A Threshold Cryptosystem without a Trusted Party

(Extended abstract)

Torben Pryds Pedersen
Aarhus University, Computer Science Department

Abstract

In a threshold cryptosystem n members share the secret key of an organization such that k members ($1 \leq k \leq n$) must cooperate in order to decipher a given ciphertext. In this note it is shown how to implement such a scheme without having a trusted party, which selects the secret key and distributes it to the members. In stead, the members choose the secret key and distribute it verifiably among themselves. Subsequently, this key can be used for authentication as well as secret communication.

1 Introduction

The concept of group oriented cryptography was introduced in [Des88] as a means of sending messages to a group of people, such that only certain subsets of the members are able to read the message (decipher the ciphertext).

The members of a group are said to be *known* if the sender has to know them (a public key for each member), and a group is called *anonymous* if there is a single public key for the group independently of the members. In general, an anonymous group has a much shorter public key than groups with known members, but it is difficult to handle the situation where a member leaves the group, as this usually requires a new secret key to be selected. Desmedt presents crypto-systems for both types of groups in the case where deciphering requires the cooperation of all members.

Group-oriented cryptography has been further studied in [Fra90], [DF90] and [Hwa91]. Frankel used in [Fra90] the organization of individuals in groups to reduce the problem of distributing and managing public keys. His solution required clerks at the sending as well as the receiving organization.

In [DF90], Desmedt and Frankel modified the El Gamal public key cryptosystem [EG85] so that any k members of the organization can decipher the received ciphertext (for anonymous groups), and in [Hwa91] this property was obtained for organizations with known members.

D.W. Davies (Ed.): Advances in Cryptology - EUROCRYPT '91, LNCS 547, pp. 522-526, 1991.

In this paper only anonymous groups are considered. The main purpose is to improve the threshold cryptosystem proposed by Desmedt and Frankel in two ways. First, it is shown how to avoid the trusted party, who selects and distributes the secret key. Secondly, it is shown how to share the secret key (chosen by the members) such that each member of the group can verify, that the share is correct. This property is important as the shares are no longer computed by a trusted party, and therefore it is not reasonable to expect that they are computed correctly.

In the next section some notation is introduced. Then the protocols for selecting and distributing the key are presented, and in Section 4 possible applications of the these keys are presented. Finally a conclusion and an open problem are given.

2 Notation

In this note p and q denote large primes such that q divides $p-1$, G_q is the unique subgroup of $\mathbb{Z}_p^*$ of order q, and g is a generator of G_q. For all elements a and $b \neq 1$ in G_q, the discrete logarithm of a with respect to b is defined and it is denoted $\log_b(a)$. It is easy to test if an element $a \in \mathbb{Z}_p^*$ is in G_q since

$$a \in G_q \iff a^q = 1.$$

3 Selection of the Keys

In this section we show how n members of a group $(P_1, \ldots, P_n)$ can select a public of the form (p, q, g, h), where $g, h \in G_q$ and the corresponding secret key is $x = \log_g h$, such that for a fixed parameter k $(1 \leq k \leq n)$, k members must cooperate in order to use the secret key. It will be assumed that $n \geq 2k-1$. As (k, n)-threshold schemes allow at most $k-1$ cheating participants, this means that a majority of the participants is assumed to be honest. It is not hard to generalize the protocols such that l members select the secret key and distribute it to the n members of the group (where $n \geq l \geq 2k-1$).

It will be assumed that the members of the group have previously agreed on the primes p and q and the generator g of G_q.

3.1 Selecting and Distributing the Keys

Let $C(m, r)$ denote a commitment to $m \in \{0,1\}^*$ using the random string r. Then the keys are selected as follows:

1. P_i chooses $x_i \in \mathbb{Z}_q$ at random (uniform distribution) and computes $h_i = g^{x_i}$. Then a random string r_i is chosen and $C_i = C(h_i, r_i)$ is broadcast to all members.

2. When all n members have broadcast a commitment, each P_i opens C_i.

3. The public key, h, is computed as $h = \prod_{i=1}^{n} h_i$.

Now all members know the public key, but they cannot find the secret key $x = \sum_{i=1}^{n} x_i$ unless they all work together (or some of them can find discrete logarithms).

If P_i chooses x_i at random then the distribution of the secret key is polynomially indistinguishable from the uniform distribution.

Next it is shown how x can be distributed such that any k members can find it if necessary (if $k = 1$ or $k = n$ this is trivial). The proposed method extends the ideas presented in [IS91] to verifiable secret sharing. P_i distributes x_i as follows ($h_1, \ldots, h_n$ are publicly known):

1. P_i chooses at random a polynomial $f_i(z) \in \mathbb{Z}_q(z)$ of degree at most $k-1$ such that $f_i(0) = x_i$. Let

$$f_i(z) = f_{i0} + f_{i1}z + \ldots + f_{i,k-1}z^{k-1}$$

where $f_{i0} = x_i$.

2. P_i computes $F_{ij} = g^{f_{ij}}$ for $j = 0, \ldots, k-1$ and *broadcasts* $(F_{ij})_{j=1,\ldots,k-1}$ ($F_{i0} = h_i$ is known beforehand).

3. When everybody have sent these $k-1$ values, P_i sends $s_{ij} = f_i(j)$ *secretly* and a signature on s_{ij} to P_j for $j = 1, \ldots, n$ (in particular P_i keeps s_{ii}).

