3/01

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique

المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

École Nationale Polytechnique

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

Département d'Electronique

Projet de Fin d'Etudes

en vue de l'obtention du diplôme
d'Ingénieur en Electronique

Thème

Cryptographie Visuelle

Encadré par :
M^me^ L. HAMAMI
M^r^ KHELALFA

Étudié par :
M^lle^ A. ALLOUM
M^r^ S. BEGHOUL

Promotion : juin 2001

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

ملخّص

تتمحور الفكرة الاساسية لموضوع مشروعنا هذا حول امكانية التّرميز(وضع الرّموز) بكلّ أمن للمختلف الدّعائم المكتوبة (نصوص مطبوعة، مخطوطات، صور...الخ).

يعتبر هذا الاسلوب الجديد من التّرميز، المعروف ايضاب التّرميز البصري موضوعا حديثا في مجال المعلوماتية لأنّه يسمح بحلّ رموزالصور المرموزة دون اللّجوء الى استخدام الحسوب و أيضا دون اكتساب مبادئ دقيقة في علم التّرميز(الكربتوغرافية) الامر الّذي يبرّر استعماله بكلّ بساطة وبأمن كامل.

لقد تو سّع استخدام هذا التّرميز البصري الى متغيّرات بصرية لاشكالية توزيع السّر على أدنى عدد من المستعملين ك من بين عدد كلّي ن.

يمكن ان نلا حظ ، بفضل هذا التّرميز و بعد توزيع شفّاف على كلّ مستعمل ، بأنّ أيّة مجموعة من الشفّافات المتراكبة تعيد تشكيل الصّورة الاصلية و لو أنّ (ك ـ ١) منها لا تحمل أيّة معلومة.

الغرض من هذا المشروع تقديم مساهمة مزدوجة .

- تتمثّل المساهمة الا و لى في تقديم المعلومة المرموزة بصريا و المحتواة في المقالات النّظرية الموجّهة لجمهور واسع كما تتمثّل أيضا في استخراج و وضع شكل الخوارزميات التّي تتضمّنها هذه المقالات.

- أمّا المساهمة الثّانية فتكمن في تصوير برنامج (لو جيسيال) كفيل بتحقيق عدّة مخطّطات بصرية.

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

Abstract

The ability to encrypt written material such as printed texts, hand notes and pictures in a perfectly secure way, and to decode the material with the human visual system is the idea behind this engineering's project.

This form of data encryption is called « Visual Cryptography » and is relatively new in the area of Computer Science.type of Cryptographic scheme, allow to decode images without any cryptographic computation. The scheme is perfectly secure and very easy to implement.

It's extended into visual variants of the k out of n secret sharing problem, in which a dealer provides a transparencies to each one of the n users, any k of them can see the image by stacking their transparencies, but any (k-1) of them gain no information about it.

This project will provide two contributions:

The first is present action of the visual cryptography information contained in the theorical papers for a more general public and to distill algorithms implicit in these papers.

The second contribution of this project is to design and implement visual cryptography software.

Keywords: Visual Cryptography, Secret Sharing Schemes, Threshold Cryptosystems, human Visual system, Authentification, Identification, Data security

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

Résumé

L'idée maîtresse de ce projet de fin d'études s'ordonne autour de la possibilité de crypter, en toute sécurité, divers supports écrits (textes, imprimés, notes manuscrites, images, etc ...), et de les décrypter à l'aide du système visuel humain.

Ce nouveau type de cryptage, appelé cryptage visuel, représente un aspect récent du domaine de l'informatique. Il permet de décoder des images cryptées sans aucun calculateur ni pré-requis en cryptographie ; d'où sa simplicité d'utilisation et sa parfaite sécurité.

Le cryptage visuel a été étendu aux variantes visuelles du problème de partage de secret chez un nombre minimal d'utilisateurs noté k parmi leur nombre total noté n.

On peut observer, grâce à ce cryptage, et après l'attribution d'un transparent à chacun des n utilisateurs, que n'importe quel ensemble de transparents superposés reconstruisent l'image initiale mais que (k-1) d'entre eux n'obtiennent aucune information.

Ce projet de fin d'étude vise une double contribution :

La première étant la présentation de l'information du cryptage visuel traitée dans des articles théoriques destinés à un public plus général et également l'extraction et la formalisation des algorithmes implicites de ces mêmes articles.

La deuxième contribution réside dans la conception d'une implémentation qui réalise plusieurs schémas visuels.

Mots clés :Cryptage visuel, Schémas de partage de secret, Cryptosystèmes à seuil, , Système visuel humain, Identification, Authentification, Sécurité de l'information

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

REMERCIEMENTS

La soutenance d'un mémoire marque une étape importante dans la vie d'un étudiant. Nous tenons à saisir cette opportunité pour exprimer nos vifs remerciements à toutes celles et à tous ceux qui nous ont soutenus tout au long de la préparation de ce travail, encouragé à le poursuivre et à l'achever.

Nos remerciements les plus distingués vont, en premier lieu, à notre corps professoral et plus particulièrement à notre promotrice M^{me} HAMAMI et notre co-promoteur M^{r} KHELALFA pour leur concours précieux, leur disponibilité et leur engagement qui sont pour beaucoup dans la réalisation de ce mémoire.

Nos remerciements vont, également, à M^{r} BOURIDANE pour sa serviabilité et sa documentation.

Nous avons également une grande dette à l'égard d'Internet pour son aide généreuse.

Enfin, nous ne terminerons pas nos remerciements sans exprimer notre reconnaissance à notre environnement familial dont le soutien moral et matériel ainsi que son encouragement permanent ont beaucoup contribué à la réalisation de ce mémoire.

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

Dédicaces

Je dédie ce mémoire à toutes les personnes qui sont chères à ma vie.
A savoir

Mon père et ma mère qui m'ont permis d'atteindre ce niveau d'études et qui ont contribué à ma réussite.

Mes sœurs qui m'ont aidé de loin comme de prés.

Mes amis : parmi lesquels : Amira , Meriem, Mohsen, Salim, Farid, ZS Sofiane, Mehdi, Mouméne, Islem.....

L'élue de mon cœur qui j'espère le restera pour toujours.

SALAH BEGHOUL

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

Dédicace :

La plus grande gratitude est celle que l'on adresse à celui qui écoute et connaît l'ombre de nos intentions, l'intention dispense la formulation, la formulation n'en sera jamais assez gratifiante.
Je dédie mes efforts à mes parents qui en sont la source, les racines, l'énergie, la force, l'aboutissement, et l'avenir.
Je remercie mon frère pour l'exemple et pour ses trésors de conseils.
Je remercie ma sœur pour ses trésors d'énergie et de lumière.
Je remercie tous ceux qui ont jusqu'aujourd'hui apporté un plus quelque il soit, en connaissance, en sagesse, en savoir, en présence, en bien être et même en leçons, en questions et en un mot une pierre à ma construction ou bien même un caillou.
Je remercie les encouragements et la gratitude que j'ai rencontrés en certains.
Je remercie tous ceux la, et je leur retourne toute ma gratitude.
Je dédie cette goutte de savoir à tous ceux pour qui la connaissance n'est pas gloire mais passion.
Je la dédie à ceux pour qui rien n'est facile ni difficile, à tous ceux qui crée l'intérêt lorsqu'il n'existe pas, à tous ceux qui arrive à projeter l'abstrait dans la réalité et rapprocher le langage des livres, des savants et philosophes aux hommes pour qu'enfin on dise : « l'homme évolue, apprend et se civilise ».

Amira Alloum

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

SOMMAIRE

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

SOMMAIRE

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

SOMMAIRE

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

SOMMAIRE

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechniques

Lorsqu'on évoque le thème de la cryptographie, on pense souvent à des algorithmes complexes manipulant des structures de données à partir d'éléments de base de l'ordinateur tel que le mot mémoire. Ces algorithmes sont en mesure de faire prendre des journées de calcul aux meilleurs ordinateurs pour être cassés.

Cet aspect confère à la cryptographie un caractère confidentiel qui réserve l'exclusivité de la manipulation aux professionnels de l'informatique.

Une nouvelle orientation de la cryptographie est amorcée à partir du concept de l'exploitation du système visuel humain qui manipule et traite des structures de données plus évoluées : les images.

Cette nouvelle forme de cryptographie offre la possibilité d'encrypter des mots manuscrits ou même des images ou photos en toute sécurité. Quant au décodage, il repose totalement sur le système visuel humain.

Ces supports écrits sont utilisés et manipulés sous forme d'images numérisées, consistant en une matrice de pixel (picture element) pour pouvoir les exploiter et effectuer les traitements souhaités.

En effet, l'image est considérée comme le support d'information le plus performant tant au niveau de son accessibilité et de son universalité qu'au plan de ses diverses applications. Elle est apparue, d'abord, d'un point de vue scientifique, comme un outil de connaissance essentiel et, plus récemment, comme une technologie innovante et performante dans les secteurs industriels de la production automatisée. Cet intérêt, amplifié par les possibilités croissantes de l'usage des images, est lié fondamentalement au fait que chez l'homme la vision est le sens perceptif dominant (près de 60% du cortex cérébral humain est utilisé par le système visuel humain).

En parallèle, le système visuel humain est mis en exploitation dans la plupart des domaines traitant des images qu'elles soient fixes ou animées (télévision par exemple), en tant que paramètre de référence par rapport à l'image à partir de laquelle on fixe certains paramètres.

Dans notre nouveau schéma de cryptage, le système visuel représente une référence qui permet de décider des constructions du codage des images de même qu'il constitue un paramètre indispensable pour fixer les caractéristiques de ces images.

En plus de cela, le système visuel humain participe au protocole de cryptage en tant qu'outil de décryptage qui permet de reconstruire l'image, de moyenner et de percevoir le contraste.

Ce schéma de cryptage de données, intitulé cryptage visuel, est encore récent puisqu'introduit en 1994 par M.Naor et S.Shamir lors de l'Eurocrypt 94. Il a été étendu aux variantes visuelles du problèmpe du partage visuel du secret k parmi n

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

(cryptosystèmes à seuil) et a déjà fait ses preuves dans le domaine de la cryptographie classique.

Le principe de base consiste à diviser une image en n images séparées imprimées chacune sur un transparent, telle qu'aucune d'elles ne porte d'informations sur le secret (image originale). Elles sont assimilées à du bruit anodin.

La superposition d'au moins k transparents reconstitue l'image. Cependant, empiler (k-1) quelconque à partir de ces mêmes transparents n'apporte aucune information.

Ainsi un tel schéma de cryptage peut être effectué par le plus simple des calculateurs. Par contre, le décryptage ne requiert aucun calculateur ni aucune connaissance ou prérequis en cryptologie étant donné que le système visuel l'accomplit selon les formes désirées.

En conséquence, aucun calculateur aussi puissant et performant soit-il ne pourra cassé un tel cryptosystème. Mais il suffit de réunir k utilisateurs honnêtes parmi n répondant à certaines conditions vérifiables pour leur permettre de participer au décryptage de l'information.

Exposer le paradigme du cryptage visuel, les méthodes de construction de ses multiples schémas et les protocoles dans lesquels elles interviennent constitue la première contribution de ce projet de fin d'études dont le but est de faire connaître cette théorie à une plus grande communauté et à formaliser les algorithmes implicites des articles théoriques.

La seconde contribution du projet concerne la conception de l'implémentation d'un software de cryptage visuel susceptible de mettre en pratique l'importante partie théorique.

Au total, le projet s'ordonnera autour de trois parties principales :

La première partie est consacrée aux notions fondamentales de la cryptographie classique qui représente une entrée en matière nécessaire compte tenu du fait qu'il n'existe pas de travail antécédent relatif à cette discipline au sein de notre école. Cette partie sera, quant à elle, subdivisée en deux chapitres après une approche historique de la cryptographie depuis sa genèse jusqu'à nos jours.

Le premier chapitre traite des principes de base de sécurité de l'information et des notions fondamentales destinées à faire comprendre les bases de notre sujet. Quant au second, il attrait aux cryptosystèmes à seuils qui sont un cas particulier du partage de secret.

La deuxième partie est inhérente à la théorie du cryptage visuel et comprend cinq chapitres.

Le premier constitue une présentation générale de ce cryptage visuel. Quant au second, il énonce les protocoles et méthodes d'authentification et d'identification visuelles. Le chapitre 3 relate le modèle et les propriétés d'un schéma de cryptage visuel. Le chapitre 4 formule les techniques de construction des schémas de cryptage visuel. Enfin, le chapitre 5 expose les ouvertures et le développement du cryptage visuel.

La troisième partie de ce mémoire se rapporte à la présentation de notre logiciel et s'articule autour de trois chapitres.

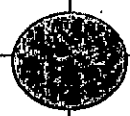
Le premier concerne la présentation de notre langage d'implémentation et la justification de ce choix. Le second chapitre traite de l'implémentation du logiciel et de la formalisation des algorithmes. Enfin, dans le dernier chapitre il est question de tester notre logiciel et d'en commenter les résultats.

PARTIE I

Notions fondamentales de cryptologie

- **Historique et état de l'art**
- **Chapitre 1 : Principes de base de sécurité de l'information**
- **Chapitre 2 : Cryptosystèmes à seuil**

Historique et état de l'art



- L'ère artisanale
- L'ère technique
- L'ère actuelle

Notre sujet est relatif à la cryptologie du grec *kryptos (*caché*) et logos (*science). Ce terme peut être assimilé à la science du secret ou à l'art de coder un message de façon à le rendre incompréhensible sauf pour son destinataire. Etant donné que ce mémoire traite de la cryptologie sous le prisme d'une réflexion technique, nous nous proposons tout d'abord de l'introduire par une étude historique et contemporaine de la cryptographie (du grec *graphein,* écrire), de ces écritures secrètes qui sont à la base de cette cryptologie dont l'actualité souligne l'importance.

La cryptographie peut être appréhendée de plusieurs façons. En effet nous sommes passés d'une cryptographie tenant aux divinités à une cryptographie liée aux intérêts personnels et, plus tard, étatiques. Par la suite, il y a eu le passage d'une cryptographie mécanique à une cryptographie mathématique.

D'une manière générale, l'histoire de la cryptographie reste marquée par trois grandes périodes. La première que certains spécialistes qualifient d'artisanale va jusqu'à la fin du XIX^{ème} siècle. La deuxième, caractérisée par des avancées techniques notables, se situe entre 1883 et 1975. La troisième recouvre l'époque contemporaine.

- **L'ère artisanale**

La cryptographie est apparue sous la forme de hiéroglyphes égyptiens, il y a environ 3900 ans, sur la tombe de Khnumhopet II. Les inscriptions étaient formées de hiéroglyphes rarement utilisés au lieu des hiéroglyphes habituels. Rédigées, non pas pour rendre l'écriture secrète, mais plutôt pour donner un caractère solennel à l'épitaphe, ces inscriptions incorporent l'un des caractères essentiels de la cryptographie : une transformation délibérée de l'écriture.

C'est dans une autre grande civilisation de l'antiquité, la Mésopotamie, que l'on trouve d'autres traces de cryptographie d'un niveau déjà avancé. Des tablettes cunéiformes datant de 1500 avant Jésus-Christ (J.-C.) ont été retrouvées sur les bords du Tigre. Elles comportent des formules concernant l'émaillage de la poterie que le scribe voulait jalousement garder secrètes. Cette technique était également en usage en Syrie et en Babylonie.

Dans la Bible, la méthode de chiffrement utilisée consiste à remplacer la dernière lettre de l'alphabet hébreu par la première et réciproquement. Cette méthode porte le nom d'*Atbash.*

En Inde, l'utilisation des écritures secrètes semble assez avancée dés la fin du IV^{ème} siècle avant J.-C.

En Grèce antique, l'écrivain Polybius a inventé un procédé cryptographique où les lettres de l'alphabet étaient disposées dans une table de 5 colonnes et 5 rangées.

	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	IJ	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Figure 1 : Cryptogramme de Polybius

Polybius proposait de transmettre ces nombres au moyen de torches. Les cryptologues modernes ont vu dans ce procédé des caractéristiques très intéressantes, comme : la conversion de lettres en chiffres et la réduction du nombre de symboles utilisés. Cette méthode a donné naissance à la méthode par substitution.

Dans la Chine antique, on avait recours à la stéganographie qui vise à dissimuler les messages secrets que le porteur dissimulait sur lui ou avalait.

Ce procédé se retrouvait également en Grèce où l'on pouvait cacher l'existence d'un message en tondant un héraut sur le crâne duquel on tatouait l'information. Une fois la repousse des cheveux faite, une seconde tonte était nécessaire pour que le destinataire du message soit informé.

Pour correspondre avec ses amis, Jules César utilisait une méthode de substitution très simple : chaque lettre de l'alphabet était avancée de trois rangs.

A B C D E F G H I J K ...
d e f g h i j k l m n ...

Figure 2 : Cryptogramme de Jules César

Figure 3 : Jules César (56 avant J.-C.)

S'agissant de la civilisation arabe, l'intérêt pour la cryptographie s'est manifesté au moyen âge. En 855, le savant Abu Bakr Ahmad ben Ali ben Wahshiyya an-Natabi décrit plusieurs cryptosystèmes dont le chiffrement par substitution (Davidien). La civilisation arabe a été également la première à s'intéresser aux méthodes de cryptanalyse et à les faire progresser de façon significative. En 1412, Qalqashandi, un écrivain arabe, a écrit dans une encyclopédie en 14 volumes, une section sur la cryptologie où il décrit des méthodes par substitution et par transposition. Des notions importantes pour la cryptanalyse apparaissent également pour la première fois tels que la fréquence des symboles dans le Coran et le développement de la lexicographie.

Pendant ce temps, en Occident, les gouvernements de l'Europe de l'Ouest avaient recours à l'usage de la cryptographie pour communiquer avec leurs ambassadeurs. Les premières grandes avancées ont lieu lors de la Renaissance italienne (1350-1600). En 1379, Lavinde, secrétaire de Clément VII, originaire de Parme, met au point des systèmes cryptographiques dont la *substitution monoalphabétique.* Il fournit un ensemble de clés individuelles pour chacun des 24 correspondants de ClémentVII. La collection de clés de Lavinde est la plus ancienne collection existante. Certains mots fréquemment utilisés sont remplacés par un code constitué de deux lettres. Par la suite, il y a eu la mise en place de systèmes de chiffrement modernes. Leon Battista Alberti (peintre, musicien, sculpteur, architecte) est considéré comme le père de la cryptologie occidentale. En 1467, il invente la *substitution polyalphabétique*, technique qui permet à plusieurs symboles de l'alphabet secret de représenter un symbole de l'alphabet en clair. Ce procédé rend plus difficile le décryptage utilisant la fréquence des lettres. Le cadran inventé par Alberti est composé de deux disques en cuivre contenant l'alphabet latin, le premier étant fixe, le second mobile. Pour commencer à chiffrer, une lettre prédéterminée du disque intérieur est alignée avec n'importe quelle lettre du disque extérieur, considérée comme le début de l'alphabet secret. Les disques restent fixes pour chiffrer les premiers mots en prenant la correspondance de chaque lettre dans l'alphabet secret ainsi désigné. Après avoir chiffré plusieurs mots, on exécute une rotation des disques de telle sorte que la lettre prédéterminée au départ soit alignée avec une nouvelle lettre du disque extérieur. Aujourd'hui, cette technique a perdu de sa valeur mais elle a eu le mérite d'inventer la technique de rotation de disques et de changer ainsi l'alphabet secret plusieurs fois pour un message.

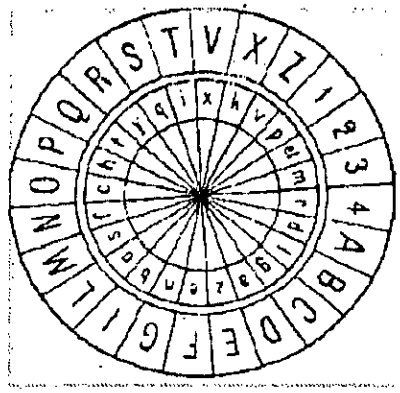


Figure 4 : Cadran chiffrant d'Alberti

L'avancée significative suivante fut exécutée par Johannes Trithemius, un moine bénédictin allemand qui avait un goût particulier pour les sciences occultes. Il a écrit une série de six livres nommés *Polygraphiae*. Dans le premier volume, il décrit une table qu'il a imaginée et intitulée *tabula recta*. Dans cette table, l'alphabet est répété sur 26 lignes, avec un décalage à gauche de une lettre pour chaque nouvelle rangée. Le codage d'un message est ensuite assez simple : pour coder la première lettre du message clair on prend la correspondance entre cette lettre dans la première rangée et la lettre dans la seconde rangée. Pour la deuxième lettre, on prend la correspondance sur la troisième ligne, et ainsi de suite...

Ce tableau est amélioré en 1553 par Giovan Batista Belaso, l'inventeur de la notion de clé littérale appelée mot de passe. Le principe est simple, on choisit un mot de passe associé au texte clair et répété autant de fois que nécessaire. Pour coder une lettre du texte clair, on utilise la rangée de la table de Trithemius, dont la première lettre est celle figurant sur la ligne contenant le mot de passe.

ABCDEFGHIJKLMNOPQRSTUVWXYZ
BCDEFGHIJKLMNOPQRSTUVWXYZA
CDEFGHIJKLMNOPQRSTUVWXYZAB
DEFGHIJKLMNOPQRSTUVWXYZABC
EFGHIJKLMNOPQRSTUVWXYZABCD
FGHIJKLMNOPQRSTUVWXYZABCDE
.....

Figure 5 : Cryptogramme de Giovan Batista Belaso

En 1563, Giovanni Battista Porta publie un livre «De Furtivis Literarum Notis» dans lequel il décrit et analyse les cryptosystèmes existants. Le cryptographe français le

plus connu du XVI^{ème} siècle est Blaise de Vigenère qui améliore en 1585 la table de Trithemius par la modification de l'utilisation de la clé. En 1628, Antoine Rossignol remarqué, deux ans auparavant par le prince de Condé, organise le Service du Chiffre.

Pendant ce temps, de nombreux cryptographes, employés par le gouvernement français, fondent le Cabinet Noir. Au XVIII^{ème} siècle, les Cabinets Noirs étaient communs dans toute l'Europe. L'un des plus célèbres se trouvait à Vienne, c'était le *Geheime Kabinets-Kanzlei*. Il était dirigé par le Baron Ignaz de Koch de 1749 à 1763. Les cryptanalystes interceptaient et décryptaient des courriers diplomatiques (dont des lettres de Napoléon, Talleyrand...).

L'Angleterre possédait également son Cabinet Noir. C'est probablement Wallis, connu dans son pays comme le plus grand mathématicien avant Newton, qui en était à l'origine. En 1854, Charles Wheatstone et Lyon Playfair inventent le système *Playfair*, le premier système utilisant des paires de symboles pour le chiffrement. La clé est disposée dans un carré 5 x 5, complétée par les lettres de l'alphabet qui n'apparaissent pas encore, dans l'ordre alphabétique. Exemple de tableau pour une clé égale à CHARLES :

C H A R L
E S B D F
G I/J K M N
O P Q T U
V W X Y Z

En 1863, Friedrich W. Kasiski, un officier prussien, propose la première méthode de cryptanalyse qui permet de casser presque tous les systèmes de chiffrement existants. La méthode consistait à rechercher la répétition de chaîne de caractères dans le texte chiffré. La distance entre ces répétitions est utilisée pour trouver la longueur de la clé. C'est un phénomène que Porta à observer à son époque mais qu'il n'avait pas réellement identifié. La rencontre de la répétition d'une portion de la clé avec une répétition dans le texte clair produit une répétition dans le texte chiffré. Une fois la longueur de la clé, l, connue, on peut faire des statistiques sur chaque n^{ième} caractère. La fréquence d'apparition donne le caractère qu'il représente dans l'alphabet de chiffrement. Les répétitions sont parfois le fruit du hasard et il faut parfois plusieurs essais avant de trouver la longueur réelle de la clé. Néanmoins, cette méthode était beaucoup plus efficace que toutes les méthodes existantes.

Lors de la deuxième moitié du XIX^{ème} siècle, alors que les Cabinets Noirs tendent à disparaître les uns après les autres, une invention révolutionne la cryptographie : c'est le télégraphe. Cette innovation entraîne l'introduction de dizaines de nouveaux systèmes de chiffrement. David Kahn dit que le télégraphe a créé la cryptographie moderne qui marque le début d'une ère nouvelle.

- **L'ère technique**

Les principes de Kerckhoffs

En 1883, Kerckhoffs publie dans le Journal des sciences militaires un article dans lequel il énonce des principes qui décrivent les caractéristiques fondamentales d'un système de chiffrement :

1. Le système doit être, matériellement sinon mathématiquement, indéchiffrable,
2. Il faut qu'il n'exige pas le secret et qu'il puisse sans inconvénient tomber entre les mains de l'ennemi,
3. La clé doit pouvoir être communiquée, retenue sans le secours de notes écrites et être changée ou modifiée au gré des correspondants,
4. Il faut qu'il soit applicable à la correspondance télégraphique,
5. Il faut qu'il soit portatif et que son maniement ou son fonctionnement n'exige pas le concours de plusieurs personnes,
6. Il est nécessaire, vu les circonstances qui en commandent l'application, que le système soit d'un usage facile ne demandant ni tension d'esprit, ni la connaissance d'une longue série de règles à observer.

Kerckhoffs dénonce ainsi des idées bien ancrées au sein de l'armée qui prétendent que l'indéchiffrabilité n'est pas une condition absolue à l'utilisation d'un système cryptographique.

Selon cette affirmation, les instructions chiffrées transmises à un instant donné doivent rester secrètes durant uniquement les quelques heures qui suivent leur transmission. Kerckhoffs prétend, au contraire, que le secret doit rester crucial au-delà de cette période. Il cite l'exemple d'un message chiffré, intercepté et déchiffré qui peut servir à déchiffrer d'autres messages chiffrés avec la même clé. Concernant la nécessité du secret, Kerckhoffs évoque le danger du système matériel de chiffrement. Il démontre

la facilité de déchiffrer les cryptogrammes obtenus avec les méthodes de substitution en se fondant sur la fréquence de chacune des lettres de l'alphabet dans un texte.

Au début du XX^{ème} siècle, les efforts substantiels déployés, en Angleterre, dans le domaine de la cryptanalyse ont permis aux services anglais de casser la plupart des cryptosystèmes de leurs ennemis. Le groupe de cryptanalystes fut dénommé *Room 40*. D'octobre 1914 à février 1919, 15000 messages de l'ennemi furent décryptés.

La première guerre mondiale fut une suite de véritables batailles sur le plan technique. C'est elle qui amena la cryptologie à maturité. Dès 1914, le système allemand *Ubchi* est cassé par Cartier,Olivari et Schwab grâce à l'utilisation d'une double transposition avec la même clé. Ubchi est abandonné le 18 novembre 1914 au profit du système de chiffrement de Vigenère avec la clé (*ABC*) suivi d'une transposition simple. Ce système est cassé à son tour par le lieutenant Georges Jean Painvin. L'*ABCD*, variante de l'*ABC* résiste aux attaques jusqu'au mois d'avril 1916 où apparaissent des systèmes fondés sur des substitutions plus difficiles à casser.

En janvier 1917, on voit réapparaître les grilles tournantes et les méthodes par transposition. En mars 1918, c'est l'émergence du système (*ADFGVX*), ainsi nommé en raison de l'apparition de ces six lettres seulement dans les cryptogrammes. Painvin pense, d'abord, à un tableau de substitution 5 x 5 (car la lettre V ne figure dans aucun des premiers cryptogrammes interceptés) suivi d'une transposition. Ne disposant pas d'assez de textes chiffrés, il réussit, néanmoins, à reconstituer le tableau et la transposition et par la suite à découvrir la clé des messages émis. Un nouveau cryptogramme où figure la lettre V est cassé par Painvin.

Inventé en 1917 par Vernam, le système à masque jetable (dit one-time pad) est le seul algorithme de chiffrement inconditionnellement sûr. C'est à dire que la connaissance du message chiffré n'apporte aucune information sur le message clair. Le principe est très simple ; la clé qui est générée de façon aléatoire, doit être aussi longue que le texte à chiffrer et ne peut être utilisé qu'une seule fois. Une addition digit-à- digit est effectuée entre le message clair et la clé pour obtenir le message chiffré. Les inconvénients de ce système concernent la taille de la clé, la non réutilisation du masque et la nécessité de posséder un générateur aléatoire pour créer ce masque.

Le recours à des machines chiffrantes change fondamentalement la nature de la cryptographie et de la cryptanalyse. Les systèmes de chiffrement reposent sur les

mêmes bases mathématiques mais les méthodes de chiffrement deviennent électroniques et fiables.

Une grande avancée technologique est enregistrée avec l'invention du rotor, appareil composé d'un disque épais à deux faces comportant 26 contacts en laiton séparés par un matériau isolant. Il dispose d'un clavier style machine à écrire pour entrer le texte clair en envoyant les impulsions électriques correspondantes à la face d'entrée. Le texte chiffré est généré à partir du rotor pour être imprimé ou transmis. Le rotor est plus sûr que les systèmes précédents car après chaque lettre chiffrée, le rotor pivote de telle sorte que l'alphabet soit décalé d'une lettre. Ainsi, on obtient un système de chiffrement à base de substitution polyalphabétique avec une grande période. Un deuxième rotor peut être ajouté et la période obtenue est alors de 676. La première machine chiffrante à rotor a été réalisée par l'américain Edward Hugh Hebern en 1921.

Durant la prohibition aux Etats-Unis, l'alcool est transporté par des contrebandiers qui utilisent un système complexe de chiffrement. C'est une femme, Elizabeth Smith Friedman, cryptanalyste, qui a aidé la police à arrêter de nombreux contrebandiers et a forcé les autres à changer souvent de clé.

La deuxième guerre mondiale a permis à la cryptographie de connaître un développement considérable grâce à l'utilisation de la machine chiffrante Enigma inventée par l'allemand Arthur Scherbius. Le premier modèle (nommé modèle A) a la taille d'une caisse enregistreuse ; il a été remplacé rapidement par le modèle B dont la taille est celle d'une machine à écrire. Les modèles C et D ont apporté des améliorations en ce qui concerne la taille de la machine.

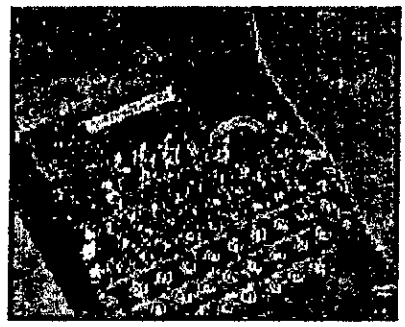


Figure 6 : Machine Enigma

Les Allemands, qui utilisaient Enigma durant la seconde guerre mondiale, ne se doutaient pas que leurs messages étaient régulièrement interceptés et décryptés par les alliés ; notamment par une équipe anglaise dirigée par Alan Turing. Le travail des cryptanalystes a été pour beaucoup dans l'issue de ce conflit mondial.

En 1948, Shannon apporte une autre contribution à la cryptographie moderne en livrant des techniques avancées de mathématiques. Il publie, la même année, un article intitulé « A Communication Theory of Secrecy Systems » où l'élément le plus important est le développement d'une mesure cryptographique appelée « unicity distance ». Il s'agit d'un nombre qui indique la quantité de texte chiffré qu'il est nécessaire de connaître pour pouvoir déterminer de façon certaine le texte clair d'origine. Ce nombre est calculé ainsi :

$$H(K) / (|M| - H(M))$$

où H(K) l'entropie, est la quantité d'informations contenue dans la clé et H(M) est la quantité d'information contenue dans chaque symbole de l'alphabet. Shannon note qu'avec un système travaillant avec une clé aléatoire de longueur supérieure ou égale à la longueur du message, le texte clair ne peut pas être déterminé à partir du texte chiffré. Ce type de chiffrement est le one-time pad de Vernam utilisé, d'ailleurs, durant la Guerre froide pour le téléphone rouge entre Moscou et Washington.

- **L'ère actuelle**

La montée en puissance des outils informatiques et l'essor des besoins civils, induits par le développement et la vulgarisation d'Internet caractérisent cette période. De nouvelles techniques ont vu le jour. Les plus remarquables ont été la cryptographie à clé publique et la notion de signature électronique, introduites en 1976 par Whitfield Diffie et Martin Hellman de l'université de Stanford.

Dans le domaine bancaire et gouvernemental, le système DES « Data Encryption System » est un standard publié en 1977. Il s'agit du premier algorithme dont le code source a été rendu public. Depuis la démocratisation de l'Internet et de la baisse des prix du matériel micro-informatique, les particuliers se sont intéressés à l'échange de documents électroniques. Leur contenu n'étant pas protégé il peut être lu voire modifié par des tiers indiscrets. Les systèmes cryptographiques, réservés jusqu'alors aux militaires, en raison des puissances de calcul nécessaires, sont devenus accessibles aux

particuliers. Au cours des années 1970, la NSA (National Security Agency) avait le monopole de la cryptographie militaire américaine.

De nos jours, la cryptographie n'est plus le domaine réservé aux militaires et son utilisation s'étend à d'autres domaines jugés de plus en plus stratégiques comme : la diplomatie, l'industrie, le commerce, ...

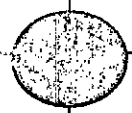
L'ére actuelle a été marquée par la performance des calculateurs de toutes sortes, les cartes à puces, les algorithmes complexes, les clés aléatoires, etc... Cependant les calculateurs n'ont presque plus de secrets pour l'homme, la cryptographie à la pointe de l'art se pratique par les privés. Le piratage et le hacking caractérisent aussi cette ère, casser des programmes, logiciels et mêmes des algorithmes de cryptage fait partie des pratiques courantes.

On a donc eu besoin d'explorer dans de nouvelles directions. On a conçut alors d'exploiter le système visuel humain pour élaborer de nouvelles cryptographies : la cryptographie visuelle.

Aujourd'hui, l'homme s'inspire de ses propres systèmes biologiques et de leur complexité, pour les tenter de les reproduire et tirer profit de leur intelligibilité extrême afin de répondre à ses exigences.

On a ainsi retracé l'évolution et le développement de la cryptologie qui s'est fait sous l'influence des besoins et des exigences de l'homme, selon les circonstances de son ère.

CHAPITRE 1 Principes de base de sécurité de l'information



1.1 Entrée en la matière

1.2 Aspects élémentaires de la sécurité de l'information

1.2 La gestion des clés « Key Management »

1.1 Entrée en la matière

La **cryptologie** est la science qui étudie les communications secrètes. Elle repose sur le principe fondamental de pouvoir communiquer en toute sécurité sur un canal peu sûr, c'est-à-dire ouvert à toutes écoutes et/ou manipulations frauduleuses.

Il ne faut pas confondre cette science avec la stéganographie, dont l'objectif est également l'échange de messages secrets par leur dissimulation dans l'environnement ou dans une masse d'informations.

La cryptologie comporte deux domaines d'études complémentaires : la **cryptographie** et la **cryptanalyse**.

La **cryptographie** regroupe l'ensemble des méthodes qui permettent de coder un message afin de le rendre incompréhensible pour quiconque ne disposant de moyens de le déchiffrer. On parle d'**encrypter** un message lorsqu'il s'agit de le crypter ou le chiffrer ; le code qui en résulte prend le terme de **cryptogramme**. L'action de lui restituer sa forme originelle s'appelle le **décryptage**.

Par contre, la **cryptanalyse** est l'art de révéler les messages ayant fait l'objet d'un encryptage. Lorsqu'on réussit à déchiffrer, au moins une fois, un cryptogramme, on dit que l'algorithme ayant servi à le crypter a été **cassé**. Une tentative de cryptanalyse s'appelle une **attaque**.

Le message à protéger s'appelle le **texte en clair**, dont le cryptage génère le **texte chiffré** à transmettre au destinataire. Il est préférable de ne pas utiliser le mot code pour désigner le texte chiffré parce qu'historiquement ce terme désigne, plutôt, une correspondance linguistique. C'est à dire une convention passée entre deux interlocuteurs, mais qui n'a rien à voir avec le résultat d'un cryptage.

On appelle **clé** l'information secrète utilisée pour encrypter un message et, plus tard, le décrypter. Elle permet à celui qui dispose de la clé à entrer en toute légalité dans le secret. Cependant, on peut concevoir des algorithmes qui n'utilisent pas de clés. Dans ce cas l'algorithme lui-même constitue le secret et son principe ne doit en aucun cas être dévoilé.

On désigne par **cryptosystème** l'ensemble composé *d'un algorithme*, des *textes en clair*, des *textes chiffrés* et *clés possibles*. Tous ces éléments permettent d'appréhender la notion de performance d'un cryptosystème basée sur des critères de sécurité, de coût et de facilité d'emploi qui sont souvent antagonistes mais aussi évolutifs. L'évolution de la société de l'information et le développement des réseaux de communication ont conduit à la démocratisation des cryptosystèmes, tout en faisant apparaître de nouvelles exigences.

La confidentialité des messages ne suffit plus, il faut également garantir leur intégrité et leur authenticité.

Selon le «Handbook of Applied Cryptography» [5] : la cryptographie est l'étude des techniques mathématiques relatives aux aspects de la sécurité de l'information telles la confidentialité, l'intégrité des données, l'authentification des entités, et l'authentification de l'origine des données.

Donc, la cryptographie ne se limite pas seulement à des moyens d'assurer la sécurité de l'information mais représente aussi et surtout un ensemble de techniques.

1.2 Aspects élémentaires de la sécurité de l'information

1.2.1 Confidentialité

Elle permet l'échange de messages secrets entre des individus. Elle consiste à chiffrer le message clair à envoyer, de manière inintelligible, puis à le déchiffrer à la réception par le destinataire légitime.

La sécurité du protocole ne repose presque jamais sur le fait que l'algorithme de chiffrement / déchiffrement doit être tenu secret. Cette idée est connue sous le nom du principe de *Kerckoffs*. Le principe préconise que la sécurité doit reposer sur les clés, c'est-à-dire des données supplémentaires nécessaires au chiffrement et déchiffrement des messages.

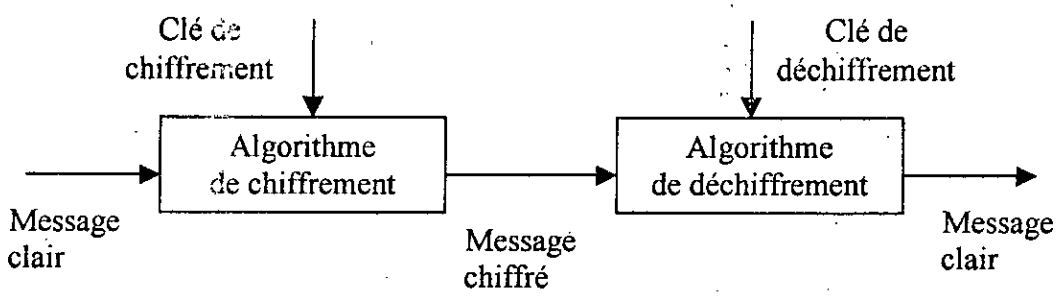


Figure 1.1 : Confidentialité

Les cryptosystèmes à algorithmes restreints - où le secret qui préserve les données est l'algorithme lui-même - ne sont plus guère utilisés aujourd'hui. Si l'algorithme est dévoilé, de manière intentionnelle ou pas, le système s'effondre complètement. En effet, la construction d'un algorithme sûr requiert de bonnes compétences en cryptologie car les ordinateurs actuels permettent diverses techniques puissantes et rapides pour casser un algorithme.

