t}$. By using the entropy function it is possible to give an equivalent definition of a k-secure non-interactive t-conference KDS.

Definition 3.2 *Let $\mathcal{U} = \{U_1,\dots,U_n\}$ be a set of users and let k, $k \le n-t$, be an integer. A non-interactive t-conference key distribution scheme for $\mathcal{U}$ is k-secure if*

1′. *Each t users can non-interactively compute the common key.* For all different $i_1,\dots,i_t \in \{1,2,\dots,n\}$, $H(S_{i_1,\dots,i_t}|U_{i_1}) = \cdots = H(S_{i_1,\dots,i_t}|U_{i_t}) = 0$.

2′. *Any group of k users have no information on a key they should not know.* For all users $U_{j_1},\dots,U_{j_k}$ such that $j_1,\dots,j_k \notin \{i_1,\dots,i_t\}$, $H(S_{i_1,\dots,i_t}|U_{j_1}\dots U_{j_k}) = H(S_{i_1,\dots,i_t})$.

Notice that $H(S_{i_1,\dots,i_t}|U_{i_1}) = \cdots = H(S_{i_1,\dots,i_t}|U_{i_t}) = 0$, for all different $i_1,\dots,i_t \in \{1,2,\dots,n\}$, means that each set of values held by the user U_{i_l}, $l = 1,2,\dots,t$, corresponds to a unique value of the common key. In fact, by definition, $H(S_{i_1,\dots,i_t}|U_{i_l}) = 0$ is equivalent to the fact that for all $u_{i_l} \in U_{i_l}$ with $p(u_{i_l}) > 0$, a unique value $s_{i_1,\dots,i_t} \in S_{i_1,\dots,i_t}$ such that $p(s_{i_1,\dots,i_t}|u_{i_l}) = 1$ exists. Moreover, $H(S_{i_1,\dots,i_t}|U_{j_1}\dots U_{j_k}) = H(S_{i_1,\dots,i_t})$ is equivalent to saying that $S_{i_1,\dots,i_t}$ and $U_{j_1} \times \cdots \times U_{j_k}$ are statistically independent, i.e., for all $(u_{j_1},\dots,u_{j_k}) \in U_{j_1} \times \cdots \times U_{j_k}$, we have $p(s_{i_1,\dots,i_t}|u_{j_1},\dots,u_{j_k}) = p(s_{i_1,\dots,i_t})$.

Property 1$'$. in Definition 3.2 states that any t users can compute the same common key. Actually, each user U_i can calculate $t!$ keys for the same conference. Property 1$'$. does not say anything on the relationship among these $t!$ keys: all $t!$ keys could be equal so one key uniquely determines the other keys, that is $H(S_{\sigma(i_1),\ldots,\sigma(i_t)}|S_{i_1,\ldots,i_t}) = 0$, for all permutation $\sigma : \{i_1, i_2, \ldots, i_t\} \rightarrow \{i_1, i_2, \ldots, i_t\}$; or the keys could be all different and given one key we do not know anything on the other keys, that is $H(S_{\sigma(i_1),\ldots,\sigma(i_t)}|S_{i_1,\ldots,i_t}) = H(S_{\sigma(i_1),\ldots,\sigma(i_t)})$. Our lower bounds are valid in both cases, since they are based only on Property 1$'$. and 2$'$.. On the other hand, in this paper all schemes that realize k-secure t-conference KDS are symmetric, that is schemes in which the common key is symmetric: $s_{i_1,\ldots,i_t} = s_{\sigma(i_1),\ldots,\sigma(i_t)}$ for all permutations $\sigma : \{i_1, i_2, \ldots, i_t\} \rightarrow \{i_1, i_2, \ldots, i_t\}$.

Definition 3.2 does not say anything on the entropies of random variables $S_{i_1,\ldots,i_t}$ and $S_{i'_1,\ldots,i'_t}$. For example, we could have either $H(S_{i_1,\ldots,i_t}) > H(S_{i'_1,\ldots,i'_t})$ or $H(S_{i_1,\ldots,i_t}) \leq H(S_{i'_1,\ldots,i'_t})$. Our results apply for the general case of arbitrary entropies on keys, but for clarity we often state our results for the simpler case that all entropies on keys are equal, i.e. $H(S_{i_1,\ldots,i_t}) = H(S_{i'_1,\ldots,i'_t})$ for all t-uples of users $(U_{i_1}, \ldots, U_{i_t})$ and $(U_{i'_1}, \ldots U_{i'_t})$, and we denote this entropy by $H(S)$.

