A Distributed RSA Signature Scheme for General Access Structures

Javier Herranz, Carles Padró and Germán Sáez

Dept. Matemàtica Aplicada IV, Universitat Politècnica de Catalunya
C. Jordi Girona, 1-3, Mòdul C3, Campus Nord, 08034 Barcelona, Spain
e-mail: {jherranz,matcpl,german}@mat.upc.es

Abstract

In a distributed digital signature scheme, a set of participants shares a secret information that allows them to compute a valid signature for a given message. These systems are said to be robust if they can tolerate the presence of some dishonest players.

Up to now, all the proposed schemes consider only threshold structures: the tolerated subsets of corrupted players as well as the subsets of players who can sign a message are defined according to their cardinality.

We propose a framework that is more general than the threshold one, considering a general access structure of players allowed to sign and a general family of dishonest players that the scheme can tolerate. If these general structures satisfy some combinatorial conditions, we can design a distributed and secure RSA signature scheme for this setting. Our construction is based on the threshold scheme of Shoup [30].

Keywords. Distributed digital signatures, RSA signatures, secret sharing schemes.

1 Introduction

The area of distributed cryptography, which is generally known as threshold cryptography, has been very active since it was introduced in the works of Boyd [5], Okamoto [24] and Desmedt [10]. The reason of this liveliness is the increasing number of situations in which an operation has to be made or supervised by more than a single party: transactions between companies, distributed certification authorities, distributed key generation, etc.

In particular, many distributed digital signature schemes have been proposed. In these schemes, some participants share a secret information that enables some subsets of them (those in the access structure) to compute a valid signature. The advantages of collective digital signatures with respect to individual ones are several:

- security increase: in an individual digital signature, an adversary obtains all the secret information if he can corrupt one party. In a distributed scheme, however, an adversary must corrupt a determined group of players to obtain some useful secret information.
- reliability increase: if the only player of an individual scheme is not able to compute the signature, it is not computed in any way. In a collective scheme, although some

participants have technical problems, for example, the rest of players can compute a valid signature.

In order to formalize these aspects, the situation is modelized by an external adversary who can corrupt some subsets of dishonest players (those in the adversary structure). Participants can be dishonest in two different ways: they can try to obtain the secret information necessary to compute signatures without the consent of the honest players but executing the protocol correctly; or they can also try to boycott the process forging their shares of secret information.

In order to tolerate these situations, the scheme must have the properties of *unforgeability* and *robustness*. A scheme is said to be unforgeable if any subset of dishonest players can not obtain any information that allows them to compute a signature without the consent of some honest party. A scheme is said to be robust if it can detect lying participants, and they can't avoid the honest players to generate a valid signature.

It is also desirable that the resulting signature is a standard one; that is, the receiver of the signature can not distinguish if it has been generated in a distributed way or not.

Previous work. The proposals of distributed digital signature schemes made until now can be divided in two groups, according to the individual signature scheme that they use.

With respect to signature schemes based on the discrete logarithm problem, early proposals were made by Desmedt and Frankel [11], but they were not robust and require the presence of a trusted party. Pedersen [26] and Harn [21] avoid the need of a trusted party, but neither do they consider the question of robustness. Langford [22] proposes a robust scheme, but the number of corrupted players tolerated is not optimal (the scheme has not "optimal-resilience"). Finally, Gennaro et al. [19] propose an optimal and robust scheme which does not require a trusted party. Recently, Stinson and Strobl [35] propose the threshold version of Schnorr's signature scheme.

With respect to schemes based on RSA signatures [28], first attempts were also made by Frankel and Desmedt [13] and De Santis et al. [9]; they work in a concrete extension of a polynomial ring, and this fact leads to a scheme that requires either interaction or very large shares. Gennaro et al. [18] complete and make robust the proposal of De Santis et al., but it is still inefficient. Frankel et al. [14], [15] and Rabin [27] propose robust schemes that are besides proactive (they tolerate adaptative adversary structures), but with a high computational complexity. Shoup [30] proposes a robust, non-interactive, efficient and conceptually simple scheme. It requires a dealer to generate the keys and distribute the shares among the players. Finally, Miyazaki et al. [23], Damgård and Koprowski [8] and Fouque and Stern [12] propose robust schemes in which is not necessary the presence of a dealer; this is possible thanks to the works of Boneh and Franklin [4], Catalano et al. [7] and Frankel et al. [16] (this last work, revisited in [2]), who show how can a group of players jointly generate the keys and distribute the shares of a RSA system.

Almost all the proposals we have mentioned have a characteristic in common: the adversary structure is a threshold one, that is, the scheme tolerates the presence of up to t corrupted players. The most logical solution then, in order to get unforgeability, is to define the access structure (those subsets of players who will be able to generate a signature) as a threshold structure, too. In this case, it must contain subsets of at least $t+1$ players. We must remark here that the threshold Schnorr's signature scheme in [35] also works with more general access structures.

Our contribution. We propose a framework which is more general than the threshold one; we consider a general access structure of players that are allowed to sign a

message, and a general family of subsets of dishonest players that the system can tolerate (that is, a general adversary structure). We find the necessary combinatorial conditions for these structures in order to reach robustness and unforgeability in a distributed signature scheme.

We design a scheme that runs in this scenario, generalizing the threshold scheme of Shoup [30]. Our scheme is a first approach to the goal of designing distributed protocols (in this case, for RSA signatures) that run with access structures which are more general than the threshold ones.

We modify standard linear secret sharing schemes, defined over a finite field, and consider them defined in $\mathbb{Z}$. The efficiency of the resulting distributed RSA signature scheme will depend on the access structure. We also give some examples of interesting access structures that are not threshold and for which our scheme provides an efficient distributed signature scheme.

Organization of the paper. In Section 2, we explain some of the tools which are basic to construct and understand our system, as secret sharing schemes (in particular, vector space ones), or the proposal of threshold signature scheme by Shoup [30]. In Section 3, we set a more general framework for the adversary and access structures, we explain our proposal of distributed signature scheme, we prove its security, we discuss its efficiency and we present some examples in which the resulting scheme is very efficient. Finally, in Section 4 we sum up the contribution of our work and the problems which remain open in the area of distributed signature schemes.

2 Preliminaries

2.1 Secret Sharing Schemes

Secret sharing schemes are an important component of distributed cryptography. In these schemes, a secret value is shared among the participants in a set $\mathcal{P}$ in such a way that only qualified subsets of $\mathcal{P}$ can reconstruct the secret from their shares. A secret sharing scheme is said to be *perfect* if the subsets that are not qualified to reconstruct the secret have absolutely no information on it. That is, the security of a perfect secret sharing scheme is *unconditional*. A comprehensive introduction to secret sharing schemes can be found in [33, 34, 31].

