Dynamic Threshold Cryptosystems *

(A New Scheme in Group Oriented Cryptography)

Hossein Ghodosi
Josef Pieprzyk
Rei Safavi-Naini

Center for Computer Security Research
Department of Computer Science
University of Wollongong
NSW, 2522
Australia
e-mail: hossein/josef/rei@cs.uow.edu.au

Abstract

A new concept in group oriented cryptography – *the dynamic threshold cryptosystem* – is presented in the paper. The dynamic system has many advantages over the existing threshold cryptosystems. It is set up by the sender who can not only compose the group of recipients but select the threshold as well. Implementations of the dynamic system based on the ElGamal and the RSA public key cryptosystems are given and their properties are discussed. The proposed solutions are practical and non-interactive. There is no need for trusted dealers or tamper proof modules. The generalization of the concept for an arbitrary access structure concludes the paper.

1 Introduction

Cryptography provides tools for the implementation of secure services that guarantee protection of legal rights of individuals and groups. In general, implementations of such services become more complex for groups than for individuals as groups can exhibit different structures and richer relations among participants. Clearly, the security requirements can vary widely depending upon the size of the group and its internal structure.

*Support for this project was provided in part by the Australian Research Council under the reference number A49530480 and the ATERB grant

Desmedt [3] considered a group as an entity who can carry out cryptographic operations in similar way to an individual. A new field – the group-oriented cryptography appeared. The first proposed group oriented cryptosystem, which was based on mental games [12], was introduced by Desmedt [3]. Later Desmedt and Frankel [5] discussed practical non-interactive solutions for group oriented cryptography. In their solutions, the co-operation of t or more participants from a group of n members ($t \leq n$) was necessary to obtain the plaintext (message) from a cryptogram. This was called a (t, n) *threshold cryptosystem.*

To set up a (t, n) threshold cryptosystem, there is a need to distribute secret shares among the members of a group. The distribution of shares can be done by a trusted party (see [5], [6]) or collectively using verifiable interaction by the members of the group (see [14]). The main drawback of threshold cryptosystems with a trusted party is that participants may not be able to agree on who can be trusted. On the other hand, threshold cryptosystems with collective generation of secret elements can work only when a majority of participants is honest. Moreover, participants have to interact to verify the validity of the secret. In all threshold cryptosystems existing so far, once a system is set up it is impossible to change its parameters – it remains static until the next regeneration of secret elements.

The paper addresses the confidentiality problem when a single sender would like to communicate a message to a group of participants in such a way that only an authorized subset of the group can read the message. The proposed solutions are based on the assumption that all participants have their individual public key cryptosystems already set up and there is a public registry (*"White Pages"*) which holds the authentic public elements of the cryptosystems. The sender creates the dynamic system from the public information available from the public registry.

The proposed dynamic schemes are practical and non-interactive. There is no need for trusted parties or dealers (except for the public registry). The schemes are based on the ElGamal [8] and the RSA [15] public key cryptosystems.

2 Background

Unlike the classical cryptography, the group-oriented cryptography allows groups to carry out cryptographic operations. Of course, members of a group still perform their own cryptographic transformations independently. However, to obtain the final (hopefully correct) result, they need to co-operate by passing their partial computations to a so-called combiner. If a subset of co-operating members is authorized, the combiner can successfully recover the correct final result.

The collection of all authorized subsets is called *the access structure.* A particularly simple access structure is when every t out of n ($t \leq n$) participants are authorized and any smaller number of participants does not belong to the access structure. Systems with these access structures are called *(t,n) threshold systems.*

Any threshold cryptosystem has to have a threshold secret sharing scheme incorporated in it. The concept of secret sharing was independently introduced by Blakley [2] and Shamir [16]. Their constructions allow to recreate the secret if the number of

co-operating participants is equal to or bigger than some fixed number t. The first non-interactive threshold decryption of ElGamal cryptosystem was proposed by Desmedt and Frankel [5]. Pedersen [14] showed that there is no need for a trusted party in order to construct the ElGamal threshold cryptosystem.

The construction of threshold decryption based on the RSA cryptosystem seemed to be impossible [5]. Frankel [9] attempted to solve the problem but his solution was not practical (as it needed the help of trusted clerks). Later Desmedt and Frankel [6] proposed a solution to the problem. Laih and Harn [11] presented a shared decryption scheme based on the RSA cryptosystem, but in some cases their scheme is time consuming. A good review of literature in this area can be found in [4].