On distingue deux grandes classes de systèmes cryptographiques : les systèmes symétriques et les systèmes asymétriques.

a) La cryptographie symétrique

La caractéristique principale de la cryptographie symétrique, dite aussi à ***clé privée***, est l'utilisation d'une clé commune à l'émetteur du message et à son destinataire. Plus généralement, la clé (souvent utilisée aussi bien pour le chiffrement que pour le déchiffrement) est utilisée par tous les membres d'un groupe communiquant entre eux sans être divulguée en dehors de ce groupe.

Les systèmes symétriques comportent deux inconvénients majeurs qui sont liés au partage du secret :

- D'une part, la distribution des clés secrètes nécessite un canal sécurisé,
- D'autre part, le fait que la même clé soit connue de tous les membres du groupe facilite une fuite du secret, notamment dans le cas de groupes dynamiques.

Il est possible d'éviter le partage d'un secret avec beaucoup d'utilisateurs en instaurant une clé pour des sous-groupes d'utilisateurs. Le nombre de clés peut augmenter rapidement. Ainsi, si chaque sous-groupe est constitué de deux personnes, on a alors n(n -1)/2 clés privées en circulation pour un groupe initial de n membres.

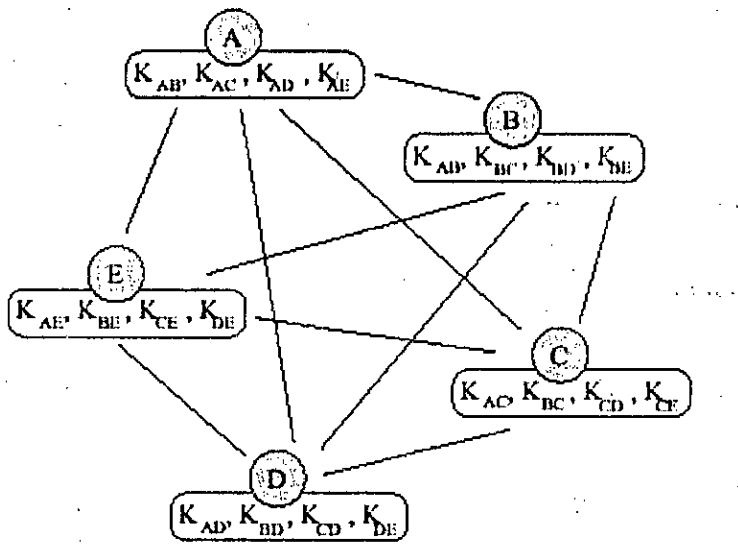


Figure 1.2 : Système symétrique

Malgré leurs inconvénients, les systèmes symétriques sont très employés car ils permettent, généralement, un chiffrement / déchiffrement rapide.

Le système cryptographique symétrique, D.E.S. (Data Encryption Standard), a été publié en 1975 par IBM. Il est, aujourd'hui, le système le plus utilisé dans le monde et

résiste encore solidement aux attaques. Il existe aussi RC4, RC5, IDEA et d'autres algorithmes tel que Blowfish.

Cependant, la taille des clés utilisées (56 bits) s'est révélée trop faible pour des applications nécessitant une forte sécurité, ce qui a conduit à son remplacement par l'A.E.S. (Advanced Encryption Standard).

La plupart des algorithmes, à clé secrète, fonctionnent selon les principes suivants :

- Le premier consiste à traiter l'information de manière linéaire, bit après bit. Il s'agit généralement de réalisations matérielles, particulièrement bien adaptées à ce type de traitement,
- Le second consiste à découper l'information en blocs d'une certaine taille, si possible adaptée à la taille des registres du processeur utilisé. Ce type de chiffres est parfaitement adapté aux réalisations logicielles.

Le principe général consiste à crypter le texte en le découpant en blocs de taille fixe, l'algorithme se chargeant de rendre l'information illisible. Cependant ce système comporte une faille dans la mesure où s'il crypte chaque bloc de manière indépendante, tous les blocs seraient chiffrés de la même manière ; la cryptanalyse (attaque à texte chiffré) n'en devient que plus aisée et facile. Pour y remédier, on réinjecte le résultat précédant dans le bloc suivant. On distingue quatre méthodes de base relatives au traitement par blocs :

- **ECB** (*Electronic Code Book*). Cas le plus simple : chaque bloc est crypté de manière indépendante.

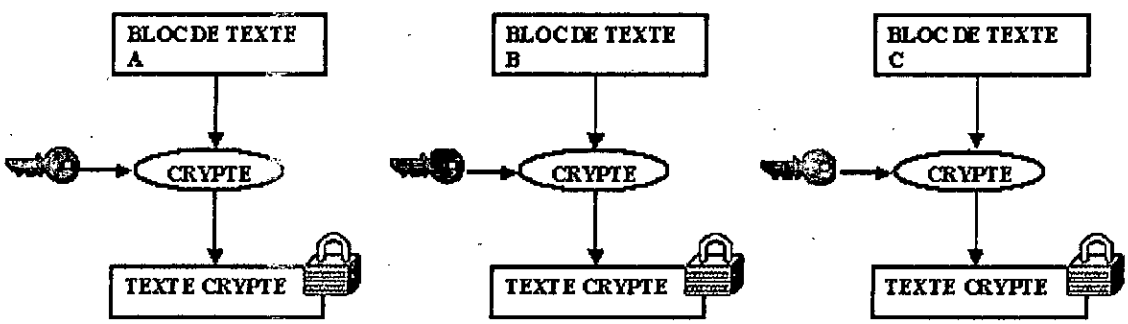


Figure 1.3 : Schéma ECB

- **CBC** (*Cipher Block Chaining*). Un vecteur d'initialisation est combiné avec le texte clair. Ensuite le résultat de chaque encryptage de bloc est combiné avec le texte clair du bloc suivant.

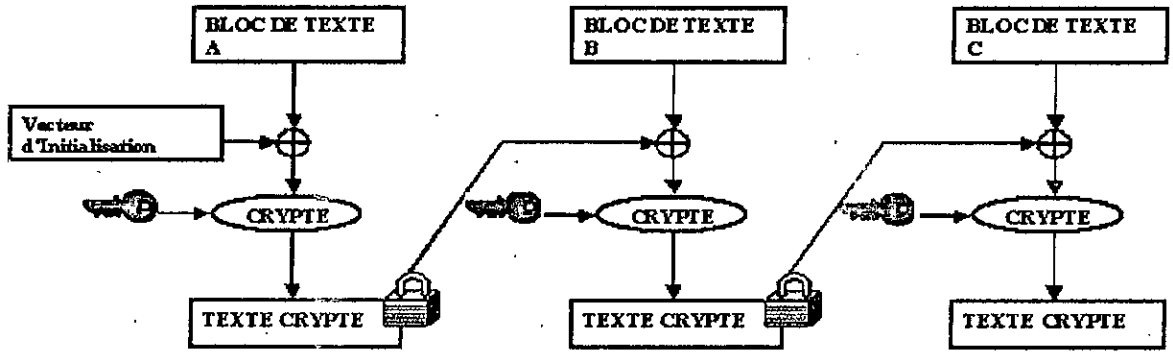


Figure 1.4 : Schéma CBC

- **CFB** (*Cipher Feedback*). Le bloc crypté est combiné avec la clé du bloc suivant devant être crypté.

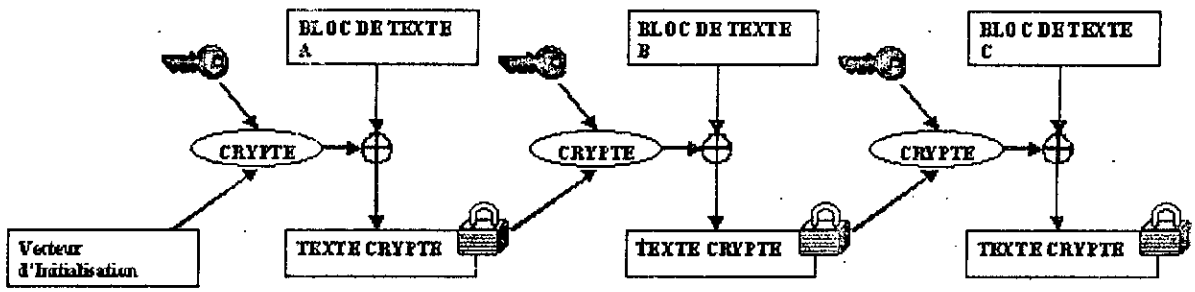


Figure 1.5 : Schéma CFB

- **OFB** (Output Feedback). La clé est modifiée à chaque itération et combinée avec la clé suivante

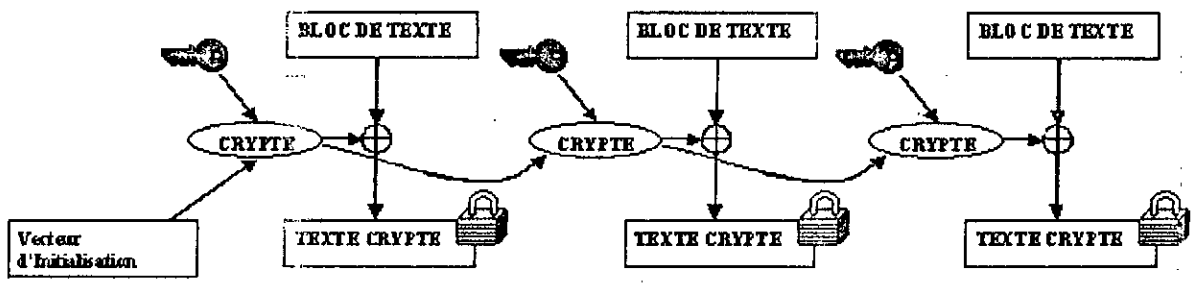


Figure 1.6 : Schéma OFB

Ces différents modes peuvent êtres combinés entre eux et accepter diverses techniques notamment pour le calcul du vecteur d'initiation et la méthode de combinaison des éléments entre eux.

En fonction de l'algorithme, plusieurs paramètres sont à prendre en compte parmi lesquels la longueur de la clé et celle du bloc en entrée (en clair) et en sortie (crypté). Ces opérations étant souvent répétitives il est question alors de ronde et on dit que tel algorithme est sûr à partir de 16 rondes.

Le but général est de disperser le texte en clair et de le permuter pour que le texte final s'éloigne le plus possible du texte initial. Il y a lieu, aussi, de faire attention à la clé parce que c'est elle qui assure la sécurité du cryptogramme (texte crypté).

Protocole de communication à l'aide d'un cryptosystème à clé secrète

- Choisir un cryptosystème,
- Choisir une clé,
- Chiffrer le texte à émettre à l'aide de l'algorithme et de la clé sélectionnée,
- Envoyer le texte chiffré,
- Réceptionner et déchiffrer à l'aide du même algorithme et de la même clé,.
- Lire le message.

Problèmes

(a) La sécurité du système entier repose sur la sécurité de la clé c'est à dire son secret.
La divulgation de la clé remet en cause :

- La confidentialité : le message peut être lu ou déchiffré,
- L'intégrité : le message peut être modifié, rechiffré et envoyé,
- L'authentification : le message renvoyé et rechiffré remplace le message initial en se faisant passer pour ce dernier.

Conclusion : Les clés ont plus de valeur que tout le message.

Solution

- Nécessité de changer les clés fréquemment,
- Les distribuer secrètement.

(b) Si on utilise une clé différente pour chaque paire d'utilisateurs dans un réseau, le nombre de clés augmente jusqu'à n(n-1)/2 en fonction du nombre d'utilisateurs.

Solution

Il est indispensable de limiter le nombre d'utilisateurs de clés.

b) La cryptographie asymétrique

Dans la cryptographie asymétrique, dite aussi à ***clé publique***, chaque utilisateur désirant recevoir des messages chiffrés possède son propre couple de clés (clé privée, clé publique). Il est fondamental que la connaissance de la clé publique n'entraîne pas la détermination de la clé privée.

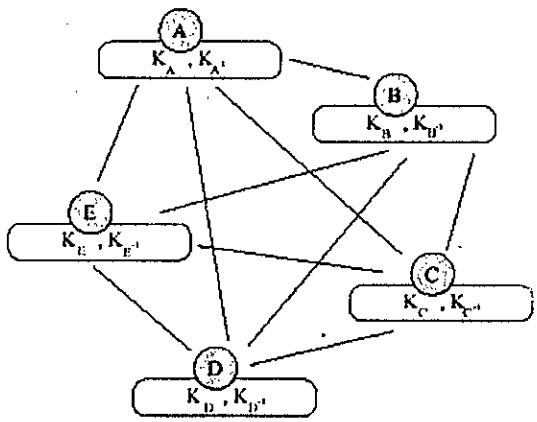


Figure 1.7 : Système asymétrique

Ainsi, si une personne A veut communiquer avec une autre personne B elle utilise la clé publique de B pour chiffrer le message et B pourra effectuer le déchiffrement grâce à sa clé privée correspondante. Pour recevoir des messages chiffrés de quiconque, il est nécessaire de diffuser ouvertement sa clé publique, éventuellement en l'inscrivant sur un annuaire.

La cryptographie asymétrique corrige, en partie, les inconvénients de la cryptographie symétrique : il n'y a plus de partage de secret et la diffusion des clés est facilitée.

En revanche, la lenteur des algorithmes de chiffrement / déchiffrement asymétrique est parfois rédhibitoire à leur utilisation. Des précautions sont à prendre pour s'assurer de l'authenticité du propriétaire des clés.

Ce principe de la cryptographie asymétrique est apparu publiquement en 1976, dans l'article [de Diffiee et Hellman]. La publication d'un algorithme, basé sur le principe de la cryptographie asymétrique, est apparue en 1978 lorsque Rivest, Shamir et Adleman ont présenté leur schéma, R.S.A. (Rivest Shamir Adleman), encore utilisé dans le monde.

Pour éviter d'éventuelles interceptions de clés, des algorithmes complexes ont été développés tel que PGP (Pretty Good Privacy) qui utilise plusieurs types d'algorithmes

(clés publiques et privées) pour sécuriser les correspondances du courrier électronique. Il bénéficie ainsi des avantages de chaque type et limite les risques en ce qui concerne la gestion des clés, l'authentification et la confidentialité des données.

Protocole de communication d'un cryptosystème à clé publique

- **Première alternative**
  1) Choisir un cryptosystème à clé publique,
  2) Le récepteur envoie à l'émetteur la clé publique,
  3) L'émetteur chiffre le message et l'envoie,
  4) Le récepteur déchiffre le message à l'aide de sa clé privée.

- **Deuxième alternative :**
  1) L'émetteur collecte la clé publique d'une base de données,
  2) L'émetteur chiffre le message et l'envoie à l'aide de la clé publique,
  3) Au moment de la consultation de son courrier, le récepteur déchiffre le message à l'aide de sa clé privée.

Le récepteur dans le second cas n'est pas concerné par le protocole tant qu'il ne lit pas son courrier.

Attaques contre la cryptographie à clé publique

L'obtention d'une clé publique se fait à partir d'une base de données sûre et publique.

Cependant, un pirate peut toujours être tenté :

1) D'insérer, dans la base de données, sa propre clé publique au lieu de celle du récepteur et de réorienter les envois.

   Solution : La base de données doit être protégée en écriture sauf pour l'arbitre

2) De remplacer la clé par une autre au cours de la transmission de celle-ci (première alternative).

   Solution : Signer chaque clé publique par la clé privée de l'organisme ou de l'arbitre en tant que :

   - Autorité d'authentification de clé,
   - Centre de Distribution de Clé (C.D.C).

Le C.D.C signe un message composite comportant :

1) Le nom de l'utilisateur.
2) La clé publique,
3) Les autres informations concernant l'utilisateur.

Le message composite signé est stocké dans la base de données C.D.C.

3) De remplacer la clé stockée chez l'émetteur par sa clé, en faussant ainsi la base de données le tout signé par sa clé privée comme s'il était le C.D.C.

c) Cryptosystèmes hybrides

Les algorithmes à clé publique sont plus lents que les algorithmes à clé privée ou secrète.

Solution : Une clé publique est utilisée pour assurer la sécurité de la distribution des clés, et une clé privée est utilisée pour protéger les messages transmis durant l'échange d'information.

De tels systèmes sont appelés cryptosystèmes hybrides.

1.2.2 Intégrité

La confidentialité des messages reste insuffisante sans une garantie de leur intégrité. Il s'agit de prouver que le message chiffré par l'émetteur est bien identique au message déchiffré par le destinataire et qu'il n'y a pas eu de modifications provenant d'un attaquant.

Pour cela, on utilise fréquemment une fonction de hachage qui calcule une empreinte du message, de longueur plus courte que celui-ci. Le message envoyé est accompagné de son empreinte et le destinataire peut alors calculer l'empreinte du message reçu en la comparant avec l'empreinte envoyée. Une bonne fonction de hachage est rapide à calculer, à sens unique, et sans collision, c'est-à-dire qu'il doit être impossible, en pratique, de trouver deux messages distincts qui se hachent en la même valeur.

Cependant, il faut s'assurer que c'est bien l'émetteur déclaré qui a émis le message et son empreinte (un attaquant pourrait remplacer à la fois le message et l'empreinte) en signant numériquement le haché du message. La signature du haché est préférable à la signature du message lui-même parce que le haché est beaucoup plus court que le message et donc mieux adapté aux algorithmes à clé publique, généralement plus lents

que les algorithmes à clé privée. En outre, le hachage permet de masquer des relations entre les différents messages signés.

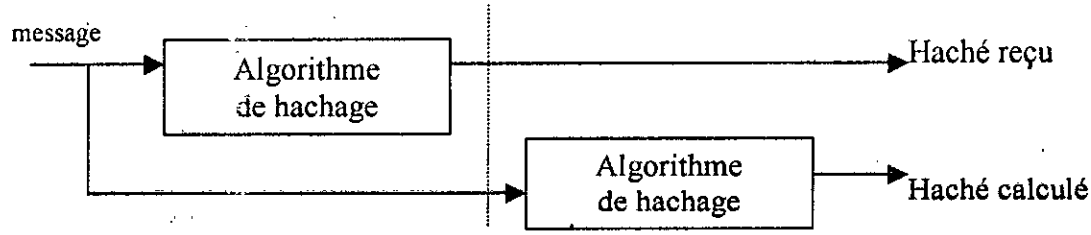


Figure 1.8 : Intégrité

1.2.3 Authenticité

Si la confidentialité et l'intégrité semblent définies universellement, l'authenticité regroupe plusieurs notions dont les définitions varient d'un auteur à l'autre.

a) Authentification de personnes (Identification)

L'identification consiste pour une personne à prouver son identité à une entité informatique.

Très utilisés, les procédés d'identification sont très variés. Leur choix dépend de l'usage que l'on souhaite en faire. Par exemple, un simple code secret numérique (généralement appelé PIN) sera utilisé pour s'identifier auprès d'une carte bancaire, alors qu'un mot de passe plus long sera privilégié pour un accès à un réseau informatique, afin de rendre les recherches exhaustives sur l'ensemble des mots de passe impossibles en pratique. Comme un mot de passe est rarement aléatoire, des recherches utilisant des heuristiques peuvent être menées à bien. Utiliser des mots de passe aléatoires n'améliorerait pas le problème, car les individus seraient alors obligés de noter leurs mots de passe pour s'en souvenir.

D'autres méthodes peuvent alors être utilisées : pour les accès les plus sécurisés, on peut noter la reconnaissance de l'empreinte digitale, la reconnaissance de la voix, la signature manuscrite (éventuellement dynamique) ou encore le système réputé le plus sûr à ce jour, à savoir la reconnaissance du dessin rétinien, mais qui est plus contraignant et plus difficile à mettre en œuvre.

Quoiqu'il en soit, le principe est toujours le même : l'entité informatique pose, implicitement ou explicitement, une requête à l'individu devant s'identifier, et celui-ci doit y répondre correctement, éventuellement avec une réitération du processus pour assurer une meilleure sécurité. La requête peut être très diversifiée : question, reconnaissance physique,...

b) Authentification d'entités

L'authentification d'entités informatiques poursuit les mêmes objectifs que l'authentification de personnes, en tenant compte des différences inévitables qui apparaissent : il n'est plus possible de se baser sur des caractéristiques physiques, mais il est en revanche possible de requérir des calculs plus importants.

L'entité désirant s'authentifier est fréquemment appelée le prouveur, et l'autre entité le vérifieur. On demande aux protocoles d'authentification d'assurer les conditions suivantes :

- Un prouveur honnête doit toujours être reconnu comme tel par un vérifieur honnête,
- Un prouveur malhonnête doit toujours être reconnu comme tel par un vérifieur honnête,
- Un vérifieur ne doit pas pouvoir se faire passer pour un prouveur qui se serait authentifié auprès de lui précédemment
- L'écoute de plusieurs authentifications d'un prouveur honnête ne doit pas permettre de se faire passer pour lui ensuite.

La figure 1.9 représente les échanges entre le prouveur et le vérifieur dans un schéma d'authentification dit à deux passes.

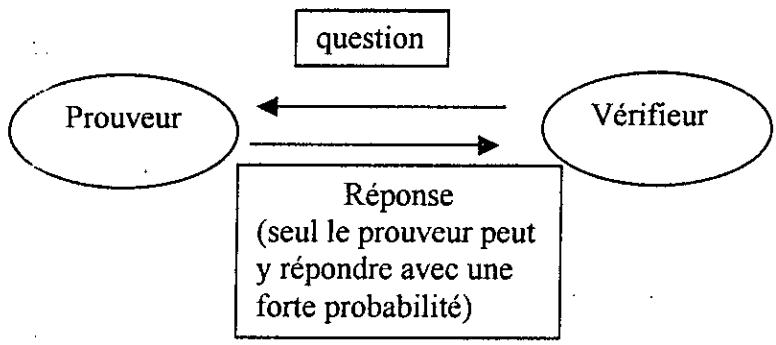


Figure 1.9 : Schéma d'authentification à deux passes

c) Authentification de documents (Signature numérique)

La signature numérique est au message numérique ce que la signature manuscrite est au document papier : son objectif est de certifier l'identité de l'auteur du message.

Contrairement à l'authentification d'entités informatiques qui ne possède qu'une valeur éphémère, la signature numérique perdure dans le temps, éventuellement pour une durée donnée. D'autre part, elle introduit la notion de tiers de confiance, qui pourra vérifier la validité de la signature.

La sécurité d'une signature manuscrite repose sur le fait qu'elle est très difficilement reproductible pour une personne qui n'en est pas l'auteur légitime, mais aussi qu'une

copie trop parfaite (reproduction par photocopie par exemple) n'est pas crédible. Il est facile de voir qu'un tel procédé ne peut être utilisé dans le monde numérique, et notamment qu'il n'est pas concevable de scanner la signature manuscrite et de l'adjoindre au message.

La signature numérique, introduite pour la première fois par Diffie et Hellman, ne va donc pas dépendre uniquement de l'auteur, mais également du message, comme indiqué sur la figure 1.10.

Pour être utilisable en pratique, un algorithme de signature numérique doit répondre à certaines exigences :

- Chaque signataire doit pouvoir signer facilement n'importe quel document,
- Tout le monde doit pouvoir vérifier, uniquement avec des données publiques, et sans l'aide du signataire, que la signature correspond à celle de l'émetteur du message,
- Tout le monde doit pouvoir vérifier que la signature correspond au message reçu,
- Personne ne doit pouvoir signer un message en se faisant passer pour une autre personne, et ainsi après avoir signé un message, le signataire ne doit pas pourvoir à nier sa signature : c'est la propriété de *non-répudiation* de la signature numérique.

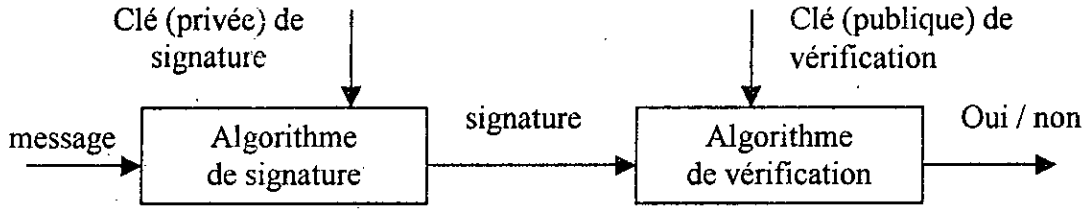


Figure 1.10 : Signature Numérique

Les contraintes posées ci-dessus montrent que la cryptographie à clé publique sera préférée à la cryptographie à clé privée. Certains algorithmes symétriques sont toutefois utilisés pour les signatures numériques, mais la dernière contrainte n'est alors pas respectée car la clé étant commune entre le vérifieur et le prouveur (le signataire), celui-ci peut répudier la signature en prétextant une malhonnêteté du vérifieur qui aurait pu signer lui-même le message.

1.3 La gestion des clés « *Key Management* »

1.3.1 Introduction et définitions

La gestion des clés est un ensemble de processus et mécanismes qui maintiennent l'établissement des clés ainsi que l'entretien de relations renforcées et avancées entre les parties.

L'établissement des clés *(Key Management)* consiste à mettre à disposition des clés dans l'optique d'une utilisation cryptographique ultérieure. L'établissement des clés peut être divisé en deux étapes : l'accord de l'attribution de la clé et le transport e la clé.

La préoccupation principale de la gestion de clé est celle de fournir des procédures sécurisées dans le but de manipuler le matériel cryptographique à clé et d'assurer son utilisation dans les mécanismes symétriques ou asymétriques de cryptographie. Elle concerne :

- La registration ou l'enregistrement de l'utilisateur,
- La génération de la clé,
- La distribution de la clé,
- Le remplacement de la clé,
- La mémorisation ou sauvegarde de la clé,
- L'effacement de la clé.

L'objet central de la gestion de clé constitue la distribution de clé ou l'établissement d'une clé dans le souci d 'assurer :

- L'origine,
- L'intégrité,
- La confidentialité,
- L'authentification des entités.

Afin de protéger les clés, on pourrait utiliser certaines techniques cryptographiques connues. Cependant beaucoup des propriétés de protocoles de gestion de clé ne dépendent pas de ces algorithmes de cryptographie mais de la structure du message échangé. La plupart du temps des bogues figurant dans de tels protocoles ne résultent pas de la vulnérabilité de l'algorithme cryptographique mais ont pour source des erreurs à un degré plus haut du design.

1.3.2 Distribution des clés :

Les clés doivent être générées et choisies d'une manière aléatoire.

La distribution des clés peut se faire :

1. Manuellement,
2. Automatiquement : cette méthode emploie différents types de message, une transaction est lancée par la requête de la clé adressée à un mécanisme central (centre de distribution de clé) où une entité avec laquelle une clé sera échangée. Les messages peuvent contenir les clés mais aussi d'autres dispositifs à clé comme les normes, les types d'entités, les clés ID, les valeurs aléatoires, les contre-attaques, les compteurs.

Il est donc nécessaire de concevoir un moyen de distribuer les clés à travers le même canal de communication que les données. Malheureusement, la distribution sécurisée des clés requiert aussi une protection cryptographique, on a pensé alors à utiliser d'autres clés pour encrypter les clés originales, ceci étant une solution à la distribution des clés à travers les canaux non sécurisés. Cette notion nous permet de réduire le nombre de clés à protéger par rapport aux clés utilisées. On a ainsi introduit la notion de hiérarchie des clés réduisant efficacement le nombre de clés non distribuées automatiquement ; les clés peuvent ainsi être indexées selon leurs fonctions ou utilisations.

1.3.3 Principes de sécurités d'un système de gestion de clé

Un mécanisme ou système de gestion de clé peut être assimilé à une enceinte évoluant dans des procédures de gestion de clé. L'exigence fondamentale en sécurité adressée à un système de gestion de clé est le contrôle du matériel des clés pendant sa durée de vie afin de prévenir :

- La révélation de la clé,
- Les accès non autorisés,
- La modification,
- La substitution,
- La réutilisation (replay).

La sécurité d'un système de gestion de clé dépend de :

- La fonctionnalité des mesures de sécurité,
- La qualité des mesures de sécurité (la force de l'algorithme cryptographique, et la qualité de l'implémentation),
- La sécurité physique de certains dispositifs et canaux ou réseaux de communications,

- Avoir confiance en les gestionnaires de clé et en les constructeurs du système.

Ce dernier problème est facilité par la division des responsabilités ; une seule personne ne doit pas posséder toute l'information à propos d'un matériel important (par exemple copie complète d'une clé). Le processus de division des connaissances d'une information entre différentes entités séparées opérant en accord est appelé ***contrôle dual.***

Les clés distribuées manuellement seront protégées en utilisant la sécurité physique et le contrôle dual car les canaux physiques réalisent la confidentialité et l'authentification.

Le danger du compromis d'une clé augmente avec :

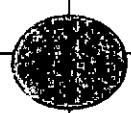
- Le temps de l'opération,
- La quantité de données enchiffrées par cette clé.

Le changement fréquent des clés ou alors même leur hiérarchisation sont des techniques qui présentent une bonne solution pour éviter la sécurité physique qui est souvent coûteuse et à laquelle on fait recours lorsque les clés ne peuvent être encodées par d'autres clés.

Un exemple de contrôle dual est l'utilisation de deux clés générées indépendamment, enregistrées puis transportées séparément et enfin soumises à une opération de XOR dans un environnement sécurisé afin de produire la clé actuelle de cryptographie.

On a ainsi introduit le principe de partage de secret dont relève le cas particulier cryptographie à seuil qui sera présentée dans le chapitre II de cette partie.

CHAPITRE 2 Cryptosystèmes à seuil



2.1 Introduction : Importance du concept

Aujourd'hui, les communications concernent, de moins en moins, les structures traditionnelles d'émetteur / récepteur, d'individu vers individu mais intéressent de plus en plus les structures de groupes, d'entreprises ou d'organisations qui ont besoin de communiquer davantage, de transmettre l'information, de traiter des transactions entre elles ou avec des structures individuelles.

De plus, la sécurité n'est plus l'affaire d'une seule personne mais celle des groupes. Par exemple, l'autorité de signature des documents (négoces, contrats...) se partage entre plusieurs cosignataires. Ce qui entraîne la nécessité de garantir l'authenticité des messages envoyés par un groupe de personnes vers un autre groupe sans provoquer une expansion dans le nombre de clés ou de messages (la taille de l'information).

Pour éviter tout problème de **gestion de clé** et assurer le partage du pouvoir de décider et d'être informé, une organisation doit avoir une seule clé publique, alors que l'autorité de signer de décrypter ou d'utiliser un cryptosystème (la clé privée) doit être partagée, pour empêcher les abus de pouvoir et, par conséquent, assurer la crédibilité du groupe.

Dans ces cas de figures on fixe souvent un seuil de personnes (décision, clé...) au-delà duquel l'information est valide et au-dessous duquel elle est indéchiffrable invalide ou non révélée.

Le but des cryptosystèmes à seuil est de concevoir dans une société électronique des systèmes où le pouvoir d'exécuter certaines opérations est partagé. Deux propriétés fondamentales se dégagent des systèmes à seuil :

- Principe de seuil et partage de décision
- Structure d'accès général

Une autre propriété des cryptosystèmes à seuil est la sécurité et la fiabilité qu'exigent certaines organisations.

2.2 Notions fondamentales

2.2.1 Notions relative au secret morcelé

Toute information divulguée à un utilisateur ayant de mauvaises intentions peut se révéler dangereuse. Pour remédier à cela, on peut faire appel au secret morcelé. La technique consiste à prendre le message et à le découper en morceaux. Chaque morceau pris isolément n'a aucune signification alors que les morceaux rassemblés constituent

l'information secrète à ne pas divulguer. Ainsi si une personne mal intentionnée part avec un morceau du secret, cette information est inutilisable par elle-même. Le morcellement le plus simple partage un message entre deux personnes.

Voici un protocole où I partage un message entre A et B.

1. I engendre une chaîne aléatoire de bits, R, ayant la même longueur que le message M.
2. I combine M avec R par ou exclusif pour obtenir P. Ainsi : $M \oplus R = P$.
3. I donne P à A et R à B.

Pour reconstruire le message, A et B n'ont qu'une étape à effectuer :

4. A et B combinent leurs morceaux par ou exclusif pour reconstruire le message. Ainsi : $P \oplus R = M$.

Cette technique si tout est fait correctement est parfaitement sûre. Chaque élément étant lui-même sans valeur. Ce n'est qu'ensemble qu'A et B peuvent reconstruire le message.

De manière générale, I chiffre le message avec un masque jetable. Il donne ensuite le masque à une personne et le texte chiffré à une autre. Ce système de chiffrement est absolument sûr dans la mesure où aucune puissance de calcul ne permet de retrouver le message avec un seul morceau. Il est facile d'étendre ce schéma à plusieurs personnes. Pour morceler un message entre plus de deux personnes on combine d'autres chaînes aléatoires de bits avec le message.

Il s'agit là d'un protocole à juge-arbitre. I a tous les pouvoirs et peut faire ce qu'il veut. Il peut donner une information incohérente en prétendant que ce sont les bons éléments du secret ; personne ne pourra le savoir avant sa reconstruction. Il peut donner un morceau à A, B, C et D, puis dire à tout le monde que seuls A, C et D sont indispensables pour reconstruire le secret. B n'aura pas de poids dans la chaîne du secret.

Cependant, ce protocole comporte un défaut. En effet, si l'un des morceaux est perdu, le message l'est aussi. Ainsi si une personne disparaît avec un morceau du secret les autres n'auront qu'une information commune : la longueur du message. Le morceau de la personne disparue est aussi important que chacun des autres morceaux.

2.2.2 Notions relatives au secret réparti

Pour qu'une une information demeure secrète sans que personne ne puisse l'utiliser à mauvais escient, il suffit de contrôler le nombre de personnes susceptibles de garder le secret et de le décrypter. Il faut aussi prévoir le cas où l'un des détenteurs du secret est absent.

Seul un schéma de répartition, appelé schéma à seuil, est à même de permettre cela. Au niveau le plus simple, on peut prendre n'importe quel message et le diviser en n éléments appelés parts, de telle manière que n'importe quel ensemble de m parts puisse être utilisé pour reconstruire le message. Plus précisément on parlera de schéma à seuil (m,n).

Avec un schéma à seuil (3,4), I divise son secret entre 4 personnes A, B, C et D de manière à ce que trois d'entre elles puissent assembler leurs parts ensemble pour reconstruire le message. Si C est absent, A, B et D peuvent le faire. Si B est absent, A, C et D peuvent le faire également. Mais si C et B sont tous les deux absents, il sera impossible pour A et D de reconstruire le message.

Les schémas à seuil sont très variés. Cette idée a été inventée indépendamment par Adi Shamir et G.R Blackley, chacun à sa manière.

2.2.3 Algorithmes de partage de secret

Un schéma de partage de secret ou schéma à seuil, fonctionne de la manière suivante. Un message est divisé en n parties, appelées parts, de telle sorte que n'importe quel sous-ensemble de m parts puisse reconstruire le message et que n'importe quel sous-ensemble de (m-1) parts ne le permette pas. On parle alors de schéma à seuil (m,n).

Adi Shamir utilise des équations polynomiales sur un corps fini pour construire un schéma à seuil du polynôme d'interpolation de Lagrange.

L'aspect intéressant du partage de secret est que si les coefficients sont choisis arbitrairement ; (m-1) personnes, par exemple, disposant d'une puissance de calcul infinie ne peuvent rien apprendre à propos du message si ce n'est sa longueur.

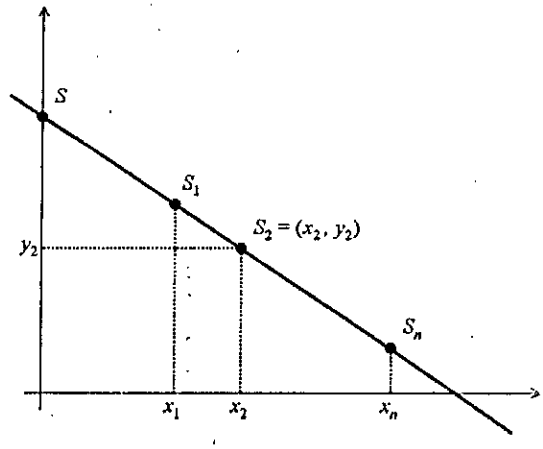


Figure 2.1 : Schéma de partage de secret de Shamir (2 parmi n)

Blakley a inventé un schéma qui utilise des points dans l'espace appelé schéma vectoriel. Le message est défini comme un point dans un espace à m dimensions. Chaque part est l'équation d'un hyperplan de (m-1) dimensions qui inclut ce point. L'intersection de n'importe quel ensemble de m de ces hyperplans détermine exactement ce point. Par exemple, si trois parts sont nécessaires pour reconstruire le message celui-ci sera défini comme un point dans un espace à trois dimensions. Chaque part est un plan différent. Avec deux parts, on sait que le point est quelque part sur la droite d'intersection des deux plans. Avec trois parts, on peut déterminer le point exactement : c'est l'intersection des trois plans.

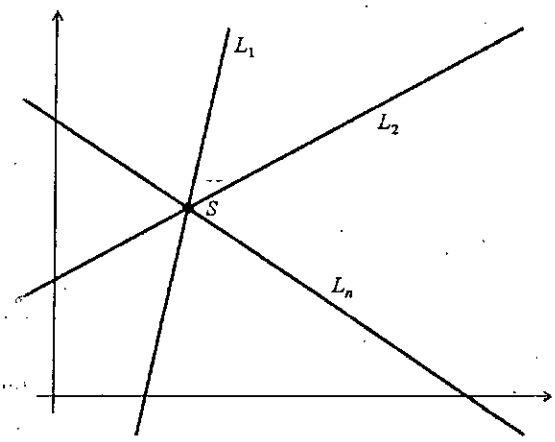


Figure 2.2 : Schéma de partage de secret de Blakley

2.2.4 Schéma à seuil avancé

Les exemples précédents illustrent seulement les schémas à seuil les plus simples. Il existe d'autres algorithmes pour la construction des schémas plus compliqués.

Pour créer un schéma où une personne doit bénéficier d'une importance plus grande que les autres, il suffit de donner plus de parts à cette personne. Supposons qu'il faille cinq parts pour reconstruire le secret. Si une personne dispose de trois parts différentes tandis que toutes les autres n'en ont qu'une seule, alors il lui sera possible avec deux autres personnes de reconstruire le secret.