An *access structure* Γ on a finite set $\mathcal{P}$ of participants is the family of subsets of $\mathcal{P}$ that are authorized to reconstruct the secret. The access structure has to be *monotone*, that is, if $A_1 \in \Gamma$ and $A_1 \subset A_2 \subset \mathcal{P}$, then $A_2 \in \Gamma$. Therefore, an access structure can be determined by the family of minimal authorized subsets, $\Gamma_0 \subset \Gamma$, which is called the *basis* of Γ.

A secret sharing scheme is said to be *ideal* if the length of the secret is equal to the length of the shares of the participants. There are access structures that can not be realized by an ideal secret sharing scheme; however, there are interesting families of structures which can always be realized by such an ideal scheme.

2.1.1 Linear Secret Sharing Schemes

The *vector space construction* is a useful method to construct ideal schemes that was introduced by Brickell in [6]. Let $\mathcal{P}$ be a set of n participants. Let Γ be an access structure on $\mathcal{P}$ and $D \notin \mathcal{P}$ a special participant called the dealer. Γ is said to be a *vector space access structure* if, for some vector space $E = K^r$ over a finite field K, there exists

a function

$$\psi : \mathcal{P} \cup \{D\} \longrightarrow E$$

such that $A \in \Gamma$ if and only if the vector $\psi(D)$ can be expressed as a linear combination of the vectors in the set $\psi(A) = \{\psi(i) | i \in A\}$. If Γ is a vector space access structure, we can construct an ideal secret sharing scheme for Γ with set of secrets K: given a secret value $k \in K$, the dealer takes a random element $\mathbf{w} \in E$, such that $\mathbf{w} \cdot \psi(D) = k$ and the share of a participant $i \in \mathcal{P}$ is $s_i = \mathbf{w} \cdot \psi(i) \in K$. A scheme constructed in this way is called a *vector space secret sharing scheme*. Let $A \in \Gamma$ be an authorized subset, ; then, $\psi(D) = \sum_{i \in A} c_i^A \psi(i)$, for some $c_i^A \in K$. In order to recover the secret, the players of A compute

$$\sum_{i \in A} c_i^A s_i \;=\; \sum_{i \in A} c_i^A \mathbf{w} \cdot \psi(i) \;=\; \mathbf{w} \cdot \sum_{i \in A} c_i^A \psi(i) \;=\; \mathbf{w} \cdot \psi(D) \;=\; k$$

Vector space secret sharing schemes are a particular ideal case of *linear secret sharing schemes*, which are essentially equal to vector space ones we have explained, but now every participant can be associated with more than one vector. These schemes have been considered under other names such as geometric secret sharing schemes or monotone span programs. Linear secret sharing schemes realize obviously more access structures than vector space ones. In fact, Simmons, Jackson and Martin [32] proved in a constructive way that any access structure Γ can be realized by a linear secret sharing scheme. In general, this construction gives schemes that are inefficient in the sense that the length of the shares is quite larger than the length of the secret.

2.1.2 Shamir's Secret Sharing Schemes

Shamir's secret sharing scheme was introduced in [29] and it realizes threshold access structures, that is $\Gamma = \{A \subset \mathcal{P} \;:\; |A| \geq t\}$, for some threshold t. To share a secret k in a finite field K, the dealer chooses a random polynomial $f(z) = k + a_1 z + \cdots + a_{t-1} z^{t-1} \in K[z]$ of degree $t-1$. The share of participant p_i is $s_i = f(i)$, for $i = 1, \ldots, n$.

Let $A = \{i_1, \ldots, i_t\}$ be a subset of t participants. They have t different values of the polynomial $f(z)$, of degree $t-1$, so they can interpolate the value $k = f(0)$. From the Lagrange interpolation formula, we have

$$k \;=\; f(0) \;=\; \sum_{j \in A} \lambda_{0,j}^A f(j)$$

where $\lambda_{0,j}^A = \dfrac{\prod_{i \in A \setminus \{j\}} -i}{\prod_{i \in A \setminus \{j\}} j-i}$.

Shamir's secret sharing scheme is a particular case of vector space secret sharing schemes, taking $E = K^t$ and ψ defined by $\psi(D) = (1, 0, \ldots, 0)$ and $\psi(p_i) = (1, i, i^2, \ldots, i^{t-1})$.

As we have said in the introduction, all distributed signature schemes proposed until now consider only threshold structures. They use Shamir's secret sharing scheme or some variant of it. If we want to consider more general structures, we should use more general secret sharing schemes, as vector space or linear ones.

2.2 The proposal of Shoup

The distributed digital signature scheme proposed by Shoup [30] is based on the RSA cryptosystem [28]. Let $n = pq$ be the product of two large primes, e an integer $1 < e \leq n$

with $\gcd(e, \phi(n)) = 1$ and $d = e^{-1} \bmod \phi(n)$. The values n and e are public, while the values p, q and d are kept secret.

The signature for a certain message x (which results from applying a hash function h to a text M, $x = h(M)$) is $y = x^d \bmod n$. The receiver verifies $y^e = x \bmod n$.

In the threshold proposal of Shoup, a dealer distributes shares of the value d among a set of participants $\mathcal{P} = \{1, \ldots, \ell\}$ in such a way that subsets of up to t dishonest players are tolerated, and any subset of k participants can generate a valid signature. In order to achieve unforgeability, it is necessary $k \geq t + 1$; to achieve robustness, it is necessary $\ell - t \geq k$. Both inequalities imply $\ell \geq 2t + 1$.

The scheme proposed by Shoup requires some restrictions in the generation of the RSA keys:

- the two primes p and q must be safe primes; that is, $p = 2p' + 1$ and $q = 2q' + 1$, with p' and q' themselves primes. Then, if we define $m = p'q'$, we have $\phi(n) = 4m$.
- the public exponent e must be a prime $e > \ell$ (remember that ℓ is the number of participants), such that $\gcd(e, m) = 1$.

All this work is done by the dealer, who also computes $d = e^{-1} \bmod m$ (and not $\bmod \phi(n)$, as in standard RSA). He chooses at random a polynomial $f(z) = d + a_1 z + \cdots + a_{k-1} z^{k-1}$, of degree $k - 1$, with $a_i \in \{0, \ldots, m - 1\}$. For $1 \leq i \leq \ell$, the dealer computes $s_i = f(i) \bmod m$ (note that m is unknown to the participants). He sends each player his share s_i. The dealer also chooses a random $v \in Q_n$, where Q_n is the subgroup of squares in $\mathbb{Z}_n^*$, and computes $v_i = v^{s_i}$, for $1 \leq i \leq \ell$. He makes public the verification keys v and $\{v_i\}_{1 \leq i \leq \ell}$.

