University “Alexandru Ioan Cuza” of Iaşi
Faculty of Computer Science

TECHNICAL REPORT

Secret Sharing Schemes with Applications in Security Protocols

(Ph.D. Thesis)

Sorin Iftene

TR 07-01, January 2007

ISSN 1224-9327

Universitatea “Alexandru Ioan Cuza” Iaşi
Facultatea de Informatică
Str. Berthelot 16, 6600-Iaşi, Romania
Tel. +40-32-201090, email: bibl@infoiasi.ro

Secret Sharing Schemes with Applications in Security Protocols

Sorin Iftene

*Thesis submitted to the "Al. I. Cuza" University of Iaşi
for the degree of Doctor of Philosophy
in Computer Science*

Department of Computer Science
"Al. I. Cuza" University of Iaşi
Iaşi, Romania
October 2006

Sorin Iftene
Department of Computer Science
"Al. I. Cuza" University of Iaşi
Iaşi, Romania
E-mail: siftene@infoiasi.ro

Prof. Dr. Gheorghe Grigoraş, Chair ("Al. I. Cuza" University of Iaşi)
Prof. Dr. Ferucio Laurenţiu Ţiplea, Supervisor ("Al. I. Cuza" University of Iaşi)
Prof. Dr. Adrian Atanasiu (University of Bucharest)
Prof. Dr. Victor Patriciu (Military Technical Academy)
Prof. Dr. Constantin Popescu (University of Oradea)

Preface

A secret sharing scheme starts with a *secret* and then derives from it certain *shares* (or *shadows*) which are distributed to users. The secret may be recovered only by certain predetermined groups which belong to the *access structure*. Secret sharing schemes have been independently introduced by Blakley [12] and Shamir [134] as a solution for safeguarding cryptographic keys. Secret sharing schemes can be used for any situation in which the access to an important resource has to be restricted. We mention here the case of opening bank vaults or launching a nuclear missile.

In the first secret sharing schemes only the number of the participants in the reconstruction phase was important for recovering the secret. Such schemes have been referred to as *threshold* secret sharing schemes. There are secret sharing schemes that deal with more complex access structures than the threshold ones. We mention here the *weighted threshold* secret sharing schemes in which a positive weight is associated to each user and the secret can be reconstructed if and only if the sum of the weights of the participants is greater than or equal to a fixed threshold, the *hierarchical* (or *multilevel*) secret sharing schemes in which the set of users is partitioned into some levels and the secret can be recovered if and only if there is an initialization level such that the number of the participants from this level or higher levels is greater than or equal to the initialization level threshold, the *compartmented* secret sharing schemes in which the set of users is partitioned into compartments and the secret can be recovered if and only if the number of participants from any compartment is greater than or equal to a compartment threshold and the total number of participants is greater than or equal to a global threshold. Ito, Saito, and Nishizeki [90], Benaloh and Leichter [9] have proposed constructions for realizing any *monotone* (i.e., if a group belongs to the access structure, so does a larger group) access structure.

The schemes in which the unauthorized groups gain no information about the secret are referred to as *perfect*. Karnin, Greene, and Hellman [97] have proved, using the concept of entropy, that in any perfect threshold secret sharing scheme the shares must be at least as long as the secret and, later on, Capocelli, De Santis, Gargano, and Vaccaro [27] have extended this result to the case of any perfect secret sharing scheme. In an *ideal* [20] (perfect) secret sharing scheme the shares are as long as the secret. *Ramp* schemes have appeared as a solution for the situations in which smaller shares are required.

The applications of the secret sharing schemes can be categorized as *secure multiparty computation* protocols, i.e., protocols which allow to some users to compute $f(x_1, \ldots, x_m)$ such that the input x_i is known only by the i^{th} user. Threshold cryptographic protocols and some e-voting or e-auction protocols are special cases of secure multiparty computation protocols.

Structure of the Thesis

Chapter 1 presents the basic definitions and concepts related to secret sharing schemes. After discussing the concept of access structure, we review the most important mathematical models for secret sharing. Perfect secret sharing schemes are presented in comparison with the computational-secure ones. Finally, some terms and indicators for characterizing the size of the shares in a secret sharing scheme are presented.

Chapter 2 describes the most important constructions for realizing different access structures. Our contribution consists in the application of the general variant of the Chinese remainder theorem in designing several classes (as threshold, weighted threshold, compartmented or more general ones) of secret sharing schemes and general information dispersal schemes. The proposed secret sharing schemes based on the Chinese remainder theorem provide the flexibility for performing a required compromise between the size of the shares and the level of security. We also propose a secret sharing framework based on determinants. The main idea is that a matrix over an arbitrary commutative ring is “puzzled” and the set of the resulted pieces is partitioned in shares such that every maximal unauthorized access set cannot solve the puzzle and such that every minimal authorized access set can do so. The secret will be the module of the determinant of the formed matrix. Two different methods are presented and some interesting changeability properties are discussed.

Additional interesting capabilities, as multiplicative/homomorphic properties, dealer-free setup, verifiability, changeability, or sharing many secrets are discussed in Chapter 3.

Chapter 4 presents the most important applications of secret sharing in security protocols. Our contribution consists in constructing threshold cryptographic schemes and a multi-authority e-voting scheme in which the parties involved may have non-equal weights, all relying on the secret sharing schemes based on the general variant of the Chinese remainder theorem.

Our conclusions and some interesting future work directions are presented in Chapter 5.

The required background on number theory, information theory, and cryptography is provided in the appendices.

I would not have finished this thesis without the assistance of various people. First of all I would like to thank my supervisor Professor Ferucio Laurenţiu Ţiplea for giving me the opportunity of an academic career and introducing me to the wonderful world of cryptography. I am grateful for his advice, trust and continuous encouragement. Also, I would like to thank Ioana Boureanu, for carefully reading my thesis and providing valuable comments for its improvement. My thanks also go to Professor Keith Martin and Professor Josef Pieprzyk, for kindly answering to my questions and providing me some important papers.

Lastly, but perhaps most importantly, I would like to thank my family for its continuous support.

To my family and my teachers

Basic Notation

Sets

$X \subseteq Y$	X is a subset of Y
$X \subset Y$	X is a subset of Y and $X \neq Y$
$\mathcal{P}(X)$	the powerset of X, i.e., the set of all subsets of X
$\cup$	the set union
$\cap$	the set intersection
$\setminus$	the set difference
$\overline{X}$	the complement of set X with respect to a given superset
$\times$	the cartesian product
X^n	$\underbrace{X \times X \times \cdots \times X}_{n \text{ times}}$
$\vert X \vert$	the cardinality of X
$\emptyset$	the empty set
$\mathbf{Z}$	the set of integers
$\mathbf{N}$	the set of positive integers (natural numbers)
$\mathbf{GF}_q$	the Galois field of order q, where q is a prime power

Number Theory

$\vert x \vert$	the length of a positive integer x - usually, the number of bits of x
$n!$	n factorial, i.e., $1 \cdot 2 \cdots n$
$a \ \mathtt{div}\ b$	the quotient of the integer division of a by b
$a \ \mathtt{mod}\ b$	the remainder of the integer division of a by b
$a \mid b$	a divides b
$(a_1, \ldots, a_n)$	the greatest common divisor of the integers $a_1, \ldots, a_n$
$[a_1, \ldots, a_n]$	the least common multiple of the integers $a_1, \ldots, a_n$
$a \ \equiv b \ mod \ m$	a is congruent modulo m with b
$a \ \mathsf{mod}\ m$	the unique element r, $-\lceil \frac{m}{2} \rceil \leq r \leq \lceil \frac{m}{2} \rceil$, such that $a \equiv r \ mod \ m$
$\mathbf{Z}_m$	the set $\{0, 1, \ldots, m-1\}$, for some $m \geq 1$
$\oplus_m$	the operation $\oplus$ modulo m, for some $m \geq 1$ ($a \oplus_m b = a \oplus b \ \mathtt{mod}\ m$)
$\mathbf{Z}_m^*$	the set $\{a \in \mathbf{Z}_m \vert (a, m) = 1\}$
a^{-1} $(mod\ m)$	the multiplicative inverse of a modulo m, for some $a \in \mathbf{Z}_m^*$
$\phi(m)$	the value of Euler's totient function in m

$\lambda(m)$	the value of Carmichael's reduced totient function in m
$ord_m(a)$	the order of a modulo m
$log_\alpha \beta$	the discrete logarithm to the base α of β

Information Theory

$(p_x \vert x \in \mathcal{X})$	a probability distribution over a finite set $\mathcal{X}$
$H(\mathcal{X})$	the entropy of the set $\mathcal{X}$ with respect to some probability distribution
$H(\mathtt{X})$	the entropy of the random variable $\mathtt{X}$
$H(\mathtt{X} \vert \mathtt{Y})$	the conditional entropy of $\mathtt{X}$ given $\mathtt{Y}$

Secret Sharing

n	the total number of participants
$1, 2, , \ldots, n$	the (labels of) participants
$\mathcal{A}$	the authorized access structure ($\mathcal{A} \subseteq \mathcal{P}(\{1, 2, \ldots, n\})$)
$\mathcal{A}_{min}$	the set of the minimal authorized groups
$\overline{\mathcal{A}}$	the unauthorized access structure (the complement of $\mathcal{A}$)
$\overline{\mathcal{A}}_{max}$	the set of the maximal unauthorized groups
k	the threshold (for the case of threshold secret sharing)
$\mathcal{S}$(or $\mathcal{S}_0$)	the set of secrets
S	the secret
$\mathcal{S}_{shares}$	the set of shares
$\mathcal{S}_i$	the set of shares assigned to the user i
I_i	the share corresponding to the user i
$S \leftrightarrow_{\mathcal{A}}^{(split,combine)} I_1, \ldots, I_n$	$I_1, \ldots, I_n$ are the shares of the secret S with respect to the $\mathcal{A}$-secret sharing scheme $(split, combine)$
$S \leftrightarrow I_1, \ldots, I_n$	$I_1, \ldots, I_n$ are the shares corresponding to the secret S (when $\mathcal{A}$ and $(split, combine)$ are clear from context)

Contents

Chapter 1

Preliminaries

A secret sharing scheme starts with a *secret* and then derives from it certain *shares* (or *shadows*) which are distributed to users. The secret may be recovered only by certain predetermined groups. More exactly, suppose we have n users labeled with the numbers $1, \ldots, n$ and let us consider a set of groups $\mathcal{A} \subseteq \mathcal{P}(\{1, 2, \ldots, n\})$. Informally, an $\mathcal{A}$*-secret sharing scheme* is a method of generating $(S, (I_1, \ldots, I_n))$ such that

- (*correctness*) - for any $A \in \mathcal{A}$, the problem of finding the element S, given the set $\{I_i \mid i \in A\}$, is "easy";
- (*security*) - for any $A \in \overline{\mathcal{A}}$, the problem of finding the element S, given the set $\{I_i \mid i \in A\}$, is "hard".

S will be referred to as the *secret* and $I_1, \ldots, I_n$ will be referred to as the *shares* (or the *shadows*) of S. The set from where the secrets are chosen will be denoted by $\mathcal{S}$ or by $\mathcal{S}_0$ and the set of the shares assigned to the i^{th} user will be denoted by $\mathcal{S}_i$, for all $1 \leq i \leq n$. Sometimes, it will be useful to consider the set of all possible shares. We will denote this set by $\mathcal{S}_{shares}$ ($\mathcal{S}_{shares} = \cup_{i=1}^{n} \mathcal{S}_i$).

Usually, a secret sharing scheme is coordinated by a *dealer* (*administrator*) who has to be a mutually trusted party, but there are secret sharing schemes which can be configured without the presence of a dealer (see Section 3.2). In some situations, the value of the secret is predetermined. In this case the secret is called *explicit*. The dealer receives this value, derives the corresponding shares, and securely distributes them to the users. In case of an *implicit* secret, there is no restriction on its value excepting that it has to belong to some domain. The dealer first generates the secret in some predetermined domain, then derives the corresponding shares and, finally, securely distributes the shares to the users.

The reconstruction of the secret can be made by the participants after they pool together their shares or by a special party, called *combiner*, after receiving the shares from the users of an authorized group.

1.1 Access Structures

The (*authorized*) *access structure* of a secret sharing scheme is the set of all groups which are designed to reconstruct the secret. The elements of the *access structure* will be referred to as the *authorized groups/sets* and the rest are called *unauthorized groups/sets*.

As Ito, Saito, and Nishizeki have remarked in [90], any access structure $\mathcal{A}$ must satisfy the natural[1] condition

$$(\forall B \in \mathcal{P}(\{1, 2, \ldots, n\}))((\exists A \in \mathcal{A})(A \subseteq B) \Rightarrow B \in \mathcal{A}).$$

This intuitively means that if a group can recover the secret, so can a larger group. Benaloh and Leichter called such access structures *monotone* in [9]. We have a dual property for the unauthorized access structure $\overline{\mathcal{A}}$:

$$(\forall B \in \mathcal{P}(\{1, 2, \ldots, n\}))((\exists A \in \overline{\mathcal{A}})(B \subseteq A) \Rightarrow B \in \overline{\mathcal{A}})$$

that intuitively means that if a group cannot recover the secret, neither can a smaller group.

In this thesis we will consider only monotone access structures.

Any monotone authorized access structure $\mathcal{A}$ is well specified by the set of the minimal authorized groups, i.e., the set

$$\mathcal{A}_{min} = \{A \in \mathcal{A} | (\forall B \in \mathcal{A} \setminus \{A\})(\neg(B \subseteq A))\}.$$

In this case we will also use the notation $\mathcal{A} = cl(\mathcal{A}_{min})$ and $\mathcal{A}_{min}$ will be referred to as the *basis* of $\mathcal{A}$. In general, the *closure* of some $\mathcal{C} \subseteq \mathcal{P}(\{1, 2, \ldots, n\})$, denoted by $cl(\mathcal{C})$, is defined as

$$cl(\mathcal{C}) = \{A \in \mathcal{P}(\{1, 2, \ldots, n\}) | (\exists C \in \mathcal{C})(C \subseteq A)\}.$$

The unauthorized access structure $\overline{\mathcal{A}}$ is well specified by the set of the maximal unauthorized groups, i.e., the set

$$\overline{\mathcal{A}}_{max} = \{A \in \overline{\mathcal{A}} | (\forall B \in \overline{\mathcal{A}} \setminus \{A\})(\neg(A \subseteq B))\}.$$

Example 1.1.1 Let us consider $n = 4$ and the access structure $\mathcal{A} = \{\{1, 2\}, \{1, 2, 3\}, \{1, 2, 4\}, \{1, 2, 3, 4\}, \{3, 4\}, \{1, 3, 4\}, \{2, 3, 4\}\}$. We obtain that $\mathcal{A}_{min} = \{\{1, 2\}, \{3, 4\}\}$, $\overline{\mathcal{A}} = \{\emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{1, 3\}, \{1, 4\}, \{2, 3\}, \{2, 4\}\}$, and $\overline{\mathcal{A}}_{max} = \{\{1, 3\}, \{1, 4\}, \{2, 3\}, \{2, 4\}\}$.

[1] There are papers (see, for example, [10] or [15]) that consider non-monotone access structures. More exactly, in these schemes, there are positive and negative shares which lead to veto capabilities. As Obana and Kurosawa have remarked in [119], this feature is possible only in the case that we assume that the reconstruction machine is trustworthy. The simplest solution for the veto feature is that the opposing participants give a special "veto" share in the secret reconstruction phase, leading to an incorrect secret.

We will also assume that $\emptyset \notin \mathcal{A}$ (or, equivalently, $\mathcal{A}_{min} \neq \{\emptyset\}$) because, otherwise, any group can recover the secret and, thus, there is no secret at all. Such access structures are also referred to as *Sperner systems.*

Another natural condition for an access structure $\mathcal{A}$ is to be *connected* [22], i.e., any user i belongs to some minimal authorized group.

The *rank* (*co-rank*) of the access structure $\mathcal{A}$ is defined as the maximum (minimum) number of the participants in a minimal authorized group. If the rank and the co-rank of $\mathcal{A}$ are both equal with some positive integer r, we will say that $\mathcal{A}$ is r*-homogeneous* (or r*-uniform*). In this case, every minimal authorized group has exactly r members. The *intersection number* of $\mathcal{A}$ is the maximum number of participants who belong to two different minimal authorized groups.

Operations with Access Structures

We present next some operations with access structures. Some constructions of dealer-free secret sharing schemes (see Section 3.2) or constructions of new secret sharing schemes from old ones (see Section 2.9) are based on such operations.

The following definitions are taken from [104, 105].

Definition 1.1.1 Let $\mathcal{A}$ be an access structure over $\{1, 2, \ldots, n\}$. The *dual* of $\mathcal{A}$, denoted by $\mathcal{A}^*$, is the access structure over $\{1, 2, \ldots, n\}$ given by

$$A \in \mathcal{A}^* \Leftrightarrow \overline{A} \notin \mathcal{A}.$$

It is easy to prove that $\mathcal{A}^*$ is also a monotone access structure and $(\mathcal{A}^*)^* = \mathcal{A}$. Several papers, e.g., [51], [92], [52], have considered the relation between realizing an access structure and realizing its dual.

Definition 1.1.2 Let $\mathcal{A}$ be an access structure and $P \subseteq \{1, 2, \ldots, n\}$. The *restriction* of $\mathcal{A}$ on P, denoted by $\mathcal{A}|_P$, is the access structure over $\overline{P}$ given by

$$A \in \mathcal{A}|_P \Leftrightarrow A \in \mathcal{A}.$$

Intuitively, the access structure $\mathcal{A}|_P$ includes all the sets from $\mathcal{A}$ that do not contain any element from the set P.

Example 1.1.2 Let us consider the access structure $\mathcal{A} = \{\{1, 2\}, \{1, 2, 3\}, \{1, 2, 4\}, \{1, 2, 3, 4\}, \{3, 4\}, \{1, 3, 4\}, \{2, 3, 4\}\}$ and $P = \{3\}$. We obtain

$$\mathcal{A}|_P = \{\{1, 2\}, \{1, 2, 4\}\}.$$

Definition 1.1.3 Let $\mathcal{A}$ be an access structure and $P \subseteq \{1, 2, \ldots, n\}$. The *contraction* of $\mathcal{A}$ with respect to P, denoted by $\mathcal{A} \cdot P$, is the access structure over $\overline{P}$ given by

$$A \in \mathcal{A} \cdot P \Leftrightarrow A \cup P \in \mathcal{A}.$$

Intuitively, the contraction of $\mathcal{A}$ with respect to P is the access structure that results after the shares of the users from P were made public.

Example 1.1.3 Let us consider the access structure $\mathcal{A} = \{\{1,2\},\ \{1,2,3\},\ \{1,2,4\},\ \{1,2,3,4\},\ \{3,4\},\ \{1,3,4\},\ \{2,3,4\}\}$ and $P = \{3\}$. We obtain

$$(\mathcal{A} \cdot P)_{min} = \{\{1,2\}, \{4\}\}.$$

Definition 1.1.4 Let $\mathcal{A}$ be an access structure over the set $\mathcal{U}_1$, $\mathcal{B}$ be an access structure over the set $\mathcal{U}_2$, and $P \in \mathcal{U}_1$. The *insertion* of $\mathcal{B}$ in $\mathcal{A}$ at P, denoted by $\mathcal{A}(P \to \mathcal{B})$, is the access structure over $(\mathcal{U}_1 \setminus \{P\}) \cup \mathcal{U}_2$ given by

$$A \in \mathcal{A}(P \to \mathcal{B}) \Leftrightarrow (A \cap \mathcal{U}_1 \in \mathcal{A} \vee ((A \cap \mathcal{U}_1) \cup \{P\} \in \mathcal{A} \wedge A \cap \mathcal{U}_2 \in \mathcal{B})).$$

Intuitively, the insertion of $\mathcal{B}$ in $\mathcal{A}$ at P is the access structure obtained from $\mathcal{A}$ by replacing the element P with sets from $\mathcal{B}$.

Example 1.1.4 Let us consider the access structure $\mathcal{A} = \{\{1,2\},\ \{1,2,3\},\ \{1,2,4\},\ \{1,2,3,4\},\ \{3,4\},\ \{1,3,4\},\ \{2,3,4\}\}$ over $\{1,2,3,4\}$, the access structure $\mathcal{B} = \{\{4\}, \{4,6\},\ \{4,5\},\ \{5,6\}, \{4,5,6\}\}$ over $\{4,5,6\}$, and $P = 2$. We obtain

$$(\mathcal{A}(P \to \mathcal{B}))_{min} = \{\{1,4\}, \{3,4\}, \{1,5,6\}\}.$$

We present two useful special cases of the insertion construction.

Definition 1.1.5 Let $\mathcal{A}$ be an access structure over the set $\mathcal{U}_1$, and $\mathcal{B}$ be an access structure over the set $\mathcal{U}_2$.

The *sum* of $\mathcal{A}$ and $\mathcal{B}$, denoted by $\mathcal{A} + \mathcal{B}$, is the access structure over $\mathcal{U}_1 \cup \mathcal{U}_2$ given by

$$A \in \mathcal{A} + \mathcal{B} \Leftrightarrow (A \cap \mathcal{U}_1 \in \mathcal{A} \vee A \cap \mathcal{U}_2 \in \mathcal{B}).$$

The *product* of $\mathcal{A}$ and $\mathcal{B}$, denoted by $\mathcal{A} \cdot \mathcal{B}$, is the access structure over $\mathcal{U}_1 \cup \mathcal{U}_2$ given by

$$A \in \mathcal{A} \cdot \mathcal{B} \Leftrightarrow (A \cap \mathcal{U}_1 \in \mathcal{A} \wedge A \cap \mathcal{U}_2 \in \mathcal{B}).$$

Remark 1.1.1 $\mathcal{A} + \mathcal{B}$ is in fact $(\{\{a\}, \{b\}, \{a,b\}\}(a \to \mathcal{A}))(b \to \mathcal{B})$ and $\mathcal{A} \cdot \mathcal{B}$ is in fact $(\{\{a,b\}\}(a \to \mathcal{A}))(b \to \mathcal{B})$, where $a, b \notin \mathcal{U}_1 \cup \mathcal{U}_2$.

Example 1.1.5 Let us consider the access structure $\mathcal{A} = \{\{1,2\}, \{4\}, \{1,4\}, \{2,4\},\ \{1,2,4\}\}$ over $\{1,2,4\}$ and the access structure $\mathcal{B} = \{\{4\}, \{4,6\},\ \{4,5\},\ \{5,6\},\ \{4,5,6\}\}$ over $\{4,5,6\}$. We obtain

$$(\mathcal{A} + \mathcal{B})_{min} = \{\{1,2\}, \{4\}, \{5,6\}\},$$

$$(\mathcal{A} \cdot \mathcal{B})_{min} = \{\{4\}\}.$$

1.2 Models for Secret Sharing

Intuitively, a secret sharing scheme is a method of splitting a secret into shares such that the secret can be determined only by the authorized groups. Depending on the "quantity" of the secret-information leaked to an unauthorized group, the secret sharing schemes can be classified as

- *Perfect* secret sharing schemes - the shares of any unauthorized group give no information (in information-theoretic sense) about the secret;
- *Computational-secure* secret sharing schemes - some information about the secret is leaked to the unauthorized groups, but the problem of finding the secret is intractable[2].

  Ramp schemes (see Section 2.8), the secret sharing schemes based on information dispersal (see Section 2.1.4, Section 2.7.5), or the secret sharing schemes based on public information (see Section 2.7.6) are examples of computational-secure secret sharing schemes. Such schemes can lead to shorter shares (see also Section 1.3).

We will describe next some models for describing more accurately the notion of secret sharing scheme.

Brickell-Davenport Model

Brickell and Davenport have proposed an elegant model for secret sharing in [22]. They have defined a secret sharing scheme as a matrix $\mathcal{M}$ with some special properties. The matrix $\mathcal{M}$ has $n+1$ columns, the first one corresponding to the dealer and the rest corresponding to the users. For any $i \in \{0, 1, \ldots, n\}$, let $S(i) = \{\mathcal{M}_{r,i} | r \text{ is a row in } \mathcal{M}\}$. We obtain that $\mathcal{S} = S(0)$ and $\mathcal{S}_i = S(i)$, for all $1 \leq i \leq n$. The dealer chooses an element $S \in \mathcal{S}$ and a row r of $\mathcal{M}$ such that $\mathcal{M}_{r,0} = S$. The corresponding shares will be $I_i = \mathcal{M}_{r,i}$, for all $1 \leq i \leq n$. The matrix $\mathcal{M}$ is public information, while r is private. An authorized group A, by pooling together their shares, can identify the row r of $\mathcal{M}$ and determine the secret S as $S = \mathcal{M}_{r,0}$. In order to assure the correctness of this scheme, the matrix $\mathcal{M}$ must satisfy the following relation:

$$(\forall r, r' \text{rows of } \mathcal{M})((\forall j \in A)(\mathcal{M}_{r,j} = \mathcal{M}_{r',j}) \Rightarrow \mathcal{M}_{r,0} = \mathcal{M}_{r',0}),$$

for any authorized group A.

For a group A, $A \subseteq \{1, \ldots, n\}$ and i, $i \notin A$, we will say that A *has no information* about i^{th} share, and we will denote this by $A \not\to i$, if

$$(\forall r \text{ row of } \mathcal{M})(\forall \beta \in S(i))(\exists r' \text{ row of } \mathcal{M})((\forall j \in A)(\mathcal{M}_{r,j} = \mathcal{M}_{r',j}) \wedge \mathcal{M}_{r',i} = \beta).$$

[2] A problem is called *intractable* if there is no polynomial deterministic/randomized algorithm for solving it.

Otherwise, we will say that A *has some information* about i^{th} share and we will denote this by $A \to i$.

We will say that A *knows* the i^{th} share, denoted by $A \Rightarrow i$, if

$$(\forall r, r'\text{rows of } \mathcal{M})((\forall j \in A)(\mathcal{M}_{r,j} = \mathcal{M}_{r',j}) \Rightarrow \mathcal{M}_{r,i} = \mathcal{M}_{r',i}).$$

Thus, the authorized access structure can be expressed as

$$\mathcal{A} = \{A \subseteq \{1, \ldots, n\} | A \Rightarrow 0\}.$$

A secret sharing scheme $\mathcal{M}$ is called *perfect* if

$$(\forall A \in \mathcal{P}(\{1, \ldots, n\}))((A \to 0) \Rightarrow (A \Rightarrow 0)).$$

Intuitively, this property means that no information about the secret is leaked to the unauthorized groups.

Brickell-Stinson Model

Brickell and Stinson have proposed an interesting model for secret sharing in [23]. We present its refined variant, as described by Stinson in [141]. In this model, a secret sharing scheme is represented as a special set $\mathcal{F}$ of *distribution rules.* A distribution rule is a function $f : \{0, 1, \ldots, n\} \to \mathcal{S} \cup \cup_{i=1}^{n} \mathcal{S}_i$ such that $f(0) \in \mathcal{S}$ and $f(i) \in \mathcal{S}_i$, for all $1 \leq i \leq n$. If we consider $\mathcal{S}_0 = \mathcal{S}$, then a secret sharing scheme can be viewed as a special subset of the product[3] of the family $(\mathcal{S}_i | i \in \{0, 1, \ldots, n\})$. More exactly, we say that a set $\mathcal{F} \subseteq \prod_{i \in \{0,1,\ldots,n\}} \mathcal{S}_i$ is a perfect $\mathcal{A}$-secret sharing scheme if the following conditions hold:

- $(\forall A \in \mathcal{A})(\forall f, g \in \mathcal{F})((\forall i \in A)(f(i) = g(i)) \Rightarrow f(0) = g(0));$
- $(\forall B \in \overline{\mathcal{A}})(\forall f \in \prod_{i \in B} \mathcal{S}_i)(\exists \lambda(B, f) \in \mathbf{N})(\forall S \in \mathcal{S})$
  $(|\{g \in \mathcal{F} | g(0) = S \wedge (\forall i \in B)(f(i) = g(i))\}| = \lambda(B, f)).$

An element $f \in \mathcal{F}$ represents a possible distribution of shares to the users, where $f(0)$ is the secret and $f(i)$ is the share corresponding to the i^{th} user, for all $1 \leq i \leq n$.

Entropy-based Models

Karnin, Greene, and Hellman [97], Kothari [98], and later on, Capocelli, De Santis, Gargano, and Vaccaro [27] have used the concept of *entropy* (see Appendix B) for measuring the quantity of uncertainty about the secret.

The next definition is taken from [11]:

[3] In general, the *product* of a family of sets $(\mathcal{S}_i | i \in I)$, denoted by $\prod_{i \in I} \mathcal{S}_i$, is defined as the set $\{f : I \to \cup_{i \in I} \mathcal{S}_i | (\forall i \in I)(f(i) \in \mathcal{S}_i)\}$.

Definition 1.2.1 Suppose we have n users labeled with the numbers $1, \ldots, n$ and consider a set of groups $\mathcal{A} \subseteq \mathcal{P}(\{1, 2, \ldots, n\})$. A *perfect $\mathcal{A}$-secret sharing scheme* is a collection of random variables $(\mathtt{S}, \mathtt{I}_1, \ldots, \mathtt{I}_n)$ such that

- (*correctness*) - for any $A \in \mathcal{A}$, $H(\mathtt{S}|\{\mathtt{I}_i \mid i \in A\}) = 0$;
- (*security*) - for any $A \in \overline{\mathcal{A}}$, $H(\mathtt{S}|\{\mathtt{I}_i \mid i \in A\}) = H(\mathtt{S})$.

In a non-perfect secret sharing scheme, the second item is replaced by

- for any $A \in \overline{\mathcal{A}}$, $H(\mathtt{S}|\{\mathtt{I}_i \mid i \in A\}) > 0$.

A model based on binary relations

We propose a new model based on binary relations.

Definition 1.2.2 Suppose we have n users labeled with the numbers $1, \ldots, n$ and consider a set of groups $\mathcal{A} \subseteq \mathcal{P}(\{1, 2, \ldots, n\})$. A *perfect $\mathcal{A}$-secret sharing scheme* is a pair $(\mathit{split}, \mathit{combine})$, where

- *split* is a function from $\mathcal{S}$ to $\mathcal{S}_1 \times \cdots \times \mathcal{S}_n$;
- *combine* is a binary relation from $\cup_{A \in \mathcal{P}(\{1,2,\ldots,n\})} A \times \mathcal{S}_A$ to $\mathcal{S}$, where $\mathcal{S}_A$ denotes the set $\mathcal{S}_{i_1} \times \cdots \times \mathcal{S}_{i_l}$, for any $A = \{i_1, \ldots, i_l\}$, $i_1 < \cdots < i_l$,

such that

1. (*correctness*)

   $$(\forall S, S' \in \mathcal{S})(\forall A \in \mathcal{A})(((A, \mathit{pr}_A(\mathit{split}(S))), S') \in \mathit{combine} \Leftrightarrow S' = S),$$

   where $\mathit{pr}_A : \mathcal{S}_1 \times \cdots \times \mathcal{S}_n \to \mathcal{S}_A$ is given by $\mathit{pr}_A(I_1, \ldots, I_n) = (I_{i_1}, \ldots, I_{i_l})$ for any $A = \{i_1, \ldots, i_l\}$, $i_1 < \cdots < i_l$;

2. (*security*)

   $$(\forall S \in \mathcal{S})(\forall A \in \overline{\mathcal{A}})(\{S' \in \mathcal{S} | ((A, \mathit{pr}_A(\mathit{split}(S))), S') \in \mathit{combine}\} = \mathcal{S}).$$

If $(\mathit{split}, \mathit{combine})$ is an $\mathcal{A}$-secret sharing scheme, we will say that the secret sharing scheme $(\mathit{split}, \mathit{combine})$ *realizes* the access structure $\mathcal{A}$. The relation $\mathit{split}(S) = (I_1, \ldots, I_n)$ will be also denoted as

$$S \leftrightarrow_{\mathcal{A}}^{(\mathit{split},\mathit{combine})} I_1, \ldots, I_n.$$

This notation will be used also to denote that S can be obtained by applying *combine* to any authorized subset of $\{I_1, \ldots, I_n\}$. In case that $\mathcal{A}$ and $(\mathit{split}, \mathit{combine})$ are clear from context, this notation will be simplified to $S \leftrightarrow I_1, \ldots, I_n$.

Through this thesis, secret sharing schemes will be described by specifying the set of secrets, the sets of shares corresponding to each participant, the descriptions of the function *split* and of $combine_A$, for some minimal authorized group A (we remark that it is sufficient to specify the reconstruction method only for the minimal authorized groups - any authorized group may decide first on a minimal authorized subgroup which will reconstruct the secret).

1.3 Information Rates

A very important aspect in secret sharing is the size of the shares. The efficiency and the security of a system decrease as the quantity of the information that must be kept secret increases. Several authors have studied this subject and they have introduced several terms and indicators for characterizing the size of the shares in a secret sharing scheme.

The concept of entropy can be used for analyzing the amount the information in a secret sharing scheme. Indeed, the entropy $H(\mathcal{X})$ (or $H(\mathtt{X})$ in case of a random variable) can be interpreted as the average number of bits required for representing an element of $\mathcal{X}$ with respect to some probability distribution. Thus, $H(\mathtt{S})$ can be viewed as the size of the secret and $H(\mathtt{I}_i)$ can be viewed as the size of the i^{th} share. Karnin, Greene, and Hellman [97] have proved that, for any perfect threshold secret sharing scheme, the relation

$$H(\mathtt{S}) \leq H(\mathtt{I}_i)$$

holds true, for all $1 \leq i \leq n$. Later on, Capocelli, De Santis, Gargano, and Vaccaro [27] have extended this result to the case of any perfect secret sharing scheme.

Brickell [20] has introduced the *information rate* ρ of a secret sharing scheme as $\rho = \frac{log_2(|\mathcal{S}_{shares}|)}{log_2(|\mathcal{S}|)}$ and used the term *ideal* for a perfect secret sharing schemes with information rate 1.

Blundo, De Santis, Gargano, and Vaccaro [16] have introduced more general information rates using the notion of entropy. They have defined the (*worst-case*) *information rate* of a secret sharing scheme as

$$\rho = \frac{H(\mathtt{S})}{max_{i \in \{1,\ldots,n\}} H(\mathtt{I}_i)}.$$

For uniform probability distributions over the secrets and shares sets, the worst-case information rate becomes

$$\rho = \frac{log_2|\mathcal{S}|}{max_{i \in \{1,\ldots,n\}} log_2|\mathcal{S}_i|},$$

as it has already been introduced by Brickell and Stinson [23].

Blundo, De Santis, Gargano, Vaccaro have also introduced the *average information rate* of a secret sharing scheme as

$$\overline{\rho} = \frac{H(\mathtt{S})}{\frac{\sum_{i=1}^{n} H(\mathtt{I}_i)}{n}}.$$

For uniform probability distributions over the secrets and shares sets, the average information rate becomes

$$\overline{\rho} = \frac{n log_2|\mathcal{S}|}{\sum_{i=1}^{n} log_2|\mathcal{S}_i|},$$

as it has already been introduced in [14], [104], or [105].

The information rates of any perfect secret sharing scheme satisfy the relation $0 < \rho \leq \overline{\rho} \leq 1$. A perfect $\mathcal{A}$-secret sharing scheme with the information rate $\rho = 1$ will be called *ideal*. Any access structure $\mathcal{A}$ which can be realized by an ideal secret sharing scheme will be also called ideal. The researchers have succeeded in finding ideal realizations for many (classes of) access structures (e.g., the threshold, multilevel or compartmented ones). For the non-ideal access structures it is interesting to find better upper bounds for the information rates. The first non-ideal access structure has been presented by Benaloh and Leichter [9]. It is given by $\mathcal{A}_{min} = \{\{1,2\},\{2,3\},\{3,4\}\}$. Cappocelli, De Santis, Gargano, and Vaccaro [27] have proven that, for this access structure, there is a user that receives a share of size at least $\frac{3}{2}$ times the size of secret (with respect to perfect secret sharing). Csirmaz [38] has found an access structure over n users so that any perfect secret sharing scheme will assign a share of length about $\frac{n}{log\ n}$ times the size of the secret to some user.

Chapter 2

Constructions of Secret Sharing Schemes

2.1 Threshold Secret Sharing Schemes

In the first secret sharing schemes only the number of the participants in the reconstruction phase was important for recovering the secret. Such schemes have been referred to as *threshold* secret sharing schemes.

Definition 2.1.1 Let $n \geq 2$, $2 \leq k \leq n$. The access structure

$$\mathcal{A} = \{A \in \mathcal{P}(\{1, 2, \ldots, n\}) \mid \ |A| \geq k\}$$

will be referred to as the (k, n)-*threshold* access structure.

We obtain $\mathcal{A}_{min} = \{A \in \mathcal{P}(\{1, 2, \ldots, n\}) \mid \ |A| = k\}$, $\overline{\mathcal{A}} = \{A \in \mathcal{P}(\{1, 2, \ldots, n\}) \mid \ |A| \leq k - 1\}$, and $\overline{\mathcal{A}}_{max} = \{A \in \mathcal{P}(\{1, 2, \ldots, n\}) \mid \ |A| = k - 1\}$. In this case, an $\mathcal{A}$-secret sharing scheme will be referred to as an (k, n)*-threshold secret sharing scheme.*

2.1.1 Blakley's Scheme

In Blakley's scheme [12], as presented in [123], the secret is an element of the vector space $\mathbf{GF}_q^k$. The shares are any n distinct $(k-1)$-dimensional hyperplanes that contain the secret, where an $(k-1)$-dimensional hyperplane is a set of form

$$\{(x_1, \ldots, x_k) \in \mathbf{GF}_q^k | \alpha_1 \cdot x_1 + \cdots + \alpha_k \cdot x_k = \beta\},$$

where $\alpha_1, \ldots, \alpha_k, \beta$ are arbitrary elements of the field $\mathbf{GF}_q$. The secret can be obtained by intersecting any k shares.

In Figure 2.1 we pictorially describe the case $k = 2$.

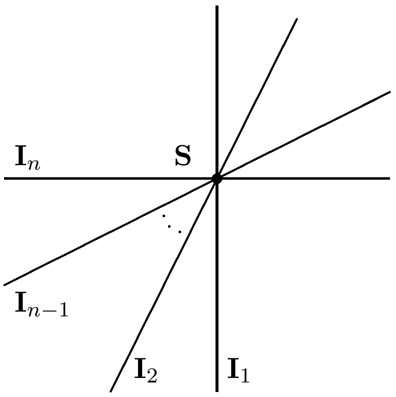

Figure 2.1: Blakley's scheme for threshold $k = 2$

Blakley's scheme is not perfect because any unauthorized group knows that the secret lies on the intersection of their hyperplanes. Nevertheless, this scheme can be modified to achieve perfect security by choosing the secret as a single coordinate of a point in $\mathbf{GF}_q^k$, with the cost of affecting the information rate.

2.1.2 Shamir's Scheme

Shamir's scheme [134] is based on polynomial interpolation - given any k pairs $(x_1, y_1), \ldots, (x_k, y_k)$ with $x_i \neq x_j$ for all $1 \leq i < j \leq k$, there is one and only one polynomial $P(x)$ of degree $k-1$ such that $P(x_i) = y_i$, for all $1 \leq i \leq k$.

- The secret S is chosen as the free coefficient of a random polynomial P of degree $k-1$ over the field of the positive integers modulo a large prime;

- The shares $I_1, \ldots, I_n$ are chosen as $I_i = P(x_i)$, for all $1 \leq i \leq n$, where $x_1, \ldots, x_n$ are pairwise distinct public values;

- Having the shares $\{I_i | i \in A\}$, for some group A with $|A| = k$, the secret can be obtained using Lagrange's interpolation formula as

$$S = \sum_{i \in A} (I_i \cdot \prod_{j \in A \setminus \{i\}} \frac{x_j}{x_j - x_i}).$$

If $P(x) = a_{k-1}x^{k-1} + \cdots + a_1x^1 + a_0$, the secret can also be obtained from the shares $I_{i_1}, \ldots, I_{i_k}$ by solving the system of equations

$$\left\{ \begin{array}{rcl} a_{k-1}x_{i_1}^{k-1} + \cdots + a_1x_{i_1}^1 + a_0 & = & I_{i_1} \\ & \vdots & \\ a_{k-1}x_{i_k}^{k-1} + \cdots + a_1x_{i_k}^1 + a_0 & = & I_{i_k} \end{array} \right. .$$

The above system has k equations and k unknowns $(a_{k-1}, \ldots, a_1, a_0)$ and it has a unique solution because the determinant

$$\left| \begin{array}{llll} x_{i_1}^{k-1} & \cdots & x_{i_1}^1 & 1 \\ x_{i_2}^{k-1} & \cdots & x_{i_2}^1 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ x_{i_k}^{k-1} & \cdots & x_{i_k}^1 & 1 \end{array} \right|$$

is a non-zero Vandermonde determinant. By this point of view, the polynomial $P(x)$ can be chosen of degree at most $k-1$.

Shamir's scheme is perfect. Indeed, having only $k-1$ shares, the system of equations

$$\left\{ \begin{array}{rcl} a_{k-1}x_{i_1}^{k-1} + \cdots + a_1x_{i_1}^1 & = & I_{i_1} - a_0 \\ & \vdots & \\ a_{k-1}x_{i_{k-1}}^{k-1} + \cdots + a_1x_{i_{k-1}}^1 & = & I_{i_{k-1}} - a_0 \end{array} \right.$$

with $k-1$ equations and $k-1$ unknowns $(a_{k-1}, \ldots, a_1)$ has a unique solution, for any a_0. Thus, all possible values of the secret are equally likely.