Deux personnes ou plus peuvent recevoir plusieurs parts. Chaque personne différente peut recevoir un nombre différent de parts. Peu importe la façon dont les parts sont distribuées, le secret peut être reconstruit à partir de m parts quelconques tandis qu'une personne ou un groupe disposant de (m-1) parts ne peut le faire.

En général chaque type de schéma de partage peut être imaginé et réalisé. Il suffit pour cela de trouver un système d'équations correspondant au schéma particulier choisi.

2.2.5 Partage de secret sans révélation de parts

Le défaut des schémas précédents découle du fait que lorsque des personnes décident de se réunir pour reconstruire le secret elles doivent révéler leurs parts. L'utilisation du schéma de Lagrange peut remédier à cette situation. Si le secret partagé est une clé privée alors les n détenteurs de parts peuvent chacun signer partiellement le document. Après la $n^{ième}$ signature partielle, le document est signé avec la clé privée partagée et aucun des détenteurs de parts n'est en mesure de savoir quoi que ce soit sur les autres parts. Ce concept est exploré de manière plus approfondie par Yvo Desmedt et Yair Frankel.

2.2.6 Partage de secret avec des tricheurs

Cet algorithme modifie le schéma à seuil (m,n) standard pour détecter les tricheurs et peut être utilisé avec n'importe quel schéma.

Si une personne s'introduit dans une réunion de reconstruction du secret avec sa fausse part, celle-ci a une forte probabilité de ne pas être valide. Un secret impossible est un secret inutile.

Cependant, une personne se confondant à un tricheur peut apprendre le secret si l'hypothèse que m autres parts sont valides sont faites. L'idée de base consiste à avoir une suite de k secrets tel qu'aucun des participants ne sache d'avance quel est le bon

secret. Chaque secret est plus grand que le précédent sauf pour le vrai secret. Les participants combinent leurs parts pour engendrer les secrets les uns après les autres jusqu'à ce qu'ils créent un secret qui soit inférieur au secret précédent. C'est le bon.

Les tricheurs seront rapidement dévoilés avant que le vrai secret ne soit révélé. Des complications peuvent naître quand les participants dévoilent leurs parts une après l'autre.

2.2.7 Partage de secret avec possibilité d'empêchement

Prenons l'exemple d'un secret divisé entre 50 personnes de telle manière que 10 d'entre elles peuvent se réunir pour reconstruire le secret. Il est plus difficile de réaliser le même schéma de partage de secret avec la contrainte supplémentaire que 20 personnes peuvent se mettre ensemble et empêcher les autres de reconstruire le secret quel que soit leur nombre.

L'idée de base consiste à ce que chacun reçoit une part oui et une part non. Au moment de reconstruire le secret, chacune des personnes soumet une de sa part. La part choisie dépend de leur volonté de reconstruire ou non le secret. S'il y a m ou plus parts oui et moins de n parts non le secret peut être reconstruit sinon, il ne peut l'être. Toutefois, rien n'empêche les partisans du oui de se réunir séparément sans les partisans du non pour reconstruire le secret. Pour cela l'hypothèse qu'ils sont suffisamment nombreux doit être faite sous réserve de soumettre leurs parts à un ordinateur central.

2.3 Synthèse

En plus de leurs applications diverses dans le contrôle d'accès, les schémas à seuil ont répondu à beaucoup d'applications dans un large éventail de protocoles cryptographiques incluant les calculs répartis sûrs (les schémas de vote cryptographiques par exemple), la récupération de clés, le paiement électronique et les cryptosystèmes à seuil (les schémas de signature de groupe par exemple).

Les cryptosystèmes à seuil ont donc été introduits dans le cadre de la cryptographie classique et utilisés pour améliorer ou généraliser de nombreux algorithmes cryptographiques. Et ce grâce à leur fiabilité, efficacité et la multiplicité des alternatives qu'ils proposent quant à la structure participante au cryptage, tel que l'anonymat, la détection des espions et tricheurs etc....

Les schémas à seuil peuvent être appliqués à n'importe quel type de données comme par exemple les images. Mais pour cela l'image I sera chiffrée par une chaine de caractères binaires K(I) qui représentera le secret réparti. Lors de la reconstruction du

secret une seconde conversion de la chaîne de caractère vers l'image sera réalisée. Tout cela étant, bien sûr, réalisé par un ordinateur.

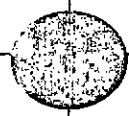
Cette notion de seuil a été reprise pour être appliquée à une nouvelle forme de cryptographie où l'utilisation de ce concept sera différente, mais son but et ses avantages resteront les mêmes.

Cette nouvelle forme de cryptographie qui en est encore à son début est la cryptographie visuelle, une cryptographie qui s'en remet principalement au système visuel de l'homme, plutôt qu'aux performances des calculateurs.

La différence entre les schémas de partage visuels et les schémas de partage conventionnels réside dans la manière de reconstruire le secret. Dans le schéma de partage visuel, les calculs (au décryptage) sont accomplis par le système visuel humain.

La théorie du cryptage visuel constitue la deuxième partie de notre document à travers laquelle cette théorie sera présentée, décrite et détaillée.

CHAPITRE 1 Principes de base de sécurité de l'information



1.1 Notions d'identification et d'authentification visuelles

1.2 Motivations et applications envisagées

1.3 Présentation et travail antécédent

1.1 Notions d'identification et d'authentification visuelles

L'authentification et l'identification sont les thèmes principaux les plus traités en cryptographie.

Lors d'un protocole d'*authentification,* un informateur tente de transmettre un message à un destinataire pendant qu'un adversaire contrôle le canal par où transite la communication.

Si l'adversaire modifie l'information, la change ou la remplace le destinataire devra détecter le changement avec une haute probabilité et signale que la communication a été touchée.

Lors d'un protocole d'*identification,* un utilisateur doit prouver son identité à un vérificateur. Un adversaire tentant de se faire passer pour l'utilisateur ne doit pas être capable (sinon à très faible probabilité) de convaincre le vérificateur qu'il communique avec l'utilisateur.

Dans cette étude, il sera question de scénarios où le destinataire dans un protocole *d'authentification* ou l'utilisateur dans un protocole *d'identification* sont des personnes n'ayant point la capacité d'effectuer des calculs complexes ou puissants, ni celle de mémoriser d'importantes quantités de données. Il n'est pas exigé de ces personnes d'utiliser des dispositifs quelconques de calcul à l'exception de ses capacités naturelles.

Un système est aussi sûr que le moins sûr de ses composants. Le facteur humain dans les protocoles cryptographiques n'a pas été traité rigoureusement auparavant, on s'est limité à l'analyse des systèmes cryptographiques dans lesquels la partie humaine peut être isolée et examinée.

L'authentification par un récepteur humain est un système cryptographique dans lequel l'humain doit résoudre un problème de décision «s'il accepte ou rejette le message reçu ».

L'identification d'un être humain est un protocole où l'adversaire ne peut pas prendre le rôle de l'utilisateur.

L'examen de ces problèmes a pour objectif la construction de protocoles cryptographiques fonctionnels dans lesquels la partie humaine n'utilise pas de dispositifs à part ses facultés et capacités naturelles. L'implémentation de tels protocoles est sans doute moins coûteuse étant donné que l'on a recours à moins de matériel.

Bien que l'homme ne puisse accomplir les calculs effectués aisément par les calculateurs et machines, la perception visuelle humaine peut exécuter des tâches considérées comme compliquées au calcul et à la modélisation.

Les systèmes présentés dans cette étude utilisent les capacités visuelles de l'utilisateur humain. Dans ces systèmes la partie humaine et l'autre partie partagent une information sous la forme d'une image sur un transparent.

Ces systèmes sont basés sur l'idée de la cryptographie visuelle introduite, pour la première fois, par Naor et Shamir à l'Eurocrypt 94 et exposée à la communauté scientifique. Les sessions de l'Eurocrypt comme celles de l'Asiacrypt et Crypto sont sponsorisées, chaque année, par l'Association Internationale de Recherches Cryptologiques (IACR). Depuis, plusieurs articles ou projets ont été présentés afin de développer la cryptographie visuelle, ceci sera relaté plus loin dans l'exposé du travail antécédent.

La sécurité de ces systèmes ne dépend d'aucune supposition de calcul. A la place on pose des hypothèses concernant les facultés et capacités visuelles de l'homme qui sont vérifiables par des tests empiriques.

1.2 Motivations et applications envisagées

L'interaction cryptographique homme-machine a été étudiée dans les deux contextes que nous examinons, certification et identification :

1) Le problème d'authentification a été étudié précédemment dans le contexte du paiement électronique.

Toutes les solutions ayant été suggérées exigent un canal sécurisé entre l'utilisateur (qui est le destinataire) et son ordinateur central sécurisé (l'informateur). Ces méthodes sont applicables uniquement pour les messages textuels (ou seulement numériques).

La certification ou authentification par un récepteur humain, à travers des techniques visuelles est destinée à aider les utilisateurs qui reçoivent des messages d'une partie éloignée à travers un canal non sécurisé. Elle concerne, également, toutes sortes de messages textes, graphiques, manuscrits scannés, etc...

2) les schémas classiques d'identification humaine qui n'exigent pas de dispositifs externes dans le contexte du contrôle d'accès, libèrent l'utilisateur humain d'emporter des appareils informatiques pour le processus de l'identification, seulement ces anciennes méthodes ne sont pas prouvées sécurisées pour exécuter différentes identifications.

Cependant les techniques visuelles permettent un plus grand nombre d'identifications pour un certain montant de mémorisation exigé de l'utilisateur.

Cette propriété aide l'utilisateur à exécuter des identifications sûres avec plusieurs vérificateurs.

Les systèmes d'identification visuels sont qualifiés pour être de technologie à bon marché (*low tech*). De plus, leur implémentation est offerte à tous ceux qui le souhaitent indépendamment. Tout cela libère la sécurité de sa dépendance aux fournisseurs hardware externes.

Pour les applications envisagées on peut considérer un utilisateur qui emploie un terminal et un réseau dont la connexion à un ordinateur éloigné, n'est pas sécurisée. Parmi les applications d'actualité, figure celle qui correspond aux menaces relatives aux paiements électroniques et qui consiste en la certification des messages envoyés d'un portefeuille électronique, le plus communément une *Smartcard*, à son propriétaire

Pour les applications envisagées. Soit :

A: informateur (parfois c'est une «Smartcard » c'est-à-dire carte intelligente).

B : récepteur.

P : adversaire ou espion (parfois c'est un «point of sale » c'est-à-dire point de vente ou de débit).

Dans le plan nous suggérons que B soit équipé avec un (petit) transparent qu'il a acquis d'une manière sûre (main propre, courrier sous-scellé, réseau sécurisé). Quand B place le transparent sur une image qui lui a été envoyée par A à travers un réseau non sécurisé, la combinaison des deux images correspondra au message transmis à B.

L'idée d'équiper B avec un transparent pour l'aider à l'authentification ou l'autoriser à s'identifier peut paraître étrange. Cependant, cette procédure possède d'évidents avantages :

- Un transparent est beaucoup moins cher qu'un périphérique ou dispositif de calcul et les systèmes utilisés possèdent des transparents qui peuvent être assez petits pour être portés dans un portefeuille. De plus, la production des transparents est très simple et les utilisateurs peuvent construire leurs propres systèmes de certification ou identification sans devoir baser leur sécurité sur les fabrications du matériel externe, ce qui minimise le nombre de personnes impliquées.

- Les processus de certification et d'identification sont très simples : l'utilisateur doit seulement placer le transparent sur un écran ou un message imprimé et

envisager le résultat. Il n'a pas besoin de fournir un index de nombres dans un ordinateur ou consulter un code-book.

- Les méthodes de certification visuelle suggérées ont l'avantage supplémentaire d'être applicables non seulement pour les messages textuels mais à tout genre d'images visuelles.
- La sécurité des méthodes de certification et d'identification ne dépendent pas des suppositions de calculs. La probabilité d'échec peut être majorée.

Cette application rejoint deux notions déjà rencontrées précédemment. D'une part, le transparent envoyé sur le réseau non sécurisé qui rejoint la notion de clé publique et, d'autre part le transparent se trouvant chez B et acquis d'une manière sûre (courrier, main propre) qui rejoint celle de clé privée.

Il est possible d'envisager une petite fenêtre transparente dans les documents confidentiels tels que les contrats, les relevés d'identités bancaires, les chèques ou même les billets de banques en introduisant de petites parties transparentes contenant des signatures ou des logos que l'on pourrait identifier ou certifier à l'aide d'un autre transparent et d'un instrument grossissant, tous deux dans la détention de l'identificateur. Ceci serait rajouté aux techniques de watermarking déjà existantes pour prévenir la contrefaçon, par exemple.

Ces techniques trouvent beaucoup d'applications dans le domaine militaire étant données leur facilité d'emploi et leur liberté d'utilisation. Il suffit que les détenteurs des transparents se réunissent sans que chacun sache, au préalable, ce que contient son propre transparent. Ainsi, l'ennemi ne sera pas informé au cas où l'un d'entre eux est capturé.

Ainsi le cryptage visuel est sans doute une économie de temps (rapidité de la perception humaine à reconstruire l'image), d'argent(le type de matériel utilisé) et enfin une économie en coût de formation (stage d'apprentissage d'utilisation dans les entreprises par exemple) et en temps d'apprentissage (n'exigeant aucune connaissance en cryptologie ni en programmation). Le cryptage visuel peut être utilisé par tous, il est en plus une solution économique de cryptage sûre.

1.3 Travail antécédent et présentation

La cryptographie visuelle a été introduite par Naor et Shamir lors de l'Eurocrypt 94. C'est un mécanisme de chiffrement parfaitement sûr et le processus de déchiffrement est exécuté par le système visuel humain. Le texte chiffré est une page imprimée et la clé

est un imprimé transparent de la même dimension. Quand les deux parts sont superposées et alignées avec soin le texte en clair est révélé.

Aucune de ces deux parts ne peut dévoiler, à elle seule, de renseignements au sujet du texte clair. Ce chiffrement peut être considéré comme un schéma 2 parmi 2 du partage de secret (les deux parts constituent le texte chiffré et la clé) et peut être généralisé à un schéma k parmi n de partage de secret.

Un schéma de cryptographie visuelle à seuil (k,n) pour un ensemble P de n participants est une méthode encodant une image secrète SI en n images appelées parts, tel que chaque participant reçoit une part. N'importe quel sous-ensemble qualifié d'un cardinal d'au moins k participants peut reconstruire l'image secrète. Par contre, pour tout sous-ensemble de moins de k participants ou sous-ensemble interdit il sera impossible de reconstruire l'image et de se faire révéler le secret.

La reconstruction de l'image se réalise par la superposition des parts d'un ensemble qualifié, et par leur impression sur un transparent. Aucune connaissance ni pré-requis en cryptographie ne sont exigés des participants de même qu'aucun calcul cryptographique n'est nécessaire.

Dans le schéma 2 parmi 2, le texte clair est traité comme une image ou une collection de pixels. Chaque pixel étant représenté par un carré de 2x2 subpixels

(Figure 1.1).

Chaque image élémentaire du texte en clair est divisée en deux parts de telle sorte que dans chaque part deux subpixels soient noirs et les deux autres transparents.

On suppose que dans la première part les deux subpixels supérieurs sont noirs.

Si dans l'autre part les deux subpixels inférieurs sont noirs, en empilant les deux parts ensemble on compose alors une image où les quatre subpixels sont noirs.

Si, d'un autre côté, les deux subpixels supérieurs de la deuxième part sont noirs, la superposition des deux parts révèle une image dans laquelle deux subpixels seulement sont noirs ensemble.

La première alternative est utilisée pour coder un pixel noir, alors que la seconde est utilisée pour coder une pixel blanc.

Il y a six chemins pour placer deux subpixels noirs dans le carré 2x2.

Pour chaque pixel une de ces combinaisons sera choisie, aléatoirement, dans la première part. L'équiprobabilité des choix attribut un caractère aléatoire au transparent.

La deuxième part sera la même que la première si l'image élémentaire est blanche ou contiendra des subpixels complémentaires si l'image élémentaire est noire.

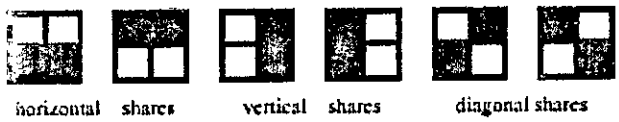


Figure 1.1 : Les dispositions possibles de subpixels

Les schémas de cryptographie visuelle sont caractérisés par deux paramètres :

- *L'expansion du pixel :* qui représente le nombre de subpixels encodant chaque pixel de l'image originelle
- Le *contraste* : qui mesure la différence relative entre les pixels noirs et blancs reconstruits.

Le paradigme de Naor et Shamir a été appliqué à des structures d'accès plus générales selon les sous-ensembles autorisés à reconstruire le secret. Par la suite, plusieurs algorithmes ont été proposés pour la construction du schéma à seuil (k,n) en essayant particulièrement d'améliorer le contraste de certaines constructions tel que (2,n) ou alors de minimiser l'expansion du pixel.

Une variante a été proposée en tentant de dissimuler l'existence d'un secret dans les parts, ainsi le cryptage visuel s'ouvre à un caractère de stéganographie

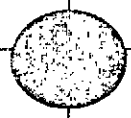
D'autres possibilités ont été présentées en ce qui concerne les méthodes de construction basées sur des parts opaques ou même sur des filtres polarisés. Mais celles ci sont basées sur des hypothèses différentes de celles des constructions classiques précédentes.

Le cas des images colorées ou en niveaux de gris a fait l'objet d'études récentes ; une réalisation a été effectuée, à ce sujet, à l'université de Louvain.

Notre étude abordera en détail les techniques de cryptage visuel développées pour les images en noir et blanc particulièrement. Il sera question ensuite des cas de la couleur, des niveaux de gris et des schémas étendus à la stéganographie.

CHAPITRE 2

Protocoles et méthodes d'authentification et d'identification visuelles



2.1 Introduction

Dans ce chapitre nous allons décrire les divers protocoles d'authentification et d'identification visuelle.

Nous relaterons, ensuite, les différentes méthodes adoptées lors de ces protocoles, puis nous discuterons des avantages et inconvénients de chacune dans une brève comparaison.

Cet exposé nous permettra de donner un aperçu sur le type de conditions qui se posent lors d'un protocole de cryptage visuel. Il nous permettra, également, de présenter les paramètres d'évaluation d'une méthode de cryptage visuel dont principalement la probabilité de détection.

2.2 Système d'authentification visuelle

Le scénario d'authentification visuelle et le protocole d'authentification utilisé dans ce scénario constituent un système d'authentification visuelle.

Définition2.1 : Scénario d'authentification visuelle

Il existe trois entités H, P, S, humaines ayant des capacités humaines.

Les capacités exigées de H doivent être établies pour chaque protocole.

Elles incluent :

- *La capacité d'identifier une image résultat de la composition de deux parties,*
- *La capacité de vérifier qu'une zone est noire,*
- *La capacité de chercher deux images similaires et les détecter.*

Il existe un paramètre de sécurité n de telle sorte que les capacités de mémorisation et le pouvoir de calcul de S et P soient polynomiales en n.

La phase d'initialisation

S (emetteur) produit une suite aléatoire r et crée un transparent T_r et quelques informations A_r fonctions de r.

Leur taille est polynomiale en n (paramètre de sécurité).

S envoie T_r et A_r à H (récepteur) sur un canal d'initialisation privé offline.

S envoie à H un ensemble d'instructions que H établira dans le protocole. Ces instructions sont publiques et peuvent être connues par P(espion) sans qu'il puisse les changer.

Après l'initialisation, toutes les communications se font sous le contrôle de P qui pourrait changer les messages communiqués.

Afin d'assurer la sécurité de tels protocoles la partie humaine du protocole doit être définie explicitement et, par conséquent, il faut isoler les capacités exigées au participant.

La sécurité du protocole doit être réduite à la supposition qu'une personne normale possède ces capacités. Cette supposition est vérifiée à travers des tests empiriques. Malgré la limitation de la puissance de P à être polynomiale en n, celle-ci n'est pas utilisée.

Les schémas suggérés sont sécurisés contre les adversaires avec une capacité mémoire et un pouvoir de calcul illimité.

***Définition 2.2* : Le protocole d'authentification visuelle**

- *S veut envoyer une information m à H dont le contenu est connu par P,*
- *S envoie le message C à H : C= f(m,r),*
- *P change C avant que H ne le reçoive,*
- *H reçoit C' et émet la décision d'accepter ou de rejeter m' en fonction de C', T_r et A_r.*

Si la sortie est acceptée m' est considérée comme ce qui est envoyé par S.

***Définition 2.3* : La Sécurité**

On suppose que H a la capacité exigée par le protocole auquel il participe, par rapport aux instructions données dans le protocole.

On suppose aussi que l'authentification visuelle du système a la propriété que P soit honnête et que H mette en sortie < Accepté m'>.

On appelle le système :

- *(1-p) authentique si quelle que soit m la probabilité que H sorte < Accepté, m' > est au plus p. m'≠m*
- *(1-p) à transformation singulière sûre, «STS » 'Single Transformation Secure' a lieu lorsque : quel que soit m que S envoie à H et quel que soit*

m'≠m (déterminé à priori) la probabilité que H sorte < Accepté, m'> est au plus p.

Le système (1-p) STS est moins sûr que le (1-p) authentique, mais il suffit pour la plupart des applications et particulièrement pour les systèmes de paiement électroniques.

Dans ce modèle l'adversaire peut changer le contenu du message de S vers H selon sa volonté.
Dans chaque pixel nous trouvons deux subpixels noirs.

Il existe deux types de *changement* qui peuvent être réalisés par P :

1. **Type1** : Changer la position des deux subpixels noirs dans les carrés de l'image, ce changement ne pouvant être détecté,

2. **Type 2** : Mettre plus de deux subpixels noirs dans un carré 2x2, ce qui produit alors un transparent illégal. Cependant cette déviation sera probablement non relevée par H sauf si elle est reproduite sur plusieurs pixels.

Il faut s'assurer que l'image ne change plus après avoir désigné et placé le transparent. Le contenu du transparent doit rester secret. Ces définitions concernent les systèmes d'authentification d'un seul message *(One Time Systems)*. Si plusieurs messages doivent être authentifiés on aura recours à des systèmes de sécurité n fois sûrs.

Il existe deux types de *mesure de la complexité*. Les mesures physiques incluent :

- **Premier type**

  - La taille de l'information que l'utilisateur doit porter,
  - Les exigences de calcul et de mémorisation de S,
  - La durée de la communication.

- **Deuxième type**

  - La complexité des opérations que l'utilisateur humain doit établir durant le processus d'authentification

Dans tous les systèmes, on propose des exigences physiques qui sont linéaires avec la taille du message et logarithmiques avec la probabilité par défaut p.

La complexité des opérations que l'être humain élabore ne seront pas mesurées en nombre d'opérations élémentaires comme pour les machines.

Dans chaque cas, on se doit d'expliquer quelles sont les capacités exigées aux participants afin que le schéma soit sécurisé.

Dans certains cas ces capacités sont quantifiées et les autres mesures de complexité sont connectées aux paramètres de ces quantifications.

Les suppositions faites à propos des capacités humaines peuvent être vérifiées expérimentalement. Lorsque ces suppositions sont vérifiées le protocole est prouvé comme étant que complètement sûr.

2. 3 Schémas d'authentification

Cette section décrit les méthodes d'authentification visuelles qui sont applicables à n'importe quel type de données numériques, textuelles ou graphiques.

Les premières méthodes sont utilisées pour une seule authentification ensuite on décrit une méthode valable pour plusieurs authentifications.

2.3.1 Méthode 1 : La zone à contenu et la zone noire

Initialisation

H reçoit un transparent qui est une part du schéma 2 parmi 2 divisé en deux zones : l'une appelée zone à contenu et l'autre zone noire.

La communication authentifiée

S envoie à H un message qui est une partie de 2 parmi 2 du schéma de partage de secret visuel. L'image est une combinaison du transparent et de cette partie où le message m se trouve dans la zone à contenu avoisinant une zone noire. Lorsque celle-ci ne l'est pas entièrement cela signifierait qu'il y a eu une attaque frauduleuse.

L'adversaire a une probabilité de succès de ½ si les suppositions suivantes à propos des *capacités* de H sont réalisées :

a) H peut détecter si la partie qu'il reçoit de S a $|m\Delta m'|$ ou plus de pixels dans lesquels le nombre de subpixels noirs est différent de deux ; tels que m et m' sont des messages quelconques sémantiquement différents. Si $|m\Delta m'|<<$ ceci implique que m n'est pas différent de m'.
b) H est capable de détecter un subpixel blanc dans la zone noire.

La première supposition prévient P des changements dans le message utilisant seulement des changements du type 2.

La seconde protège des changements du type 1 dans la zone noire.

P doit d'abord discerner entre la zone de fond et la zone noire. La probabilité de succès est de un demi (½).

Afin de réduire cette probabilité il suffit d'utiliser K zones. Il y a bien $(2^k - 1)$possibilités de partitions de K zones en zones à contenu et zones noires.

Une de ces partitions est sélectionnée aléatoirement et H connaît à l'avance la bonne combinaison. L'image qu'il observe doit être similaire dans toutes les zones à contenu et toutes les autres zones doivent être noires.

La probabilité de succès de P est $1/(2^k-1)$.

Théorème 2.1

Il existe un schéma d'authentification visuel $(1-1/(2^k-1))$ authentique qui utilise un transparent avec K zones, chacune d'elles étant assez large pour contenir le message.

La méthode suppose que H a la capacité de :

1. *Détecter un pixel blanc dans une région noire,*
2. *Distinguer, pour chacun des deux messages sémantiquement différents m et m', le cas où il y a plus de $|m\Delta m'|$ pixels avec plus ou moins 2 subpixels noirs dans le message reçu, du cas où il n'y en a aucun non conforme,*
3. *Comparer les K zones et vérifier qu'elles contiennent toutes le même message.*

Il existe une variante de cette méthode un peu moins efficace qui n'exige pas de l'utilisateur une vérification des pixels illégaux dans l'image reçue avant de placer le transparent.

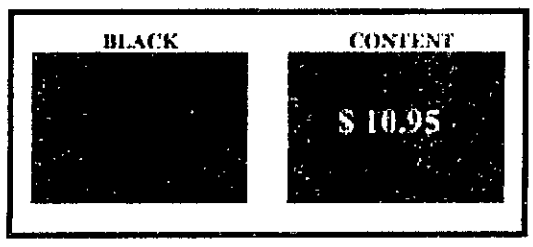


Figure 2.1 : Résultat de la superposition des transparents et image communiquée dans la méthode ' zone noire et zone à contenu'

2.3.2 Méthode 2 : Position sur l'écran

Initialisation

L'image est composée de r×c pixels. On dessine une boîte bornée de taille r'×c' pixels avec une ligne très fine dessinée à un endroit aléatoire du transparent envoyé à H.

Communication authentifiée

Le message se trouve dans la zone dessinée bornée après combinaison des transparents avec la partie communiquée, en blanc sur un fond noir qui couvre les pixels à l'intérieur et à l'extérieur du carré.

Quel que soit m'≠m, l'adversaire a une probabilité faible de succès dans le changement de m en m'. L'objet de l'adversaire est d'inverser les pixels de $m_d = m \Delta m' = (\overline{m} \cap m') \cup (\overline{m'} \cap m)$ pour l'image à l'intérieur du carré, il a été prouvé que la probabilité d'inversion des pixels est faible.

Si l'on suppose que H a une vision précise, et qu'il détecte la différence entre deux images m≠m' même si elle est en un seul pixel ; et soit $m_d^{i,j}$ l'ensemble des pixels correspondants à l'ensemble m_d dans le carré borné localisé aux coordonnées (i,j.).

Si P ne modifie pas exactement les pixels de $m_d^{i,j}$ il échoue son attaque.

Pour deux endroits différents (i,j) et (i',j') on obtient : $m_d^{i,j} \Delta m_d^{i',j'} \neq 0$. Soit donc (r-r') × (c-c') différentes zones équiprobables.

La probabilité que P réussisse est $1/\left[(r-r') \times (c-c')\right]$

Une autre supposition plus souple sur les capacités de l'utilisateur est qu'il distingue la différence entre t pixels de l'image révélée et m' dans les limites du carré borné. Si la différence est supérieure ou égale à t, P échoue.

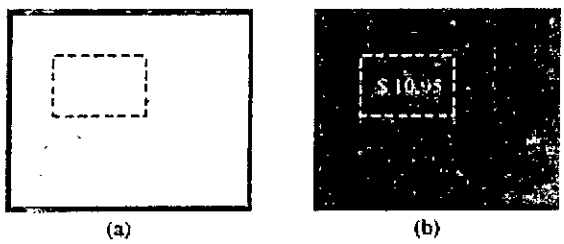


Figure 2.2 : (a) Le transparent de l'utilisateur avec boite limitée
(b) l'image combinée

Théorème2.2

Soit r et c le nombre de lignes et de colonnes dans une image.

r' et c' ces mêmes valeurs concernant une boîte limitée ou bornée.

m : message communiqué par S.

m' : message différent sémantiquement.

On suppose que le récepteur H a les capacités suivantes :

1-quelle que soit l'image avec une distance de Hamming supérieure à t, m' n'est pas capturée en tant que m.

2- H remarque si plus que t' pixels possèdent dans l'image révélée plus ou moins deux subpixels noirs

Alors l'authentification du système décrit est $(1-\left[4(t+t')/(r-r')(c-c')\right])$ *c'est un système d'authentification visuel à transformation singulière sûre.*

2.3.3 Méthode3 : Noir et Gris

La sécurité de cette méthode est exponentielle avec la distance de Hamming entre le message originale et le message que l'espion veut introduire à sa place.

L'inconvénient de cette méthode est la réduction du contraste de l'image révélée .

Précédemment on a utilisé le cas 2 parmi 2 où un pixel noir est égal à 4 subpixels noirs et un pixel blanc contient 2 subpixels noirs. Un pixel gris est définit par 3 subpixels noirs et un blanc.

Soit un pixel noir, S_1et S_2 constituent deux de ses parts construites à l'aide du schéma 2 parmi 2. Il est facile de construire une partie S_1' telle que combinée avec S_2 elles composent un pixel gris.

Cependant si S_1 est une partie d'un pixel gris, la probabilité de construire S_1', tel que S_1' combinée avec S_2 composent du noir, est au plus d'un quart(¼).

Lorsque le message est écrit en noir sur fond gris il est difficile à l'adversaire de changer un pixel de fond en pixel de message.

Lorsque le message est écrit en gris sur fond noir il est difficile d'effacer un pixel du message et de le transformer en pixel de fond.

Le schéma proposé présente le message en deux zones ; dans l'une, l'affichage est en noir sur gris et, dans la seconde, en gris sur noir.

L'utilisateur est instruit de la vérification que les messages soient similaires et égaux dans les deux zones.

Théorème 2.3

Soit t' la borne supérieure du nombre de pixels ayant un nombre de subpixels noirs différents de deux ; dans la partie envoyée par S ceci reste non remarqué par l'utilisateur.

Quel que soit m' on définit $t_{m'}$ comme le maximum de la distance de Hamming du message visualisé par rapport à m' de telle sorte que l'utilisateur accepte le message en tant que m'.

Soit t la borne supérieure de $t_{m'}$ sur tous les messages m'.

Si le message est visualisé selon le plan suggéré alors :

La distance de Hamming entre m et m' est au moins $2(t'+\frac{4}{3}(1+\varepsilon)t)$

C'est un système d'authentification visuelle (1-p) authentique, $p=2e^{-(\frac{2\varepsilon^2}{1+\varepsilon}t)}$ *est la probabilité de succès de l 'espion.*

2.3.4 Méthode 4 : Méthode à plusieurs authentifications

Les méthodes précédentes suggérées sont sûres pour une seule authentification.

Il est préférable d'avoir recours à des méthodes qui sont sûres pour plusieurs authentifications différentes.

L'application la plus directe d'un schéma à plusieurs fois est d'enregistrer à plusieurs reprises des copies indépendantes de l'un des schémas à une seule utilisation dans différentes parties d'un seul transparent.

Le nombre de copies dans un seul transparent dépendra des paramètres de sécurité qui définissent :

- La taille de la zone utilisée pour chaque copie,
- La taille du transparent.

Cette méthode n'est pas mauvaise du moment que les méthodes utilisées sont efficaces dans l'espace du transparent qu'elles occupent.

Cependant, il est possible de faire mieux étant donné qu'on est limité en pratique, d'une part par la taille du transparent que l'on tend à minimiser au maximum, et, d'autre part, par le nombre d'authentifications qui devraient être maximisé.

***Définition 2.4* : n- fois sécurité**

Un système d'authentification visuel est n fois (1-p) à transformation singulière sûre si ce qui suit est vrai :

Quels que soient les n messages $(m_1,m_2,\ldots,m_n)$.Quel que soit m_i $(1<i<n)$ communiquées de S vers H et quel que soit m' différent de m_i, la probabilité que H mette en sortie <Accepté, m'> est au plus p. Si P est honnête H doit toujours mettre en sortie <Accepté, m>.

Le schéma à plusieurs authentifications utilise les paramètres suivants :

Le message à authentifier est de taille r'×c' pixels ; r_0 et c_0 sont des paramètres de sécurité. La taille du transparent est r× c, tel que :

$$r = r_0 + n_r r'$$
$$c = c_0 + n_c c'$$

Le transparent est utilisé pour $n=n_r\, n_c$ authentifications.

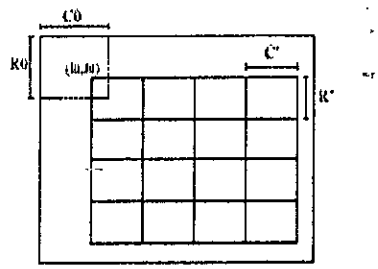


Figure 2.3 : Le transparent de l'utilisateur dans la méthode à plusieurs authentifications

Initialisation

Un point de commencement aléatoire (i_0,j_0) est choisi de telle sorte que :

$1\leq i_0 \leq r_0$, $1\leq j_0 \leq c_0$.

Une grille de n zones apparaît, chacune d'elles composée de r'×c' pixels est dessinée en trait fin sur le transparent débutant à la position (i_0,j_0).

La $i^{\underline{ème}}$ zone est définie comme étant l'intersection de la ligne $[i/n_c]$ et la colonne $(i \bmod n_c)+1$.

1 ère authentification

S envoie sa part du message m_i à H (écrit en blanc sur noir) sur la $i^{ème}$ zone de la grille. Dans tous les autres pixels de la partie, il y a exactement, deux pixels noirs dans deux positions aléatoires. Le récepteur humain vérifie que le message qu'il visualise est dans la $i^{ème}$ zone.

Théorème 2.4

On suppose quesi la distance de Hamming entre l'image visualisée et une image m' est supérieure à t ; alors le récepteur H n'aperçoit pas l'image visualisée en tant que m'.

L'utilisateur remarque si dans plus de t' pixels de l'image communiquée il n' y a pas deux subpixels noirs.

Alors le transparent de taille $(r_0 + n_r r)(c_0 + n_c c')$ *pixels peut être utilisé pour la réalisation d'un système d'authentification visuelle* $n = (n_r \times n_c)$ *fois (1-p) à transformation singulière sûre, où chaque message est de taille* $r' \times c'$, *où* $p = \left[{4(t+t')}/{(r_0 c_0)} \right]$.

2.4 Modèles et définitions de l'identification visuelle

Le scénario de l'identification visuelle est identique à celui de l'authentification. Cependant, le but est différent. Il consiste à aider l'utilisateur humain à prouver son identité au vérificateur S sans consulter de dispositif de calcul. L'objectif de l'adversaire P consiste à convaincre le vérificateur qu'il est l'utilisateur humain en question. Il n'y a pas d'intérêt à construire un protocole d'identification visuelle qui ne permettrait qu'une seule identification sécurisée car il suffirait pour cela d'attribuer un mot de passe OTP(*One Time Pass*)à l'utilisateur. Pour ce faire, il y a lieu de considérer les protocoles d'identification à plusieurs fois où un seul transparent est utilisé pour plusieurs identifications.

***Définition 2.5* : Protocole d'identification visuelle**

On définit le protocole pour la $i^{ème}$ *identification de H envers S :*

- *S envoie une requête* C_i *à H qui est une fonction de l'information secrète r,*
- *H calcule la réponse* A_i *en fonction de* C_i *et ses informations secrètes* T_r *et* A_r *et les renvoie à S,*

- *S décide si H est l'autre partie en se basant sur les messages C_i et A_i et l'information secrète r et répond enfin par accepté ou rejeté.*

Afin de se faire passer pour H, l'adversaire essaie de :

- Questionner H en se déclarant comme étant que S et lui faire une requête d'identité,
- Initier le protocole d'identification avec S et lui envoyer les réponses susceptibles de convaincre le vérificateur.

***Définition 2.6*: protocoles d'identification visuelle ℓ fois (1-p) sûre**

Un protocole d'identification visuelle est ℓ fois (1-p) sûre s'il réalise les conditions suivantes :

Après que l'adversaire ait écouté ℓ_1 identifications répondues par H et prétendu être vérificateur dans au plus ℓ_2 identifications de H, sous la contrainte $\ell_1+\ell_2 \leq \ell$:

- *S accepte toujours lorsque H demande une attribution de protocole*
- *Si l'adversaire P reçoit le message C_i envoyé par S et y répond avec un message b_i fonction de C_i et ℓ communications précédentes, alors S accepte avec une probabilité d'au plus P.*

2.5 Méthodes d'identification visuelle

Les méthodes que l'on suggère pour l'identification visuelle n'utilisent aucun schéma de partage de secret visuel, car il n'y a pas besoin de construire une image perçue par H. Lors d'un protocole d'identification H doit prouver au vérificateur S qu'il connaît certaines propriétés du transparent. On utilise alors des transparents de 10 couleurs différentes que l'on suppose facilement différentiables les unes des autres. Plusieurs ensembles de couleurs peuvent être utilisés et la sécurité dépendra de leur nombre.

L'avantage de ces méthodes est d'être à technologie bon marché. Elles permettent à chacun de construire un schéma d'identification sécurisé qui pourrait favoriser l'accès certaines zones ou identifier certaines parties des communications.

Depuis que le réseau Internet a introduit l'interface graphique universelle, l'identification visuelle peut être élaborée lorsqu'un utilisateur se connecte au serveur à distance tout en utilisant un navigateur pour afficher l'image qui lui est envoyée par le

vérificateur. Dans ce cas, on n'a pas besoin d'utiliser de logiciels spécifiques à l'identification.

Celle-ci peut se faire par une machine ou un être humain.

2.5.1 Schéma d'identification visuelle sûre pour un seul vérificateur singulier

L'unité de base du transparent n'est plus le pixel mais un carré qui est une collection de pixels.

A ***l'initialisation,*** l'utilisateur H reçoit un transparent divisé en plusieurs carrés, chacun d'eux coloré aléatoirement d'une couleur parmi les 10 citées précédemment. L'ordre des couleurs est tenu secret et n'est connu que par H et parfois le vérificateur S.