The next simple lemma proves that if a t-conference KDS is k-secure then it is k'-secure for all integers $k' < k$.

Lemma 3.1 *Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a set of users and let k, $k \leq n - t$, be an integer. In any k-secure key distribution scheme for $\mathcal{U}$, for any integer $k' < k$ it holds*

$$H(S_{i_1,\ldots,i_t}|U_{j_1} \ldots U_{j_{k'}}) = H(S_{i_1,\ldots,i_t}).$$

For all users $U_{i_1}, \ldots, U_{i_t}, U_{j_1}, \ldots, U_{j_{k'}}$ such that $j_1, \ldots, j_{k'} \notin \{i_1, \ldots, i_t\}$.

Proof : From 2$'$. of Definition 3.2 we have $H(S_{i_1,\ldots,i_t}) = H(S_{i_1,\ldots,i_t}|U_{j_1} \ldots U_{j_k})$. From (8), one gets

$$H(S_{i_1,\ldots,i_t}|U_{j_1} \ldots U_{j_k}) \leq H(S_{i_1,\ldots,i_t}|U_{j_1} \ldots U_{j_{k'}}) \leq H(S_{i_1,\ldots,i_t}).$$

Thus, $H(S_{i_1,\ldots,i_t}|U_{j_1} \ldots U_{j_{k'}}) = H(S_{i_1,\ldots,i_t})$. □

From Lemma 3.1 one has that Property 2$'$. can be equivalently written as

2$''$. *Any group of $k' \leq k$ users have no information on a key they should not know.*
For all users $U_{i_1}, \ldots, U_{i_t}, U_{j_1}, \ldots, U_{j_{k'}}$ such that $j_1, \ldots, j_{k'} \notin \{i_1, \ldots, i_t\}$, it holds
$H(S_{i_1,\ldots,i_t}|U_{j_1} \ldots U_{j_{k'}}) = H(S_{i_1,\ldots,i_t})$.

4 Lower Bound: Conference Key Distribution

In this section we prove a lower bound on the size of user's information for a k-secure t-conference KDS. Let $U_{i_1} \dots U_{i_t}$ be t users and let $A = \{j_1, \dots, j_t\}$ be a set of t indices. With S_A we denote both the set of common keys among the users $U_{i_1} \dots U_{i_t}$ and the random variable induced by these common keys, and with U_A we denote both the set of users $\{U_{i_1}, \dots, U_{i_t}\}$ and the random variable induced by the value $u_{i_1}, \dots, u_{i_t}$.

In a k-secure t-conference KDS the knowledge of k keys does not convey any information on another key. This is formalized by next lemma.

Lemma 4.1 *Let $\mathcal{U} = \{U_1, \dots, U_n\}$ be a set of n users and let r and k, $k \le n-t$, be integers. Let $X, Y_1, \dots, Y_r, Z$ be subsets of $\{1, 2, \dots, n\}$ such that $|Z| = k$, $Z \cap X = \emptyset$, $Z \cap Y_i \neq \emptyset$ and $|X| = |Y_i| = t$, for $i = 1, \dots, r$. Then, in any k-secure t-conference key distribution scheme for $\mathcal{U}$*

$$H(S_X | S_{Y_1} \dots S_{Y_r}) = H(S_X).$$

Proof : From (6) we have $H(S_X) \ge H(S_X | S_{Y_1} \dots S_{Y_r})$. To prove the lemma it is enough to prove that $H(S_X | S_{Y_1} \dots S_{Y_r}) \ge H(S_X)$. Note that $Z \cap X = \emptyset$.
First note that the conditional mutual information between S_X and $S_{Y_1} \dots S_{Y_r}$ given U_Z is

$$\begin{aligned} I(S_{Y_1} \dots S_{Y_r} \,; S_X | U_Z) &= H(S_{Y_1} \dots S_{Y_r} | U_Z) - H(S_{Y_1} \dots S_{Y_r} | U_Z S_X) \text{ (from (7))} \\ &\le H(S_{Y_1} \dots S_{Y_r} | U_Z) \text{ (from (2))} \\ &\le \sum_{l=1}^{r} H(S_{Y_l} | U_Z) \text{ (from (3) and (8))} \\ &\le 0 \text{ (from (8) and 1$'$. of Definition 3.2)} \end{aligned}$$