4. P_i verifies that the share received from, P_j (s_{ji}) is consistent with the previously published values by verifying that

$$g^{s_{ji}} = \prod_{l=0}^{k-1} F_{jl}^{i^l}.$$

If this fails, P_i broadcasts that an error has been found, publishes s_{ij} and the signature and then stops.

5. P_i computes his share of x as the sum of all shares received in step 3 $s_i = \sum_{j=1}^{n} s_{ji}$. Finally P_i signs h.

When all members have signed h, a key authentication center verifies the signatures, and if they are correct, it makes a certificate showing that h is the public key of the group.

Let f be the following polynomial over $\mathbb{Z}_q$: $f(z) = f_1(z) + \ldots + f_n(z)$. By construction $s_i = f(i)$ for every $i = 1, \ldots, n$, and thus s_i is a share of $f(0) = x$ (see [Sha79]). For each share s_i of x let σ_i denote g^{s_i}. If P_i has received correct shares, then each P_j ($j \neq i$) can compute σ_i as

$$\sigma_i = \prod_{j=1}^{n} g^{s_{ji}} = \prod_{j=1}^{n} (h_j \prod_{l=1}^{k-1} F_{jl}^{i^l}).$$

3.2 Properties of the Key Selection Scheme

The following theorem shows that any group of k people have sufficient information to find x if they have followed the key distribution protocol.

Theorem 3.1
Let $(P_i)_{i\in H}$ be a group of k members who have followed the key distribution protocol and accepted their shares. Let $f'(z) = \varphi_0 + \varphi_1 z + \ldots + \varphi_{k-1} z^{k-1}$ be the (unique) polynomial of degree at most $k-1$ such that $f'(i) = s_i$ for $i \in H$. Then $g^{\varphi_i} = \prod_{j=1}^{n} F_{ji}$ for $i = 0, \ldots, k-1$.

This theorem implies that $\varphi_0 = x$ as

$$\prod_{j=1}^{n} F_{j0} = \prod_{j=1}^{n} h_j = h.$$

During the key distribution protocol P_i publishes much information about x_i. The following lemma shows that this information is of no use to a collusion of l members $(1 \leq l < k)$

Lemma 3.2
Given a group of $l < k$ members $(P_j)_{j\in D}$. For any h_i and any set of shares $(s_{ij})_{j\in D}$, it is possible to generate in polynomial time (in $|q|$) a random set $(F_{it})_{t=1,\ldots,k-1}$ satisfying

$$g^{s_{ij}} = \prod_{t=0}^{k-1} F_{it}^{j^t} \qquad \text{for } j \in D,$$

where $F_{i0} = h_i$.

4 Applications

The keys selected as described in Section 3 can be used for secret communication as well as authentication. Someone knowing the public key can encipher the plaintext $m \in G_q$ as (see [EG85])

$$(c_1, c_2) = (g^y, h^y m) \qquad \text{where } y \in \mathbb{Z}_q^* \text{ is chosen at random.}$$

Any k members can decipher the ciphertext (c_1, c_2) using the deciphering protocol described in [DF90]. Using Lemma 3.2 it can be shown that

Theorem 4.1
The group-oriented crypto-system is as secure as the ElGamal public-key crypto-system against all kinds of attack.

In the full version of the paper it is shown that any k members can represent the organization in an identification scheme and that any k members can construct digital signatures on behalf of the organization. Furthermore, these authentication schemes are proven secure.

5 Conclusion

In this note we have improved the threshold cryptosystem suggested in [DF90] on two points:

1. A trusted party for selecting and distributing the secret is no longer needed.
2. Each member of the group can verify that his share of the secret key corresponds to the public key.

Even though a trusted party is still needed when deciphering (see [DF90]), this paper shows that no trusted party needs to know the secret key of any member.

This paper still leaves open the problem of constructing an efficient anonymous threshold cryptosystem secure against chosen ciphertext attack without any a trusted party at all.

References

[Des88] Y. Desmedt. Society and group oriented cryptography: A new concept. In *Advances in Cryptology - proceedings of CRYPTO 87*, Lecture Notes in Computer Science, pages 120 – 127, 1988.

[DF90] Y. Desmedt and Y. Frankel. Threshold cryptosystems. In *Advances in Cryptology - proceedings of CRYPTO 89*, Lecture Notes in Computer Science, pages 307 – 315, 1990.

[EG85] T. El Gamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In *Advances in Cryptology - proceedings of CRYPTO 84*, Lecture Notes in Computer Science. Springer-Verlag, 1985.

[Fra90] Y. Frankel. A practical protocol for large group oriented networks. In *Advances in Cryptology - proceedings of EUROCRYPT 89*, Lecture Notes in Computer Science, pages 56 – 61. Springer-Verlag, 1990.

[Hwa91] T. Hwang. Cryptosystem for group oriented cryptography. In *Advances in Cryptology - proceedings of EUROCRYPT 90*, Lecture Notes in Computer Science, pages 352 – 360. Springer-Verlag, 1991.

[IS91] I. Ingemarsson and G. J. Simmons. A protocol to set up shared secret schemes without the assistance of a mutually trusted party. In *Advances in Cryptology - proceedings of EUROCRYPT 90*, Lecture Notes in Computer Science, pages 266 – 282. Springer-Verlag, 1991.

[Sha79] A. Shamir. How to share a secret. *CACM*, 22:612–613, 1979.