Soit N le nombre de carrés du transparent et d le nombre de carrés en question dans une identification singulière.

Le ***protocole*** d'identification se présente comme suit:

- S choisit des carrés aléatoires,
- S envoie à H une image complètement noire exceptées les zones des d carrés qui sont blanches,
- H superpose son transparent à l'image et envoie à S les couleurs placées aux endroits adéquats,
- S accepte si les réponses sont correctes.

La stratégie de l'adversaire sera de questionner l'utilisateur ℓ fois et d'apprendre les couleurs de $d\ell$ carrés. P n'a pas d'informations sur les autres carrés.

La probabilité de succès de l'adversaire est $(\frac{1}{10}+\frac{9d\ell}{10N})^d$. Un transparent avec N carrés peut être utilisé pour ℓ = N / 9d identifications et la sécurité sera plus grande que $1-5^{-d}$.

Théorème 2.5

Un transparent avec N carrés colorés de 10 couleurs, peut être utilisé pour un schéma d'identification visuelle l fois $(1-(\frac{1}{10}+\frac{9d\ell}{10N})^d)$ *sûr, telle qu'à chaque identification l'utilisateur envoie au vérificateur les couleurs de d carrés.*

2.5.2 Schéma d'identification visuelle sûre contre une coalition de vérificateurs

Dans ce schéma, l'information secrète r_i que chaque vérificateur S_i reçoit contient les couleurs d'un sous-ensemble aléatoire de (1-q)×N carrés du transparent de l'utilisateur ($0< q < 1$).

Le protocole d'identification est identique au précédent sauf que le vérificateur questionne l'utilisateur à propos des couleurs, des carrés aléatoires appartenant à l'ensemble des carrés dont il connaît la couleur. La densité de ce schéma, c'est-à-dire le nombre important de carrés pouvant être mémorisés dans un seul transparent, permet à ce schéma d'être sûr contre les importantes coalitions.

Théorème 2.6

Lorsque, $\ell \le \frac{(N * q^k)}{2d}$; un transparent avec N carrés colorés de 10 couleurs peut être utilisé pour une ! fois $1-(1-\frac{9}{20}(1-\frac{d}{(1-q)N})^{\ell})^d$ schéma d'identification visuelle sûre, contre k vérificateurs, dans lequel chaque utilisateur doit envoyer les valeurs de d couleurs à chaque identification.

2.6 Synthèse

Toutes les méthodes suggérées sont sûres mais dépendent de la capacité de l'adversaire. Il a été remarqué, auparavant, que l'identification visuelle à plusieurs fois est une technologie à très bon marché et peut être implémentée sans investissements importants.

En comparant les méthodes d'authentification visuelle à une seule fois, on constate que l'avantage de la première méthode (méthode de zone à contenu et zone noire) est la commodité et la simplicité des conditions exigées de l'utilisateur pour assurer la sécurité. Son inconvénient est la perte de surface qu'implique une sécurité plus importante.

L'avantage de la méthode de positionnement sur l'écran est la proportionnalité de la probabilité d'erreur au nombre de pixels et non leur redondance dans la zone. Son inconvénient est que la probabilité de succès de l'espion ne peut pas être petite. Par conséquent, les capacités exigées de l'utilisateur humain sont plus grandes et plus nombreuses.

L'avantage de la méthode noir et gris est que la probabilité de non-détection est exponentiellement petite avec la distance entre les messages sémantiquement différents.

Son inconvénient est la perte de contraste et les capacités additionnelles exigées de l'utilisateur.

En comparaison avec la méthode à une fois, la méthode d'authentification à plusieurs fois présente l'avantage de réduire substantiellement la zone du transparent nécessaire pour l'authentification et au maintien d'un certain niveau de sécurité.

Il serait plus intéressant de trouver une méthode d'authentification où la sécurité est proportionnelle à la taille du message ou une autre méthode qui ne réduise pas le contraste et dont la sécurité serait exponentielle à la différence de Hamming entre les messages.

Un autre problème important se pose. Il attrait à l'examen des capacités humaines pouvant être facilement vérifiées en basant la sécurité des méthodes visuelles sur celles ci (proposer, par exemple, une meilleure mesure que la différence de Hamming pour la différenciation des images).

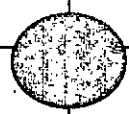
Il serait intéressant par exemple, de quantifier les capacités exigées aux participants, afin d'évaluer et modéliser la mesure de la complexité du système visuel humain, à travers les deux types de mesures physiques citées auparavant. La complexité n'est pas mesurable en nombre d'opérations élémentaires comme pour les machines.

Il figure également, la perspective de désigner une méthode à même de permettre à un informateur humain d'authentifier le message qu'il envoie sans avoir d'interaction à deux voies. Parmi les préoccupations apparentées au sujet, figure celle de trouver une fonction à sens unique facilement calculable par les humains.

Tels sont les protocoles, scénarios et méthodes d'authentification et d'identification visuelle, que nous avons tenu à relater en nous basant sur l'exemple du schéma 2 parmi 2. L'application de ces méthodes est aussi possible dans le cas général des deux schémas à seuil «k parmi n » et «n parmi n », objet des prochains chapitres, que l'on se propose de présenter et d'expliciter pour découvrir un aspect plus détaillé et plus technique du cryptage visuel.

CHAPITRE 3

Modèle et propriétés d'un schéma de cryptage visuel



3.1 Introduction

Dans ce chapitre, il s'agira de présenter les paramètres essentiels intervenant dans un Schéma de Cryptage Visuel et de définir ensuite le Schéma de Cryptage Visuel (VCS) en énonçant quelques conditions dont nous relèverons les plus importantes des propriétés et problèmes d'un VCS.

Nous procéderons enfin à une généralisation des structures d'accès général d'utilisateurs.

Tout cela nous permettra de mieux concevoir la nécessité d'appliquer un schéma optimal répondant à la coexistence de divers problèmes et limitations.

3.2 Modèles et définitions

Les informations à crypter seront traitées sous forme d'images.

Dans un premier temps il sera question, dans cette étude, des images en noir et blanc.

Une image est représentée sous forme d'une collection de pixels noirs et blancs où chacun d'eux sera traité séparément.

Chaque pixel apparaîtra sous n versions modifiées, chacune d'elle appelée part (*share*) sera imprimée sur un transparent pour le modèle de base, afin de pouvoir les empiler ou superposer pour reconstruire le pixel original, visualisant le résultat combiné à l'aide d'un projecteur, par exemple.

Chaque pixel de chaque part est composé de m subpixels noirs et blancs disposés à très grande proximité de telle sorte que le système visuel humain moyenne les contributions de noir et de blanc.

La structure résultante peut être décrite par des matrices booléennes de dimension n×m tel que :

$S_{ij} = 1$ Si le j^{eme} subpixel du i^{eme} transparent est noir,
$S_{ij} = 0$ Si le j^{eme} subpixel du i^{eme} transparent est blanc.

Chaque ligne correspond à une part ou transparent (*share*) et chaque colonne décrit la couleur d'un même subpixel sur chaque transparent.

Empiler ou superposer les transparents revient à effectuer une opération de «OU » aux lignes de la matrice.

Le niveau de gris du transparent résultant de la reconstruction est proportionnel au poids de Hamming du vecteur V, résultat de l'opération «OU » sur toutes les lignes.

Le niveau de gris dans un pixel est interprété par le système visuel humain du participant. La décision d'attribuer le pixel au blanc ou noir repose sur certaines lois du contraste que l'on énoncera plus loin.

La structure algébrique du schéma est celle d'un semi-groupe étant donnée l'absence de la propriété de symétrie relative à l'additivité des couleurs. Cette dernière propriété annule et exclut les techniques classiques d'encryptage consistant en l'ajout de bruit blanc au texte clair et son extraction au texte chiffré lors du décryptage.

Elle exclut aussi la représentation d'un pixel blanc par une totalité de subpixels blancs.

La distinction entre les couleurs avec la prise en compte des conditions précédentes est gérée par les notions de seuil d et de différence relative $\alpha > 0$ que l'on utilisera comme dans la définition suivante :

Définition 3.1

Un schéma de partage de secret visuel k parmi n consiste en deux collections de matrices booléennes C_0 et C_1.

Pour partager un pixel blanc(respectivement Noir) on choisit aléatoirement une des matrices S de C_0 (respectivement C_1).

La matrice choisie définit la couleur des m subpixels dans chacun des n transparents ou parts (shares).

1- $\forall S \in C_0$ le vecteur V résultant du «OU» de k quelconques des n lignes satisfait : $H(V) \leq d - \alpha(m).m$.

2- $\forall S \in C_1$ le vecteur V résultant du «ou » de k quelconques des n lignes satisfait $H(V) \geq d$.

3- $\forall \{i_1, i_2, \ldots, i_q\}$ sous ensemble de $\{1,2,3,,n\}$ avec $q<k$, les deux collections D_t, $t=0,1$, de matrices de dimensions $q \times m$; obtenues par la restriction de chacune des matrices des collections C_t, $t=0,1$, aux lignes $i_1, i_2, \ldots, i_q$; sont impossibles à distinguer, dans le sens où elles contiennent les mêmes matrices avec les mêmes fréquences.

- Les deux premières propriétés sont relatives au *contraste* de l'image.

Le contraste est la mesure de la différence entre les pixels reconstruits noirs et blancs. Les références littéraires suggèrent que le contraste entre deux régions optiques soit caractérisé par la différence relative de l'irradiation des régions.

α : exprime la différence relative dans le poids entre les pixels reconstruits noirs et blancs, à l'origine. Le contraste est évalué par $\alpha^* m$.

On observe donc, une perte en contraste dans l'image reconstruite.

On obtient un schéma de cryptage visuel VCS optimal en maximisant α.

Les propriétés 1 et 2 de la définition 3.1, expriment que l'empilement ou la superposition de k transparents au moins, permet de révéler l'image partagée et dissimulée ; elles expriment aussi, en portant des seuils fonction de d et α au poids de Hamming, des conditions établissant une logique de décision sur la nature ou la couleur du pixel reconstruit (s'il est noir ou blanc).

- La seconde propriété est appelée *Sécurité.* Elle implique que la superposition de moins de k transparents ou parts, ne révèle aucune information sur l'image originale dissimulée ni sur la couleur du pixel. Le niveau de gris ou poids de Hamming de V est le même quelle que soit la collection de matrices sélectionnées.

Dans la plupart des constructions, il existe une fonction *f,* telle que la part reconstruite par la superposition de $q< k$ transparents, concerne tous les vecteurs V avec $H(V)=f(q)$, munie d'une distribution de probabilité uniforme, indépendamment du choix des matrices C_0 ou C_1. Un tel schéma est qualifié d'*uniforme*.

- Une autre propriété importante du Schéma de Cryptage Visuel est *l'expansion de pixel* m, qui représente nombre de subpixels encodant un seul pixel de l'image originale.

  Cette mesure représente la perte en résolution de l'image reconstruite par rapport à l'image originale.

  Un Schéma de Cryptage Visuel optimal correspond à un m aussi petit que possible.

- C_0 et C_1 sont des ensembles de collection de matrices du schéma. On suppose que leurs cardinaux soient égaux, leur construction consistera alors en la permutation, modulo (m !) des colonnes d'une seule matrice A_0 et A_1 de chacune des collections C_0 et C_1 respectivement. Ainsi le cardinal de chacun de ces ensembles sera m ! matrices.

Ceci correspond au cas où A_0 et A_1 sont des matrices de bases. Une matrice de base A_i est définies par le fait qu'elle réalise les conditions de contraste et de sécurité ; puis la collection C_i sera l'ensemble des permutations de cette matrice.

On observe l'avantage de l'implémentation de ce schéma qui économise de l'espace mémoire ; étant donné qu'elle est déterminée par A_0, A_1 et la permutation de leur m colonnes, opération à réaliser par le calculateur.

On résume les paramètres caractéristiques d'un schéma de cryptage visuel en la liste suivante :

1. n : le nombre de participants, fixé et prédéterminé,
2. k : le seuil des participants qualifiés ; déterminé par les besoins de sécurité,
3. h : la blancheur du pixel reconstruit blanc, le nombre de subpixels blanc dans celui ci,
4. ℓ : la blancheur du pixel reconstruit noir, le nombre de subpixels blancs dans celui ci,
5. r : la cardinal de C_0,C_1,
6. m :l'expansion du pixel
7. α : le contraste.

On pourra ainsi reformulé la définition 3.1 comme il suit :

Définition 3.2

Un schéma de partage de secret visuel k parmi n consiste en deux collections de matrices booléennes C_0 et C_1.

Pour partager un pixel blanc(resp. Noir) on choisit aléatoirement une des matrices S de C_0 (resp. C_1).

La matrice choisie définit la couleur des m subpixels dans chacun des n transparents ou parts (shares).

1- $\forall S \in C_0$ le vecteur V résultant du « ou » de k quelconques des n lignes satisfait $H(V) \leq m - h$

2- $\forall S \in C_1$ le vecteur V résultant du « ou » de k quelconques des n lignes satisfait $H(V) \geq m - \ell$

3- $\forall \{i_1, i_2, \ldots, i_q\}$ sous ensemble de $\{1,2,3,,n\}$ avec $q<k$, les deux collections D_t, $t=0,1$, de matrices de dimensions $q \times m$; obtenues par la restriction de chacune des matrices

des collections C_t, $t=0,1$, aux lignes $i_1,i_2,\ldots,i_q$; sont impossibles à distinguer, dans le sens où elles contiennent les mêmes matrices avec les mêmes fréquences.

Quant à la relation entre l'expansion du pixel, le schéma choisi ou même le nombre de transparents générés, elle sera étudiée sans le prochain chapitre tout en considérant les deux aspects suivants relatifs à la disposition des subpixels à l'intérieur du pixel originel substitué :

1. L'équiprobabilité

C'est d'une manière aléatoire, que l'on choisit une matrice S^i de la collection C_i pour chaque pixel de chaque transparent. Le choix est équiprobable ; et l'apparition d'un pixel dans une part est équiprobable aux autres pixels, qui contiennent un même nombre de subpixels disposés différemment.

Cela donne au transparent un caractère aléatoire, faisant ressembler son contenu à du bruit.

Seulement on préfère qu'il y ait autant de subpixels noirs que blancs dans un pixel, pour qu'il en résulte un gris médium, et qu'à travers tout le transparent il y'ait autant de subpixels noirs que blancs.

Ainsi on conserve une certaine équiprobabilité et l'on attribue un caractère aléatoire aux transparents. Cela explique le fait que l'on ne puisse en extraire aucune information et qui justifie leur ressemblance à du buit blanc.

2. Structure et disposition des subpixels

On se réfère souvent au cas où le pixel se décompose en carré avec autant de lignes que de colonnes de subpixels car cela nous évite la distorsion de l'image en proportion et son extension en longueur ou en largeur selon les cas.

Ainsi, on ne respecte pas toujours les formules mathématiques et les techniques théoriques. Lorsqu'il s'agit de choisir l'expansion du pixel ou la répartition des couleurs on a souvent recours à des arrondis pour éviter la distorsion, tout en prenant en compte la perte en résolution, en contraste et surtout la sécurité du système.

3.3 Généralisation d'un Schéma de Cryptage Visuel à une structure générale

Soit P {1,2,..,n} un ensemble d'élements appelé ensemble de participants.

2^P est l'ensemble des sous ensembles de P.

Γ_{qual} est l'ensemble des qualifiés.

Γ_{forb} est l'ensemble interdit.

$\Gamma_{qual} \subseteq 2^P / \Gamma_{forb} \subseteq 2^P$, $\Gamma_{qual} \cap \Gamma_{forb} = \emptyset$.

$(\Gamma_{qual}, \Gamma_{forb})$ est une structure d'accés.

Γ_0 est l'ensemble des ensembles qualifiés minimaux :

$\Gamma_0 = \{A \in \Gamma_{qual} : A' \notin \Gamma_{qual}, \forall A' \subset A\}$

p est un participant essentiel si :

$\exists X \subseteq P$ tel que :

$X \cup \{p\} \in \Gamma_{qual}$; $X \notin \Gamma_{qual}$.

Un participant non essentiel n'a pas de participation active à la reconstruction du message.

- Une structure d'accès robuste (*strong*) est telle que :

  1. Γ_{qual} est monotone croissante
  2. Γ_{forb} est monotone décroissante
  3. $\Gamma_{qual} \cup \Gamma_{forb} = 2^P$;

Γ_0 est appelée base

$\Gamma_{qual} = \{C \subseteq P : B \subseteq C \text{ pour } B \in \Gamma_0\}$

Γ_{qual} est une cloture de Γ_0.

Comme tous les Γ_{forb} sont monotones décroissantes et car un sous-ensemble d'un ensemble interdit est interdit il reste seulement les deux autres conditions 1 et 3.

Il faut noter aussi qu'un sur-ensemble d'un ensemble qualifié est qualifié si l'on superpose les mêmes éléments que ceux de l'ensemble qualifié. Cela n'exclut pas, cependant, la possibilité d'utiliser tous les transparents.

Définition 3.2

Soit (Γ_{qual}, Γ_{qual}) une structure d'accès pour un ensemble de n participants.

Deux collections de matrices booléennes de dimensions $n \times m$, C_0 et C_1 constituent un schéma de cryptage visuel (Γ_{qual}, Γ_{qual}, m) VCS s'il existe une valeur $\alpha(m)$ et un ensemble $\{(X, d_x)\}_{X \in \Gamma_{equal}}$ satisfaisant :

1.Chaque ensemble qualifié $X=\{i_1, i_2, \ldots, i_p\} \in \Gamma_{equal}$ peut recouvrir l'image partagée, par la superposition des transparents. Formellement :

$\forall M \in C_0$, le Vecteur V résultat du « ou » des lignes $i_1, i_2, \ldots, i_p$ satisfait

$H(V) \leq d_x - \alpha(m) \times m$;

alors que

$\forall M \in C_1$, le Vecteur V résultat du « ou » des lignes $i_1, i_2, \ldots, i_p$ satisfait

$H(V) \geq d_x$;

2.Chaque ensemble interdit $X=\{i_1, i_2, \ldots, i_p\} \in \Gamma_{forb}$ n'a aucune information sur l'image partagée. Plus Formellement :

Les deux collections de $p \times m$ matrices D_t, $t=0,1$, obtenues par la restriction de chacune des matrices $n \times m$ de C_t, $t=0,1$, aux lignes $i_1, i_2, \ldots, i_p$ sont impossibles à distinguer les unes des autres, dans le sens ou elles contiennent les mêmes matrices avec les mêmes fréquences.

On a ainsi énoncé la généralisation des conditions de contraste et de sécurité.

3.4 Synthèse

Dans ce chapitre on s'est intéressé à la définition des propriétés générales d'un Schéma de Cryptage visuel et à la présentation de ses paramètres caractéristiques.

En effet, un schéma de cryptage visuel est caractérisé par plusieurs différents paramètres, et construire un schéma fonctionnel relève le challenge d'établir un équilibre entre ces paramètres coexistants et parfois antagonistes.

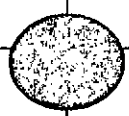
Plusieurs Schémas de cryptage visuel sont possibles parmi lesquels :

- k parmi k
- k parmi n
- Structure générale.

En conséquence, pour chaque schéma on propose plusieurs techniques de construction des matrices puis des parts. L'optimisation du schéma VCS, qui se base sur la modélisation mathématique et le développement algébrique, tout en adoptant des techniques de calcul numériques fera l'objet du chapitre suivant.

CHAPITRE 4

Techniques de construction des schémas de cryptage visuel



4.1 Introduction

Le cryptage visuel est organisé à travers des schémas choisis selon le domaine d'application, et la structure participante, notamment les structures qualifiées et interdites, les priorités et même les exceptions envisagées.

Assurer la sécurité et la qualité de l'information reçue (la résolution et le contraste) répond à des limitations sur les paramètres caractéristiques de l'image, mais aussi à des conditions sur les structures algébriques et les constructions matricielles.

Chaque schéma comprend des techniques de construction différentes et supporte des conditions et outils spécifiques.

Dans cette section, il sera question de présenter, pour chacun des schémas de cryptage visuel, une sélection des techniques de constructions des collections de matrices de bases qui les définiront selon ce qui a été énoncé dans le chapitre précédent.

Les schémas de cryptage visuel seront traités selon qu'ils soient à :

- k parmi k participants,
- k parmi n participants,
- Structure d'accès générale de participants.

La description de ces schémas sera introduite à travers une série de cas particuliers des deux premiers puis l'on progressera, graduellement, vers des cas de plus en plus généraux dans l'ordre des schémas cités ci dessus.

4.2 Cas de k et n petits

4.2.1 k=2

a) Le schéma 2 parmi 2

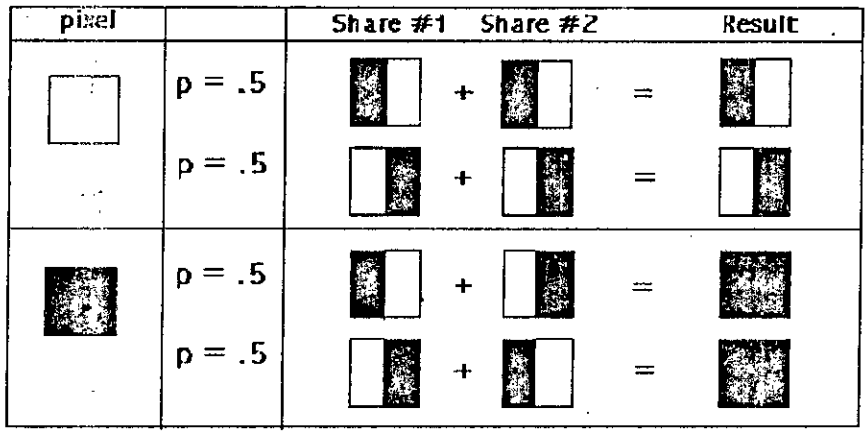


Figure 4.1 : Schéma de base du schéma de cryptage visuel 2 parmi 2

Dans un schéma 2 parmi 2, il est possible d'adopter le choix de subpixel comme indiqué dans la figure 4.1. Ce choix comporte un inconvénient relatif à la distorsion dans les proportions de l'image comme il a été relevé dans le chapitre précédent.

Pour résoudre cette problématique, on propose des combinaisons de 4 subpixels disposés dans un carré en structure de deux lignes × deux colonnes, selon un arrangement de 2 subpixels noirs et deux subpixels blancs. (voir figure 4.2)

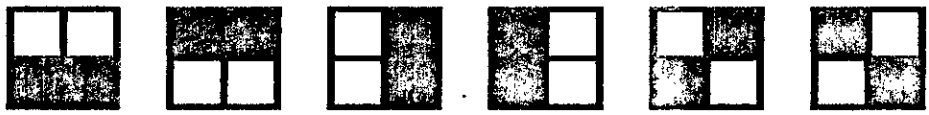


Figure 4.2 : Structure des Pixels des parts du schéma 2 parmi 2

Il en résulte les collections de matrices de base suivantes :

$$C_0=\{\begin{bmatrix}0&1&0&1\\0&1&0&1\end{bmatrix}\} \bmod 4\ !,$$

$$C1=\{\begin{bmatrix}0&1&0&1\\1&0&1&0\end{bmatrix}\} \bmod 4\ !.$$

Pour encoder un pixel blanc (respectivement noir), on choisit une matrice de C_0 (respectivmentC_1).

Chaque ligne représente une description de la série de subpixels d'un pixel dans un seul transparent ou une seule part.

Lorsque l'on superpose les deux transparents pour reconstruire un pixel blanc à l'origine, on obtient un gris médium composé de deux subpixels blancs et deux subpixels noirs.

Dans le cas du noir, on obtient un recouvrement total de subpixels en noir.

Le contraste dans ce cas est de $\alpha=1/2$.

b) Schéma 2 parmi n

- 1ère construction

On fixe dans cette construction les matrices selon ce qui suit :

$$C_0=\{\begin{bmatrix} 1 & 0 & . & . & . & 0 \\ 1 & 0 & . & . & . & 0 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ 1 & 0 & . & . & . & 0 \end{bmatrix} \} \text{ mod n !}$$

$$C1=\{\begin{bmatrix} 1 & 0 & . & . & . & 0 \\ 0 & 1 & . & . & . & 0 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ 0 & 0 & . & . & . & 1 \end{bmatrix} \} \text{ mod n !}$$

Un pixel blanc reconstruit a un poids de Hamming de 1.

Un pixel noir reconstruit a un poids de Hamming de 2.

Plus on superpose de transparents ou de parts plus le contraste augmente.

Autrement que dans le cas n petit, la perte en résolution sera trop importante (expansion du pixel égale à m).

- 2éme construction

On fixe m tel que $C_m^{m/2} \geq n$.

Considérer tous les sous-ensembles de cardinal m/2 d'un ensemble principal de cardinal m.

La $i^{éme}$ ligne de S^1 correspond au $i^{ème}$ sous-ensemble tel que :

$S^1[i,j]=1$ si ! le $j^{éme}$ élément de l'ensemble principal est un élément du $i^{ème}$ sous-ensemble .

S^0 est une matrice de dimension n×m dont toutes les lignes sont sous forme de : $1^{m/2}0^{m/2}$.

C_0 et C_1 sont obtenues à partir des permutations des colonnes des matrices de base que l'on a définies.

Le contraste minimum est de 1/m et dépend de l'expansion du pixel qui est limitée par la condition $C_{m/2}^{m} \geq n$.

4.2.2 k=3

a) Le schéma 3 parmi 3

$$C_0=\{\begin{bmatrix}0&0&1&1\\0&1&0&1\\0&1&1&0\end{bmatrix}\} \bmod 4\,!,\quad C_1=\{\begin{bmatrix}1&1&0&0\\1&0&1&0\\1&0&0&1\end{bmatrix}\} \bmod 4\,!$$

Les six parties décrites par les matrices sont exactement celles utilisées dans le schéma 2 parmi 2.(Figure 4.2)

Chaque paire de transparents, selon qu'elle appartienne à C_0 ou C_1, contient 3 subpixels noirs et un subpixel blanc. Par conséquent, l'analyse d'une part ou une paire de transparents rend impossible la distinction entre C_0 et C_1.

Cependant, la superposition de 3 transparents de C_0 permet d'obtenir 3 subpixels noirs et un subpixel blanc alors que la superposition de 3 transparents de C_1 aboutit à un recouvrement total des subpixels en noir.

b) Généralisation 3 parmi n

B est une matrice de dimension n × (n-2) dont tous les éléments sont égaux à 1.

I est la matrice identité de dimension n × n.

BI est la concaténation de B et I de dimension n × (2n-2)

C(BI) est la matrice complémentaire de BI.

$C_0 = \{C(BI)\}$, $C_1 = \{BI\}$.

Si les parts sont choisies de C0 ou C1, chaque part ou transparent singulier contiendra

(n-1) subpixels noirs et (n-1) subpixels blancs et chaque paire de transparents superposés comportera (n-1) subpixels noirs.

Un triplet de transparents superposés contiendra n subpixels noirs si les parts sont choisies de C_0 et (n+1) subpixels noirs dans le cas de C_1.

Le contraste dans ce schéma est évalué à 1/(2n-2).

4.2.3 k=4

Le schéma 4 parmi 4 peut être résolu pour un pixel par la configuration des parts de pixel schématisée dans la figure 4.3.

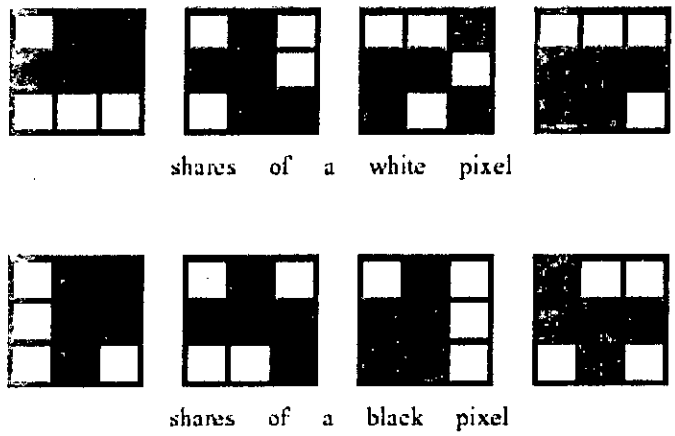


Figure 4.3 : Structure des pixels des parts du schéma 4 parmi 4

Chaque part singulière contient 5 subpixels noirs qu'elle soit choisie parmi la collection C_0 ou C_1. De la même façon, chaque paire de parts superposées contient 7 subpixels noirs et chaque triplet contient 8 subpixels noirs.

Par contre, de la superposition de 4 transparents ou parts résultera 8 subpixels noirs s'ils sont choisis à partir de la matrice de la collection C_0, et 9 subpixels noirs si la sélection est faite à partir de la collection C_1.

Il est possible de réduire le nombre de subpixels total de 9 à 8 subpixels mais le problème de distorsion dans les proportions de l'image persiste.

4.3 Cas général k parmi k

Dans ce qui va suivre, il sera question de la description de deux constructions qui répondent au schéma k parmi k de partage de secret, en utilisant une expansion de pixel de 2^k puis de 2^{k-1} respectivement.

On essayera de prouver que la seconde construction est plus optimale.

Ensuite, il s'agira de décrire l'algorithme d'une troisième technique, inspirée de la deuxième construction, plus apte à l'implémentation et plus proche de l'application.

C'est la raison pour laquelle nous avons choisi d'utiliser cette dernière technique dans notre implémentation du logiciel.

4.3.1 Construction 1

- Pour définir les 2 collections de matrices C_0 et C_1 on utilise 2 listes de vecteurs : $J_1^0, J_2^0, \ldots, J_k^0$ et $J_1^1, J_2^1, \ldots, J_k^1$.

$J_1^0, J_2^0, \ldots, J_k^0$ sont des vecteurs à K dimensions sur le champ de Galois GF[2] à 2 éléments {0,1}. Chaque (k-1) de ces vecteurs est indépendant sur GF[2] mais non pas les k.

On propose alors la construction suivante :

- $J_i^0 = 0^{i-1}10^{k-i}$ si $1 \le i < k$
- $J_k^0 = 1^{k-1}0$

$J_1^1, J_2^1, \ldots, J_k^1$ sont des vecteurs à K dimensions sur le champ de Galois GF[2] à 2 éléments {0,1}, linéairement indépendants.

Chacune des ces listes est relative à l'une des matrices de base S^t, $t \in \{0,1\}$, du schéma k parmi k, et de dimension $k \times 2^k$.

Rappel

GF[n] : *«Galois Field » est une notation anglo-saxonne qui indique un champ de Galois fini à n éléments. C' est un corps fini à n éléments noté aussi parfois* **Fn**. *On relève le fait qu'il existe des corps finis* **Fp** *pour tous les nombres premiers p, ce sont les corps (**Zp**, +, ×), et qu'il existe d'autres corps à $q = p^n$ éléments pour toutes les puissances de nombres premiers ($q = p^n$).Il faut aussi savoir que tous les corps finis sont commutatifs (théorème de Wedderburn).*

- Les colonnes des matrices S^t sont indexées par des vecteurs de dimension k sur GF[2] ; il existe bien 2^k vecteurs différents sur l'espace vectoriel des vecteurs à k dimension sur GF[2].

  Pour $t \in \{0,1\}$, S^t est définie comme suit :

$$\forall\, i : 1 \le i \le k \text{ et } \forall x \text{ de dimension k sur GF[2]} :$$
$$S^t[\, i\, , x\,] = < J_i^t , x >$$

  $< x, y >$ indique le produit scalaire sur GF[2].

Lemme 4.1

Le schéma ci-dessus est un schéma k parmi k avec les paramètres suivants :

- *L'expansion du pixel $m = 2^k$,*
- *La différence relative où le contraste $\alpha = 1/2^k$,*
- Le cardinal de chaque collection $r = |C_0| = |C_1| = 2^k$!.

4.3.2 Construction2

Soit un ensemble de cardinal k, $W=\{e_1,e_2,\ldots,e_k\}$.

Soit $\pi_1, \pi_2,\ldots, \pi_{2^k-1}$ la liste de tous les sous-ensembles de W de cardinal pair.

Soit $\sigma_1,\sigma_2,\ldots,\sigma_{2^k-1}$ la liste de tous les sous-ensembles de W de cardinal impair.

Chacune de ces listes définit l'une des matrices de base S^0 et S^1 du schéma k parmi k et de dimension $k\times 2^{k-1}$, et ce de la manière suivante :

$\forall$ i t.q : $1\leq i \leq k$, $\forall$j t.q : $1\leq j \leq 2^{k-1}$:

- $S^0[i,j]=1$ si ! $e_i \in \pi_j$
- $S^1[i,j]=1$ si ! $e_i \in \sigma_j$

Les collections C_0 et C_1 résultent des permutations des colonnes des matrices de base correspondantes à S^0 et S^1.

Lemme 4.2

Le schéma ci-dessus est un schéma k parmi k avec les paramètres suivants :

- *L'expansion du pixel* $m=2^{k-1}$.
- *La différence relative ou le contraste* $\alpha=1/2^{k-1}$.
- *Le cardinal de chaque collection* $r=/C_0/=/C_1/=2^{k-1}!$.

On constate que le contraste est plus élevé dans cette deuxième construction et que l'expansion du pixel est plus petite. On en déduit une diminution des pertes en résolution et en contraste et une amélioration de la qualité de l'image reconstruite. La deuxième construction, plus optimale, est préférable à la première. C'est pourquoi nous allons nous nous en inspirer pour la description de la construction suivante qui constitue une technique plus proche de l'application, et plus faisable. C'est l'algorithme de cette dernière méthode qui sera à la base de notre implémentation.

Le théorème suivant énonce que les résultats de la seconde construction sont les plus optimaux de tous les schémas k parmi k.

Théorème 4.1

Quel que soit le schéma k parmi k : $\alpha\leq 1/2^{k-1}$ *et* $m\geq 2^{k-1}$.

4.3.3 Construction 3

Cette technique générale de construction de schéma k parmi k est la plus utilisée pour sa simplicité et sa facilité d'implémentation comme il a été expliqué précédemment. Elle a pour objet de construire les deux matrices de base à dimension

$k \times 2^{k-1}$ pour un schéma k parmi k.

Première étape :

Produire les 2^k séquences binaires possibles sur k bits.

Deuxième étape :

Choisir les 2^{k-1} séquences à nombre pair de 1.
Chaque séquence représente une colonne S^0, lister les colonnes.

Troisième étape :

Choisir les 2^{k-1} séquences à nombre impair de 1
Chaque séquence représente une colonne S^1, lister les colonnes.

Quatrième étape :

La collection C_0 (resp.C_1) est composée des matrices résultantes de la permutation des colonnes de S^0 (resp. S^1). Chaque matrice correspond à une permutation, le cardinal de chaque collection sera le nombre de permutations. 2^{k-1} !.

Cinquième étape :

Pour encoder un pixel blanc (resp.noir) sélectionner aléatoirement une matrice S^0 (resp. S^1) de C_0 (resp.C_1).
Faire correspondre chaque ligne décrivant la disposition des subpixels à un transparent.
Répéter l'opération autant de fois qu'il y a de pixels existant dans l'image.

Dans les deux premières constructions du schéma k parmi k, une relation directe est établie entre le type du schéma caractérisé par le nombre k de transparents ou participants, et l'expansion du pixel m.

En conclusion, on peut dire que l'expansion du pixel dépend du nombre de parts partagées ; il en est de même du contraste. Cependant, la variation étant exponentielle

l'expansion du pixel m augmente rapidement tandis que le contraste diminue, induisant une dégradation rapide de l'image pour les schémas à de nombreux participants coïncidant avec un k élevé.

4.4 Cas général k parmi n

4.4.1 Constructions de Shamir et Naor

Considérons C un schéma k parmi k de cryptage visuel aux paramètres α, m et r.

C se base sur deux collections de matrices booléennes de dimension k × m :

$C_0=\{T_1^0,T_2^0,\ldots,T_r^0\}$, $C_1=\{T_1^1, T_2^1,\ldots,T_r^1\}$.

En outre, on suppose que le schéma est uniforme (cf. chapitre 3, section 3.2).

Soit H une collection de ℓ fonctions tel que :

1. $\forall$ h$\in$H on a h : $\{1\ldots n\}\rightarrow\{1\ldots k\}$

2. $\forall$ les sous-ensembles B$\subset$ {1..n} de cardinal k.

   $\forall$ q , $1\leq q\leq k$ la probabilité qu'une fonction choisie aléatoirement h$\in$H donne q différentes valeurs sur B est la même, elle est notée β_q

A partir du schéma C et l'ensemble H on construit un schéma C' à k parmi n selon la description suivante :

- Soit l'ensemble principal V =U × H (de cardinal m×ℓ où chaque élément est indexé par un membre de U et un membre de H).
- Chaque t, tel que $1\leq t\leq r^{\ell}$, est indexé par un vecteur $(t_1,t_2,\ldots,t_\ell)$ où chaque t_i satisfait : $1\leq t_i\leq r$.
- La matrice S_t^b pour $t=(t_1,t_2,\ldots,t_\ell)$ et b$\in\{0,1\}$ est définie par :

$$S_t^b[i,(j,h)] = T_{th}^b[h(i),j].$$

Le t_h signifie le $h^{ème}$ élément de t ; $1\leq h\leq\ell$ et $1\leq t_h\leq r$. Ailleurs h est une fonction élément de H.

H peut être construite à partir d'une collection de ℓ fonctions de hachages indépendantes les unes des autres.

Il est à noter que si l'on répétait la construction ℓ fois pour chaque fonction de H à partir d'un même schéma C on obtiendrait des schémas complètement indépendants.

Lemme 4.3:

Si C est un schéma aux paramètres m, α, r alors C' est un schéma aux paramètres m' = $m.\ell$; $\alpha'=\alpha.\beta_k$; $r'=r^{\ell}$.

4.4.2 Construction de Verheul et Van Tilborg

Soit $V_m(q)$ un champ vectoriel sur l'espace de Galois GF[q] à q éléments à m dimension.

Définition 4.1 :

Pour q choisi, fixé et $m \geq 1$, un n-arc est défini en tant que sous-ensemble A de $V_m(q)$ de cardinal n, $/A/ = n$.

Ce sous-ensemble a pour propriété que chaque m éléments dans A sont linéairement indépendants.

Le maximum n pour lequel un n-arc sur Vm(q) existe et est noté r(q,m).

D'autres notations seront utilisés dans les prochaines constructions, ce sont :

h : le nombre de subpixels blancs dans un pixel reconstruit d'origine blanche.

ℓ : nombre de subpixels blancs dans un pixel reconstruit d'origine noire.

q : le nombre d'éléments de l'espace de Galois qui est égal à deux dans le cas de l'ensemble binaire {0,1}.

b : l'expansion du pixel vu que m est la dimension de l'espace vectoriel.

a) Construction 1

Pour construire un schéma k parmi n de partage de secret visuel VCS, on utilisera des fonctions linéairement indépendantes définies sur $V_m(q)$.