Since the mutual information is non-negative we have

$$I(S_{Y_1} \dots S_{Y_r} \,; S_X | U_Z) = 0$$

From (5) it follows $I(S_X \,; S_{Y_1} \dots S_{Y_r} | U_Z) = I(S_{Y_1} \dots S_{Y_r} \,; S_X | U_Z)$ and thus

$$H(S_X | U_Z) = H(S_X | U_Z S_{Y_1} \dots S_{Y_r}). \tag{9}$$

Finally, one gets

$$\begin{aligned} H(S_X | S_{Y_1} \dots S_{Y_r}) &\ge H(S_X | U_Z \, S_{Y_1} \dots S_{Y_r}) \text{ (from (6))} \\ &= H(S_X | U_Z) \text{ (from (9))} \\ &= H(S_X) \text{ (from 2$'$. of Definition 3.2)} \end{aligned}$$

which proves the lemma. □

We assume that all keys have the same entropy, i.e. $H(S_{j_1,\ldots,j_t}) = H(S)$ for all different $j_1, \ldots, j_t$. Next theorem states a lower bound on the size of information held by each user.

Theorem 4.1 *Let $\mathcal{U}$ be a set of n users and let k, $k \leq n-t$, be an integer. In any k-secure t-conference key distribution scheme, the entropy $H(U_i)$ of each user U_i satisfies*

$$H(U_i) \geq \binom{k+t-1}{t-1} H(S).$$

Proof : Consider the set of indices $I = \{j_1, \ldots, j_{k+t-1}\}$ and an index i such that $i \notin I$. Let $m \doteq \binom{k+t-1}{t-1} - 1$. Construct $A, B_1, \ldots, B_m, C$ as follows. Set C is equal to $C \doteq \{j_1, \ldots, j_k\}$, set A is equal to $A = \{i, j_{k+1}, \ldots, j_{k+t-1}\}$, and, finally, set B_l, for $l = 1, \ldots, m$ is constructed taking the element i along with any $(t-1)$ elements from the set I, with the exception of $\{j_{k+1}, \ldots, j_{k+t-1}\}$, that is,

$$B_l \in \Big\{\{i, x_1, \ldots, x_{t-1}\} | x_1, \ldots, x_{t-1} \in I,\ \{x_1, \ldots, x_{t-1}\} \neq \{j_{k+1}, \ldots, j_{k+t-1}\}\Big\}.$$

We have

$$\begin{aligned}
H(U_i) &= H(S_{B_1} \ldots S_{B_m} S_A) - H(S_{B_1} \ldots S_{B_m} S_A | U_i) + H(U_i | S_{B_1} \ldots S_{B_m} S_A) \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(from (4) and (5))} \\
&\geq H(S_{B_1} \ldots S_{B_m} S_A) - \sum_{l=1}^{m} H(S_{B_l} | U_i) - H(S_A | U_i) + H(U_i | S_{B_1} \ldots S_{B_m} S_A) \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(from (3) and (8))} \\
&= H(S_{B_1} \ldots S_{B_m} S_A) + H(U_i | S_{B_1} \ldots S_{B_m} S_A) \quad \text{(from 1$'$. of Definition 3.2)} \\
&\geq H(S_{B_1} \ldots S_{B_m} S_A) \quad \text{(from (2))} \\
&= H(S_{B_1}) + H(S_{B_2} | S_{B_1}) + \cdots + H(S_{B_m} | S_{B_1} \ldots S_{B_{m-1}}) + H(S_A | S_{B_1} \ldots S_{B_m}) \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(from (3))}
\end{aligned}$$

Sets $Z = A$, $X = C$, $Y_l = B_l$ for $l = 1, \ldots, m$ satisfy the hypothesis of Lemma 4.1. Thus we have $H(S_A | S_{B_1} \ldots S_{B_m}) = H(S_A)$. Moreover, for each h, $1 \leq h \leq m$, sets $X = B_h$, $Z = I \setminus B_h$ and $Y_l = B_l$, for $l = 1, \ldots, h-1$, satisfy the hypothesis of Lemma 4.1. Thus, $H(S_{B_h} | S_{B_1} \ldots S_{B_{h-1}}) = H(S_{B_h})$ and,

$$\begin{aligned}
H(U_i) &\geq H(S_{B_1}) + H(S_{B_2}) + \cdots + H(S_{B_m}) + H(S_A) \\
&= (m+1) H(S) \\
&= \binom{k+t-1}{t-1} H(S)
\end{aligned}$$