Ghodosi, Pieprzyk, and Safavi-Naini [69] have remarked that, if the degree of the polynomial $P(x)$ is known to be $k-1$ (or, equivalently, $a_{k-1} \neq 0$), then the scheme is not perfect. Indeed, in this case, any $k-1$ users can determine an element b_0 which is not the secret, i.e., $b_0 \neq a_0$. More exactly, a polynomial $Q(x) = b_{k-2}x^{k-2} + \cdots + b_1x^1 + b_0$ can be determined using Lagrange's interpolation formula, such that $Q(x_{i_j}) = I_{i_j} = P(x_{i_j})$, for all $1 \leq j \leq k-1$, leading to the system

$$\left\{ \begin{array}{rcl} a_{k-1}x_{i_1}^{k-1} + (a_{k-2} - b_{k-2})x_{i_1}^{k-2} + \cdots + (a_1 - b_1)x_{i_1}^1 + (a_0 - b_0) & = & 0 \\ & \vdots & \\ a_{k-1}x_{i_{k-1}}^{k-1} + (a_{k-2} - b_{k-2})x_{i_{k-1}}^{k-2} + \cdots + (a_1 - b_1)x_{i_{k-1}}^1 + (a_0 - b_0) & = & 0 \end{array} \right. .$$

If we assume, by contradiction, that $a_0 = b_0$, the above system with $k-1$ equations and $k-1$ unknowns $(a_{k-1}, \ldots, a_1)$ has in this case a unique solution, namely $a_{k-1} = 0, a_{k-2} = b_{k-2}, \ldots, a_1 = b_1$ that contradicts that $a_{k-1} \neq 0$. Thus, any $k-1$ users can determine an element b_0 which is not the secret and, consequently, their uncertainty about the secret does not coincide with the uncertainty of an outsider.

Shamir has proposed choosing $x_i = i$, for all $1 \leq i \leq n$. In this case, the secret can be reconstructed as

$$\sum_{i \in A} (I_i \cdot \prod_{j \in A \setminus \{i\}} \frac{j}{j-i}),$$

for any group A with $|A| = k$.

Example 2.1.1 (with artificially small parameters)
Let $n = 5$ and $k = 3$. Let us consider the polynomial $P(x) = 2x^2 + 7x + 10$ over the field $\mathbf{Z}_{11}$. The secret is $S = 10$ and the corresponding shares are $I_1 = P(1) = 8$, $I_2 = P(2) = 10$, $I_3 = P(3) = 5$, $I_4 = P(4) = 4$, and $I_5 = P(5) = 7$. Having the shares I_1, I_2 and I_4, the secret can be reconstructed as

$$8 \cdot \frac{2}{2-1} \cdot \frac{4}{4-1} + 10 \cdot \frac{1}{1-2} \cdot \frac{4}{4-2} + 4 \cdot \frac{1}{1-4} \cdot \frac{2}{2-4}.$$

As Shamir himself has remarked, his scheme has some interesting features:

- The size of every share does not exceed the size of the secret (Shamir's scheme is ideal);
- It is dynamic, in sense that, if the threshold k is kept fixed, some existing shadows can be excluded or some new shadows can be generated, without affecting the other shadows;
- The shadows can be changed without changing the secret[1].

McEliece and Sarwate [112] have remarked that Shamir's scheme is closely related to Reed-Solomon codes and that decoding algorithms for such codes can be used for generalizing Shamir's scheme.

2.1.3 Threshold Secret Sharing based on the Chinese Remainder Theorem

We present next the most important threshold secret sharing schemes based on the Chinese remainder theorem.

2.1.3.1 Mignotte's Scheme

Mignotte's threshold secret sharing scheme [117] uses special sequences of integers, referred to as *Mignotte sequences*.

[1] This is in fact the *proactivity* feature that will be discussed in Section 3.4

Definition 2.1.2 Let n be an integer, $n \geq 2$, and $2 \leq k \leq n$. An (k,n)-*Mignotte sequence* is a sequence of pairwise coprime positive integers $p_1 < p_2 < \cdots < p_n$ such that

$$\prod_{i=0}^{k-2} p_{n-i} < \prod_{i=1}^{k} p_i.$$

The above relation is equivalent with

$$max_{1 \leq i_1 < \cdots < i_{k-1} \leq n}(p_{i_1} \cdots p_{i_{k-1}}) < min_{1 \leq i_1 < \cdots < i_k \leq n}(p_{i_1} \cdots p_{i_k}).$$

Given a publicly known (k,n)-Mignotte sequence, the scheme works as follows:

- The secret S is chosen as a random integer such that $\beta < S < \alpha$, where $\alpha = \prod_{i=1}^{k} p_i$ and $\beta = \prod_{i=0}^{k-2} p_{n-i}$;
- The shares I_i are chosen as $I_i = S \texttt{ mod } p_i$, for all $1 \leq i \leq n$;
- Given k distinct shares $I_{i_1}, \ldots, I_{i_k}$, the secret S is recovered using the standard Chinese remainder theorem, as the unique solution modulo $p_{i_1} \cdots p_{i_k}$ of the system

$$\left\{ \begin{array}{lcl} x & \equiv & I_{i_1} \ mod \ p_{i_1} \\ & \vdots & \\ x & \equiv & I_{i_k} \ mod \ p_{i_k} \end{array} \right. .$$

Indeed, the secret S is an integer solution of the above system by the choice of the shadows. Moreover, S lies in $\mathbf{Z}_{p_{i_1} \cdots p_{i_k}}$ because $S < \alpha$. On the other hand, having only $k-1$ distinct shares $I_{i_1}, \ldots, I_{i_{k-1}}$, we obtain only that $S \equiv x_0 \texttt{ mod } p_{i_1} \cdots p_{i_{k-1}}$, where x_0 is the unique solution modulo $p_{i_1} \cdots p_{i_{k-1}}$ of the resulted system ($S > \beta \geq p_{i_1} \cdots p_{i_{k-1}} > x_0$). Therefore, in order to assure a reasonable level of security, (k,n)-Mignotte sequences with a large factor $\frac{\alpha - \beta}{\beta}$ must be chosen (a method of generating such sequences is presented in [99, page 9]).

Obviously, Mignotte's scheme is not perfect, but it can lead to small shares and, thus, can be used in applications in which the compactness of the shares is the deciding factor.

We have extended Mignotte's threshold scheme in [79] by introducing the generalized Mignotte sequences whose elements are not necessarily pairwise coprime.

Definition 2.1.3 Let n be an integer, $n \geq 2$, and $2 \leq k \leq n$. A *generalized* (k,n)-*Mignotte sequence* is a sequence $p_1, \ldots, p_n$ of positive integers such that

$$max_{1 \leq i_1 < \cdots < i_{k-1} \leq n}([p_{i_1}, \ldots, p_{i_{k-1}}]) < min_{1 \leq i_1 < \cdots < i_k \leq n}([p_{i_1}, \ldots, p_{i_k}]).$$

It is easy to see that every (k,n)-Mignotte sequence is a generalized (k,n)-Mignotte sequence. Moreover, if we multiply every element of a (generalized) (k,n)-Mignotte sequence $p_1,\ldots,p_n$ by a fixed element $\delta \in \mathbf{Z}$, $(\delta, p_1\cdots p_n) = 1$, we obtain a generalized (k,n)-Mignotte sequence.

The generalized Mignotte scheme works like Mignotte's scheme, with $\alpha = min_{1\leq i_1<\cdots<i_k\leq n}([p_{i_1},\ldots,p_{i_k}])$ and $\beta = max_{1\leq i_1<\cdots<i_{k-1}\leq n}([p_{i_1},\ldots,p_{i_{k-1}}])$. In this case, the general variant of the Chinese remainder theorem must be used for recovering the secret. Example 2.1.2 illustrates our extension.

Example 2.1.2 (with artificially small parameters)
Let $n = 5$, $k = 3$, $p_1 = 10$, $p_2 = 14$, $p_3 = 18$, $p_4 = 22$, $p_5 = 26$. The secret is $S = 615$ and the corresponding shares are $I_1 = 5$, $I_2 = 13$, $I_3 = 3$, $I_4 = 21$, and $I_5 = 17$. In the reconstruction phase, if we have the first three shares, the secret S can be recovered as the unique solution modulo 630 of the system of equations

$$\left\{\begin{array}{lcl} x & \equiv & 5 \; mod \; 10 \\ x & \equiv & 13 \; mod \; 14 \\ x & \equiv & 3 \; mod \; 18 \end{array}\right. .$$

We will use **CRT_Ore** algorithm described in Appendix A. We obtain $c_1 = 63$, $c_2 = 45$, and $c_3 = 35$. We will use the extended Euclidean algorithm. We start with $V_{63} = (1,0,0)$, $V_{45} = (0,1,0)$, $V_{35} = (0,0,1)$, and we obtain next $V_{(63,45)} = (-2,3,0)$, $V_{(9,35)} = (-8,12,-1)$, and, thus, $\alpha_1 = -8$, $\alpha_2 = 12$, $\alpha_3 = -1$. We obtain that the solution of the system is $S = ((-8)\cdot 63\cdot 5 + 12\cdot 45\cdot 13 + (-1)\cdot 35\cdot 3) \texttt{ mod } 630 = 615$.

2.1.3.2 Asmuth-Bloom Scheme

This scheme, proposed by Asmuth and Bloom in [1], also uses special sequences of integers. More exactly, a sequence of pairwise coprime positive integers $p_0, p_1 < \cdots < p_n$ is chosen such that

$$p_0 \cdot \prod_{i=0}^{k-2} p_{n-i} < \prod_{i=1}^{k} p_i.$$

Given a publicly known Asmuth-Bloom sequence, the scheme works as follows:

- The secret S is chosen as a random element of the set $\mathbf{Z}_{p_0}$;
- The shares I_i are chosen as $I_i = (S + \gamma\cdot p_0) \texttt{ mod } p_i$, for all $1 \leq i \leq n$, where γ is an arbitrary integer such that $S + \gamma\cdot p_0 \in \mathbf{Z}_{p_1\cdots p_k}$;
- Given k distinct shares $I_{i_1},\ldots,I_{i_k}$, the secret S can be obtained as $S = x_0 \texttt{ mod } p_0$, where x_0 is obtained, using the standard variant of the

Chinese remainder theorem, as the unique solution modulo $p_{i_1} \cdots p_{i_k}$ of the system

$$\left\{ \begin{array}{lll} x & \equiv & I_{i_1} \; mod \; p_{i_1} \\ & \vdots & \\ x & \equiv & I_{i_k} \; mod \; p_{i_k} \end{array} \right. .$$

Goldreich, Ron, and Sudan [72] have proposed choosing $p_0, p_1, \ldots, p_n$ as prime numbers of the same size. Quisquater, Preneel, and Vandewalle [130] have proven that, by choosing $p_0, p_1, \ldots, p_n$ as consecutive primes, the resulted schemes are asymptotically perfect and asymptotically ideal (for technical details, the reader is referred to [130]).

The sequences used in Asmuth-Bloom scheme can be also generalized by allowing modules that are not necessarily pairwise coprime in an obvious manner. We can use any sequence $p_0, p_1, \cdots, p_n$ such that

$$p_0 \cdot max_{1 \le i_1 < \cdots < i_{k-1} \le n}([p_{i_1}, \ldots, p_{i_{k-1}}]) < min_{1 \le i_1 < \cdots < i_k \le n}([p_{i_1}, \ldots, p_{i_k}]).$$

It is easy to see that if we multiply the elements, excepting the first one, of a (generalized) Asmuth-Bloom sequence $p_0, p_1, \cdots, p_n$ with a fixed element $\delta \in \mathbf{Z}$, $(\delta, p_0 \cdots p_n) = 1$, we obtain a generalized Asmuth-Bloom sequence.

2.1.4 Threshold Secret Sharing based on Information Dispersal

Krawczyk [100] has proposed combining perfect threshold secret sharing schemes with encryption and information dispersal in order to decrease the size of shares with the cost of decreasing the level of security. Threshold information dispersal schemes have been introduced by Rabin [131].

Definition 2.1.4 Let n be an integer, $n \geq 2$, and $2 \leq k \leq n$. Informally, an (k,n)-*threshold information dispersal scheme* is a method of generating $(S, (F_1, \ldots, F_n))$ such that for any set A with $|A| = k$, the problem of finding the element S, given the set $\{F_i \mid i \in A\}$, is “easy”.

The elements S will be referred to as the *information* and $F_1, \ldots, F_n$ will be referred to as the *fragments* of S.

As it can be seen from the above definition, the difference between information dispersal and secret sharing is that in the case of the information dispersal there is no restriction on the unauthorized groups.

Krawczyk has presented a threshold information dispersal scheme very similar to Shamir’s threshold secret sharing scheme:

- The information S is chosen as the vector of the coefficients of a random polynomial $P(x)$ of degree $k-1$ over some field;

- The fragments $F_1, \ldots, F_n$ are chosen as follows: $F_i = P(x_i)$, for all $1 \leq i \leq n$, where $x_1, \ldots, x_n$ are pairwise distinct public values;
- Having the fragments $\{F_i | i \in A\}$, for some group A with $|A| = k$, the polynomial $P(x)$ and, thus, the information S, can be obtained using Lagrange's interpolation formula as

$$\sum_{i \in A} (F_i \cdot \prod_{j \in A \setminus \{i\}} \frac{x - x_j}{x_i - x_j}).$$

The difference between this scheme and Shamir's threshold secret sharing scheme is that in case of the information dispersal scheme, the information is chosen as the whole polynomial as opposed to the perfect secret sharing scheme in which the secret is chosen only as the free coefficient.

Krawczyk's technique for computational-secure threshold secret sharing is described next.

- The dealer makes public an encryption method $e_{\{\}}$;
- The dealer chooses a random key K and computes $\overline{S} = e_K(S)$;
- The dealer uses an (k, n)-threshold information dispersal scheme in order to split the information $\overline{S}$ into n fragments $F_1, \ldots, F_n$;
- The dealer uses a perfect (k, n)-threshold secret sharing scheme for constructing the shares $K_1, \ldots, K_n$ corresponding to the secret K;
- The shares are chosen as $I_i = (F_i, K_i)$, for all $1 \leq i \leq n$, and then the shares are securely distributed to users;
- Given k distinct shares $I_{i_1} = (F_{i_1}, K_{i_1}), \ldots, (F_{i_k}, K_{i_k})$, the secret S can be recovered as follows:
    - The element $\overline{S}$ is recovered using the reconstruction algorithm applied to $F_{i_1}, \ldots, F_{i_k}$;
    - The key K is recovered using the reconstruction algorithm applied to $K_{i_1}, \ldots, K_{i_k}$;
    - The secret S is recovered as $S = d_K(\overline{S})$.

The length of the i^{th} share is $|F_i| + |K_i|$ and, thus, it depends both on the information dispersal scheme and on the perfect secret sharing scheme used. In case of using a length preserving[2] encryption function and ideal threshold information dispersal (i.e., $|F_i| = \frac{|\overline{S}|}{k}$, for all $1 \leq i \leq n$) and ideal secret sharing, each share is of length $\frac{|S|}{k} + |K|$.

[2] An encryption function $e_{\{\}}$ is *length preserving* if $|e_K(x)| = |x|$, for any key K and any plaintext x.

2.2 Unanimous Consent Schemes

In case $\mathcal{A} = \mathcal{A}_{min} = \{1, 2, \ldots, n\}$, an $\mathcal{A}$-secret sharing scheme will be referred to as a *unanimous consent* secret sharing scheme of rank n. In these schemes, the presence of all users is required in order to recover the secret. A unanimous consent secret sharing scheme of rank n is equivalent with an (n, n)-threshold secret sharing scheme and, thus, any (n, n)-threshold secret sharing scheme can be used in order to realize unanimous consent. It is important to design more efficient unanimous consent schemes because such schemes are used as building blocks in general secret sharing (see Section 2.7.1 or Section 2.7.2) or in dealer-free secret sharing (see Section 3.2).

Karnin, Greene, and Hellman have proposed a very simple unanimous consent scheme in [97]:

- The secret S is chosen as a random number from $\mathbf{Z}_m$;
- The dealer generates the shares I_i as random numbers from $\mathbf{Z}_m$, for all $1 \leq i \leq n-1$, and $I_n = S - \sum_{i=1}^{n-1} I_i \texttt{ mod } m$;
- The secret S can be reconstructed as $S = \sum_{i=1}^{n} I_i \texttt{ mod } m$.

2.3 Secret Sharing for Graph-based Access Structures

The 2-homogeneous access structures (i.e., the access structures in which all minimal access sets have two elements) are also referred to as *graph-based* access structures because in this case the minimal access groups can be specified by the edges of a graph.

Graph-based access structures have proven to be very important mostly by the results on information rates of the secret sharing schemes which realize them (see, for example, [14], [17], [27], [23], [22], [142]). We mention that the first access structure which cannot be realized by an ideal secret sharing scheme (see also Section 1.3) is specified by the graph represented in Figure 2.2.

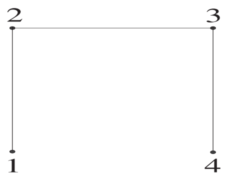

Figure 2.2: Graph representation of a non-ideal access structure

We present an interesting result due to Brickell and Davenport [22]:

Theorem 2.3.1 *Let G be a connected graph. Then there is an ideal secret sharing scheme for the access structure specified by G if and only if G is a complete multipartite*[3] *graph.*

A construction of an ideal secret sharing scheme that realizes the access structure specified by the graph $K_{n_1,n_2,\ldots,n_l}$ is presented next ([142]). Let $V_1,\ldots,V_l$ be the parts of the graph $K_{n_1,n_2,\ldots,n_l}$. The dealer chooses some pairwise distinct elements $x_1,\ldots,x_l \in \mathbf{GF}_q$. The shares corresponding to some secret $S \in \mathbf{GF}_q$ will be defined as $I_i = x_j S + r$, for all $i \in V_j$ and $1 \leq j \leq l$, where r is an arbitrary fixed element from $\mathbf{GF}_q$. Any two users u_1, u_2, $u_1 \in V_{j_1}$, $u_2 \in V_{j_2}$, $j_1 \neq j_2$, can obtain the secret S as $S = (I_{u_1} - I_{u_2})(x_{j_1} - x_{j_2})^{-1}$.

2.4 Weighted Threshold Secret Sharing Schemes

In a *weighted threshold* secret sharing scheme, a positive weight is associated to each user and the secret can be reconstructed if and only if the sum of the weights of the participants is greater than or equal to a fixed threshold. Such schemes have been considered for the first time by Shamir [134]. Shamir has discussed the case of sharing a secret between the executives of a company such that the secret can be recovered by three executives, or by an executive and a vice-president, or by the president alone. Shamir's solution for this case is based on a threshold secret sharing scheme with the threshold 3. The main idea is to give more shares to the more important users. Thus, the president receives three shares, each vice-president receives two shares and, finally, every simple executive receives a single share.

The weighted threshold access structures can be introduced as follows.

Definition 2.4.1 Let $n \geq 2$, $\omega = (\omega_1,\ldots,\omega_n)$ be a sequence of positive integers, and a positive integer w such that $2 \leq w \leq \sum_{i=1}^{n} \omega_i$. The access structure

$$\mathcal{A} = \{A \in \mathcal{P}(\{1,2,\ldots,n\}) \mid \sum_{i \in A} \omega_i \geq w\}$$

will be referred to as the (ω, w, n)-*weighted threshold access structure.*

The parameters $\omega_1,\ldots,\omega_n$ will be referred to as the *weights* and w will be referred to as the *threshold* of the scheme. If $\mathcal{A}$ is an (ω, w, n)-weighted threshold access structure, then any $\mathcal{A}$-secret sharing scheme will be referred to as an (ω, w, n)-*weighted threshold secret sharing scheme.*

An (k, n)-threshold secret sharing scheme is nothing else than an (ω, w, n)-weighted threshold secret sharing scheme with $\omega_1 = \cdots = \omega_n = 1$ and $w = k$.

[3] The *complete multipartite graph* $K_{n_1,n_2,\ldots,n_l}$ is a graph with $\sum_{j=1}^{l} n_j$ vertices in which the set of vertices is partitioned in parts of size n_j, $1 \leq j \leq l$, such that vw is an edge if and only if v and w are in different parts.

Weighted threshold secret sharing has been considered, for instance, in [4], [5], and [118].

Benaloh and Leichter [9] have proven that there exist access structures that are not weighted threshold. We present their example that proves this statement.

Example 2.4.1 (Benaloh and Leichter [9])
Let $n = 4$ and $\mathcal{A}_{min} = \{\{1,2\},\{3,4\}\}$. Suppose that this access structure is a weighted threshold access structure with the weights ω_1, ω_2, ω_3, ω_4, and the threshold w. Thus, $\omega_1+\omega_2 \geq w$ and $\omega_3+\omega_4 \geq w$. If we sum these inequalities we obtain $\omega_1+\omega_2+\omega_3+\omega_4 \geq 2w$, and, thus, $2\cdot max(\omega_1,\omega_2)+2\cdot max(\omega_3,\omega_4) \geq 2w$ which leads to $max(\omega_1,\omega_2) + max(\omega_3,\omega_4) \geq w$. Thus, one of the sets $\{1,3\}$, $\{1,4\}$, $\{2,3\}$ or $\{2,4\}$ is an authorized access set!

As Shamir has suggested in [134], the simplest method to realize an (ω, w, n)-weighted threshold secret sharing scheme is to use a (w, N)-threshold secret sharing scheme, where $N = \sum_{i=1}^{n} \omega_i$. More exactly, if we consider that $s_1, \ldots, s_N$ are the shares corresponding to a secret S with respect to an arbitrary (w, N)-threshold secret sharing scheme, we can define the shares corresponding to the weighted threshold access structure as $I_i = \{s_j | j \in P_i\}$, for all $1 \leq i \leq n$, where $\{P_1, \ldots, P_n\}$ is an arbitrary partition of the set $\{1, 2, \ldots, N\}$ such that $|P_i| = \omega_i$, for all $1 \leq i \leq n$.

Weighted Threshold Secret Sharing based on the Chinese Remainder Theorem

In [86] we have extended the threshold secret sharing schemes based on the Chinese remainder theorem in order to realize weighted threshold secret sharing.

We first extend the (generalized) threshold Mignotte sequences in a natural manner.

Definition 2.4.2 Let $n \geq 2$, $\omega = (\omega_1, \ldots, \omega_n)$ be a sequence of weights, and w be a threshold. An (ω, w, n)-*Mignotte sequence* is a sequence $p_1, \ldots, p_n$ of positive integers such that

$$\underset{\sum_{i\in A}\omega_i \leq w-1}{max_{A\in\mathcal{P}(\{1,2,\ldots,n\})}}([\{p_i | i \in A\}]) < \underset{\sum_{i\in A}\omega_i \geq w}{min_{A\in\mathcal{P}(\{1,2,\ldots,n\})}}([\{p_i | i \in A\}]) \qquad (2.1)$$

Remark 2.4.1 In case $\omega_1 = \cdots = \omega_n = 1$ and $w = k$, a sequence $p_1, \ldots, p_n$ is an (ω, w, n)-Mignotte sequence if and only if $p_1, \ldots, p_n$ is a generalized (k, n)-Mignotte sequence in sense of Definition 2.1.3.

In the same case, an ordered sequence $p_1, \ldots, p_n$ with pairwise coprime elements is an (ω, w, n)-Mignotte sequence if and only if $p_1, \ldots, p_n$ is an (k, n)-Mignotte sequence in sense of Definition 2.1.2.

For arbitrary weights and thresholds, an (ω, w, n)-Mignotte sequence can be constructed as follows. Let $p'_1, \ldots, p'_N$ be a generalized (w, N)-Mignotte sequence, where $N = \sum_{i=1}^{n} \omega_i$ and define $p_i = [\{p'_j \mid j \in P_i\}]$, for all $1 \leq i \leq n$, where $\{P_1, \ldots, P_n\}$ is an arbitrary partition of the set $\{1, 2, \ldots, N\}$ such that $|P_i| = \omega_i$, for all $1 \leq i \leq n$. We obtain that

$$\begin{aligned} \underset{\substack{A\in\mathcal{P}(\{1,2,\ldots,n\})\\ \sum_{i\in A}\omega_i\leq w-1}}{max}([\{p_i|i \in A\}]) &= \underset{\substack{A\in\mathcal{P}(\{1,2,\ldots,n\})\\ \sum_{i\in A}\omega_i\leq w-1}}{max}([\{[\{p'_j \mid j \in P_i\}] \mid i \in A\}]) \\ &= \underset{\substack{A\in\mathcal{P}(\{1,2,\ldots,n\})\\ \sum_{i\in A}\omega_i\leq w-1}}{max}([\{p'_j \mid j \in \cup_{i\in A} P_i\}]). \end{aligned}$$

Moreover, for any set $A \in \mathcal{P}(\{1, 2, \ldots, n\})$ with $\sum_{i\in A} \omega_i \leq w - 1$ we obtain that $|\{p'_j \mid j \in \cup_{i\in A} P_i\}| = \sum_{i\in A} |P_i| = \sum_{i\in A} \omega_i \leq w - 1$ and, thus,

$$\underset{\substack{A\in\mathcal{P}(\{1,2,\ldots,n\})\\ \sum_{i\in A}\omega_i\leq w-1}}{max}([\{p_i|i \in A\}]) \leq max_{1\leq i_1<\cdots<i_{w-1}\leq N}([\{p'_{i_1}, \ldots, p'_{i_{w-1}}\}]) \tag{2.2}$$

By the same reason, we obtain that

$$min_{1\leq i_1<\cdots<i_w\leq N}([\{p'_{i_1}, \ldots, p'_{i_w}\}]) \leq \underset{\substack{A\in\mathcal{P}(\{1,2,\ldots,n\})\\ \sum_{i\in A}\omega_i\geq w}}{min}([\{p_i|i \in A\}]). \tag{2.3}$$

Using relations (2.2), (2.3), and the fact that $p'_1, \ldots, p'_N$ is a generalized (w, N)-Mignotte sequence, we obtain that the relation (2.1) holds true, which implies that the sequence $p_1, \ldots, p_n$ is indeed an (ω, w, n)-Mignotte sequence.

Example 2.4.2 illustrates this construction.

Example 2.4.2 (with artificially small parameters)
Let us consider $n = 4$, the weights $\omega_1 = \omega_2 = 1$, $\omega_3 = \omega_4 = 2$, and the threshold $w = 3$. We obtain $N = 6$. The sequence 7, 11, 13, 17, 19, 23 is a generalized $(3, 6)$-Mignotte sequence and, if we consider the partition $\{\{6\}, \{5\}, \{1, 4\}, \{2, 3\}\}$ of the set $\{1, 2, 3, 4, 5, 6\}$, we obtain that the sequence 23, 19, [7, 17], [11, 13] is an $((1, 1, 2, 2), 3, 4)$-Mignotte sequence.

These sequences can be used for constructing weighted threshold secret sharing schemes in an obvious way. More exactly, having an (ω, w, n)-Mignotte sequence $p_1, \ldots, p_n$, we may construct an (ω, w, n)-weighted threshold secret sharing scheme as follows:

- the secret S is an arbitrary integer in the interval $[\beta + 1, \alpha - 1]$, where $\alpha = \underset{\substack{A\in\mathcal{P}(\{1,2,\ldots,n\})\\ \sum_{i\in A}\omega_i\geq w}}{min}([\{p_i|i \in A\}])$ and
$\beta = \underset{\substack{A\in\mathcal{P}(\{1,2,\ldots,n\})\\ \sum_{i\in A}\omega_i\leq w-1}}{max}([\{p_i|i \in A\}])$;

- the shares are chosen as $I_i = S \texttt{ mod } p_i$, for all $1 \leq i \leq n$.

Having a set of shares $\{I_i \mid i \in A\}$, where A satisfies $\sum_{i \in A} \omega_i \geq w$, the secret S can be obtained as the unique solution modulo $[\{p_i|i \in A\}]$ of the system of equations

$$\left\{ \begin{array}{lll} x & \equiv & I_i \; mod \; p_i, \quad i \in A. \end{array} \right.$$

Indeed, the secret S is the unique solution modulo $[\{p_i|i \in A\}]$ of the above system of equations because S is an integer solution of the system by the choice of the shares $I_1, \ldots, I_n$ and, moreover, $S \in \mathbf{Z}_{[\{p_i|i \in A\}]}$, by the choice of the secret S $(S < \alpha)$.

Having a set of shares $\{I_i \mid i \in A\}$, where A satisfies $\sum_{i \in A} \omega_i \leq w - 1$, the only information we can obtain by finding the unique solution x_0 in $\mathbf{Z}_{[\{p_i|i \in A\}]}$ of the system of equations

$$\left\{ \begin{array}{lll} x & \equiv & I_i \; mod \; p_i, \quad i \in A \end{array} \right.$$

is that $S \equiv x_0 \; mod \; [\{p_i|i \in A\}]$. Indeed, the secret S is not the unique solution modulo $[\{p_i|i \in A\}]$ of the above system of equations because $S \notin \mathbf{Z}_{[\{p_i|i \in A\}]}$, by the choice of the secret S $(S > \beta)$. By choosing (ω, w, n)-Mignotte sequences with a large factor $\frac{\alpha - \beta}{\beta}$, the problem of finding the secret S, knowing that S is in the interval $[\beta + 1, \alpha - 1]$ and $S \equiv x_0 \; mod \; [\{p_i|i \in A\}]$, for some unauthorized access set A, is intractable.

Example 2.4.3 (with artificially small parameters)
Consider $n = 4$, the weights $\omega_1 = \omega_2 = 1$, $\omega_3 = \omega_4 = 2$, and the threshold $w = 3$. The corresponding weighted threshold access structure is given by $\mathcal{A}_{min} = \{\{1,\ 3\},\ \{1,\ 4\},\ \{2,\ 3\},\ \{2,\ 4\},\ \{3,\ 4\}\}$ and $\overline{\mathcal{A}}_{max} = \{\{1,\ 2\},\ \{3\},\ \{4\}\}$. According to Example 2.4.2, the sequence 23, 19, 119, 143 is an $((1,1,2,2),3,4)$-Mignotte sequence. We obtain that
$\alpha = min([23,\ 119],\ [23,\ 143],\ [19,\ 119],\ [19,\ 143],\ [119,\ 143]) = 2261$ and
$\beta = max([23, 19],\ 119,\ 143) = 437$.

An $((1,1,2,2),3,4)$-weighted threshold secret sharing scheme is described next.

- the secret S is chosen in the interval $[438,\ 2260]$, for example, $S = 601$;
- the shares are $I_1 = 601 \texttt{ mod } 23 = 3$, $I_2 = 601 \texttt{ mod } 19 = 12$, $I_3 = 601 \texttt{ mod } 119 = 6$, and $I_4 = 601 \texttt{ mod } 143 = 29$.

Having the shares $I_1 = 3$ and $I_3 = 6$, the secret S can be obtained as the unique solution in $\mathbf{Z}_{2737}$ of the system of equations

$$\left\{ \begin{array}{lll} x & \equiv & 3 \; mod \; 23 \\ x & \equiv & 6 \; mod \; 119 \end{array} \right.$$

that is indeed 601.

But having the shares $I_1 = 3$ and $I_2 = 12$, the secret S cannot be obtained as the unique solution in $\mathbf{Z}_{437}$ of the system of equations

$$\begin{cases} x & \equiv & 3 \; mod \; 23 \\ x & \equiv & 12 \; mod \; 19 \end{cases}$$

because this solution is 164.

We will describe next how to construct weighted threshold Asmuth-Bloom sequences ([85]). Let $n \geq 2$, $\omega = (\omega_1, \ldots, \omega_n)$ a sequence of weights, and w a threshold. An (ω, w, n)-*Asmuth-Bloom sequence* is a sequence of positive integers $p_0, p_1, \ldots, p_n$ such that

$$p_0 \cdot max_{\substack{B\in\mathcal{P}(\{1,2,\ldots,n\}) \\ \sum_{i\in B}\omega_i \leq w-1}} ([\{p_i | i \in B\}]) < min_{\substack{A\in\mathcal{P}(\{1,2,\ldots,n\}) \\ \sum_{i\in A}\omega_i \geq w}}([\{p_i | i \in A\}]).$$

An (ω, w, n)-Asmuth-Bloom sequence can be constructed as follows. Let $p'_0, p'_1, \ldots, p'_N$ be a generalized threshold (w, N)-Asmuth-Bloom sequence, where $N = \sum_{i=1}^{n} \omega_i$ and define $p_0 = p'_0$ and $p_i = [\{p'_j \mid j \in P_i\}]$, for all $1 \leq i \leq n$, where $\{P_1, \ldots, P_n\}$ is an arbitrary partition of the set $\{1, 2, \ldots, N\}$ such that $|P_i| = \omega_i$, for all $1 \leq i \leq n$. It is easy to prove that the sequence $p_0, p_1, \ldots, p_n$ is indeed an (ω, w, n)-Asmuth-Bloom sequence.

Example 2.4.4 illustrates this construction.

Example 2.4.4 (with artificially small parameters) Let us consider $n = 4$, the weights $\omega_1 = \omega_2 = 1$, $\omega_3 = \omega_4 = 2$, and the threshold $w = 3$. We obtain $N = 6$. The sequence 5, 17, 19, 23, 29, 31, 37 is a $(3, 6)$-Asmuth-Bloom sequence and, if we consider the partition $\{\{6\},\ \{5\},\ \{1,\ 4\},\ \{2,\ 3\}\}$ of the set $\{1,\ 2,\ 3,\ 4,\ 5,\ 6\}$, we obtain that the sequence 5, 37, 31, $17 \cdot 29$, $19 \cdot 23$ is an $((1, 1, 2, 2), 3, 4)$-Asmuth-Bloom sequence.

2.5 Hierarchical Secret Sharing Schemes

In case of *hierarchical* (or *multilevel*) secret sharing, the set of users is partitioned into some levels $L_1, L_2, \ldots, L_m$, corresponding to their hierarchy, L_1 being the highest level of hierarchy and L_m the lowest one. A level-threshold k_j is assigned to the j^{th} level, for all $1 \leq j \leq m$. We can naturally assume that $k_1 < k_2 < \cdots < k_m$. The secret can be recovered if and only if there is a level (called *initialization level*) such that the number of participants from this level or higher levels is greater than or equal to the initialization level threshold.

The hierarchical access structures can be introduced as follows.

Definition 2.5.1 Let $\mathcal{L} = \{L_1, L_2, \ldots, L_m\}$ be a partition of $\{1, 2, \ldots, n\}$ and let us consider a sequence of *level thresholds* $\mathcal{K} = (k_1, k_2, \ldots, k_m)$, where

$1 \leq k_j \leq |L_j|$, for all $1 \leq j \leq m$, and $k_1 < k_2 < \cdots < k_m$. The $(\mathcal{L}, \mathcal{K})$-*multilevel access structure* is given by

$$\mathcal{A} = \{A \in \mathcal{P}(\{1, 2, \ldots, n\}) \mid (\exists j = \overline{1, m})(|A \cap \cup_{l=1}^{j} L_l| \geq k_j)\}.$$

In this case, an $\mathcal{A}$-secret sharing scheme will be referred to as a $(\mathcal{L}, \mathcal{K})$-*multilevel secret sharing scheme.*

An (k, n)-threshold secret sharing scheme is nothing else than a $(\mathcal{L}, \mathcal{K})$-multilevel secret sharing scheme with $\mathcal{L} = \{\{1, 2, \ldots, n\}\}$ $(m = 1)$ and $\mathcal{K} = (k)$.

Let L_j^+ denote the set of users from level j or higher, i.e., $L_j^+ = \cup_{l=1}^{j} L_l$. The access structure can be expressed as

$$\mathcal{A} = \cup_{j=1}^{m} \{A \subseteq L_j^+ \mid |A| \geq k_j\}.$$

Thus, one obvious way of realizing multilevel secret sharing consists in using an arbitrary $(k_j, |L_j^+|)$-threshold secret sharing scheme for any $j = \overline{1, m}$. The main drawback of this idea is that any participant from level j will receive $m - j + 1$ sub-shares, for all $j = \overline{1, m}$, and, thus, the participants from the highest level will receive significantly large shares.

Multilevel secret sharing has been considered for the first time by Simmons [137], and then by Brickell [20], but their solutions are far from being practical.

Ghodosi, Pieprzyk, and Safavi-Naini have proposed an ideal scheme in [70]. Their method is based on the notion of *extension* of a threshold secret sharing scheme.

Definition 2.5.2 Let $(split_1, combine_1)$ be an (k_1, n_1)-threshold secret sharing scheme and $(split_2, combine_2)$ be an (k_2, n_2)-threshold secret sharing scheme, with the same secrets set. We say that $(split_2, combine_2)$ is an *extension* of $(split_1, combine_1)$ if for any secret S the relations (1) and (2) imply (3), where

$$\begin{array}{llll}
(1) & S & \leftrightarrow^{(split_1, combine_1)}_{(k_1, n_1)} & I_1^1, \ldots, I_{n_1}^1 \\
(2) & S & \leftrightarrow^{(split_2, combine_2)}_{(k_2, n_2)} & I_1^2, \ldots, I_{n_2}^2 \\
\hline
(3) & S & \leftrightarrow^{(split, combine)}_{(k_2, n_1 + n_2)} & I_1^1, \ldots, I_{n_1}^1, I_1^2, \ldots, I_{n_2}^2,
\end{array}$$

for some $(split, combine)$.

We start with an $(k_1, |L_1^+|)$-threshold secret sharing scheme, then we extend it to an $(k_2, |L_2^+|)$-threshold secret sharing scheme, and so on, until the final $(k_m, |L_m^+|)$-threshold secret sharing scheme is obtained as the extension of the previous scheme. Thus, each user will receive a single share.

We present only the extension from the $(k_1, |L_1^+|)$-scheme to the $(k_2, |L_2^+|)$-scheme, the rest of the extensions being performed in a similar manner. For the initial scheme, the dealer chooses a polynomial $P_1(x) = S + a_{1,1}x + \cdots + a_{1,l_1}x^{l_1}$ over $\mathbf{Z}_q$, where $l_1 = k_1 - 1$ and distributes the shares $I_i = P_1(i)$ to the users $i \in L_1$. In order to construct the extension to the $(k_2, |L_2^+|)$-scheme, a polynomial $P_2(x) = S + a_{2,1}x + \cdots + a_{2,l_2}x^{l_2}$ over $\mathbf{Z}_q$ must be chosen such that any group of k_2 participants from $L_1 \cup L_2$ can recover the secret but any smaller group gets no information about the secret. By imposing $P_1(i) = P_2(i) = I_i$, for any $i \in L_1$, the following set of $2|L_1|$ equations are known:

$$\text{From } P_1(x) \left\{ \begin{array}{lcl} S + a_{1,1}i + \cdots + a_{1,l_1}i^{l_1} & = & I_i, \quad i \in L_1, \end{array} \right.$$

$$\text{From } P_2(x) \left\{ \begin{array}{lcl} S + a_{2,1}i + \cdots + a_{2,l_2}i^{l_2} & = & I_i, \quad i \in L_1. \end{array} \right.$$

In the reconstruction phase, any group $A \subseteq L_2$, $|A| = k_2$, has to recover the secret. These users bring the following k_2 equations

$$\text{From } P_2(x) \left\{ \begin{array}{lcl} S + a_{2,1}i + \cdots + a_{2,l_2}i^{l_2} & = & I_i, \quad i \in A, \ A \subseteq L_2, \ |A| = k_2. \end{array} \right.$$

We obtain a set of $2|L_1| + k_2$ equations with $1 + l_1 + l_2 + |L_1|$ unknowns (the secret S, $a_{1,1}, \ldots, a_{1,l_1}$, $a_{2,1}, \ldots, a_{2,l_2}$, and the shares I_i, $i \in L_1$). If we want that the obtained system of equations admits a unique solution, then the relation $2|L_1| + k_2 = 1 + l_1 + l_2 + |L_1|$ must hold. Thus, the dealer can choose a proper value for l_2, the degree of $P_2(x)$.

In the case that a group $A \subseteq L_1 \cup L_2$, $|A| = k_2$, want to recover the secret S by pooling together their shares, this is possible because the number of unknown shares in the new obtained system is still $|L_1|$ - indeed, there are $|L_1| - |L_1 \cap A|$ unknown shares corresponding to the participants from level L_1 and $k_2 - |L_2 \cap A|$ unknown shares corresponding to the participants from level L_2 (and $|L_1 \cap A| + |L_2 \cap A| = k_2$).

The dealer chooses a polynomial $P_2(x)$ of degree l_2 such that $P_2(0) = S$ and $P_2(i) = P_1(i)$, for all $i \in L_1$, and sends the shares $I_i = P_2(i)$ to the users $i \in L_2$.

Tassa [144] has remarked that there are situations which impose more rigid restrictions. More exactly, even participants from higher levels can be replaced by ones from lower levels, a minimal number of participants from higher levels may be required for the reconstruction of the secret. Tassa has presented the case of authorizing electronic fund transfers that requires the presence of at least one vice-president or manager department. The classical hierarchical schemes cannot be used in this case because in these schemes the secret can be reconstructed by any relatively large group of users from lower levels. Tassa has introduced and analyzed a new class of access structures, obtained from the classical hierarchical access structures by replacing the existential quantifier with the universal quantifier.

Definition 2.5.3 Let $\mathcal{L} = \{L_1, L_2, \ldots, L_m\}$ be a partition of $\{1, 2, \ldots, n\}$ and let us consider a sequence of *level thresholds* $\mathcal{K} = (k_1, k_2, \ldots, k_m)$, where $1 \leq k_j \leq |L_j|$, for all $1 \leq j \leq m$. The $(\mathcal{L}, \mathcal{K})$*-strong multilevel access structure* is given by

$$\mathcal{A} = \{A \in \mathcal{P}(\{1, 2, \ldots, n\}) \mid (\forall j = \overline{1, m})(|A \cap \cup_{l=1}^{j} L_l| \geq k_j)\}.$$

In this case, an $\mathcal{A}$-secret sharing scheme will be referred to as a $(\mathcal{L}, \mathcal{K})$*-strong multilevel secret sharing scheme.*

Tassa's method for strong multilevel secret sharing is based on Birkhoff interpolation. Let $X = \{x_1, \ldots, x_k\}$ be a set of elements of a finite field $\mathbf{GF}_q$, $E = (e_{i,j})_{1 \leq i \leq k, 0 \leq j \leq l}$ be a matrix with elements 0 or 1, and $C = \{c_{i,j} | (i,j) \in I(E)\}$ be a set of elements of $\mathbf{GF}_q$, where $I(E) = \{(i,j) \in \{1, \ldots, k\} \times \{0, \ldots, l\} | e_{i,j} = 1\}$. Birkhoff interpolation problem for the instance (X, E, C) consists in finding a polynomial $P(x)$ of degree at most $|I(E)| - 1$ over $\mathbf{GF}_q$ such that[4] $P^{(j)}(x_i) = c_{i,j}$, for all $(i,j) \in I(E)$. In case that the matrix E has only one column, corresponding to derivates of order zero, we obtain Lagrange interpolation problem. As opposed to Lagrange interpolation problem, Birkhoff interpolation problem may have no solution or the solution may be not unique and, thus, some precautions must be taken.