Pour une base quelconque de $V_m(q)$ il y a une correspondance Injective entre les fonctions $F(\underline{x})$ sur $V_m(q)$ et les vecteurs $\underline{f}$ de $V_m(q)$ donnée par :

$$F(\underline{x})=<\underline{f}, \underline{x}>= f_1x_1+f_2x_2+\ldots+f_mx_m$$

Les fonctions $F_i(\underline{x})$ pour $1 \leq i \leq \ell$ sont linéairement indépendantes si et seulement si les vecteurs correspondants $\underline{f_i}$ $1 \leq i \leq \ell$ sont linéairement indépendants.

Considérant une suite de vecteurs dans $V_m(q)$ $\underline{u_1},\underline{u_2},\ldots,\underline{u_{(q^m)}}$; la représentation matricielle S de dimension $n \times q^m$ des n fonctions $Fi(\underline{x})$ pour $1 \leq i \leq n$ respectivement aux vecteurs u_j pour $1 \leq j \leq q^m$ est définie par :

$$S_{ij} = F_i(\underline{u_j}) \text{ pour } 1 \leq i \leq n \text{ et } 1 \leq j \leq q^m.$$

Un schéma de partage visuel k parmi n aux paramètres $b=(q^k-1)/(q-1)$; $h=1$; $\ell=0$; $r=|C_0|=|C_1|=q^k$! peut être obtenu à travers les étapes suivantes :

1. Choisir un espace fini à q éléments, avec $r(q,k-1) \geq n$ et $r(q,k) \geq n$.

Eliminer $\underline{0}$ de $V_m(q)$ et supprimer tous les multiples.

Construire $PG_{m-1}(q)$ un espace projeté de dimension (m-1) sur GF[q].Il contient $(q^m-1)/(q-1)$ vecteurs.

Considérons un nombre de vecteurs dans $V_m(q)$, $u_1,u_2,\ldots,u_{(q^m)}$ et soit $v_1,v_2,\ldots,v_{(q^m-1)/(q-1)}$ l'ensemble réduit des vecteurs générants $PG_{m-1}(q)$

2. Choisir une série de fonctions $F_1, F_2,\ldots,F_n$ sur $V_k(q)$, tel que n'importe quel k fonctions d'entre elles sont linéairement indépendantes. Cela est possible grâce à la définition de r(q, k).

Soit S la représentation matricielle des fonctions F_i, $1 \leq i \leq n$, respectivement à la série de vecteurs u_j, $1 \leq j \leq q^k$.

Soit S' la restriction de S aux colonnes indexées par v_j, $1 \leq j \leq (q^m-1)/(q-1)$.

Remplacer tous les éléments non nuls par 1.

La matrice S' de dimension $n \times (q^m-1)/(q-1)$ est une matrice binaire qui génère la classe C_1.

3. Choisir n fonctions G_1', G_2',... G_n' définies sur $V_{k-1}(q)$ tel que (k-1) d'entre elles sont indépendantes.

Etendre leur définition dans un sens canonique pour obtenir $G_1,G_2,\ldots,G_n$ sur $V_k(q)$ tel que $G_i(x_1,x_2,\ldots,x_{k-1},x_k)=G_i'(x_1,x_2,\ldots\ldots x_{k-1})$

Chaque (k-1) parmi ces fonctions sont linéairement indépendantes mais chaque k parmi elles sont dépendantes.

La dimension de $V_{k-1}(q)$ est k-1.

Soit T la représentation des G_i, et T' la restriction de T selon les colonnes indéxées par les vecteurs v_j en remplaçant les éléments non nuls par des 1.

T' constitue la matrice binaire de dimension $(q^m-1)/(q-1)$ générant la classe C_0.

En général on choisit m = k, m étant dans ce cas la dimension de l'espace vectoriel. Ce qui assure l'inexistence de colonnes nulles dans la première représentation pour C_1.

Pour construire un schéma k parmi k, il suffit de prendre q=2. Ainsi l'expansion du pixel sera modestement améliorée par rapport à la première construction k parmi k, elle sera de (2^k-1) au lieu de 2^k.

Si (n-1) est premier, on choisit q =n-1 ; l'expansion du pixel sera $((n-1)^k-1)/(n-2)$.

Le contraste dans ce modèle sera $(q-1)/(q^k-1)$, il est plus élevé que dans la méthode de Shamir-Naor où il est estimé à $\beta_k/2^{k-1}$ où $\beta_k \leq 1$.

On constate que l'expansion du pixel ainsi que le contraste dépendent de :

- La structure algébrique représentée par le choix du nombre premier d'éléments **q** du corps de Galois.
- La dimension de l'espace vectoriel choisi **m** et que l'on pose souvent égale au seuil de participant **k** pour travailler avec des espaces vectoriels de même dimension. Ce qui revient à dire, comme dans le schéma précédent, que l'expansion du pixel et le contraste sont dépendants du seuil de participants qualifiés k.

b) <u>Construction 2</u>

Comme dans le schéma précédent, commençons par quelques considérations générales :

Soit $G(\underline{x})$ et $F_1(\underline{x})$, $F_2(\underline{x})$,...., $F_n(\underline{x})$ des fonctions définies sur $V_m(q)$.

Soit $\underline{u}_j$, $1 \leq j \leq q^{m-1}$ une série de vecteurs $\underline{x}$ dans $V_m(q)$ pour lesquels $G(\underline{x})=0$: ce sont des éléments du Ker de G.

Soit $\underline{v}_j$, $1 \leq j \leq q^{m-1}$ une série de vecteurs $\underline{x}$ dans $V_m(q)$ pour lesquels $G(\underline{x})=1$.

Ces séries de vecteurs définissent des représentations de matrices tel que :

$$S_{ij} = F_i(\underline{u}_j) \text{ pour } 1 \leq i \leq n \text{ et } 1 \leq j \leq q^{m-1}$$
$$T_{ij} = F_i(\underline{v}_j) \text{ pour } 1 \leq i \leq n \text{ et } 1 \leq j \leq q^{m-1}.$$

Un schéma de partage visuel k parmi n robuste avec les paramètres $b=q^{k-1}$, $h=1$, $\ell=0$, $|C_0| = |C_1| = q^{k-1}$! peut être obtenu après la réalisation des étapes suivantes :

Choisir un ensemble fini de taille q tel que $r(q,k) \leq n+1$

1. Choisir (n+1) fonctions sur $V_k(q)$, appelées $F_1, F_2, \ldots, F_n$ et G tel que chaque k fonctions d'entre elles, sont linéairement indépendantes.
2. Soit S la représentation matricielle des fonctions F_i pour $1 \leq i \leq n$ respectivement à la liste des vecteurs $\underline{u}_j$ pour $1 \leq j \leq q^{k-1}$ de $G^{-1}(0)$.

   Remplacer les éléments non nuls dans la matrice par des 1

   La matrice de dimension $n \times q^{k-1}$ générée par S forme la classe C_1.
3. Soit T, la représentation matricielle des fonctions F_i pour $1 \leq i \leq n$ respectivement à la liste des vecteurs $\underline{v}_j$ pour $1 \leq j \leq q^{k-1}$ de $G^{-1}(1)$.

   Remplacer les éléments non nuls dans la matrice par des 1

   La matrice de dimension $n \times q^{k-1}$ générée par T forme la classe C_0.

Lorsque k = n et q=2 ce schéma revient à la construction 2 du schéma k parmi k .

Si n est premier on pourra poser q=n, l'expansion du pixel sera n^{k-1} et le contraste de $1/n^{k-1}$.

Cette deuxième construction de Verheul et Van Tilborg offre un schéma k parmi n où l'expansion du pixel est souvent inférieure à celle de la première construction. Cependant, il existe des situations où le contraire est vrai comme le cas où (n-1) est un nombre premier ou une puissance d'un premier. L'expansion du pixel de la première construction sera $((n-1)^{k-1})/(n-2)$ quant à la seconde construction elle sera de n^{k-1}, ce qui est supérieur à la précédente.

Dans ce schéma, comme dans le précédent, l'expansion du pixel ainsi que le contraste dépendent de :

- La structure algébrique représentée par le choix du nombre premier d'éléments **q** du corps fini de l'espace de Galois.
- Le seuil de participants qualifiés k sans l'introduire d'une grandeur intermédiaire m comme dans le schéma précédent

Cette construction a le mérite d'être plus simple.

4.4.3 Construction 3 : Construction basée sur les ordres de multiplicités

a) Equations de construction dans le cas général

Soit $f_{c,i}$ la multiplicité de la colonne c dans S^i.

Définition 4.2 :

Soit (S_0,S_1) les matrices de base d'un schéma de cryptage visuel k parmi n. Elles sont dans une forme canonique si pour i=0,1 ; les deux propriétés suivantes sont satisfaites :

1. *Pour chaque colonne c et c' tel que H(c)=H(c') il en résulte fc,i =fc',i*
2. *Pour chaque colonne c il résulte :*

$$f_{c,i} = \begin{cases} f_{\bar{c},i} & \text{si k est pair} \\ f_{\bar{c},1-i} & \text{si k est impair} \end{cases}$$

On appelle un tel schéma un schéma de cryptage visuel à seuil canonique k parmi n.

Dans un schéma canonique (k,n) de cryptage visuel à seuil, d'après la première propriété de la définition, toutes les colonnes ayant un poids de Hamming égal apparaissent avec la même multiplicité.

On définit alors la multiplicité d'une colonne de poids j dans S^i noté $h_{j,i}$ tel que $h_{j,i} = f_{c,i}$ si H(c)= j.

Par conséquent, chaque schéma canonique de cryptage visuel (k,n) peut être simplement décrit par la paire de vecteurs : $(h_{0,0}\,;\, h_{1,0}\,;\ldots\,;\, h_{n,0})$ et $(h_{0,1}\,;\, h_{1,1}\,;\ldots\,;h_{n,1})$.

L'expansion du pixel m d'un schéma VCS canonique (k,n) est égal à :

$$m = \sum_{j=0}^{n} h_{j,0} C_n^j = \sum_{j=0}^{n} h_{j,1} C_n^j$$

Lemme 4.3

$S(h_0)$ et $S(h_1)$ sont les matrices de base d'un schéma (k,n) VCS avec une expansion de pixel m et un contraste α si et seulement si les propriétés suivantes sont satisfaites :

1. $\sum_{j=0}^{n} h_{j,0} C_n^j = \sum_{j=0}^{n} h_{j,1} C_n^j = m$
2. Pour $1 \le \ell \le k\text{-}1$ $\sum_{j=0}^{n-\ell} h_{j,0} C_{n-\ell}^j = \sum_{j=0}^{n-l} h_{j,1} C_{n-\ell}^j$.
3. $\sum_{j=0}^{n-k} C_{n-k}^j (h_{j,0} - h_{j,1}) = \alpha.m$

Ainsi, on observe qu'un nombre de conditions sont redondantes. C'est pourquoi, l'écriture des conditions du lemme précédent a été simplifiée par la répétition de l'application de l'identité de Pascal pour les coefficients binomiaux.

On pose m=1 sans perte de généralité ou alors on considère que les coefficients ont été divisés par m pour obtenir :

- $\alpha = \sum_{j=0}^{n-k} C_{n-k}^j (h_{j,0} - h_{j,1})$
- $\sum_{j=0}^{n} h_{j,0} C_n^j = 1$
- $\sum_{j=0}^{n} h_{j,1} C_n^j = 1$
- Pour $1 \le \ell \le$ k-1 : $\sum_{j=0}^{n-\ell} C_{n-\ell}^j (h_{j,0} - h_{j,1}) = 0$.
- Pour $0 \le$ j $\le$ n : $h_{j,0} \ge 0$ *et* $h_{j,1} \ge 0$.

Corollaire 4.1

Soit Σ un schéma de cryptage visuel canonique (k,n) :

- *Si k est impair pour j=0,...,n il en résulte que* $h_{j,0}=h_{n-j,1}$
- *Si k est pair pour j=0,...,n il en résulte que* $h_{j,0}=h_{n-j,0}$ *et que* $h_{j,1}=h_{n-j,1}$

On obtient alors pour k impair :

- $\alpha = \sum_{j=0}^{n-k} (h_{j,0} - h_{n-j,0}) C_{n-k}^j$
- $\sum_{j=0}^{n} h_{j,0} C_n^j = 1$
- $\sum_{j=0}^{n-l} (h_{j,0} - h_{n-j,0}) C_{n-l}^j$ pour ℓ=1,...,k-1
- hj,0 ≥0 pour j=0,...,n

On obtient, grâce à ce schéma, des équations à (n+1) inconnues avec k+1 équations et n contraintes.

Le nombre d'inconnus est réduit grâce au corollaire 4.1.

b) Schéma (k,n) avec contraste optimal

Dans un schéma de cryptage visuel (k,n) à contraste optimal, on suppose que les deux matrices de base n'ont pas de colonnes en commun. Ainsi, on obtient un meilleur contraste qui donne le résultat suivant :

Résultat

Dans un schéma de cryptage visuel(k,n) à contraste optimal, construit à partir des matrices de base canoniques, on a :

1. *Si $h_{j,1-i} > 0$ alors $h_{j,i}=0$*
2. *Si k est pair $h_{j,i} = h_{n-j,i}$*
3. *Si k est impair $h_{j,i}=h_{n-j,1-i}$*

D'où si n est pair et k impair $h_{n/2,0}=h_{n/2,1}=0$.

c) Cas particulier (n-1,n) à contraste optimal

Soit le schéma de cryptage visuel (n-1,n) à contraste optimal, construit à partir de matrices de base canoniques.

A partir des résultats du lemme 4.4 on obtient une construction précise du schéma requis (voir lemme 4.5.)

Lemme 4.4

Soit $n \geq 3$, dans n'importe quel schéma de cryptage visuel (n-1) parmi n à contraste optimal construit à partir de matrices de base canonique on a :

1. *$h_{j,0} > 0$, si $!\, j<n/2$ et j pair ou $j>n/2$ et j impair.*
2. *$h_{j,1} > 0$, si $!\, j<n/2$ et j impair ou $j>n/2$ et j pair.*

Lemme 4.5

Soit $n \geq 3$, dans n'importe quel schéma de cryptage visuel (n-1) parmi n à contraste optimal construit à partir des matrices de base canoniques, les $h_{i,j}$ satisfont :

- *Si n est pair*

  Pour j=0,...,⌊ (n-2)/4 ⌋on a $h_{2j,0}=h_{n-2j,1}=n/2-2j$;

  Pour j=0,...,⌊ (n-4)/4 ⌋on a $h_{2j+1,1}=h_{n-(2j+1),0}=n/2-(2j+1)$

- *Si n est impair*

  Pour j=0,...,⌊ n/4 ⌋ on a $h_{2j,0}=h_{n-2j,0}=n-4j$

  Pour j=0,...,⌊ (n-5)/4 ⌋on a $h_{2j+1,1}=h_{n-(2j+1),1}=n-(4j+2)$.

En résumé, on obtient une construction précise dépendant de n qu'il est possible de présenter aussi sous la forme suivante :

- n pair et $0 \leq j \leq n$:

$$h_{j,0} = h_{n-j,1} = \begin{cases} n/2 - j & \text{si j pair et } j < n/2 \\ j - n/2 & \text{si j impair et } j > n/2 \\ 0 & \text{ailleurs} \end{cases}$$

- n impair et $j=0,\ldots,\lfloor n/2 \rfloor$:

$$h_{j,0}=h_{n-j,0}= \begin{cases} n-2j & \text{si j pair et } j < n/2 \\ 0 & \text{ailleurs} \end{cases}$$

$$h_{j,1}=h_{n-j,1}= \begin{cases} n-2j & \text{si j impair et } j < n/2 \\ 0 & \text{ailleurs} \end{cases}$$

On en déduit l'expansion de pixel après l'application de la relation

$$m = \sum_{j=0}^{n} h_{j,0} C_n^j = \sum_{j=0}^{n} h_{j,1} C_n^j$$

pour enfin obtenir :

Pour $n \geq 3$ dans un schéma de cryptage visuel à seuil de (n-1) parmi n, à contraste optimal et construit à partir de matrices de base canoniques, l'expansion de pixel est donnée par :

$$m = \begin{cases} \frac{n}{4} C_n^{n/2} & \text{si } n \text{ pair} \\ n C_{(n-1)/2}^{n-1} & \text{si } n \text{ impair} \end{cases}$$

Théorème 4.2

Quel que soit n ≥ 3 et quel que soit-le schéma de cryptographie visuelle canonique (n-1) parmi n à contraste optimal, le contraste est donné par :

$$\alpha_{max} = \begin{cases} [\frac{n}{4} C_n^{n/2}]^{-1} & \text{si } n \text{ pair} \\ [nC_{(n-1)/2}^{n-1}]^{-1} & \text{si } n \text{ impair} \end{cases}$$

Dans cette section, nous avons, d'abord, présenté la définition d'un schéma canonique ; ensuite, celle d'un schéma à contraste optimal ; enfin nous avons exposé une construction précise d'un schéma à (n-1) parmi n .

Pour le cas général k parmi n, un système d'équations linéaires, relatif aux poids des colonnes de chacune des matrices de base et à leurs ordres de multiplicités, présente une caractéristique imprécise du système car le nombre d'équations est inférieur au nombre d'inconnues.

4.4.4 Construction 4 : Algorithme à base d'une fonction de hachage parfaite

Cette méthode se base sur les fonctions de hachage parfaites utilisées pour développer une matrice de départ PHF destinées à générer les collections de matrices de base qui caractériseront notre schéma k parmi n.

Rappel : Fonctions de hachage à sens unique

Une fonction de hachage à sens unique peut avoir d'autres appellations : fonction de compression, fonction de contraction, digest, empreinte digitale, code correcteur cryptographique, code de vérification d'intégrité, code de détection de manipulation, code d'authentification, etc...

Une fonction à sens unique est une fonction relativement aisée à calculer mais l'inverse est considérablement plus difficile.

Une fonction de hachage est une fonction mathématique ou autre qui convertit une chaîne de caractères de taille arbitraire en une chaîne de caractères de taille fixe (souvent inférieure).

Une fonction de hachage à sens unique est à la fois une fonction de hachage et une fonction à sens unique.

Une fonction de hachage à sens unique H(M) opère sur un message M de longueur arbitraire et fournit en sortie une valeur de hachage de longueur fixe h, tel que : h= H(M) et h est de longueur fixe m.

Les caractéristiques traditionnelles d'une fonction de hachage à sens unique sont :

- Etant donné M, il est facile de calculer h.
- Etant donné h, il est difficile de calculer M.
- Etant donné M, il est difficile de trouver un autre message M' tel que : H(M)=H(M').

La notion de difficulté dépend du niveau de sécurité spécifique requis pour la situation ; mais la plupart des réalisations concrètes définissent cette difficulté par le recours à 2^{64} opérations et parfois même plus.

Définition

Une fonction de hachage parfaite aux paramètres n, m et w est un ensemble de fonctions de hachage F où:

$h : A \rightarrow B$, $\forall h \in F$ et pour $|A| = n$ et $|B| = m$.

Ayant la propriété que $\forall X \subset A$, *il existe au moins une fonction* $h \in F$ *tel que* $h||_X$ *est injective (one-to-one).*

On note alors par PHF(n ; N; m ; w) la matrice de dimension N × n de m symboles (entiers dans ce cas). F est l'ensemble des fonctions de hachage où $||F|| = N$.

La matrice PHF a pour propriété que quel que soit le sous-ensemble de w colonnes il existe au moins une ligne où les éléments indexés par cette ligne et chacune de ces w colonnes sont distincts les uns des autres.

Pour construire le schéma de cryptage visuel k parmi n, on se base sur les collections de matrices de base du schéma correspondant k parmi k.

Notre choix a porté sur cette construction dans l'implémentation du logiciel de cryptage visuel, dont voici les étapes essentielles de l'algorithme de cette construction :

Première étape

Calculer le cardinal de l'ensemble des fonctions de hachage parfaites correspondant au nombre de lignes de la matrice PHF.

$$N = \ell = C_{n-\lceil k/2 \rceil}^{\lfloor k/2 \rfloor}$$

Deuxième étape

Déterminer la PHF aux paramètres PHF(n ; ℓ ; k ; k), étant une matrice de dimension

$\ell \times n$.

Pour chaque ligne i de la PHF, on construit une fonction de hachage h_i relative à un sous-ensemble ordonné $x_i = \{X_1, X_2, \ldots, X_k\}$ de l'ensemble $\{1,2,\ldots,n\}$:

$$h_i(X) = \begin{cases} 1 & \text{si } X < X_2 \\ 2i-1 & \text{si } X = X_{2(i-1)} < X < X_{2i} \text{ pour } i=2,\ldots, k/2 \\ 2i & \text{si } X = X_{2i} \text{ pour } i=2,\ldots, k/2 \\ k & \text{si } X \geq X_k \end{cases}$$

De telle sorte que la ligne i soit : $h_i(1)\, h_i(2)\, h_i(3) \ldots\ldots h_i(n)$.

Si cette ligne existe déjà dans la matrice on passera à la suivante sans l'ajouter une deuxième fois.

Répéter l'opération pour les C_n^k sous-ensembles possibles jusqu'à l'obtention de ℓ lignes.

On obtient ainsi la matrice $\ell \times n$ tel que dans chaque k colonne il existe au moins une ligne où les éléments de ces colonnes sont distincts.

Troisième étape

Transposer la matrice PHF

Quatrième étape

Remplacer chaque élément α (entier) de la matrice PHF^T par la ligne α de la matrice de base S^0 (resp.S^1) du schéma k parmi k pour obtenir la matrice $\hat{S}^0$ (resp. $\hat{S}^1$) de la collection C_0 (resp.C_1) du schéma k parmi n.

Cinquième étape

Obtenir les collections C_0 et C_1 à partir des permutations des colonnes de $\hat{S}^0$ et $\hat{S}^1$.

Sixième étape

Pour encoder un pixel blanc (resp.noir) sélectionner aléatoirement une matrice S^0 (resp. S^1) de C_0 (resp.C_1).

Faire correspondre chaque ligne décrivant la disposition des subpixels à un transparent.

Répéter l'opération autant de fois qu'il y a de pixels existant dans l'image.

Telle est la description de la construction k parmi n à base de fonctions de hachage. L'expansion du pixel $k \times \ell$ étant importante on en déduit que ce schéma est caractérisé par une importante perte en résolution. Il est donc préférable de l'utiliser pour des schémas à petit seuil k.

Dans ce cas, l'expansion du pixel ne dépend pas seulement de k, qui est le seuil des participants autorisés mais aussi de n le nombre total de participants. On constate cela à partir du calcul de ℓ qui est fonction de n et k.

4.5 Construction d'un schéma de Cryptage Visuel à structure d'accès générale

Cette section sera consacrée à la présentation de deux techniques de construction de schémas de cryptage visuel pour une structure d'accès générale comme cela a été exposé dans le chapitre 3 partie 3.3.

Une structure d'accès générale est caractérisée par des ensembles qualifiés et des ensembles interdits

4.5.1 Construction 1

Soit $(\Gamma_{qual}, \Gamma_{forb})$ une structure d'accès robuste sur l'ensemble des participants $P=\{1,2,..,n\}$.

Z_M est la collection des ensembles interdits à cardinal maximal :

$$Z_M = \{B \in \Gamma_{forb} : B \cup \{i\} \in \Gamma_{qual} ; \forall\, i \in P/B\}$$

Une application cumulative (β,T) pour Γ_{qual} est un ensemble fini T, avec une application β, tel que :

$$\beta: P \rightarrow 2^T.$$

Pour $Q \subseteq P$ on a :

$$\bigcup_{a \in Q} \beta(a) = T \Leftrightarrow Q \in \Gamma_{Qual}.$$

On peut construire une application cumulative tel que (β,T), pour un Γ_{qual} quelconque en utilisant une collection d'ensembles interdits à cardinal maximal $Z_M = \{F_1, F_2, \ldots, F_t\}$.

Soit $T=\{T_1, T_2, \ldots, T_t\}$ et : $\forall i \in P$, $\beta(i) = \{T_j \mid i \notin F_j, 1 \leq j \leq t\}$

$$\forall x \in \Gamma_{qual} : \bigcup_{i \in x} \beta(i) = T.$$

Par contre, à n'importe quel ensemble $x \in \Gamma_{forb}$ il manquera au moins un élément T_j de T à recouvrir.

Une matrice cumulative (*cumulative array*) est une matrice booléenne de dimension $|P| \times |T|$ notée CA tel que :

- CA(i,j)=1 si l'élément $i \notin F_j$.

En d'autres termes, β(i) est l'ensemble qui désigne les colonnes non nulles de la ligne i. Le numéro de la colonne est indiqué par les indices des éléments T_j de l'ensemble β(i). Les éléments de T servent donc à indexer les colonnes.

Le cardinal de T est désigné par le cardinal de Z_M qui lui est égal.

Il est possible, à partir de la construction de la matrice cumulative CA, d'aboutir à un schéma de cryptage visuel pour n'importe quelle structure d'accès robuste et ce selon les étapes suivantes :

- Soit $\hat{S}^0$ et $\hat{S}^1$ les matrices de base qui constituent un schéma de cryptage visuel t parmi t, tel que $t = |Z_M|$.
- S^0 et S^1 sont les matrices de base d'une structure d'accès $(\Gamma_{qual}, \Gamma_{forb})$.
- ∀ i fixé $j_{i,1}, j_{i,2}, \ldots, j_{i,Gi}$ sont les entiers j tels que CA(i,j)=1, les indices des colonnes non nulles dans la ligne i.
- L'$i^{ème}$ ligne de S^0 (resp. S^1) consiste en le «ou » des lignes de $j_{i,1}, j_{i,2}, \ldots, j_{i,Gi}$ de $\hat{S}^0$ (resp. $\hat{S}^1$).

On obtient ainsi une matrice de dimension |P|× nombre de colonnes de $\hat{S}^0$ ou $\hat{S}^1$.

Ce schéma n'est valable que lorsque les matrices du schéma t parmi t sont des matrices de base qui vérifient les conditions de la définition du chapitre 3 section 3.3.

Théorème 4.3

Soit (Γ_{qual} , Γ_{forb}) une structure d'accès robuste (strong), et soit Z_M la famille des ensembles interdits à cardinal maximal. Il existe un schéma (Γ_{qual}, Γ_{forb},m) de cryptage visuel avec $m=2^{|Z_M|-1}$ *et* $d_x=m$.

Ce théorème revient à optimiser l'expansion de pixel du schéma k parmi k qui est 2^{k-1}. Etant donné que la construction se base sur des matrices d'un schéma k parmi k tel que k=|Z_M|, l'expansion du pixel dépend donc du cardinal de l'ensemble des ensembles interdits.

4.5.2 Construction 2

Cette construction concerne les schémas de cryptage visuel utilisant les petits schémas de cryptage visuel pour en construire de plus importants et de plus étendus par rapport au nombre de participants et à l'ensemble des qualifiés.

Soit (Γ'_{qual} , Γ'_{forb}) et (Γ''_{qual} , Γ''_{forb}) deux structures d'accès sur un ensemble à n participants de P.

Si i∈P n'est pas un participant essentiel à*(Γ'_{qual} , Γ'_{forb}) (resp. (Γ''_{qual} , Γ''_{forb}), on suppose* que i∈ Γ'_{forb} (resp . Γ''_{forb}).
On suppose qu'il existe un schéma de cryptage visuel VCS (Γ'_{qual} , Γ'_{forb}, m') et (Γ''_{qual} , Γ''_{forb} , m'') aux matrices de base : R^0 et R^1 ; T^0 et T^1.

Le but consiste à construire un VCS pour une structure d'accès ($\Gamma'_{qual} \cup \Gamma''_{qual}$, $\Gamma'_{forb} \cap \Gamma''_{forb}$).
A partir des matrices R0, R1, T0 et T1 on construit deux paires de matrices $(\hat{R}^0,\hat{R}^1)$ et $(\hat{T}^0,\hat{T}^1)$ chacune à n lignes.

- Pour i=1,…,n

  Si i n'est pas un participant essentiel de *(Γ'_{qual} , Γ'_{forb})*, la $i^{ème}$ ligne de $\hat{R}^0$ (resp. $\hat{R}^1$) est nulle.

Si i est un participant essentiel de $(\Gamma'_{qual}, \Gamma'_{forb})$, la $i^{ème}$ ligne de de $\hat{R}^0$ (resp. $\hat{R}^1$) est la $i^{ème}$ ligne de R^0.

Répéter la même procédure pour $\hat{T}^0, \hat{T}^1$, relativement à la strucure $(\Gamma''_{qual}, \Gamma''_{forb})$.

- La matrice de base S^0 pour $(\Gamma_{qual}, \Gamma_{forb})$ sera la concaténation de $\hat{R}^0$ et $\hat{T}^0$.
  La matrice de base S^1 pour $(\Gamma_{qual}, \Gamma_{forb})$ sera la concaténation de $\hat{R}^1$ et $\hat{T}^1$.

$$S^0 = \hat{R}^0 \circ \hat{R}^1$$

$$S^1 = \hat{T}^0 \circ \hat{T}^1$$

Ainsi, on obtient la construction requise avec une expansion de pixel doublée, ce qui pourrait être un inconvénient. Cependant, lorsqu'on manipule de petits schémas avec une petite expansion de pixel la concaténation reste acceptable jusqu'à un certain seuil de schémas à déterminer.

La généralisation sera donc énoncée par le corollaire suivant :

Corollaire 4.2

Soit une structure d'accès $(\Gamma_{qual}, \Gamma_{forb})$.

Si

$$\Gamma_{qual} = \bigcup_{i=1}^{q} \Gamma_{(i,qual)}$$

$$\Gamma_{forb} = \bigcap_{i=1}^{q} \Gamma_{(i,forb)}$$

Et si pour chaque i=1,...,q, il existe un schéma de cryptage visuel $(\Gamma_{(i,qual)}, \Gamma_{(i,forb)}, m_i)$ construit avec des matrices de base,

Il existe alors un schéma de cryptage visuel à structure d'accès $(\Gamma_{qual}, \Gamma_{forb}, m)$ construit avec des matrices de base où $m = \sum_{i=1}^{q} m_i$.

Si les q structures sont robustes la structure résultante l'est aussi.

Il est facile de percevoir, à travers ce corollaire, la possibilité de construire un schéma de cryptage visuel à une structure d'accès générale à partir de plusieurs petites structures d'accès indépendantes. Le contraire est également possible lorsqu'il s'agit de décomposer une structure d'accès selon plusieurs structures d'accès élémentaires qui la construisent dans le sens du corollaire précédent. La décomposition la plus évidente est

celle qui comprend les différents ensembles qualifiés minimaux de l'ensemble de base Γ_0. Il en résulte, donc, une estimation de l'expansion du pixel à partir des cardinaux des ensembles éléments de Γ_0.

Théorème 4.3

Soit $(\Gamma_{qual}, \Gamma_{forb})$ une structure d'accès ayant pour base Γ_0.

Il existe un schéma de cryptage visuel $(\Gamma_{qual}, \Gamma_{forb}, m)$ tel que : $m = \sum_{x \in \Gamma_0} 2^{|x|-1}$.

Comme dans les autres cas, l'expansion du pixel dépend de la structure d'accès qui se limitait à un seul paramètre k dans le cas des schémas précédents (k parmi n) et (k parmi k) ou dépendait du cardinal de la structure interdite dans la première construction de cette section. Dans cette seconde construction, l'expansion du pixel dépendra des différents cardinaux des ensembles de la structure qualifiée minimale et même du nombre des ensembles.

Le choix du nombre de participants dans chaque ensemble de la structure qualifiée minimale, peut être limité et ce, pour préserver la qualité de l'image notamment, sa résolution. Il est préférable d'utiliser plusieurs ensembles à petit cardinal, plutôt que quelques ensembles à cardinal important, car la variation de l'expansion du pixel est exponentielle par rapport au cardinal de chaque sous-ensemble.

4.6 Synthèse

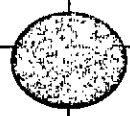
Nous avons décrit un nombre de constructions sélectionnées à partir de chacun des schémas de cryptage visuel ; il en existe beaucoup d'autres développées ou en voie de l'être dont le but est d'améliorer les caractéristiques de l'image reconstruite, principalement, la résolution et le contraste. Un autre objectif de ces constructions est de trouver des méthodes plus simples à la programmation et l'implémentation, car comme on l'a déjà vu l'aspect théorique de l'algèbre et de la géométrie est prépondérant dans la plupart des constructions décrites.

Parmi les méthodes que nous avons décrites, nous avons sélectionné celles qui présentent une meilleure faisabilité (notamment ...) afin d'une implémentation dans la pratique.

Le souci d 'amélioration ne s'arrête pas aux préoccupations par rapport à la qualité de l'image mais il nous porte aussi à travailler sur d'autres types d'images (couleur, niveaux de gris) et d'accomplir d'autres fonctions (stéganographie par exemple). Ces ouvertures et élargissements seront présentées dans le chapitre suivant.

CHAPITRE 5

Ouvertures et développement du cryptage visuel



5.1 Introduction

Le besoin d'accomplir une sécurité évoluée (intelligible), moderne et de qualité, dicte les bases du développement du cryptage visuel vers de nouvelles orientations.

Améliorer le contraste qui n'est pas, encore, défini avec précision ; manipuler des images couleur et en niveaux de gris plus proches aux usages actuels, et enfin rajouter une touche de stéganographie aux schémas de cryptage visuel, telles sont les ouvertures qui motivent les recherches actuelles qui vont faire l'objet du présent chapitre.

5.2 La problématique du contraste

De tous les paramètres qui caractérisent les schémas de cryptage visuel, aucun ne pose autant de difficultés que le contraste. La discorde ne se pose pas seulement par rapport à sa définition mais aussi au niveau de sa nomenclature. Ce que certains qualifient de contraste d'autres l'attribuent à la perte en contraste.

Le contraste α est une mesure de lisibilité.

Etant donné que le noir reconstruit et le blanc reconstruit apparaissent en tant que collection de subpixels noirs et blancs, l'image reconstruite est moins claire que l'image d'origine. Le contraste sera une mesure du degré de clarté dans l'image reconstruite par rapport à l'image originale.

Une construction parfaite (impossible en pratique) possédera un contraste de 1.

Puisque l'on ne considère le contraste que dans l'image reconstruite, on pourra garder k, n et r fixes et ne se préoccuper que des variations de h, ℓ et m.

Intuitivement, on observe que l'accroissement de m correspond à la diminution du contraste α. Il est évident aussi que le contraste augmente avec l'accroissement de h et diminue avec le décroissement de ℓ.

Examinons les mesures actuelles du contraste et voyons pourquoi sont-elles inadéquates.

1) Définition de Naor Shamir

$$\alpha_{NS} = \frac{h-l}{m}$$

Cette définition, à caractère intuitif, reprend le fait qu'effectivement α dépend de h, ℓ et m. Toutefois, l'expérimentation montre que les notions intuitives sont insuffisantes.

Dans le cas de cette définition, α_{NS} dépend de la différence entre h et ℓ mais en réalité les valeurs de h et ℓ sont aussi importantes.

Une démonstration de cet argument est illustrée par les figures 5.2 et 5.3 qui ont le même contraste α_{NS} mais des paramètres h et ℓ différents. On voit bien que l'image de la figure 5.2 est clairement plus identifiable.

C'est pour cette raison que la définition de Naor et Shamir est inadéquate.

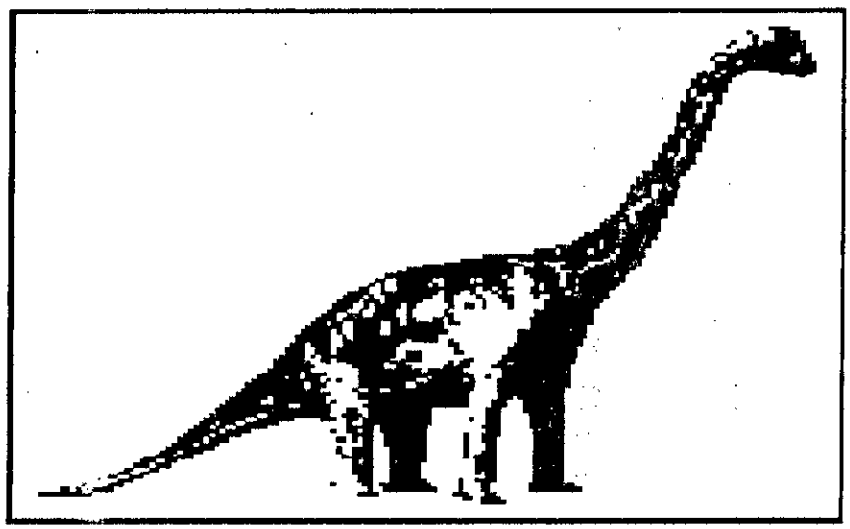


Figure 5.1 : Image originelle

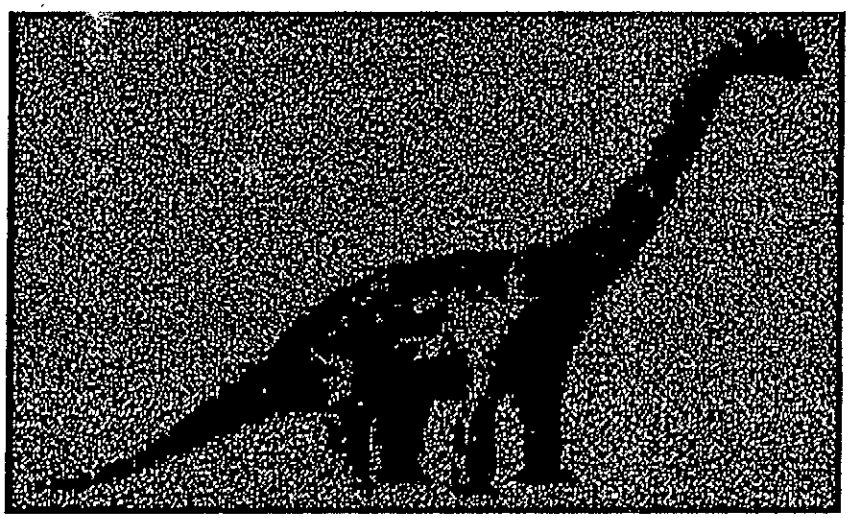


Figure 5.2 : Image reconstruite aux paramètres : h=2, ℓ=0, m=9

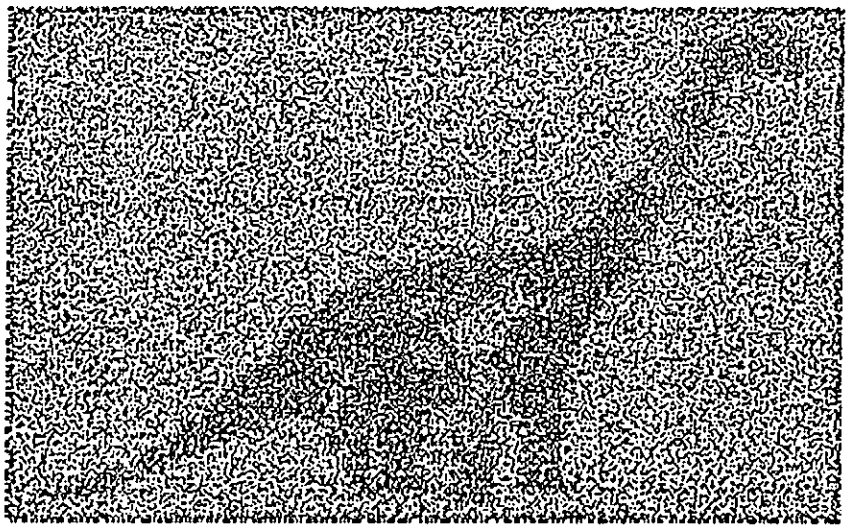


Figure 5.3 : Image reconstruite aux paramètres : h=6, ℓ=4, m=9

2) Définition de Verheul et Van Tilborg

$$\alpha_{VV} = \frac{h-l}{m(h+l)}$$

Dans cette définition, les valeurs de h et ℓ sont aussi importantes que leur différence. Cependant, cette définition est limitée lorsque le noir est parfaitement reconstruit ℓ=0 dans les deux cas suivants :

- h disparaît de l'équation ce qui signifie que le contraste ne dépend que de m. Un exemple est représenté par les figures 5.4 et 5.5 pour confirmer que cela est erroné car la reconstruction du blanc mesurée par h est aussi importante. Les deux exemples de figures ont une même expansion du pixel mais une blancheur h différente pour un recouvrement total du noir ℓ=0. On obtient une clarté de l'image différente dans chacune des images.