Hence the theorem follows. □

A particular case of Theorem 4.1 is when $t = 2$ and $k = n - 2$. In this case the key of a pair of users cannot be computed (even one of its bits cannot be computed) by an adversary coalition of the other $n - 2$ users. Each user holds at least $n - 1$ pieces of information of size equal to the size of the common key. The total number of pieces of information held by all users is at least $n(n - 1)$. This is the well know problem of n^2 keys. The bound $H(U_i) \geq \binom{k+t-1}{t-1} H(S)$ is achieved by the protocol we next propose.

5 Protocols for Key Distribution

In this section we design and analyze protocols for k-secure t-conference key distribution which are applicable to hierarchical KDS as well (as will be later explained). The scheme we propose when applied to 2-party KDS is a particular case of the Blom's scheme [2] based on MDS linear codes, and, in particular based on polynomials.

Blom's protocol for a k-secure (2-conference) KDS for n users is as following. Let G be a (publicly known) generator matrix of a $(n, k + 1)$ MDS linear code over $GF(q)$ (see [13] for definitions and analysis of such codes) and let D be a secret random matrix with elements in $GF(q)$. From the matrices G and D, construct a $n \times n$ symmetric matrix K whose entries will be the users' keys. The matrix K is equal to $K = (DG)^T G$. The information given to user U_i consists of the row i of $(DG)^T$. If user U_i wants to communicate with user U_j then he computes the inner product of the held vector with the column j of G and he obtains the common key $s_{i,j} = K(i, j)$.

We propose the following protocol (to be extendible to various other applications in the sequel) for a k-secure t-conference KDS. Let $P(x_1, \ldots, x_t)$ be a symmetric polynomial in t variables of degree k with coefficients over $GF(q)$, $q > n$, that is, $P(x_1, \ldots, x_t) = P(x_{\sigma(1)}, \ldots, x_{\sigma(t)})$ for all permutations $\sigma : \{1, 2, \ldots, t\} \rightarrow \{1, 2, \ldots, t\}$. To each user U_i the server gives the polynomial $f_i(x_2, \ldots, x_t) = P(i, x_2 \ldots, x_t)$, that is the polynomial obtained by evaluating $P(x_1, \ldots, x_t)$ at $x_1 = i$. If users $U_{j_1}, \ldots, U_{j_t}$ want to set up a conference key then each user U_{j_i} evaluates $f_{j_i}(x_2, \ldots, x_t)$ at $(x_2, \ldots, x_t) = (j_1, \ldots, j_{i-1}, j_{i+1}, \ldots, j_t)$. The conference key is equal to $s_{j_1, \ldots, j_t} = P(j_1, \ldots, j_t)$.

As we mentioned above, when $t = 2$ our scheme is a particular case of Blom's scheme. Indeed, the generator matrix G of the MDS code is constructed by setting the entry $G(i, j)$ to j^{i-1}.

Theorem 5.1 *In the scheme based on symmetric polynomial, if all coefficients of the symmetric polynomial in t variables of degree k are uniformly chosen in $GF(q)$, then the t-conference key distribution scheme is k-secure, and optimal.*

The scheme proposed meets the bound provided by Theorem 4.1, when all coefficients are uniformly chosen. Indeed, in a symmetric polynomial $P(x_1, \ldots, x_r)$ the coefficient $a_{i_1, \ldots, i_r}$ is equal to $a_{\sigma(i_1), \ldots \sigma(i_r)}$, for all permutations $\sigma : \{i_1, i_2, \ldots, i_r\} \rightarrow \{i_1, i_2, \ldots, i_r\}$. Thus, the number of coefficients of a symmetric polynomial in r variables of degree k is equal to the number of possible ways of choosing with repetitions r elements (corresponding to indices $i_1, \dots, i_r$) from a set of $k+1$ elements (each i_j can assume $k+1$ values). This is equal to $\binom{k+r}{r}$.