In Tassa's strong multilevel secret sharing scheme, as in Shamir's threshold scheme, the secret is the free coefficient of some polynomial $P(x)$. The share I_i will be generated as $P^{(j)}(x_i)$ for a derivate order j that depends on the position of the user i in hierarchy. The main idea is that the users from levels with lower index will receive shares with lower derivate orders which contain more information than the higher ones. The scheme is described next.

- The dealer generates the polynomial $P(x)$ of degree at most $k_m - 1$ over $\mathbf{GF}_q$, $P(x) = S + a_1 x^1 + \cdots + a_{k_m - 1} x^{k_m - 1}$;
- An element $\mathtt{i} \in \mathbf{GF}_q$ is assigned to the user i, for all $1 \leq i \leq n$;
- For any level j, each user i from the j^{th} level will receive the share $I_i = P^{(k_{j-1})}(\mathtt{i})$, where $k_0 = 0$;
- In the reconstruction phase, Birkhoff interpolation will lead to the secret (the reader is referred to [144] for some methods of assigning

[4] For a polynomial $P(x)$ over $\mathbf{GF}_q$, $P(x) = \sum_{i=0}^{l} a_i x^i$, its first derivate, denoted by $P'(x)$, is defined as $P'(x) = \sum_{i=0}^{l} i a_i x^{i-1}$. We can further define $P^{(0)}(x) = P(x)$ and $P^{(i+1)}(x) = (P^{(i)}(x))'$, for all $i \geq 0$.

field elements to users such that Birkhoff interpolation problem admits a unique solution).

The next example illustrates the presented scheme.

Example 2.5.1 ([144])
Let $m = 3$, $k_1 = 2$, $k_2 = 4$, and $k_3 = 7$. In this case, a set $A \subseteq \{1, 2, \dots, n\}$ is authorized if and only if A has at least 7 members, of whom at least 4 are from the second level or higher, of whom at least 2 are from the first level. The dealer generates some polynomial $P(x)$ of degree at most 6 over $\mathbf{GF}_q$ such that $P(0) = S$. Any user i from the first level will receive $P(\mathtt{i})$, any user i from the second level will receive $P^{(2)}(\mathtt{i})$, and, finally, any user i from the third level will receive $P^{(4)}(\mathtt{i})$.

Tassa has remarked that Birkhoff interpolation can also be used for realizing ordinary multilevel secret sharing, by performing two small modifications to the previous scheme, namely, the secret will be chosen as the coefficient of the highest power and the users from the levels with lower index will receive shares with higher derivate orders. The scheme is described next.

- The dealer generates the polynomial $P(x)$ of degree at most $k_m - 1$ over $\mathbf{GF}_q$, $P(x) = a_0 + a_1 x^1 + \cdots + S x^{k_m - 1}$;
- An element $\mathtt{i} \in \mathbf{GF}_q$ is assigned to the user i, for all $1 \leq i \leq n$;
- For any level j, each user i from the j^{th} level will receive the share $I_i = P^{(k_m - k_j)}(\mathtt{i})$;
- In the reconstruction phase, Birkhoff interpolation will lead to the secret.

The next example illustrates the presented scheme.

Example 2.5.2 ([144])
Let $m = 3$, $k_1 = 2$, $k_2 = 4$, and $k_3 = 7$. In this case, a set $A \subseteq \{1, 2, \dots, n\}$ is authorized if and only if A has at least 7 members, or at least 4 members from the second level or higher, or at least 2 members from the first level. The dealer generates some polynomial $P(x)$ of degree at most 6 over some field $\mathbf{GF}_q$, $P(x) = \sum_{i=0}^{6} a_i x^i$ such that $a_6 = S$. Any user i from the first level will receive $P^{(5)}(\mathtt{i})$, any user i from the second level will receive $P^{(3)}(\mathtt{i})$, and, finally, any user i from the third level will receive $P(\mathtt{i})$.

Both schemes are ideal.

2.6 Compartmented Secret Sharing Schemes

In case of *compartmented* secret sharing, the set of users is partitioned into compartments $C_1, C_2, \ldots, C_m$. Besides a global threshold k, a threshold k_j is assigned to the j^{th} compartment, for all $1 \leq j \leq m$. The secret can be recovered if and only if the number of participants from any compartment is greater than or equal to the corresponding compartment threshold, and the total number of participants is greater than or equal to the global threshold.

The compartmented access structures can be introduced as follows.

Definition 2.6.1 Let $\mathcal{C} = \{C_1, C_2, \ldots, C_m\}$ be a partition of $\{1, 2, \ldots, n\}$ and let us consider a sequence of *compartment thresholds* $\mathcal{K} = (k_1, k_2, \ldots, k_m)$, where $1 \leq k_j \leq |C_j|$, for all $1 \leq j \leq m$, and a *global threshold* k, $\sum_{j=1}^{m} k_j \leq k \leq n$. The $(\mathcal{C}, \mathcal{K}, k)$-*compartmented access structure* is given by

$$\mathcal{A} = \{A \in \mathcal{P}(\{1, 2, \ldots, n\}) \mid (|A| \geq k) \wedge (\forall j = \overline{1, m})(|A \cap C_j| \geq k_j)\}.$$

In this case, an $\mathcal{A}$-secret sharing scheme will be referred to as a $(\mathcal{C}, \mathcal{K}, k)$-*compartmented secret sharing scheme.*

An (k, n)-threshold secret sharing scheme is nothing else than a $(\mathcal{C}, \mathcal{K})$-compartmented secret sharing scheme with $\mathcal{C} = \{\{1, 2, \ldots, n\}\}$ $(m = 1)$ and $\mathcal{K} = (k, k)$.

The compartmented secret sharing has been discussed for the first time by Simmons in [137]. He has presented the example of an official action that requires that at least two U.S. members and at least two U.S.S.R. members be simultaneously present for its initiation. Brickell [20] has proposed an elegant solution for the case $k = \sum_{j=1}^{m} k_j$ by expressing the secret S as a combination of some compartment secrets $s_1, \ldots, s_m$ and using an $(k_j, |C_j|)$-threshold secret sharing scheme for obtaining the shares $\{I_i | i \in C_j\}$ corresponding to the compartment secret s_j, for all $1 \leq j \leq m$. In the reconstruction phase, if the number of participants from the j^{th} compartment is greater than or equal to the k_j, for all $1 \leq j \leq m$, then all compartment secrets can be recovered and, thus, the secret S can be obtained (remark that in this case the access structure can be simplified to $\{A \in \mathcal{P}(\{1, 2, \ldots, n\}) \mid (\forall j = \overline{1, m})(|A \cap C_j| \geq k_j)\}$).

Ghodosi, Pieprzyk, and Safavi-Naini have proposed an ideal scheme for the general case in [70]. We will present it for the case $k > \sum_{j=1}^{m} k_j$.

- We start with a unanimous consent secret sharing scheme of rank m and we construct the shares $s_1, \ldots, s_m$ corresponding to a secret S;

- For each $1 \leq j \leq m$, we will consider the polynomial $P_j(x) = s_j + a_{j,1}x + \cdots + a_{j,k_j-1}x^{k_j-1} + b_1 x^{k_j} + \cdots + b_l x^{k_j+l-1}$ over a finite field, where $l = k - \sum_{j=1}^{m} k_j$. The shares are generated as $I_i = P_j(i)$, for all $1 \leq j \leq m$ and for all $i \in C_j$;

- Let us consider an authorized group A. Thus, $|A| \geq k$ and $|A \cap C_j| \geq k_j$, for all $1 \leq j \leq m$. The system of equations

  $$\{s_j + a_{j,1} i + \cdots + a_{j,k_j-1} i^{k_j-1} + b_1 i^{k_j} + \cdots + b_l i^{k_j+l-1} = I_i,\ 1 \leq j \leq m,\ i \in A \cap C_j,$$

  has the unknowns $s_j, a_{j,1}, \ldots, a_{j,k_j-1}$ corresponding to the j^{th} compartment that appear in at least k_j equations, $1 \leq j \leq m$, and $b_1, \ldots, b_l$, that appear in all equations, and, thus, we obtain a system with $\sum_{j=1}^{m} k_j + l = k$ unknowns and with at least k equations. This system has a unique solution and solving it leads us to $s_1, \ldots, s_m$, and, finally, to S;

- Let us consider an unauthorized group A. There are two possibilities:

  - $|A| < k$ - in this case, we obtain a system of equations with k unknowns but the number of equations is less than k - such a system has not a unique solution;
  - There is $1 \leq j \leq m$ such that $|A \cap C_j| < k_j$ - the unknowns $s_j, a_{j,1}, \ldots, a_{j,k_j-1}$ appear only in $|A \cap C_j|$ equations, and, thus, the element s_j cannot be obtained, and neither can the secret S.

Compartmented Secret Sharing based on the Chinese Remainder Theorem

In [85] we have extended Brickell's construction to the case $\sum_{j=1}^{m} k_j < k$ as follows.

- The secret is chosen as $S = s_0 + s_1 + \cdots + s_m$, where $s_0, s_1, \ldots, s_m$ are positive integers;

- The shares are chosen as $I_i = (g_i, c_i)$, for any $1 \leq i \leq n$, where

  - $g_1, \ldots, g_n$ are the shares corresponding to the secret s_0 with respect to an arbitrary (k, n)-threshold secret sharing scheme - these elements will be referred to as the *global* components of the shares;
  - for every $1 \leq j \leq m$, $\{c_i | i \in C_j\}$ are the shares corresponding to the secret s_j with respect to an arbitrary $(k_j, |C_j|)$-threshold secret sharing scheme - these elements will be referred to as the *compartment* components of the shares.

Remark 2.6.1 (correctness)

Let A be an authorized access group. Thus, $|A| \geq k$ and, for all $j = \overline{1, m}$, $|A \cap C_j| \geq k_j$. Having at least k of the shares $g_1, \ldots, g_n$, the value s_0 can be obtained. Then, for any $j = \overline{1, m}$, having at least k_j of the shares $\{c_i | i \in C_j\}$, the value s_j can be obtained, and finally, the secret S can be obtained as $S = s_0 + s_1 + \cdots + s_m$.

Remark 2.6.2 (security)

Let A be an unauthorized access group. There are two possibilities:

- $|A| < k$ - in this case, the value s_0 cannot be determined;

- There is an compartment j such that $|A \cap C_j| < k_j$ - in this case the value s_j cannot be determined.

In both cases, the secret S cannot be reconstructed.

Using perfect threshold secret sharing schemes as building blocks can lead to large shares. We propose using the threshold secret sharing schemes based on the Chinese remainder theorem in order to decrease the size of shares, maintaining, in the same time, a reasonable level of security. For simplicity, we will use only Mignotte's scheme, but we must mention that the technique can be also applied to Asmuth-Bloom scheme.

Let $k_0 = k$ and $C_0 = \{1, 2, \ldots, n\}$. For any $0 \leq j \leq m$, we will generate and broadcast a generalized $(k_j, |C_j|)$-Mignotte sequence $(p_{j,i} | i \in C_j)$. Let $\beta_j = max_{i_1,\ldots,i_{k_j-1} \in C_j}([p_{j,i_1}, \ldots, p_{j,i_{k_j-1}}])$, $\alpha_j = min_{i_1,\ldots,i_{k_j} \in C_j}([p_{j,i_1}, \ldots, p_{j,i_{k_j}}])$, for $0 \leq j \leq m$. The secret S is chosen as $S = \sum_{j=0}^{m} s_j$, where $\beta_j < s_j < \alpha_j$. We may use a generalized Mignotte sequence twice in case that $k_j = k_l$ and $|C_j| = |C_l|$, for some $1 \leq j < l \leq m$. The components of the shares will be chosen as

$$g_i = s_0 \bmod p_{0,i},$$

$$c_i = s_{c(i)} \bmod p_{c(i),i},$$

where $c(i)$ is the unique element j, $1 \leq j \leq m$, such that $i \in C_j$, for all $1 \leq i \leq n$. Example 2.6.1 illustrates this scheme.

Example 2.6.1 (with artificially small parameters)
Let us consider $n = 6$, $\mathcal{C} = \{\{1, 2, 3\}, \{4, 5, 6\}\}$, the compartment thresholds $k_1 = 2$, $k_2 = 2$ and the global threshold $k_0 = 5$. The sequence 5, 7, 11, 13, 17, 19 is a $(5, 6)$-Mignotte sequence, with $\alpha_0 = 85085$ and $\beta_0 = 46189$, and the sequence 7, 11, 13 is a $(2, 3)$-Mignotte sequence with $\alpha_1 = \alpha_2 = 77$ and $\beta_1 = \beta_2 = 13$. We choose $s_0 = 50000$, $s_1 = 30$, and $s_2 = 40$. The secret will be $S = 50070$ and the shares $I_1 = (0, 2)$, $I_2 = (6, 8)$, $I_3 = (5, 4)$, $I_4 = (2, 5)$, $I_5 = (3, 7)$, and $I_6 = (11, 1)$.

Having the shares $I_1 = (0, 2)$, $I_2 = (6, 8)$, $I_4 = (2, 5)$, $I_5 = (3, 7)$, and

$I_6 = (11, 1)$, we solve the systems

$$\left\{\begin{array}{lcl} x & \equiv & 0 \; mod \; 5 \\ x & \equiv & 6 \; mod \; 7 \\ x & \equiv & 2 \; mod \; 13 \\ x & \equiv & 3 \; mod \; 17 \\ x & \equiv & 11 \; mod \; 19 \end{array}\right. ,$$

$$\left\{\begin{array}{lcl} x & \equiv & 2 \; mod \; 7 \\ x & \equiv & 8 \; mod \; 11 \end{array}\right. ,$$

$$\left\{\begin{array}{lcl} x & \equiv & 7 \; mod \; 11 \\ x & \equiv & 1 \; mod \; 13 \end{array}\right.$$

and obtain, respectively, $s_0 = 50000$, $s_1 = 30$, $s_2 = 40$, and finally $S = 50070$.

Let analyze the security of the scheme. Let B be an unauthorized group and consider $\Delta_B = \{j \in \{0, 1, \ldots, m\} |\; |A \cap C_j| < k_j\}$. The information obtained from the shares corresponding to B leads to a set of possible vectors $(s_0, s_1, \ldots, s_m)$ of cardinality at least $\prod_{j \in \Delta_B} \frac{\alpha_j - \beta_j}{\beta_j}$. The generalized Mignotte sequences can be thus generated accordingly to the unauthorized access structure in order to obtain a suitable security level.

Although the shares of our scheme have two components, by using generalized Mignotte's scheme as a building block, the sizes of shares can be smaller than the size of the secret. Further improvements can be obtained by choosing the Mignotte sequences and the values $s_0, s_1, \ldots, s_m$ such that the global components of some shares coincide with the corresponding compartment ones, i.e., $g_i = c_i$, for some $i \in \{1, 2, \ldots, n\}$. In this case we can define the share I_i as $I_i = g_i = c_i$. For this, we can generate first $s_1, \ldots, s_m$ and $c_1, \ldots, c_n$ and determining s_0 by solving the system of equations

$$\left\{\begin{array}{lcl} x & \equiv & c_1 \; mod \; p_{0,1} \\ & \vdots & \\ x & \equiv & c_{k_0} \; mod \; p_{0,k_0} \end{array}\right. .$$

We will choose $I_i = g_i = c_i$, for all $1 \le i \le k_0$, $g_i = s_0 \bmod p_{0,i}$ and $I_i = (g_i, c_i)$, for all $k_0 + 1 \le i \le n$. Further improvements can be obtained in case that $s_0 \bmod p_{0,i} = s_{c(i)} \bmod p_{c(i),i}$, for some $k_0 + 1 \le i \le n$.

Example 2.6.2 illustrates the reduction of the shares.

Example 2.6.2 (with artificially small parameters)
Let us reconsider Example 2.6.1. We choose $s_1 = 30$ and $s_2 = 40$. We

obtain $c_1 = 2$, $c_2 = 8$, $c_3 = 4$, $c_4 = 5$, $c_5 = 7$, and $c_6 = 1$. The system

$$\begin{cases} x & \equiv & 2 \ mod\ 5 \\ x & \equiv & 8 \ mod\ 7 \\ x & \equiv & 4 \ mod\ 11 \\ x & \equiv & 5 \ mod\ 13 \\ x & \equiv & 7 \ mod\ 17 \end{cases}$$

has the solution $s_0 = 32817$. The secret will be $S = 32887$ and the shares $I_1 = 2$, $I_2 = 8$, $I_3 = 4$, $I_4 = 5$, $I_5 = 7$, and $I_6 = (4, 1)$.

Thus, in case that k_0 is close to n, real improvements related to the size of shares can be made. However, every compression of a share, i.e., any equalization of form $g_i = c_i$, can affect the security with a factor of $\frac{\alpha_{c(i)} - \beta_{c(i)}}{\beta_{c(i)}}$. Thus, depending of the intended application, a compromise between the size of the shares and the level of security must be made.

Tassa and Dyn [145] have recently introduced and analyzed a new class of access structures, namely the *compartmented access structures with upper bounds* (the previous compartmented access structures being renamed as *compartmented access structures with lower bounds*).

Definition 2.6.2 Let $\mathcal{C} = \{C_1, C_2, \ldots, C_m\}$ be a partition of $\{1, 2, \ldots, n\}$ and let us consider a sequence of *compartment thresholds* $\mathcal{K} = (k_1, k_2, \ldots, k_m)$ and a *global threshold* k such that $\sum_{j=1}^{m} k_j \geq k$. The $(\mathcal{C}, \mathcal{K}, k)$-*compartmented access structure with upper bounds* is given by

$$\mathcal{A} = \{B \in \mathcal{P}(\{1, 2, \ldots, n\}) \mid (\exists A \subseteq B)((|A| = k) \wedge (\forall j = \overline{1, m})(|A \cap C_j| \leq k_j))\}.$$

Tassa and Dyn have proposed a solution of realizing such access structures using the bivariate interpolation (the reader is referred to [145] for more details).

2.7 General Secret Sharing Schemes

2.7.1 Cumulative Secret Sharing Schemes

The notion of *cumulative map* has been introduced by Simmons, Jackson, and Martin in [138], inspired by a paper of Ito, Saito, and Nishizeki [90].

Definition 2.7.1 Let $\mathcal{A}$ be an access structure over $\{1, 2, \ldots, n\}$. An $\mathcal{A}$-*cumulative map* is a pair (m, f) where m is a positive integer and f is a function from $\{1, 2, \ldots, n\}$ to $\mathcal{P}(\{1, 2, \ldots, m\})$ such that

$$(\forall A \in \mathcal{P}(\{1, 2, \ldots, n\}))(\cup_{i \in A} f(i) = \{1, 2, \ldots, m\} \Leftrightarrow A \in \mathcal{A}).$$

An $\mathcal{A}$-cumulative map (m, f) can be combined with a unanimous consent scheme of rank m in order to construct an $\mathcal{A}$-secret sharing scheme as follows. A unanimous consent secret sharing scheme of rank m is used for constructing the shares $s_1, \ldots, s_m$ corresponding to a secret S. The shares corresponding to the access structure $\mathcal{A}$ are chosen as

$$I_i = \{s_j | j \in f(i)\},$$

for all $1 \leq i \leq n$.

The correctness and the security of this construction are based on the remark that, for any $A \in \mathcal{P}(\{1, 2, \ldots, n\})$, $\cup_{i \in A} I_i = \{s_j | j \in \cup_{i \in A} f(i)\}$ and, thus, $\cup_{i \in A} I_i = \{s_1, \ldots, s_m\} \Leftrightarrow \cup_{i \in A} f(i) = \{1, 2, \ldots, m\} \Leftrightarrow A \in \mathcal{A}$.

For efficiency reasons, it is important to find $\mathcal{A}$-cumulative maps (m, f) with a small m. A lower bound for the component m of an $\mathcal{A}$-cumulative map (m, f) has been established in [138]:

Proposition 2.7.1 *Any $\mathcal{A}$-cumulative map (m, f) must satisfy $m \geq |\overline{\mathcal{A}}_{max}|$.*

Proof Let (m, f) be an $\mathcal{A}$-cumulative map. Let us consider the function $h : \overline{\mathcal{A}}_{max} \to \{1, 2, \ldots, m\}$, given by

$h(B) =$ "an element i such that $i \in \{1, 2, \ldots, m\} \setminus \cup_{j \in B} f(j)$",

for any $B \in \overline{\mathcal{A}}_{max}$. We have to remark that h is well-defined by the fact that for any $B \in \overline{\mathcal{A}}_{max}$, $\cup_{j \in B} f(j) \subset \{1, 2, \ldots, m\}$, and by the Axiom of Choice. We will show that h is an injective function which will finally lead to $m \geq |\overline{\mathcal{A}}_{max}|$.

Let $B_1, B_2 \in \overline{\mathcal{A}}_{max}$ such that $h(B_1) = h(B_2)$. Thus, there is an element $i \in (\{1, 2, \ldots, m\} \setminus \cup_{j \in B_1} f(j)) \cap (\{1, 2, \ldots, m\} \setminus \cup_{j \in B_2} f(j))$ which is equivalent with $i \in \{1, 2, \ldots, m\} \setminus \cup_{j \in B_1 \cup B_2} f(j)$. Thus, the set $B_1 \cup B_2$ is an unauthorized access set that includes B_1, a maximal unauthorized access set. The only possibility is that $B_1 \cup B_2 = B_1$ which is equivalent with $B_2 \subseteq B_1$. Similarly, we can prove that $B_1 \subseteq B_2$ which will finally lead to $B_1 = B_2$. □

An $\mathcal{A}$-cumulative map (m, f) with $m = |\overline{\mathcal{A}}_{max}|$ has been called *minimal* in [91]. Ito, Saito, and Nishizeki [90] have inspired the construction of the following minimal $\mathcal{A}$-cumulative map. Let $\overline{\mathcal{A}}_{max} = \{B_1, B_2, \ldots, B_m\}$. The function f is defined as

$$f(i) = \{j \in \{1, 2, \ldots, m\} | i \notin B_j\}.$$

For an access structure $\mathcal{A}$ be over $\{1, 2, \ldots, n\}$, an $\mathcal{A}$-cumulative map (m, f) may be represented, as has been proposed in [138], using an $n \times m$ matrix $\mathcal{C}^{(m,f)}$, with binary entries, given by

$$\mathcal{C}^{(m,f)}_{i,j} = \begin{cases} 1, & \text{if } j \in f(i); \\ 0, & \text{otherwise} \end{cases}$$

for all $1 \leq i \leq n$ and $1 \leq j \leq m$.

In case of the minimal cumulative map described above, the corresponding matrix, denoted simply as $\mathcal{C}^{\mathcal{A}}$, is given by

$$\mathcal{C}^{\mathcal{A}}_{i,j} = \begin{cases} 1, & \text{if } i \notin B_j; \\ 0, & \text{otherwise} \end{cases}$$

for all $1 \leq i \leq n$ and $1 \leq j \leq m$, where $\overline{\mathcal{A}}_{max} = \{B_1, B_2, \ldots, B_m\}$.

In [91], the matrix $\mathcal{C}^{\mathcal{A}}$ has been referred to as the $\mathcal{A}$-*cumulative array.*

Example 2.7.1 illustrates this construction.

Example 2.7.1 Let $n = 4$ and $\mathcal{A}_{min} = \{\{1,2\},\{3,4\}\}$. In this case, $\overline{\mathcal{A}}_{max} = \{\{1,3\},\{1,4\},\{2,3\},\{2,4\}\}$ and $m = 4$. The cumulative array for the access structure $\mathcal{A}$ is $\mathcal{C}^{\mathcal{A}} = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$. In this case, the shares of a secret S with respect to an $\mathcal{A}$-secret sharing scheme can be defined as $I_1 = \{s_3, s_4\}$, $I_2 = \{s_1, s_2\}$, $I_3 = \{s_2, s_4\}$ and $I_4 = \{s_1, s_3\}$, where s_1, s_2, s_3, s_4 are the shadows of S with respect to a unanimous consent scheme of rank 4.

For another constructions of cumulative secret sharing schemes, the reader is referred to [71].

The main problem of this technique is that there exist access structures in which the number of the maximal unauthorized sets is quite large, leading to large shares. This is the case of the (k,n)-threshold access structures, where $|\overline{\mathcal{A}}_{max}| = \frac{n!}{(k-1)!(n-k+1)!}$.

2.7.2 Constructions based on Formula Templates

Benaloh and Leichter [9] have represented the access structures using formulae. More exactly, for a monotone authorized access structure $\mathcal{A}$ over $\{1,2,\ldots,n\}$, they have defined the set $\mathcal{F}_{\mathcal{A}}$ as the set of monotone formulae (i.e., formulae constructed using only conjunctions and disjunctions) over a set of variables $\{v_1, v_2, \ldots, v_n\}$ such that, for every $F \in \mathcal{F}_{\mathcal{A}}$, the interpretation of F with respect to an assignation of the variables is *true* if and only if the true variables correspond to a set $A \in \mathcal{A}$. They have remarked that such formulae can be used as templates for describing how a secret can be shared with respect to a given access structure. Because the formulae can be expressed using only $\wedge$ operators and $\vee$ operators, it is sufficient to indicate how to split the secret across these operators.

First we inductively define the shares of a secret S with respect to a formula F as follows

$$Shares(S,F) = \begin{cases} (S,i), & \text{if } F = v_i,\ 1 \leq i \leq n; \\ \cup_{i=1}^{k} Shares(S,F_i), & \text{if } F = F_1 \vee F_2 \vee \cdots \vee F_k; \\ \cup_{i=1}^{k} Shares(s_i,F_i), & \text{if } F = F_1 \wedge F_2 \wedge \cdots \wedge F_k, \end{cases}$$

where, for the case $F = F_1 \wedge F_2 \wedge \cdots \wedge F_k$, we can use any unanimous consent scheme of rank k for deriving the shares $s_1, \ldots, s_k$ corresponding to the secret S. Finally, we can define the shares with respect to the access structure $\mathcal{A}$ as $I_i = \{s | (s, i) \in \mathit{Shares}(S, F)\}$, for all $1 \leq i \leq n$, where F is an arbitrary formula in the set $\mathcal{F}_{\mathcal{A}}$ (for example, $F = \vee_{A \in \mathcal{A}_{min}} \wedge_{i \in A} v_i$).

Example 2.7.2 illustrates this construction.

Example 2.7.2 Let $n = 3$ and let us consider an authorized access structure $\mathcal{A}$ given by $\mathcal{A}_{min} = \{\{1, 2\}, \{2, 3\}\}$. For example, the formula

$$F = (v_1 \wedge v_2) \vee (v_2 \wedge v_3)$$

is in the set $\mathcal{F}_{\mathcal{A}}$. In this case, $\mathit{Shares}(S, F)$, for some secret S, can be obtained as

$$\begin{array}{rcl} \mathit{Shares}(S, F) & = & \mathit{Shares}(S, v_1 \wedge v_2) \cup \mathit{Shares}(S, v_2 \wedge v_3) \\ & = & (\mathit{Shares}(s_1, v_1) \cup \mathit{Shares}(s_{2,1}, v_2)) \cup \\ & & \cup\ (\mathit{Shares}(s_{2,2}, v_2) \cup \mathit{Shares}(s_3, v_3)) \\ & = & \{(s_1, 1), (s_{2,1}, 2), (s_{2,2}, 2), (s_3, 3)\}, \end{array}$$

where $s_1, s_{2,1}$ and, respectively, $s_{2,2}, s_3$ are the shares of the secret S with respect to two arbitrary unanimous consent schemes of rank 2. Thus, the shares corresponding to the secret S with respect to the access structure $\mathcal{A}$ are $I_1 = \{s_1\}$, $I_2 = \{s_{2,1}, s_{2,2}\}$, and $I_2 = \{s_3\}$.

Remark 2.7.1 A shadow I_i may contain many sub-shadows, one sub-shadow for each minimal access set to which i belongs. Thus, an ordering of these sub-shadows is required in order to select the correct sub-shadow corresponding to a certain access set in the reconstruction phase.

Remark 2.7.2 Benaloh and Leichter have also proposed using $\mathtt{threshold}_{\mathtt{k,m}}$ operators[5] in order to construct smaller formulae, reducing in this way the size of the shadows. In this case, the definition of $\mathit{Shares}(S, F)$ can be extended for these operators as follows:

$$\mathit{Shares}(S, F) = \cup_{i=1}^{m} \mathit{Shares}(s_i, F_i),$$

if $F = \mathtt{threshold}_{k,m}(F_1, \ldots, F_m)$, where $s_1, \ldots, s_m$ are the shadows corresponding to the secret S with respect to an arbitrary (k, m)-threshold secret sharing scheme.

[5] For $m \geq 1$, $1 \leq k \leq m$, $\mathtt{threshold}_{k,m}(F_1, \ldots, F_m)$ denotes the formula

$$\bigvee_{1 \leq i_1 < i_2 < \cdots < i_k \leq m} (\bigwedge_{j=1}^{k} F_{i_j})$$

Thus, $F_1 \vee F_2 \vee \cdots \vee F_m = \mathtt{threshold}_{1,m}(F_1, \ldots, F_m)$ and
$F_1 \wedge F_2 \wedge \cdots \wedge F_m = \mathtt{threshold}_{m,m}(F_1, \ldots, F_m)$.

Example 2.7.3 Let $n = 4$ and $\mathcal{A}$ be the access structure given by $\mathcal{A}_{min} = \{\{2,3\},\{1,2,4\},\{1,3,4\}\}$. For example, the formula

$$F = (v_2 \wedge v_3) \vee (v_1 \wedge v_2 \wedge v_4) \vee (v_1 \wedge v_3 \wedge v_4)$$

is in the set $\mathcal{F}_{\mathcal{A}}$. Using the `threshold` operator, we can obtain a shorter formula, namely,

$$(v_2 \wedge v_3) \vee \texttt{threshold}_{3,4}(v_1, v_2, v_3, v_4).$$

Vinod, Narayanan, Srinathan, Rangan, and Kim [150] have proposed a computational-secure secret sharing scheme for the access structures which can be represented using boolean circuits. Besides the classical gates $\wedge$ and $\vee$, the boolean circuits use a special one - the $FANOUT$ gate that takes a single input and produces two copies of it as output (for more details, the reader is referred to [150]).

2.7.3 General Secret Sharing based on the Chinese Remainder Theorem

In [85] we have extended the threshold secret sharing schemes based on the Chinese remainder theorem in order to deal with more general access structures.

For this, we have generalized the threshold Mignotte and Asmuth-Bloom sequences in a natural manner, the rest of these secret sharing schemes remaining unaffected.

We begin with introducing the extended Mignotte sequences.

Definition 2.7.2 Let n be a positive integer, $n \geq 2$ and $\mathcal{A} \subseteq \mathcal{P}(\{1, 2, \ldots, n\})$. An $\mathcal{A}$-*Mignotte sequence* is a sequence of positive integers $p_1, \ldots, p_n$ such that

$$\beta = max_{B \in \overline{\mathcal{A}}}([\{p_i | i \in B\}]) < min_{A \in \mathcal{A}}([\{p_i | i \in A\}]) = \alpha.$$

The above property is equivalent with

$$max_{B \in \overline{\mathcal{A}}_{max}}([\{p_i | i \in B\}]) < min_{A \in \mathcal{A}_{min}}([\{p_i | i \in A\}]).$$

If $\mathcal{A}$ is specified by $\mathcal{A}_{min} = \{A \in \mathcal{P}(\{1, 2, \ldots, n\}) \mid |A| = k\}$ then any $\mathcal{A}$-Mignotte sequence is a generalized threshold (k, n)-Mignotte sequence in sense of Definition 2.1.3. It is easy to see that if we multiply the elements of an $\mathcal{A}$-Mignotte sequence $p_1, \cdots, p_n$ with a fixed element $\delta \in \mathbf{Z}$, $(\delta, p_1 \cdots p_n) = 1$, we also obtain an $\mathcal{A}$-Mignotte sequence.

In Sections 2.1.3 and 2.4 we have presented how to generate Mignotte (and Asmuth-Bloom) sequences for the threshold, and, respectively, for the weighted threshold access structures. We demonstrate next that our extensions can be applied also to some non-weighted threshold access structures. Let reconsider Example 2.4.1 in which we have proven that the access structure specified by$\mathcal{A}_{min} = \{\{1,2\},\{3,4\}\}$ is not weighted threshold . We will present now how to realize this access structure using the proposed extension of Mignotte's scheme. In fact, the main problem is to find an $\mathcal{A}$-Mignotte sequence. More exactly, we are interested in finding a sequence of positive integers p_1, p_2, p_3, p_4 such that

$$max([p_1,p_3],[p_1,p_4],[p_2,p_3],[p_2,p_4]) < min([p_1,p_2],[p_3,p_4]).$$

It is interesting to remark that this access structure cannot be realized using sequences of pairwise coprime numbers. Indeed, there is no $\mathcal{A}$-Mignotte sequence with pairwise coprime elements, because, otherwise, the above inequality will lead to $p_1p_3 < p_1p_2$ and $p_2p_4 < p_3p_4$ and, thus, to $p_3 < p_2$ and $p_2 < p_3$!

If q_1, q_2, q_3, q_4 are pairwise coprime, then the sequence $p_1 = q_1q_2, p_2 = q_3q_4, p_3 = q_1q_3, p_4 = q_2q_4$ is an $\mathcal{A}$-Mignotte sequence. In this case, the general variant of the Chinese remainder theorem must be used for recovering the secret.

Asmuth-Bloom sequences can be extended as follows.

Definition 2.7.3 Let n be a positive integer, $n \geq 2$ and $\mathcal{A} \subseteq \mathcal{P}(\{1,2,\ldots,n\})$. An $\mathcal{A}$-*Asmuth-Bloom sequence* is a sequence of positive integers $p_0, p_1, \ldots, p_n$ such that

$$p_0 \cdot max_{B \in \overline{\mathcal{A}}}([\{p_i | i \in B\}]) < min_{A \in \mathcal{A}}([\{p_i | i \in A\}]).$$

The above property is equivalent with

$$p_0 \cdot max_{B \in \overline{\mathcal{A}}_{max}}([\{p_i | i \in B\}]) < min_{A \in \mathcal{A}_{min}}([\{p_i | i \in A\}]).$$

If $\mathcal{A}$ is specified by $\mathcal{A}_{min} = \{A \in \mathcal{P}(\{1,2,\ldots,n\}) \mid |A| = k\}$ then any $\mathcal{A}$-Asmuth-Bloom sequence is a generalized threshold (k,n)-Asmuth-Bloom sequence. It is easy to see that if we multiply the elements, excepting the first one, of an $\mathcal{A}$-Asmuth-Bloom sequence $p_0, p_1, \cdots, p_n$ with a fixed element $\delta \in \mathbf{Z}$, $(\delta, p_0 \cdots p_n) = 1$, we also obtain an $\mathcal{A}$-Asmuth-Bloom sequence.

Remark 2.7.3 If $p_1, \ldots, p_n$ is an $\mathcal{A}$-Mignotte sequence then $p_0, p_1, \ldots, p_n$ is an $\mathcal{A}$-Asmuth-Bloom sequence, for any $p_0 < \frac{\alpha}{\beta}$.

If $p_0, p_1, \ldots, p_n$ is an $\mathcal{A}$-Asmuth-Bloom sequence then $p_1, \ldots, p_n$ is an $\mathcal{A}$-Mignotte sequence.

The access structure given by $\mathcal{A}_{min} = \{\{1,2\},\{3,4\}\}$ cannot be realized using Asmuth-Bloom sequences of pairwise coprime numbers. Indeed, there is no $\mathcal{A}$-Asmuth-Bloom sequence with pairwise coprime elements, because, otherwise, the condition $p_0 \cdot max([p_1, p_3], [p_1, p_4], [p_2, p_3], [p_2, p_4]) < min([p_1, p_2], [p_3, p_4])$ will lead to $p_0p_1p_3 < p_1p_2$ and $p_0p_2p_4 < p_3p_4$ and, thus, to $p_0p_3 < p_2$ (which implies $p_0^2p_3 < p_0p_2$), and $p_0p_2 < p_3$, which will finally lead to $p_0^2 < 1$!

If q_1, q_2, q_3, q_4 are pairwise coprime, then the sequence $p_0, p_1 = q_1q_2, p_2 = q_3q_4, p_3 = q_1q_3, p_4 = q_2q_4$ is an $\mathcal{A}$-Asmuth-Bloom sequence, for any $p_0 < min(q_1, q_2, q_3, q_4)$. Indeed, in this case, $min([p_1, p_2], [p_3, p_4]) = q_1q_2q_3q_4$ and $max([p_1, p_3], [p_1, p_4], [p_2, p_3], [p_2, p_4]) = \frac{q_1q_2q_3q_4}{q_i}$, for some $i \in \{1, 2, 3, 4\}$. In this case, the general variant of the Chinese remainder theorem must be used for recovering the secret.

2.7.4 General Secret Sharing based on Determinants

In [81] we have proposed a secret sharing framework based on determinants. We present first some basic facts about determinants. For more details on determinants, the reader is referred to [24].

Definition 2.7.4 Let $m \geq 1$ and A be a square matrix of dimension m over an arbitrary commutative ring $(R, +, \cdot)$, $A = (a_{i,j})_{1 \leq i,j \leq m}$. The *determinant* of A, denoted by $det(A)$ (or by $|A|$), is given by

$$det(A) = \sum_{\sigma \in \mathcal{P}_m} ((-1)^{inv(\sigma)} \prod_{i=1}^{m} a_{i,\sigma(i)}),$$

where $\mathcal{P}_m$ represents the set of the permutations of the set $\{1, 2, \ldots, m\}$ and $inv(\sigma)$ is the number of pairs (i, j) such that $1 \leq i < j \leq m$ and $\sigma(i) > \sigma(j)$.

For computing the determinant of a matrix A of dimension m, $A = (a_{i,j})_{1 \leq i,j \leq m}$, several approaches exist:

- using definition - not very efficient for large matrices;
- using the expansion along a row (or a column) - more exactly, for a row index i, $1 \leq i \leq m$, we have that $det(A) = \sum_{j=1}^{m} (-1)^{i+j} a_{i,j} M_{i,j}$, where $M_{i,j}$ is the determinant of the matrix obtained from A by removing its i^{th} row and its j^{th} column. This formula can be recursively used in order to obtain the required determinant. At each step, we have to pick a row (or a column) with many zeros. This method is also not very efficient for large matrices;
- using matrix transformations - the matrix A is transformed by some operations into a matrix of some particular form as the *row echelon form* (see, for example, [24]) or the *Smith normal form* (see, for example, [54]). Kaltofen has proposed in [96] an efficient method of computing determinants without divisions.

The main idea behind this scheme is that a matrix over an arbitrary commutative ring $(R, +, \cdot)$ is "puzzled" and the set of the resulted pieces is partitioned in the shares such that every maximal unauthorized access set cannot solve the puzzle and such that every minimal authorized access set can do it. The secret will be the module[6] of the determinant of the formed matrix. Let $\mathcal{A}$ be a monotone authorized access structure. We will describe two methods of realizing $\mathcal{A}$-secret sharing.

Method I

Suppose that $B_1, B_2, \ldots, B_m$ are the maximal unauthorized access sets corresponding to $\mathcal{A}$ and let A be a nonsingular square matrix of dimension m, $A = (a_{i,j})_{1 \le i,j \le m}$. The secret will be chosen as $S = |det(A)|$. The corresponding shares will be defined as $I_i = \{(a_{j,1}, \ldots, a_{j,m}) | i \notin B_j\}$, for all $1 \le i \le n$. Every minimal access authorized group can combine their rows of the matrix, eventually eliminating the duplicates, in order to obtain the secret.

Correctness of the scheme
Suppose we have $\{I_i | i \in C\}$, for some set $C \in \mathcal{A}_{min}$. We will prove that the matrix A can be determined. Assume on the contrary that there is a row of the matrix, namely the j^{th} row, that is not contained in $\cup_{i \in C} I_i$. By the definition of the shares, we obtain that $i \in B_j$, for all $i \in C$, and, hence $C \subseteq B_j$ which leads, by the fact that $C \in \mathcal{A}$ and that $\mathcal{A}$ is a monotone access structure, to $B_j \in \mathcal{A}$!

Security of the scheme
Suppose we have $\{I_i | i \in B_l\}$, for some $1 \le l \le m$. We will prove that the matrix A cannot be determined. Assume on the contrary that the set $\cup_{i \in B_l} I_i$ contains all the rows of the matrix, i.e., for all $1 \le j \le m$, there exists $i \in B_l$ such that $(a_{j,1}, \ldots, a_{j,m}) \in I_i$. By the definition of the shares, we obtain that for all $1 \le j \le m$, there is $i \in B_l$ such that $i \notin B_j$. For $j = l$ we obtain that there is $i \in B_l$ such that $i \notin B_l$!

Example 2.7.4 Let $n = 3$ and $\mathcal{A}$ be an access structure given by $\mathcal{A}_{min} = \{\{1,2\},\{2,3\}\}$. In this case $\{1,3\}$ and $\{2\}$ are the maximal unauthorized sets, and, thus, $m = 2$. Let us consider the matrix $A = \begin{pmatrix} 2 & 5 \\ 3 & 6 \end{pmatrix}$ over the field of positive numbers modulo 7. Then, the secret will be $S = |det(A)| = 3$ and the shares will be $I_1 = \{(3,6)\}$, $I_2 = \{(2,5)\}$, and $I_3 = \{(3,6)\}$.

[6] The *module* of an element depends on the ring $(R, +, \cdot)$. For example, for an element a of the field of the real numbers, $|a| = \begin{cases} a, & \text{if } a \ge 0; \\ -a, & \text{if } a < 0 \end{cases}$, for an element a of the field of the positive integers modulo an odd prime p, $|a| = \begin{cases} a, & \text{if } a \le \frac{p-1}{2}; \\ p-a, & \text{otherwise} \end{cases}$

Method II.
Suppose that $C_1, C_2, \ldots, C_l$ are the minimal authorized access sets corresponding to $\mathcal{A}$, of cardinality m_1,m_2,…,m_l, and let $A_1, A_2, \ldots, A_l$ be some nonsingular square matrices of dimensions m_1,m_2,…,m_l, such that $|det(A_1)| = |det(A_2)| = \cdots = |det(A_l)| = S$, $A_k = (a^k_{i,j})_{1\leq i,j\leq m_k}$, for $1 \leq k \leq l$. The shares of the secret S will be generated as

$$I_i = \{(a^k_{f_k(i),1}, \ldots, a^k_{f_k(i),m_k})|i \in C_k,\ 1 \leq k \leq l\},$$

for all $1 \leq i \leq n$, where f_k is an arbitrary injective function from C_k to $\{1, \ldots, m_k\}$, for all $1 \ \leq \ k \ \leq \ l$.