Note that for the majority of cryptosystems published so far, senders do not know whether the recipient is an individual or a group. Hwang [10] presented a cryptosystem in which the sender knows the participants. Unfortunately, the proposed scheme does not work well with threshold systems. As mentioned in [10], both conventional and public key cryptosystems are applicable in this scheme and therefore one can claim that a share decryption based on conventional cryptosystems (like the DES) is possible (while no such scheme has been proposed so far).

3 Shortcomings of existing threshold cryptosystems

In this section a list of common shortcomings of the existing threshold cryptosystems is presented. Some of the listed drawbacks have been partially eliminated in some schemes. For example, Pedersen [14] has proposed a version of the ElGamal threshold cryptosystem [5] in which there is no need for a trusted dealer, but that scheme suffers from the other drawbacks listed below.

1. Secure channels need to exist to transmit initial conditions (shares). In some circumstances secure channels may not be available or may introduce an unacceptable delay.

2. A trusted party (dealer) is necessary to generate and distribute shares among the participants. Although in [14] it is shown that there is no need for a trusted dealer in ElGamal threshold cryptosystems, the RSA threshold cryptosystem still needs a trusted party.

3. The membership of the group is static. Once a (t, n) threshold cryptosystem is set up, the sender cannot modify the access structure of the system. If there is a danger of conspiracy of some members of the group, the dealer has to predict this before the setup of the system and perhaps select a big enough threshold parameter t.

4. The parameters of the system are fixed. As discussed in [3], a (t, n) threshold cryptosystem is not flexible enough to handle emergency cases when everyone in the group is expected be able to retrieve the message from a cryptogram.

5. Multiple application of threshold schemes when the sender wants to transmit messages to several groups.

6. Participants need to keep secret elements for all groups (corresponding cryptosystems) to which they belong.

7. Actions performed by the currently active authorized subset are anonymous. The action can be attributed to the group only. The members who did act, stay anonymous. This problem can be crucial in group signature schemes when we would sometimes prefer to be able to identify the members of the signing group.

8. Difficulties with securing secrets (shares) of the threshold scheme. There is no protection against an illegal disclosure of shares by either a careless (although trusted) dealer during the transmission via sometimes insecure channels, or perhaps via a masquerading false dealer who interferes with original shares (the secure channel has to allow to detect illegal modifications of shares). The authenticity of shares becomes crucial when there is no trusted dealer in the system (see [14]). In such cases, there is no guarantee that the system has not been introduced to other parties. The two groups may not be aware about the existence of each other.

4 Model of dynamic threshold cryptosystems

We assume that a dynamic threshold cryptosystem satisfies the following conditions:

- there is no need for a trusted party (dealer) to set up the system,
- the system is set up by individual participants independently (there is no need to transmit shares via secure channels),
- each participant generates a pair of elements: the secret key and the public key. There is a trusted public registry ("White Pages") which keeps the list of authentic public keys of all potential participants,
- the group of receivers (who are able to process cryptogram) and the threshold parameter t are chosen by the sender (the group and the system parameters are set up at the time of transmission),
- cryptograms created by the sender can be decrypted (correctly) by a subgroup of at least t co-operating participants.

Suppose $\mathcal{P} = \{P_1, \ldots, P_m\}$ is a collection of all participants. Every participant announces their public key and keeps the other secret. As a matter of fact, there should be a way to verify the authenticity of public keys by setting up a trusted public registry (White Pages). The registry keeps the list of (authentic) public keys for all participants.

A sender S may or may not belong to $\mathcal{P}$. It is at the sender's discretion to create a group $G \subset \mathcal{P}$ of participants who are able to decrypt (collectively) a cryptogram generated by the sender. The group G retrieves the message only if the number of co-operating participants is equal to or bigger then the threshold t. The parameter t is chosen by the sender.

The notations we use are: $\mathcal{M}$ is the message set, $\mathcal{C}$ is the cryptogram set, $\mathcal{E}$ is the set of encryption keys, $\mathcal{D}$ is the set of decryption keys, and $\mathcal{S}$ is the set of partial decryptions.