6 Non-Interactive versus Interactive Schemes

In Section 4 we proved that in a non-interactive k-secure t-conference KDS, for each user U_i it holds $H(U_i) \geq \binom{k+t-1}{t-1} H(S)$. In this section we prove that if we allow interaction among users (not with the server!) to set up a common key, then the lower bound can be beaten!

The idea of the protocol is the following. We construct a non-interactive $(k+t-2)$-secure 2-conference KDS using the protocol in [2]. Given a group of t users that want to compute a conference key, the user with the largest identity in the group chooses as conference key a random value in $GF(q)$. Then he sends this value to the other $t-1$ users by using the $(k+t-2)$-secure 2-conference KDS. More formally the protocol for users $U_1, \dots U_n$, is the following (based on the scheme presented above).

1. The server chooses a symmetric polynomial $P(x, y)$ of degree $k+t-2$, with coefficients over $GF(q)$, $q > n$, by randomly choosing its coefficients.
2. To each user U_i the server gives the polynomial $f_i(y) = P(i, y)$, that is the polynomial obtained by evaluating $P(x, y)$ at $x = i$.
3. If users $U_{i_1}, \dots, U_{i_t}$, where $i_1 < i_2 < \cdots < i_t$, want to set up a conference key, then:
   - 3.1 User U_{i_t} randomly chooses a secret key s in $GF(q)$.
   - 3.2 User U_{i_t} evaluates the polynomial $f_{i_t}(y)$ at $y = i_l$, for $l = 1, \dots, t-1$, and, then, he computes temporary keys $s_{i_t, i_l} = f_{i_t}(i_l)$ (which is equal to $P(i_t, i_l)$).
   - 3.3 User U_{i_t} sends to user U_{i_l} the value $\alpha_l = s_{i_t, i_l} \otimes s$, for $l = 1, \dots, t-1$, where $\otimes$ is the bitwise xor.
   - 3.4 For $l = 1, \dots, t-1$:
     User U_{i_l}, first computes $s_{i_t, i_l} = s_{i_l, i_t} = f_{i_l}(i_t)$ (which is equal to $P(i_l, i_t) = P(i_t, i_l)$). Then, U_{i_l} computes s by taking the bitwise xor between s_{i_t, i_l} and the value α_l received by U_{i_t}.

The above protocol is k-secure, since the KDS that is established at steps 1 and 2 is $(k+t-2)$-secure.

In the above protocol only $k+t-1$ elements of $GF(q)$ are distributed by the server and kept by each user.

This, proves a separation between the interactive and the non-interactive case for information-theoretically key distribution schemes for dynamic conferences.

7 Conference Key Distribution and Communication Graph

In a non-interactive 2-conference KDS for n users each pair of users is able to compute a common key. It can be the case that some pairs of users will never need to compute a common key. This situation can arise when a computer network has a topology which is not the complete graph; here each computer takes the place of a user in a KDS, and two computers can communicate if and only if there is a link between them. As an example, consider a ring of n computers $\mathcal{R} = \{C_0, C_1, \ldots, C_{n-1}\}$: computer C_i can communicate with only two computers, C_{i-1} and C_{i+1} (arithmetic on indices is modulo n) so it will never need to compute a common key with C_{i+2}.

In this section we analyze this situation.

Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a set of users. A *communication structure* $\mathcal{C}$ is a subset of $\mathcal{U} \times \mathcal{U}$. The communication structure contains all pairs of users for which the server has to provide a common key. A convenient way to represent a communication structure is by a graph G, in which each vertex U_i corresponds to user U_i, and there is an edge (U_i, U_j) if and only if $(U_i, U_j) \in \mathcal{C}$. We call the graph associated to a communication structure the *communication graph*.

Definition 3.2 can be extended to a key distribution scheme for any communication structure $\mathcal{C}$, as follows.

Definition 7.1 *Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a set of users, let $k \leq n-2$, be an integer, and let $\mathcal{C} \subseteq \mathcal{U} \times \mathcal{U}$ be a communication structure. A non-interactive key distribution scheme for $\mathcal{C}$ is k-secure if*

1. *Each pair of users in $\mathcal{C}$ can non-interactively compute the common key.* For all $(U_i, U_j) \in \mathcal{C}$, $H(S_{i,j}|U_i) = H(S_{i,j}|U_j) = 0$.
2. *Any group of k users have no information on a key they should not know.* For all users $U_i, U_j, U_{i_1}, \ldots, U_{i_k}$ such that $i, j \notin \{i_1, \ldots, i_k\}$, $H(S_{i,j}|U_{i_1} \ldots U_{i_k}) = H(S_{i,j})$.