Example 2.7.5 Let $n = 4$ and $\mathcal{A}$ be an access structure given by $\mathcal{A}_{min} = \{\{1, 2\}, \{1, 3, 4\}\}$. We will consider the matrices $A_1 = \begin{pmatrix} 4 & 2 \\ 2 & 5 \end{pmatrix}$ and $A_2 = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 5 \\ 3 & 2 & 6 \end{pmatrix}$ over the field of positive numbers modulo 7. Then, the secret will be $S = |det(A_1)| = |det(A_2)| = 2$ and the shares will be $I_1 = \{(2, 5), (2, 1, 5)\}$, $I_2 = \{(4, 2)\}$, $I_3 = \{(1, 2, 3)\}$, and $I_4 = \{(3, 2, 6)\}$.

We present next some interesting changeability properties of our scheme. First we consider the problem of changing the shares without changing the secret. This may be the case when the secret is explicit and some of the current shares have been compromised. The dealer wants to generate new shares using the old ones. Our scheme provides this capability in a natural manner. We will use *unimodular* matrices, i.e., matrices with determinant ±1, for modifying the shares without changing the secret. We illustrate this idea in the following examples.

Example 2.7.6 Let us reconsider Example 2.7.4 and consider the unimodular matrix $B = \begin{pmatrix} 2 & 3 \\ 1 & 2 \end{pmatrix}$. We obtain that $B \cdot A = \begin{pmatrix} 6 & 0 \\ 1 & 3 \end{pmatrix}$. The new shadows will be $I_1 = \{(1, 3)\}$, $I_2 = \{(6, 0)\}$, and $I_3 = \{(1, 3)\}$.

Example 2.7.7 Let us reconsider Example 2.7.5 and consider the unimodular matrices $B_1 = \begin{pmatrix} 2 & 3 \\ 1 & 2 \end{pmatrix}$ and $B_2 = \begin{pmatrix} 1 & 2 & 4 \\ 0 & 3 & 6 \\ 0 & 0 & 5 \end{pmatrix}$. We obtain that $B_1 \cdot A_1 = \begin{pmatrix} 0 & 5 \\ 1 & 5 \end{pmatrix}$ and $B_2 \cdot A_2 = \begin{pmatrix} 3 & 5 & 2 \\ 3 & 1 & 2 \\ 1 & 3 & 2 \end{pmatrix}$. The new shadows will be $I_1 = \{(1, 5), (3, 1, 2)\}$, $I_2 = \{(0, 5)\}$, $I_3 = \{(3, 5, 2)\}$, and $I_4 = \{(1, 3, 2)\}$.

We show next that our scheme supports the dynamic update of the authorized access structure in the sense that, without modifying the secret, we can perform some operations on the minimal authorized sets. For simplicity, we will consider only the case of removing an arbitrary element from some minimal authorized set, but we have to mention that this technique can be extended to other kinds of operations on the access structure in a natural manner. More exactly, let us consider a monotone authorized access structure $\mathcal{A}$ and suppose that $C_1, C_2, \ldots, C_l$ are the minimal authorized access sets corresponding to $\mathcal{A}$. Let us consider an arbitrary element $i \in \{1, 2, \ldots, n\}$ and choose an arbitrary set C_j such that $i \in C_j$. In accordance with our second secret sharing method, the shadow I_i will contain a row vector $(v_1^{i,j}, \ldots, v_{m_j}^{i,j})$ corresponding to the minimal access set C_j, where $m_j = |C_j|$. The shares I_k, for all $k \in C_j \setminus \{i\}$, will be actualized as follows:

$$I_k := I_k \cup \{(s, v_s^{i,j}) | s \in P_k\},$$

where $\{P_k | k \in C_j \setminus \{i\}\}$ is an arbitrary partition of the set $\{1, 2, \ldots, m_j\}$. In this way, by combining the shadows corresponding to the set $C_j \setminus \{i\}$, the vector $(v_1^{i,j}, \ldots, v_{m_j}^{i,j})$ can be reconstructed and, thus, the set $C_j \setminus \{i\}$ becomes a minimal authorized access set. We will illustrate this technique in the following example.

Example 2.7.8 Let us reconsider Example 2.7.5 in which we have presented an $\mathcal{A}$-secret sharing scheme where $\mathcal{A}_{min} = \{\{1, 2\}, \{1, 3, 4\}\}$. We have obtained the secret $S = 2$ and the shares $I_1 = \{(2, 5), (2, 1, 5)\}$, $I_2 = \{(4, 2)\}$, $I_3 = \{(1, 2, 3)\}$, and $I_4 = \{(3, 2, 6)\}$.

Suppose that we want to eliminate the element 1 from the minimal access set $\{1, 3, 4\}$. The row vector of the share I_1 corresponding to the minimal access set $\{1, 3, 4\}$ is $(2, 1, 5)$ and if we consider $P_3 = \{1, 3\}$ and $P_4 = \{2\}$, the shadows I_3 and I_4 will become $I_3 = \{(1, 2, 3), \{(1, 2), (3, 5)\}\}$ and $I_4 = \{(3, 2, 6), \{(2, 1)\}\}$.

2.7.5 General Secret Sharing based on Information Dispersal

Béguin and Cresti [2, 3] have extended the Krawczyk's technique (see Section 2.1.4) to the general monotone access structure by introducing the *general information dispersal schemes.*

Definition 2.7.5 Let $\mathcal{A}$ be an access structure. Informally, an $\mathcal{A}$-*information dispersal scheme* is a method of generating $(S, (F_1, \ldots, F_n))$ such that, for any $A \in \mathcal{A}$, the problem of finding the information S, given the set of fragments $\{F_i \mid i \in A\}$, is "easy".

Béguin and Cresti proposed the following general information dispersal algorithm:

- For all $i \in \{1, 2, \ldots, n\}$ choose the positive integers p_i and q_i such that

$$(\forall A \in \mathcal{A}_{min})(\sum_{i \in A} \frac{p_i}{q_i} \geq 1) \tag{2.4}$$

  and let $k' = [q_1, \ldots, q_n]$ and $n' = k' \cdot \sum_{i=1}^{n} \frac{p_i}{q_i}$ (remark that $k' \leq n'$);

- Use an (k', n')-threshold information dispersal scheme (see Section 2.1.4) for constructing the fragments $F_1', \ldots, F_{n'}'$ corresponding to the information S;

- Define the fragments $F_1, \ldots, F_n$ as follows: $F_i = \{F_j' | j \in P_i\}$, where $\{P_1, \ldots, P_n\}$ is an arbitrary partition of the set $\{1, 2, \ldots, n'\}$ such that $|F_i| = k' \cdot \frac{p_i}{q_i}$, for all $1 \leq i \leq n$;

- Having the fragments $\{F_i | i \in A\}$, for some minimal authorized group A, the information S can be recovered using the reconstruction algorithm from the (k', n')-threshold information dispersal scheme. Indeed, by the choice of the fragments and of the elements p_i and q_i, we will successively obtain that

$$\begin{array}{rcl} | \cup_{i \in A} F_i | & = & | \cup_{i \in A} \{F_j' | j \in P_i\} | \\ & = & \sum_{i \in A} |P_i| \\ & = & \sum_{i \in A} k' \cdot \frac{p_i}{q_i} \\ & = & k' \cdot \sum_{i \in A} \frac{p_i}{q_i} \\ & \geq & k', \end{array}$$

  and, thus, at least k' elements from the set $\{F_1', \ldots, F_{n'}'\}$ can be gathered.

A possibility of choosing the elements p_i and q_i such that relation 2.4 holds true is $p_i = 1$ and $q_i = \min(\{|A| \mid A \in \mathcal{A}_{min} \wedge i \in A\}$. Indeed, in this case, for a minimal group A, we will obtain that $q_i \leq |A|$, for all $i \in A$ and thus $\sum_{i \in A} \frac{p_i}{q_i} \geq \sum_{i \in A} \frac{1}{|A|} = 1$. Béguin and Cresti have proposed another method of choosing the elements p_i and q_i, based on linear programming.

In [83] we have presented a general information dispersal scheme based on the Chinese remainder theorem:

- The information S is chosen as an arbitrary positive integer such that $S < min_{A \in \mathcal{A}_{min}}([\{p_i | i \in A\}])$ where $p_1, \ldots, p_n$ are arbitrary positive integers;

- The fragments $F_1, \ldots, F_n$ are chosen as $F_i = S \bmod p_i$, for all $1 \leq i \leq n$;

- Having a set of fragments $\{F_i \mid i \in A\}$ for some $A \in \mathcal{A}_{min}$, the information S can be obtained as the unique solution modulo $[\{p_i | i \in A\}]$

of the system of equations

$$\left\{ \begin{array}{rcl} x & \equiv & F_i \; mod \; p_i, \quad i \in A. \end{array} \right.$$

Indeed, the information S is the unique solution modulo $[\{p_i | i \in A\}]$ of the above system of equations because S is an integer solution of the system by the choice of the fragments $F_1, \ldots, F_n$ and, moreover, $S \in \mathbf{Z}_{[\{p_i|i \in A\}]}$, because $S < min_{A \in \mathcal{A}_{min}}([\{p_i | i \in A\}])$.

The next example illustrates this scheme.

Example 2.7.9 Let $n = 4$ and $\mathcal{A}_{min} = \{\{1, 2\}, \{3, 4\}\}$. Let us consider $p_1 = 9$, $p_2 = 16$, $p_3 = 12$, and $p_4 = 18$. Suppose that the information is $S = 61$. We obtain the fragments $F_1 = 7$, $F_2 = 13$, $F_3 = 1$, and $F_4 = 7$. If we have the first two fragments, the information S can be obtained as the unique solution modulo 144 of the system of equations

$$\left\{ \begin{array}{rcl} x & \equiv & 7 \; mod \; 9 \\ x & \equiv & 13 \; mod \; 16 \end{array} \right. .$$

An important issue in information dispersal is the size of fragments comparing to the size of the information. In our scheme, by finding $p_1, \ldots, p_n$, and the biggest $\alpha \geq 1$ such that

$$(max(p_1, \ldots, p_n))^{\alpha} < S < min_{A \in \mathcal{A}_{min}}([\{p_i | i \in A\}])$$

we can obtain short fragments. Indeed, in this case we obtain that $F_i < p_i \leq max(p_1, \ldots, p_n) < S^{\frac{1}{\alpha}}$ and, thus, $|F_i| < \frac{|S|}{\alpha}$, for all $1 \leq i \leq n$.

2.7.6 General Secret Sharing based on Public Information

In this section we describe a construction for computational-secure general secret sharing based on public information due to Cachin [26].

- Setup (performed by the dealer)
  - Choose an additive group $\mathbf{G}$ and randomly generate the secret $S \in \mathbf{G}$ and the shares $I_1, \ldots, I_n \in \mathbf{G}$;
  - Secretly send I_i to the i^{th} user, for all $1 \leq i \leq n$;
  - For any $A \in \mathcal{A}_{min}$, compute and broadcast $T_A = S - f(\sum_{i \in A} I_i)$, where f is an arbitrary one-way function, $f : \mathbf{G} \rightarrow \mathbf{G}$;
- Reconstruction - let A be a minimal authorized group - the secret S can be reconstructed as $S = f(\sum_{i \in A} I_i) + T_A$.

Another scheme based on public information has been proposed by Pinch in [127]. These schemes have the disadvantage that a large amount of information must be broadcasted and authenticated.

2.8 Ramp Secret Sharing Schemes

Ramp scheme have appeared as a solution for the situations in which smaller shares are required. Secret sharing schemes which can achieve $H(\mathtt{I}_i) < H(\mathtt{S})$, for some $1 \leq i \leq n$, must be non-perfect. Ramp schemes consider some *semi-access groups* who can obtain some information about the secret. Thus, ramp schemes may assure a certain compromise between the level of security and the size of the shares.

Ramp schemes have been introduced independently for the threshold case by Blakley and Meadows [13] and by Yamamoto [153]. Let $n \geq 2$, $1 \leq k \leq n$ and $1 \leq l \leq k$. Informally, a (l, k, n)-*threshold ramp scheme* is a method of generating $(S, (I_1, \ldots, I_n))$ such that

- for any $A \in \mathcal{P}(\{1, 2, \ldots, n\})$ such that $|A| \geq k$, the problem of finding the element S, given the set $\{I_i \mid i \in A\}$, is "easy";
- for any $A \in \mathcal{P}(\{1, 2, \ldots, n\})$ such that $k - l + 1 \leq |A| \leq k - 1$, some information about S can be found having $\{I_i \mid i \in A\}$;
- for any $A \in \mathcal{P}(\{1, 2, \ldots, n\})$ such that $|A| \leq k-l$, no information about S can be found (in information-theoretic sense) having $\{I_i \mid i \in A\}$.

In this case, the semi-access sets are the sets A such that $k-l+1 \leq |A| \leq k - 1$. In the case that $l = 1$, the class of (l, k, n)-threshold ramp schemes coincides with the class of perfect (k, n)-threshold secret sharing schemes and in the case that $l = k$, the class of (l, k, n)-threshold ramp schemes coincides with the class of (k, n)-threshold information dispersal schemes.

Blakley and Meadows [13] have pointed out that Shamir's threshold secret sharing scheme can be transformed into a threshold ramp scheme by choosing the secret S as a vector $(P(x_1), \ldots, P(x_l))$ instead of $P(0)$. We will present next the scheme of Franklin and Young [64] which is based also on Shamir's scheme. The dealer chooses a polynomial P of degree at most $k - 1$ over the field of the positive integers modulo a large prime, $P(x) = a_0 + a_1 x + \cdots + a_{l-1} x^{l-1} + a_l x^l + \cdots + a_{k-1} x^{k-1}$. The secret is chosen as $S = (a_0, a_1, \ldots, a_{l-1})$ and the shares as $I_i = P(i)$, for all $1 \leq i \leq n$. Any k users can obtain S using Lagrange's formula, but any $k - l$ users cannot obtain any information about the secret.

The *linear ramp schemes* ([18]) are ramp schemes in which the amount of secret information obtained by a semi-access group grows linearly with respect to the size of the group. More exactly, a linear (l, k, n)-threshold ramp scheme is a collection of random variables $(\mathtt{S}, \mathtt{I}_1, \ldots, \mathtt{I}_n)$ such that

- for any $A \in \mathcal{P}(\{1, 2, \ldots, n\})$ such that $|A| \geq k$, $H(\mathtt{S}|\{\mathtt{I}_i \mid i \in A\}) = 0$;
- for any $A \in \mathcal{P}(\{1, 2, \ldots, n\})$ such that $k - l + 1 \leq |A| \leq k - 1$,
$$H(\mathtt{S}|\{\mathtt{I}_i \mid i \in A\}) = \frac{k - |A|}{l} H(\mathtt{S});$$

- for any $A \in \mathcal{P}(\{1, 2, \ldots, n\})$ such that $|A| \leq k - l$,

$$H(\mathtt{S}|\{\mathtt{I}_i \mid i \in A\}) = H(\mathtt{S}).$$

According to [18], in any linear (l, k, n)-threshold ramp scheme, the relation $H(\mathtt{I}_i) \geq \frac{H(\mathtt{S})}{l}$ holds true, for all $1 \leq i \leq n$. A linear (l, k, n)-threshold ramp scheme with $H(\mathtt{I}_i) = \frac{H(\mathtt{S})}{l}$, for all $1 \leq i \leq n$, has been referred to as *optimal* in [93].

Ogata, Kurosawa, and Tsujii [121] have introduced more general ramp schemes. An access structure $\mathcal{A}$ is partitioned in three parts: $\mathcal{A}_1$, the family of access sets, $\mathcal{A}_2$, the family of semi-access sets, and $\mathcal{A}_3$, the family of non-access sets.

Definition 2.8.1 Let $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ as above. An $(\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$-ramp scheme is a collection of random variables $(\mathtt{S}, \mathtt{I}_1, \ldots, \mathtt{I}_n)$ such that

- for any $A \in \mathcal{A}_1$, $H(\mathtt{S}|\{\mathtt{I}_i \mid i \in A\}) = 0$;
- for any $A \in \mathcal{A}_2$, $0 < H(\mathtt{S}|\{\mathtt{I}_i \mid i \in A\}) < H(\mathtt{S})$;
- for any $A \in \mathcal{A}_3$, $H(\mathtt{S}|\{\mathtt{I}_i \mid i \in A\}) = H(\mathtt{S})$.

In case $\mathcal{A}_1 = \{A \in \mathcal{P}(\{1, 2, \ldots, n\}) || A| \geq k\}$, $\mathcal{A}_2 = \{A \in \mathcal{P}(\{1, 2, \ldots, n\})| k - l + 1 \leq |A| \leq k - 1\}$, and $\mathcal{A}_3 = \{A \in \mathcal{P}(\{1, 2, \ldots, n\}) || A| \leq k - l\}$, any $(\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$-ramp scheme is in fact a (l, k, n)-threshold ramp scheme.

They have proposed the following construction in case that $\mathcal{A}_1$ and $\mathcal{A}_1 \cup \mathcal{A}_2$ are monotone. The secret S is chosen as a pair of sub-secrets (s_1, s_2). The shares are chosen as $I_i = (I_{i,1}, I_{i,2})$, for all $1 \leq i \leq n$, where $I_{1,1}, \ldots, I_{n,1}$ are the shares corresponding to the secret s_1 with respect to a perfect $\mathcal{A}_1$-secret sharing scheme and $I_{1,2}, \ldots, I_{n,2}$ are the shares corresponding to the secret s_2 with respect to a perfect $\mathcal{A}_1 \cup \mathcal{A}_2$-secret sharing scheme.

2.9 Constructions based on Decompositions

In this section we will present some constructions in which the secret sharing schemes are used as building blocks for realizing larger schemes. This topic has been addressed, for instance, in [17], [105], [111], [142].

We will present first Martin's method [105] for realizing insertions (see Section 1.1). In this section we will use the model for secret sharing based on distribution rules (see Section 1.2). Although Martin's method has been presented using a variation of Brickell-Davenport model, any matrix that represents a secret sharing scheme can be thought as a special set of distribution rules.

Let $\mathcal{A}$ be an access structure over the set $\mathcal{U}_1$, $\mathcal{B}$ be an access structure over the set $\mathcal{U}_2$, and $P \in \mathcal{U}_1$. Let $\mathcal{F}_A$ be an $\mathcal{A}$-secret sharing scheme with the set of secrets $\mathcal{S}_{A,0}$ and the sets of shares $\mathcal{S}_{A,i}$, $i \in \mathcal{U}_1$ (thus, $\mathcal{F}_A \subseteq \prod_{i\in\{0\}\cup\mathcal{U}_1} \mathcal{S}_{A,i}$) and $\mathcal{F}_B$ be a $\mathcal{B}$-secret sharing scheme with the set of secrets $\mathcal{S}_{B,0}$ and the sets of shares $\mathcal{S}_{B,i}$, $i \in \mathcal{U}_2$ (thus, $\mathcal{F}_B \subseteq \prod_{i\in\{0\}\cup\mathcal{U}_2} \mathcal{S}_{B,i}$). Without loss of generality, we will suppose that $\mathcal{S}_{B,0} = \mathcal{S}_{A,P}$. An $\mathcal{A}(P \to \mathcal{B})$-secret sharing scheme can be constructed as follows. For each $f \in \mathcal{F}_A$ and for each $g \in \mathcal{F}_B$ such that $g(0) = f(P)$ we will add an element $h \in \prod_{i\in\{0\}\cup(\mathcal{U}_1\setminus\{P\})\cup\mathcal{U}_2} \mathcal{S}_{A|B,i}$ (where $\mathcal{S}_{A|B,i} = \begin{cases} S_{A,0}, & \text{if } i = 0; \\ S_{A,i}, & \text{if } i \in \mathcal{U}_1; \\ S_{B,i}, & \text{otherwise} \end{cases}$) in $\mathcal{F}_{\mathcal{A}(P\to\mathcal{B})}$, given by

$$h(i) = \begin{cases} f(0), & \text{if } i = 0; \\ f(i), & \text{if } i \in (\mathcal{U}_1 \setminus \{P\}); \\ g(i), & \text{otherwise.} \end{cases}$$

It is easy to prove that $\mathcal{F}_{\mathcal{A}(P\to\mathcal{B})}$ is indeed an $\mathcal{A}(P \to \mathcal{B})$-secret sharing scheme. Intuitively, in this construction, the share of the user P from the $\mathcal{A}$-secret sharing scheme is replaced with the its shares corresponding to the $\mathcal{B}$-secret sharing scheme. The next example illustrates this construction.

Example 2.9.1 ([105]) Let $\mathcal{A} = \{\{1,2\},\{1,3\},\{1,2,3\}\}$, $\mathcal{B} = \{\{4,5\}\}$, $P = 2$ and consider their realizations (given as matrices) $\mathcal{F}_A = \left(\begin{array}{cc|c|c} \mathbf{0} & \mathbf{1} & \mathbf{2} & \mathbf{3} \\ \hline 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \end{array}\right)$ and $\mathcal{F}_B = \left(\begin{array}{|c|cc} \mathbf{0} & \mathbf{4} & \mathbf{5} \\ \hline 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{array}\right)$. Then $\left(\begin{array}{ccccc} \mathbf{0} & \mathbf{1} & \mathbf{4} & \mathbf{5} & \mathbf{3} \\ \hline 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \end{array}\right)$ is a realization of the access structure $\mathcal{A}(P \to \mathcal{B})$.

We will present next Stinson's decomposition construction [142]. We present first the notion of decomposition of an access structure.

Definition 2.9.1 Let $\mathcal{A}$ be an access structure and $\lambda \geq 1$. A $\lambda-$*decomposition* of $\mathcal{A}$ is a sequence $\mathcal{A}_1, \ldots, \mathcal{A}_m$ such that

1. $\mathcal{A}_j \subseteq \mathcal{A}$, for all $1 \leq j \leq m$;
2. For any $A \in \mathcal{A}_{min}$, there are $1 \leq i_1 < i_2 < \cdots < i_\lambda \leq m$ such that $A \in \mathcal{A}_{i_j}$, for all $1 \leq j \leq \lambda$.

Stinson has proven that if the closures of the access structures from a decomposition of a certain access structure can be realized, then that access structure can be also realized. More exactly, let us consider an access structure $\mathcal{A}$ and let $\mathcal{A}_1, \ldots, \mathcal{A}_m$ be a λ-decomposition of $\mathcal{A}$ such that there exists

$\mathcal{F}^j$, a $cl(\mathcal{A}_j)$-secret sharing scheme, having the set of secrets $\mathcal{S}_{j,0} = \mathbf{GF}_q$, for all $1 \leq j \leq m$. Suppose that there exist some vectors $v_1, \ldots, v_m \in \mathbf{GF}_q^\lambda$ such that

$$(*)\ \{v_j | A \in \mathcal{A}_j\} \text{ generates } \mathbf{GF}_q^\lambda,$$

for all $A \in \mathcal{A}_{min}$. An $\mathcal{A}$-secret sharing scheme $\mathcal{F}$ with the set of secrets $\mathbf{GF}_q^\lambda$ can be defined as follows. Let $S \in \mathbf{GF}_q^\lambda$, $S = (S_1, \ldots, S_\lambda)$. Define a distribution rule f for the secret S ($f(0) = S$) by $f(i) = (f_S^j(i) | i \in \cup \mathcal{A}_j)$, for all $1 \leq i \leq n$, where f_S^j is a distribution rule from $\mathcal{F}^j$ with the secret $S \cdot v_j$ ($\cdot$ denotes the scalar product in $\mathbf{GF}_q^\lambda$). Example 2.9.2 illustrates this construction.

Example 2.9.2 ([142]) Let $\mathcal{A}$ be a graph-based access structure that corresponds to C_5, the cycle with five vertices. The sequence $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4, \mathcal{A}_5$ given by $\mathcal{A}_1 = \{\{1,2\},\{2,3\}\}$, $\mathcal{A}_2 = \{\{2,3\},\{3,4\}\}$, $\mathcal{A}_3 = \{\{3,4\},\{4,5\}\}$, $\mathcal{A}_4 = \{\{4,5\},\{5,1\}\}$, $\mathcal{A}_5 = \{\{5,1\},\{1,2\}\}$, is a $2-$decomposition of $\mathcal{A}$. For each $1 \leq j \leq 5$, $\mathcal{A}_j$ is the access structure specified by the graph $K_{1,2}$ and, thus, there is $\mathcal{F}^j$, an ideal $cl(\mathcal{A}_j)$-secret sharing scheme (see Section 2.3). Let us consider the following distribution rules for sharing some secret $s \in \mathbf{GF}_q$:

$$\begin{array}{lll}
f^1_{r_1,s}(1) = r_1 & f^1_{r_1,s}(2) = r_1 + s & f^1_{r_1,s}(3) = r_1 \\
f^2_{r_2,s}(2) = r_2 & f^2_{r_2,s}(3) = r_2 + s & f^2_{r_2,s}(4) = r_2 \\
f^3_{r_3,s}(3) = r_3 & f^3_{r_3,s}(4) = r_3 + s & f^3_{r_3,s}(5) = r_3 \\
f^4_{r_4,s}(4) = r_4 & f^4_{r_4,s}(5) = r_4 + s & f^4_{r_4,s}(1) = r_4 \\
f^5_{r_5,s}(5) = r_5 & f^5_{r_5,s}(1) = r_5 + s & f^5_{r_5,s}(2) = r_5,
\end{array}$$

where $r_1, r_2, r_3, r_4, r_5 \in \mathbf{GF}_q$.

Let us consider $v_1 = (1,0)$, $v_2 = (0,1)$, $v_3 = (1,1)$, $v_4 = (1,2)$, and $v_5 = (1,3)$ which satisfy the requirement $(*)$. We will replace s in each distribution rule with a linear combination of S_1 and S_2, as specified by the vectors v_j, $1 \leq j \leq 5$, and we obtain the following set of distributions rules for sharing $S = (S_1, S_2)$ with respect to the access structure $\mathcal{A}$:

$$\begin{array}{rcc}
f_{r_1,r_2,r_3,r_4,r_5,S_1,S_2}(1) & = & (r_1, r_4, r_5 + S_1 + 3S_2), \\
f_{r_1,r_2,r_3,r_4,r_5,S_1,S_2}(2) & = & (r_2, r_5, r_1 + S_1), \\
f_{r_1,r_2,r_3,r_4,r_5,S_1,S_2}(3) & = & (r_3, r_1, r_2 + S_2), \\
f_{r_1,r_2,r_3,r_4,r_5,S_1,S_2}(4) & = & (r_4, r_2, r_3 + S_1 + S_2), \\
f_{r_1,r_2,r_3,r_4,r_5,S_1,S_2}(5) & = & (r_5, r_3, r_4 + S_1 + 2S_2).
\end{array}$$

Chapter 3

Extended Capabilities of Secret Sharing Schemes

In this section we will present some additional capabilities of secret sharing schemes and we will also point out their possible applications.

3.1 Multiplicative/Homomorphic Properties

The notion of *multiplicative* secret sharing schemes has been introduced by Desmedt in [43], for the threshold case. A more general definition is presented next.

Definition 3.1.1 Let $\mathcal{S}$ be the set of secrets, $\mathcal{S}_{shares}$ be the set of possible shares and let us consider an associative and commutative binary operation $\odot$ over $\mathcal{S}$. We say that an $\mathcal{A}$-secret sharing scheme $(split, combine)$ is *multiplicative with respect to* $\odot$ if for any set $A \in \mathcal{A}$ there is a family of public functions $(f_{(i,A)}|i \in A)$ from $\mathcal{S}_{shares}$ to $\mathcal{S}$ such that

$$(\forall S \in \mathcal{S})((S \leftrightarrow_{\mathcal{A}}^{(split,combine)} I_1, \ldots, I_n) \Rightarrow S = \odot_{i \in A} f_{(i,A)}(I_i)).$$

This definition has been reconsidered in [45] for the case that the operation $\odot$ may be non-commutative. In this case, for any authorized group A, a public ordering of its elements must be known.

A very important particular subclass of multiplicative secret sharing schemes is the class of *linear* secret sharing schemes [48, 49]. A linear $\mathcal{A}$-secret sharing scheme is a multiplicative secret sharing scheme in which the functions $f_{(i,A)}$ are homomorphisms (the sets of secrets and shares are structured as groups and $\odot$ is the group operation over the set of secrets).

- Shamir's scheme (denoted by `Shamir`) is linear. Indeed, in case that $S \leftrightarrow_{(k,n)}^{\mathtt{Shamir}} I_1, \ldots, I_n$, then S can be expressed as $S = \sum_{i \in A} f_{(i,A)}(I_i)$, where $f_{(i,A)} : \mathbf{Z}_q \to \mathbf{Z}_q$ is given by $f_{(i,A)}(x) = x \cdot \prod_{j \in A \setminus \{i\}} \frac{j}{j-i}$, for every authorized set A and for all $i \in A$.

- The extended Mignotte scheme (denoted by $\mathtt{ExtMig}$) is multiplicative in the following sense: if $S \leftrightarrow_{\mathcal{A}}^{\mathtt{ExtMig}} I_1, \ldots, I_n$, then the secret S can be expressed, according to Ore's algorithm (see Appendix A), as

$$S = \sum_{i \in A} f_{(i,A)}(I_i) \ \mathtt{mod}\ [\{p_i | i \in A\}],$$

where the function $f_{(i,A)} : \mathbf{N} \to \mathbf{N}$ is given by

$$f_{(i,A)}(x) = c_{(i,A)}(\alpha_{(i,A)} \ \mathtt{mod}\ [\{p_i | i \in A\}])x,$$

where

  - $c_{(i,A)} = \frac{[\{p_i | i \in A\}]}{p_i}$;
  - the numbers $\alpha_{(i,A)}$ are arbitrary integers that satisfy

$$\sum_{i \in A} \alpha_{(i,A)} c_{(i,A)} = 1,$$

for every authorized set A and for all $i \in A$.

- The extended Asmuth-Bloom scheme (denoted by $\mathtt{Ext(A-B)}$) is multiplicative in the following sense: if $S \leftrightarrow_{\mathcal{A}}^{\mathtt{Ext(A-B)}} I_1, \ldots, I_n$, then the secret S can be expressed as

$$S = (\sum_{i \in A} f_{(i,A)}(I_i) \ \mathtt{mod}\ [\{p_i | i \in A\}]) \ \mathtt{mod}\ p_0,$$

where the functions $f_{(i,A)} : \mathbf{N} \to \mathbf{N}$ are defined as above, for every authorized set A and for all $i \in A$. We have to remark that, in the last two cases, the involved binary operations over the secrets sets depend on A.

As we will see in Section 4.2 and Section 4.3.2, multiplicative secret sharing schemes can be used in designing threshold cryptographic primitives and, respectively, e-voting schemes.

Benaloh has introduced the notion of secret sharing homomorphisms in [7]. Intuitively, in a homomorphic secret sharing scheme, the compositions of the shares are shares of the composition of the secrets.

Definition 3.1.2 Let $\mathcal{S}$ and $\mathcal{S}_1, \ldots, \mathcal{S}_n$ be the set of secrets and, respectively, the set of shares corresponding to each user. Let us consider some binary operations $\oplus$ and $\otimes_1, \ldots, \otimes_n$ over these sets. We say that an $\mathcal{A}$-secret sharing scheme $(split, combine)$ is $(\oplus, \otimes_1, \ldots, \otimes_n) - homomorphic$ if for any $S_1, S_2 \in \mathcal{S}$ the relations (1) and (2) imply (3), where

$$\begin{array}{llllll}
(1) & S_1 & \leftrightarrow_{\mathcal{A}}^{(split,combine)} & I_1^1, & \ldots, & I_n^1 \\
(2) & S_2 & \leftrightarrow_{\mathcal{A}}^{(split,combine)} & I_1^2, & \ldots, & I_n^2 \\
\hline
(3) & S_1 \oplus S_2 & \leftrightarrow_{\mathcal{A}}^{(split,combine)} & I_1^1 \otimes_1 I_1^2, & \ldots, & I_n^1 \otimes_n I_n^2
\end{array}$$

In case that $\mathcal{S}_1 = \mathcal{S}_2 = \cdots = \mathcal{S}_n = \mathcal{S}_{shares}$ and $\otimes_1 = \otimes_2 = \cdots = \otimes_n = \otimes$, a $(\oplus, \otimes_1, \ldots, \otimes_n)$-homomorphic secret sharing scheme will be simply called $(\oplus, \otimes)$-homomorphic.

Remark 3.1.1 1. Frankel and Desmedt have proven in [62] that any ideal homomorphic secret sharing scheme in which the set of secrets is structured as a commutative group is multiplicative;

2. As Desmedt has remarked in [44], any multiplicative secret sharing schemes in which the functions $f_{(i,A)}$ are of form $f_{(i,A)}(x) = constant_{i,A}x$, the shares belong to a module[1], the values $constant_{i,A}$ are scalars, and the secrets belong to a submodule, is homomorphic.

- Shamir's scheme over the field $\mathbf{Z}_q$ is $(+_q, +_q)$-homomorphic. Indeed, if S_1 has the shares $(I_1^1, \ldots, I_n^1)$ and S_2 has the shares $(I_1^2, \ldots, I_n^2)$, then for any group A, $|A| = k$, $S_1 = \sum_{i \in A}(I_i^1 \cdot \prod_{j \in A \setminus \{i\}} \frac{j}{j-i})$, $S_2 = \sum_{i \in A}(I_i^2 \cdot \prod_{j \in A \setminus \{i\}} \frac{j}{j-i})$ and, thus, $S_1 +_q S_2 = \sum_{i \in A}((I_i^1 +_q I_i^2) \cdot \prod_{j \in A \setminus \{i\}} \frac{j}{j-i})$. By the same reasons, we obtain that, if $a \in \mathbf{Z}_q$ and a secret S has the shares $I_1, \ldots, I_n$ with respect to Shamir's scheme, then the elements $a \cdot_q I_1, \ldots, a \cdot_q I_n$ are the shares corresponding to the secret $a \cdot_q S$.

  We have to remark that Shamir's scheme can be easily adapted in order to become $(\cdot_p, +_q)$-homomorphic, where p is a prime such that $q|(p-1)$. Let $\alpha \in \mathbf{Z}_p$ such that $ord_p(\alpha) = q$ and let us choose $s \in \mathbf{Z}_q$ having the shares $s_1, \ldots, s_n$ with respect to Shamir's (k, n)-threshold secret sharing scheme. We may consider the secret $S = \alpha^s \texttt{ mod } p$ with the shares $I_1 = s_1, \ldots, I_n = s_n$, and the reconstruction can be performed as $S = \prod_{i \in A} \alpha^{I_i \cdot \prod_{j \in A \setminus \{i\}} \frac{j}{j-i}} \texttt{ mod } p$. It is easy to verify that this variant of Shamir's scheme is indeed $(\cdot_p, +_q)$-homomorphic. Moreover, in this case, if we choose the shares of the secret S as $I_1 = \alpha^{s_1} \texttt{ mod } p, \ldots, I_n = \alpha^{s_n} \texttt{ mod } p$, the obtained scheme is $(\cdot_p, \cdot_p)$-homomorphic (the reconstruction is performed in this case as $S = \prod_{i \in A} I_i^{\prod_{j \in A \setminus \{i\}} \frac{j}{j-i}} \texttt{ mod } p$).

[1] A *module* is a vector space in which the scalars belong to a ring instead of a field.

- The extended Mignotte scheme has homomorphic proprieties. Let $\otimes$ be a binary operation over $\mathbf{Z}$, $\otimes \in \{+, -, \cdot\}$. If $p_1, \ldots, p_n$ is an extended Mignotte sequence, then the corresponding secret sharing scheme is $(\otimes, \otimes_{p_1}, \ldots, \otimes_{p_n})$-partial homomorphic, in sense that, if S_1 and S_2 are some secrets such that $\beta < S_1 \otimes S_2 < \alpha$, with the corresponding shares $(I_1^1, \ldots, I_n^1)$, and, respectively, $(I_1^2, \ldots, I_n^2)$, then $(I_1^1 \otimes_{p_1} I_1^2, \ldots, I_n^1 \otimes_{p_n} I_n^2)$ are the shares corresponding[2] to the secret $S_1 \otimes S_2$.

- The extended Asmuth-Bloom scheme has homomorphic proprieties in case that the extended Asmuth-Bloom sequence has the property $p_0 | p_i$, for all $1 \leq i \leq n$. In this case, $p_0 | [\{p_i | i \in A\}]$, for all $A \in \mathcal{A}_{min}$ and, thus, the secret can be expressed as

$$\begin{aligned} S &= (\textstyle\sum_{i \in A} f_{(i,A)}(I_i) \ \texttt{mod} \ [\{p_i | i \in A\}]) \ \texttt{mod} \ p_0, \\ &= \textstyle\sum_{i \in A} f_{(i,A)}(I_i) \ \texttt{mod} \ p_0, \end{aligned}$$

  for some authorized group A, where the functions $f_{(i,A)}$, for $i \in A$, are presented above. Thus, any extended Asmuth-Bloom scheme based on a sequence with such properties is $(\otimes_{p_0}, \otimes_{p_0}, \ldots, \otimes_{p_0})$ -homomorphic, where $\otimes \in \{+, -\}$. Unfortunately, the property $p_0 | p_i$ also implies that

$$\begin{aligned} I_i \ \texttt{mod} \ p_0 &= ((S + \gamma \cdot p_0) \ \texttt{mod} \ p_i) \ \texttt{mod} \ p_0 \\ &= (S + \gamma \cdot p_0) \ \texttt{mod} \ p_0 \\ &= S, \end{aligned}$$

  and, thus, the security of the scheme is entirely compromised. It will be interesting to find Asmuth-Bloom sequences which lead to homomorphic proprieties, without affecting the security of the scheme. One solution would be to find $\mathcal{A}$-Asmuth-Bloom sequences $p_0, p_1, \ldots, p_n$ such that $p_0 | [\{p_i | i \in A\}]$, for all $A \in \mathcal{A}_{min}$ but $p_0 \not| p_i$, for all $1 \leq i \leq n$.

As we will see in Section 4.1 and Section 4.3.1, homomorphic secret sharing schemes can be used in designing generic secure multiparty computation protocols and, respectively, e-voting schemes.

In the end of this section we will present the notion of *homomorphisms of secret sharing scheme* as introduced by Burmester [25].

Definition 3.1.3 Let $(split_1, combine_1)$ and $(split_2, combine_2)$ be two $\mathcal{A}$-secret sharing schemes having the sets of secrets $\mathcal{S}_0^1$, respectively, $\mathcal{S}_0^2$, and the sets of shares $\mathcal{S}_1^1, \ldots, \mathcal{S}_n^1$ and, respectively, $\mathcal{S}_1^2, \ldots, \mathcal{S}_n^2$. Let us consider some functions $h_0, h_1, \ldots, h_n$, with $h_i : S_i^1 \rightarrow S_i^2$, for all $0 \leq i \leq n$. The vector

[2] This property follows directly from the properties of the congruences. More exactly, if $S_1 \equiv I_i^1 \ mod \ p_i$ and $S_2 \equiv I_i^2 \ mod \ p_i$, then $S_1 \otimes S_2 \equiv I_i^1 \otimes_{p_i} I_i^2 \ mod \ p_i$.

of functions $(h_0, h_1, \ldots, h_n)$ is a *homomorphism* from $(split_1, combine_1)$ to $(split_2, combine_2)$ if for all $S \in \mathcal{S}_0^1$

$$(S \leftrightarrow_{\mathcal{A}}^{(split_1, combine_1)} I_1, \ldots, I_n) \Rightarrow (h_0(S) \leftrightarrow_{\mathcal{A}}^{(split_2, combine_2)} h_1(I_1), \ldots, h_n(I_n)).$$

It is interesting to remark that any $(\oplus, \otimes_1, \ldots, \otimes_n)$-homomorphic secret sharing scheme $(split, combine)$ corresponds to a homomorphism from[3] $(split, combine) \times (split, combine)$ to $(split, combine)$ given by $h_0((S_1, S_2)) = S_1 \oplus S_2$ and $h_i((I_i^1, I_i^2)) = I_i^1 \otimes_i I_i^2$, for all $1 \leq i \leq n$.

Burmester has pointed out that the homomorphisms of secret sharing schemes can be used for developing verifiable signature sharing schemes. A verifiable signature sharing scheme enables the distribution of a digital signature among some proxies such that each proxy can verify whether a valid signature can be reconstructed later, without revealing the signature itself until it is reconstructed. Verifiable signature sharing can be used, for instance, in auction schemes (see, for example, [63]).

3.2 Dealer-Free Secret Sharing Schemes

In this section we present some methods in which certain secret sharing schemes can be configured without the presence of a dealer. In case of dealer-free secret sharing, the secret will be considered implicit.

Meadows was the first to discuss the dealer-free secret sharing in [113] for the threshold case. In this paper, the first k users generates their own shares and the rest are generated using a black box which, as remarked by Jackson, Martin, and O'Keefe in [94], plays the role of a mutually trusted party. Thus, although this paper does not present a real dealer-free secret sharing scheme, it is the first paper which have considered secret sharing without a dealer.

As it has been remarked in [94], the unanimous consent structure of rank n can be realized by the next dealer-free scheme:

- Every participant chooses his share I_i as a random number from $\mathbf{Z}_m$;
- The secret S is generated (and can be reconstructed) as $S = \sum_{i=1}^{n} I_i \bmod m$.

Ingemarsson and Simmons proposed an elegant scheme for dealer-free threshold secret sharing in [88]. In this scheme, the i^{th} user first chooses an arbitrary element S_i that will be the share of some secret S with respect to a unanimous secret sharing scheme of rank n and then the element S_i is shared among the rest of users. The next example illustrates their idea.

[3] The *product* of two secret sharing schemes over the same access structure is defined as follows. If, for $i \in \{1, 2\}$, a secret S_i has the shares $I_1^i, \ldots, I_n^i$ with respect to the i^{th} secret sharing scheme, then the secret (S_1, S_2) will have the shares $(I_1^1, I_1^2), \ldots, (I_n^1, I_n^2)$ with respect to the product secret sharing scheme.