Definition 4.1 *A dynamic threshold cryptosystem is a collection of two algorithms:*

1. *sender who composes the group of intended receivers G (for simplicity we assume that $G = \{P_1, \ldots, P_n\}$), selects the threshold parameter t, collects authentic public keys $\{e_1, \ldots, e_n\} \in \mathcal{E}^n$ of participants of the group G from White Pages, and applies the encryption function*

$$f_E : \mathcal{M} \times \mathcal{E}^n \to \mathcal{C}$$

The cryptogram $C \in \mathcal{C}$ is broadcast to all participants $\mathcal{P}$.

2. *combiner who collects partial decryptions $f_D(C, d_i) \in \mathcal{S}$ from a subgroup $G_\ell \subset G$ (for the sake of simplicity assume that $G_\ell = \{P_1, \ldots, P_\ell\}$) and decrypts the cryptogram as*

$$f_{COM} : \mathcal{S}_1 \times \ldots \times \mathcal{S}_\ell \to \mathcal{M}$$

where $\mathcal{S}_i$ is the set of partial decryptions for P_i ($i = 1, \ldots, \ell$). The decryption is always successful if the number of co-operating participants (members of G_ℓ) is equal to or larger than the threshold ($\ell \geq t$) and $G_\ell \subset G$.

Next we are going to discuss some possible implementations of dynamic threshold cryptosystems.

5 ElGamal system

The ElGamal system [8] is a public key cryptosystem which can be seen as an extension of the Diffie-Hellman public key distribution system [7]. Its security relates to the difficulty of the discrete logarithm problem (an overview of algorithms for the discrete logarithm problem is given in [13]). All computations in the ElGamal cryptosystem are done in $GF(p)$ where p is a large enough prime. The designer R of the systems (R is also the receiver) selects randomly a primitive element $g \in GF(p)$, chooses randomly a nonzero integer $s \in GF(p)$ and computes $y = g^s \bmod p$. Next R publishes g, p, and y as the public parameters of the system and keeps s secret. Now, suppose that the sender S wants to communicate a message M to R, where $M \in GF(p)$. To do this, S chooses a random integer $k \in GF(p)$, computes the cryptogram $C = (g^k, My^k)$, and sends it to R. In order to decipher the cryptogram C, R needs to raise g^k to the power s (to get y^k). Then R applies the multiplicative inverse y^{-k} to decrypt the cryptogram and retrieves the message

$$My^k \times y^{-k} = M \bmod p.$$

Desmedt and Frankel [5] constructed an ElGamal threshold cryptosystem which combines the ElGamal system [8] with the Shamir secret sharing scheme [16]. In their construction any subset B of t participants can recalculate the secret $s = \sum_{i \in B} s_i \bmod \phi(p)$ where s_i is the (modified) share of the participant P_i and $\phi(p) = p - 1$ is a large enough *Mersenne* prime ($p = 2^m$). To decrypt the cryptogram, every $P_i \in B$ calculates $(g^k)^{s_i} \bmod p$ and sends the result to the combiner. Using t values of g^{ks_i}, the combiner computes $y^k = g^{ks} = \prod_{i \in B} g^{ks_i}$ and then decrypts the message (using the multiplicative inverse y^{-k}).

Since the prime number p is public, we assume that all ElGamal cryptosystems are defined over the same field $GF(p)$.

In the final version of this paper we will consider the case when ElGamal public key cryptosystems are constructed over different Galois fields.

5.1 ElGamal dynamic (n, n) threshold cryptosystem

Let a group G of n participants agree to use an ElGamal public key cryptosystem over $GF(p)$ with a primitive element g where $p = 2^u$ and $p - 1$ is a Mersenne prime (*in the final version of the paper we will consider the case when primitive elements are different*). Let $d_1, d_2, \ldots, d_n$, $d_i \neq 0$ be the secrets and $e_1 = g^{d_1}, e_2 = g^{d_2}, \ldots, e_n = g^{d_n}$ be the corresponding public keys associated with n members of the group G. Assume that for any two distinct non empty subsets A and B of the public keys the equation

$$\prod_{i \in A} e_i \neq \prod_{j \in B} e_j \bmod p$$

holds where $A, B \subset G$.