Let $x \in \mathbb{Z}_n^*$ be the message which is wanted to be signed. Each player i broadcasts his signature share $x_i = x^{2\Delta s_i} \in Q_n$, where $\Delta = \ell!$, along with a "proof of correctness", which is basically a non-interactive proof of knowledge that the discrete logarithm of x_i^2 to the base $x^{4\Delta}$ is the same as the discrete logarithm of v_i to the base v (see [30] for the details). Therefore, the rest of participants can detect if the share x_i is inconsistent with the verification keys, and can reject player i if he has lied. Once all dishonest players have been rejected, and because of the restriction $\ell - t \geq k$, there are at least k valid signature shares. We will suppose, without loss of generality, they are corresponding to players in $A = \{1, \ldots, k\}$. Note that the secret value d has been shared with a variant of Shamir's secret sharing scheme, so the idea is to interpolate the value of d. But the operations must be done in $\mathbb{Z}_m$, and m is unknown for the participants. In particular, they can't calculate inverses in $\mathbb{Z}_m$, so they will not be able to compute the interpolation coefficients

$$\lambda_{0,j}^A = \frac{\prod_{i \in A \setminus \{j\}} -i}{\prod_{i \in A \setminus \{j\}} j - i} \bmod m$$

To solve this situation, and since these denominators all divide $j!(\ell - j)!$ which in turn divides $\Delta = \ell!$, we will consider $\Delta \lambda_{0,j}^A$, which are clearly integers. Then, we have

$$\Delta d \; = \; \sum_{j=1}^{k} \Delta \lambda_{0,j}^A s_j \mod m$$

Now the players in A compute

$$\omega = x_1^{2\lambda_{0,1}^A} \cdots x_k^{2\lambda_{0,k}^A} \; = \; x^{4\Delta \sum_{j=1}^{k} \Delta \lambda_{0,j}^A s_j} \; = \; x^{4\Delta^2 d} \mod n$$

(because $x^{\phi(n)} = x^{4m} = 1 \bmod n$, as Fermat's little theorem says). The same property implies that $\omega^e = x^{4\Delta^2} \bmod n$.

Since $e > \ell$ is prime, we have $\gcd(e, 4\Delta^2) = 1$, and the players can obtain from the extended Euclidean algorithm integers a and b such that $4\Delta^2 a + eb = 1$.

Then, the signature for the message x is $y = \omega^a x^b$. In effect, $y^e = \omega^{ea} x^{be} = x^{4\Delta^2 a + eb} = x \bmod n$. Note that the receiver can not distinguish if the signature has been generated by a set of participants or by a single one.

A RSA signature for the message x has been distributely generated, but the value of the secret key, d, has not been recovered anytime. This is convenient, because we do not want to change the value of d each time we make a signature.

We also note that the dealer uses a variation of Shamir's scheme to share a secret over the ring $\mathbb{Z}_m$ (not over a field).

3 A Secure Scheme for General Structures

3.1 General Structures

Now we will explain our proposal for the case of more general structures, not only threshold ones. Let $\mathcal{P} = \{1, \ldots, \ell\}$ be a set of participants. For any family $\mathcal{B} \subset 2^{\mathcal{P}}$ of subsets of $\mathcal{P}$, we denote $\overline{\mathcal{B}} = 2^{\mathcal{P}} \backslash \mathcal{B}$ and $\mathcal{B}^c = \{\mathcal{P} \backslash B \ : \ B \in \mathcal{B}\}$.

Now the tolerated subsets of dishonest players are not necessarily defined according to their cardinality. We have a general *adversary structure*, $\mathcal{A} \subset 2^{\mathcal{P}}$, which must be obviously monotone decreasing: if $B_1 \in \mathcal{A}$ is a tolerated subset of dishonest players, then $B_2 \in \mathcal{A}$ for all $B_2 \subset B_1$. Analogously to the case of monotone increasing structures, we can define the basis $\mathcal{A}_0$ of the adversary structure as the maximal subsets of $\mathcal{A}$, that is, $\mathcal{A}_0 = \{B \in \mathcal{A} \mid B' \notin \mathcal{A}, \text{ for all } B' \supset B\}$.

On the other hand, we have the monotone increasing access structure Γ, that is, the collection of subsets of participants that will be able to generate a valid signature. If we want to achieve unforgeability, any subset in $\mathcal{A}$ can not be in Γ, that is

$$\mathcal{A} \cap \Gamma = \emptyset \tag{1}$$

This condition is the same as $\Gamma \subset \overline{\mathcal{A}}$, or equivalently, $\mathcal{A} \subset \overline{\Gamma}$, where $\overline{\Gamma}$ is the set of non-authorized subsets. Since $\overline{\Gamma}$ is monotone decreasing, we can consider the family $(\overline{\Gamma})_0 \subset \overline{\Gamma}$ of maximal non-authorized subsets.

To achieve robustness, for any $B \in \mathcal{A}$, the subset $\overline{B}$ formed by the rest of players of $\mathcal{P}$ must be in Γ. In other words

$$\mathcal{A}^c \subset \Gamma \tag{2}$$

From these two conditions (1) and (2), it must be $\mathcal{A}^c \cap \mathcal{A} = \emptyset$. We recall that a monotone decreasing structure $\mathcal{A} \subset 2^{\mathcal{P}}$ is said to be $\mathcal{Q}^2$ in $\mathcal{P}$ if $\mathcal{A}^c \subset \overline{\mathcal{A}}$ or equivalently, if there are not two subsets in $\mathcal{A}$ that cover all the set $\mathcal{P}$. In the threshold case, $\mathcal{Q}^2$ condition is $\ell \geq 2t + 1$.

In conclusion, the adversary structure $\mathcal{A}$ must be $\mathcal{Q}^2$ in $\mathcal{P}$ and both structures $\mathcal{A}$ and Γ must verify

$$\mathcal{A}^c \subset \Gamma \subset \overline{\mathcal{A}}$$

We now explain how to construct, if all these conditions hold, an unforgeable and robust distributed signature scheme.

3.2 Our Proposal

Let $\mathcal{P} = \{1, \ldots, \ell\}$ be the set of participants, and $\mathcal{A} \subset 2^{\mathcal{P}}$ an adversary structure such that it is $\mathcal{Q}^2$ in $\mathcal{P}$. We also consider an access structure Γ verifying $\mathcal{A}^c \subset \Gamma \subset \overline{\mathcal{A}}$.