- Un contraste parfait α_{VV}=1 n'est possible que si m=1.

  Cela signifie que lors de la reconstruction parfaite des pixels noirs (ℓ=0) et blancs (h=m) on obtient un contraste égal à 1/m ce qui signifie que l'image reconstruite sera moins claire que l'originale. Ce qui est absurde car la reconstruction parfaite du blanc est une limite de h=m impossible à réaliser en pratique.

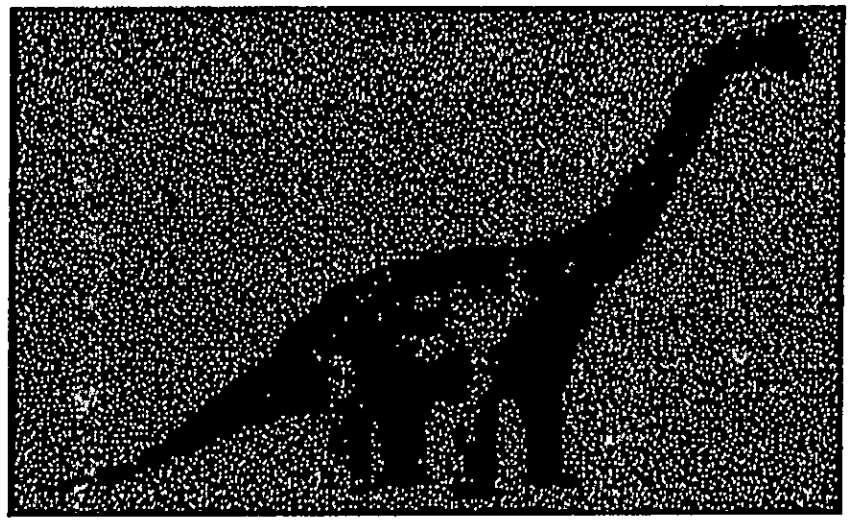


Figure 5.4 : Image reconstruite avec les paramètres h=1, ℓ=0, m=9

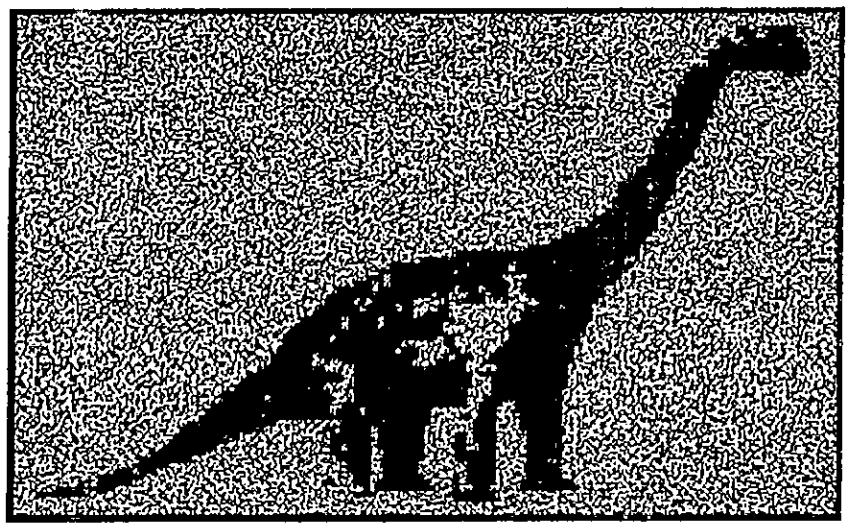


Figure 5.5 : Image reconstruite avec les paramètres h=5, ℓ=0, m=9

3) Nouvelle définition du contraste

Stinson et Eisen proposent une nouvelle définition :

$$\alpha_T = \frac{h-l}{m+l}$$

Celle-ci combine les caractéristiques positives des deux précédentes :

- Le contraste varie avec m, h, ℓ.
- Le contraste varie avec (h-ℓ) et diminue lorsque ℓ croit.
- Lorsqu'il y a reconstruction totale du noir (ℓ=0) alors α_T = h/m. Ce qui concorde avec le fait que le contraste dépend toujours de h.
- Lorsque le noir et le blanc sont parfaitement reconstruits ℓ=0 et h=m le contraste α_T=1.

Un exemple est présenté avec des valeurs égales de α_T =1/3 .

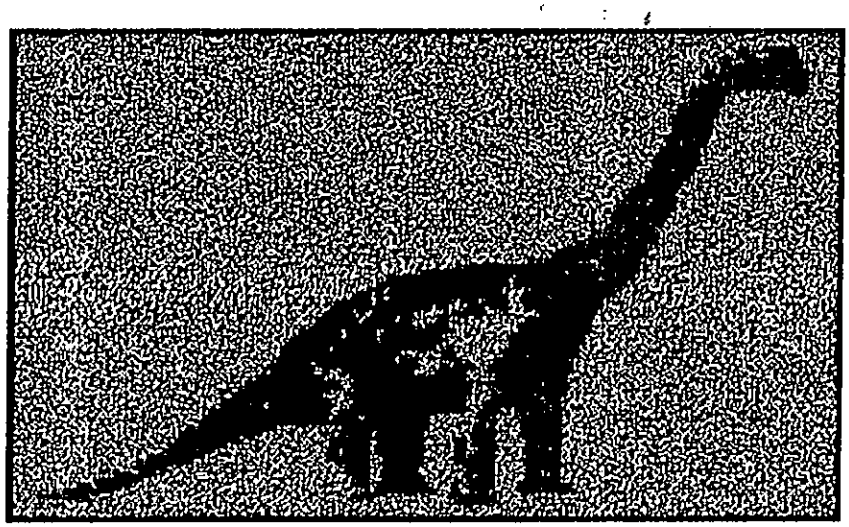


Figure 5.6 : Image reconstruite pour h=3, ℓ=0 et m=9

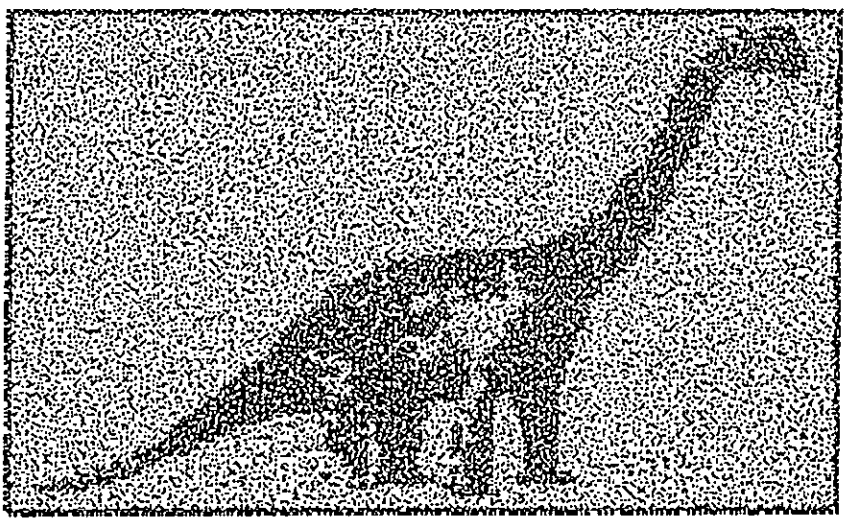


Figure 5.7 : Image reconstruite pour h=7,ℓ=3 et m=9

La définition du contraste et sa mesure figurent encore parmi les problèmes principaux du cryptage visuel puis intervient la préoccupation de l'optimiser. Car dans un schéma optimal le contraste doit être maximal.

Dans les définitions de α_{NS} et α_{VV}, l'interaction entre m, ℓ et h n'est pas évidente. Chaque paramètre paraît influer sur le contraste indépendamment des autres. Mais cela n'est qu'une apparence.

Tout d'abord, m ne varie pas seulement en fonction de h et ℓ mais aussi en fonction de k et n, comme cela a été constaté dans les constructions des schémas VCS du chapitre 4.

Optimiser le contraste revient à déterminer l'expansion de pixel minimal $m_{min}(k,n,h,\ell)$ pour une combinaison de k, n, h et ℓ , puis à l'introduire dans l'expression du contraste, car dans toutes les définitions du contraste, un m minimal (optimal) correspond à un contraste maximal (optimal).

On obtient alors l'expression de m_{min} en fonction de h et ℓ pour k et n fixés que l'on introduira dans l'expression du contraste α_T qui sera fonction de h et ℓ (variables indépendantes).

L'optimisation du contraste consiste à maximiser la fonction $\alpha_T(h, \ell)$ par rapport à h et ℓ.

On introduit le théorème suivant :

Théorème 5.1

Pour un quadruplé (k,n,h,ℓ) donné réalisant :

1. $k \in \mathbb{Z}$; $k \geq 2$.
2. $n \in \mathbb{Z}$; $n \geq 2$.
3. $k \leq n$.
4. $h \in \mathbb{Z}$; $h \geq 1$.
5. $\ell \in \mathbb{Z}$; $\ell \geq 0$.
6. $h > \ell$

un schéma de cryptage visuel existe.

La problématique d'optimisation d'un schéma VCS consiste à trouver une valeur minimale m pour laquelle un schéma (k,n,h,ℓ,m) existe. Ce qui revient à déterminer un contraste maximal et donc optimal pour ce même schéma.

Les recherches ont abouti à des expressions formelles de m_{min} en fonction de h, ℓ et n pour le cas k=2. il a même été proposé des programmes linéaires de calcul de m minimal, mais cela relève du cas particulier.

Parmi les questions ouvertes relatives à l'optimisation d'un schéma de cryptage visuel, figure celle de trouver une expression explicite de $m_{min}(k,n,h,\ell)$ pour $k \geq 2$.

5.3 La dissimulation de l'existence d'un schéma de cryptage visuel

Dans les schémas de cryptage visuel précédents, un nombre de transparents ou parts est généré. Chaque transparent est semblable aux autres, il apparaît comme un arrangement aléatoire de pixels noirs et blancs.

Il a été décrit et développé dans [13] un schéma de cryptage visuel qui propose que les transparents portent des images anodines.

On appelle image originale toute image apparente sur un transparent.

L'image secrète est celle qui est révélée lors de la superposition des transparents des participants qualifiés.

Les motivations de la cryptographie visuelle étendue sont les suivantes :

- Lorsque l'on superpose toutes les parts qualifiées, on obtient l'image secrète. L'image originale se dissipe avec l'avantage que chaque participant reconnaisse son transparent devenu significatif. Cette technique permet d'identifier les participants par l'image que portent leurs transparents.

- Une deuxième variante des schémas de cryptage visuel étendu consiste à partager plus d'un secret entre un ensemble de participants. Par exemple, chaque paire de participants décode un secret différent.

- Un autre but est recherché à travers ces schémas et consiste à dissimuler l'existence l'information secrètes dans une masse d'informations (graphiques dans notre cas). Ce qui répond à la définition de la stéganographie. Ainsi on accède à une sécurité plus élevée et plus subtile.

On définit un schéma de cryptage visuel étendu EVCS pour une structure d'accès générale selon la définition suivante :

Définition 5.3

- *Soit $(\Gamma_{qual}, \Gamma_{forb})$ une structure d'accès pour un ensemble de n participants.*

Une famille de 2^n paires de collections de matrices booléennes de dimension $n \times m$,

- *$\{(C_w^{c1\ldots cn}, C_b^{c1\ldots cn})\}$ constitue un schéma de cryptage visuel étendu $(\Gamma_{qual}, \Gamma_{forb}, m)$ VCS s'il existe des valeurs $\alpha_F(m)$, $\alpha_S(m)$ et un ensemble $\{(X^{c1\ldots c2\in\{b,w\}}, d_x)\}_{X\in\Gamma equal}$ satisfaisant :*

1. *Chaque ensemble qualifié $X \in \Gamma_{equal}$ peut recouvrir l'image partagée par la superposition des transparents.*
   - *Formellement :*
   - *$\forall X \in \Gamma_{qual}$, et $\forall c_1, \ldots c_n \in \{b,w\}$, le seuil tx et la différence relative $\alpha_F(m)$ sont tels que :*
   - *$\forall M \in C_w^{c1,\ldots,cn}\ H(M_X) \le d_x - \alpha_F(m) \times m$*
   - *alors que*
   - *$\forall M \in C_b^{c1,\ldots,cn}\ H(M_X) \ge d_x$;*
2. *Chaque ensemble interdit $X=\{i_1, i_2, \ldots, i_p\} \in \Gamma_{forb}$ n'a aucune information sur l'image partagée.*
   - *Plus Formellement :*
   - *$\forall X \in \Gamma_{forb}$ et $\forall c_1, \ldots c_n \in \{b,w\}$ Les deux collections de $p \times m$ matrices $D_t^{c1\ldots cn}$, $t=\{b,w\}$, obtenues par la restriction de chacune des matrices $n \times m$ de $C_t^{c1\ldots cn}$, $t=0,1$, aux lignes $i_1, i_2, \ldots, i_p$ sont impossibles à distinguer les unes des autres, dans le sens où elles contiennent les mêmes matrices avec les mêmes fréquences.*
3. *Après avoir encodé les images originales, celles-ci restent significatives jusqu'à ce que n'importe quel utilisateur reconnaisse l'image de son transparent.*
   - *Plus formellement :*
   - *$i \in \{1 \ldots n\}$ et $\forall c1, \ldots, ci\text{-}1, ci\text{+}1, \ldots, cn \in \{b,w\}$, il en résulte que :*
   - $\min_{M\in M_b} H(M_i) - \max_{M\in M_w} H(M_i) \ge \alpha_S(m) m$

Où $M_b = \bigcup_{c_1,\ldots,c_{i-1},c_{i+1},\ldots,c_n \in \{b,w\}} C_w^{c_1\ldots c_{i-1} b c_{i+1}\ldots c_n}$

Et $M_w = \bigcup_{c_1,\ldots,c_{i-1},c_{i+1},\ldots,c_n \in \{b,w\}} C_w^{c_1\ldots c_{i-1} w c_{i+1}\ldots c_n}$

Les valeurs α_F *(m)* et α_S *(m)* sont les différences relatives de l'image finale et du transparent respectivement.

$C_b^{c_1 \ldots c_n}$ est la collection de matrices pour encoder un pixel noir tel que dans chaque transparent ce même pixel est, respectivement, de couleurs suivantes $c_1, c_2, \ldots c_n$.

$C_w^{c_1 \ldots c_n}$ est la collection de matrices pour encoder un pixel blanc tel que dans chaque transparent ce même pixel est respectivement de couleurs suivantes $c_1, c_2, \ldots c_n$

Dans les deux premières constructions, on retrouve les conditions sur le contraste et la sécurité. La dernière condition exprime le fait que les images originales ne sont pas modifiées une fois qu'on les a encodées en utilisant les 2^n paires de collections *($C_w^{c1 \ldots cn}$, $C_b^{c1 \ldots cn}$), tel que chaque utilisateur reconnaisse son transparent.*

Cette troisième condition exprime que la différence entre le minimum du poids de Hamming d'un pixel noir du transparent et le maximum du poids de Hamming d'un pixel blanc de ce transparent est supérieure ou égale à la différence minimale de pixel existante entre un pixel blanc et noir dans ce même transparent.

Les couleurs d'un pixel dans chaque image originale $c_1 \ldots c_n$, indexent et désignent la paire de collections de matrices que l'on utilise *($C_w^{c1 \ldots cn}$, $C_b^{c1 \ldots cn}$)*. Ensuite les matrices seront choisies aléatoirement dans la collection correspondante à la couleur du pixel dans l'image secrète.

La construction de ces matrices repose sur les principes de la théorie des hypergraphes colorés pour le schéma k parmi n. Quant au schéma k parmi k l'algorithme est donné ci aprés :

Construction d'un schéma de cryptage visuel étendu k parmi k :

Soit $(\alpha_F, \alpha_S)=(d/e,f/g)$ *et* $h=eg(k-1)-2^{k-1}dg(k-1)-kef$.
Tel que : $2^{k-1}\alpha_F + \frac{k}{k-1}\alpha_S \leq 1$.

- En entrée

1. Les matrices de base d'un schéma k parmi k : S^0 et S^1 de dimension $k \times 2^{k-1}$.
2. T une matrice $k \times h$ nulle.
3. Les couleurs $c_1, \ldots, c_k \in \{b;w\}$ des pixels dans les k images originales.

4. La couleur c∈{b;w} du pixel de l'image secrète

- La génération des k transparents :

1. Construire une matrice D de dimension k × k tel que :

pour i = 1 jusqu'à k faire
si c_i = b alors poser tous les éléments de la ligne i de D égaux à 1.
Sinon
mettre l'élément (i; i) de D à1 et le reste de la ligne i à 0.

2. La collection $C_c^{c_1 \ldots c_k}$ est construite selon les matrices obtenues lors de la permutation des colonnes de la matrice

$$S_c^{c_1,\ldots,c_k} = \begin{cases} \underbrace{S^0 \circ \ldots \circ S^0}_{(k-1)dg} \circ \underbrace{D \circ \ldots \circ D}_{ef} \circ T / c = w \\ \underbrace{S^0 \circ \ldots \circ S^0}_{(k-1)dg} \circ \underbrace{D \circ \ldots \circ D}_{ef} \circ T / c = b \end{cases}$$

3. soit M la matrice choisie aléatoirement de la collection $C_c^{c_1,\ldots,c_k}$

- En sortie

La matrice M.

Ces différentes étapes représentent la construction des matrices du schéma de cryptage visuel étendu k parmi k de dimension k × (e.g.(k-1)) .

On voit que l'expansion du pixel dépend, dans ce cas, du contraste des images originales comme du contraste de l'image secrète ; et bien sûr du nombre de participants.

Les contrastes α_F et α_S , fixés d'avance, répondent à la condition : $2^{k-1}\alpha_F + \frac{k}{k-1}\alpha_S \leq 1$ qui définit la *région admissible.*

Cette ouverture permettra, par exemple, d'utiliser le cryptage visuel en tant que moyen de certification ou d'authentification pour l'argent ou les documents officiels.

Exemple

Soit la structure d'accès (Γ_{qual}, Γ_{forb}) sur l'ensemble P={1,2,3}.

L'ensemble qualifié Γ_{qual} ={{1,2},{2,3},{1,2,3}}

Le reste des sous-ensembles de 2^P sont interdits.On considère que :$\alpha_F = \alpha_S$=1/4.

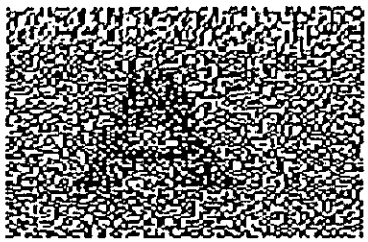


Figure 5.8 : Part du participant 1

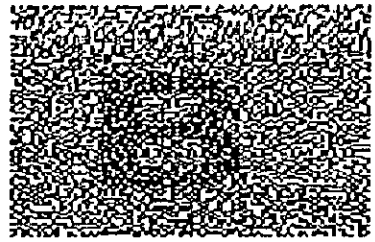


Figure 5.9 : Part du participant 2

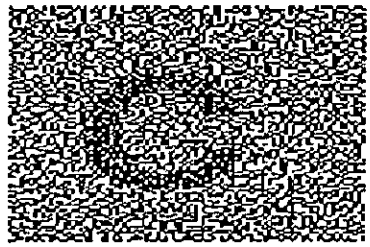


Figure 5.10 : Part du participant 3

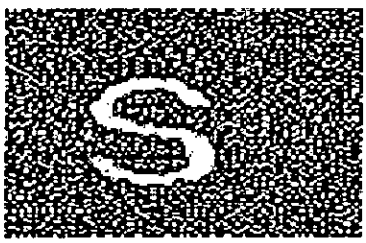


Figure 5.11 : Superposition de{1,2}

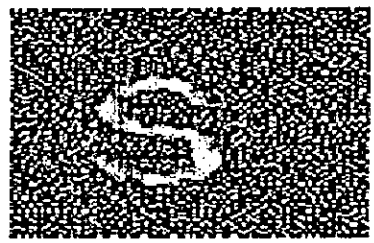


Figure 5.12 : Superposition de {2, 3}

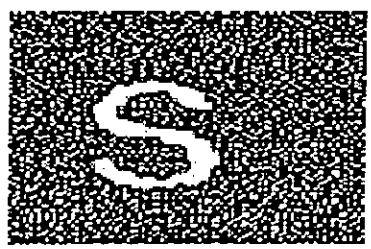


Figure 5.13 : Superposition de {1,2,3}

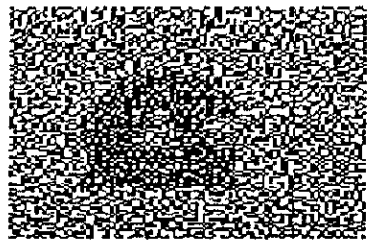


Figure 5.14 : Superposition {1,3} ensemble interdit

5.4 Encoder une image en niveaux de gris

Considérons une image à niveau de gris où le ton du pixel s'étend de 0(blanc) à 255(noir) niveaux de gris. On présentera le cryptage visuel de ces images à travers le cas particulier 2 parmi 2.

Pour encoder une image à niveau de gris, on pourrait envisager de coder chaque pixel à niveau de gris g, dans une matrice de 16× 16 subpixels contenant g subpixels noirs et 256-g subpixels blancs.

Il est possible ensuite appliquer les techniques de cryptage visuel étudiées précédemment à chaque subpixel.

L'inconvénient de cette approche réside dans le fait que le nombre total de pixels est multiplié par 16× 16, puis par l'expansion du pixel m. Le nombre de pixels de l'image originale est multiplié par 16 ×16 ×m dans l'image reconstruite. On observe, alors, une perte considérable en résolution. Si l'on démarre, par exemple, d'une image 100×100 pixels et que l'on adopte une méthode de cryptage 2 parmi 2 avec une expansion de pixel(m=4), on obtient une image reconstruite de 3200×3200 pixels noirs et blancs.

Ce problème peut être évité en codant le pixel à niveau de gris par un cercle à demi plein en noir.

Un pixel blanc, par exemple, sera partagé en deux cercles demi-pleins, identiques dans leur disposition ou inclinaison. Ce qui produit un minimum de noir lorsque l'on superpose ces deux pixels en phase avec une proportion de 50% entre noir et blanc. (Voir figure 5.15)

Figure 5.15 : Encrypter les pixels à niveau de gris avec des cercles demi-pleins : cas du pixel blanc

Un pixel gris est encodé par deux pixels inclinés de telle manière que, de leur superposition résulte un pourcentage de noir par rapport au blanc proportionnel au niveau de gris de l'image d'origine (Voir image 5.16).

Figure 5.16 : Encrypter les pixels à niveau de gris avec des cercles demi-pleins : cas du pixel gris

Quant au codage du pixel noir, il suffit de l'effectuer de telle manière à le reconstruire par la superposition de deux pixels en opposition de phase ou inclinés l'un par rapport à l'autre de 180°. (Voir figure 5.16).

Figure 5.17 : Encrypter les pixels à niveau de gris avec des cercles demi-pleins : cas du pixel noir

Ainsi, les cercles sont placés et orientés aléatoirement dans le premier transparent. Dans le deuxième transparent, leur disposition doit correspondre au niveau de gris de l'image originale.

5.5 Cryptage visuel des images en couleur

Parmi les extensions des schémas de cryptage visuel, il a été proposé d'ajouter celle des couleurs au support de la cryptographie visuelle.

Pour simplifier la présentation, on décrira uniquement le cas 2 parmi 2.

Dans [11], il a été proposé une méthode qui encrypte les images colorées. Celle-ci peut être résumée à travers les étapes suivantes :

- Pour une couleur c, chaque pixel est divisé en c subpixels,
- Chaque subpixel est divisé en c régions fixes, chacune relative à une des c couleurs,
- Dans chaque subpixel i, une seule région fixe sera colorée par la couleur c_i associée au subpixel, les autres régions de ce même subpixel sont noires. On dit alors que le subpixel a la couleur c_i,
- Les subpixels étant construits ainsi, la superposition de deux d'entre eux est noire sauf si les deux subpixels sont de la même couleur.
- Dans le premier transparent, les subpixels sont disposés aléatoirement. Cependant, dans le second transparent seul le subpixel représentant la couleur originale coïncide avec son similaire du premier transparent.

Il en résulte que seule cette couleur transperce le subpixel qui la représente lors de la superposition des transparents.

L'inconvénient majeur de cette méthode est que l'on observe une réduction considérable de la résolution. Si l'on encrypte une image à c couleurs on obtient une perte de résolution avec un facteur c^2 qui représente également l'expansion du pixel.

Exemple

Soit une image à 4 couleurs : rouge, vert, bleu, jaune.

Dans cet exemple nous allons expliquer comment encrypter un pixel jaune.

Celui-ci est divisé en 4 subpixels.

Chaque subpixel contient 4 régions fixes, chacune représentant une des 4 couleurs de base.

Chaque subpixel représente une couleur de base c_i en colorant la région correspondante à cette couleur et en laissant les autres sombres.

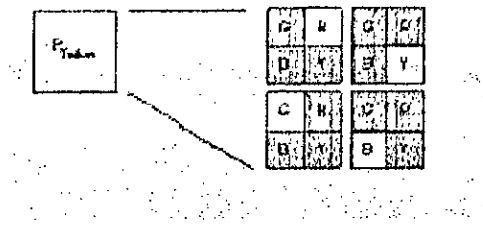


Figure 5.18 : Premier transparent du pixel jaune crypté

Dans le premier transparent, les subpixels sont distribués d'une manière aléatoire.

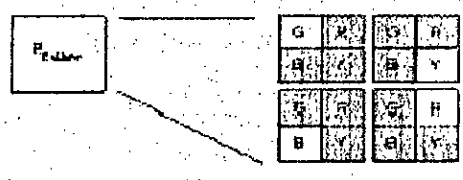


Figure 5.19 : Second transparent du pixel jaune crypté

Dans le second transparent, le subpixel représentant le jaune se trouve dans la même position que son semblable dans le premier transparent. Quant aux autres subpixels, ils sont distribués de telle sorte qu'aucun ne coïncide avec son semblable lors de la superposition des transparents. Ce qui revient à dire qu'aucune autre région colorée, à part la région jaune, ne pourra être alignée à sa similaire du second transparent lors de la superposition.

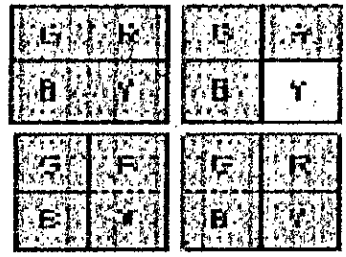


Figure 5.20 : Superposition des deux transparents : reconstruction du pixel jaune

Lorsque les transparents sont superposés, tous les subpixels sont totalement noirs sauf un quart du subpixel représentant le jaune.

On observe dans l'image reconstruite que seule une région sur 16 est colorée, les autres étant noires. Il en résulte une image principalement sombre ou noire avec des points colorés qui la traversent de part et d'autre.

Cette observation est justifiée par la perte considérable en résolution.

Dans la méthode précédente, seules deux situations sont susceptibles d'avoir lieu. D'une part, deux régions de différentes couleurs coïncident et le résultat de leur superposition est noir ; d'autre part, ces rég ions sont de couleur similaire et le résultat de leur superposition correspond à cette même couleur.

Une autre approche propose une solution différente qui consiste à superposer deux couleurs différentes pour en générer une troisième.

Dans la version de base de ce schéma, chaque pixel est divisé en quatre subpixels de couleur élémentaire : rouge, bleu, vert ou blanc.

Ces subpixels peuvent apparaître dans un ordre aléatoire à l'intérieur du pixel. Il en résultera 24 possibilités de superpositions différentes entre deux pixels. Si ces pixels sont suffisamment petits, le système visuel humain moyenne les 24 combinaisons possibles en 24 couleurs différentes.

Pour encrypter un pixel, il suffit de choisir une combinaison aléatoire de subpixels pour le premier transparent. Quant au deuxième transparent, on sélectionne un ordre pour les subpixels pouvant favoriser la couleur résultante de la combinaison des transparents à se rapprocher de la couleur d'origine.

L'avantage de cette méthode réside dans le fait que l'on représente 24 couleurs avec une réduction de résolution de 4 au lieu de 24^2.

Cependant, ces 24 couleurs ne sont fixées qu'une fois les couleurs de base choisies, ce qui constitue l'inconvénient principal de cette méthode.

D'autres schémas reposent sur la caractérisation de chaque couleur par une longueur d'onde λ. Il est ainsi possible de construire des filtres qui ne se laissent traverser que par des lumières de longueurs d'onde appartenant à un certain intervalle. Cette alternative physique de partage visuel de secret utilise deux filtres choisis de telle sorte que leur superposition absorbe toutes les couleurs sauf celle du pixel de l'image originale.

5.6 Synthèse

Un paradigme est souvent rendu intéressant par l'étendue de ses applications et les variantes qu'il propose. Le présent chapitre nous révèle que les schémas de base du cryptage visuel, traités en détail précédemment, ne sont pas rigides mais présentent, au

contraire, des ouvertures et élargissements qui permettent à la théorie du cryptage visuel d'évoluer pour être exploitée au mieux dans les cryptosystèmes futurs.

Les cryptosystèmes modernes reposent, le plus souvent, sur des réalisations software présentées sous forme d'une interface graphique d'utilisateurs (GUI) conçue de la manière la plus intelligible que possible.

De plus, le cryptage fait partie aujourd'hui des pratiques privées. N'importe qui peut être en mesure de protéger ses informations et ses données même sans avoir suffisamment de connaissances dans le domaine cryptographique ou informatique. Un cryptosystème peut être assimilé à un produit prêt à l'utilisation.

Le cryptage visuel facilite l'exécution de cette notion étant donné que le décryptage est réalisé par le système visuel humain. Quant au cryptage il est le résultat d'une succession d'étapes dont principalement :

- Le choix de la méthode de construction,
- Le passage du formalisme mathématique algébrique aux techniques numériques et algorithmiques (programmation).

Après avoir défini et présenté le paradigme du cryptage visuel dans cette deuxième partie, nous allons élaborer une réalisation pratique qui met en exécution les idées de la théorie du cryptage visuel sous la forme d'un cryptosystème moderne convivial à l'utilisation. Tel sera l'objet de la partie III de ce document intitulée : implémentation du logiciel.

PARTIE III

Implémentation du logiciel

- **Chapitre 1 : Justification du choix du langage**
- **Chapitre 2 : Implémentation du logiciel et formalisation des algorithmes**
- **Chapitre 3 : Résultats et tests**

CHAPITRE 1 Justification du choix du langage

1.1 Introduction

1.2 Choix du langage de programmation et analyse des besoins

1.3 L'approche "langage de script"

1.4 Présentation du langage Python

1.5 Autres caractéristiques du langage

1.6 Domaines d'application

1.7 Synthèse

1.1 Introduction

Dans cette partie nous présenterons un compte rendu de l'implémentation du logiciel que nous avons réalisé et qui consiste en un cryptosystème de partage de secret visuel capable de crypter des images bitmaps par les constructions générales k parmi k et k parmi n.

Pour réaliser ce logiciel, on a été amené à :

- Choisir les constructions et méthodes générales qui se rapprochent le plus de la pratique,
- Etudier et détailler le formalisme mathématique afin d'en extraire l'algorithme,
- Choisir un langage de programmation adapté aux besoins recherchés.

L'introduction de cette partie sera consacrée à la justification du choix du langage de programmation Python ; nous présenterons, ensuite, notre interface en retraçant les étapes directrices de la conception du software réalisé. Enfin, nous exposerons quelques tests destinés à évaluer les résultats de notre réalisation.

Dans ce premier chapitre, nous citerons d'abord les critères qui ont décidé du choix du langage de programmation pour notre application ; nous y justifierons ensuite l'utilisation de Python, puis on passera à la présentation de ce langage avec un bref exposé de ses caractéristiques principales.

1.2 Choix du langage de programmation et analyse des besoins

En commençant un nouveau projet, la première question que devrait se poser un programmeur est : «quel est le langage de programmation le plus approprié à mon application ? ». Il existe, certes, de nombreux langages de programmation, mais rares sont ceux qui sont largement utilisés. A cela il convient d'ajouter que de nouveaux langages «spécialisés » ou «généralistes » voient sans cesse le jour. Le plus souvent, leur diffusion reste restreinte voire confidentielle. Pour des raisons de portabilité, d'efficacité et plus généralement de spécificité relative à son application le programmeur préfère tel langage plutôt qu'un autre.

Un point qui doit être mis fortement en exergue est la nature dynamique de beaucoup d'applications informatiques scientifiques développées dans le domaine de la recherche.

Une application développée dans un contexte de recherche ne peut pas être un énorme « package » monolithique qui ne change jamais et que l'utilisateur final n'est pas supposé modifier.

Une application est un programme dont le code change en permanence pour explorer de nouveaux aspects du problème à résoudre, essayer de nouvelles méthodes numériques, rechercher de meilleures performances sur de nouvelles architectures de machines.

L'objectif principal des chercheurs est de pouvoir rompre avec le lourd cycle traditionnel : *« programmer, compiler, éditer, exécuter, debugger »* imposé par la mise en œuvre des langages de programmation classiques (Fortran, C, C++,...).

Les chercheurs aimeraient :

- pouvoir construire une application rapidement,
- prototyper de nombreuses variations sur une idée à partir de l'assemblage d'éléments déjà existants,
- disposer de riches bibliothèques faciles à exploiter même si elles sont d'origines hétérogènes, en particulier dans les domaines des algorithmes numériques et des représentations graphiques.

Cela leur permettraient de mettre en œuvre facilement les concepts de développement à partir de composants de haut niveau, dans un contexte interactif ouvert, sans avoir à acquérir une compétence approfondie sur les systèmes informatiques et les langages de programmation.

Grâce aux techniques du "développement orienté objet ", quelques progrès ont été faits dans cette direction mais l'objectif est loin d'être atteint. Les langages objets comme le C++ sont difficiles à maîtriser et les interfaces entre les différents composants sont complexes à réaliser.

1.3 L'approche "langage de script"

Depuis quelques années, on assiste à l'existence d'un changement jugé fondamental dans la manière dont certaines grandes applications sont développées. Ce changement est marqué par une transition allant des langages que l'on qualifie de «orientés système » ou de «haut niveau » comme le FORTRAN, le C, le C++,... vers l'emploi de langages dits de «script » comme Perl, TCl, Python... Ces langages de script sont conçus pour des tâches différentes des langages de «haut niveau » ce qui conduit à des différences fondamentales.

Les langages de «haut-niveau » ou traditionnels ont été conçus pour construire et manipuler des structures de données et des algorithmes à partir d'éléments de base de l'ordinateur. tel que le «mot mémoire ». En revanche, le rôle principal des langages de script, basés sur l'existence d'un ensemble de composants puissants (programmés en langage de haut niveau), est de connecter ces composants entre eux. Les langages de «script » et les langages de «haut niveau » sont complémentaires.

D'après les expériences de différents chercheurs, l'introduction des langages de sript dans un grand code a un effet profond sur la structure de l'application. Avec le temps le code devient plus modulaire et mieux organisé. Ces gains sont atteints, sans perte significative de performance, sans augmentation de la complexité, sans majoration des coûts de développement. Le gain est très important dans la phase de maintenance et d'évolution du produit.

Ainsi, on constate que les langages de script répondent mieux aux attentes et aux besoins de la recherche cités précédemment. Ceci s'est avéré exact lors des essais d'implémentation des méthodes de cryptage visuel en langage évolué C++. Les opérations à effectuer par l'utilisateur sur les images se sont révélées nombreuses ; la création d'un script ''shell'' ou une ''batchfile'' serait encombrante compte tenu du nombre de cas à traiter et du nombre de slides à générer et à manipuler.

Par ailleurs, le langage C++ étant un langage de haut niveau la compilation puis l'exécution font que le temps d'exécution augmente même pour de simples modifications sur le programme.

Toutes ces raisons justifient notre choix pour les langages de script ; mais la question suivante reste posée :

« Comment choisir le langage adéquat à notre application parmi les nombreux langages de script existant ? »

Notre application concerne, principalement, un traitement sur les images. Elle se caractérise par de nombreux calculs matriciels et opérations logiques sur les matrices. Ce qui nécessite le recours à des librairies propres aux images qui nous permettraient d'intervenir directement sur le pixel.

Un langage extensible est recommandé pour pouvoir se servir de librairies écrites en d'autres langages de plus haut niveau, pour permettre de les combiner et de les mettre en œuvre rapidement en langage compilé (C, C++, Fortran, Ada, ...)

Les langages de script sont le plus souvent recommandés pour la présentation d'une interface qui serait plus rapide, plus conviviale et plus intelligible grâce à l'utilisation de librairies spécialisées.

Pour alléger l'écriture du programme et le rendre plus lisible, un langage dynamique (évaluation des chaînes de caractères) et dynamiquement typé (les objets ne sont pas déclarés à l'avance) serait préférable.

Pour une programmation plus flexible un langage réflectif supportant la métaprogrammation (la capacité pour un objet de se rajouter ou s'enlever des attributs ou méthodes, ou changer de classe) serait souhaitable.

Enfin, l'utilisation de notre logiciel ne sera que plus étendue si celui-ci est portable et multi-plateforme.

Pour satisfaire toutes ces exigences, un certain consensus s'est réalisé autour du langage Python. D'ailleurs, c'est ce qui justifie notre choix du langage de script Python auquel on a combiné le module d'interface graphique Tkinter et la librairie d'image PIL.