A sender (who is not the member of G) computes the public key of the group G as

$$E = \prod_{i=1}^{n} e_i = g^{\sum_{i=1}^{n} d_i} \bmod p$$

and broadcasts the cryptogram

$$C = (g^k, ME^k)$$

for a message $M \in \mathcal{M} = GF(2^u)$. The element k is secret and known to the sender only. Note that the sender uses (n, n) threshold system. Clearly, the secret key d which corresponds to the public key E of the group G, is unknown unless all participants of the group G reveal their secrets d_i.

A combiner who acts on behalf of G, asks the participants for their partial decryptions of the form $(g^k)^{-d_i}$ for $i = 1, 2, \ldots, n$. After receiving all of them, the combiner computes

$$\prod_{i=1}^{n} g^{-kd_i} = (\prod_{i=1}^{n} g^{d_i})^{-k} = E^{-k} \bmod p$$

and recovers the message as $ME^k \times E^{-k} = M \bmod p$.

The sender is able to choose a group G arbitrarily from all participants by a proper selection of their public keys. The main drawback of this system is that the threshold $t = n$ and all participants of the group G have to co-operate (be present) for a successful decryption.

5.2 ElGamal dynamic (t, n) threshold cryptosystem

Suppose a sender wants to communicate a message $M \in \mathcal{M}$ to a group G such that any t out of n subset of participants can decrypt the cryptogram. We are going to modify ElGamal scheme discussed above. Let $B_1, B_2, \ldots, B_q$ be all possible t out of n subsets of

the group G and let $E_1, E_2, \ldots, E_q$ be the corresponding public keys of these sets, that is $E_i = \prod_{P_j \in B_i} e_j$. The sender then chooses at random $k_1, k_2, \ldots, k_q$ $(0 < k_i < (p-1))$ and calculates the cryptogram

$$(g^{k_1}, M(E_1)^{k_1}, g^{k_2}, M(E_2)^{k_2}, \ldots, g^{k_q}, M(E_q)^{k_q}).$$

Now, every subset of t participants can use the relevant part of the cryptogram and decrypt the message. The number of components in the cryptogram relates to the number of subsets of t participants out of n and is equal to:

$$q = \left(\begin{array}{c} n \\ t \end{array} \right) = \frac{n!}{t!(n-t)!}$$

It is obvious that this solution is not practical as the cryptogram tends to be long (and highly redundant) when the parameters t and n are growing. Even for modest parameters such as $t = 6$ and $n = 10$, the cryptogram would have 210 pairs.

5.3 Practical ElGamal dynamic (t, n) threshold cryptosystems

Consider the decryption algorithm of the ElGamal cryptosystem. It requires the receivers to take the first element of cryptogram g^k and compute $e^k = (g^k)^d$ using the secret $d = \sum_{i=1}^{n} d_i$. The multiplicative inverse of the result can be used to retrieve the message. It is clear that an outsider who does not know the secret d but knows the integer k, can also recover the message, e.g., by applying the public key of the participant and the known integer k.

Assume that the sender wants to send an encrypted message to a group G such that every t out of n subset of participants is able to decrypt the cryptogram (both the group G and the threshold parameter t are chosen by the sender). First the sender selects at random a polynomial $f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{t-1} x^{t-1}$ (of degree $t-1$) such that $f(0) = a_0 = k$ and computes the initial conditions $c_i = f(x_i)$ for $i = 1, 2, \ldots, n$. Next the sender creates the cryptogram $(g^k, ME^k, t, c_1 e_1^k, c_2 e_2^k, \ldots, c_n e_n^k)$ and broadcasts it to the group $(E = g^d = \prod_{i=1}^{n} e_i)$.

Suppose that an authorized subset B $(|B| = t)$ of participants wants to decrypt the cryptogram. First each participant P_i $(P_i \in B)$ retrieves their initial condition from the relevant part of the cryptogram, that is, $c_i e_i^k$ (using the multiplicative inverse of $(g^k)^{d_i} = e_i^k$). Next as in the ElGamal threshold cryptosystems, every participant calculates their modified share k_i as

$$k_i = c_i \prod_{\substack{j \neq i \\ j \in B}} \frac{(-x_j)}{(x_i - x_j)} \bmod \phi(p).$$

Note that the set B has to be known to all its members. Clearly, $\sum_{i \in B} k_i = k \bmod \phi(p)$. Now, they compute their partial results $E^{k_i} \bmod p$ and send them to the combiner. The combiner multiplies all partial results

$$\prod_{\substack{i=1 \\ i \in B}}^{t} E^{k_i} = E^{(\sum_{i \in B} k_i)} = E^k \bmod p$$

and then retrieves the message using the multiplicative inverse of the result.