In our distributed signature scheme, the secret to be shared will belong to a ring $\mathbb{Z}_m$ (instead of a field $\mathbb{Z}_p$, as usually happens in standard secret sharing schemes), that must also remain secret to the participants. The solution that we propose is to define the secret sharing scheme over the integers. So we must modify the definition of standard linear secret sharing schemes (in particular, the vector space ones), which are defined over a finite field, in order to adapt them to our needs. This variation is quite straightforward.

We will have a monotone increasing structure Γ. We say that a function $\psi : \mathcal{P} \cup \{D\} \longrightarrow \mathbb{Z}^r$ realizes Γ when $A \in \Gamma$ if and only if there exists $\{c_i^A\}_{i \in A}$, with $c_i^A \in \mathbb{Q}$, such that $\psi(D) = \sum_{i \in A} c_i^A \psi(i)$.

Then we construct a sort of vector space secret sharing scheme over the integers as follows:

1. if the dealer D wants to share a secret $k \in \mathbb{Z}_m$, he chooses a random vector $\mathbf{w} \in (\mathbb{Z}_m)^r$ such that $\mathbf{w} \cdot \psi(D) = k \mod m$
2. he sends to the participant i his share $s_i = \mathbf{w} \cdot \psi(i) \in \mathbb{Z}$ (without reducing it modulus m)
3. let A be an authorized subset, $A \in \Gamma$; then, $\psi(D) = \sum_{i \in A} c_i^A \psi(i)$, for some $c_i^A \in \mathbb{Q}$. In order to "recover" the secret, the players of A compute

$$\sum_{i \in A} c_i^A s_i \;=\; \sum_{i \in A} c_i^A \mathbf{w} \cdot \psi(i) \;=\; \mathbf{w} \cdot \sum_{i \in A} c_i^A \psi(i) \;=\; \mathbf{w} \cdot \psi(D) \;=\; k + gm$$

This scheme is not a standard perfect secret sharing scheme: non-authorized subsets have not any information about the secret, as desired; however, authorized subsets do not recover exactly the secret k, but this value plus a multiple gm of the unknown modulus m. This additional value will not affect the functioning of our protocol, if the modulus m is taken properly by the dealer.

It is interesting to note that the constructive method of Simmons, Jackson and Martin [32] can also be applied in this scenario, and so any access structure Γ can be realized by a linear secret sharing scheme over $\mathbb{Z}$. In general, this scheme will not be ideal, that is, each participant will have associated more than one vector. For simplicity, we will consider the case in which this secret sharing scheme is ideal. But the scheme can be easily adapted to the general linear case.

By definition of maximal subset, we have that $B \cup \{i\} \in \Gamma$, for each maximal non-authorized subset $B \in (\overline{\Gamma})_0$ and all $i \notin B$; that is, $\psi(D) \in \langle \psi(i), \{\psi(j)\}_{j \in B} \rangle$, in $\mathbb{Q}$. We can deduce from this condition that there exist rational numbers $f_0^{B,i}, \{f_j^{B,i}\}_{j \in B}$ such that

$$\psi(i) = f_0^{B,i} \psi(D) \;+\; \sum_{j \in B'} f_j^{B,i} \psi(j)$$

In our distributed signature scheme, we are going to need an integer value $\Delta = \Delta(\psi)$ such that the following two conditions hold (using the notation introduced above):

(i) $\Delta c_i^A \in \mathbb{Z}$, for all $A \in \Gamma_0$ and $i \in A$, and all possible values of c_i^A.

(ii) $\Delta f_0^{B,i} \in \mathbb{Z}$ and $\Delta f_j^{B,i} \in \mathbb{Z}$, for all $B \in (\overline{\Gamma})_0$, $j \in B$ and $i \notin B$, and all possible values of $f_0^{B,i}$ and $f_j^{B,i}$.

We explain here a method to obtain such a value Δ that can be applied to any access structure. We will see in Section 3.4 some examples of access structures in which an appropriate Δ can be found in a more efficient way.

For each minimal authorized subset $A \in \Gamma_0$, there will be a factor M_A such that $\tilde{c}_i^A = M_A c_i^A$ is an integer, for all $i \in A$. This factor M_A is the determinant of some non-zero minor, with maximal order, of the matrix G_A whose columns are the vectors $\{\psi(i)\}_{i \in A}$. We define $Minors_A = \{\text{non-zero minors of matrix } G_A, \text{ with maximal order}\}$. Then we define the value M_A in the following way:

$$M_A = \operatorname*{lcm}_{g_A \in Minors_A} \{|\det\ g_A|\}$$

It is clear that this factor M_A cancels all the denominators in all the possible solutions $\{c_i^A\}_{i \in A}$. But we are looking for a factor that cancels all the denominators, for all the minimal authorized subsets $A \in \Gamma_0$. We define

$$\Delta_1 = \operatorname*{lcm}_{A \in \Gamma_0} \{M_A\}$$

With respect to the second condition that must satisfy the value Δ, we must consider systems of equations with the following characteristic: the columns of the matrix of this system are the vectors $\psi(D)$ and $\{\psi(j)\}_{j \in B}$, where $B \in (\overline{\Gamma})_0$ is a maximal non-authorized subset. We note this matrix $G_{D,B}$. As before, we define $Minors_{D,B} = \{\text{non-zero minors of matrix } G_{D,B}, \text{ with maximal order}\}$, the value

$$M_{D,B} = \operatorname*{lcm}_{g_{D,B} \in Minors_{D,B}} \{|\det\ g_{D,B}|\}$$

and the value

$$\Delta_2 = \operatorname*{lcm}_{B \in (\overline{\Gamma})_0} \{M_{D,B}\}$$

Finally, the factor Δ is

$$\Delta = \operatorname{lcm}\{\Delta_1, \Delta_2\}$$

Key generation.

Suppose we have an appropriate function $\psi : \mathcal{P} \cup \{D\} \longrightarrow \mathbb{Z}^r$ defining a linear secret sharing scheme over the integers that realizes the access structure Γ. The access structure Γ, the function ψ and the value Δ (that depends only on ψ) are public. The dealer chooses $p = 2p' + 1$ and $q = 2q' + 1$ in such a way that p, q, p' and q' are primes large enough; we define $m = p'q'$. The value $n = pq$ is public ($\phi(n) = 4m$ remains secret to the participants). The dealer chooses the public exponent e as a positive integer such that $\gcd(e, m\Delta) = 1$. He also computes $d = e^{-1} \bmod m$.