1.4 Présentation du langage Python

Python a été développé à Amsterdam en 1991 par Guido van Rossum à l'université d'Amsterdam. Il était destiné à être un langage avancé de scripting pour le système d'exploitation Amoeba.

Python est un langage portable, dynamique, extensible, gratuit, qui permet une approche modulaire et orientée objet de la programmation. Il est utilisé pour combiner et rassembler un large éventail de composants softwares qui en font un langage à structure de données puissantes de haut niveau.

Python est souvent comparé aux autres langages comme Java, JavaScript, Perl, Tcl, etc...

Ces comparaisons se basent souvent sur les problèmes posés par ces langages. Mais en pratique, le choix du programme reste tributaire des contraintes de la réalité comme le coût, la disponibilité, le temps d'apprentissage, l'investissement ou même les tendances.

Les programmeurs préfèrent, souvent, Python car son utilisation procure une productivité plus élevée.

La syntaxe élégante de Python et son typage dynamique ajoutés à sa nature interprétée permettent un développement rapide des applications sur la plupart des plates-formes.

Ainsi, Python constitue un langage idéal pour rendre les idées exécutables (making ideas executable) en raison de sa rapidité du développement justifiée précédemment et pour communiquer les idées étant donné sa lisibilité. Cette dernière combinée au système d'exceptions permet de réduire considérablement le coût de la maintenance des programmes.

A toutes ces caractéristiques il y a lieu d'ajouter le fait que Python soit gratuit.

1.5 Autres caractéristiques du langage

- Python est **portable** non seulement sur les différentes variantes d'Unix mais aussi sur les OS propriétaires: MacOS, BeOS, NeXTStep, Ms-DOS et les différentes variantes de Windows. Un nouveau compilateur, baptisé JPython, est écrit en Java et génère du bytecode Java,
- Python gère ses ressources (mémoire, descripteurs de fichiers...) sans intervention du programmeur, par un mécanisme de **comptage de références**,
- Python est (optionnellement) **multi-threadé**,
- Python est **dynamique** (l'interpréteur peut évaluer des chaînes de caractères représentant des expressions ou des instructions Python), **orthogonal** (un petit nombre de concepts suffit à engendrer des constructions très riches), **réflectif** (il supporte la *métaprogrammation*, par exemple la capacité pour un objet de s'ajouter ou de s'enlever des attributs ou des méthodes, ou même de changer de classe en cours d'exécution) et **introspectif** (un grand nombre d'outils de développement, comme le *debugger* ou le *profiler*, est implanté en Python lui-même),
- Comme *Scheme* ou *SmallTalk*, Python est **dynamiquement typé**. Tout objet manipulable par le programmeur possède un type bien défini à l'exécution, qui n'a pas besoin d'être déclaré à l'avance,
- L'utilisation de types de données évoluées (listes, dictionnaires, librairies écrites en langage évolué) conduit à des programmes compacts et lisibles,
- Python est un **langage d'extension** pour les systèmes de logiciels complexes, il est à son tour extensible par des librairies écrites en d'autres langages,

- La **librairie standard** de Python et les paquetages contribués donnent accès à une grande variété de services : chaînes de caractères et expressions régulières, services Unix standard (fichiers, *pipes*, signaux, sockets, threads...), protocoles Internet (Web, News, FTP, CGI, HTML...), persistances et bases de données, interfaces graphiques,
- Python est un langage en évolution,
- Son concepteur a dit «l'avenir de Python serait assuré s'il lui arrivait malheur ».

1.6 Domaines d'application

Les domaines d'application naturels de Python incluent entre autres :

- L'apprentissage de la programmation objet,
- Les scripts d'administration système ou d'analyse de fichiers textuels,
- Tous les développements liés à l'Internet et en particulier au Web : scripts CGI, navigateurs Web, moteurs de recherche, agents intelligents, objets distribués... ,
- L'accès aux bases de données (relationnelles),
- La réalisation d'interfaces graphiques utilisateurs,
- Le calcul scientifique et l'imagerie. Python ne sert pas alors à écrire les algorithmes mais à combiner et mettre en œuvre rapidement des librairies de calcul écrites en langage compilé (C, C++, Fortran, Ada,...),
- Le prototypage rapide d'applications. L'idée générale est de commencer à écrire une application en Python et de la tester. Trois cas peuvent alors se présenter :
    - Les performances sont satisfaisantes après optimisation éventuelle du code Python. On livre alors le produit tel quel au client,
    - Les performances ne sont pas satisfaisantes mais l'analyse de l'exécution du programme (à l'aide du *profiler* de Python) montre que l'essentiel du temps d'exécution se passe dans une petite partie du programme. Les fonctions ou les types de données correspondants sont alors réécrites en C ou en C++, sans modification du reste du programme,
    - Sinon, il est toujours possible de réécrire tout le programme en utilisant la version Python comme brouillon,

Même dans le moins avantageux des trois cas, le temps de développement aura été sensiblement plus court si le programme avait été développé directement en C ou en C++.

1.7 Synthèse

Dans ce qui précède, nous avons essayé de présenter le langage utilisé pour la réalisation de notre application. On relève que celle-ci vise à concrétiser la théorie du cryptage visuel, à la démystifier et à confirmer sa simplicité. La raison principale de l'utilisation de python est qu'il constitue un langage capable d'exécuter les idées de la théorie rapidement, intelligiblement, facilement et fidèlement.

C'est un logiciel *libre* qui favorise la mise en exécution et leur communication à travers la lisibilité des programmes. L'ensemble de ces données reflète les tendances actuelles de la recherche scientifique.

CHAPITRE 2

Implémentation du logiciel et formalisation des algorithmes

2.1 Introduction

Dans ce chapitre, il sera question de présenter l'implémentation effectuée dans le cadre de ce projet qui représente l'application de schémas de cryptage visuels présentés dans le chapitre 4.

Nous commencerons, tout d'abord, par la présentation de l'interface ; ensuite nous énoncerons l'exécution de notre application pour donner, enfin, un aperçu sur les différents modules traitant des méthodes de cryptage visuel explicitées précédemment.

2.2 Présentation de l'interface

L'interface graphique de notre logiciel de cryptage visuel a été réalisée en utilisant la librairie graphique Tkinter de Python. Celle-ci permet le développement et la création de différents menus, boutons, barres, etc...

L'interface générale de notre application se présente comme suit :

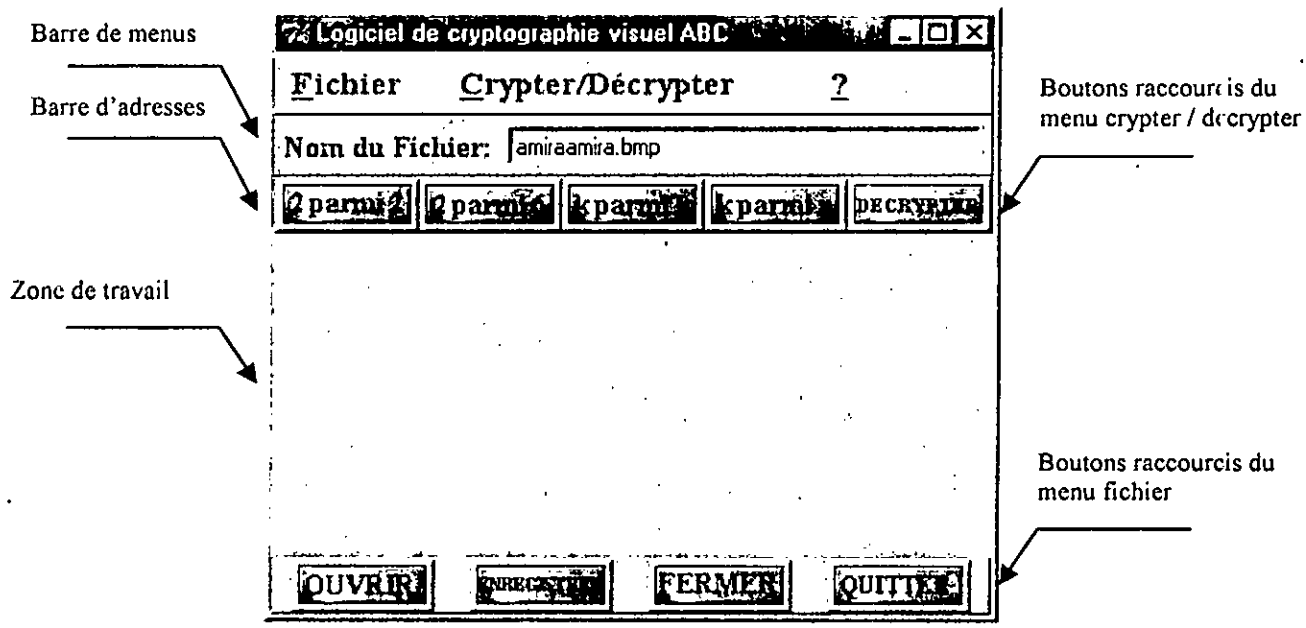


Figure 2.1 Interface graphique

Comme l'illustre la figure 2.1 l'espace de notre interface se compose de 5 zones :

- Une barre de menus,
- Une barre d'adresse,
- Une zone de travail,
- Des boutons raccourcis du menu crypter/décrypter,
- Des boutons raccourcis du menu fichier.

Au cours de l'exécution de notre logiciel, un nombre variable de fenêtres s'afficheront à l'écran et disparaîtront. Selon le choix de k et n on obtiendra :

- L'image originale
- n transparents
- L'image reconstruite

L'utilisation de fenêtres indépendantes est préférée afin de comparer l'image reconstruite à l'image originale ; étant donné aussi que la taille de l'image reconstruite varie selon le schéma de cryptage choisi.

Nous allons décrire chaque zone et donner la fonction de chacune des touches ou menus.

2.2.1 Barre de menus

La barre de menus de notre logiciel est composée de 3 boutons menus déroulants :

- Fichier
- Cryptage/Décryptage
- ?

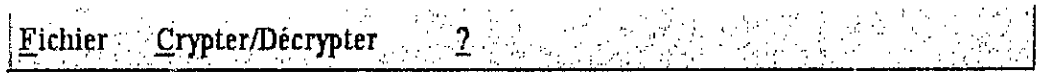


Figure 2.2 : Barre de menus

a) Menu fichier

Le menu déroulant fichier est composé de 4 commandes :

- Ouvrir
- Enregistrer
- Fermer
- Quitter

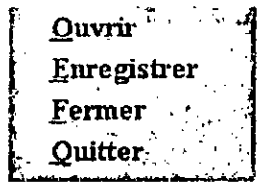


Figure 2.3 : Options du menu fichier

a.1 Ouvrir

Cette commande permet d'importer une image du disque, de la charger et de l'afficher dans la fenêtre image originale. L'encryptage concerne les fichiers Bitmaps (.bmp) et TIF(.tif), de préférence des images monochromes ou même en niveaux de gris car le logiciel les convertira, les affichera et les traitera comme des images noir et blanc. Il résulte de la commande Ouvrir la boite de dialogue suivante :

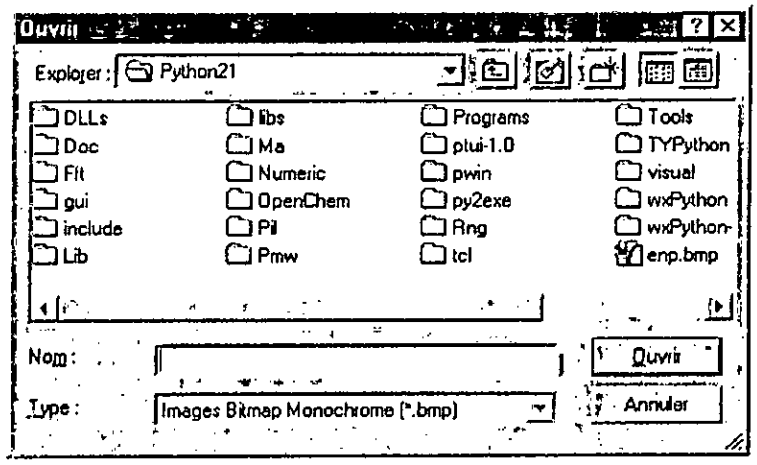


Figure 2.4 : Boite de dialogue ouvrir

a.2 Enregistrer

Cette commande permet d'enregistrer dans le disque dur, l'image reconstruite et les différents transparents qui ont servi à crypter l'image originale. Les extensions

indiquant le numéro de chaque transparent et le type de l'image reconstruite, sont assurées par le logiciel.

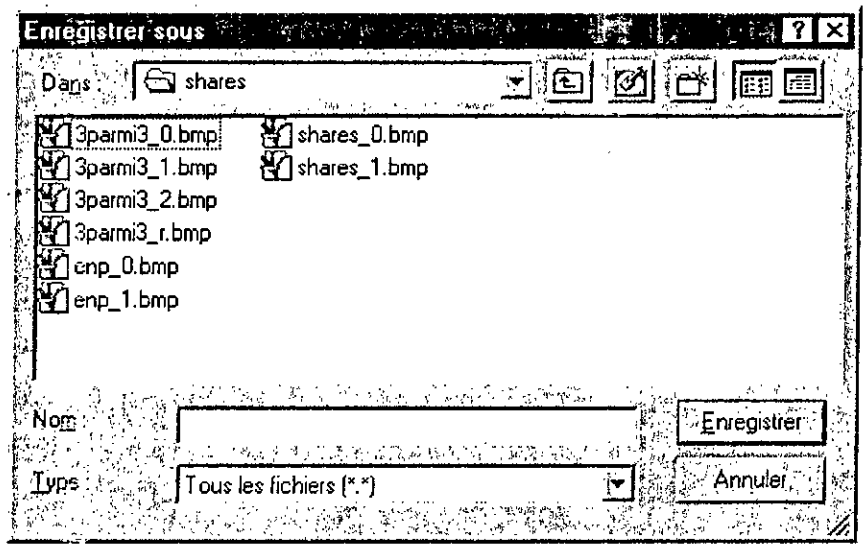


Figure 2.5 : Boite de dialogue enregistrer

a.3 Fermer

Cette commande permet la fermeture des transparents et de l'image reconstruite afin de procéder à un autre schéma de cryptage ou de changer d'image originale.

a.4 Quitter

Cette option permet de quitter le logiciel.

Remarque : les pointillés sous les menus servent à détacher les menus

b) Menu cryptage/décryptage

Ce menu déroulant comporte 2 sous-menus : (crypter et décrypter) selon que l'on désire crypter une image originale ou décrypter une image au préalable cryptée :

Figure 2.6 : Menu cryptage/décryptage

b.1 Crypter

Le sous-menu crypter comporte à son tour 2 sous-menus selon la technique de cryptage choisie k parmi ou k parmi n:

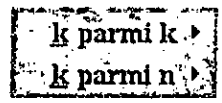


Figure 2.7 : Sous-menu crypter

Chacun des sous-menus comporte 2 ou 3 options selon le choix d'un cas particulier de cryptage ou la méthode générale :

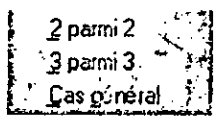


Figure 2.8 : Option du sous-menu k parmi k

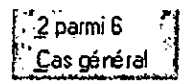


Figure 2.9 : Option du sous-menu k parmi n

- L'option 2 parmi 2 permet un cryptage direct selon la méthode 2 parmi 2 exposée dans le cas particulier du chapitre 4 de la partie II, tel que l'on divise le pixel en 4 subpixels au lieu de 2 pour éviter la distorsion.

- L'option 2 parmi 6 permet un cryptage direct selon la méthode 2 parmi 6 explicitée dans la deuxième construction 2 parmi n de la section concernant les cas particuliers du chapitre 4 de la partie II de ce document.

- L'option k parmi k permet un cryptage direct selon la méthode générale k parmi k après ouverture d'une boite de dialogue.

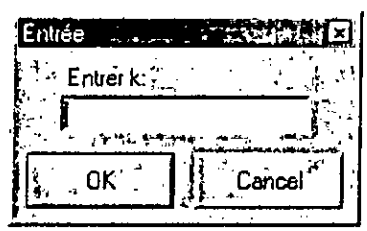


Figure 2.10 : Boite de dialogue avec champs d'entrée pour le seuil k

- La commande k parmi n permet un cryptage direct selon la méthode k parmi n. Tout d'abord, une boite de dialogue se déclenche pour l'entrée du paramètre n (nombre de transparents ou participants) puis pour l'entrée du seuil k comme précédemment.

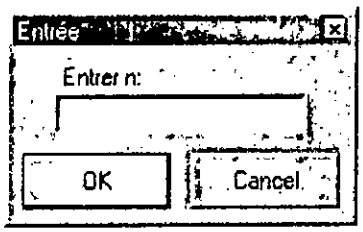


Figure 2.11 : Boite de dialogue d'entrée n

b.2 Décrypter

La commande décrypter déclenche l'ouverture d'une boite de dialogue permettant de choisir la liste de transparents à superposer lors du décryptage d'une image déjà cryptée :

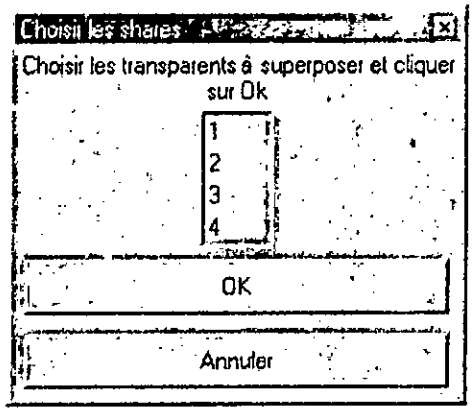


Figure 2.12 : Boite de dialogue de décryptage

c) Menu ?

Le menu ? comporte 2 options :

- Aide
- A propos

c.1 Aide

Cette option permet d'afficher l'aide du logiciel facilitant ainsi son utilisation :

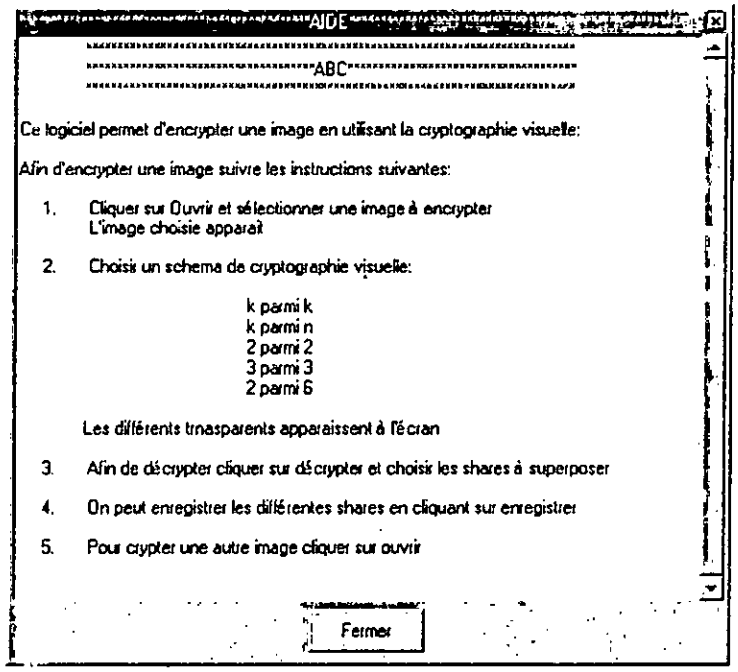


Figure 2.13 : Aide du logiciel

c.2 A propos

Cette option permet l'affichage d'une fenêtre communiquant des informations sur le logiciel :

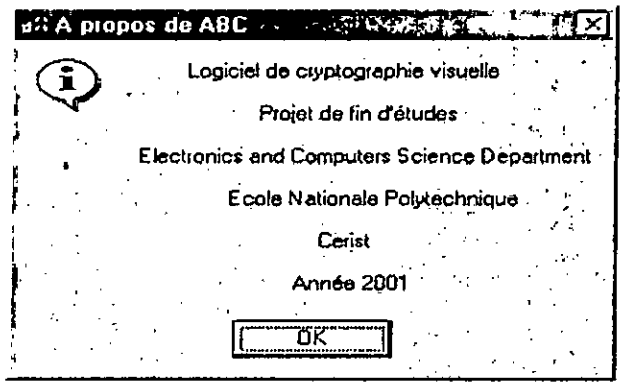


Figure 2.14 : A propos du logiciel

2.2.2 Barre d'adresse

La barre d'adresse permet l'affichage du nom de l'image originale choisie par l'utilisateur.

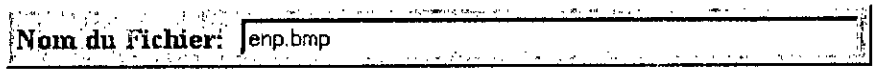


Figure 2.15 : Barre d'adresse

2.2.3 Zone de travail

La zone de travail permet de placer nos images, les visualiser et les comparer.

2.2 4.Boutons raccourcis du menu crypter/décrypter

Ces boutons sont des raccourcis du menu crypter/décrypter et jouent les mêmes fonctions que les commandes des sous-menus du menu crypter décrites précédemment à savoir :

- les k généraux : k parmi k et k parmi n
- les k particuliers : 2 parmi 2, 2 parmi 6

Figure 2.16 : Boutons raccourcis du menu crypter/décrypter

2.2.5 Boutons raccourcis du menu fichier

Ces boutons sont des raccourcis du menu fichier et jouent les mêmes fonctions que les options ouvrir, enregistrer, fermer et quitter décrites précédemment.

Figure 2.17 : Boutons raccourcis du menu fichier

2.3 Schématisation des étapes de traitement essentielles

Afin d'expliquer l'implémentation de notre logiciel, on a opté pour une représentation en blocs simplificatrice de l'ordre logique et chronologique du traitement.

Etant donné l'utilisation d'un langage à script orienté objet, notre programme à caractère modulaire a été divisé en quatre modules essentiels.

Afin d'expliquer l'implémentation de notre logiciel, on a opté pour une représentation en blocs simplificatrice de l'ordre logique et chronologique du traitement.

Etant donné l'utilisation d'un langage à script orienté objet, notre programme à caractère modulaire a été divisé en quatre modules essentiels.

Modules	Fonctions
Vckgui	Interfacage
Ttvck	Acquisition, chargement et construction des images
Nofn	Implémentation de la méthode k parmi k
Nofn	Implémentation de la méthode k parmi n

L'interaction entre ces modules comme leurs fonctions seront schématisées ci-jointement par les diagrammes suivants.
Le modèle des schémas est inspiré de la logique de la programmation orientée objet :

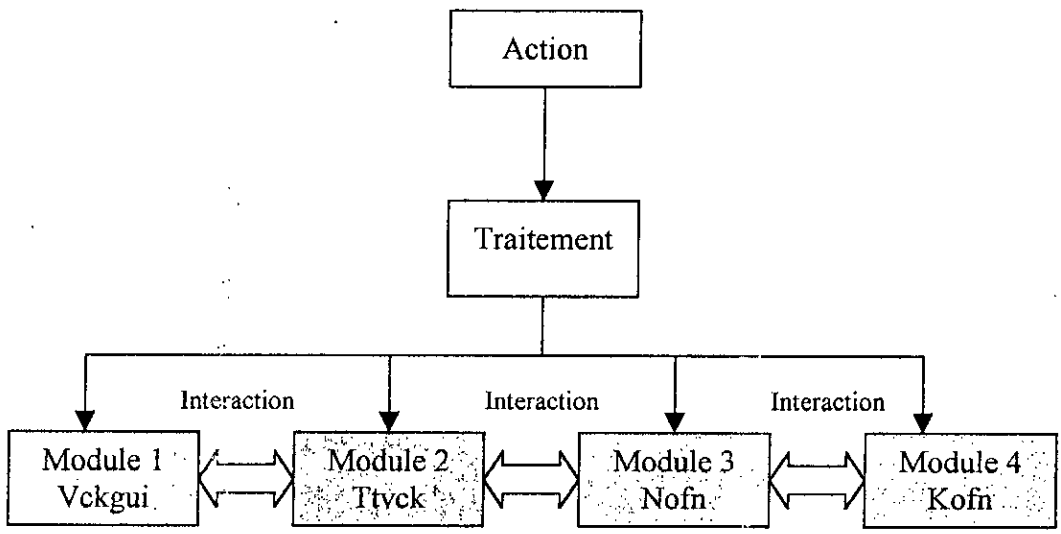


Figure 2.18 : Modèle de la schématisation de l'algorithme

Le module 'gui' concernant la définition de l'interface et l'interaction du programme avec l'utilisateur, s'est essentiellement reposé sur les méthodes de la librairie Tkinter qui est une librairie de classes basé sur 'Tk Toolkit' développée par John Ousterhout pour son langage TCL distribué par Sun Microsystems. Tk est l'une des librairies graphiques les plus utilisées sous Unix/X-Window

Le module 'acc', concernant le chargement des images, leur acquisition, leur codage, leur construction et leur sauvegarde s'est essentiellement reposé sur la librairie PIL (Python Imaging Librairies). Cette librairie rajoute à l'interpréteur Python de puissantes capacités et outils de traitement de l'image.

Cette librairie d'image a été conçue afin d'assurer un accès rapide aux données sauvegardées sous une forme à base de pixels.

La librairie PIL contient aussi certaines fonctionnalités basiques de traitement d'images, incluants les opérations sur les points, le filtrage, et les conversions des espaces de couleur.

point

Cette librairie supporte aussi le dimensionnement des images, les rotations, et les transformations affines sur celles-ci.

Nous allons donc décrire la plupart des commandes du menu Fichier et du menu Crypter/Décrypter, dans l'objectif de tracer les lignes directrices de notre implémentation software des méthodes de cryptage visuel sous une forme de boite à outils.

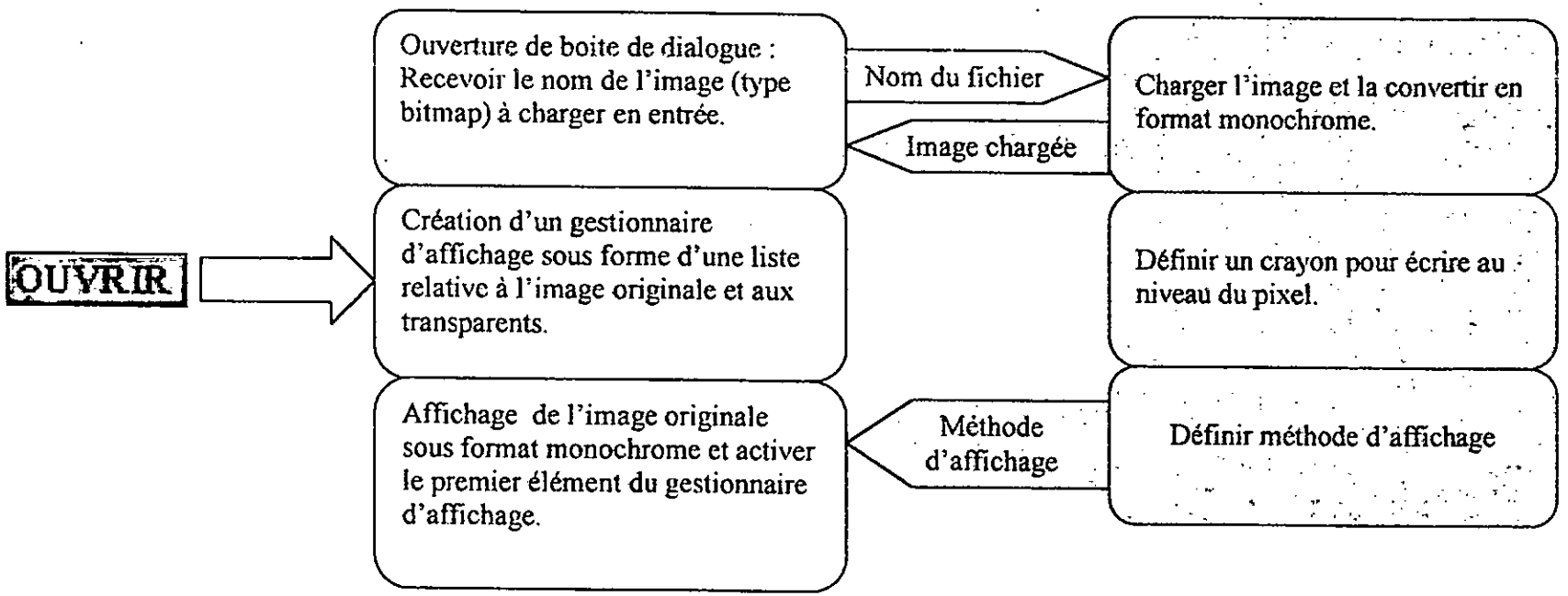


Figure 2.20 : Bloc diagramme de la commande ouvrir

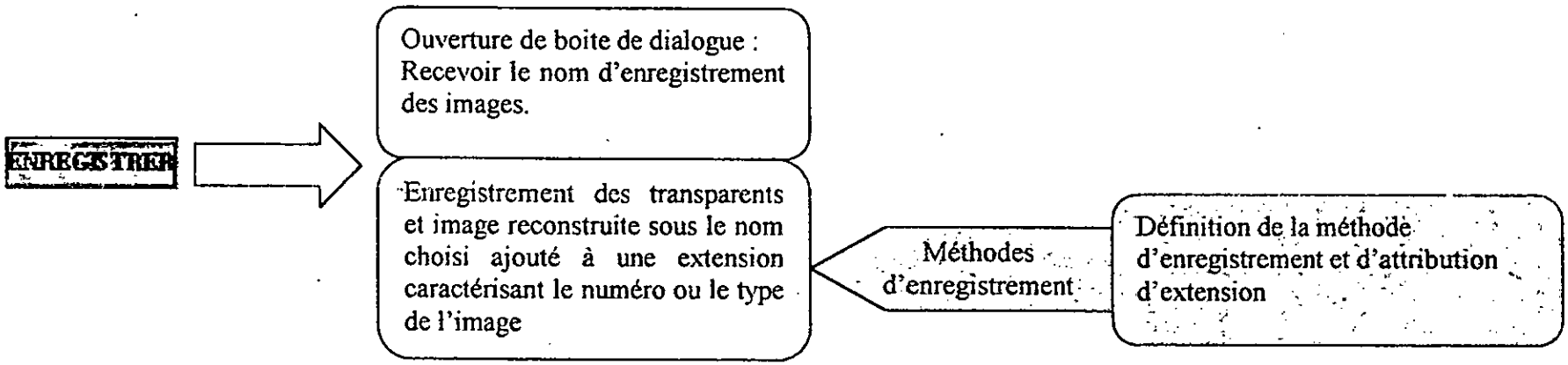


Figure 2.21 : Bloc diagramme de la commande enregistrer

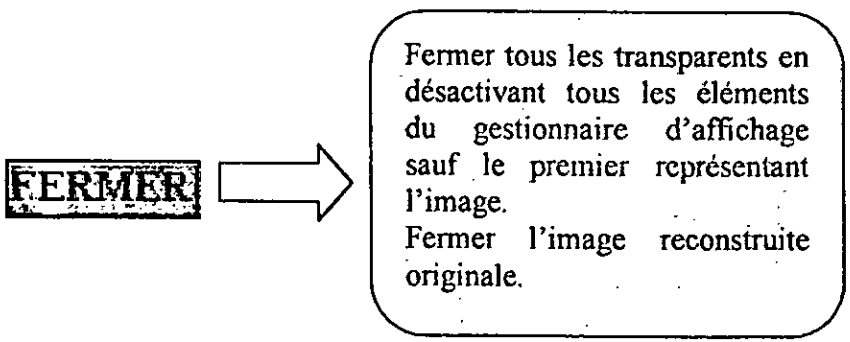


Figure 2.22 : Bloc diagramme de la commande fermer

Figure 2.23 : Bloc diagramme de la commande quitter

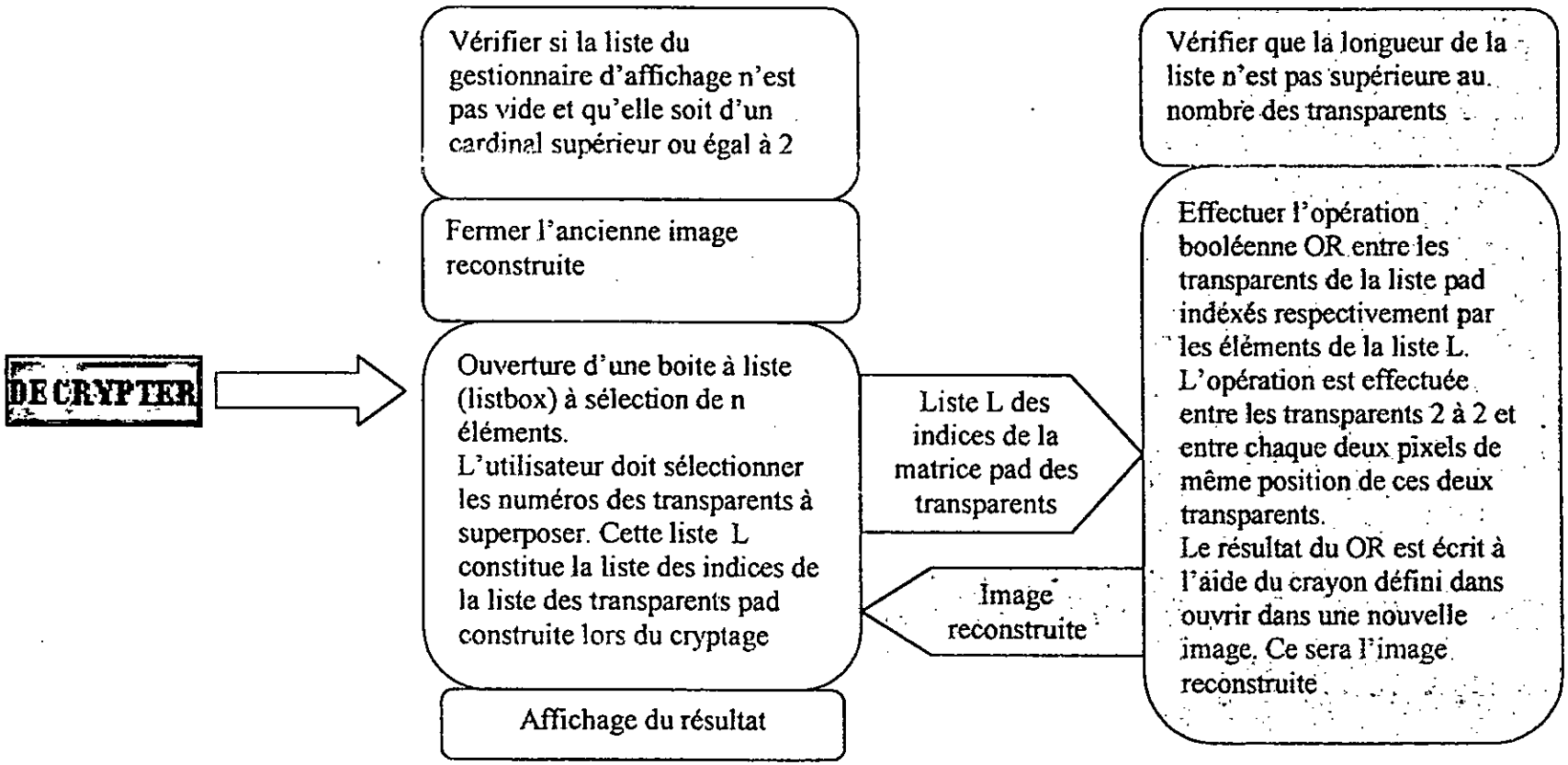


Figure 2.24 : Bloc diagramme de la commande décrypter

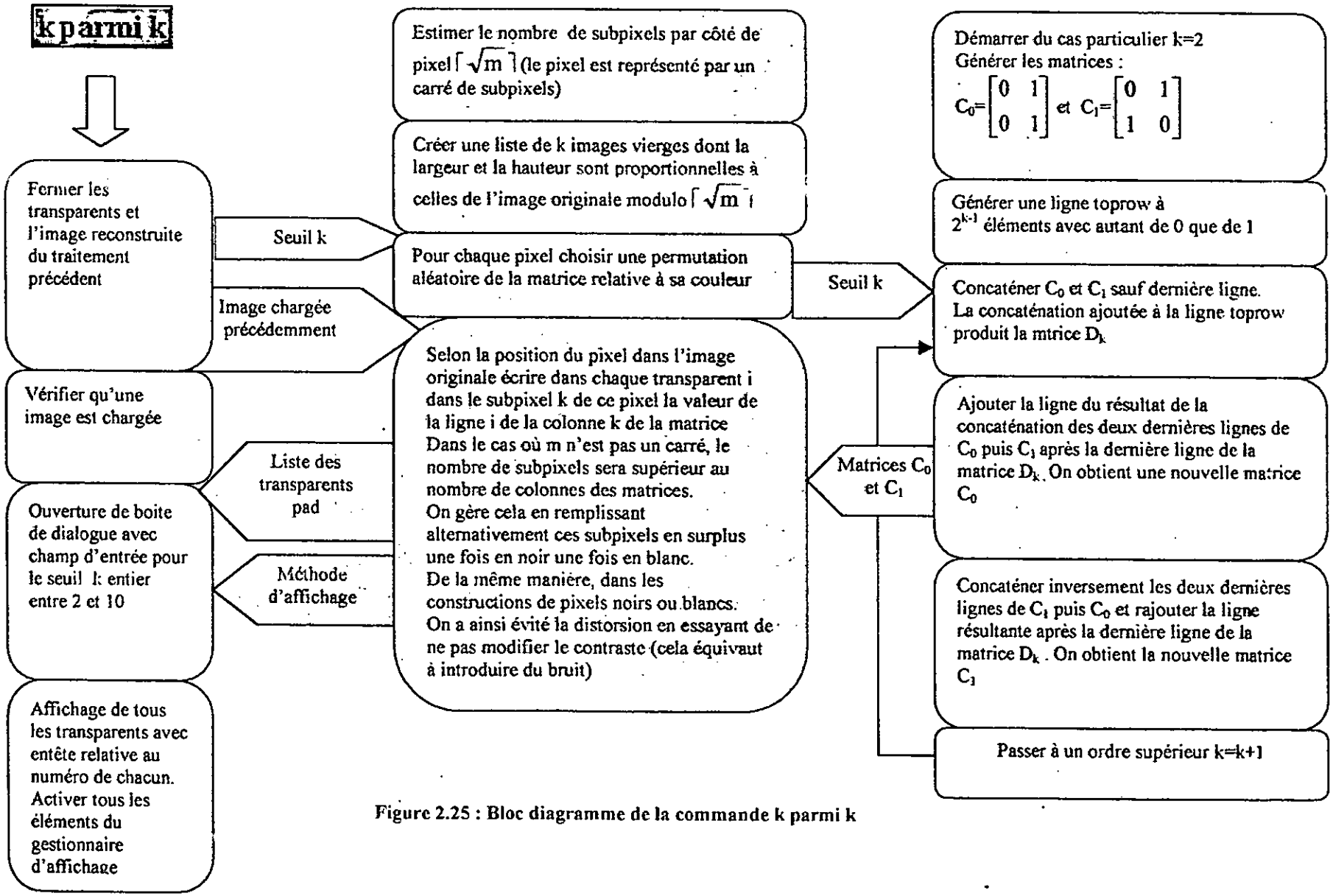


Figure 2.25 : Bloc diagramme de la commande k parmi k

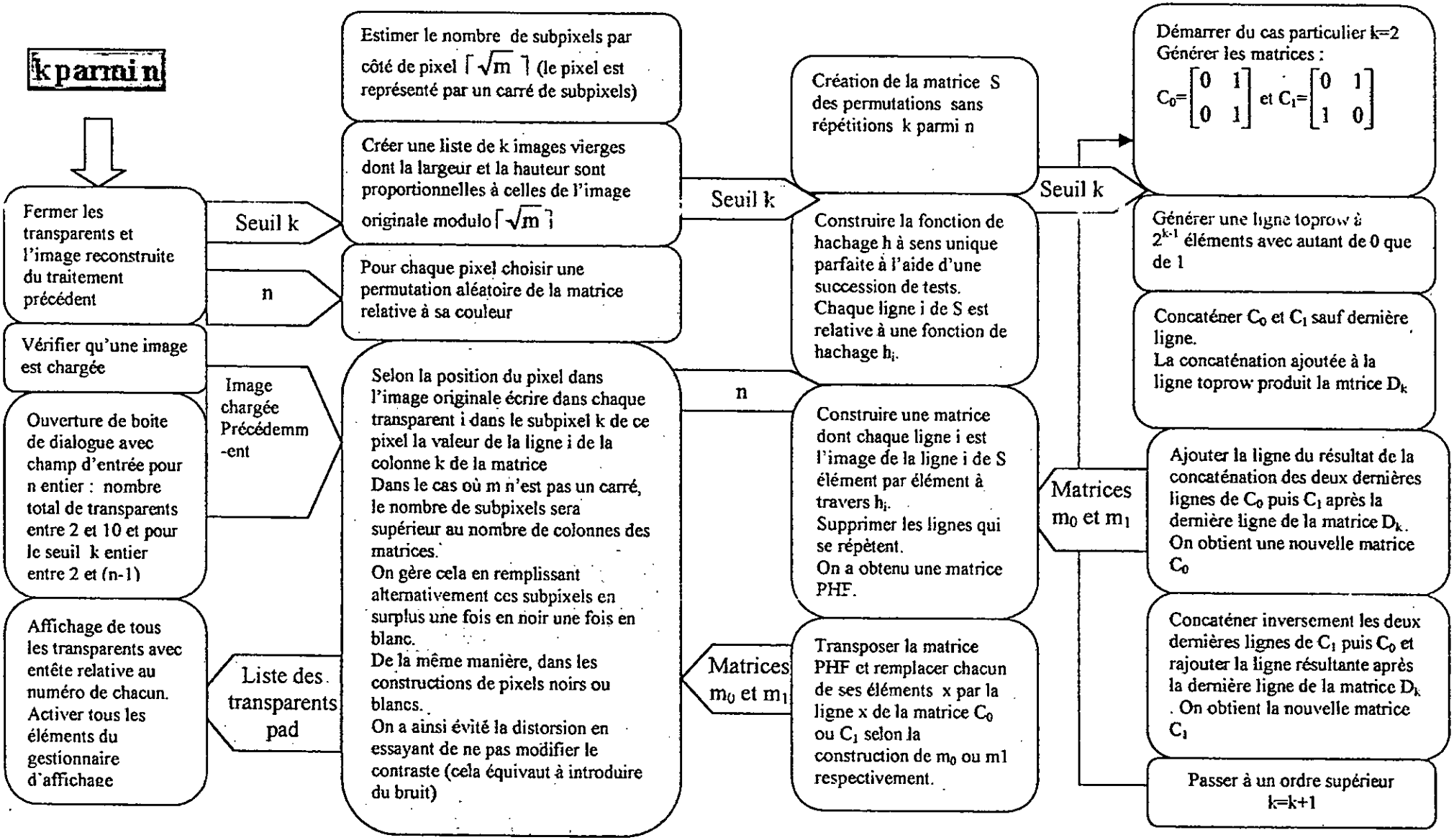


Figure 2.26 : Bloc diagramme de la commande k parmi n

A travers les blocs précédents, nous avons décrit la formalisation des méthodes de cryptage visuel pour les cas généraux k parmi n et k parmi k.