Note that the secret sharing which is applied in this scheme is used to distribute the random value of k among the intended participants of the group (using the ElGamal public key cryptosystem). Only those who know corresponding secret keys can retrieve shares c_i and later decrypt the cryptogram.

5.4 Security of ElGamal dynamic threshold cryptosystems

A cryptogram generated by the system contains two pieces of information. The first piece (the first pair) carries a proper cryptogram. The second gives additional information about the shares of the secret element k. It is clear that the decryption algorithm for the first pair of elements in the cryptogram is as secure as the original ElGamal cryptosystem (see [8]) and therefore the security of the scheme is equivalent to the difficulty of solving the appropriate instance of the discrete logarithm problem.

On the other hand, each element of the second piece of information $c_i e_i^k$ is in fact the cryptogram of the message c_i which has been sent to participant P_i. If an opponent can infer any information about either c_i (or k_i) from the elements given in the cryptogram (or partial decryptions), then he/she is able to get the same information from the threshold ElGamal cryptosystem (see [5]). In other words the security of the scheme is equivalent to the security of the ElGamal public key cryptosystem, and therefore, is equivalent to the difficulty of solving of the corresponding instance of the discrete logarithm problem.

6 RSA system

The RSA public key cryptosystem [15] uses the modulus $N = pq$ where p, q are strong primes i.e. $p = 2p' + 1$ and $q = 2q' + 1$ (p' and q' are large and distinct primes). The creator R of the system generates two strong primes p and q and selects a random integer e such that e and $\lambda(N)$ are co-prime ($\lambda(N)$ is the least common multiplier of two integers $p-1$ and $q-1$, so $\lambda(N) = 2p'q'$). Next R publishes e and N as the public parameters of the system, while keeps the values of p, q, and d secret (d has the property that $ed = 1 \bmod (\lambda(N))$). It is clear that computing d is easy for R who knows $\lambda(N)$ but is difficult (equivalent to the factorization of N) for someone who does not know $\lambda(N)$.

Suppose an individual S wants to send a message M to R, where $0 \leq M < N$. S computes the cryptogram $C = M^e \bmod N$ and communicates it to R. Since for every integer a ($gcd(a,p) = 1$ and $gcd(a,q) = 1$) the equation $a^{\lambda(N)} = 1 \bmod N$ holds, R can decrypt the cryptogram as follows:

$$C^d = (M^e)^d = M^{ed} = M^{1(\bmod \lambda(N))} = M \bmod N$$

6.1 RSA dynamic $(1, n)$ threshold cryptosystem

Consider a group G of n participants $P_1, P_2, \cdots, P_n$ each with their own RSA public key cryptosystem. Their public keys e_i and moduli N_i are stored in a trusted public registry. The keys d_i are secret elements known to their owners only. Assume further that $N_i = p_i q_i$ where p_i, q_i are strong primes and $e_i d_i = 1 \bmod (\lambda(N_i)$ where $\lambda(N_i) = 2p_i'q_i'$ ($p_i = 2p_i' + 1$

and $q_i = 2q_i' + 1$). All pairs of N_i, N_j are co-prime for $i \neq j$ $(i, j = 1, 2, \ldots, n)$ and without the loss of generality, we assume that $N_1 < N_2 < \cdots < N_n$.

Suppose that a sender S (who is not a member of G) wants to communicate a message $0 \leq M < N_1$ to the group G. The sender computes the public key

$$E = \prod_{i=1}^{n} e_i$$

and the modulus of the group $N = N_1 \times N_2 \times \cdots \times N_n$ (both computations are done in Z). Then S broadcasts the cryptogram $(N, C = M^E \bmod N)$ for the message M.

Although there is an unknown integer D such that $E \times D = 1 \bmod \lambda(N)$, no member of the group G can solve this congruence, as $\lambda(N)$ is unknown (P_i knows the value of $\lambda(N_i)$ only). However each participant P_i can solve the following congruence:

$$E \times D_i = 1 \bmod \lambda(N_i).$$

Having D_i, the participant P_i can retrieve the message because

$$C^{D_i} = M^{(E \times D_i)} = M \bmod N_i.$$

Note that the sender creates the group from all participants whose RSA cryptosystems are registered in White Page registry. Parameters of dynamic threshold cryptosystems depend on the current group and can be easily computed from White Page entries of all participants.