Again, the problem is the same as in the threshold case: to generate the signature, the participants in A must do some operations in the exponent, that is, in $\mathbb{Z}_m$, but they do not know the value of m. Therefore, they will not be able to calculate inverses in $\mathbb{Z}_m$, so they will have to multiply some rational values by a certain constant that cancels the denominators. This fact is the reason for which we need the value Δ.

Once this initialization (or key generation) phase is performed, the protocol of the distributed signature scheme is the following.

Generation of the shares.

1. The dealer chooses a random vector $\mathbf{w} \in (\mathbb{Z}_m)^r$, such that $\mathbf{w} \cdot \psi(D) = d \bmod m$.

2. He sends each player his secret share $s_i = \mathbf{w} \cdot \psi(i)$, for $1 \leq i \leq \ell$.

3. The dealer also chooses a random $v \in Q_n$, where Q_n is the subgroup of squares in $\mathbb{Z}_n^*$, and computes $v_i = v^{s_i}$, for $1 \leq i \leq \ell$. He makes public the verification keys v and $\{v_i\}_{1 \leq i \leq \ell}$.

Distributed generation of a RSA signature.

If the players want to sign a message $x = h(M) \in \mathbb{Z}_n^*$, where M is a plaintext and h is a hash function, the process is the following:

1. Each player i, for $1 \leq i \leq \ell$, computes and broadcasts his signature share as $x_i = x^{2\Delta s_i} \bmod n$. He also makes public a "proof of correctness" of this share, that is, a non-interactive proof of knowledge of a value α such that the discrete logarithm of x_i^2 to the base $x^{4\Delta}$ is the same as the discrete logarithm of v_i to the base v and is equal to α. This proof must not give any information about the value α, that is, the secret share s_i (see [30] for the details).

2. Each player j, for $1 \leq j \leq \ell$, verifies the proof of correctness of player i, using the verification keys v and $\{v_i\}_{1 \leq i \leq \ell}$, and publicly complains if the proof is not correct.

3. Players who receive complains from an authorized subset $A \in \Gamma$ are rejected.

4. Once the dishonest players have been rejected, and because of the restriction $\mathcal{A}^c \subset \Gamma$, there is at least one authorized subset $A \in \Gamma$ in which all the signature shares are valid. We can consider, for simplicity, that A is a minimal authorized subset, that is, $A \in \Gamma_0$.

5. Each player in A can obtain rational numbers $\{c_i^A\}_{i \in A}$ such that $\psi(D) = \sum_{i \in A} c_i^A \psi(i)$; multiplying this values by factor Δ, he obtains integer values $\tilde{c}_i^A = \Delta c_i^A$, and then he computes the value $\omega = \prod_{i \in A} x_i^{2\tilde{c}_i^A} = \ldots = x^{4\Delta^2 d} \bmod n$.

6. Since $\gcd(e, \Delta) = 1$, each player in A can obtain integers a and b such that $4\Delta^2 a + eb = 1$.

7. The signature for the message x is $y = \omega^a x^b$. In effect, $y^e = \omega^{ea} x^{be} = x^{4\Delta^2 a + eb} = x \bmod n$.

Some remarks.

- The resulting signature is a standard RSA signature, so the verifier can not distinguish if the signature has been generated by a single person or by a group of participants.

- The secret shares of the participants, $\{s_i\}_{i \in \mathcal{P}}$, are not reduced $\bmod m$, unlike the threshold scheme by Shoup [30]. Suppose that they have $s_i = \mathbf{w} \cdot \psi(i) \bmod m$, and suppose there exists a dishonest subset $B \in \mathcal{A}$ such that $\{\psi(j)\}_{j \in B}$ are not linearly independent over $\mathbb{Q}$ (note that this fact never occurs in the threshold case). Then, players in B could obtain a linear rational combination of their vectors equal to zero; thus, the same combination applied to the shares $\{s_j\}_{j \in B}$ will be equal to zero, in $\mathbb{Z}_m$, that is, players in B could obtain a multiple of m, and so break the system.

- Although these shares s_i are not reduced modulus m, they can be bounded by rmL, where $L = \max_{i \in \mathcal{P}, 1 \leq j \leq r} \{|\psi(i)_j|\}$ is the maximum value of the components of public vectors $\psi(i)$. This detail is necessary in order to prove the simulability or our scheme.

3.3 Security Analysis

The security proof of our scheme is similar to the security proof of the threshold scheme by Shoup [30]. The idea is to prove that the distributed scheme is *simulatable*, and then use this fact to reduce the security of the distributed scheme to the security of standard RSA signature scheme [28].

Definition 3.1. A distributed signature scheme is said to be simulatable if, for any efficient (polynomial time) adversary, there exists an efficient algorithm $Sim = (Sim_1, Sim_2)$ such that:

1. Sim_1 takes as input the public key of the scheme and simulates the view of the adversary on the execution of this key generation phase.

2. Sim_2 takes as input the public key of the scheme, the transcript of Sim_1, a message and its signature, and simulates the view of the adversary on the execution of the distributed generation of this signature.

This definition has been taken from [18], although it is quite standard. The algorithms Sim_1 and Sim_2 know and can use the public information related to the access structure; for example, the threshold t in a threshold scheme, or the function ψ and the value Δ in a more general case. We can see this information as a part of the public key.

Now we prove that our scheme is simulatable, and how this fact leads to the unforgeability of it.

Proposition 3.1. *The distributed signature scheme proposed in Section 3.2 is simulatable.*

Proof. We show how to construct an algorithm that simulates the view of an adversary during a real execution of the protocol.

1. We first construct Sim_1. The input of this algorithm is a public key (n, e), along with the publicly known access structure Γ, the function $\psi : \mathcal{P} \cup \{D\} \longrightarrow \mathbb{Z}^r$ defining the linear secret sharing scheme over the integers that realizes Γ, and the value Δ.

   Let $B \in \mathcal{A}$ the set of players corrupted by the adversary. Because of the necessary combinatorial condition $\mathcal{A} \subset \overline{\Gamma}$, there exists a maximal non-authorized subset $B' \in (\overline{\Gamma})_0$ such that $B \subset B'$. The real share of a player j in B is $s_j = \mathbf{w} \cdot \psi(j)$, where $\mathbf{w}$ is a random vector in $(\mathbb{Z}_m)^r$. To simulate these shares, Sim_1 chooses a random vector $\tilde{\mathbf{w}}$ in $(\mathbb{Z}_{\tilde{m}})^r$, where $\tilde{m} = \lfloor n/4 \rfloor - 1$. Then, it computes $\tilde{s_j} = \tilde{\mathbf{w}} \cdot \psi(j)$, for all $j \in B'$.