Example 3.2.1 (A dealer-free $(2,3)$-threshold secret sharing scheme)

- Each participant chooses a random number $S_i \in \mathbf{Z}_m$;
- The secret S is chosen as $S = \sum_{i=1}^{3} S_i \texttt{ mod } m$;
- The i^{th} user chooses some integers I_i^j such that $S_i = \sum_{j \neq i} I_i^j \texttt{ mod } m$ and distributes I_i^j to the j^{th} user, for all $1 \leq i \leq 3$ and $1 \leq j \leq 3$, $j \neq i$;
- The share I_i is chosen as $I_i = (S_i, (I_j^i | j \in \{1,2,3\} \setminus \{i\}))$ for all $i \in \{1,2,3\}$;
- Having two shares, for example I_1 and I_2, the secret S can be reconstructed as $S = S_1 + S_2 + I_3^1 + I_3^2 \texttt{ mod } m$.

Jackson, Martin, and O'Keefe [94] have extended the previous technique in order to construct a dealer-free $\mathcal{A}$-secret sharing scheme as follows:

- Use a unanimous secret sharing scheme of rank a for constructing the shares $S_1, \ldots, S_a$ corresponding to the secret S, for some $1 \leq a \leq n$;
- The i^{th} user constructs the shares of S_i with respect to an appropriate $\mathcal{A}_i$-secret sharing scheme and distributes these shares to the users included in $\mathcal{A}_i$, for all $1 \leq i \leq a$.

They have first proposed using $a = n$ and $\mathcal{A}_i = cl(\{\{i\}\} \cup \mathcal{A})$, for all $1 \leq i \leq n$, and then they have proposed a more efficient method based on the contraction construction (see Section 1.1). More exactly, they have proposed using $\mathcal{A}_1 = cl(\{\{1\}\} + \mathcal{A})$, $\mathcal{A}_2 = cl(\{\{2\}\} + \mathcal{A} \cdot \{1\})$,..., $\mathcal{A}_a = cl(\{\{a\}\} + \mathcal{A} \cdot \{1, 2, \ldots, a-1\})$, for some $1 \leq a \leq n$. The next example illustrates this construction.

Example 3.2.2 (A dealer-free $(3,4)$-threshold secret sharing scheme) Let $\mathcal{A}_{min} = \{\{1,2,3\},\{1,2,4\},\{1,3,4\},\{2,3,4\}\}$ and consider $a = 3$. We obtain $\mathcal{A}_{1min} = \{\{1\},\{2,3,4\}\}$, $\mathcal{A}_{2min} = \{\{2\},\{3,4\}\}$, and $\mathcal{A}_{3min} = \{\{3\},\{4\}\}$. Thus, the first user chooses S_1 and construct the shares $S_1, S_{1,2}, S_{1,3}, S_{1,4}$ of the secret S_1 with respect to an $\mathcal{A}_1$-secret sharing scheme and distributes the elements $S_{1,j}$ to the j^{th} user, for $j \in \{2,3,4\}$. The second user chooses S_2 and constructs the shares $S_2, S_{2,3}, S_{2,4}$ of S_2 with respect to an $\mathcal{A}_2$-secret sharing scheme and distributes the elements $S_{2,j}$ to the j^{th} user, for $j \in \{3,4\}$. Finally, the third user chooses S_3 and constructs the shares $S_3, S_{3,4}$ of S_3 with respect to an $\mathcal{A}_3$-secret sharing scheme and distributes the element $S_{3,4}$ to the fourth user. The secret will be $S = S_1 + S_2 + S_3$ and the corresponding shares are $I_1 = S_1$, $I_2 = (S_2, S_{1,2})$, $I_3 = (S_3, S_{1,3}, S_{2,3})$, and $I_4 = (S_{1,4}, S_{2,4}, S_{3,4})$. Every group of three participants can recover S. For example, having I_2, I_3, I_4, the elements S_2 and S_3 can be directly obtained, whereas S_1 can be obtained from $S_{1,2}, S_{1,3}, S_{1,4}$.

In the same paper, Jackson, Martin, and O'Keefe have remarked that any $(\oplus, \otimes)$-homomorphic $\mathcal{A}$-secret sharing scheme (see Section 3.1) can be used to construct a dealer-free $\mathcal{A}$-secret sharing scheme. They have also remarked that the previous definitions of homomorphic secret sharing scheme do not assure perfect secrecy and introduced the *perfect homomorphic* secret sharing schemes in which the compositions of the shares of any unauthorized group do not provide any information about the composition of the secrets.

- The i^{th} participant chooses an element S_i and constructs, using a perfect $(\oplus, \otimes)$-homomorphic secret sharing scheme, the shares $I_i^1, \ldots, I_i^n$ corresponding to the secret S_i and securely distributes I_i^j to the j^{th} participant, for all $1 \leq j \leq n$, $j \neq i$;
- The secret S will be $S = \oplus_{i=1}^n S_i$;
- The shares are chosen as $I_i = \otimes_{j=1}^n I_j^i$, $1 \leq i \leq n$.

3.3 Verifiability

The secret sharing schemes presented in Chapter 2 assume that the parties involved behave honestly. In this section we discuss some solutions for the case in which the dealer or some users may behave maliciously. The case of a possible dishonest dealer has been discussed for the first time by Chor, Goldwasser, Micali, and Awerbuch [31], who have introduced the notion of *verifiable secret sharing schemes* in which every user can verify that he has received a valid share. The problem of cheating in the reconstruction phase has been discussed by McEliece and Sarwate [112], and later on, by Tompa and Wool [149]. As Schoenmakers has remarked in [133], verifiable secret sharing can also be seen as a solution for the problem of cheating - the shares presented in the reconstruction phase may be verified with respect to the distribution phase.

3.3.1 Feldman's Scheme

The scheme of Chor, Goldwasser, Micali, and Awerbuch [31] has a great disadvantage - it is *interactive*, i.e., some interaction between users is required in order to verify the consistency of the shares. Moreover, the communication complexity in their scheme is exponential. Feldman [56] has proposed a non-interactive scheme for achieving verifiability in Shamir's threshold secret sharing scheme. The main idea is to use a one-way function f such that $f(x+y) = f(x) \cdot f(y)$ (it can be proven by induction that $f(ix) = f(x)^i$, for any element x and natural number i) and to broadcast $f(a_0), \ldots, f(a_{k-1})$, where $P(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1}$ is the polynomial used in Shamir's scheme. The consistency of the share $I_i = P(i)$ can be tested by verifying that

$$f(I_i) \stackrel{?}{=} f(a_0) \cdot f(a_1)^{i^1} \cdots f(a_{k-1})^{i^{k-1}}.$$

Indeed, by the homomorphic property of the function f,

$$f(a_0 + a_1 i + \cdots + a_{k-1} i^{k-1}) = f(a_0) \cdot f(a_1)^{i^1} \cdots f(a_{k-1})^{i^{k-1}}.$$

A good candidate for the function f is $f : \mathbf{Z}_q \to \mathbf{Z}_p$, $f(x) = \alpha^x \texttt{ mod } p$, where p and q are odd primes such that $q|(p-1)$, and $\alpha \in \mathbf{Z}_p^*$ is an element of order q. In this case we obtain the following scheme:

- There are generated the primes p and q such that $q|(p-1)$, and $\alpha \in \mathbf{Z}_p^*$ an element of order q. All these numbers are public;
- The dealer generates the polynomial $P(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1}$ over $\mathbf{Z}_q$ such that $a_0 = S$ and makes public $\alpha_i = \alpha^{a_i} \texttt{ mod } p$, for all $0 \le i \le k-1$;
- The dealer securely distributes the share $I_i = P(i)$ to the i^{th} user, for all $1 \le i \le n$;
- Each user can verify the correctness of the received share I_i by testing

$$\alpha^{I_i} \texttt{ mod } p \stackrel{?}{=} \prod_{j=0}^{k-1} \alpha_j^{i^j} \texttt{ mod } p.$$

A similar scheme has been presented in [124].

3.3.2 Pedersen's Scheme

Feldman's scheme has the disadvantage that $f(S)$ is broadcasted and, thus, the privacy of the secret depends on the hardness of inverting f. Pedersen [126] has proposed the following non-interactive and information-theoretic secure verifiable variant of Shamir's threshold secret sharing scheme:

- There are generated the primes p and q such that $q|(p-1)$, and $g, h \in \mathbf{Z}_p^*$ elements of order q. All these numbers are public;
- The dealer broadcasts $E_0 = g^S h^t \texttt{ mod } p$, where $t \in \mathbf{Z}_q$;
- The dealer generates the polynomials $P(x) = S + P_1 x + \cdots + P_{k-1} x^{k-1}$ and $Q(x) = t + Q_1 x + \cdots + Q_{k-1} x^{k-1}$ over $\mathbf{Z}_q$ and makes public $E_i = g^{P_i} h^{Q_i} \texttt{ mod } p$, for all $1 \le i \le k-1$;
- The dealer securely distributes $I_i = (P(i), Q(i))$ to the i^{th} user, for all $1 \le i \le n$;

- Each user can verify the correctness of the received share $I_i = (s_i, t_i)$ by testing
$$g^{s_i} h^{t_i} \mathtt{mod}\ p \overset{?}{=} \prod_{j=0}^{k-1} E_j^{ij} \mathtt{mod}\ p.$$

This scheme has also the advantage that it is easy to derive a verifiable sharing for a linear combination of some secrets. More exactly, it is easy to verify that if S_1 and S_2 have the shares $(s_{i,1}, t_{i,1})$, and, respectively, $(s_{i,2}, t_{i,2})$, for $1 \leq i \leq n$, then $S_1 +_q S_2$ has the shares $(s_{i,1} +_q s_{i,2}, t_{i,1} +_q t_{i,2})$, for $1 \leq i \leq n$. Moreover, if S has the shares $(s_1, t_1), \ldots, (s_n, t_n)$, then $a \cdot_q S$ has the shares $(a \cdot_q s_1, a \cdot_q t_1), \ldots, (a \cdot_q s_n, a \cdot_q t_n)$, for any $a \in \mathbf{Z}_p$. This property can be used for assuring verifiability for some dealer-free threshold cryptosystems (see Section 4.2.1) or in the case of e-voting schemes based on homomorphic secret sharing schemes (see Section 4.3.1).

3.3.3 Verifiability for Schemes based on Chinese Remainder Theorem

In [84] we have indicated how to achieve verifiability for the extended Mignotte scheme. We have used that, if $I_1, \ldots, I_n$ represent correct shares of a secret S, then $S \equiv I_i\ mod\ p_i$, for all $1 \leq i \leq n$. Such a congruence is equivalent with $\alpha_i^S \equiv \alpha_i^{I_i}\ mod\ m_i$, for any positive integer m_i and any element $\alpha_i \in \mathbf{Z}_{m_i}^*$ of order p_i (p_i must be a divisor of $\phi(m_i)$).

The administrator makes public the values $(m_i, \alpha_i, \alpha_{S,i})$, where $\alpha_{S,i} = \alpha_i^S \mathtt{mod}\ m_i$, for all $1 \leq i \leq n$, and securely sends $I_1, \ldots, I_n$ to users.

The i^{th} user, after receiving I_i, can verify that his share is correct by computing $\alpha_i^{I_i} \mathtt{mod}\ m_i$ and comparing the result with $\alpha_{S,i}$. Moreover, the integrity of the shares is also assured. The security of this feature is based on the intractability of the discrete logarithm problem.

A non-interactive modular verifiable secret sharing scheme has been proposed in [102] for the case of the Asmuth-Bloom scheme.

3.3.4 Publicly Verifiable Secret Sharing

Stadler [140] has introduced[4] the notion of *publicly verifiable* secret sharing schemes. In these schemes, the correctness of the shares with respect to the secret can be verified by everyone, not only by the participants.

In a publicly verifiable secret sharing scheme $(split, combine)$ the shares corresponding to some secret S are encrypted by the dealer, using the public keys of the shareholders, and the encrypted shares $ES_i = e_{k_i}(I_i)$, for $1 \leq i \leq n$, are broadcasted. A public algorithm $PubVerify$ is provided for testing

[4] In fact, as noted in [140], the scheme described in [31] has already achieved public verifiability, but the later ones ([56], [124], [126]) have not been designed for public verifiability.

the validity of the encrypted shares such that, if a set of encrypted shares passes the $PubVerify$ test, then their decryptions will lead to the correct secret:

$$(\exists S' \in \mathcal{S})(\forall A \in \mathcal{A}_{min})$$

$$(PubVerify(\{ES_i | i \in A\}) = 1 \Rightarrow ((A, (I_i | i \in A)), S') \in combine)$$

and $S' = S$ *if dealer was honest.*

$\mathbf{DLEQ}(\alpha_1, \mathbf{y_1}, \alpha_2, \mathbf{y_2})$

The Prover		The Verifier
Choose $r \in \mathbf{Z}_q^*$ Compute $\alpha_{r,1} = \alpha_1^r$ Compute $\alpha_{r,2} = \alpha_2^r$		
	$\xrightarrow{\alpha_{r,1}, \alpha_{r,2}}$	
		Choose $c \in \mathbf{Z}_q^*$
	$\xleftarrow{c}$	
Compute $s = r - c \cdot x \texttt{ mod } q$		
	$\xrightarrow{s}$	
		Test $\alpha_{r,1} \overset{?}{=} \alpha_1^s \cdot y_1^c$ Test $\alpha_{r,2} \overset{?}{=} \alpha_2^s \cdot y_2^c$

Figure 3.1: Chaum-Pedersen protocol for proving $log_{\alpha_1} y_1 = log_{\alpha_2} y_2 (= x)$

We present next a simpler and more efficient publicly verifiable secret sharing scheme due to Schoenmakers [133]. Schoenmakers' scheme is based on a proof that $log_{\alpha_1} y_1 = log_{\alpha_2} y_2$, where $\alpha_1, \alpha_2, y_1, y_2 \in \mathbf{G}$, for some group $\mathbf{G}$ of prime order q (the proof of knowing $x \in \mathbf{Z}_q^*$ such that $y_1 = \alpha_1^x$ and $y_2 = \alpha_2^x$). Such a proof, denoted by $DLEQ(\alpha_1, y_1, \alpha_2, y_2)$, due to Chaum and Pedersen [30], is presented in Figure 3.1. This interactive proof can be transformed into a non-interactive one, using Fiat-Shamir technique [57].

The rest of the scheme is described next.

1. **Setup Phase**
   - There are chosen the group $\mathbf{G}$ of prime order q, and some generators α, β of $\mathbf{G}$; these elements are broadcasted;
   - The i^{th} user generates a secret key $t_i \in \mathbf{Z}_q^*$ and broadcasts $y_i = \beta^{t_i}$ as his public key;
2. **Distribution Phase**

   - The dealer generates the polynomial $P(x) = a_0 + a_1 x + \cdots + a_{k-1}x^{k-1}$ of degree at most $k-1$ over $\mathbf{Z}_q^*$ such that $a_0 = s$, for some $s \in \mathbf{Z}_q^*$, and makes public $\alpha_i = \alpha^{a_i}$, for all $0 \leq i \leq k-1$;
   - The secret will be $S = \beta^s$;
   - The dealer broadcasts the encrypted shares $Y_i = {y_i}^{P(i)}$, for all $1 \leq i \leq n$;
   - The dealer demonstrates the correctness of the encrypted shares by showing a proof of knowledge of the value $P(i)$ such that $X_i = \alpha^{P(i)}$ and $Y_i = y_i^{P(i)}$, where $X_i = \prod_{j=0}^{k-1} \alpha_j^{i^j}$, for all $1 \leq i \leq n$; more exactly, the dealer broadcasts the concatenation of the non-interactive versions of $DLEQ(\alpha, X_i, y_i, Y_i)$, for $1 \leq i \leq n$;

3. **Reconstruction Phase**
   - Using his private key, the i^{th} user finds the share $I_i = \beta^{P(i)}$ as $I_i = Y_i^{t_i^{-1} \bmod q}$;
   - A group A with $|A| = k$ can recover the secret as
   $$S = \prod_{i \in A} I_i^{\prod_{j \in A \setminus \{i\}} \frac{j}{j-i}}.$$

   In the reconstruction phase, the correctness of the share I_i can be shown by adding a proof of the knowledge of the value t_i such that $y_i = \beta^{t_i}$ and $Y_i = I_i^{t_i}$.

Tang, Pei, Liu, and He [143] have proposed replacing Feldman's verifiability feature with Pedersen's one in order to obtain a non-interactive and information-theoretic secure publicly verifiable secret sharing scheme. They have presented a protocol for proving the knowledge of x_1, x_2 such that $X = g_1^{x_1} g_2^{x_2}$ and $Y = h_1^{x_1} h_2^{x_2}$. This protocol, denoted by, $DLEQ(X, Y, g_1, g_2, h_1, h_2)$, which is an extension of Chaum-Pedersen protocol, has been then used for replacing $DLEQ(\alpha_1, y_1, \alpha_2, y_2)$ in Schoenmakers' scheme.

3.3.5 Detection/Identification of Cheaters

Dealing with a possible malicious behavior of some users in the reconstruction phase has two aspects:

- *detection of cheaters* - when the frauds are detected but not the parties involved. One simple solution for the detection of cheaters is that, as suggested by Karnin, Greene, and Hellman in [97], the dealer uses a one-way function f and broadcasts $f(S)$. The digest of the output of the reconstruction algorithm can be compared with the public digest and, thus, any fraud can be detected. The resulted scheme depends on the strength of the one-way function f;

- *identification of cheaters* - when the authors of the frauds are detected. The dealer may also broadcast the digests of the shares $f(I_1), \ldots, f(I_n)$. In case of detecting a fraud, the author(s) can be identified by using the public digests. The resulted scheme depends on the strength of the one-way function f.

The term *robust* will be used for the schemes that consider and solve the problem of cheaters.

Tompa and Woll [149] have remarked that in case of Shamir's scheme, a user can affect the reconstruction process such that the correct secret can be recovered only by him. Let us consider Shamir's (k, n)-threshold secret sharing scheme with the secret S and the shares $I_1, \ldots, I_n$. Let A be a minimal authorized group and consider some fixed user $j \in A$. The user j can construct, using Lagrange interpolation formula, a polynomial $Q(x)$ of degree at most $k-1$ such that $Q(0) = a$, where a is an element chosen by the user j, and $Q(x_i) = 0$, for all $i \in A$, $i \neq j$ (the values x_i are public). By the $(+,+)$-homomorphic property of Shamir's scheme, $\{I_i | i \in A \setminus \{j\}\} \cup \{I_j + Q(x_j)\}$ will be the shares of the users from A corresponding to the secret $S + a$. Thus, if the user j enters the false share $I_j + Q(x_j)$ instead of I_j, the reconstruction algorithm applied to the shares of the users from A will output $S + a$ instead of the correct secret S. Only user j knows a and, thus, the correct secret can be recovered only by him. Tompa and Woll have proposed that the shares be chosen as $I_i = (x_i, P(x_i))$, which will solve the presented problem, with the cost of doubling the size of each share.

We present a very simple method of detecting cheating in Shamir's secret sharing scheme, method due to Ogata, Kurosawa, and Stinson [120]. Their scheme is based on *planar difference sets*. A planar difference set modulo $p = q(q-1)+1$ is a set of q positive integers $\{t_1, t_2, \ldots, t_q\} \subset \mathbf{Z}_p$ such that

$$\{t_i -_p t_j | 1 \leq i \neq j \leq q\} = \{1, 2, \ldots, p-1\}$$

(i.e., for any element s, $1 \leq s \leq p-1$, s can be uniquely written as $s = t_i -_p t_j$, $1 \leq i \neq j \leq q$).

Let $\{t_1, t_2, \ldots, t_q\}$ be a public planar difference set modulo $p = q(q-1)+1$ such that p is a prime. The secrets set is chosen as $\mathcal{S} = \{t_1, t_2, \ldots, t_q\}$. For sharing a secret $S \in \mathcal{S}$, the dealer chooses a polynomial $P(x)$ of degree at most $k-1$ over $\mathbf{Z}_p$ such that $P(0) = S$ and securely distributes $I_i = P(i)$ to the i^{th} user. In the reconstruction phase, Lagrange's formula applied to the shares of a group A, $|A| = k$, leads to $S = \sum_{i \in A}(I_i \cdot \prod_{j \in A \setminus \{i\}} \frac{j}{j-i})$. If the obtained value S is not in the set $\{t_1, t_2, \ldots, t_q\}$, then at least one participant must cheat. From the properties of the planar difference set, it can be shown that the probability of cheating success is $\frac{1}{q}$.

3.4 Changeability

In this section we will discuss some possible solutions for the situations in which an explicit secret must be shared according to a new authorized access structure. The access structure designed to protect a certain secret may change during the lifetime of a secret for several reasons:

- The share of a user may be invalided because he must leave the organization, his share has been stolen or lost, or for other security reasons (*disenrollment*);
- A new person may enter the organization and need to be included in some access groups (*enrollment*);
- The entire security policy may change (e.g., for the case of threshold secret sharing, we may raise the problem of *threshold changeability* in case that the threshold must be decreased or increased, reflecting the degree of mutual trust).

If the succession of the access structures is known in advance, the dealer may act accordingly, by distributing (or preparing somehow) in advance the shares corresponding to the predicted access structures. In case that the new access structure is not known in advance, the simplest and the most obvious solution is that the dealer sets up a new secret sharing scheme corresponding to the new access structure. This solution, besides inefficiency, has also other major drawbacks. For example, the dealer may no longer be active. In this case, a *secret redistribution* [49, 109] may be performed by the participants from the initial access structure. This solution involves, in general, communication over secure channels, feature that may be also impractical.

We will present next the some interesting schemes which provide changeability features.

Desmedt-Jajodia Redistribution Scheme

Desmedt and Jajodia [49] have indicated how to transform the shares of a secret S with respect to the access structure $\mathcal{A}$ over $\{1, 2, \ldots, n\}$ into shares of the same secret corresponding to the access structure $\mathcal{A}'$ over $\{1, 2, \ldots, n'\}$ without recovering the secret in the process. Their technique is based on linear secret sharing schemes (see Section 3.1). The technique is described next:

- Suppose that $I_1, \ldots, I_n$ are the shares of a secret S with respect to a linear $\mathcal{A}$-secret sharing scheme based on the homomorphisms $\psi_{i,A}$, for $A \in \mathcal{A}$, $i \in A$;

- The i^{th} user derives the shares $s_{i,1},\ldots,s_{i,n'}$ of the secret I_i with respect to a linear $\mathcal{A}'$-secret sharing scheme based on the homomorphisms $\psi_{i,(j,A')}$, for $A' \in \mathcal{A}'$, $j \in A'$, for all $1 \leq i \leq n$; the i^{th} user securely sends $s_{i,j}$ to the j^{th} new user;
- Suppose that, for any $A \in \mathcal{A}$, $i \in A$, and for any $A' \in \mathcal{A}'$, $j \in A'$ there are some appropriate homomorphisms $\psi'_{j,A'}$ and $\psi'_{j,(i,A)}$ such that
$$\psi_{i,A} \circ \psi_{i,(j,A')} = \psi'_{j,A'} \circ \psi'_{j,(i,A)};$$
- The j^{th} new user, after receiving $s_{i,j}$, for all $i \in A$ (for some fixed $A \in \mathcal{A}$) can compute his share as
$$I'_j = \sum_{i \in A} \psi'_{j,(i,A)}(s_{i,j}),$$
for all $1 \leq j \leq n'$.

The correctness of this technique can be easily justified as follows:

$$\begin{array}{rcll}
S & = & \sum_{i\in A} \psi_{i,A}(I_i) & \\
 & = & \sum_{i\in A} \psi_{i,A}(\sum_{j\in A'} \psi_{i,(j,A')}(s_{i,j})) & \\
 & = & \sum_{i\in A} \sum_{j\in A'} \psi_{i,A}(\psi_{i,(j,A')}(s_{i,j})) & (\psi_{i,A} \text{ is homomorphism}) \\
 & = & \sum_{j\in A'} \sum_{i\in A} \psi_{i,A}(\psi_{i,(j,A')}(s_{i,j})) & (+ \text{ is commutative}) \\
 & = & \sum_{j\in A'} \sum_{i\in A} \psi'_{j,A'}(\psi'_{j,(i,A)}(s_{i,j})) & (\psi_{i,A} \circ \psi_{i,(j,A')} = \psi'_{j,A'} \circ \psi'_{j,(i,A)}) \\
 & = & \sum_{j\in A'} \psi'_{j,A'}(\sum_{i\in A} \psi'_{j,(i,A)}(s_{i,j})) & (\psi'_{j,A'} \text{ is homomorphism}) \\
 & = & \sum_{j\in A'} \psi'_{j,A'}(I'_j) & (\text{by the definition of } I'_j)
\end{array}$$

We obtain that $I'_1,\ldots,I'_{n'}$ are indeed the shares corresponding to the secret S with respect to the access structure $\mathcal{A}'$ and, thus, any fixed group $A \in \mathcal{A}$ can complete the redistribution process, without recovering S.

Proactivity

An important particular case for secret redistribution is when $\mathcal{A}' = \mathcal{A}$. This case reduces to renewing the shadows without changing the secret. This feature is referred to as *proactivity* in [76] and its periodic appliance has the advantage that, even if an adversary obtains some shares, he cannot obtain any useful information about the secret after the shares are renewed.

Threshold Changeability

Martin, Pieprzyk, Safavi-Naini, and Wang have discussed the case of changing thresholds in the absence of secure channels in [107]. They have introduced the notion of *threshold changeable* threshold secret schemes as follows:

Definition 3.4.1 An (k,n)-threshold secret sharing scheme $(split, combine)$, with $split : \mathcal{S} \to \mathcal{S}_1 \times \cdots \times \mathcal{S}_n$, is called *threshold changeable to* k' if there are publicly known functions $h_i : \mathcal{S}_i \rightarrow \mathcal{S}'_i$ and *combine'* such that[5] $((h_1 \times \cdots \times h_n) \circ split, combine')$ is an (k', n)-threshold secret sharing scheme (possibly not perfect).

Their first proposal starts with a (v, kv, nv)-ramp scheme (see Section 2.8), where v is an arbitrary positive integer, that is then transformed to an (k, n)-threshold secret sharing scheme threshold changeable to k', for any $k \leq k'$ such that $k'|kv$. More exactly, starting with a (v, kv, nv)-ramp scheme with the secret S and the shares $s_1, \ldots, s_{nv}$, they have constructed the shares corresponding to the secret S with respect to a perfect (k, n)-threshold access structure as $I_i = \{s_j | j \in P_i\}$, where $\{P_1, \ldots, P_n\}$ is a partition of the set $\{1, \ldots, nv\}$ such that $|P_i| = v$, for all $1 \leq i \leq n$. Let us consider some functions h_i given by $h(I_i) = I'_i$, where I'_i is an arbitrary subset of I_i such that $|I'_i| = \frac{kv}{k'}$. The union of any shares $I'_{i_1}, \ldots, I'_{i_{k'}}$ will include at least kv shares of the initial ramp scheme and, thus, the secret can be recovered.

They have also described a geometrical construction for an (k, n)-threshold secret sharing scheme threshold changeable to k'. We will present next an $(2, n)$-threshold secret sharing scheme threshold changeable to 3. The starting point is the following $(2, 3, n)$-ramp scheme:

- The secret S is a line contained in a publicly known plane Π;
- Having an arbitrary plane Π_1 that intersects Π in S, the shares are generated as any n points on Π_1, but not on S, such that any three points are non-collinear.

Any three shares are sufficient for determining Π_1 and the secret S is obtained as $S = \Pi \cap \Pi_1$. Having only two shares, we can obtain only a line on Π_1, line that intersects Π in a point that is situated on line S. Having only one share, no information about the secret is obtained.

This scheme can be used for constructing a perfect $(2, n)$-threshold secret sharing scheme as follows. We use two instances of the presented $(2, 3, n)$-ramp scheme for the same publicly known plane Π, the first based on the plane Π_1, the second based on the plane Π_2, where $\Pi_1 \neq \Pi_2$ and $\Pi_1 \cap \Pi_2 \cap \Pi = S$. The shares will be chosen as pairs of points, in which the first component is from Π_1, and the second component is from Π_2. If these points are carefully chosen, the lines PP' and QQ', obtained from any two shares (P, Q) and (P', Q'), intersect Π in two different points, points that are also situated on S, leading thus to the secret.

For the threshold changeability feature, the authors considered the functions h_i, $1 \leq i \leq n$, where h_i simply extracts the first component, corresponding to the point on the plane Π_1, from these pairs. They have also indicated how to extend this scheme to the general case.

[5] The function *split'* is given by $split'(S) = (h_1(I_1), \ldots, h_n(I_n))$, where $(I_1, \ldots, I_n) = split(S)$.

Disenrollment

Secret sharing schemes which tolerate the invalidation of some shares have been discussed independently by Blakley, Blakley, Chan, and Massey [11] and by Martin [106]. The term *disenrollment capability* has been introduced in [11] for categorizing this feature.

We will present next the threshold secret sharing scheme with disenrollment capability due to Charnes, Pieprzyk, and Safavi-Naini [29]. This scheme combines Shamir's secret sharing scheme with exponentiation in Galois fields.

- There are generated the primes p and q such that $q|(p-1)$, and $\alpha \in \mathbf{Z}_p^*$ an element of order q. All these numbers are public;
- The dealer generates the polynomial $P(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1}$ over $\mathbf{Z}_q$ and securely distributes the *initial condition* $c_i = P(i)$ to the i^{th} user, for all $1 \leq i \leq n$;
- The secret will be $S = \alpha^{P(0)} \texttt{ mod } p$ and every user computes his *share* as $I_i = \alpha^{c_i} \texttt{ mod } p$, for all $1 \leq i \leq n$;
- Having $\{I_i | i \in A\}$, $|A| = k$, the combiner can obtain the secret as

$$S = \prod_{i \in A} I_i^{\prod_{j \in A \setminus \{i\}} \frac{j}{j-i}} \texttt{ mod } p.$$

The disenrollment feature can be implemented as follows:

- The combiner is notified that a share must be invalidated. From this moment, the combiner ignores any request of reconstructing the secret;
- The combiner generates β, a new primitive root modulo p, by choosing a parameter $r \in \mathbf{Z}_q^*$ and considering $\beta = \alpha^r \texttt{ mod } p$; the combiner broadcasts the value β while the value r is kept secret;
- Any group A of k users who want to reconstruct the secret compute their *new shares* as $I_i' = \beta^{c_i} \texttt{ mod } p = \alpha^{r \cdot c_i} \texttt{ mod } p$ and send them to the combiner;
- The combiner can obtain the secret as

$$S = (\prod_{i \in A} I_i'^{\prod_{j \in A \setminus \{i\}} \frac{j}{j-i}} \texttt{ mod } p)^{r^{-1} \texttt{ mod } q} \texttt{ mod } p.$$

Moreover, the correctness of the initial conditions can be tested by combining the presented scheme with Feldman's or Pedersen's technique for verifiable secret sharing (see Section 3.3).

3.5 Multi-Secret Sharing

There are situations in which many secrets need to be shared, possibly each with respect to a different access structure. More exactly, let us consider some access structures $\mathcal{A}_1, \ldots, \mathcal{A}_m$ over the set $\{1, 2, \ldots, n\}$. Informally, an $(A_1, \ldots, \mathcal{A}_m)$-*multi-secret sharing scheme* is a method of generating $((S_1, \ldots, S_m), (I_1, \ldots, I_n))$ such that

- for any $j \in \{1, 2, \ldots, m\}$ and for any $A \in \mathcal{A}_j$, the problem of finding the element S_j, given the set $\{I_i \mid i \in A\}$, is "easy";

- for any $j \in \{1, 2, \ldots, m\}$ and for any $A \in \overline{\mathcal{A}_j}$, the problem of finding the element S_j, given the set $\{I_i \mid i \in A\}$, is "hard".

The trivial solution in this case is to use a secret sharing scheme for each access structure. More exactly, if we generate, for each $j \in \{1, 2, \ldots, m\}$, the shares $I_1^j, \ldots, I_n^j$ corresponding to the secret S_j with respect to an $\mathcal{A}_j$-secret sharing scheme, we may choose $I_i = (I_i^j | 1 \leq j \leq m)$ as the shares of $(S_1, \ldots, S_m)$ with respect to the $(A_1, \ldots, \mathcal{A}_m)$-multi-secret sharing scheme.

A natural question is that if this construction is the best possible. An interesting observation of Moti Young, presented by Di Crescenzo in [36], categorizes the investigation of multi-secret sharing as the study of a particular case of the *direct product problem* - deciding if complexity of an algorithm solving m instances for a given problem can be smaller than the complexity of the algorithm which solves the instances independently, one by one.

Some improvements, with respect to the worst-case/average information rate, have been obtained for some particular multi-access structures. We present here one simple example, due to Di Crescenzo [36].

Example 3.5.1 Let us consider $m = 2$, $\mathcal{A}_1 = \{\{1,2\}, \{1,3\}, \{1,2,3\}\}$, and $\mathcal{A}_2 = \{\{1,2\}, \{2,3\}, \{1,2,3\}\}$. For $j \in \{1,2\}$, a simple ideal $\mathcal{A}_j$-secret sharing scheme can be constructed as follows. The shares corresponding to a secret $S_j \in \mathbf{GF}_{2^l}$ can be constructed as $I_j^j = a_j \oplus S_j$, $I_{3-j}^j = a_j$, and $I_3^j = a_j$, where a_j is a random element of $\mathbf{GF}_{2^l}$. In the reconstruction phase, the secrets can be obtained by simple XOR operations. The average share size in this case is $H(\mathtt{I}_1) + H(\mathtt{I}_2) + H(\mathtt{I}_3) = 6H(\mathtt{S}_1)$ (for simplicity, we have supposed that $H(\mathtt{S}_1) = H(\mathtt{S}_2)$).

A more efficient multi-secret sharing scheme can be constructed as follows. The shares corresponding to the multi-secret (S_1, S_2) with respect to $(\mathcal{A}_1, \mathcal{A}_2)$ can be defined as $I_1 = (a \oplus S_1, b)$, $I_2 = (b \oplus S_2, a)$, and $I_3 = a \oplus b$, where a, b are random elements of $\mathbf{GF}_{2^l}$. In the reconstruction phase, the multi-secret can be obtained by simple XOR operations. In this case, the average share size is $5H(\mathtt{S}_1)$.

Multi-stage Secret Sharing

In this section we will discuss sharing many secrets with respect to a fixed access structure such that the reconstruction is made stage-by-stage, in some predetermined order, and every user keeps only one secret share.

He and Dawson [75] have proposed a multi-stage (k, n)-threshold secret sharing scheme for sharing the secrets $S_1, \ldots, S_m$ as follows.

- Setup (performed by the dealer)
  - Choose a large prime p and randomly generate $I_1, \ldots, I_n \in \mathbf{Z}_p$;
  - For all $1 \leq j \leq m$ do
    - Choose a polynomial $P_j(x)$ of degree at most $k-1$ over $\mathbf{Z}_p$ such that $P_j(0) = S_j$;
    - For all $1 \leq i \leq n$ compute $T_{i,j} = P_j(i) - f^{j-1}(I_i)$, where f is an arbitrary one-way function, $f : \mathbf{Z}_p \to \mathbf{Z}_p$, $f^0(x) = x$ and $f^{l+1}(x) = f(f^l(x))$, for any $l \geq 0$;
  - Secretly send I_i to the i^{th} user and broadcast $T_{i,j}$, for all $1 \leq i \leq n$ and for all $1 \leq j \leq m$;
- Reconstruction - the secrets are recovered in the special order $S_m, S_{m-1}, \ldots, S_2, S_1$ - let A be a group of k users - the users $i \in A$ provide the values $f^{m-1}(I_i), f^{m-2}(I_i), \ldots, f^1(I_i), f^0(I_i)$, in this order - thus, for $j = m, m-1, \ldots, 2, 1$, the secret S_j is reconstructed as

$$S_j = P_j(0) = \sum_{i \in A} ((f^{j-1}(I_i) + T_{i,j}) \prod_{l \in A \setminus \{i\}} \frac{l}{l-i}).$$

Another multi-stage secret sharing schemes can be found in [74], [28].

Chapter 4

Applications in Security Protocols

The applications of the secret sharing schemes can be categorized as *secure multiparty computation* protocols, i.e., protocols which allow to some users to compute $f(x_1, \ldots, x_m)$ where x_i is a secret input known only by the i^{th} user. The difficult part is to assure simultaneously the secrecy of each input x_i and the correctness of the output. As a very simple example, we may consider Yao's millionaires' problem - determining the richest person among two millionaires without reveling their fortunes. This corresponds to the case $f(x_1, x_2) = max(x_1, x_2)$. Threshold cryptographic protocols, some e-voting protocols (when $f(x_1, \ldots, x_m) = \sum_{i=1}^{m} x_i$) or some e-auction protocols (when $f(x_1, \ldots, x_m) = \max(x_1, \ldots, x_m)$) are special cases of secure multiparty computation protocols.

4.1 Generic Secure Multiparty Computation Protocols

In this section we discuss secure multiparty computation protocols for any polynomial function $f(x_1, \ldots, x_m)$ over a finite field. The main idea is that every user constructs the shares of his secret input x_i with respect to some access structure over a set of n combiners (we may have $n = m$ and the users and the combiners may be the same parties) and sends them to the combiners. The shares of the value $f(x_1, \ldots, x_m)$ can be somehow derived from these shares and thus $f(x_1, \ldots, x_m)$ can be computed without exposing the inputs (up to reasonably sized coalitions of combiners). Verifiable secret sharing schemes can be used for assuring the correctness. It is sufficient to discuss only the next two cases:

1. $f(x_1, x_2) = c_1 \cdot x_1 + c_2 \cdot x_2$ where c_1, c_2 are known constants - in this case, we can use Shamir's threshold secret sharing scheme for constructing

the shares $I_1^1, \ldots, I_n^1$ corresponding to the secret x_1 and the shares $I_1^2, \ldots, I_n^2$ corresponding to the secret x_2 - by the homomorphic properties of Shamir's secret sharing scheme, $c_1 \cdot I_1^1 + c_2 \cdot I_1^2, \ldots, c_1 \cdot I_n^1 + c_2 \cdot I_n^2$ are the shares corresponding to the value $f(x_1, x_2)$; this can be easily extended to the case $f(x_1, \ldots, x_n) = (x_1, \ldots, x_n) \cdot A = (y_1, \ldots, y_n)$, where A is a constant $n \times n$ matrix, such that the only information known by the i^{th} user/combiner is y_i;

2. $f(x_1, x_2) = x_1 \cdot x_2$ - suppose that the polynomials $P_1(x)$ and $P_2(x)$, both of degree at most $k-1$, are used for constructing the shares $I_1^1, \ldots, I_n^1$ corresponding to the secret x_1 and the shares $I_1^2, \ldots, I_n^2$ corresponding to the secret x_2 with respect to Shamir's (k, n)-threshold secret sharing scheme. Let us consider the polynomial $P(x) = P_1(x) \cdot P_2(x)$, which is of degree at most $2k-2$. It is obvious that $P(0) = x_1 \cdot x_2$, $P(i) = P_1(i) \cdot P_2(i) = I_i^1 \cdot I_i^2$, for all $1 \leq i \leq n$, and, thus, $I_1^1 \cdot I_1^2, \ldots, I_n^1 \cdot I_n^2$ are the shares of the value $f(x_1, x_2)$ with respect to Shamir's $(2k-1, n)$-threshold secret sharing scheme. If $n \geq 2k-1$ there is no problem for a single multiplication, but repeated multiplications will raise the degree and may lead to the situation in which there are not sufficient points for interpolation in the reconstruction phase. As Ben-Or, Goldwasser, and Wigderson have remarked in [6], there is another, more subtle problem in this case - the polynomial $P(x)$ is not a random polynomial (being a product of two polynomials, it cannot be irreducible) and thus, perfect secrecy is not provided. In order to overcome these problems, Ben-Or, Goldwasser, and Wigderson have proposed combining a degree reduction step with a randomized step for $P(x)$ while keeping the free coefficient unchanged:

   - The degree reduction step - let $P(x) = a_0 + a_1 x^1 + \cdots + a_{2k-2} x^{2k-2}$ and define the truncation of $P(x)$ to be the polynomial $Q(x) = a_0 + a_1 x + \cdots + a_{k-1} x^{k-1}$. Ben-Or, Goldwasser, and Wigderson have proven that there is a constant $n \times n$ matrix A such that $(Q(1), \ldots, Q(n)) = (P(1), \ldots, P(n)) \cdot A$ - the shares $Q(1), \ldots, Q(n)$ corresponding to the secret $x_1 \cdot x_2$ with respect to Shamir's (k, n)-threshold secret sharing scheme can be thus constructed using the linear form protocol discussed above;

   - The randomization step - the main idea is that, if we add the shares of the secret 0 with respect to a dealer-free secret sharing scheme (using, for example, Shamir's scheme which is $(+,+)$-homomorphic and choosing each local secret to be zero) to the shares of some secret, we obtain different shares of the same secret. Thus, the degree reduction step can be applied to $P(x) + \sum_{i=1}^{n} R_i(x)$, where $R_i(x)$ is a random polynomial of degree at most $2k-2$ such that $R_i(0) = 0$, for all $1 \leq i \leq n$.

Gennaro, Rabin, and Rabin have proposed a simpler multiplication protocol in [68] by realizing both the degree reduction and the randomization in a single step. Let $P(x) = P_1(x) \cdot P_2(x) = a_0 + a_1 x + \cdots + a_{2k-2}x^{2k-2}$. For $1 \leq i \leq 2k-1$, $I_i^1 \cdot I_i^2 = P_1(i) \cdot P_2(i) = x_1 \cdot x_2 + a_1 i + \cdots + a_{2k-2} i^{2k-2}$. The last identity may be expressed as

$$A \cdot \begin{pmatrix} x_1 \cdot x_2 \\ a_1 \\ \vdots \\ a_{2k-2} \end{pmatrix} = \begin{pmatrix} I_1^1 \cdot I_1^2 \\ I_2^1 \cdot I_2^2 \\ \vdots \\ I_{2k-1}^1 \cdot I_{2k-1}^2 \end{pmatrix},$$

where $A = (a_{i,j})_{1 \leq i,j \leq 2k-1}$, $a_{i,j} = i^{j-1}$. It is obvious that A is non-singular and, thus, it has an inverse. If $(\alpha_1, \ldots, \alpha_{2k-1})$ is the first row of A^{-1} (this row is a known constant), we obtain that $x_1 \cdot x_2 = \sum_{i=1}^{2k-1} \alpha_i (I_i^1 \cdot I_i^2)$. The users/combiners can use now the linear form protocol for deriving the shares corresponding to the secret $x_1 \cdot x_2$ with respect to Shamir's (k, n)-threshold secret sharing scheme.