Now, we describe a k-secure (2-conference) KDS for a communication structure $\mathcal{C}$. First, we do not take into account the communication structure and construct a k-secure KDS for all users as if each pair has to compute a common key. User U_i could receive more information than needed. If the degree of vertex U_i in the communication graph is less than k, then the piece of information given to U_i could consist of only the actual keys he needs for communicating.

Below we describe a non-interactive k-secure key distribution scheme for a communication structure $\mathcal{C}$. In the following, $deg(U_i)$ denotes the cardinality of the set $\{U_j | (U_i, U_j) \in \mathcal{C}\}$.

1. The server chooses a symmetric polynomial $P(x, y)$ of degree k with coefficients over $GF(q)$, $q > n$, by randomly choosing its coefficients.
2. To each user U_i, the server gives the following pieces of information:
   - 2.1 If $deg(U_i) > k$ then the server gives to user U_i the polynomial $f_i(y) = P(i, y)$, that is the polynomial obtained by evaluating $P(x, y)$ at $x = i$.
   - 2.2 If $deg(U_i) \leq k$ and $U_{i_1}, \ldots, U_{i_m}$, where $m = deg(U_i)$, are the adjacent vertices of U_i in the communication graph G, then the server gives to user U_i the pieces $\alpha_j = P(i, i_j)$, where $j = 1, \ldots, m$.

This protocol is k-secure. The proof is analogous to the proof of Theorem 5.1.

Theorem 7.1 *The above described non-interactive key distribution scheme for a communication structure C is k-secure.*

It is easy to see that in previous protocol each user U_i receives $\min\{k+1, deg(u_i)\}$ pieces of information, that is the size of the information he has is $\min\{k+1, deg(u_i)\}$ the size of the common key. The following theorem proves that the protocol is optimal with respect to the size of the information held by each user. In the following theorem we suppose that all keys have the same entropy, i.e. $H(S_{i,j}) = H(S)$ for all i and j.

Theorem 7.2 *Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a set of users, let k, $k \leq n-2$, be an integer, and let G be a communication graph on $\mathcal{U}$. In any k-secure key distribution scheme for G, the entropy $H(U_i)$ of each user U_i satisfies*

$$H(U_i) \geq \mu \cdot H(S),$$

where $\mu = \min\{k+1, deg(u_i)\}$.

Proof : Let (U_i, U_{j_1}), ... ,(U_i, U_{j_μ}) be elements of the communication structure described by graph G. That is, the server has to provide a common key for such pairs of users. Then, one has

$$H(U_i) = H(S_{i,j_1} \ldots S_{i,j_\mu}) - H(S_{i,j_1} \ldots S_{i,j_\mu} | U_i) + H(U_i | S_{i,j_1} \ldots S_{i,j_\mu})$$

(from (4) and (5))

$$\geq H(S_{i,j_1} \ldots S_{i,j_\mu}) - \sum_{l=1}^{\mu} H(S_{i,j_l} | U_i) + H(U_i | S_{i,j_1} \ldots S_{i,j_\mu}) \quad \text{(from (3) and (8))}$$
$$= H(S_{i,j_1} \ldots S_{i,j_\mu}) + H(U_i | S_{i,j_1} \ldots S_{i,j_\mu}) \quad \text{(from 1. of Definition 7.1)}$$
$$\geq H(S_{i,j_1} \ldots S_{i,j_\mu}) \quad \text{(from (2))}$$
$$= H(S_{i,j_1}) + H(S_{i,j_2} | S_{i,j_1}) + \cdots + H(S_{i,j_\mu} | S_{i,j_1} \ldots S_{i,j_{\mu-1}}) \quad \text{(from (3))}$$
$$= H(S_{i,j_1}) + H(S_{i,j_2}) + \cdots + H(S_{i,j_\mu}) \quad \text{(from Lemma 4.1)}$$
$$= \mu H(S)$$

□

Analogously to KDSs, in t-conference KDS we can consider the case when not all the t-tuples of users need to set up a common key. Let $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a set of users. A *t-communication structure* C_t is a subset of $\mathcal{U}^t$. The communication structure contains all t-tuple of users for which the protocol has to provide a conference key. A convenient way to represent a t-communication structure is by an hypergraph H in which each vertex U_i corresponds to user U_i, and there is a hyperedge $(U_{i_1}, \ldots, U_{i_t})$ if and only if $(U_{i_1}, \ldots, U_{i_t}) \in C_t$. We will call the hypergraph associated with a t-communication structure the *communication hypergraph*. Definition 7.1, the previously described protocol, and Theorem 7.2 can be extended to a key distribution scheme for any t-communication structure C_t.