   The values $\{\tilde{s}_j\}_{j \in B}$ are the output of Sim_1; since the statistical distance between the uniform distribution on $\{0, \ldots, m\}$ and the uniform distribution on $\{0, \ldots, \tilde{m}\}$ is $\mathcal{O}(n^{-1/2})$, we can deduce that the distribution of the real values $\{s_j\}_{j \in B}$ and the distribution of the simulated values $\{\tilde{s}_j\}_{j \in B}$ are computationally indistinguishable, as desired. It is in this last point where we need the shares (the real as well as the simulated ones) to be bounded.

2. Now we describe Sim_2. The input of this algorithm will be the public key (n, e) of the scheme, the transcript of Sim_1 (in particular, $\{\tilde{s}_j\}_{j\in B'}$), a message $x = h(M)$ and its signature y. Sim_2 computes $\tilde{x}_j = x^{2\Delta\tilde{s}_j}$, for all $j \in B'$. Given that $B' \in (\overline{\Gamma})_0$, we know that $B' \cup \{i\} \in \Gamma$, for all $i \in \mathcal{P}\backslash B'$; this condition is equivalent, in $\mathbb{Q}$, to $\psi(i) \in \langle \psi(D), \{\psi(j)\}_{j\in B'}\rangle$. That is, there exist rational numbers $f_0^{B',i}, \{f_j^{B',i}\}_{j\in B'}$ such that

$$\psi(i) = f_0^{B',i}\psi(D) \ + \sum_{j\in B'} f_j^{B',i}\psi(j)$$

Due to the second condition of Lemma 3.1, we have that the values $\Delta f_0^{B',i}$ and $\{\Delta f_j^{B',i}\}_{j\in B'}$ are all integers. Therefore, the algorithm Sim_2 can compute the simulated signature shares

$$\tilde{x}_i = y^{2(\Delta f_0^{B',i} \ + \ e \sum\limits_{j\in B'} \Delta f_j^{B',i}\tilde{s}_j)} \mod n$$

for all $i \in \mathcal{P}\backslash B'$.

Using the same technique, Sim_2 can also generate the values $\tilde{v}, \tilde{v}_1, \ldots, \tilde{v}_\ell$, computationally indistinguishable from the verification keys that the adversary would see on a real execution of the protocol.

Zero-knowledge simulability of the "proofs of correctness" can be achieved exactly in the same way as in Shoup's scheme (see [30] for the details).

□

Theorem 3.1. *In the random oracle model [3], the distributed signature scheme proposed in the Section 3.2 is secure (robust and unforgeable under a chosen message attack), assuming that the standard RSA signature scheme is secure.*

Proof. The scheme is obviously *robust*, due to the requirements on the structures Γ and $\mathcal{A}$.

The proof of *soundness* for the "proofs of correctness", in the random oracle model, is the same as in the threshold case (see [30]), because this part of the protocol is identical.

Suppose now the distributed scheme to be *forgeable* under chosen message attacks. That is, there exists an efficient adversary $\mathcal{A}_{Dist}$ that chooses some messages, executes the distributed signature protocol for them (obtaining their signatures, all the broadcast information and all the secret information of players corrupted by the adversary), and then is able to compute a valid signature for a message that has not been signed before. In this case, we could use this algorithm to construct a successful chosen message attack $\mathcal{A}_{RSA}$ against standard RSA signature scheme:

1. $\mathcal{A}_{RSA}$ runs Sim_1 with input the same public key used by $\mathcal{A}_{Dist}$, to obtain the view of $\mathcal{A}_{Dist}$ on an execution of the key generation phase.

2. For each message chosen by $\mathcal{A}_{Dist}$ to be signed during its attack, $\mathcal{A}_{RSA}$ executes the standard RSA signature scheme and obtains its signature ($\mathcal{A}_{RSA}$ is allowed to do this, by definition of chosen message attack [20]); then $\mathcal{A}_{RSA}$ executes Sim_2 with input the public key, the chosen message and its signature, and the transcript of Sim_1, and thus obtains the view of $\mathcal{A}_{Dist}$ on the executions of the distributed signature protocol.

3. $\mathcal{A}_{RSA}$ can use this view to run $\mathcal{A}_{Dist}$ and get a valid signature for a message non signed before.

But such an attack against standard RSA signatures is assumed to be computationally infeasible, in the random oracle model, so we conclude that our first assumption was wrong; that is, the distributed signature scheme is unforgeable under chosen message attacks. □

3.4 Efficiency and Examples

Our scheme runs with any access structure, so it is logical that its efficiency can be good or worse depending on the case. In some cases, the computation of the value Δ will require a high computational work. But this computation must be done only once, and can be off-line, before the running of the protocol. In other cases, there will exist an efficient method to obtain a value Δ satisfying the requirements of our scheme; we will see some examples of this case.

Depending on the access structure Γ and the function ψ defining the corresponding linear secret sharing scheme, the size of Δ can be very large. This fact also happens in the threshold case, where the presence of the value $\Delta = \ell!$ can make the system quite inefficient, if the total number of players ℓ is very large.

Now we present some examples of structures which are different from the threshold ones, and in which our distributed signature scheme can be implemented in an efficient way; that is, a factor Δ canceling all the inverses in the coefficients of the linear combinations can be computed without calculating all the corresponding minors.

We are going to consider *bipartite structures* [25], in which there are two classes of participants and all participants in the same class play an equivalent role in the structure. This structures have interesting practical applications, and they can not be realized by a threshold secret sharing scheme, so the threshold proposal of Shoup could not be used in these cases.

Example 1 Let us consider a group of players divided in two sets $X = \{P_1, \ldots, P_{r_1}\}$ and $Y = \{Q_1, \ldots, Q_{r_2}\}$, with $r_1 \geq 2$ or $r_2 \geq 3$, such that all players in the same set play an equivalent role in the access structure. Suppose that any two players in the set X have enough power to sign a message, while any subset of the set Y can not sign a message without the participation of at least one player of the set X; furthermore, if only one player of the set X participates, at least two players of the set Y are also needed in order to generate a signature. This structure can tolerate dishonest behaviors of some subsets of players, for example any subset of set Y, or any subset formed by a player of X and a player of Y. The access structure is defined by $\Gamma_0 = \{\{P_{i_1}, P_{i_2}\} : 1 \leq i_1 < i_2 \leq r_1\} \cup \{\{P_i, Q_{k_1}, Q_{k_2}\} : 1 \leq i \leq r_1 \text{ and } 1 \leq k_1 < k_2 \leq r_2\}$; and we can tolerate corruption of any subset that is not authorized, that is, $\mathcal{A}_0 = (\overline{\Gamma})_0 = \{\{P_i, Q_k\} : 1 \leq i \leq r_1 \text{ and } 1 \leq k \leq r_2\} \cup \{Y\}$. Note that these structures are not threshold ones, and that they verify the necessary combinatorial conditions of Section 3.1.