6.2 RSA dynamic (t, n) threshold cryptosystems

Suppose a sender wants to communicate a message $0 \leq M < \prod_{i=1}^{t} N_i$ to a group $G = \{P_1, \ldots, P_n\}$ such that any t out of n participants can decrypt the cryptogram. First the sender computes $m_i = M \bmod (N_i)$ for $i = 1, \ldots, n$. Next the sender selects at random an odd integer k and a polynomial $f(x) = a_0 + a_1 x + \cdots + a_{t-1} x^{t-1}$ in $GF(p)$, $p < N_1$ such that $f(0) = a_0 = k$. Although the prime number p has to be bigger than the integer k, it is possible to keep p as small as desired but the field $GF(p)$ has to be large enough so the guessing of k is not "easy". Then the sender calculates $c_i = f(x_i)$, $i = 1, \ldots, n$ for public x_i. Finally the sender composes and broadcasts the cryptogram

$$(N, p, t, (c_1^{e_1}, m_1^k) \bmod N_1, \ldots, (c_n^{e_n}, m_n^k) \bmod N_n).$$

Every t participants of the group G can co-operate to recreate the unknown exponent k. To do this each participant P_i of an authorized subset $B \subset G$ computes $c_i = (c_i^{e_i})^{d_i} \bmod N_i$ and broadcasts it. Note that no participant can cheat as the broadcast c_i can be readily verified by everyone. The integer k can be computed by all - the knowledge of k does not help an attacker to gain any information about the message. Now, participants of the authorized group B know k and their m_i^k. So, every participant retrieves their partial message m_i and sends it to the combiner. Using the Chinese Remainder Theorem, the combiner can recover a unique message M from t partial messages m_i.

6.3 Efficiency of the scheme

Consider the most practical and efficient existing RSA (t, n) threshold cryptosystem which was proposed by Desmedt and Frankel [6]. In their system, the number of exponentiations which is required to generate the cryptogram for a message $0 \leq M < N_0$ and to recover the message from the cryptogram, is $t + 1$ (we do not count computations involved in the share handling – for details see [6]). In our RSA dynamic threshold cryptosystem, the number of exponentiations which is necessary to send a message $0 \leq M < \prod_{i=1}^{t} N_i$, is *at most* $2n + 2t$. However, if we consider the fact that messages in our scheme are approximately t times longer than messages of an ordinary RSA threshold cryptosystem, the number of exponentiations which is necessary to transmit messages of the same size is $(2n + 2t)/t$. If we assume that $n < 2t$ (this is true in most applications), then the number of exponentiations (for sending a message of the size of a single modulus) is always less than 6 (which is not greater than $t + 1$ when $t > 4$). In other words, the RSA dynamic (t, n) threshold cryptosystem is more efficient than existing RSA threshold cryptosystems (for $t > 4$).

Threshold	Ordinary RSA	Dynamic RSA
(4,4)	5	4
(4,5)	5	18/4=4.5
(4,6)	5	5
(4,7)	5	5.5
(5,9)	6	28/5=5.6
⋮	⋮	⋮
(10,15)	11	5

The above table compares the two systems. Entries in two last columns are numbers of "elementary" exponentiations.

6.4 Security

To break the RSA dynamic cryptosystem, attackers need to find t (out of n) partial messages so they are able to compose (by the Chinese Remainder Theorem) the original message. But this is equivalent to "breaking" t RSA public key cryptosystems.

7 Generalization of dynamic cryptosystems

Dynamic cryptosystems for a general access structures Γ over the set $\mathcal{P} = \{P_1, \ldots, P_n\}$ can be designed using dynamic threshold cryptosystems. A possible realization would involve cumulative arrays [17]. Roughly speaking, cumulative arrays distribute m shares among n participants in such a way that every subset $A \in \Gamma$ is able to recreate all m shares. Any subset of participants $A \notin \Gamma$ cannot. To design a dynamic system for Γ, we first find a corresponding cumulative array. Next we create a (m, m) dynamic cryptosystem with pieces of cryptogram addressed to participants according to the cumulative array. This approach can be used when we know how to construct dynamic (m, m) threshold

cryptosystems. The main disadvantage of this construction is that, in general, it may not be "optimal" (*more details will be given in the final version of the paper*).