8 Applications: Authentication and Master Keys

The polynomial-based scheme proposed applies to settings where a limited coalition of up to a certain security parameter k of adversaries are expected. A basic application is a secure conference key generation. The setting is ideal for the case of a master key generation (to derive further temporal keys), or authentication of conference members based on conventional cryptosystems using the key in authentication protocols (such as the ones described in [1]) and without the need of going to an on-line server (as in [19]). For authentication applications it has a necessary and elegant feature as it connects the IDs of parties to the authentication master key (an ID-based authentication method). Further, additional authenticating information can be attached (as explained in the following sub-sections). The advantage of the system ¿from operational point of view is the disposal of the necessity to contact an on-line remote server, the alternative cost is, naturally, the on-line key computation (evaluation) cost, (this can be somewhat reduced if keys are cached).

8.1 Mixed User Groups

It may be desired to have an asymmetric protocol where the two parties should not be considered equal. For example, one party is a server, and the other a client (e.g., a server-user model). The protocol, in this case, will not only authenticate the name of the user (say), but also the fact that it is an entity with a status of user (rather than a server); users will not be able to claim to be servers. In this case we can modify the scheme to use asymmetric polynomials. This asymmetric scheme can be used to define status (type) of users in various security domains.

8.2 Two-level hierarchical polynomial

Another use for the scheme is for a hierarchical transfer of trust. This can be done either in the symmetric or in the asymmetric polynomial methods. Let us demonstrate here a two level hierarchy of authority servers (domains) and users. The system's polynomial has four (sets of) variables $Q(x,y,z,w)$. $Q(x,y,\cdot,\cdot) = Q(y,x,\cdot,\cdot)$ and $Q(\cdot,\cdot,z,w) = Q(\cdot,\cdot,w,z)$. The first half of variables are to be evaluated under the servers' names and the later half to be evaluated under the users' names. This gives an identification of both the user and its domain (server) in an authentication process. This can be extended to a few levels.

the symmetric polynomial

8.3 Uses for internetworking

In an inter-enterprise environment, using the above method — an organization (company) can issue permits (authentication polynomials) to its own employees, without knowing the main polynomial. Whenever an employee of this company uses the network, it is clear that he indeed has received its authorization ¿from that company (since he must send C, otherwise he will not be able to authenticate itself). Moreover, if it is desired to revoke the permit of this company, it is not necessary to revoke the permit of each of its employees separately, rather revoke the server's authorization and eliminate the right to its users.

8.4 Additional control variables

A multi-variate polynomial may have additional uses. Additional meanings can be assigned to a few additional variables, for example:

- Time-stamp: The polynomial can be evaluated at a specific date by the distributor. The entity using it will have to specify the date it received it (otherwise it will not be able to generate to authentication key). Thus , validity and expiration can easily be decided.
- Group membership: Members of a specific group will be given private polynomials evaluated also under the name of the group (while others will be given the polynomial evaluated under the names of other groups).
- Permission to access a certain resource for access-control mechanism can be embedded in the private polynomial computation.

To conclude, we have modeled, analyzed, and designed dynamic optimal conference key distribution schemes, presented the advantage of interaction in this setting, and presented modifications and essential applications.