4.2 Threshold Cryptography

According to [68], *threshold* (or *group-oriented*) cryptography can be defined as "the study of efficient multiparty computation protocols for cryptographic functions". Thus, in threshold cryptography the capacity of performing cryptographic operations such as[1] decryption or digital signature generation is shared among members of a certain group. The first attempts on realizing threshold cryptographic schemes are due to Itakura and Nakamura [89], Croft and Harris [37], Boyd [19], and Desmedt [42].

An obvious solution is to use a secret sharing scheme for splitting the secret key; the secret key is first recovered and then the cryptographic operation is performed (see, for example, [77]). The main problem of this solution is that after the full recovering of the secret key, any member of the recovering group can perform the operation only by himself.

A much better solution is to combine multiplicative secret sharing schemes (see Section 3.1) with homomorphic[2] cryptographic operations. More exactly, the shares $I_1, \ldots, I_n$ corresponding to the private key k_d (or k_s) from

[1] Other cryptographic operations can be transformed in threshold ones. A digital signature scheme with threshold verification has been discussed in [41]. The symmetric cryptographic primitives can be also shared as it will be presented in Section 4.2.3.

[2] Let $g_{\{\}}$ be a parameterized function, and $\odot, \otimes$ be two associative and commutative binary operations over the parameters space and, respectively, over the range of $g_{\{\}}$. We say that $g_{\{\}}$ is $(\odot, \otimes)$- *homomorphic* if

$$g_{k_1 \odot k_2}(x) = g_{k_1}(x) \otimes g_{k_2}(x),$$

for all k_1, k_2, and x.

a public-key cryptosystem (or a digital signature scheme) are constructed using a multiplicative secret sharing scheme by a dealer (there exist also dealer-free threshold cryptographic primitives). Afterwards, the dealer securely distributes the shares to the users. After this, the presence of the dealer is no longer required. If an authorized group of users A want to cooperatively compute an expression of form $g_{k_d}(x)$, for some x, they individually compute the *partial results* of form $y_i = g_{f_{(i,A)}(I_i)}(x)$, for $i \in A$, and send them to a *combiner* which will compute the final result as $\otimes_{i\in A} y_i$. This reasoning is correct because

$$g_{k_d}(x) = g_{\odot_{i\in A} f_{(i,A)}(I_i)}(x) = \otimes_{i\in A} g_{f_{(i,A)}(I_i)}(x).$$

In this way, the secret key will not be revealed to the members of the group A or to the combiner - this is, in fact, the fundamental requirement of the threshold cryptography.

4.2.1 Threshold Discrete Logarithm based Schemes

Desmedt-Frankel Scheme

In [46], Desmedt and Frankel have proposed a threshold ElGamal scheme based on Shamir's threshold secret sharing scheme.

The main idea is to combine the multiplicative properties of Shamir's secret sharing scheme with the $(+_q, \cdot_p)$-homomorphic property of the ElGamal decryption component, $g_a(\gamma) = \gamma^a \ \mathtt{mod}\ p$ (see also Appendix C). More exactly, Shamir's secret sharing scheme is used by the dealer for constructing the shares $I_1, \ldots, I_n$ corresponding to the private key a. The shares are securely distributed to the users. The private key can be expressed, for any group A with $|A| = k$, as

$$a = \sum_{i\in A} (I_i \cdot c_{i,A}) \ (\mathtt{mod}\ q)$$

where

$$c_{i,A} = \prod_{j\in A\setminus\{i\}} \frac{x_j}{x_j - x_i},$$

for all $i \in A$. The values $c_{i,A}$, for $i \in A$, can be computed from the public values $x_1, \ldots, x_n$ and A. The expression $\gamma^a \ \mathtt{mod}\ p$ can be computed as

$$\prod_{i\in A} \gamma^{I_i \cdot c_{i,A}} \ \mathtt{mod}\ p.$$

Thus, in order to cooperatively decrypt a message (γ, δ), the members of a group A with $|A| = k$, compute the partial results $y_i = \gamma^{I_i \cdot c_{i,A}} \ \mathtt{mod}\ p$, for $i \in A$, and send them to a combiner, together with (γ, δ). The combiner computes $\gamma^a \ \mathtt{mod}\ p$ as $\prod_{i\in A} y_i \ \mathtt{mod}\ p$, then obtains $(\gamma^a)^{-1} \ \mathtt{mod}\ p$, and, finally, the plaintext x as $(\gamma^a)^{-1} \cdot \delta \texttt{ mod } p$.

Dealer-Free Threshold *ElGamal* Cryptosystem

Pedersen has proposed a dealer-free threshold variant of ElGamal cryptosystem in [125]. Suppose that n users want to setup together the ElGamal cryptosystem such that any k of them can decrypt an incoming encrypted message. The following steps are performed:

- Users agree on the public parameters (p, q, α) where p is a large prime, such that $p-1$ has a large prime divisor q, α is an element of order q;
- The user i chooses his private key $a_i \in \mathbf{Z}_q^*$ and broadcasts the partial public key $\beta_i = \alpha^{a_i} \texttt{ mod } p$;
- The public key of the entire group will be $\beta = \prod_{i=1}^{n} \beta_i \texttt{ mod } p$ and the private key of the entire group will be $a = \sum_{i=1}^{n} a_i \texttt{ mod } q$;
- The user i derives the shares $I_1^i, \ldots, I_n^i$ corresponding to the secret a_i with respect to Shamir's (k, n)-threshold secret sharing scheme and secretly sends I_j^i to the j^{th} user, for any $1 \leq j \leq n$, $j \neq i$;
- The user i computes $I_i = \sum_{j=1}^{n} I_i^j \texttt{ mod } q$; because Shamir's scheme is $(+_q, +_q)$-homomorphic, $I_1, \ldots, I_n$ computed as above are the shares of the secret key a.

As we will see in Section 4.3.2, dealer-free threshold cryptosystems can be used as building blocks in designing e-voting schemes.

Threshold *ElGamal* based on the Chinese remainder theorem

In [84] we have indicated how the extended Mignotte secret sharing scheme can be combined with ElGamal decryption in order to obtain threshold decryption.

The dealer decides on an authorized access structure $\mathcal{A}$ and generates an $\mathcal{A}$-Mignotte sequence $p_1, \ldots, p_n$ with a large factor $\frac{\alpha-\beta}{\beta}$ such that $\beta < a < \alpha$. The private key a will be the secret and the its corresponding shares $I_1, \ldots, I_n$ will be securely distributed to the users. By the multiplicative properties of the extended Mignotte scheme, the secret key a can be expressed as

$$a = \sum_{i \in A} f_{(i,A)}(I_i) \texttt{ mod } [\{p_i | i \in A\}],$$

for any authorized group A, where $f_{(i,A)}$, for $i \in A$, are some public functions (see Section 3.1).

Suppose that an authorized group of users A want to decrypt a message (γ, δ). If they individually compute the elements $y_i = \gamma^{f_{(i,A)}(I_i)} \texttt{ mod } p$, then

$\prod_{i\in A} y_i \texttt{ mod } p = \gamma^{\sum_{i\in A} f_{(i,A)}(I_i)} \texttt{ mod } p$. On the other hand,

$$\gamma^a \texttt{ mod } p = \gamma^{\sum_{i\in A} f_{(i,A)}(I_i) \texttt{ mod } [\{p_i|i\in A\}]} \texttt{ mod } p$$

Thus, if the access set A additionally satisfies the condition

$$\sum_{i\in A} f_{(i,A)}(I_i) \equiv (\sum_{i\in A} f_{(i,A)}(I_i) \texttt{ mod } [\{p_i|i\in A\}]) \; mod \; ord_p(\gamma)$$

(for example, we may have $(p-1)|[\{p_i|i \in A\}]$ which leads, using that $ord_p(\gamma)|(p-1)$, to $ord_p(\gamma)|[\{p_i|i \in A\}]$) then $\gamma^a \texttt{ mod } p$ can be obtained as $\prod_{i\in A} y_i \texttt{ mod } p$, and the message x can be finally obtained as $(\gamma^a \texttt{ mod } p)^{-1} \cdot \delta \texttt{ mod } p$. Consequently, if p and $\mathcal{A}$ are chosen such that $(p-1)|[\{p_i|i \in A\}]$, for all $A \in \mathcal{A}$, then the decryption can be carried on by any authorized group of users.

Threshold *DSS* Schemes

The first author who has considered threshold *DSS* is Langford [101], for a slightly modified variant of the *DSS*, obtained by switching r with r^{-1} (see also Appendix C).

Gennaro, Jarecki, Krawczyk, and Rabin have proposed more efficient threshold *DSS* schemes in [67]. We present next one of their schemes, in which the capacity of signing is shared among n users such that any $2k-1$ users can collectively produce a valid *DSS* signature.

1. Setup Phase
   - The dealer generates the prime p such that $p-1$ has a large prime divisor q, and α an element of order q; all this information is public;
   - The dealer generates a private polynomial $P(x)$ of degree at most $k-1$ over $\mathbf{Z}_q^*$, and securely sends the share $a_i = P(i)$ to user i, for all $1 \le i \le n$;
   - The dealer broadcasts $\beta = \alpha^{P(0)} \texttt{ mod } p$; $a = P(0)$ will be the secret key;
2. Threshold Signature Generation (for the message $x \in \mathbf{Z}_q^*$)
   - The users collectively derive the shares $r_1, \ldots, r_n$ of some secret $r \in \mathbf{Z}_q^*$, using a dealer-free variant of Shamir's (k,n)-threshold secret sharing scheme;
   - The users collectively derive the shares $z_1^1, \ldots, z_1^n$ and, respectively, $z_2^1, \ldots, z_2^n$ of the secrets $z_1 = 0, z_2 = 0$, using a dealer-free variant of Shamir's $(2k-1,n)$-threshold secret sharing scheme;
   - Compute $\gamma = (\alpha^{r^{-1}} \texttt{ mod } p) \texttt{ mod } q$

- The users collectively derive the shares $\epsilon_1, \ldots, \epsilon_n$ of some secret $\epsilon \in \mathbf{Z}_q^*$, using a dealer-free variant of Shamir's (k, n)-threshold secret sharing scheme;
- The i^{th} user broadcasts $v_i = r_i \cdot \epsilon_i + z_1^i \texttt{ mod } q$ and $w_i = \alpha^{\epsilon_i} \texttt{ mod } p$ (we have to remark that, by the homomorphic properties of the Shamir's secret sharing scheme, $v_1, \ldots, v_n$ are the shadows corresponding to the secret $r \cdot \epsilon \texttt{ mod } q$ with respect to Shamir's $(2k-1, n)$-threshold secret sharing scheme and $w_1, \ldots, w_n$ are the shadows corresponding to the secret $\alpha_\epsilon = \alpha^\epsilon \texttt{ mod } p$, with respect to a variant of Shamir's (k, n)-threshold secret sharing scheme in which the reconstruction method is given by $\alpha_\epsilon = \prod_{i \in A} w_i^{\prod_{j \in A \setminus \{i\}} \frac{j}{j-i}} \texttt{ mod } p$, for a group A, with $|A| = k$);
- Each user locally obtains first $z = r \cdot \epsilon \texttt{ mod } q$ from any $2k-1$ correct (with respect to an additional verifiability procedure) shares from $v_1, \ldots, v_n$, then α_ϵ from any k correct values from $w_1, \ldots, w_n$, and, finally, the value $\gamma = (\alpha_\epsilon^{z^{-1}} \texttt{ mod } p) \texttt{ mod } q$ $(\gamma = (\alpha^{r^{-1}} \texttt{ mod } p) \texttt{ mod } q)$;
- The value γ is published;

- Compute $\delta = r \cdot (x + a \cdot \gamma) \texttt{ mod } q$
  - The i^{th} user broadcasts $\delta_i = r_i(x + a_i\gamma) + z_2^i \texttt{ mod } q$ (we have to remark that $\delta_1, \ldots, \delta_n$ are the shadows corresponding to the secret δ with respect to Shamir's $(2k-1, n)$-threshold secret sharing scheme);
  - Each user locally obtains δ from any $2k-1$ correct shares from $\delta_1, \ldots, \delta_n$;
  - The value δ is broadcasted.

4.2.2 Threshold *RSA* Schemes

Boyd's Schemes

Threshold *RSA* multisignatures have been considered by Boyd [19], for some particular threshold access structures. In a first scheme, Boyd has considered the $(2, 2)$- threshold case. The main idea is to consider the private keys with two components for the *RSA* scheme. More exactly, the dealer generates some parameters e and d_1, d_2 such that

$$e \cdot d_1 \cdot d_2 \equiv 1 \; mod \; \phi(N).$$

The values d_1 and d_2 are securely distributed to two signers. The signature of a message $x \in \mathbf{Z}_N$ is initiated by one signer (for example, by the first signer) which computes $s_1 = x^{d_1} \texttt{ mod } N$ and sends it to the second signer. The second signer can first recover the message x (in order to see what he is signing) by computing $s_1^{d_2 \cdot e}$ $\texttt{mod}$ N, and then he can compute the final signature as $s_2 = s_1^{d_2}$ $\texttt{mod}$ N. The validity of the final signature with respect to the message x can be verified by checking $x \overset{?}{=} s_2^e$ $\texttt{mod}$ N. As it can be easily seen, the order of signing is not important. Boyd has extended this scheme to the more general $(2, n)$-case.

Boyd has also discussed the unanimous consent threshold RSA. The main idea is to generate e and $d_1, \cdots, d_n$ such that

$$e \cdot \sum_{i=1}^{n} d_i \;\equiv\; 1 \; mod \; \phi(N).$$

For a message x, each signer individually computes the partial signature $s_i = x^{d_i}$ $\texttt{mod}$ N, for all $1 \leq i \leq n$. The partial signatures are sent to a combiner which computes the final signature s as $s = \prod_{i=1}^{n} s_i$ $\texttt{mod}$ N.

Desmedt-Frankel Scheme

In [46], Desmedt and Frankel have risen the problem of more general threshold RSA. Although the RSA private transformation, $g_d(x) = x^d$ $\texttt{mod}$ N, is $(+_{\lambda(N)}, \cdot_N)$-homomorphic, they have remarked that Shamir's threshold secret sharing scheme cannot be used directly for threshold RSA because Lagrange's interpolation requires a field structure. There are two possible solutions for this problem. The first solution is to extend Shamir's scheme to work over finite modules, as proposed, for example, by De Santis, Desmedt, Frankel, and Yung in [40]. The second one is to find some tricks such that the interpolation can be carried on, as proposed in [47], [136], [39].

Desmedt and Frankel [47] have proposed a solution for the case that p and q are *safe*[3] primes, i.e., $p = 2p' + 1$ and $q = 2q' + 1$ with p' and q' primes. The main idea is that, if the parameters are carefully chosen, the computations can be carried on correctly. Let us consider Lagrange's formula for reconstructing the secret in Shamir's secret sharing scheme whereas performing the operations modulo $\lambda(N)$:

$$S = \sum_{i \in A} (P(x_i) \cdot \prod_{j \in A \setminus \{i\}} \frac{-x_j}{x_i - x_j}) \; \texttt{mod} \; 2p'q'.$$

The main problem is that not all expressions $x_i - x_j$ have multiplicative inverses modulo $2p'q'$. In fact, the main problem is related to the parity of these expressions, because any three expressions $(x_i - x_j)$, $(x_j - x_l)$, and $(x_i - x_l)$ cannot be simultaneously odd. Desmedt and Frankel have proposed choosing the public elements $x_1, \ldots, x_n$ as odd positive integers smaller than p and q, the polynomial $P(x)$ such that $P(x_1), \ldots, P(x_n)$ be all even, and

[3] In fact, as it has been remarked in [47], it is sufficient that $\frac{\lambda(N)}{2}$ is a product of large primes.

the secret as $S = P(0) = d - 1$. In this case, the secret can be expressed as

$$S = \sum_{i \in A}((\frac{\frac{P(x_i)}{2}}{\frac{\prod_{j=1,j\neq i}^{n}(x_i-x_j)}{2}} \bmod p'q') \cdot \prod_{j \in \{1,\ldots,n\}\setminus A}(x_i - x_j) \cdot \prod_{j \in A\setminus\{i\}}(-x_j)) \bmod 2p'q'.$$

The dealer computes the shares

$$I_i = \frac{\frac{P(x_i)}{2}}{\frac{\prod_{j=1,j\neq i}^{n}(x_i-x_j)}{2}} \bmod p'q',$$

for $1 \le i \le n$ and securely sends them to the users. In order to cooperatively compute $x^d \bmod N$, for some $x \in \mathbf{Z}_N$, the members of a group A with $|A| = k$, compute the partial results

$$y_i = x^{I_i \cdot \prod_{j \in \{1,\ldots,n\}\setminus A}(x_i-x_j) \cdot \prod_{j \in A\setminus\{i\}}(-x_j)} \bmod N,$$

for $i \in A$ and send them to a combiner, together with the message x. The combiner obtains the final signature as $(\prod_{i \in A} y_i \bmod N) \cdot x \bmod N$.

Shoup's Scheme

Shoup [136] has proposed a more efficient and robust scheme for the case that p and q are safe primes, $p = 2p' + 1$, $q = 2q' + 1$. The main idea is to perform the base computations in $\mathbf{Q}_N$, the subgroup of squares[4] in $\mathbf{Z}_N^*$, and the exponent operations in $\mathbf{Z}_{p'q'}$ (this is convenient because $\mathbf{Q}_N$ is cyclic of order $p'q'$ and, thus, its order has no small prime factors). The scheme is described next:

1. Setup Phase
   - The dealer generates two safe primes $p = 2p' + 1$, $q = 2q' + 1$;
   - The dealer computes $N = pq$ and generates a prime e such that $e > n$, $e \neq p'$, $e \neq q'$;
   - The public key is (e, N);
   - The dealer chooses $d \in \mathbf{Z}_{p'q'}^*$ such that $ed \equiv 1 \; mod \; p'q'$;
   - The dealer generates the polynomial $P(x) = a_0 + a_1x + \cdots + a_{k-1}x^{k-1}$ of degree at most $k - 1$ over $\mathbf{Z}_{p'q'}$, such that $a_0 = d$ and computes the shares $I_i = P(i) \; (mod \; p'q')$, for $1 \le i \le n$, and securely sends them to users;
   - The dealer generates $v \in \mathbf{Q}_N$ and broadcasts v and $v_i = v^{I_i}$, for $1 \le i \le n$, as verification keys (remark that v is, with great probability, a generator of $\mathbf{Q}_N$ and, thus, the shares I_i are completely determined by the elements v_i);

[4] A positive integer a is called a *square* in $\mathbf{Z}_N^*$ if there is $b \in \mathbf{Z}_N^*$ such that $a \equiv b^2 \; mod \; N$.

2. Signature generation for a message $x \in \mathbf{Z}_N^*$ (for general messages, a hash function h mapping messages to elements of $\mathbf{Z}_N^*$ can be used)

   - The partial signature of the i^{th} user on message x is $y_i = x^{2\Delta I_i}$, for $1 \leq i \leq n$, where $\Delta = n!$;
   - The i^{th} user sends y_i and a non-interactive proof of its correctness (proof that the discrete logarithm to the base $x^{4\Delta}$ of y_i^2 equals the discrete logarithm to the base v of v_i (see Figure 3.1 from Section 3.3.4)) to the combiner;
   - The combiner computes the final signature for a group A, $|A| = k$ following the next steps:
     - Compute $z = \prod_{i \in A} y_i^{2\lambda_i^A}$, where $\lambda_i^A = \Delta \cdot \frac{\prod_{j \in A \setminus \{i\}} j}{\prod_{j \in A \setminus \{i\}}(j-i)}$ (remark that $z = x^{4\Delta^2 d}$ - indeed, by Lagrange's interpolation formula, the relation $\Delta \cdot d \equiv \sum_{i \in A} \lambda_i^A \cdot I_i \; mod \; p'q'$ holds true);
     - Determine, using the extended Euclidian algorithm, the integers α and β such that $\alpha \cdot 4\Delta^2 + \beta \cdot e = 1$ (such integers always exist because e is coprime to $4\Delta^2$);
     - Compute the final signature of the group A as
$$y = z^{\alpha} x^{\beta}.$$
       Indeed, $y^e = x^{4\Delta^2 de\alpha} x^{\beta e} = x^{\alpha 4\Delta^2 + \beta e} = x$.

Damgård and Dupont [39] have extended this scheme for the case of general modules.

Threshold *RSA* based on the Chinese remainder theorem

In [82] we have shown how to accomplish threshold *RSA* using the extended Mignotte secret sharing scheme. We remind the fact that this secret sharing scheme is multiplicative (see Section 3.1) and, thus, if the secret exponent d has the shares $I_1, \ldots, I_n$, then it can be expressed as

$$d = \sum_{i \in A} f_{(i,A)}(I_i) \; \mathtt{mod} \; [\{p_i | i \in A\}],$$

for any authorized group A, where $f_{(i,A)}$, for $i \in A$, are some public functions and $p_1, \ldots, p_n$ is the utilized $\mathcal{A}$-Mignotte sequence. The main problem is how to combine $x^{f_{(i,A)}(I_i)} \; \mathtt{mod} \; N$ for some authorized access set A in order to obtain $x^d \; \mathtt{mod} \; N$. One elegant solution to this problem is to choose the sequence $p_1, \ldots, p_n$ such that $[p-1, q-1] | [\{p_i | i \in A\}]$, for any authorized set A. In this case we obtain that

$$\begin{aligned} y = x^d \texttt{ mod } N &= x^{\sum_{i\in A} f_{(i,A)}(I_i) \texttt{ mod } [\{p_i|i\in A\}]} \texttt{ mod } N \\ &= x^{\sum_{i\in A} f_{(i,A)}(I_i)} \texttt{ mod } N \\ &= \prod_{i\in A} x^{f_{(i,A)}(I_i)} \texttt{ mod } N \\ &= \prod_{i\in A}(x^{f_{(i,A)}(I_i)} \texttt{ mod } N) \texttt{ mod } N, \end{aligned}$$

for any authorized set A. The sequence $p_1, \ldots, p_n$ can be obtained by multiplying every element of an $\mathcal{A}$-Mignotte sequence $p'_1, \ldots, p'_n$ with $[p-1, q-1]$, providing that $([p-1, q-1], p'_1 \cdots p'_n) = 1$.

Example 4.2.1 illustrates our scheme.

Example 4.2.1 (A $(2,3)$-threshold *RSA* scheme based on Mignotte scheme (with artificially small parameters))

Let $N = 481$, $p = 13$, $q = 37$, $d = 401$, $x = 39$, $n = 3$, and $k = 2$. Let us consider the numbers $p_1 = 180$, $p_2 = 252$, and $p_3 = 396$. The sequence p_1, p_2, p_3 is a generalized $(2,3)$-Mignotte sequence that satisfies that $[p-1, q-1]$ divides $[p_1, p_2]$, $[p_1, p_3]$, and $[p_2, p_3]$. The shares corresponding to the secret d are $I_1 = 41$, $I_2 = 149$, and $I_3 = 5$. Suppose that we want to compute the value $y = x^d \texttt{ mod } N$ having I_1 and I_3. In this case, because

$$d = (11 \cdot 1 \cdot \texttt{41} + 5 \cdot 1978 \cdot \texttt{5}) \texttt{ mod } 1980,$$

y can be obtained as

$$(39^{11\cdot 1\cdot \texttt{41}} \texttt{ mod } 481) \cdot (39^{5\cdot 1978\cdot \texttt{5}} \texttt{ mod } 481) \texttt{ mod } 481,$$

which leads to the correct result $y = 143$.

Next, we will show how to accomplish threshold *RSA* using the extended Asmuth-Bloom threshold secret sharing scheme. This scheme is also multiplicative in the sense that, if the secret exponent d has the shares $I_1, \ldots, I_n$, then it can be expressed as:

$$d = (\sum_{i\in A} f_{(i,A)}(I_i) \texttt{ mod } [\{p_i|i \in A\}]) \texttt{ mod } p_0,$$

for every authorized set A and for all $i \in A$.

We may choose the sequence $p_0, p_1, \ldots, p_n$ such that $[p-1, q-1]|p_0$ and $[p-1, q-1]|[\{p_i|i \in A\}]$, for any authorized set A. In this case we obtain that

$$y = x^d \texttt{ mod } N = \prod_{i\in A}(x^{f_{(i,A)}(I_i)} \texttt{ mod } N) \texttt{ mod } N.$$

Example 4.2.2 illustrates our scheme.

Example 4.2.2 (A $(2,3)$-threshold RSA scheme based on Asmuth-Bloom scheme (with artificially small parameters))

Let $N = 481$, $p = 13$, $q = 37$, $d = 71$, $x = 39$, $n = 3$, and $k = 2$. Let us consider the numbers $p_0 = 72$, $p_1 = 4068$, $p_2 = 4572$, and $p_3 = 4716$. The sequence p_0, p_1, p_2, p_3 is indeed a generalized $(2,3)$-Asmuth-Bloom sequence that satisfies that $[p-1, q-1]$ divides p_0, $[p_1, p_2]$, $[p_1, p_3]$, and $[p_2, p_3]$. If we choose $\gamma = 150$, the shares corresponding to the secret d are $I_1 = 2735$, $I_2 = 1727$, and $I_3 = 1439$. Suppose that we want to compute the value $y = x^d \texttt{ mod } N$ having I_2 and I_3. In this case, because

$$d = ((131 \cdot 32 \cdot \texttt{1727} + 127 \cdot 598899 \cdot \texttt{1439}) \texttt{ mod } 598932) \texttt{ mod } 72,$$

y can be obtained as

$$(39^{131 \cdot 32 \cdot \texttt{1727}} \texttt{ mod } 481) \cdot (39^{127 \cdot 598899 \cdot \texttt{1439}} \texttt{ mod } 481) \texttt{ mod } 481,$$

which leads to the correct result $y = 130$.

It is important to remark that the modules used in the schemes above can not be pairwise prime and, thus, the extended secret sharing schemes based on the general variant of the Chinese remainder theorem must be used.

4.2.3 Threshold Symmetric Cryptographic Primitives

As opposed to the public-key cryptographic primitives, the symmetric ones do not provide any homomorphic properties. The main idea for realizing threshold symmetric cryptographic primitives is to iterate these schemes and "share the iteration". This idea has been presented by Brickell, Di Crescenzo and Frankel [21], and later by Martin, Safavi-Naini, Wang, and Wild [110], for the case of symmetric cryptosystems, and by Martin, Pieprzyk, Safavi-Naini, Wang, and Wild [108] for the case of message authentication codes (*MACs*). We will present next several methods of iterations and discuss the corresponding threshold features.

Cascade Iteration

Let $e_{\{\}}$ be a symmetric encryption function and l be a positive integer, $l \geq 2$. The *l-fold composition* (or *cascade*) of $e_{\{\}}$, denoted by $e^l_{\{\}}$, is given by

$$e^l_{(k_1,\ldots,k_l)}(x) = e_{k_l}(e_{k_{l-1}}(\cdots e_{k_2}(e_{k_1}(x))\cdots)),$$

for any keys $k_1, \ldots, k_l$ and any plaintext x. The main advantage of cascade iteration is that there is no message expansion.

The notion of *sequence sharing scheme* has been introduced in [21]. For an access structure $\mathcal{A}$ over $\{1, 2, \ldots, n\}$, an $\mathcal{A}$*-sequence sharing scheme* is an $\mathcal{A}$-secret sharing scheme in which the secrets are of form $\mathtt{k} = (k_1, \ldots, k_{n(\mathcal{A})})$, for some $n(\mathcal{A}) \geq n$ and the shares are sets of component keys. Such schemes can be used in order to provide threshold decryption as follows. The sender has $\mathtt{k} = (k_1, \ldots, k_{n(\mathcal{A})})$ and the receivers have sets of component keys, corresponding to some access structure $\mathcal{A}$. The sender computes $y = e_{\mathtt{k}}^{n(\mathcal{A})}(x)$, for some plaintext x and sends the cryptotext y to the group of receivers. An authorized group of receivers can gather all component keys $k_1, \ldots, k_{n(\mathcal{A})}$ and obtain the plaintext x as $x = d_{k_1}(d_{k_2}(\cdots d_{k_{n(\mathcal{A})-1}}(d_{k_{n(\mathcal{A})}}(y))\cdots))$, where $d_{\{\}}$ is the decryption function corresponding to $e_{\{\}}$. The same idea can be used for realizing threshold encryption.

For efficiency reasons, it is important to find $\mathcal{A}$-sequence sharing schemes with a small number of component keys $n(\mathcal{A})$. As it has been remarked in [110], an $\mathcal{A}$-sequence sharing scheme is nothing else than an $\mathcal{A}$-cumulative scheme [5] with a publicly known ordering of the component keys and, thus, $n(\mathcal{A}) \geq |\overline{\mathcal{A}}_{max}|$ (according to Proposition 2.7.1). This may be a problem, because, for some access structures, this implies a very large $n(\mathcal{A})$. For example, if $\mathcal{A}$ is the (k, n)-threshold access structure, $|\overline{\mathcal{A}}_{max}| = \frac{n!}{(k-1)!(n-k+1)!}$. Moreover, in this case, each receiver has to hold $\frac{(n-1)!}{(k-1)!(n-k)!}$ component keys.

The authors of [110] have proposed using *generalized cumulative maps* in order to reduce the number of component keys.

Definition 4.2.1 Let $\mathcal{A}$ be an access structure over $\{1, 2, \ldots, n\}$. A *generalized* $\mathcal{A}$*-cumulative map* is a set $\{((k_1^j, k_2^j, \ldots, k_{m_j}^j), f_j) | 1 \leq j \leq l\}$ where $l \geq 1$ and $m_j \geq 1$, $k_1^j, k_2^j, \ldots, k_{m_j}^j$ are arbitrary keys, and f_j is a function from $\{1, 2, \ldots, n\}$ to $\mathcal{P}(\{k_1^j, k_2^j, \ldots, k_{m_j}^j\})$, for all $1 \leq j \leq l$, such that

$$(\forall A \in \mathcal{P}(\{1, 2, \ldots, n\}))((\exists 1 \leq j \leq l)(\cup_{i \in A} f_j(i) = \{k_1^j, k_2^j, \ldots, k_{m_j}^j\}) \Leftrightarrow A \in \mathcal{A}).$$

For $l = 1$ we obtain the regular $\mathcal{A}$-cumulative maps.

Such sequences with publicly known orderings of the component keys can be used for realizing threshold symmetric decryption in an obvious manner. The sender produces a partial cryptotext corresponding to each sequence of keys $(k_1^j, \ldots, k_{m_j}^j)$, for $1 \leq j \leq l$, and sends them to the group of receivers. The main advantage of this technique is that, in some cases, the total number

[5] At this point, an equivalent definition of cumulative map will be used (see also Section 2.7.1). More exactly, an $\mathcal{A}$-cumulative map is a pair $(\{k_1, \ldots, k_m\}, f)$ where m is a positive integer, $k_1, \ldots, k_m$ are arbitrary keys, and f is a function from $\{1, 2, \ldots, n\}$ to $\mathcal{P}(\{k_1, k_2, \ldots, k_m\})$ such that

$$(\forall A \in \mathcal{P}(\{1, 2, \ldots, n\}))(\cup_{i \in A} f(i) = \{k_1, k_2, \ldots, k_m\} \Leftrightarrow A \in \mathcal{A}).$$

of component keys ($\sum_{j=1}^{l} m_j$) and the number of the keys held by each user can be significantly reduced, with the cost of increasing the size of the cryptotext. The next example illustrates this technique.

Example 4.2.3 ([110])
Let $\mathcal{A}$ be the $(2,9)$-threshold access structure. A generalized $\mathcal{A}$-cumulative map is $\{((k_1^1,k_2^1),f_1),((k_1^2,k_2^2),f_2),((k_1^3,k_2^3),f_3),((k_1^4,k_2^4),f_4)\}$ where f_1,f_2,f_3, and f_4 are given by

$$\begin{array}{llll} f_1(i)=\{k_1^1\}, & \text{for } 1\le i\le 8 & \text{and } f_1(i)=\{k_2^1\}, & \text{for } i=9\\ f_2(i)=\{k_1^2\}, & \text{for } i\in\{1,2,3,4,9\} & \text{and } f_2(i)=\{k_2^2\}, & \text{for } i\in\{5,6,7,8\}\\ f_3(i)=\{k_1^3\}, & \text{for } i\in\{1,2,5,6,9\} & \text{and } f_3(i)=\{k_2^3\}, & \text{for } i\in\{3,4,7,8\}\\ f_4(i)=\{k_1^4\}, & \text{for } i\in\{1,3,5,7,9\} & \text{and } f_4(i)=\{k_2^4\}, & \text{for } i\in\{2,4,6,8\}. \end{array}$$

The total number of component keys is, in this case, 8 while the total number of component keys corresponding to the minimal cumulative map is 9. The main gain is the significant reduction of the number of component keys held by each user - 4 for the technique based on the presented generalized cumulative map as opposed to 8 for the case of minimal cumulative map.

The generalized cumulative maps have been recently investigated in [103].

XOR Iteration

Let $e_{\{\}}$ be a symmetric encryption function, $\mathcal{A}$ be an access structure over $\{1,2,\ldots,n\}$. An $\mathcal{A}$-cumulative map is used to distribute component keys of $\mathtt{k}=(k_1,\ldots,k_{n(\mathcal{A})})$. The threshold *XOR*-encryption of a message x with respect to key sequence $\mathtt{k}$ ([110]) is realized as

$$e_{\mathtt{k}}^{\oplus n(\mathcal{A})}(x)=(x\oplus e_{k_1}(r)\oplus e_{k_2}(r)\oplus\cdots\oplus e_{k_{n(\mathcal{A})}}(r),r),$$

where r is a random value.

The main advantage of *XOR* iteration over the cascade iteration is that no order of using the component keys is required and, thus, the values $e_{k_1}(r),\ldots,e_{k_{n(\mathcal{A})}}(r)$ can be computed in parallel. The main disadvantage of this technique is the doubling the size of the cryptotext due to the component r. The generalized cumulative maps can also be used in this case, reducing the number of component keys with the cost of increasing the size of the cryptotext.

Threshold *MACs*

A *XOR*-like construction for the case of *MACs* has been discussed in [108]. Let $F_{\{\}}$ be a *MAC* and $l\ge 2$. The *l-fold XOR MAC* induced by $F_{\{\}}$, denoted by $F_{\{\}}^l$, is given by

$$F_{\{k_1,\ldots,k_l\}}^l(x)=\oplus_{i=1}^l F_{k_i}(x),$$

for any keys $k_1, \ldots, k_l$ and any message x.

The authors have introduced the concept of *quasi cover-free family.*

Definition 4.2.2 Let X be a non-empty set, $\{X_1, \ldots, X_l\}$ be a partition of X, and $\mathcal{B} \subseteq \mathcal{P}(X)$, $\mathcal{B} = \{B_1, \ldots, B_n\}$. We say that $(X_1, \ldots, X_l; \mathcal{B})$ is a (l, k, n)-*quasi cover-free family* if the next conditions hold:

- For any $A \subseteq \{1, 2, \ldots, n\}$ with $|A| = k$, there is an element j, $1 \leq j \leq l$, such that $X_j \subseteq \cup_{i \in A} B_i$;
- For any $A \subseteq \{1, 2, \ldots, n\}$ with $|A| = k - 1$, and for any element j, $1 \leq j \leq l$, $X_j \not\subseteq \cup_{i \in A} B_i$.

In case $|X_1| = |X_2| = \cdots = |X_l| = \delta$ for some positive integer δ, $(X_1, \ldots, X_l; \mathcal{B})$ will be referred to as a (δ, l, k, n)-*quasi cover-free family.*

An (k, n)-threshold *MAC* based on quasi cover-free families can be constructed as follows:

- The receiver first generates a set of keys $X = \{k_1, \ldots, k_{\delta l}\}$, then generates a (δ, l, k, n)-quasi cover-free family $(X_1, \ldots, X_l; \mathcal{B})$, $\mathcal{B} = \{B_1, \ldots, B_n\}$, and securely distributes the set B_i to the i^{th} sender, for $1 \leq i \leq n$;
- Suppose that a group of senders A, $|A| = k$, want to authenticate a message x. They first find an index j such that $X_j \subseteq \cup_{i \in A} B_i$, then compute $y = F_{X_j}^{|X_j|}(x)$ and send (x, y, j) to the receiver;
- The receiver can verify the authenticity of the message x by computing $F_{X_j}^{|X_j|}(x)$ and comparing the result with y.

4.3 E-voting Schemes based on Secret Sharing

According to [151], "an *electronic voting* (*e-voting*) system is a voting system in which the election data is recorded, stored and processed primarily as digital information". We present next the most important requirements for an e-voting scheme (the reader is referred to [73] for more details):

- *Correctness* - according to this requirement, the announced tally is identical with the real outcome of the election;
- *Privacy* - this requirement guarantees that no reasonably sized coalitions of voters or authorities may link a voter's identity to his vote;
- *Robustness* - according to this requirement, no reasonably sized coalitions of voters or authorities may affect the election;

- *Verifiability* - this requirement assures the existence of some mechanisms for auditing the election in order to verify if it has taken place properly.

We have to remark that these properties may contradict or interrelate one with another. For example, verifiability implies the existence of some proofs for the consistency of the votes but these may affect privacy. On the other hand, verifiability is a strong supporter both for the correctness and the robustness of the scheme. We will consider only the case of *yes/no* e-voting but we have to mention that the presented schemes can be extended to parallel elections or multiway elections.

4.3.1 E-voting Schemes based on Homomorphic Secret Sharing

The parties involved are the voters $V_1, \ldots, V_m$, the tallying authorities $A_1, \ldots, A_n$ and the central authority $\mathtt{A}$. A certain value v_{yes} is assigned to a *yes* vote and a certain value v_{no} is assigned to a *no* vote. The main idea behind this e-voting technique, due to Benaloh [7, 8], is that every voter splits his vote into some sub-votes using a $(+,+)$-homomorphic secret sharing scheme and sends them to the tallying authorities which compute the sums of the incoming sub-votes. By the $(+,+)$-homomorphic property of the underlying secret sharing scheme, the sums of the sub-votes are shares of the sum of the votes and, thus, the sum of the votes S can be computed without compromising the privacy of the votes. The central authority computes the final sum, using an authorized set of the partial sums communicated by the tallying authorities. Figure 4.1 illustrates this idea.

$$\begin{array}{ccccc}
\textbf{Votes} & & & \textbf{Sub}-\textbf{Votes} & \\
v_1 & \leftrightarrow & v_{1,1} & \cdots & v_{1,n} \\
\vdots & \vdots & \vdots & & \vdots \\
v_m & \leftrightarrow & v_{m,1} & \cdots & v_{m,n} \\
\hline
\underbrace{\sum_{j=1}^{m} v_j}_{\textbf{Final Sum}} & \leftrightarrow & \underbrace{\sum_{j=1}^{m} v_{j,1}}_{\textbf{PartialSum}_\textbf{1}} & \cdots & \underbrace{\sum_{j=1}^{m} v_{j,n}}_{\textbf{PartialSum}_\textbf{n}}
\end{array}$$

Figure 4.1: An E-voting Scenario based on Homomorphic Secret Sharing

The result of the voting can be obtained by solving the equation

$$v_{yes} \cdot x + v_{no} \cdot y = S.$$

Several particular choices of the values v_{yes} and v_{no} may be made:

- $v_{yes} = 1$, $v_{no} = 0$ (as in [7]) - in this case,

$$\begin{array}{rcl} number\_votes\_yes & = & S, \\ number\_votes\_no & = & m - S; \end{array}$$

- $v_{yes} = 1$, $v_{no} = -1$ (as in [34] or [35]) - in this case,

$$\begin{array}{rcl} number\_votes\_yes & = & \frac{S+m}{2}, \\ number\_votes\_no & = & \frac{m-S}{2}; \end{array}$$

- $m \cdot v_{yes} < v_{no}$ (as in [85]) - in this case,

$$\begin{array}{rcl} number\_votes\_no & = & S \ \mathtt{div} \ v_{no}, \\ number\_votes\_yes & = & (S \ \mathtt{mod} \ v_{no}) \ \mathtt{div} \ v_{yes}. \end{array}$$

In all these cases, any secure multiparty computation scheme corresponding to the function $f(v_1, \ldots, v_m) = \sum_{j=1}^{m} v_j$ is in fact an e-voting scheme.

There are three main problems with the presented scenario:

- There is no proof that a vote is indeed chosen as an element of the set $\{v_{yes}, v_{no}\}$; thus, a vote with value 2 is equal with two votes *yes* (in case that $v_{yes} = 1$);
- There is no proof that the sub-votes are correctly derived from the vote - any incorrect sub-vote can perturb the final result;
- There is no proof that the partial sums are correctly computed from the received sub-votes - any incorrect partial sum can perturb the final result.

Cramer, Franklin, Schoenmakers, and Yung have proposed an e-voting scheme that solves all the above problems in [34]. Their scheme is in fact a combination of verifiable homomorphic secret sharing (that solves the problem of the consistency of the sub-votes and, by some extension, the problem of the consistency of the partial sums) and verifiable encryption (that assures that any vote has been chosen as an element of the set $\{v_{yes}, v_{no}\}$).

We present next their scheme in more detail.