Another approach could be based on a secret sharing scheme which can be specifically designed for Γ and later converted to a dynamic cryptosystem. Note that all "linear" secret sharing schemes can be converted into cryptosystems based on exponentiation (such as ElGamal and RSA). Apart from the Shamir secret sharing scheme, there is also a scheme based on linear codes invented by Bertilsson and Ingemersson [1].

The implementation of dynamic cryptosystems using secret sharing schemes based on linear codes will be discussed in the final version of the paper

References

[1] M. Bertilsson and I. Ingemarsson. A construction of parctical secret sharing schemes using linear block codes. *Advances in Cryptology - AUSCRYPT'92, Lecture Notes in Computer Science, J. Seberry and Y. Zheng (Eds)*, 718:67–79, 1993.

[2] G. R. Blakley. Safeguarding cryptographic keys. Proceedings of AFIPS 1979 National Computer Conference, 1979.

[3] Y. Desmedt. Society and group oriented cryptography: a new concept. In *Advances in Cryptology, Proceedings of CRYPTO'87, Ed C. Pomerance, Lecture Notes in Computer Science, Vol.293*, pages 120–127. Springer-Verlag, 1988.

[4] Y. Desmedt. Threshold cryptography. *ETT*, 5(4):449–=457, August 1994.

[5] Y. Desmedt and Y. Frankel. Threshold cryptosystems. In *Advances in Cryptology, Proceedings of CRYPTO'89, Ed G. Brassard, Lecture Notes in Computer Science, Vol.435*, pages 307–315. Springer-Verlag, 1990.

[6] Y. Desmedt and Y. Frankel. Shared generation of authenticators and signatures. In *Advances in Cryptology, Proceedings of CRYPTO'91, Ed J. Feigenbaum, Lecture Notes in Computer Science, Vol.718*, pages 457–469. Springer-Verlag, 1992.

[7] W. Diffie and M.E. Hellman. New directions in cryptography. *IEEE Trans. Inform. Theory*, IT-22(6):644–654, November 1976.

[8] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, IT-31:469–472, 1985.

[9] Y. Frankel. A practical protocol for large group oriented networks. In *Advances in Cryptology, Proceedings of EUROCRYPT'89, Eds J-J. Quisquater and J. Vanderwalle, Lecture Notes in Computer Science, Vol.434*, pages 56–61. Springer-Verlag, 1990.

[10] T. Hwang. Cryptosystem for group oriented cryptography. In *Advances in Cryptology - EUROCRYPT'90, I. Damgard (ED), Lecture Notes in Computer Science, Vol.473*, pages 353–360. Springer Verlag, May 1990.

[11] C-S. Laih and L. Harn. Generalized threshold cryptosystems. In *Advances in Cryptology - ASIACRYPT'91, Lecture Notes in Computer Science (H. Imai, R.L. Rivest and T. Matsumoto (Eds))*, pages 159–166. Springer-Verlag, 1993. Fujiyoshida, Yamanashi, Japan, November 11-14, 1991.

[12] S. Micali, O. Goldreich, and A. Wigderson. How to play any mental game. In *Proceedings of the Nineteenth ACM Symp. on Theory of Computing, STOC*, pages 218–229, May 1987.

[13] A.M. Odlyzko. Discrete logarithms in finite fields and their cryptographic significance. *Proceedings of EUROCRYPT'84, Lecture Notes in Computer Science, Advances in Cryptology, T. Beth, N. Cot, I. Ingemarsson (Eds)*, 209:224–314, 1985.

[14] T. P. Pedersen. A threshold cryptosystem without a trusted party. In *Advances in Cryptology - EUROCRYPT'91, D.W. Davies (ED), Lecture Notes in Computer Science, Vol.547*, pages 522–526. Springer Verlag, 1991.

[15] R.L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[16] A. Shamir. How to share a secret. *Communications of the ACM*, 22(11):612–613, November 1979.

[17] G.J. Simmons, W. Jackson, and K. Martin. The geometry of shared secret schemes. *Bulletin of the ICA*, 1:71–88, 1991.