Now we must find a secret sharing scheme over the integers realizing the access structure Γ defined above. An appropriate assignment is $\psi(D) = (1, 0, 0)$, $\psi(P_i) = (1, i, 0)$, for $1 \leq i \leq r_1$, and $\psi(Q_k) = (0, k, 1)$, for $1 \leq k \leq r_2$.

It is easy to see that this function $\psi : \{D\} \cup X \cup Y \longrightarrow \mathbb{Z}^3$ defines the desired access structure Γ. The factor Δ could be calculated following the method explained in Section 3.2. In this case, the minimal authorized subsets are of one of these forms: (i) $A = \{P_{i_1}, P_{i_2}\}$, with $1 \leq i_1 < i_2 \leq r_1$, or (ii) $A = \{P_i, Q_{k_1}, Q_{k_2}\}$, with $1 \leq i \leq r_1$

and $1 \leq k_1 < k_2 \leq r_2$. The minors obtained from the corresponding matrix have the form $i_2 - i_1$ in the case (i) and $k_2 - k_1$ in the case (ii). With respect to the maximal non-authorized subsets, they have one of the following forms: (iii) $B = \{P_i, Q_k\}$ with $1 \leq i \leq r_1$ and $1 \leq k \leq r_2$, or (iv) $B = \{Q_1, \ldots, Q_{r_2}\} = Y$. The minors of the corresponding matrices $M_{D,B}$ have the form i, with $1 \leq i \leq r_1$ for the case (iii) and $k_2 - k_1$, with $1 \leq k_1 < k_2 \leq r_2$ for the case (iv). In conclusion, it is easy to see that a multiple of all these minors is the value $\Delta = R!$, where $R = \max\{r_1, r_2\}$, which can be computed without calculating explicitly any minor, and which is smaller than the factor $\Delta = (r_1 + r_2)!$ corresponding to an hypothetic threshold structure with the same number of participants. Moreover, the dealer can choose the public exponent e as a random prime greater than R. The rest of the signature protocol can be performed as described in Section 3.2.

Example 2 Suppose now that we have again two sets $X = \{P_1, \ldots, P_{r_1}\}$ and $Y = \{Q_{r_1+1}, \ldots, Q_{r_1+r_2}\}$ of r_1 and r_2 players, respectively, such that $r_1 \geq 3$ and $r_2 \geq 4$ (other alternative restrictions can be imposed, in fact). We denote the total number of players as $N = r_1 + r_2$. Now a valid signature can only be generated by any subset of at least three players of the set X, or by any set including two players of set X and two players of set Y, or by any set including one player of set X and three players of set Y; but any subset formed only by players of set Y can not sign a message. This is also a bipartite access structure that is not threshold.

The access structure is defined by $\Gamma_0 = \{\{P_{i_1}, P_{i_2}, P_{i_3}\} : 1 \leq i_1 < i_2 < i_3 \leq r_1\} \cup \{\{P_{i_1}, P_{i_2}, Q_{k_1}, Q_{k_2}\} : 1 \leq i_1 < i_2 \leq r_1 \text{ and } r_1 + 1 \leq k_1 < k_2 \leq N\} \cup \{\{P_i, Q_{k_1}, Q_{k_2}, Q_{k_3}\} : 1 \leq i \leq r_1 \text{ and } r_1 + 1 \leq k_1 < k_2 < k_3 \leq N\}$; and we can tolerate corruption of any subset that is not authorized, that is, $\mathcal{A}_0 = (\overline{\Gamma})_0 = \{\{P_{i_1}, P_{i_2}, Q_k\} : 1 \leq i_1 < i_2 \leq r_1 \text{ and } r_1 + 1 \leq k \leq N\} \cup \{\{P_i, Q_{k_1}, Q_{k_2}\} : 1 \leq i \leq r_1 \text{ and } r_1 + 1 \leq k_1 < k_2 \leq N\} \cup \{Y\}$.

These structures again verify the conditions of Section 3.1, in order to provide our distributed signature scheme with the properties of unforgeability and robustness.

Now we must construct a secret sharing scheme over the integers realizing this access structure. We assign to the dealer the vector $\psi(D) = (1, 0, 0, 0)$. We assign to each participant $P_i \in X$, for $1 \leq i \leq r_1$, the vector $\psi(P_i) = (1, i, i^2, 0)$. We assign to each participant $Q_k \in Y$, for $r_1 + 1 \leq k \leq N$, the vector $\psi(Q_k) = (0, k, k^2, 1)$.

This function ψ defines the desired access structure. And it is not difficult to see that the factor $\Delta = (2N)!$ cancels all the denominators that appear in the coefficients obtained considering minimal authorized subsets as well as maximal non-authorized subsets of players. In this case, the public exponent e can be a prime greater than $2N$, where N is the total number of participants.

This value of Δ can be very large (as in the threshold case), but the necessary work in order to calculate it is not expensive. This is a sore point: maybe it can be more interesting to spend more time in the initialization phase (only performed once), calculating all the corresponding minors, and obtain therefore a smaller value for the factor Δ, which will be used in every execution of the distributed signature protocol.

4 Conclusion and open problems

We present a distributed RSA digital signature scheme, generalizing the threshold scheme of Shoup in [30]. The scheme proposed by Shoup considers only threshold structures, meaning that subsets which can generate a signature as well as tolerated subsets of corrupted participants are defined according to their cardinality.

We consider more general access and adversary structures, and state the combinatorial conditions that they must satisfy in order to design a robust and unforgeable distributed signature scheme. Depending on the structures, the efficiency of the scheme will be better or will not. An interesting open problem is how to design other distributed signature schemes running for general structures and being more efficient than this proposal, if it is possible.

As the scheme of Shoup, our scheme need the presence of a trusted dealer. In the case of schemes considering only threshold structures, some proposals of schemes which do not need a dealer have been made ([23], [8] and [12]). They are based on distributed generation of RSA keys ([4] and [16]). Both the proposal in [8] and the one in [12] adapt the scheme of Shoup in order to avoid the necessity of safe primes; thus they can use distributed generation of RSA keys to avoid the presence of a trusted dealer. Our scheme can be adapted in the same way, that is, avoiding the necessity of safe primes. But the presence of the dealer is still necessary, because the proposals of distributed generation of RSA keys run only with threshold structures, so we can't use them.