- **Setup**
  - There are generated the primes p and q such that $q|(p-1)$, $m < \frac{q}{2}$, $\alpha, h \in \mathbf{Z}_p^*$ elements of order q and a security parameter $2 \leq k \leq n$;
- **Ballot Construction**

- Each voter V_j, $1 \leq j \leq m$, chooses a vote mask[6] $b_j \in \{-1, 1\}$ for his vote $v_j \in \{-1, 1\}$;
- V_j computes $B_j = \alpha^{\delta_j} h^{b_j} \texttt{ mod } p$, where $\delta_j \in \mathbf{Z}_q^*$;
- V_j chooses the polynomials $G_j(x)$ and $H_j(x)$ of degree at most $k-1$ over $\mathbf{Z}_q^*$ such that $G_j(0) = \delta_j$ and $H_j(0) = b_j$, or, more exactly, $G_j(x) = \delta_j + \delta_{j,1}x^1 + \cdots + \delta_{j,k-1}x^{k-1}$ and $H_j(x) = b_j + \beta_{j,1}x^1 + \cdots + \beta_{j,k-1}x^{k-1}$;
- V_j broadcasts B_j, $proof(B_j)$ ($proof(B_j)$ is a proof that B_j corresponds to an element $b_j \in \{-1, 1\}$ (such a proof will be presented in the next section)), and $B_{j,1}, \ldots, B_{j,k-1}$, where $B_{j,l} = \alpha^{\delta_{j,l}} h^{b_{j,l}} \texttt{ mod } p$, for all $1 \leq l \leq k-1$;
- Any party can verify the correctness of B_j, using $proof(B_j)$;
- The voter V_j securely sends $(a_{i,j}, b_{i,j})$ to the tallying authority A_i, where $a_{i,j} = G_j(i)$ and $b_{i,j} = H_j(i)$, for all $1 \leq i \leq n$;
- Any tallying authority A_i can verify the correctness of the received information from any voter V_j by testing

$$\alpha^{a_{i,j}} h^{b_{i,j}} \texttt{ mod } p \overset{?}{=} B_j \prod_{l=1}^{k-1} B_{j,l}^{il} \texttt{ mod } p;$$

- **Vote Casting**
  - Any voter V_j chooses his vote $v_j \in \{-1, 1\}$ and broadcasts the element s_j, where $s_j \in \{-1, 1\}$, such that $v_j = s_j \cdot b_j$;

- **Tallying**
  - Each tallying authority A_i, $1 \leq i \leq n$, broadcasts (S_i, T_i), where $S_i = \sum_{j=1}^{m} b_{i,j} s_j \textsf{ mod } q$ and $T_i = \sum_{j=1}^{m} a_{i,j} s_j \textsf{ mod } q$;
  - Any party can verify the correctness of the information broadcasted by the tallying authority A_i by testing

$$\alpha^{T_i} h^{S_i} \texttt{ mod } p \overset{?}{=} \prod_{j=1}^{m} (B_j \prod_{l=1}^{k-1} B_{j,l}^{il})^{s_j} \texttt{ mod } p;$$

  - The central authority $\texttt{A}$ (or any party) computes the final sum

$$S = \sum_{i \in A} (S_i \prod_{j \in A \setminus \{i\}} \frac{j}{j-i}) \textsf{ mod } q,$$

[6] The use of the masks allows that the most time-consuming step of this e-voting scheme, the ballot construction, be performed off-line.

for an arbitrary set $A \subseteq \{1, \ldots, n\}$, $|A| = k$, for which the information $(S_i, T_i),\ i \in A$, is consistent.

Properties of this e-voting scheme are

- Privacy - against coalitions of size $\leq k - 1$;
- Robustness - against coalitions of size $\leq n - k$;
- Public verifiability (anyone can verify the correctness of the election).

A General Multi-Authority E-Voting Scheme

In [85] we have proposed a multi-authority e-voting scheme in which, as a novelty, the tallying authorities may have non-equal weights. The parties involved are the voters $V_1, \ldots, V_m$, the tallying authorities $A_1, \ldots, A_n$ and the central authority $\mathtt{A}$. We present next the steps of our e-voting scheme.

- **Setup**
  - The central authority $\mathtt{A}$ decides on an authorized access structure $\mathcal{A}$ for the tallying authorities and generates and broadcasts an $\mathcal{A}$-Mignotte sequence $p_1, \ldots, p_n$ with a large factor $\frac{\alpha-\beta}{\beta}$;
  - The central authority $\mathtt{A}$ broadcasts the values v_{yes} and v_{no}, where $v_{yes}, v_{no} \in \{\beta + 1, \ldots, \alpha - 1\}$ are assigned to the *yes* vote and to the *no* vote;
- **Ballot Construction**
  - For $1 \leq j \leq m$, the voter V_j chooses a vote mask[7] b_j, $0 < b_j < \alpha - v_j$, for his vote $v_j \in \{v_{yes}, v_{no}\}$ and forms the ballot $B_j = v_j + b_j$;
  - The voter V_j securely sends the sub-ballot $B_{j,i} = B_j \ \mathtt{mod}\ p_i$ to the tallying authority A_i, for all $1 \leq j \leq m$ and for all $1 \leq i \leq n$;
- **Ballot Tallying**
  - At the end of ballot construction period, the tallying authority A_i computes the partial "masked" tally $T_i = \sum_{j=1}^{m} B_{j,i} \ \mathtt{mod}\ p_i$ and securely sends it to the central authority $\mathtt{A}$, for all $1 \leq i \leq n$;
  - The central authority $\mathtt{A}$ obtains the final "masked" tally $T = \sum_{j=1}^{m} B_j$ by solving, using the general variant of the Chinese remainder theorem, the system of equations

[7] The purpose of the masks is to hide any information about the voters' choices. Moreover, the ballot construction and ballot tallying can be performed in advance.

$$\left\{ \begin{array}{lcl} x & \equiv & T_i \; mod \; p_i, \quad i \in A \end{array} \right.$$

for some $A \in \mathcal{A}$; for the correctness of this reasoning, any possible final "masked" tally T must satisfy $T < \alpha$ (the relation $\beta < T$ holds true by the choice of the values v_{yes}, v_{no} and of the mask values). Indeed, in this case, by the fact that the generalized Mignotte secret sharing scheme is $(+, +_{p_1}, \ldots, +_{p_n})$-partial homomorphic, the values $T_1, \ldots, T_n$ are the shares of the element T. In order to assure $T < \alpha$, the central authority `A` may impose, for example, the condition

$$m \cdot (max(v_{yes}, v_{no}) + max(b_1, \ldots, b_m)) < \alpha;$$

- At the previous step, the central authority `A` can also verify the consistency of the values $T_i, i \in A$, by testing if $T_i \equiv T_{i'} \; mod \; (p_i, p_{i'})$, for any $i, i' \in A$;

- **Vote Casting**
  - At the end of ballot tallying period, the voter V_j securely sends b_j to the central authority `A`, for all $1 \leq j \leq m$;
- **Vote Counting**
  - At the end of vote casting period, the central authority `A` computes the sum of votes $S = \sum_{j=1}^{m} v_j$ as $S = T - \sum_{j=1}^{m} b_j$;
  - The numbers of *yes* and *no* votes can be obtained as the solution of the equation $v_{yes} \cdot x + v_{no} \cdot y = S$; if the values v_{yes} and v_{no} are chosen such that $m \cdot v_{yes} < v_{no}$, then this solution can be determined as

$$\begin{array}{lcl} number\_votes\_no & = & S \; \texttt{div} \; v_{no}, \\ number\_votes\_yes & = & (S \; \texttt{mod} \; v_{no}) \; \texttt{div} \; v_{yes}. \end{array}$$

- The central authority broadcasts *number_votes_yes*, *number_votes_no*.

Our e-voting scheme has the following properties:

- *Privacy* - in order to link a voter's identity (V_j) to his vote (v_j), at least an authorized group of tallying authorities and the central authority must collaborate. Indeed, the ballot B_j may be reconstructed only by an authorized group of tallying authorities and the mask b_j is known only by the central authority. Thus, our scheme assures privacy against any coalition formed by a group $B \in \overline{\mathcal{A}}$ of tallying authorities and the central authority `A`;

- *Verifiability* - some verifications are made in the ballot tallying phase but some proofs have to be added at the voters' level. This includes the proof that a voter really chooses a vote in $\{v_{yes}, v_{no}\}$ and the proof that the sub-ballots are properly derived. For the second part, verifiable secret sharing may be used (see Section 3.3.3);

- *Robustness* - assuming that the voters' actions are performed honestly (or using verifiable secret sharing for detecting frauds), the election carries on correctly if at least a group $A \in \mathcal{A}$ of tallying authorities and the central authority $\mathtt{A}$ act honestly.

4.3.2 E-voting Schemes based on Homomorphic Encryption and Threshold Decryption

The main idea behind this e-voting technique, proposed by Cramer, Gennaro, and Schoenmakers [35] is that the voters use a public-key encryption method established by the tallying authorities for encrypting their votes such that the product of the encrypted votes is the encryption of the sum of the votes. The sum of the votes is obtained by the tallying authorities by threshold decryption.

We present next this scheme in more details. First we introduce homomorphic encryption functions.

Definition 4.3.1 Let $e_{\{\}}$ be a parameterized function, and $\oplus, \otimes$ be two binary operations over the domain, and, respectively, over the range of $e_{\{\}}$. We say that $e_{\{\}}$ is $(\oplus, \otimes)$- *homomorphic* if for any r_1, r_2 and x_1, x_2 there is an element r such that

$$e_{r_1}(x_1) \otimes e_{r_2}(x_2) = e_r(x_1 \oplus x_2).$$

ElGamal encryption method (see Appendix C), $e_r(x) = (\gamma, \delta)$ where $\gamma = \alpha^r \mathtt{~mod~} p$ and $\delta = x \cdot \beta^r \mathtt{~mod~} p$, for some $r \in \mathbf{Z}_q$, has homomorphic properties:

- $e_{r_1}(x_1) \cdot e_{r_2}(x_2) = e_{r_1+r_2}(x_1 \cdot x_2)$, where $(\gamma_1, \delta_1) \cdot (\gamma_2, \delta_2) = (\gamma_1\gamma_2, \delta_1\delta_2)$;

- $e_{r_1}(\overline{\alpha}^{v_1} \mathtt{~mod~} p) \cdot e_{r_2}(\overline{\alpha}^{v_2} \mathtt{~mod~} p) = e_{r_1+r_2}(\overline{\alpha}^{v_1+v_2 \mathtt{~mod~} q} \mathtt{~mod~} p)$, for any element $\overline{\alpha}$ of order q.

Now we can present the e-voting scheme. The parties involved are the voters $V_1, \ldots, V_m$ and the tallying authorities $A_1, \ldots, A_n$.

- The tallying authorities, using a dealer-free threshold variant of the ElGamal cryptosystem (see Section 4.2.1), generate and broadcast the following elements: p and q large primes such that $q \mid (p-1)$ and $m < \frac{q}{2}$, $\alpha, \overline{\alpha}$ elements of order q, and $\beta = \alpha^a \mathtt{~mod~} p$, where $a \in \mathbf{Z}_q$ is the secret key;

- The voter V_j broadcasts the ballot (γ_j, δ_j) corresponding to his vote $v_j \in \{-1, 1\}$, where $\gamma_j = \alpha^{r_j} \texttt{ mod } p$ and $\delta_j = (\overline{\alpha}^{v_j} \texttt{ mod } p) \cdot \beta^{r_j} \texttt{ mod } p$, for some $r_j \in \mathbf{Z}_q$, together with a non-interactive proof of validity (see Figure 4.2);

The Prover			The Verifier
$v = 1$	$v = -1$		
$r, w, r_1, d_1 \in \mathbf{Z}_q$	$r, w, r_2, d_2 \in \mathbf{Z}_q$		
$\gamma = \alpha^r,\ \delta = \beta^r\overline{\alpha}$	$\gamma = \alpha^r,\ \delta = \beta^r\overline{\alpha}^{-1}$		
$a_1 = \alpha^{r_1}\gamma^{d_1}$	$a_1 = \alpha^w$		
$b_1 = \beta^{r_1}(\delta\overline{\alpha})^{d_1}$	$b_1 = \beta^w$		
$a_2 = \alpha^w$	$a_2 = \alpha^{r_2}\gamma^{d_2}$		
$b_2 = \beta^w$	$b_2 = \beta^{r_2}(\delta\overline{\alpha}^{-1})^{d_1}$	$\xrightarrow{\gamma,\delta,a_1,b_1,a_2,b_2}$	
			$c \in \mathbf{Z}_q$
		$\xleftarrow{c}$	
$d_2 = c - d_1$	$d_1 = c - d_2$		
$r_2 = w - rd_2$	$r_1 = w - rd_1$	$\xrightarrow{d_1,d_2,r_1,r_2}$	
			$c \overset{?}{=} d_1 + d_2$
			$a_1 \overset{?}{=} \alpha^{r_1}\gamma^{d_1}, b_1 \overset{?}{=} \beta^{r_1}(\delta\overline{\alpha})^{d_1}$
			$a_2 \overset{?}{=} \alpha^{r_2}\gamma^{d_2}, b_2 \overset{?}{=} \beta^{r_2}(\delta\overline{\alpha}^{-1})^{d_2}$

Figure 4.2: Proof that (γ, δ) is the ElGamal encryption of $\overline{\alpha}^v$ for some $v \in \{-1, 1\}$ ([35])

- At the end of voting period, these proofs are verified by the tallying authorities and the pair $(\gamma, \delta) = (\prod_{j=1}^m \gamma_j \texttt{ mod } p, \prod_{j=1}^m \delta_j \texttt{ mod } p)$ is obtained (remark that (γ, δ) is the encryption of $\overline{\alpha}^{\sum_{j=1}^m v_j} \texttt{ mod } p$);
- Then, the tallying authorities can follow the threshold decryption protocol described in Section 4.2.1 for cooperatively computing $x = \overline{\alpha}^{\sum_{j=1}^m v_j} \texttt{ mod } p$ and, finally, obtain the final sum $S = \sum_{j=1}^m v_j$ as $\log_{\overline{\alpha}} x$ using, for example, Shanks' algorithm [135].

Properties of this e-voting scheme are

- Privacy - against coalitions of size $\leq k - 1$;
- Robustness - against coalitions of size $\leq n - k$;
- Public verifiability.

Chapter 5

Conclusions and Future Work

This thesis has focused on a very important cryptographic primitive - secret sharing schemes. A secret sharing scheme starts with a secret and then derives from it certain shares (or shadows) which are distributed to some users. The secret may be recovered only by certain predetermined groups which belong to the access structure. Secret sharing schemes have appeared as an elegant solution for the problem of safeguarding cryptographic keys but their applications include now threshold cryptographic protocols and some e-voting or e-auction protocols. We have reviewed the most important secret sharing schemes for different access structures (threshold, weighted, hierarchical, compartmented, general). Some very interesting and useful extended capabilities have been also surveyed so that the applications can be easily comprehensible.

Our major contribution consists in the application of the general variant of the Chinese remainder theorem in designing several classes (as threshold in Section 2.1.3, weighted threshold in Section 2.4, compartmented in Section 2.6, or even more general ones in Section 2.7.3) of secret sharing schemes and general information dispersal schemes (see Section 2.7.5). We consider that the proposed secret sharing schemes based on the Chinese remainder theorem provide the flexibility for performing a required compromise between the size of the shares and the level of security.

We have pointed out that the secret sharing schemes based on the general variant of the Chinese remainder theorem have some interesting and useful features as multiplicative and homomorphic properties (see Section 3.1) which make them suitable for threshold cryptography and e-voting protocols. We have presented such applications in Section 4.2.1, Section 4.2.2, and Section 4.3.1. The novelty of our multi-authority e-voting scheme presented in Section 4.3.1 is that the tallying authorities may have non-equal weights.

Another important contribution is represented by the general secret sharing schemes based on determinants presented in Section 2.7.4. These schemes provide some interesting changeability properties as deriving new shares from the old ones in case that some of the old shares have been compromised (in case of an explicit secret) or removing a user from all authorized groups to which he belongs.

Some interesting and promising future work directions are presented next:

1. Finding other classes of access structures which can be realized using our framework based on the Chinese remainder theorem. More exactly, we consider the problem of realizing hierarchical access structures;

2. It is interesting to remark that there are access structures whose realizations explicitly require the non-standard variant of the Chinese remainder theorem. The access structure given by $\mathcal{A}_{min} = \{\{1, 2\}, \{3, 4\}\}$ is such an example (see Section 2.7.3). It will be interesting to find other access structures with the same property or even find a general criterion for deciding if a certain access structure may be realized using the standard variant of the Chinese remainder theorem;

3. Verifiability for the schemes based on the Chinese remainder theorem represents another promising research direction. We have addressed briefly this topic in Section 3.3.3 but we think that there are better methods to provide verifiability for these schemes;

4. Finally, we consider that finding other methods of "puzzling" a matrix over an arbitrary commutative ring and forming the shares by partitioning the set of the resulted pieces represents also a good thinking subject.

Appendix A

Basic Elements of Number Theory

We present some basic facts and notations from number theory. For more details, the reader is referred to [32], [146]. Computational aspects can be found in [148].

Divisibility and congruences

Let $a, b \in \mathbf{Z}$, $b \neq 0$. The *quotient* of the integer division of a by b will be denoted by $a \ \mathtt{div}\ b$ and the *remainder* will be denoted by $a \ \mathtt{mod}\ b$. In the case $a \ \mathtt{mod}\ b = 0$ we will say that b is a *divisor* of a and we will denote this by $b|a$. An integer $p \geq 2$ that has only two positive divisors (1 and p) is called *prime*.

Let $a_1, \ldots, a_n \in \mathbf{Z}$, $a_1^2 + \cdots + a_n^2 \neq 0$. The *greatest common divisor* (*gcd*) of $a_1, \ldots, a_n$ will be denoted by $(a_1, \ldots, a_n)$. It is well-known that there exist $\alpha_1, \ldots, \alpha_n \in \mathbf{Z}$ that satisfy $\alpha_1 a_1 + \cdots + \alpha_n a_n = (a_1, \ldots, a_n)$ (the linear form of the *gcd*). Such combinations can be found using the extended Euclidean algorithm (see [148]). In case $(a_1, \ldots, a_n) = 1$, the numbers $a_1, \ldots, a_n$ are called *coprime*.

Let $a_1, \ldots, a_n \in \mathbf{Z}$ such that $a_1 \cdots a_n \neq 0$. The *least common multiple* (*lcm*) of $a_1, \ldots, a_n$ will be denoted by $[a_1, \ldots, a_n]$.

Definition A.1 Let $a, b, m \in \mathbf{Z}$. We say that a and b are *congruent modulo* m, and we will use the notation $a \equiv b \ mod\ m$, if $m|(a - b)$.

Remark A.1 In case that $m \neq 0$, the number $a \ \mathtt{mod}\ m$ can be seen also as the unique element r, $0 \leq r < |m|$ such that $a \equiv r \ mod\ m$. In some situations, we need to perform a centered modular reduction. More exactly, for an odd positive integer m and an arbitrary integer a, the unique element r, $-\lceil \frac{m}{2} \rceil \leq r \leq \lceil \frac{m}{2} \rceil$, such that $a \equiv r \ mod\ m$ will be denoted by[1] $a \ \mathsf{mod}\ m$.

[1] $a \ \mathsf{mod}\ m = \begin{cases} a \ \mathtt{mod}\ m, & \text{if } a \ \mathtt{mod}\ m \leq \lceil \frac{m}{2} \rceil; \\ (a \ \mathtt{mod}\ m) - m, & \text{otherwise.} \end{cases}$

The most important properties of the congruences are summarized in Proposition A.1.

Proposition A.1 Let $m, m' \geq 2$, and a, b, c, d be arbitrary integers.

1. If $a \equiv b \; mod \; m$ and $b \equiv c \; mod \; m$ then $a \equiv c \; mod \; m$ (transitivity);
2. If $a \equiv b \; mod \; m$ and $c \equiv d \; mod \; m$ then $a \oplus c \equiv b \oplus d \; mod \; m$, for any $\oplus \in \{+, -, \cdot\}$;
3. $a \equiv (a \; \mathtt{mod} \; m) \; mod \; m$;
4. The relation $a \equiv b \; mod \; m$ is equivalent with $ac \equiv bc \; mod \; mc$, providing that $c \neq 0$;
5. If $ac \equiv bc \; mod \; m$ then $a \equiv b \; mod \; \frac{m}{(m,c)}$. As a particular case, we obtain that $a \equiv b \; mod \; m$ if $ac \equiv bc \; mod \; m$ and $(m, c) = 1$;
6. If $a \equiv b \; mod \; m$ and $a \equiv b \; mod \; m'$ then $a \equiv b \; mod \; [m, m']$. As a particular case, we obtain that $a \equiv b \; mod \; mm'$ if $a \equiv b \; mod \; m$, $a \equiv b \; mod \; m'$, and $(m, m') = 1$.

$\mathbf{Z}_m$ denotes the set $\{0, 1, \ldots, m-1\}$, $\mathbf{Z}_m^*$ stands for the set $\{a \in \mathbf{Z}_m | (a, m) = 1\}$. In case that $a \in \mathbf{Z}_m^*$, for some $m \geq 2$, there is a unique element $b \in \mathbf{Z}_m^*$ such that $a \cdot b \equiv 1 \; mod \; m$. The element b is denoted by a^{-1} $(mod \; m)$ and is referred to as the *multiplicative inverse* of *a modulo m*.

Euler's totient function is given by $\phi(m) = |\mathbf{Z}_m^*|$, for all $m \geq 1$, and $\phi(0) = 0$. In case that m and m' are coprime then $\phi(m \cdot m') = \phi(m) \cdot \phi(m')$. Moreover, if q is a *prime power*, i.e., there is a prime p and a positive integer l such that $q = p^l$, then $\phi(q) = \phi(p^l) = p^l - p^{l-1}$. As a particular case, if p is a prime then $\phi(p) = p - 1$.

An important result is Euler's theorem, presented next.

Theorem A.1 (Euler's theorem)
Let $m \geq 2$ and $a \in \mathbf{Z}_m^*$. Then

$$a^{\phi(m)} \equiv 1 \; mod \; m.$$

Moreover, the previous result holds true for any positive integer a that is coprime to m. An important particular case is when m is a prime. In this case we obtain Fermat's little theorem.

Theorem A.2 (Fermat's little theorem)
Let p be a prime and a a positive integer such that $p \not| a$. Then

$$a^{p-1} \equiv 1 \; mod \; p.$$

Carmichael's reduced totient function is defined as

$$\lambda(m) = min(\{l \in \mathbf{N}^* | (\forall a \in \mathbf{Z}_m^*)(a^l \equiv 1 \ mod \ m)\}).$$

In case that m is prime then $\lambda(m) = m - 1$ and if $m = p \cdot q$, $(p, q) = 1$, then $\lambda(m) = \lambda(p \cdot q) = [\lambda(p), \lambda(q)]$.

The Chinese Remainder Theorem

The Chinese remainder theorem has many applications in computer science (see [53] for an interesting survey on this topic). We only mention the *RSA* decryption algorithm proposed by Quisquater and Couvreur [129], the discrete logarithm algorithm proposed by Pohlig and Hellman [128]. The Chinese remainder theorem has been also applied in secret sharing by Mignotte [117], Asmuth and Bloom [1].

The standard variant of the Chinese remainder theorem is the next one.

Theorem A.3 *Let $k \geq 2$, $m_1, \ldots, m_k \geq 2$, $b_1, \ldots, b_k \in \mathbf{Z}$. If $(m_i, m_j) = 1$ for all $1 \leq i < j \leq k$, then the system of equations*

$$\left\{ \begin{array}{ccl} x & \equiv & b_1 \ mod \ m_1 \\ & \vdots & \\ x & \equiv & b_k \ mod \ m_k \end{array} \right.$$

has a unique solution in $\mathbf{Z}_{m_1 \cdot \ldots \cdot m_k}$.

We will present next a more general variant of the Chinese remainder theorem, variant which has been used in our secret sharing schemes.

Theorem A.4 ([122]) *The system of equations*

$$\left\{ \begin{array}{ccl} x & \equiv & b_1 \ mod \ m_1 \\ & \vdots & \\ x & \equiv & b_k \ mod \ m_k \end{array} \right.$$

has solutions in $\mathbf{Z}$ if and only if $b_i \equiv b_j \ mod \ (m_i, m_j)$ for all $1 \leq i, j \leq k$. Moreover, if the above system of equations has solutions in $\mathbf{Z}$, then it has a unique solution in $\mathbf{Z}_{[m_1, \ldots, m_k]}$.

We briefly present two algorithms for the general Chinese remainder theorem. The reader is referred to [87] for a survey on this subject which includes proofs, implementation details, and test results.

The first one is due to Ore [122].

```
**CRT_Ore**$(\mathbf{b_1},\ldots,\mathbf{b_k},\mathbf{m_1},\ldots,\mathbf{m_k})$
input: $b_1,\ldots,b_k$, $m_1,\ldots,m_k \in \mathbf{Z}$ such that $b_i \equiv b_j\ mod\ (m_i,m_j)$, $\forall 1 \leq i < j \leq k$;
output: $x$, the unique solution modulo $[m_1,\ldots,m_k]$ of the above system;
begin
    for $i$:=1 to $k$ do $c_i := \frac{[m_1,\ldots,m_k]}{m_i}$; (remark that $(c_1,\ldots,c_k)=1$)
    find $\alpha_1,\ldots,\alpha_k \in \mathbf{Z}$ that satisfy $\alpha_1c_1+\cdots+\alpha_kc_k=1$;
    $x := (\alpha_1c_1b_1+\cdots+\alpha_kc_kb_k)$ mod $[m_1,\ldots,m_k]$;
end.
```

In case $(m_i,m_j)=1$, for all $1 \leq i < j \leq k$, we obtain Gauss' algorithm [66].

The second one is due to Fraenkel [61].

```
**CRT_Fraenkel**$(\mathbf{b_1},\ldots,\mathbf{b_k},\mathbf{m_1},\ldots,\mathbf{m_k})$
input: $b_1,\ldots,b_k$, $m_1,\ldots,m_k \in \mathbf{Z}$ such that $b_i \equiv b_j\ mod\ (m_i,m_j)$, $\forall 1 \leq i < j \leq k$;
output: $x$, the unique solution modulo $[m_1,\ldots,m_k]$ of the above system;
begin
    for $i$:=1 to $k-1$ do $c_i := [m_1,\ldots,m_i]$;
    $x := b_1$ mod $m_1$;
    for $i$:=1 to $k-1$ do
        begin
            $y := \frac{b_{i+1}-x}{(c_i,m_{i+1})} \cdot (\frac{c_i}{(c_i,m_{i+1})})^{-1}$ mod $\frac{m_{i+1}}{(c_i,m_{i+1})}$;
            $x := x + y \cdot c_i$;
        end
end.
```

In case $(m_i,m_j)=1$, for all $1 \leq i < j \leq k$, we obtain Garner's algorithm [65].

The order of an element. Primitive roots

We present next the notions of order and generator only for the group $\mathbf{Z}_m^*$ but these notions can be easily adapted for any other group.

Definition A.2 Let $m \geq 2$ and $a \in \mathbf{Z}_m^*$. The *order* of a modulo m, denoted by $ord_m(a)$, is defined as

$$ord_m(a) = min(\{l \in \mathbf{N}^* | a^l \equiv 1\ mod\ m\}).$$

The most important properties of the order of an element are summarized in Proposition A.2.

Proposition A.2 Let $m \geq 2$, $a \in \mathbf{Z}_m^*$, and k, l be arbitrary integers.

1. If $a^k \equiv 1\ mod\ m$ then $ord_m(a)|k$. As a particular case, we obtain that $ord_m(a)|\phi(m)$;
2. The relation $a^k \equiv a^l\ mod\ m$ is equivalent with $k \equiv l\ mod\ ord_m(a)$;

3. The elements $a^1 \texttt{ mod } m$, $a^2 \texttt{ mod } m$,...,$a^{ord_m(a)} \texttt{ mod } m$ are pairwise distinct;

4. The next relation holds true

$$ord_m(a^k \texttt{ mod } m) = \frac{ord_m(a)}{(ord_m(a), k)}.$$

Definition A.3 Let $m \geq 2$ and $\alpha \in \mathbf{Z}_m^*$. The element α is called *primitive root modulo m* if $ord_m(\alpha) = \phi(m)$.

Remark A.2 In case that α is a primitive root modulo m, every element β from the set $\mathbf{Z}_m^*$ can be uniquely expressed as $\beta = \alpha^i \texttt{ mod } m$, where $i \in \mathbf{Z}_{\phi(m)}$. The value i will be referred to as the *discrete logarithm* (*modulo m*) *to the base* α *of* β and we will write $i = log_\alpha \beta$. While an expression of form $\beta = \alpha^i \texttt{ mod } m$ can be efficiently computed given α, i, m (see, for example, [78]), the problem of finding the discrete logarithm modulo m to the base α of β, given α, β, m, is intractable.

The most important properties of the primitive roots are summarized in Proposition A.3.

Proposition A.3 Let $m \geq 2$ and $\alpha \in \mathbf{Z}_m^*$.

1. $\mathbf{Z}_m^*$ has primitive roots if and only if $m \in \{2, 4, p^k, 2p^k\}$, where p is an odd prime and $k \geq 1$ (Gauss' theorem);

2. If $\mathbf{Z}_m^*$ has primitive roots, then there are exactly $\phi(\phi(m))$ primitives roots modulo m;

3. If $\mathbf{Z}_m^*$ has primitive roots, then α is a primitive root modulo m if and only if the next relation holds true:

$$\alpha^{\frac{\phi(m)}{r}} \not\equiv 1 \; mod \; m,$$

   for any prime divisor r of $\phi(m)$.

Proposition A.3 (3) does not always allow to efficiently generate primitive roots because computing $\phi(m)$ and factoring $\phi(m)$ are intractable for large integers m. In practice, m is chosen such that computing $\phi(m)$ and factoring $\phi(m)$ can be easily performed. For example, m may be chosen as a *safe prime*, i.e., m is of form $m = 2q + 1$, where q is also a prime number. In this case, α is a primitive root modulo m if and only if $\alpha^2 \not\equiv 1 \; mod \; m$ and $\alpha^q \not\equiv 1 \; mod \; m$.

Elements of order q may be generated using primitive roots. More exactly, if p and q are odd primes such that $q|(p-1)$, α is a primitive root modulo p, and $\beta = \alpha^{\frac{p-1}{q}}$ $\texttt{mod}\ p$, then $ord_p(\beta) = q$. Indeed, by the Proposition A.2 (4), $ord_p(\beta) = ord_p(\alpha^{\frac{p-1}{q}}\ \texttt{mod}\ p) = \frac{ord_p(\alpha)}{(ord_p(\alpha), \frac{p-1}{q})} = \frac{p-1}{(p-1, \frac{p-1}{q})} = q$.

Appendix B

Basic Elements of Information Theory

We present some basic facts and notations from information theory. For more details, the reader is referred to [33].

Given a probability distribution[1] $(p_x|x \in \mathcal{X})$ over a finite set $\mathcal{X}$, the *entropy* of $\mathcal{X}$ (*with respect to* $(p_x|x \in \mathcal{X})$), denoted by $H(\mathcal{X})$, is defined as[2]

$$H(\mathcal{X}) = -\sum_{x \in \mathcal{X}} p_x log_2 p_x.$$

Entropy satisfy $0 \leq H(\mathcal{X}) \leq log_2|\mathcal{X}|$, where $H(\mathcal{X}) = 0$ when there is $x \in \mathcal{X}$ such that $p_x = 1$ and $H(\mathcal{X}) = log_2|\mathcal{X}|$ in the case of an *uniform distribution*, i.e., $p_x = \frac{1}{|\mathcal{X}|}$, for all $x \in \mathcal{X}$.

The entropy $H(\mathcal{X})$ measures the uncertainty one has about which element of the set $\mathcal{X}$ has been chosen when the picking of the elements of $\mathcal{X}$ is made according to $(p_x|x \in \mathcal{X})$. The entropy $H(\mathcal{X})$ is also a good approximation for the average number of bits required to represent the elements of the set $\mathcal{X}$.

Let $\mathtt{X}$ be a *random variable* over $\mathcal{X}$ with distribution $(p_x|x \in \mathcal{X})$. The value $H(\mathcal{X})$ will be also referred to as the *entropy of the random variable* $\mathtt{X}$ and will be denoted by $H(\mathtt{X})$.

The *conditional entropy* of the random variable $\mathtt{X}$ given the random variable $\mathtt{Y}$, denoted by $H(\mathtt{X}|\mathtt{Y})$, is defined as

$$H(\mathtt{X}|\mathtt{Y}) = -\sum_{y \in \mathcal{Y}} p_y H(\mathtt{X}|\mathtt{Y} = y),$$

where $H(\mathtt{X}|\mathtt{Y} = y)$ is the entropy corresponding to conditional probability

[1] A *probability distribution* over a finite set $\mathcal{X}$ is a sequence $(p_x|x \in \mathcal{X})$ of real numbers from the interval $[0, 1]$ such that $\sum_{x \in \mathcal{X}} p_x = 1$.

[2] We make the convention that $p_x log_2 p_x = 0$ in case that $p_x = 0$.

distribution $p_{\mathtt{X}|\mathtt{Y}=y}$. The conditional entropy satisfies $0 \leq H(\mathtt{X}|\mathtt{Y}) \leq H(\mathtt{X})$.

The difference $H(\mathtt{X}) - H(\mathtt{X}|\mathtt{Y})$ is referred to as the *mutual information* between $\mathtt{X}$ and $\mathtt{Y}$ and is denoted by $I(\mathtt{X};\mathtt{Y})$. The mutual information satisfies $0 \leq I(\mathtt{X};\mathtt{Y})$ and $I(\mathtt{X};\mathtt{Y}) = \mathtt{I}(\mathtt{Y};\mathtt{X})$.

Appendix C

Basic Elements of Cryptography

According to [114], *cryptography* is the study of mathematical techniques related to aspects of information security such as

- *confidentiality* (also referred to as *privacy* or *secrecy*) - keeping an information inaccessible to any unauthorized party;
- *information origin authentication* - assuring that the origin of some information is as declared;
- *entity authentication* (also referred to as *identification*) - assuring that the identity of some party is as declared;
- *data integrity* - preventing unauthorized modification of an information;
- *non-repudiation* - preventing the denial of previous actions.

The reader is referred to [95, 139] for some very interesting non-technical presentations of the evolution of this domain. We only mention here that some rudimentary cryptographic scheme have been used by the Egyptians 4000 years ago and that, later on, Caesar himself has used a very simple method for secret communications. Cryptography and cryptanalysis[1], which together form *cryptology*, have become very important during the world wars.

We will discuss briefly the most important cryptographic primitives. For more details, the reader is referred to [114] or [147].

[1] "*Cryptanalysis* is the study of mathematical techniques for attempting to defeat cryptographic techniques" [114].

Cryptosystems

Cryptosystems are basic cryptographic blocks designed to assure confidentiality. In communication between two parties over an insecure channel, the messages, called *plaintexts*, are *encrypted* by the sender using an established method. The obtained texts, called *cryptotexts*, are transmitted over the communication channel. The receiver performs the reverse operation, called *decryption*, and recovers the original messages. The encryption and decryption methods depend on some parameters called *keys*. The formal definition of a cryptosystem is presented next.

Definition C.1 A *cryptosystem* (*cipher*) is a system $\mathcal{S} = (\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ of non-empty finite sets, where

- $\mathcal{P}$ denotes the set of *plaintext symbols*;
- $\mathcal{C}$ denotes the set of *cryptotext symbols*;
- $\mathcal{K}$ denotes the set of *keys*;
- $\mathcal{E}$ denotes the set of *encryption methods*, $E = \{e_k | e_k : \mathcal{P} \to \mathcal{C}, k \in \mathcal{K}\}$;
- $\mathcal{D}$ denotes the set of *decryption methods*, $D = \{d_k | d_k : \mathcal{C} \to \mathcal{P}, k \in \mathcal{K}\}$, such that
$$(\forall k \in \mathcal{K})(\forall x \in \mathcal{P})(d_k(e_k(x)) = x).$$

In order to encrypt an arbitrary text using a cryptosystem, an encoding is applied to the initial text, obtaining a sequence of plaintext symbols $x = x_1 x_2 \cdots x_l$. The encryption of x with key k is obtained as $y = e_k(x_1)e_k(x_2)\cdots e_k(x_l)$.

Cryptosystems can be classified as:

- *symmetric* (*private-key*) *cryptosystems* (see Figure C.1)- characterized by the fact that the decryption method d_k can be easily computed from the encryption method e_k, and vice-versa. We mention here *DES* cryptosystem [60] or *AES* cryptosystem [59];
- *asymmetric* (*public-key*) *cryptosystems* (see Figure C.2) - characterized by the fact that the problem of computing the decryption method d_k, given the encryption method e_k, is intractable. Any key k has two components: k_e, for encryption, and k_d, for decryption. The encryption key can be made public without affecting the security of the cryptosystem. The security of the public-key cryptosystems relies on the intractability of some problems. We mention here *RSA* cryptosystem [132] based on the intractability of factoring large integers, ElGamal cryptosystem [55] based on intractability of computing discrete logarithms, Merkle-Hellman cryptosystem [116] based on intractability of the knapsack problem.

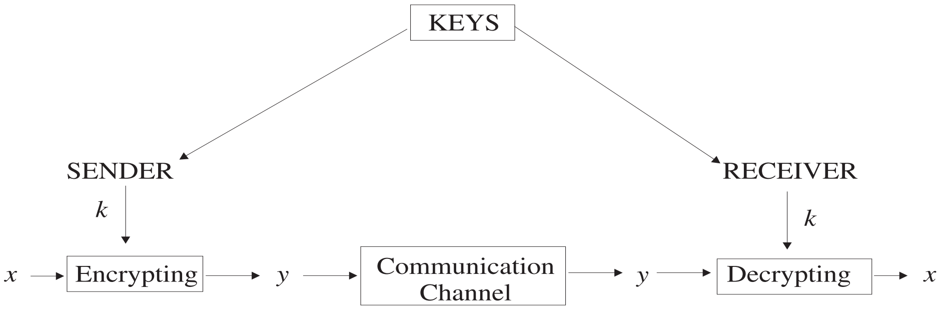

Figure C.1: Symmetric cryptosystem

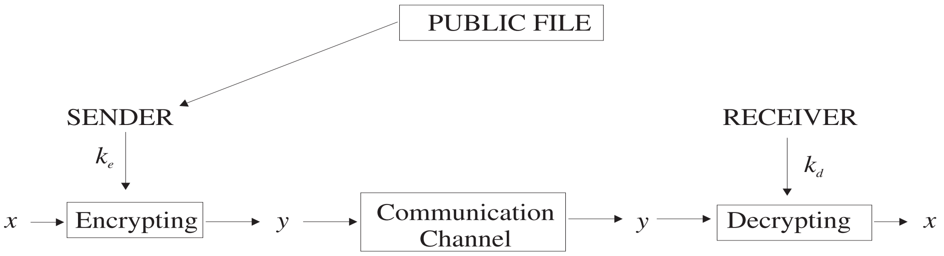

Figure C.2: Public-key cryptosystem

As opposed to public-key cryptosystems, any symmetric cryptosystem must be preceded by a *key establishment* protocol whereby a key becomes available to the parties involved. However, the symmetric cryptosystems have the advantage to be very efficiently implemented comparing to the asymmetric ones. Figure C.3 presents how the symmetric and the asymmetric cryptosystems can be combined in order to optimize the performance. A large data x is encrypted using a symmetric cryptosystem with a *session key* k_s (chosen by the sender) and the encryption of the session key using receiver's public key k_e is appended to the resulted cryptotext. The receiver recovers first the session key, using his private key k_d, and then the data x.

Public-key cryptography has been introduced by Merkle [115] and Diffie and Hellman [50]. Rivest, Shamir, and Adleman have proposed the realization of a public-key cryptosystem, known as the *RSA* cryptosystem:

- `encryption key`: $k_e = (N, e)$, where $N = p \cdot q$, p and q are distinct primes, and $e \in \mathbf{Z}^*_{\phi(N)}$;

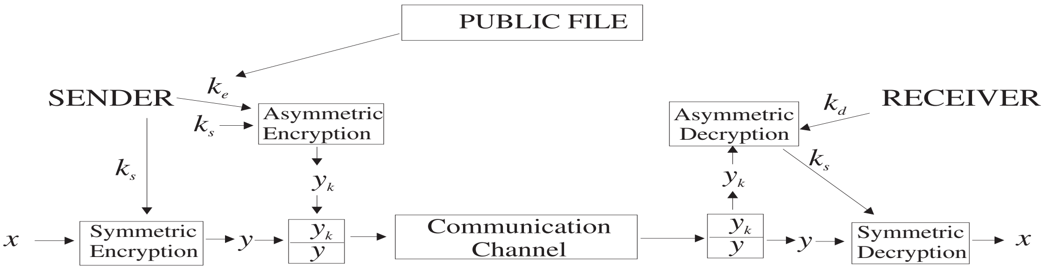

Figure C.3: Combination of symmetric/asymmetric cryptosystems

- `decryption key`: $k_d = (p, q, d)$, where d is a positive integer such that[2] $e \cdot d \equiv 1 \ mod \ \phi(N)$;
- `encryption`: a plaintext symbol $x \in \mathbf{Z}_N$ is encrypted as $y = x^e \ \mathtt{mod} \ N$;
- `decryption`: a cryptotext symbol $y \in \mathbf{Z}_N$ is decrypted as $x = y^d \ \mathtt{mod} \ N$.

The correctness of the cryptosystem is based on the fact that

$$x^{ed} \equiv x \ mod \ N,$$

for all $x \in \mathbf{Z}_N$ and e, d, N as above. The security of the *RSA* cryptosystem relies on the intractability of factoring. The prime numbers p and q must be chosen each on 512 bits. The public parameter e may be chosen, for example as $e = 2^{16} + 1$, in order to allow fast exponentiation, but the private parameter must be chosen large enough in order to avoid the small secret exponent attack due to Wiener [152].

We present next the ElGamal public-key cryptosystem [55].

- `decryption key`: $k_d = a$, where $a \in \mathbf{Z}_q^*$;
- `encryption key`: $k_e = (p, q, \alpha, \beta)$, where p is a large prime, such that $p - 1$ has a large prime divisor q, α is an element of order q, and $\beta = \alpha^a \ \mathtt{mod} \ p$;
- `encryption`: a plaintext symbol $x \in \mathbf{Z}_p$ is encrypted as a pair (γ, δ) where
  - $\gamma = \alpha^r \ \mathtt{mod} \ p$,

[2] The parameters e and d can also be chosen such that

$$e \cdot d \equiv 1 \ mod \ \lambda(N);$$

  - $\delta = x \cdot \beta^r \ \mathtt{mod}\ p$,

  where $r \in \mathbf{Z}_q^*$ is a parameter chosen by the sender;

- `decryption`: a cryptotext symbol $(\gamma, \delta) \in \mathbf{Z}_p \times \mathbf{Z}_p$ is decrypted as

$$(\gamma^a)^{-1} \cdot \delta \ \mathtt{mod}\ p.$$

The correctness of the cryptosystem can be easily justified as follows:

$$\gamma^{-a} \cdot \delta \ \mathtt{mod}\ p = \alpha^{-ar} \cdot (x \cdot (\alpha^a)^r) \ \mathtt{mod}\ p = x.$$

The security of the ElGamal cryptosystem relies on the intractability of computing discrete logarithms.