References

1. R. Bird, I. Gopal, A. Herzberg, P. Jansen, S. Kutten, R. Molva and M. Yung *Systematic Design of Two-Party Authentication,* Advances in Cryptology: Proceedings of Crypto 91, Lecture Notes in Computer Science, vol. 576, Springer-Verlag, Berlin, 1991.
2. R. Blom, *An Optimal Class of Symmetric Key Generation Systems,* Advances in Cryptology: Proceedings of Eurocrypt 84, Lecture Notes in Computer Science, vol. 209, Springer-Verlag, Berlin, 1984, pp. 335–338.
3. E. Brickell, P.J. Lee and Y. Yacobi, *Secure Audio Conferencing,* Advances in Cryptology: Proceedings of Crypto 87, Lecture Notes in Computer Science, vol. 239, Springer-Verlag, Berlin, 1987, pp. 418–426.
4. I. Csiszár and J. Körner, *Information Theory. Coding theorems for discrete memoryless systems,* Academic Press, 1981.
5. W. Diffie and M.E. Hellman, *New Direction in Cryptography,* IEEE Transaction on Information Theory, vol. 22, no. 6, December 1976, pp. 644–654.
6. M.J. Fischer, M.S. Paterson and C. Rackoff, *Secure Bit Transmission Using a Random Deal of Cards,* in *Distributed Computing and Cryptography,* AMS, 1991, pp. 173–181.
7. M.J. Fischer and R.N. Wright, *Multiparty Secret Key Exchange Using a Random Deal of Cards,* Advances in Cryptology: Proceedings of Crypto 91, Lecture Notes in Computer Science, vol. 576, Springer-Verlag, Berlin, 1991, pp. 141–155.

8. W. Fumy and M. Munzert, *A Modular Approach to Key Distribution*, Advances in Cryptology: Proceedings of Crypto 90, Lecture Notes in Computer Science, vol. 537, Springer-Verlag, Berlin, 1990, pp. 274–283.
9. R. G. Gallager, *Information Theory and Reliable Communications*, John Wiley & Sons, New York, NY, 1968.
10. L. Gong and D.J. Wheeler, *A Matrix Key-Distribution Scheme*, Journal of Cryptology, vol. 2, 1990, pp. 51–59.
11. R. Impagliazzo and S. Rudich, *Limits on the Provable Consequences of One-Way Permutations*, 21-st STOC proceedings, May 1989, pp. 44–61.
12. K. Koyama and K. Ohta, *Identity-based Conference Key Distribution*, Advances in Cryptology: Proceedings of Crypto 87, Lecture Notes in Computer Science, vol. 239, Springer-Verlag, Berlin, 1987, pp. 175–184.
13. F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error Correcting Codes*, North-Holland, New York, 1988.
14. 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, vol. 239, Springer-Verlag, Berlin, 1987, pp. 185–193.
15. K.S. McCurley, *A Key Distribution System Equivalent to Factoring*, Journal of Cryptology, vol. 1, 1988, pp. 95–105.
16. U. Maurer and Y. Yacobi, *Non-interactive Public-Key Cryptography*, Advances in Cryptology: Proceedings of Eurocrypt 91, Lecture Notes in Computer Science, vol. 547, Springer-Verlag, Berlin, 1991, pp. 498–507.
17. R. C. Merkle, *Secure Communication over Insecure Channels*, Communications of the ACM, vol. 21, Apr. 1978, pp. 294–299.
18. Fiat, Naor; and Alon (personal communication).
19. R. M. Needham and M. D. Schroeder, *Using Encryption for Authentication in Large Networks of Computers*, Communications of the ACM, vol. 21, Dec. 1978, pp. 993–999.
20. E. Okamoto and K. Tanaka, *Key Distribution System Based on Identification Information*, IEEE Journal on Selected Areas in Communications, vol. 7, no. 4, May 1989, pp. 481–485.
21. A. Shamir, *Identity-based Cryptosystems and Signature Scheme*, Proceedings of Crypto 84, pp. 47–53.
22. D.G. Steer, L. Strawczynsji, W. Diffie and M Wiener, *A Secure Audio Teleconferencing System*, Advances in Cryptology: Proceedings of Crypto 89, Lecture Notes in Computer Science, vol. 403, Springer-Verlag, Berlin, 1990, pp. 518–528.
23. S.. Tsujii and J. Chao, *A New ID-based Key Sharing Scheme*, Advances in Cryptology: Proceedings of Crypto 91, Lecture Notes in Computer Science, vol. 576, Springer-Verlag, Berlin, 1991, pp. 288–299.
24. Y. Yacobi, *A Key Distribution Paradox*, Advances in Cryptology: Proceedings of Crypto 90, Lecture Notes in Computer Science, vol. 537, Springer-Verlag, Berlin, 1990, pp. 268–273.
25. Y. Yacobi and Z. Shmueley, *On Key Distribution Systems*, Advances in Cryptology: Proceedings of Crypto 89, Lecture Notes in Computer Science, vol. 435, Springer-Verlag, Berlin, 1990, pp. 344–355.