A recent work [1] shows how can a group of players jointly generate shared RSA keys, where the modulus is the product of two safe primes; this work can be added to the scheme of Shoup in order to obtain a fully distributed RSA signature scheme with no dealer. Again, this proposal is valid considering only threshold structures.

How to find the way of jointly generating and distributing RSA keys among a group of players, without a trusted dealer, in such a way that only those subsets in the (general) access structure can generate a valid signature remains also as an open problem.

Another problem is to find relations between the complexity of a standard linear secret sharing schemes (defined over a field) and linear secret sharing schemes that we consider in this paper, defined over $\mathbb{Z}$. For example, can we transform a linear secret sharing scheme defined over $\mathbb{Z}_p$ into a linear secret sharing scheme defined over $\mathbb{Z}$ realizing exactly the same access structure than the original scheme?

References

[1] J. Algesheimer, J. Camenisch and V. Shoup. Efficient computation modulo a shared secret with application to the generation of shared safe-prime products. (2002) Available in http://eprint.iacr.org/2002/029/

[2] C. Beaver, M. Collins, P. Gemmell, A. Johnston, W.D. Neumann and R. Schroeppel. Cryptanalysis of the Frankel-MacKenzie-Yung shared RSA key generation protocol. *Internal document, Sandia National Laboratories* (2000).

[3] M. Bellare and P. Rogaway. Random oracles are practical: a paradigm for designing efficient protocols. *First ACM Conference on Computer and Communications Security* p. 62-73 (1993).

[4] D. Boneh and M. Franklin. Efficient Generation of Shared RSA Keys. *Advances in Cryptology-Crypto'97* **1294** p. 425-439 (1997). Extensive version in http://crypto.stanford.edu/∼dabo/pubs.html.

[5] C. Boyd. Digital Multisignatures. *Cryptography and Coding* p. 241-246 (1986).

[6] E.F. Brickell. Some ideal secret sharing schemes. *J.Combin. Math. and Combin. Comput.* **9** p. 105-113 (1989).

[7] D. Catalano, R. Gennaro and S. Halevi. Computing inverses over a shared secret modulus. *Advances in Cryptology-Eurocrypt'00* (2000).

[8] I. Damgård and M. Koprowski. Practical threshold RSA signatures without a trusted dealer. *Advances in Cryptology-Eurocrypt'01* (2001).

[9] A. De Santis, Y. Desmedt, Y. Frankel, and M. Yung. How to share a function securely. *Proceedings of the 26th ACM Symposium on Theory of Computing, Santa Fe* p. 522- 533 (1994).

[10] Y. Desmedt. Society and group oriented cryptography: A new concept. *Advances in Cryptology-Crypto'87* **293** p. 120-127 (1988).

[11] Y. Desmedt and Y. Frankel. Threshold cryptosystems. *Advances in Cryptology-Crypto'89* **435** p. 307-315 (1989).

[12] P.A. Fouque and J. Stern. Fully distributed threshold RSA under standard assumptions. *Proceedings of Asiacrypt 2001* p. 310-330 (2001).

[13] Y. Frankel and Y. Desmedt. Parallel reliable threshold multisignature. *Technical Report TR-92-04-02, Univ. of Wisconsin-Milwaukee.* (1992).

[14] Y. Frankel, P. Gemmell, P. MacKenzie and M. Yung. Proactive RSA. *Advances in Cryptology-Crypto'97* (1997).

[15] Y. Frankel, P. Gemmell, P. MacKenzie and M. Yung. Optimal Resilience Proactive Public-Key Cryptosystems. *FOCS'97* (1997).

[16] Y. Frankel, P. MacKenzie and M. Yung. Robust efficient distributed RSA-key generation. *Proceedings of STOC'98* (1998).

[17] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory* **31** p. 469-472 (1985).

[18] R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin. Robust and efficient sharing of RSA functions. *Advances in Cryptology-Crypto'96* p. 157-172 (1996). Full version available in http://researchweb.watson.ibm.com/security/projects.html

[19] R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin. Robust Threshold DSS Signatures. *Advances in Cryptology-Eurocrypt'96* p. 354-371 (1996).

[20] S. Goldwasser, S. Micali and R. Rivest. A Digital Signature Scheme Secure Against Adaptative Chosen-Message Attacks. *SIAM Journal of Computing* **17 (2)** p. 281-308 (1988).

[21] L. Harn. Group oriented (t, n) threshold digital signature scheme and digital multisignature. *IEE Proceedings on Computation and Digital Technologies* **5** p. 307-313 (1994).

[22] S. Lanford. Threshold DSS signatures without a trusted party. *Advances in Cryptology-Crypto'95* **963** p. 397-409 (1995).

[23] S. Miyazaki, K. Sakurai and M. Yung. On threshold RSA-signing with no dealer. *Proceedings of ICISC* **1787** (1999).

[24] T. Okamoto. A digital multisignature scheme using bijective public-key cryptosystems. *ACM Trans. on Computer Systems* **6,8** p. 432-441 (1988).

[25] C. Padró and G. Sáez. Secret sharing schemes with bipartite access structure. *IEEE Transactions on Information Theory*, Vol. 46, No. 7, p. 2596-2604 (2000). A previous version appeared in *Advances in Cryptology–Eurocrypt'98* p. 500-511 (1998).

[26] T. Pedersen. A Threshold Cryptosystem without a Trusted Party. *Advances in Cryptology-Eurocrypt'91* **547** p. 522-526 (1991).

[27] T. Rabin. A simplified approach to threshold and proactive RSA. *Advances in Cryptology-Crypto'98* **1462** p. 89-104 (1998).

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

[29] A. Shamir. How to share a secret. *Communication of de ACM* **22** p. 612-613 (1979).

[30] V. Shoup. Practical Threshold Signatures. *Proceedings of Eurocrypt'00* **1807** p. 207-220 (2000).

[31] G.J. Simmons. An introduction to shared Secret and/or shared control schemes and their application. *Contemporary Cryptology. The Science of Information Integrity.* IEEE Press p. 441-497 (1991).

[32] G. J. Simmons, W. Jackson and K. Martin. The geometry of secret sharing schemes. *Bulletin of the ICA* **1** p.71-88 (1991).

[33] D.R. Stinson. An explication of secret sharing schemes. *Des. Codes Cryptogr* **2** p. 357-390 (1992).

[34] D.R. Stinson. *Cryptography: Theory and Practice.* CRC Press Inc., Boca Raton (1995).

[35] D.R. Stinson and R. Strobl. Provably Secure Distributed Schnorr Signatures and a (t, n) Threshold Scheme for Implicit Certificates. *Sixth Australasian Conference on Information Security and Privacy (ACISP 2001)* (2001).