Digital Signatures Schemes

Digital signatures are the electronic equivalent of the handwritten signatures. A *digital signature* is attached to the message in order to prove its origin authentication. As opposed to the ordinary signature, the digital signature depends on the corresponding message. The digital signature also depends on a secret parameter (the *signing key*) known only by the signer. The digital signatures must be verifiable - thus, some information (the *verification key*) must be broadcasted in order to make the verification process possible. The formal definition of a digital signature scheme is presented next.

Definition C.2 A *digital signature scheme* is a system $\mathcal{S}ig = (\mathcal{P}, \mathcal{S}, \mathcal{K}, \mathcal{M}_s, \mathcal{M}_v)$ of non-empty finite sets, where

- $\mathcal{P}$ denotes the set of *signing symbols*;

- $\mathcal{S}$ denotes the set of *signature symbols*;

- $\mathcal{K}$ denotes the set of *keys*;

- $\mathcal{M}_s$ denotes the set of *signing methods*,
  $\mathcal{M}_s = \{sig_k | sig_k : \mathcal{P} \to \mathcal{S}, k \in \mathcal{K}\}$;

- $\mathcal{M}_v$ denotes the set of *verification methods*,
  $D = \{ver_k | ver_k : \mathcal{P} \times \mathcal{S} \to \{0, 1\}, k \in \mathcal{K}\}$ such that

$$(\forall k \in \mathcal{K})(\forall x \in \mathcal{P})(\forall y \in \mathcal{S})(ver_k(x, y) = 1 \Leftrightarrow sig_k(x) = y).$$

As we have discussed above, any key k has two components: k_s, for signing, which is private, and k_v, for verification, which is public.

We present next *RSA* digital signature scheme [132].

- `verification key`: $k_v = (N, e)$, where $N = p \cdot q$, p and q are distinct primes, and $e \in \mathbf{Z}^*_{\phi(N)}$;
- `signing key`: $k_s = (p, q, d)$, where d is a positive integer such that $e \cdot d \equiv 1 \; mod \; \phi(N)$;
- `signature generation`: the digital signature corresponding to a signing symbol $x \in \mathbf{Z}_N$ is $y = x^d \texttt{ mod } N$;
- `signature verification`: having a pair $(x, y) \in \mathbf{Z}_N \times \mathbf{Z}_N$, y is the correct signature with respect to x if and only if

$$x \overset{?}{=} y^e \texttt{ mod } N.$$

We will present next the Digital Signature Standard (DSS) [58]. We mention that our presentation differs by the original one only by switching r with r^{-1}. This is justified by the threshold versions of DSS (see Section 4.2.1).

- `signing key`: $k_s = a$, $a \in \mathbf{Z}^*_q$;
- `verification key`: $k_v = (p, q, \alpha, \beta)$ where p is a large prime, such that $p - 1$ has a large prime divisor q, α is an element of order q, and $\beta = \alpha^a \texttt{ mod } p$;
- `signature generation`: the digital signature corresponding to a signing symbol $x \in \mathbf{Z}^*_q$ is the pair (γ, δ), where
  - $\gamma = (\alpha^{r^{-1}} \texttt{ mod } p) \texttt{ mod } q$,
  - $\delta = r \cdot (x + a \cdot \gamma) \texttt{ mod } q$,

  where $r \in \mathbf{Z}^*_q$ is a parameter chosen by the signer;
- `signature verification`: having a pair $(x, (\gamma, \delta))$, the correctness of the signature (γ, δ) with respect to the message x can be verified by testing

$$\gamma \overset{?}{=} (\alpha^{x \cdot \delta^{-1}} \cdot \beta^{\gamma \cdot \delta^{-1}} \texttt{mod } p) \texttt{ mod } q.$$

  (all the operations from exponents are performed modulo q)

Indeed, if $\gamma = (\alpha^{r^{-1}} \texttt{ mod } p) \texttt{ mod } q$ and $\delta = r \cdot (x + a \cdot \delta) \texttt{ mod } q$ then

$$\begin{aligned}
(\alpha^{x \cdot \delta^{-1}} \cdot \beta^{\gamma \cdot \delta^{-1}} \texttt{mod } p) \texttt{ mod } q &= (\alpha^{x \cdot \delta^{-1}} \cdot \alpha^{a \cdot \gamma \cdot \delta^{-1}} \texttt{mod } p) \texttt{ mod } q \\
&= (\alpha^{\delta^{-1}(x + a \cdot \gamma)} \texttt{ mod } p) \texttt{ mod } q \\
&= (\alpha^{r^{-1}} \texttt{ mod } p) \texttt{ mod } q \\
&= \gamma.
\end{aligned}$$

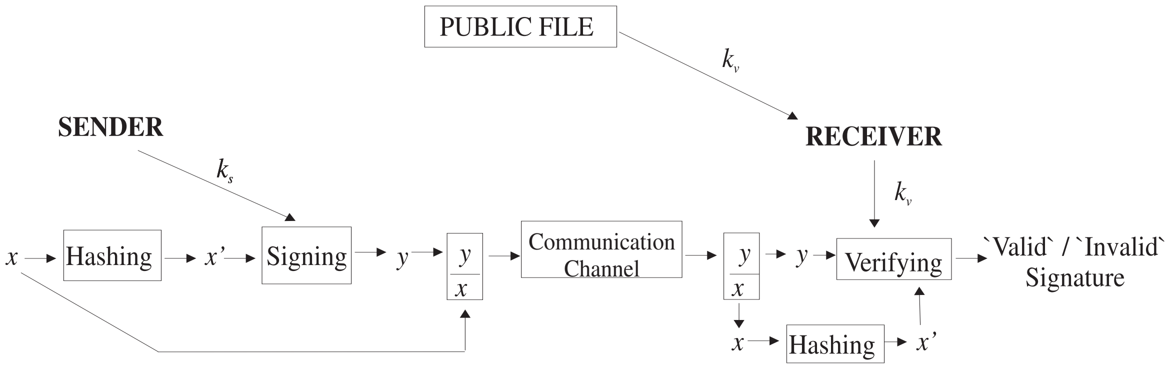

Figure C.4: Digital signatures and hash functions

By choosing p and q such that $|p| = 1024$ and $|q| = 160$, the digital signatures will be only 320 bits long.

Hash Functions

The digital signatures presented in the previous paragraph may be as long as the corresponding messages. Moreover, in case of a long message, the signing and the verification process may be very time-consuming. The idea is to sign a smaller amount of information, without compromising the security. A *hash function* h maps arbitrary binary strings to strings of some fixed length m. Thus, before signing a long message, a hash function is applied. The result, referred to as the *message digest*, is then signed. The combination of digital signature schemes and hash functions is presented in Figure C.4.

Depending on the intended application, a hash function $h : \cup_{i\geq 1}\mathbf{Z}_2^i \rightarrow \mathbf{Z}_2^m$ must satisfy certain security requirements:

1. Given $y \in \mathbf{Z}_2^m$, the problem of finding $x \in \cup_{i\geq 1}\mathbf{Z}_2^i$ such that $h(x) = y$ is intractable - functions that satisfy such requirements are called *one-way*[3] (or *preimage resistant*);

2. Given $x \in \cup_{i\geq 1}\mathbf{Z}_2^i$, the problem of finding $x' \in \cup_{i\geq 1}\mathbf{Z}_2^i$, $x' \neq x$, such that $h(x') = h(x)$ is intractable - functions that satisfy such requirements are called *weak collision resistant*;

[3] In general, the appellative *one-way* does not assume the compression property - an one-way function is a function $h : X \rightarrow Y$, such that given $y \in Y$, the problem of finding $x \in X$ such that $h(x) = y$ is intractable.

3. The problem of finding a *collision* of h, i.e., a pair (x, x'), $x, x' \in \cup_{i \geq 1} \mathbf{Z}_2^i$, $x' \neq x$, such that $h(x') = h(x)$, is intractable - functions that satisfy such requirements are called *strong collision resistant.*

An implicit requirement is that $h(x)$ is easy to compute, for any x. In order to avoid the birthday attack [114, Section 9.7.1], the size of a message digest has to be greater than 160 (bits).

There is a class of *keyed hash functions* (also referred to as *MACs* (message authentication codes)). MACs can be used for message origin authentication (as a symmetric technique) and data integrity.

Bibliography

[1] C. A. Asmuth and J. Bloom. A modular approach to key safeguarding. *IEEE Transactions on Information Theory*, IT-29(2):208–210, 1983.

[2] P. Béguin and A. Cresti. General short computational secret sharing schemes. In L. C. Guillou and J.-J. Quisquater, editors, *Advances in Cryptology - EUROCRYPT '95*, volume 921 of *Lecture Notes in Computer Science*, pages 194–208. Springer-Verlag, 1995.

[3] P. Béguin and A. Cresti. General information dispersal algorithms. *Theoretical Computer Science*, 209(1-2):87–105, 1998.

[4] A. Beimel, T. Tassa, and E. Weinreb. Characterizing ideal weighted threshold secret sharing. In J. Kilian, editor, *Theory of Cryptography, Second Theory of Cryptography Conference, TCC 2005, Cambridge, MA, USA, February 10-12, 2005*, volume 3378 of *Lecture Notes in Computer Science*, pages 600–619. Springer-Verlag, 2005.

[5] A. Beimel and E. Weinreb. Monotone circuits for monotone weighted threshold functions. *Information Processing Letters*, 97(1):12–18, 2006. (a preliminary version of this paper appeared in Proceedings of the 20th IEEE Conference on Computational Complexity, 2005).

[6] M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract). In *Proceedings of the Twentieth ACM Symposium on Theory of Computing, STOC '88*, pages 1–10, 1988.

[7] J. Benaloh. Secret sharing homomorphisms: Keeping shares of a secret secret. In A. M. Odlyzko, editor, *Advanced in Cryptology-CRYPTO' 86*, volume 263 of *Lecture Notes in Computer Science*, pages 251–260. Springer-Verlag, 1987.

[8] J. Benaloh. *Verifiable Secret-Ballot Elections.* PhD thesis, Department of Computer Science, Yale University, September 1987.

[9] J. Benaloh and J. Leichter. Generalized secret sharing and monotone functions. In S. Goldwasser, editor, *Advanced in Cryptology-CRYPTO'*

88, volume 403 of *Lecture Notes in Computer Science*, pages 27–35. Springer-Verlag, 1989.

[10] A. Beutelspacher. How to say “no”. In J.-J. Quisquater and J. Vandewalle, editors, *Advances in Cryptology - EUROCRYPT ’89*, volume 434 of *Lecture Notes in Computer Science*, pages 491–496. Springer-Verlag, 1990.

[11] B. Blakley, G. R. Blakley, A. H. Chan, and J. L. Massey. Threshold schemes with disenrollment. In E. F. Brickell, editor, *Advances in Cryptology - CRYPTO ’92*, volume 740 of *Lecture Notes in Computer Science*, pages 540–548. Springer-Verlag, 1993.

[12] G. R. Blakley. Safeguarding cryptographic keys. In *National Computer Conference, 1979*, volume 48 of *American Federation of Information Processing Societies Proceedings*, pages 313–317, 1979.

[13] G. R. Blakley and C. Meadows. Security of ramp schemes. In G. R. Blakley and D. Chaum, editors, *Advances in Cryptology-CRYPTO ’84*, volume 196 of *Lecture Notes in Computer Science*, pages 242–268. Springer-Verlag, 1985.

[14] C. Blundo. *Secret Sharing Schemes for Access Structures based on Graphs.* PhD thesis, University of Salerno, 1991. (in Italian).

[15] C. Blundo, A. De Santis, L. Gargano, and U. Vaccaro. Secret sharing schemes with veto capabilities. In G. Cohen, S. Litsyn, A. Lobstein, and G. Zemor, editors, *Algebraic Coding*, volume 781 of *Lecture Notes in Computer Science*, pages 82–89. Springer-Verlag, 1993.

[16] C. Blundo, A. De Santis, L. Gargano, and U. Vaccaro. On the information rate of secret sharing schemes. *Theoretical Computer Science*, 154(2):283–306, 1996. (a preliminary version of this paper appeared in “Advances in Cryptology – CRYPTO ’92”, E. F. Brickell, ed., Lecture Notes in Computer Science 740 (1993), 148–167).

[17] C. Blundo, A. De Santis, D. R. Stinson, and U. Vaccaro. Graph decompositions and secret sharing schemes. *Journal of Cryptology*, 8(1):39–64, 1995. (a preliminary version of this paper appeared in “Advances in Cryptology - EUROCRYPT ’92”, R. A. Rueppel, ed., Lecture Notes in Computer Science 658 (1993), 1–24).

[18] C. Blundo, A. De Santis, and U. Vaccaro. Efficient sharing of many secrets. In P. Enjalbert, A. Finkel, and K. W. Wagner, editors, *STACS ’93, 10th Annual Symposium on Theoretical Aspects of Computer Science*, volume 665 of *Lecture Notes in Computer Science*, pages 692–703. Springer-Verlag, 1993.

[19] C. Boyd. Digital multisignatures. In H. Beker and F. Piper, editors, *Cryptography and Coding, 1986*, pages 241–246. Oxford University Press, 1989.

[20] E. F. Brickell. Some ideal secret sharing schemes. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 6:105–113, 1989. (a preliminary version of this paper appeared in “Advances in Cryptology – EUROCRYPT ’89”, J.-J. Quisquater and J. Vandewalle, eds., Lecture Notes in Computer Science 434 (1990), 468-475).

[21] E. F. Brickell, G. Di Crescenzo, and Y. Frankel. Sharing block ciphers. In E. Dawson, A. Clark, and C. Boyd, editors, *Information Security and Privacy, 5th Australasian Conference, ACISP 2000*, volume 1841 of *Lecture Notes in Computer Science*, pages 457–470. Springer-Verlag, 2000.

[22] E. F. Brickell and D. M. Davenport. On the classification of ideal secret sharing schemes. *Journal of Cryptology*, 4(2):123–134, 1991. (a preliminary version of this paper appeared in “Advances in Cryptology” - CRYPTO ’89”, G. Brassard, ed., Lecture Notes in Computer Science 435 (1990), 278-285).

[23] E. F. Brickell and D. R. Stinson. Some improved bounds on the information rate of perfect secret sharing schemes. *Journal of Cryptology*, 5(3):153–166, 1992. (a preliminary version of this paper appeared in “Advances in Cryptology - CRYPTO ’90”, A. J. Menezes and S. A. Vanstone, eds., Lecture Notes in Computer Science 537 (1991), 242-252).

[24] R. L. Burden and J. D. Faires. *Numerical Analysis.* Brooks-Cole Publishing, seventh edition, 2001.

[25] M. Burmester. Homomorphisms of secret sharing schemes: A tool for verifiable signature sharing. In U. M. Maurer, editor, *Advances in Cryptology - EUROCRYPT ’96*, volume 1070 of *Lecture Notes in Computer Science*, pages 96–106. Springer-Verlag, 1996.

[26] C. Cachin. On-line secret sharing. In C. Boyd, editor, *Cryptography and Coding, 5th IMA Conference*, volume 1025 of *Lecture Notes in Computer Science*, pages 190–198. Springer-Verlag, 1995.

[27] R. M. Capocelli, A. De Santis, L. Gargano, and U. Vaccaro. On the size of shares for secret sharing schemes. *Journal of Cryptology*, 6(3):157–167, 1993. (a preliminary version of this paper appeared in “Advances in Cryptology – CRYPTO ’91”, J. Feigenbaum, ed., Lecture Notes in Computer Science 576 (1992), 101–113).

[28] T.-Y. Chang, M.-S. Hwang, and W.-P. Yang. A new multi-stage secret sharing scheme using one-way function. *Operating Systems Review*, 39(1):48–55, 2005.

[29] C. Charnes, J. Pieprzyk, and R. Safavi-Naini. Conditionally secure secret sharing schemes with disenrollment capability. In *ACM Conference on Computer and Communications Security*, pages 89–95, 1994.

[30] D. Chaum and T. P. Pedersen. Wallet databases with observers. In E. F. Brickell, editor, *Advances in Cryptology - CRYPTO '92*, volume 740 of *Lecture Notes in Computer Science*, pages 89–105. Springer-Verlag, 1993.

[31] B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch. Verifiable secret sharing and achieving simultaneity in the presence of faults. In *Proceedings of the 26th IEEE Symposium on the Foundations of Computer Science*, pages 383–395. IEEE Press, 1985.

[32] H. Cohen. *A Course in Computational Algebraic Number Theory.* Graduate Texts in Mathematics. Springer-Verlag, 4th edition, 2000.

[33] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* John Wiley & Sons, 1991.

[34] R. Cramer, M. K. Franklin, B. Schoenmakers, and M. Yung. Multi-authority secret-ballot elections with linear work. In U. Maurer, editor, *Advances in Cryptology - EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 72–83. Springer-Verlag, 1996.

[35] R. Cramer, R. Gennaro, and B. Schoenmakers. A secure and optimally efficient multi-authority election scheme. In W. Fumy, editor, *Advances in Cryptology - EUROCRYPT '97*, volume 1233 of *Lecture Notes in Computer Science*, pages 103–118. Springer-Verlag, 1997.

[36] G. Di Crescenzo. Sharing one secret vs. sharing many secrets. *Theoretical Computer Science*, 1-3:123–140, 2003.

[37] R. A. Croft and S. P. Harris. Public-key cryptography and re-usable shared secrets. In H. Beker and F. Piper, editors, *Cryptography and Coding, 1986*, pages 189–201. Oxford University Press, 1989.

[38] L. Csirmaz. The size of a share must be large. *Journal of Cryptology*, 10(4):223–231, 1997. (a preliminary version of this paper appeared in "Advances in Cryptology – EUROCRYPT '94", A. De Santis, ed., Lecture Notes in Computer Science 950 (1995), 13–22).

[39] I. Damgård and K. Dupont. Efficient threshold RSA signatures with general moduli and no extra assumptions. In S. Vaudenay, editor,

Public Key Cryptography - PKC 2005, 8th International Workshop on Theory and Practice in Public Key Cryptography, volume 3386 of *Lecture Notes in Computer Science*, pages 346–361. Springer, 2005.

[40] A. De Santis, Y. Desmedt, Y. Frankel, and M. Yung. How to share a function securely. In *Twenty-Sixth Annual ACM Symposium on Theory of Computing, STOC'94*, pages 522–533. ACM Press, 1994.

[41] M. De Soete, J.-J. Quisquater, and K. Vedder. A signature with shared verification scheme. In G. Brassard, editor, *Advances in Cryptology - CRYPTO '89*, volume 435 of *Lecture Notes in Computer Science*, pages 253–262. Springer-Verlag, 1990.

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

[43] Y. Desmedt. Threshold cryptosystems (invited talk). In J. Seberry and Y. Zheng, editors, *Advances in Cryptology - ASIACRYPT '92*, volume 718 of *Lecture Notes in Computer Science*, pages 3–14. Springer-Verlag, 1993.

[44] Y. Desmedt. Some recent research aspects of threshold cryptography. In E. Okamoto, G. I. Davida, and M. Mambo, editors, *ISW '97: Proceedings of the First International Workshop on Information Security*, volume 1396 of *Lecture Notes in Computer Science*, pages 158–173. Springer-Verlag, 1998.

[45] Y. Desmedt, G. Di Crescenzo, and M. Burmester. Multiplicative nonabelian sharing schemes and their applications to threshold cryptography. In J. Pieprzyk and R. Safavi-Naini, editors, *Advances in Cryptology - ASIACRYPT '94*, volume 917 of *Lecture Notes in Computer Science Volume*, pages 21–32. Springer-Verlag, 1995.

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

[47] Y. Desmedt and Y. Frankel. Shared generation of authenticators and signatures. In J. Feigenbaum, editor, *Advances in Cryptology - CRYPTO '91*, volume 576 of *Lecture Notes in Computer Science*, pages 457–469. Springer-Verlag, 1992.

[48] Y. Desmedt and Y. Frankel. Perfect homomorphic zero-knowledge threshold schemes over any finite abelian group. *SIAM Journal on Discrete Mathematics*, 7(4):667–679, 1994. (a preliminary version of

this paper appeared in “Sequences II: Methods in Communication, Security and Computer Science”, R. Capocelli and A. De Santis, eds., Springer Verlag, 1993, 369-378).

[49] Y. Desmedt and S. Jajodia. Redistributing secret shares to new access structures and its applications. Technical Report ISSE TR-97-01, George Mason University, Fairfax, July 1997.

[50] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654, 1976.

[51] M. van Dijk. A linear construction of secret sharing schemes. *Designs, Codes and Cryptography*, 12(2):161–201, 1997. (a preliminary version of this paper appeared in “Advances in Cryptology – EUROCRYPT ’94”, A. De Santis, ed., Lecture Notes in Computer Science 950 (1995), 23–34).

[52] M. van Dijk, W.-A. Jackson, and K. M. Martin. A note on duality in linear secret sharing schemes. *Bulletin of the Institute of Combinatorics and its Applications*, 19:93–101, 1997.

[53] C. Ding, D. Pei, and A. Salomaa. *Chinese remainder theorem: applications in computing, coding, cryptography.* World Scientific Publishing, 1996.

[54] W. Eberly, M. Giesbrecht, and G. Villard. On computing the determinant and Smith form of an integer matrix. In *The 41st Annual Symposium on Foundations of Computer Science*, pages 675–685. IEEE Computer Society, 2000.

[55] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31:469–472, 1985. (a preliminary version of this paper appeared in “Advances in Cryptology – CRYPTO ’84”, G. R. Blakley and D. Chaum, eds., Lecture Notes in Computer Science 196 (1985), 10-18).

[56] P. Feldman. A practical scheme for non-interactive verifiable secret sharing. In *Proceedings of the 28th IEEE Symposium on the Foundations of Computer Science, 1987*, pages 427–437. IEEE Press, 1987.

[57] A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In A. M. Odlyzko, editor, *Advances in Cryptology - CRYPTO ’86*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194. Springer-Verlag, 1987.

[58] FIPS 186-2, ”Digital Signature Standard”, Federal Information Processing Standards Publication 186, January 27, 2000.

(available at http://csrc.nist.gov/publications/fips/fips186-2/fips186-2-change1.pdf).

[59] FIPS 197, "Advanced Encryption Standard", Federal Information Processing Standards Publication 197, November 26, 2001. (available at http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf).

[60] FIPS 46-3, "Data Encryption Standard", Federal Information Processing Standards Publication 46-3, October 25, 1999. (available at http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf)-withdrawn on May 19, 2005.

[61] A. S. Fraenkel. New proof of the generalized Chinese remainder theorem. *Proceedings of American Mathematical Society*, 14(5):790–791, 1963.

[62] Y. Frankel and Y. Desmedt. Classification of ideal homomorphic threshold schemes over finite abelian groups (extended abstract). In R. A. Rueppel, editor, *Advances in Cryptology - EUROCRYPT '92*, volume 658 of *Lecture Notes in Computer Science*, pages 25–34. Springer-Verlag, 1993.

[63] M. K. Franklin and M. K. Reiter. The design and implementation of a secure auction service. *IEEE Transactions on Software Engineering*, 22(5):302–312, 1996.

[64] M. K. Franklin and M. Yung. Communication complexity of secure computation (extended abstract). In *Proceedings of the Twenty Fourth Annual ACM Symposium on Theory of Computing*, pages 699–710. ACM, 1992.

[65] H. Garner. The residue number system. *IRE Transactions on Electronic Computers*, EC-8:140–147, 1959.

[66] C.F. Gauss. *Disquisitiones Arithmeticae.* 1801. English translation by A. Clarke, Springer-Verlag 1986.

[67] R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin. Robust threshold DSS signatures. *Information and Computation*, 164(1):54–84, 2001. (a preliminary version of this paper appeared in "Advances in Cryptology – EUROCRYPT '96", U. M. Maurer, ed., Lecture Notes in Computer Science 1070 (1996), 354–371).

[68] R. Gennaro, M. O. Rabin, and T. Rabin. Simplified VSS and fast-track multiparty computations with applications to threshold cryptography. In *PODC '98: Proceedings of the seventeenth ACM symposium on Principles of distributed computing*, pages 101–111. ACM Press, 1998.

[69] H. Ghodosi, J. Pieprzyk, and R. Safavi-Naini. Remarks on the multiple assignment secret sharing scheme. In Y. Han, T. Okamoto, and S. Qing, editors, *ICICS '97: First International Conference on Information and Communication Security*, volume 1334 of *Lecture Notes in Computer Science*, pages 72–80. Springer-Verlag, 1997.

[70] H. Ghodosi, J. Pieprzyk, and R. Safavi-Naini. Secret sharing in multilevel and compartmented groups. In C. Boyd and E. Dawson, editors, *ACISP '98: Proceedings of the Third Australasian Conference on Information Security and Privacy*, volume 1438 of *Lecture Notes in Computer Science*, pages 367–378. Springer-Verlag, 1998.

[71] H. Ghodosi, J. Pieprzyk, R. Safavi-Naini, and H. Wang. On construction of cumulative secret sharing schemes. In C. Boyd and E. Dawson, editors, *ACISP '98: Proceedings of the Third Australasian Conference on Information Security and Privacy*, volume 1438 of *Lecture Notes in Computer Science*, pages 379–390. Springer-Verlag, 1998.

[72] O. Goldreich, D. Ron, and M. Sudan. Chinese remaindering with errors. *IEEE Transactions on Information Theory*, IT-46(4):1330–1338, 2000. (a preliminary version of this paper appeared in Proceedings of the thirty-first annual ACM symposium on Theory of computing (1999), 225-234).

[73] D. Gritzalis, editor. *Secure Electronic Voting*, volume 7 of *Advances in Information Security.* Kluwer Academic Publishers, 2003.

[74] L. Harn. Comment on "Multistage secret sharing based on one-way function". *Electronics Letters*, 31(4):262, 1995.

[75] J. He and E. Dawson. Multistage secret sharing based on one-way function. *Electronics Letters*, 30(19):1591–1592, 1994.

[76] A. Herzberg, S. Jarecki, H. Krawczyk, and M. Yung. Proactive secret sharing or: How to cope with perpetual leakage. In D. Coppersmith, editor, *Advances in Cryptology - CRYPTO '95*, volume 963 of *Lecture Notes in Computer Science*, pages 339–352. Springer-Verlag, 1995.

[77] T. Hwang. Cryptosystem for group oriented cryptography. In I. Damgård, editor, *Advances in Cryptology - EUROCRYPT '90*, volume 473 of *Lecture Notes in Computer Science*, pages 352–360. Springer-Verlag, 1991.

[78] S. Iftene. Modular exponentiation. *International Scientific Journal of Computing*, 2(3), 2003. (a preliminary version of this paper was presented in CIPC 2003, Sinaia, Romania).

[79] S. Iftene. A generalization of Mignotte's secret sharing scheme. In T. Jebelean, V. Negru, D. Petcu, and D. Zaharie, editors, *Proceedings of the 6th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, Timisoara, Romania, September 2004*, pages 196–201. Mirton Publishing House, 2004.

[80] S. Iftene. Compartmented secret sharing based on the Chinese remainder theorem. Cryptology ePrint Archive, Report 2005/408, 2005. (available at http://eprint.iacr.org/) (a newer version of this paper appeared in [85]).

[81] S. Iftene. General secret sharing based on determinants. In T. Jebelean, V. Negru, D. Petcu, and D. Zaharie, editors, *Proceedings of the 7th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, Timisoara, Romania, September 2005*, pages 154–157. IEEE Computer Society Press, 2005.

[82] S. Iftene. Threshold RSA based on the general Chinese remainder theorem. Technical Report TR 05-05, "Al.I.Cuza" University of Iaşi, Faculty of Computer Science, 2005. (available at http://www.infoiasi.ro/~tr/tr.pl.cgi).

[83] S. Iftene. General information dispersal based on the Chinese remainder theorem. In I. Dziţac, F.-G. Filip, and M.-J. Manolescu, editors, *ICCCC 2006, International Conference On Computers, Communications&Control, Oradea, Romania, June 2006*, pages 274–279, 2006. (Supplementary issue of International Journal of Computers, Communications&Control).

[84] S. Iftene. General secret sharing based on the Chinese remainder theorem. Cryptology ePrint Archive, Report 2006/166, 2006. (available at http://eprint.iacr.org/) (a newer version of this paper, not a superset, appeared in [85]).

[85] S. Iftene. General secret sharing based on the Chinese remainder theorem with applications in E-voting. In C. Dima, M. Minea, and F. L. Ţiplea, editors, *ICS 2006, International Workshop on Information and Computer Security, Timisoara, Romania*, September, 2006. (this paper will be electronically published in Electronic Notes in Theoretical Computer Science (URL: www.elsevier.com/locate/entcs)) (preliminary versions of some parts of this paper appeared in [84], [80]).

[86] S. Iftene and I. Boureanu. Weighted threshold secret sharing based on the Chinese remainder theorem. *Scientific Annals of the "Al. I. Cuza" University of Iaşi, Computer Science Section*, XV:161–172, 2005.

[87] S. Iftene and F. Chelaru. The general Chinese remainder theorem, 2006. (submitted).

[88] I. Ingemarsson and G. J. Simmons. A protocol to set up shared secret schemes without the assistance of mutually trusted party. In I. Damgård, editor, *Advances in Cryptology - EUROCRYPT '90*, volume 473 of *Lecture Notes in Computer Science*, pages 266–282. Springer-Verlag, 1991.

[89] K. Itakura and K. Nakamura. A public key cryptosystem suitable for digital multisignatures. *NEC Research & Development*, 71:1–8, 1983.

[90] M. Ito, A. Saito, and T. Nishizeki. Secret sharing scheme realizing general access structure. In *Proceedings of the IEEE Global Telecommunications Conference, Globecom '87*, pages 99–102. IEEE Press, 1987.

[91] W.-A. Jackson and K. M. Martin. Cumulative arrays and geometric secret sharing schemes. In J. Seberry and Y. Zheng, editors, *Advances in Cryptology - ASIACRYPT '92*, volume 718 of *Lecture Notes in Computer Science*, pages 48–55. Springer-Verlag, 1993.

[92] W.-A. Jackson and K. M. Martin. Geometric secret sharing schemes and their duals. *Designs, Codes and Cryptography*, 4(1):83–95, 1994.

[93] W.-A. Jackson and K. M. Martin. A combinatorial interpretation of ramp schemes. *Australasian Journal of Combinatorics*, 14:51–60, 1996.

[94] W.-A. Jackson, K. M. Martin, and C. M. O'Keefe. Mutually trusted authority-free secret sharing schemes. *Journal of Cryptology*, 10(4):261–289, 1997. (a preliminary version of this paper appeared in "Advances in Cryptology – EUROCRYPT '95", L. C. Guillou and J.-J. Quisquater, eds., Lecture Notes in Computer Science 921 (1995), 183–193).

[95] D. Kahn. *The Codebreakers: The Comprehensive History of Secret Communication from Ancient Times to the Internet.* Scribner, 1996.

[96] E. Kaltofen. On computing determinants of matrices without divisions. In P. S. Wang, editor, *International Symposium on Symbolic and Algebraic computation (ISSAC'92)*, pages 342–349. ACM Press, 1992.

[97] E. D. Karnin, J. W. Greene, and M. E. Hellman. On secret sharing systems. *IEEE Transactions on Information Theory*, IT-29(1):35–41, 1983.

[98] S. C. Kothari. Generalized linear threshold scheme. In G. R. Blakley and D. Chaum, editors, *Advances in Cryptology-CRYPTO '84*, volume 196 of *Lecture Notes in Computer Science*, pages 231–241. Springer-Verlag, 1985.

[99] E. Kranakis. *Primality and Cryptography.* Wiley-Teubner Series in Computer Science, 1986.

[100] H. Krawczyk. Secret sharing made short. In D. R. Stinson, editor, *Advances in Cryptology - CRYPTO '93*, volume 773 of *Lecture Notes in Computer Science*, pages 136–146. Springer-Verlag, 1994.

[101] S. K. Langford. Threshold DSS signatures without a trusted party. In D. Coppersmith, editor, *Advances in Cryptology - CRYPTO '95*, volume 963 of *Lecture Notes in Computer Science*, pages 397–409. Springer-Verlag, 1995.

[102] Q. Li, Z. Wang, X. Niu, and S. Sun. A non-interactive modular verifiable secret sharing scheme. In *ICCCAS'05, International Conference on Communications, Circuits and Systems, 2005*, pages 84–87, 2005.

[103] S. Long, J. Pieprzyk, H. Wang, and D. S. Wong. Generalised cumulative arrays in secret sharing. *Designs, Codes and Cryptography*, 40(2):191–209, 2006.

[104] K. M. Martin. *Discrete Structures in the Theory of Secret Sharing.* PhD thesis, Royal Holloway and Bedford New College, University of London, 1991.

[105] K. M. Martin. New secret sharing schemes from old. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 14:65–77, 1993.

[106] K. M. Martin. Untrustworthy participants in perfect secret sharing schemes. In M. J. Ganley, editor, *Cryptography and Coding III*, pages 255–264. Oxford University Press, 1993.

[107] K. M. Martin, J. Pieprzyk, R. Safavi-Naini, and H. Wang. Changing thresholds in the absence of secure channels. In J. Pieprzyk, R. Safavi-Naini, and J. Seberry, editors, *Information Security and Privacy, 4th Australasian Conference, ACISP'99*, volume 1587 of *Lecture Notes in Computer Science*, pages 177–191. Springer-Verlag, 1999.

[108] K. M. Martin, J. Pieprzyk, R. Safavi-Naini, H. Wang, and P. R. Wild. Threshold MACs. In P. J. Lee and C. H. Lim, editors, *Information Security and Cryptology - ICISC 2002*, volume 2587 of *Lecture Notes in Computer Science*, pages 237–252. Springer-Verlag, 2003.

[109] K. M. Martin, R. Safavi-Naini, and H. Wang. Bounds and techniques for efficient redistribution of secret shares to new access structures. *The Computer Journal*, 42(8):638–649, 1999.

[110] K. M. Martin, R. Safavi-Naini, H. Wang, and P. R. Wild. Distributing the encryption and decryption of a block cipher. *Designs, Codes and Cryptography*, 36(3):263–287, 2005.

[111] E. Martinez-Moro, J. Mozo-Fernandez, and C. Munuera. Compounding secret sharing schemes. Cryptology ePrint Archive, Report 2003/048, 2003. (available at http://eprint.iacr.org/).

[112] R. J. McEliece and D. V. Sarwate. On sharing secrets and Reed-Solomon codes. *Communications of ACM*, 24(9):583–584, 1981.

[113] C. Meadows. Some threshold schemes without central key distributors. *Congressus Numerantium*, 46:187–199, 1985.

[114] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography*, volume 6 of *Discrete Mathematics and Its Applications*. CRC Press, 1996.

[115] R. C. Merkle. Secure communications over insecure channels. *Communications of the ACM*, 21(4):294–299, 1978. (submitted in 1975).

[116] R. C. Merkle and M. E. Hellman. Hiding information and signatures in trapdoor knapsacks. *IEEE Transactions on Information Theory*, 24(5):525–530, 1978.

[117] M. Mignotte. How to share a secret. In T. Beth, editor, *Cryptography-Proceedings of the Workshop on Cryptography, Burg Feuerstein, 1982*, volume 149 of *Lecture Notes in Computer Science*, pages 371–375. Springer-Verlag, 1983.

[118] P. Morillo, C. Padró, G. Sáez, and J. L. Villar. Weighted threshold secret sharing schemes. *Information Processing Letters*, 70(5):211–216, 1999.

[119] S. Obana and K. Kurosawa. Veto is impossible in secret sharing schemes. *Information Processing Letters*, 58(6):293–295, 1996.

[120] W. Ogata, K. Kurosawa, and D. R. Stinson. Optimum secret sharing scheme secure against cheating. *SIAM Journal on Discrete Mathematics*, 20(1):79–95, 2006. (a preliminary version of this paper appeared in “Advances in Cryptology – EUROCRYPT ’96”, U. M. Maurer, ed., Lecture Notes in Computer Science 1070 (1996), 200–211).

[121] W. Ogata, K. Kurosawa, and S. Tsujii. Nonperfect secret sharing schemes. In J. Seberry and Y. Zheng, editors, *Advances in Cryptology - ASIACRYPT '92*, volume 718 of *Lecture Notes in Computer Science*, pages 56–66. Springer-Verlag, 1993.

[122] O. Ore. The general Chinese remainder theorem. *American Mathematical Monthly*, 59:365–370, 1952.

[123] W. Patterson. *Mathematical cryptology for computer scientists and mathematicians*. Rowman & Littlefield, 1987.

[124] T. P. Pedersen. Distributed provers with applications to undeniable signatures. In D. W. Davies, editor, *Advances in Cryptology - EUROCRYPT '91*, volume 547 of *Lecture Notes in Computer Science*, pages 221–242. Springer-Verlag, 1991.

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

[126] T. P. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In J. Feigenbaum, editor, *Advances in Cryptology - CRYPTO '91*, volume 576 of *Lecture Notes in Computer Science*, pages 129–140. Springer-Verlag, 1992.

[127] R. G. E. Pinch. On-line multiple secret sharing. *Electronic Letters*, 32:1087–1088, 1996.

[128] S. C. Pohlig and M. E. Hellman. An improved algorithm for computing logarithms over GF(p) and its cryptographic significance. *IEEE Transactions on Information Theory*, 24(1):106–110, 1978.

[129] J.-J. Quisquater and C. Couvreur. Fast decipherment algorithm for the RSA public-key cryptosystem. *IEE Electronics Letters*, 18 (21):905–907, 1982.

[130] M. Quisquater, B. Preneel, and J. Vandewalle. On the security of the threshold scheme based on the Chinese remainder theorem. In D. Naccache and P. Paillier, editors, *Public Key Cryptography, 5th International Workshop on Practice and Theory in Public Key Cryptosystems, PKC 2002*, volume 2274 of *Lecture Notes in Computer Science*, pages 199–210. Springer-Verlag, 2002.

[131] M. O. Rabin. Efficient dispersal of information for security, load balancing, and fault tolerance. *Journal of ACM*, 36(2):335–348, 1989.

[132] 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.

[133] B. Schoenmakers. A simple publicly verifiable secret sharing scheme and its application to electronic voting. In M. J. Wiener, editor, *Advances in Cryptology - CRYPTO '99*, volume 1666 of *Lecture Notes in Computer Science*, pages 148–164. Springer-Verlag, 1999.

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

[135] D. Shanks. Class number, a theory of factorization, and genera. In *Proceedings of Symposia in Pure Mathematics (1969 Number Theory Institute)*, volume 20, pages 415–440. American Mathematical Society, 1971.

[136] V. Shoup. Practical threshold signatures. In B. Preneel, editor, *Advances in Cryptology - EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Computer Science*, pages 207–220. Springer-Verlag, 2000.

[137] G. J. Simmons. How to (really) share a secret. In S. Goldwasser, editor, *Advances in Cryptology - CRYPTO '88*, volume 403 of *Lecture Notes in Computer Science*, pages 390–448. Springer-Verlag, 1990.

[138] J. Simmons, W.A. Jackson, and K. Martin. The geometry of shared secret schemes. *Bulletin of the Institute of Combinatorics and its Applications*, 1:71–88, 1991.

[139] S. Singh. *The Code Book: The Evolution of Secrecy from Mary, Queen of Scots, to Quantum Cryptography.* Doubleday Books, 1999.

[140] M. Stadler. Publicly verifiable secret sharing. In U. M. Maurer, editor, *Advances in Cryptology - EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 190–199. Springer-Verlag, 1996.

[141] D. R. Stinson. An explication of secret sharing schemes. *Designs, Codes and Cryptography*, 2:357–390, 1992.

[142] D.R. Stinson. Decomposition constructions for secret-sharing schemes. *IEEE Transactions on Information Theory*, 40(1):118–125, 1994.

[143] C. Tang, D. Pei, Z. Liu, and Y. He. Non-interactive and information-theoretic secure publicly verifiable secret sharing. Cryptology ePrint Archive, Report 2004/201, 2004. (available at http://eprint.iacr.org/).

[144] T. Tassa. Hierarchical threshold secret sharing. In M. Naor, editor, *Theory of Cryptography, First Theory of Cryptography Conference,*

TCC 2004, volume 2951 of *Lecture Notes in Computer Science*, pages 473–490. Springer-Verlag, 2004.

[145] T. Tassa and N. Dyn. Multipartite secret sharing by bivariate interpolation. In M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, editors, *Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II*, volume 4052 of *Lecture Notes in Computer Science*, pages 288–299. Springer-Verlag, 2006.

[146] F. L. Ţiplea. Algebraic Foundations of Computer Science, Spring 2006. Course Notes, "Al.I.Cuza" University of Iaşi, Faculty of Computer Science.

[147] F. L. Ţiplea. Coding and Cryptography, Spring 2006. Course Notes, "Al.I.Cuza" University of Iaşi, Faculty of Computer Science.

[148] F. L. Ţiplea, S. Iftene, C. Hriţcu, I. Goriac, R.M. Gordân, and E. Erbiceanu. MpNT: A multi-precision number theory package. Number-theoretic algorithms (I). Technical Report TR 03-02, "Al.I.Cuza" University of Iaşi, Faculty of Computer Science, 2003. (available at http://www.infoiasi.ro/˜tr/tr.pl.cgi).

[149] M. Tompa and H. Woll. How to share a secret with cheaters. *Journal of Cryptology*, 1(2):133–138, 1988. (a preliminary version of this paper appeared in "Advances in Cryptology – CRYPTO '86", A. M. Odlyzko, ed., Lecture Notes in Computer Science 263 (1987), 261-265).

[150] V. Vinod, A. Narayanan, K. Srinathan, C. P. Rangan, and K. Kim. On the power of computational secret sharing. In T. Johansson and S. Maitra, editors, *Progress in Cryptology - INDOCRYPT '03*, volume 2904 of *Lecture Notes in Computer Science*, pages 162–176. Springer-Verlag, 2003.

[151] VoteHere. Network voting system standards, April 2002.

[152] M. Wiener. Cryptanalysis of short RSA secret exponents. *IEEE Transactions on Information Theory*, IT-36(3):553–558, 1990.

[153] H. Yamamoto. On secret sharing systems using (k, L, n)-threshold scheme. *Transactions of the IECE*, J68-A(9):945–952, 1985. (in Japonese)- English Translation: Electronics and Communications in Japan Part I 69 (9)(1986), 46–64.