Comme on l'a vu lors de l'interfaçage, le logiciel propose aussi des commandes de cas particuliers 2 parmi 2 et 2 parmi 6.

Pour le cas 2 parmi 2, les étapes de cryptage et de décryptage sont similaires à celles du cas général k parmi k ; la différence réside dans la construction des matrices de base C_0 et C_1. Cette opération est inspirée du cas particulier présenté dans le chapitre 4 de la partie 2 et les matrices sont fixées dans le programme au lieu d'être construites, puis permutées autant de fois que possible avec une expansion de pixel de 4.

Ainsi chaque collection de matrices contiendra 6 matrices différentes.

Figure 2.27 : Combinaisons du pixel possibles lors de la construction du cas particulier 2 parmi 2

Quant au cas général k parmi k avec k=2, il en résultera aussi une construction à 4 subpixels étant donné que l'on a choisi un nombre de subpixel de $\lceil \sqrt{m} \rceil^2$ au lieu de m, et m=2= 2^{k-1}, donc $\lceil \sqrt{m} \rceil^2$=4. Dans le prochain chapitre, nous essayerons d'évaluer ces deux méthodes qui paraissent similaires et en comparer les résultats.

L'arrondissement ou l'approximation élevant le nombre de subpixels à $\lceil \sqrt{m} \rceil^2$, ne concerne que les schémas k parmi k, tel que 2^{k-1},n'est pas un carré.

Le cas 2 parmi 6 est un cas particulier inspiré de la construction 2 parmi n présenté dans la première section du chapitre 4 de la deuxième partie.

Ce schéma présente les mêmes étapes d'exécution que celles du k parmi n, en ce qui concerne les modules 'gui' et 'acc' ; quant à la partie construction des matrices conventionnellement exécutée par les modules 'npn' et 'kpn', elle sera substituée par une classe de 'acc' qui retourne les matrices constantes suivantes :

$$C_0 = \left\{ \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \right\}$$

$$C_1 = \left\{ \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \right\}$$

Cette construction possède un contraste de ¼ ; les constructions 2 parmi n sont celles qui ont donné le plus de résultats dans l'amélioration du contraste et c'est l'une des raisons pour lesquelles il a été proposé le plus de constructions. [15] [20] [21][22]

Dans la section suivante nous comparerons ce cas particulier au cas général k parmi n, tout en sachant d'avance que les méthodes de construction sont différentes.

2.4 Synthèse

A travers ce chapitre nous avons décrit et présenté le logiciel d'implémentation des méthodes de cryptage visuel réalisé dans le cadre de ce mémoire. Celui-ci a été présenté à partir d'une double perception : le point de vue utilisateur et le point de vue concepteur.

Cette implémentation a pour but de mettre en pratique la théorie du cryptage visuel à travers quelques méthodes sélectionnées et quelques cas particuliers qui relèveront les difficultés et problèmes du cryptage visuel dans la pratique.

Parmi ces problèmes la perte en contraste et en résolution est clairement observable dans les images reconstruites. Quant aux difficultés nous avons tenté d'éviter celle de la distorsion par une approximation à la programmation ; ce qui concorde que le formalisme mathématique ne peut pas toujours être respecté en pratique.

Etant donné la coexistence de plusieurs paramètres et considérations dans les schémas de cryptage visuel la question qui se pose est la suivante :

« Quelle sera l'influence d'une telle hypothèse sur nos constructions ? »

Les résultats pratiques pourront nous apporter une réponse à cela et une évaluation plus concrète de notre réalisation.

CHAPITRE 3 Résultats et tests

3.1 Introduction

Après la présentation de notre interface graphique et des différentes commandes, prenons des exemples pour tester notre logiciel. Ces essais seront effectués sur un micro-ordinateur ayant la configuration : fréquence de 200Mhz, 48 Mo de RAM et 8 Go de disque dur. La présentation de ces tests a pour but de relever les difficultés et problèmes de la pratique qui ne constituera, en fait, qu'un aspect de l'évaluation de notre logiciel.

Considérons une image monochrome du type bmp et encryptons la suivant les différentes techniques proposées.

3.2 Image originale

Soit l'image originale suivante :

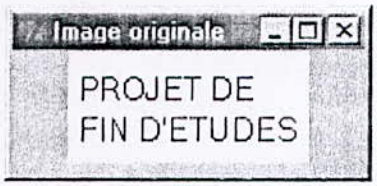


Figure 3.1 : Image originale

3.3 Cas k parmi k

3.3.1 Cas k=2

a) Cas général

On obtient les deux transparents et l'image reconstruite suivants :

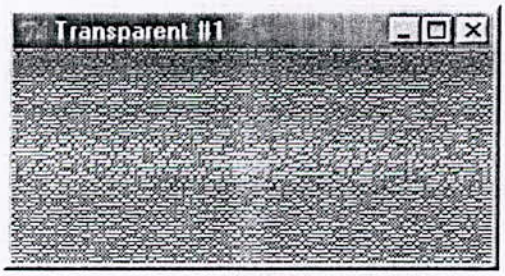


Figure 3.2 : Transparent 1 (k=2 général)

Figure 3.3 : Transparent 2 (k=2 général)

Figure 3.4 : Image décryptée (k=2 général)

Les matrices de base du schéma 2 parmi 2 selon la méthode implémentée pour les constructions k parmi k (cas général) sont les suivantes :

$$\{C_0 = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}\}, \{C_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\}.$$

Les pixels ont la structure suivante :

Figure 3.5 : Structure des pixels 2 parmi 2 : cas général (théorique)

Cette construction provoque une distorsion dans l'image reconstruite. Cette situation se retrouve dans tous les cas où l'expansion du pixel m n'est pas un carré. On essaie alors de se ramène au cas où le pixel est représenté par un carré de subpixels.

Dans notre implémentation nous commencerons par une approximation de l'expansion de pixel et nous l'étendons à $\lceil \sqrt{m} \rceil$ en ajoutant alternativement du noir et du blanc aux pixels supplémentaires.

Dans le cas 2 parmi 2 on se ramène au cas où l'expansion de pixel est de 4 et on obtient les constructions de pixel suivantes selon les matrices choisies :

Matrices de départ (2×2)	Matrice résultante (4×4)	Structures des pixels
$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$	
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}$	

Figure 3.6 : Construction d'un pixel noir

Matrices de départ (2×2)	Matrice résultante (4×4)	Structures des pixels)
$\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}$	
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$	

Figure 3.7 : Construction d'un pixel blanc

On n'effectuera pas de permutations sur les matrices 2×4 mais seulement sur les matrices 2×2 ce qui augmente la probabilité du choix aléatoire des matrices qui sera de ½ au lieu de 1/6 (dans le cas de la permutation de la matrice 2 ×4). Ce qui réduit donc la sécurité. Voyons maintenant les résultats de la construction du cas particulier et comparons-les.

b) Cas particulier

On obtient les deux transparents et l'image reconstruite suivants :

Figure 3.8 : Transparent 1 (k=2 particulier)

Figure 3.9 : Transparent 2 (k=2 particulier)

Figure 3.10 : Image décryptée (k=2 particulier)

Les matrices de base utilisées dans ce schéma sont des matrices 2×4 qui utilisent les constructions de pixels suivantes :

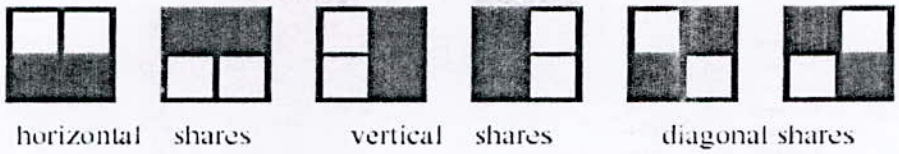


Figure 3.11 : Structure des pixels 2 parmi 2 : cas particulier

On obtient deux collections de 6 matrices C_0 et C_1, résultantes de toutes les permutations possibles des matrices de base :

$$C_0 = \left\{ \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \right\}, C_1 = \left\{ \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \right\}$$

Les constructions de pixels reconstruits résultantes de ce choix de matrices sont les suivantes :

Pixel reconstruit blanc	
Pixel reconstruit noir	

Figure 3.12 : Construction d'un pixel blanc et noir

La probabilité de choix d'une matrice d'encodage sera donc de 1/6.

c) Comparaison

On remarque que le résultat de la reconstruction des transparents dans le cas particulier est visiblement meilleur que dans le cas général et plus précisément en ce qui concerne le contraste.

L'image reconstruite dans le cas général est plus éclaircie et estompée ce qui s'explique par le fait que les pixels noirs reconstruits dans cette construction contiennent un subpixel blanc. Autrement dit, le poids de Hamming du vecteur résultant d'une opération OU des deux lignes de la matrice de base noire est de 3 alors qu'il est de 4 dans la construction du cas particulier.

Un calcul simple du contraste selon la définition de Naor-Shamir donnera un ¼ pour le cas général et ½ pour le cas particulier.

On rajoute à cela que la probabilité de choix aléatoire d'une matrice parmi les 6 sera plus faible pour le cas particulier. Ce qui est préférable pour augmenter la probabilité d'échec d'une attaque. Aussi cette dernière construction paraît plus sûre.

3.4 Cas k parmi n

3.4.1 Cas où n=4, k=2

On obtient les transparents et images reconstruites suivantes :

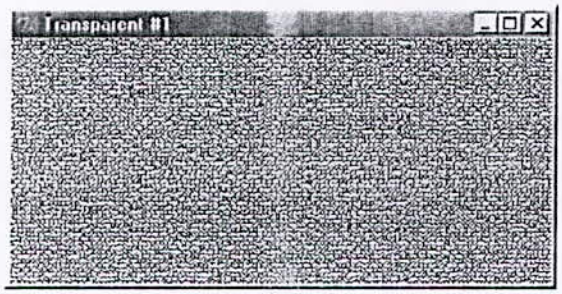


Figure 3.13 : Transparent 1 (n=4, k=2, général)

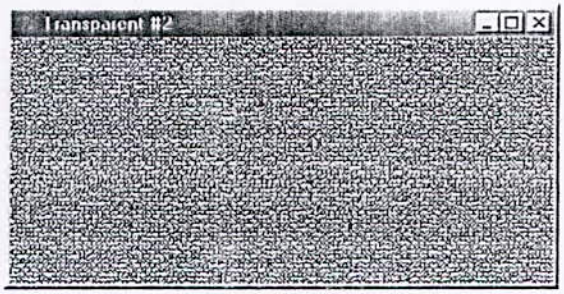


Figure 3.14 : Transparent 2 (n=4, k=2, général)

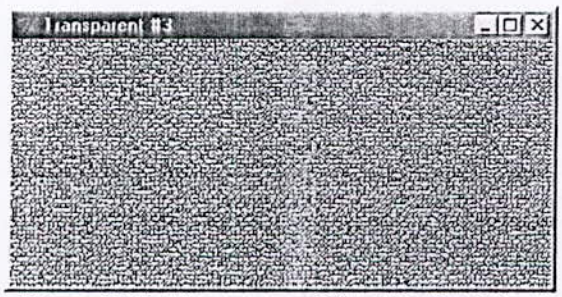


Figure 3.15 : Transparent 3 (n=4, k=2, général)

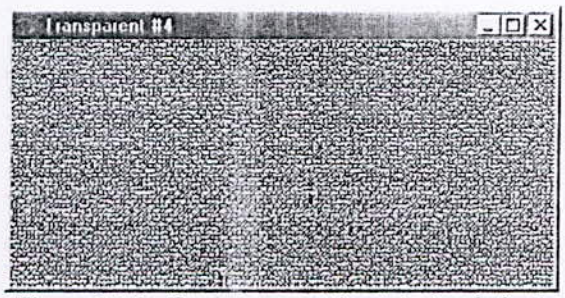


Figure 3.16 : Transparent 4 (n=4, k=2, général)

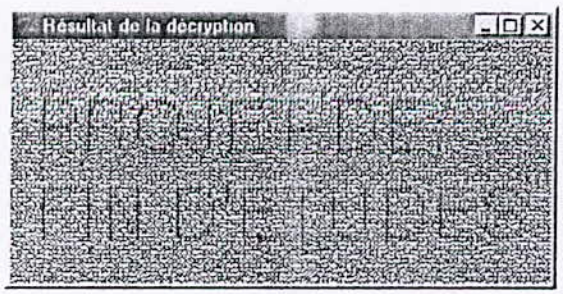


Figure 3.17 : Image décryptée avec les transparents 2 et 3 (n=4, k=2, général)

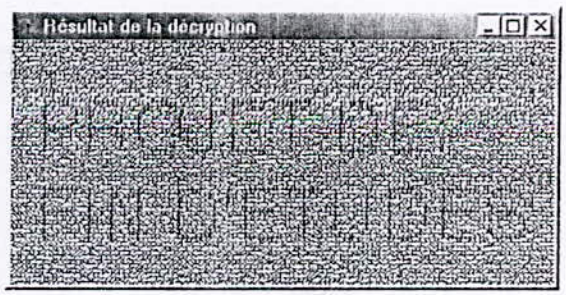


Figure 3.18 : Image décryptée avec les transparents 2, 3 et 4 (n=4, k=2, général)

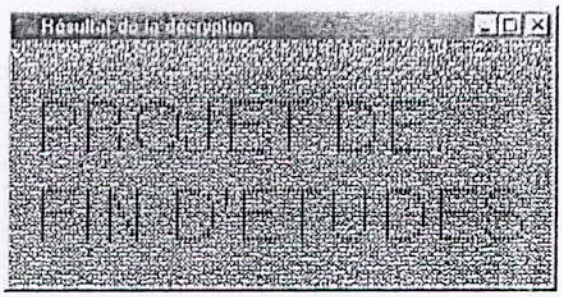


Figure 3.19 : Image décryptée avec les 4 transparents 1 (n=4, k=2, général)

3.4.2 Cas où n=6, k=2

a) Cas général

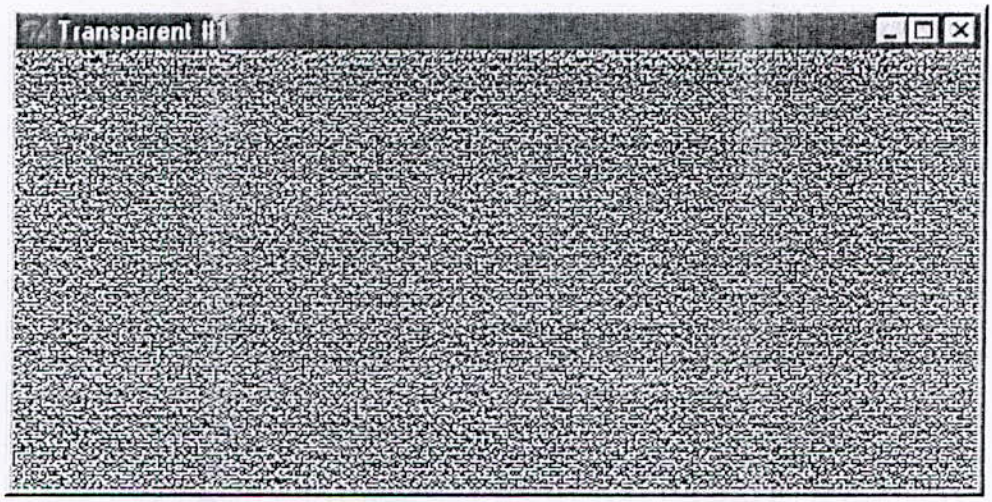


Figure 3.20 : Transparent 1 (n=6, k=2, général)

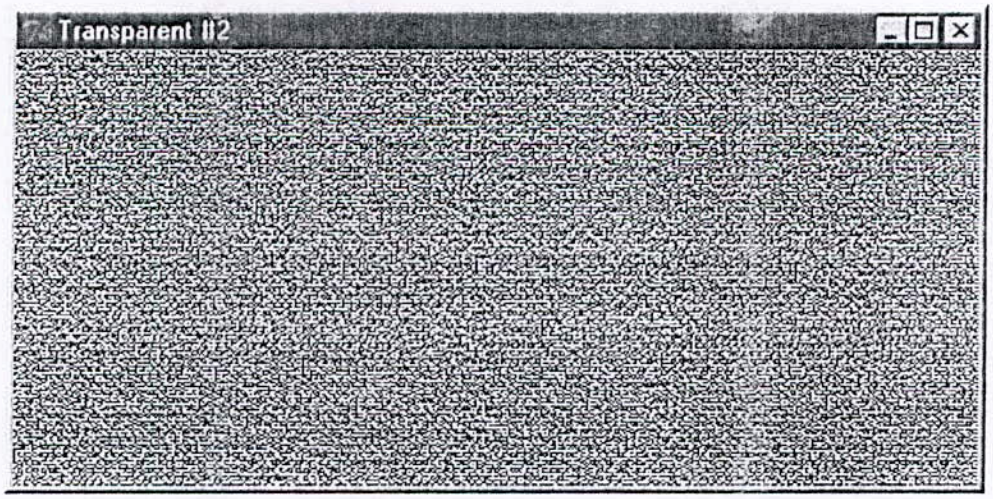


Figure 3.21 : Transparent 2 (n=6, k=2, général)

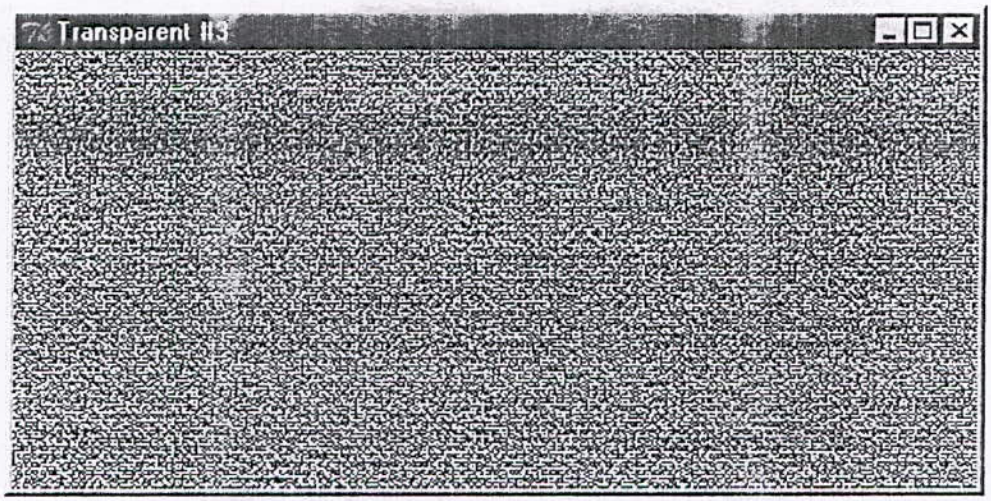


Figure 3.22 : Transparent 3 (n=6, k=2, général)

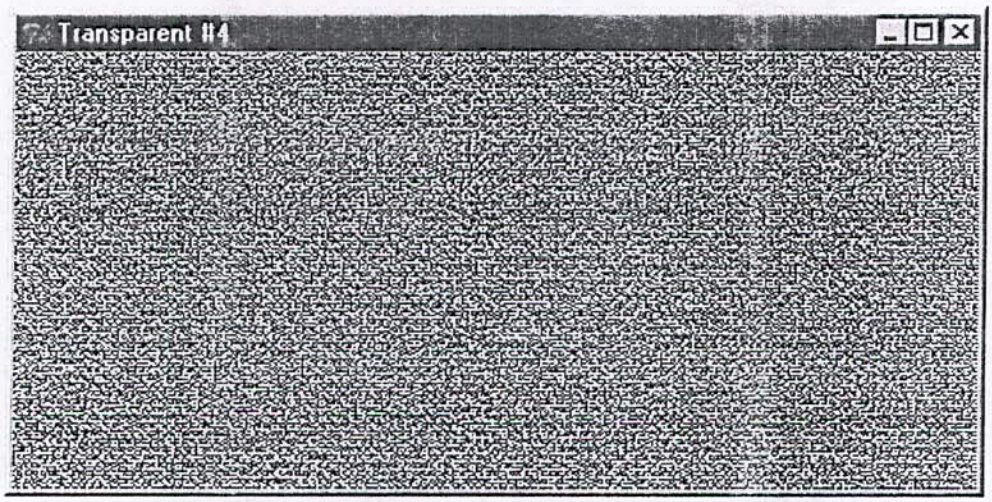


Figure 3.23 : Transparent 4 (n=6, k=2, général)

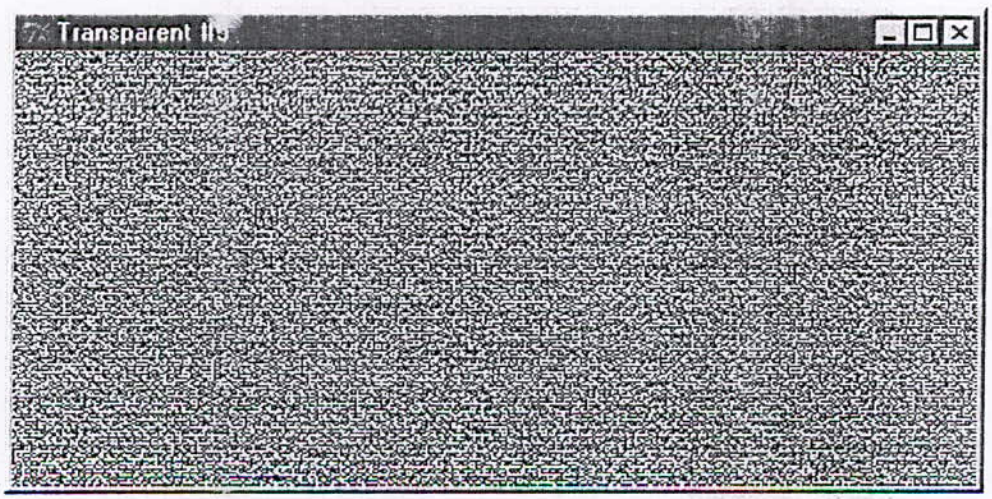


Figure 3.24 : Transparent 5 (n=6, k=2, général)

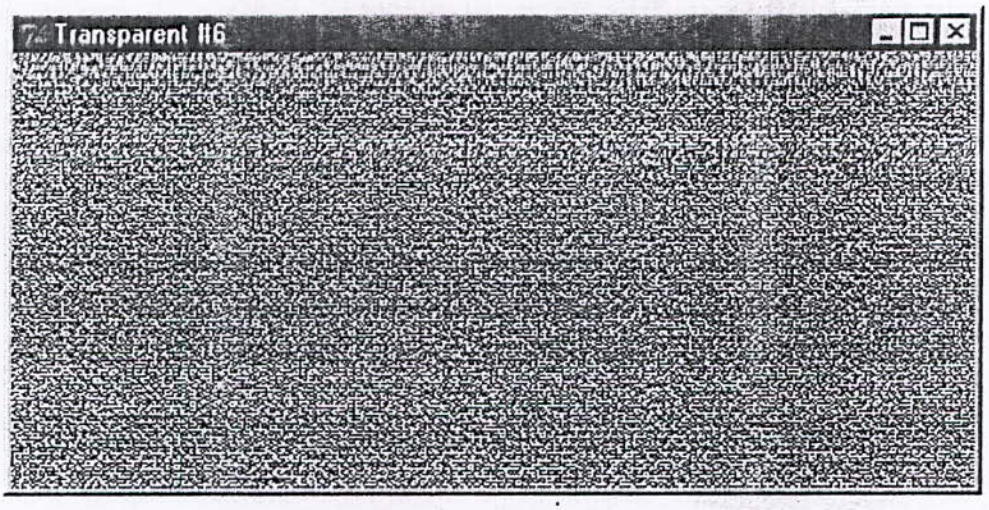


Figure 3.25 : Transparent 6 (n=6, k=2, général)

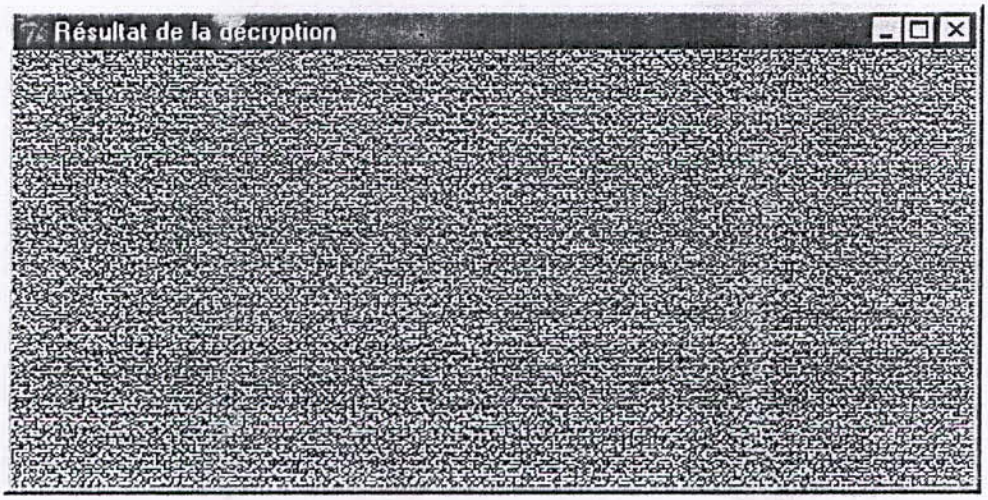


Figure 3.26 : Image décryptée avec les transparents 1 et 2 (n=6, k=2, général)

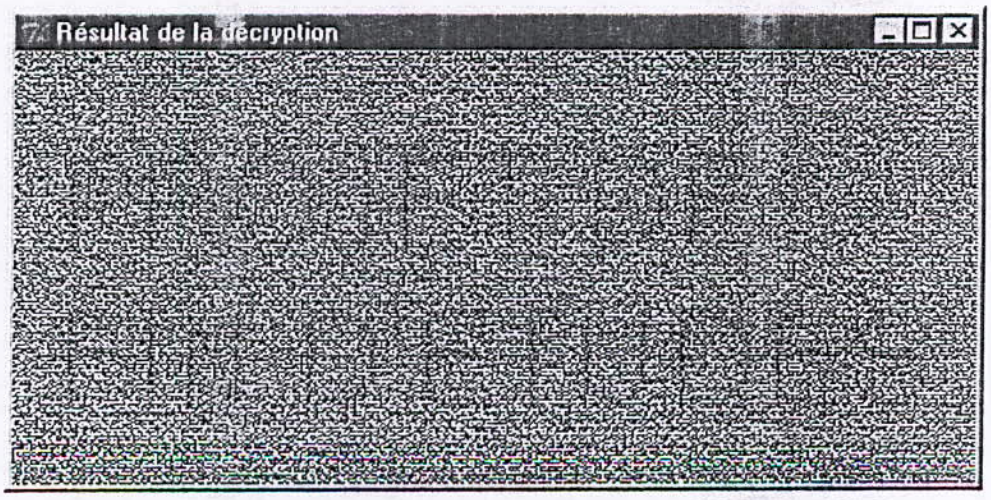


Figure 3.27 : Image décryptée avec les transparents 1, 2 et 3 (n=6, k=2, général)

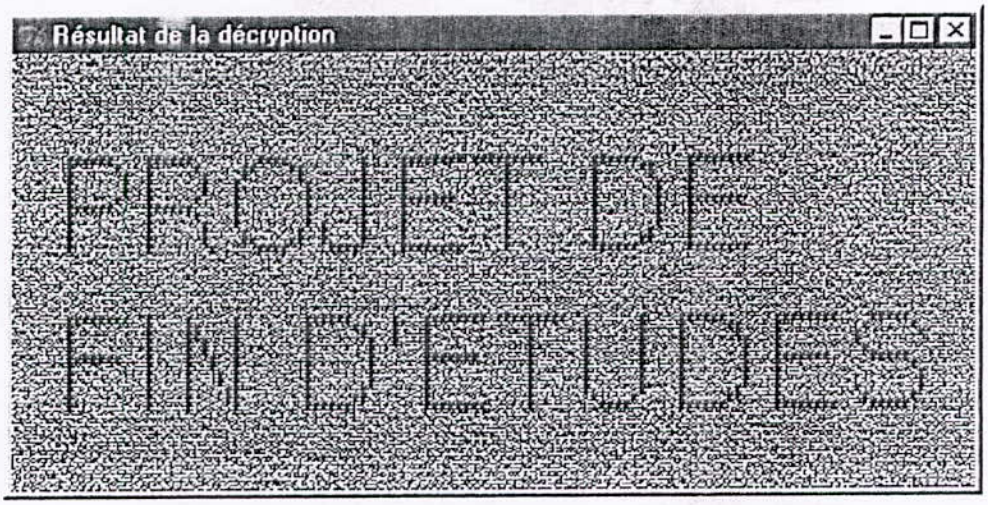


Figure 3.28 : Image décryptée avec les 6 transparents (n=6, k=2, général)

b) Cas particulier

On obtient les transparents et images reconstruites suivantes :

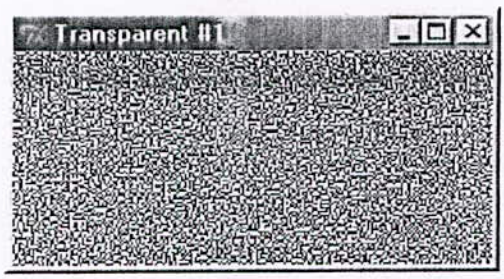


Figure 3.29 : Transparent 1 (n=6, k=2, particulier)

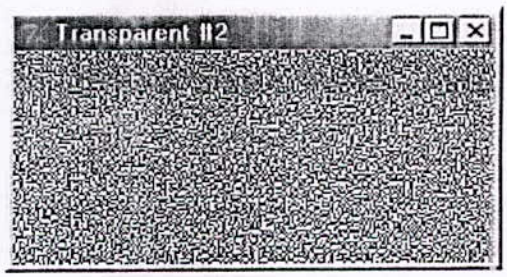


Figure 3.30 : Transparent 2 (n=6, k=2, particulier)

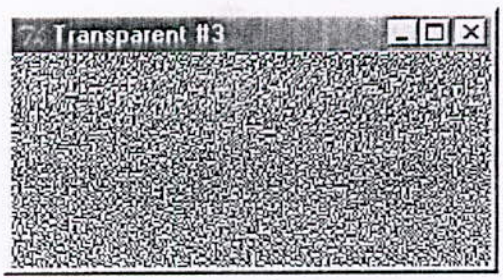


Figure 3.31 : Transparent 3 (n=6, k=2, particulier)

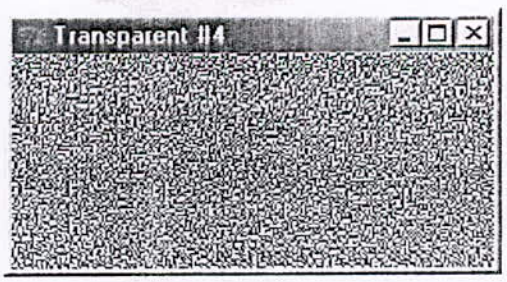


Figure 3.32 : Transparent 4 (n=6, k=2, particulier)

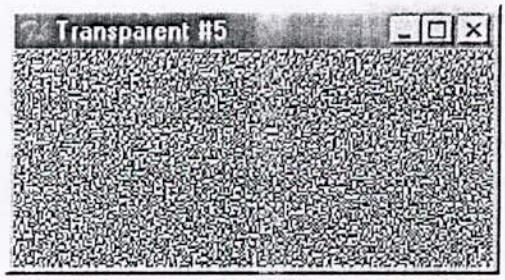


Figure 3.33 : Transparent 5
(n=6, k=2, particulier)

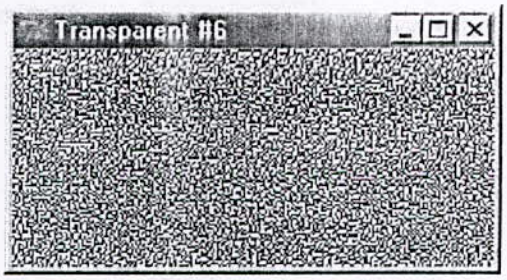


Figure 3.34 : Transparent 6
(n=6, k=2, particulier)

Figure 3.35 : Image décryptée avec les transparents 1 et 2
(n=6, k=2, particulier)

Figure 3.36 : Image décryptée avec les transparents 1, 2 et 3
(n=6, k=2, particulier)

Figure 3.37 : Image décryptée avec les 6 transparents 1 (n=6, k=2, particulier)

3.4.3 Commentaires, comparaisons et discussions

Les schémas de cryptage visuel k parmi n sont construits à partir de la méthode à base d'une fonction de hachage parfaite. Celle-ci est caractérisée par une expansion de pixel très élevée. De plus, l'approximation du nombre de subpixels est faite une fois la matrice de base k parmi n calculée. Le nombre de pixels rajoutés croît en fonction de l'expansion de pixels initiale. Cette situation traduit le fait que notre implémentation est plus efficace pour un nombre de transparents n restreint ce qui est confirmé par la différence entre les résultats des cas 2 parmi 4 et 2 parmi 6.

Ces résultats permettent de constater que la superposition de 2 transparents de la construction 2 parmi 4 reconstitue l'image secrète. Quant au cas 2 parmi 6, il est difficile de percevoir des nuances (ou un contraste quelconque), l'image n'étant pas convenablement reconstituée.

Ce dernier schéma est caractérisé par une perte importante en contraste et en résolution, auxquels est rajouté l'arrondi effectué pour éviter la distorsion.

D'un point de vue théorique, on retrouve que la construction 2 parmi 6 possède initialement une expansion de pixel m=10, notre implémentation lui rajoutant 6 pixels pour constituer un carré de subpixels. L'image originale est alors multipliée par 16.

En parallèle, nous avons implémenté le cas 2 parmi 6 comme cas particulier d'une méthode de construction 2 parmi n énoncée dans le chapitre 4 de la partie II. [9]

Cette méthode se traduit, dans ce cas, par une expansion de pixels de 4. On remarque bien qu'en proportion l'image reconstruite, dans le cas particulier, représente le quart de celle du cas général.

Dans tous les schémas k parmi n, on observe une amélioration du contraste au fur et à mesure de la superposition des transparents.

3.5 Synthèse

L'examen des divers résultats obtenus nous a permis d'observer les principaux inconvénients, ainsi que les difficultés et problèmes liés à la construction de schémas de cryptage visuel. Nous remarquons aisément des pertes en contraste et en résolution ainsi qu'une importante augmentation des dimensions des images reconstruites par rapport aux images originales.

Cependant, nous avons cherché à éviter la distorsion à travers notre implémentation mais parce que les schémas de cryptage visuel comporte un ensemble de paramètres et considérations coexistants et parfois antagonistes. Cette initiative a engendré une augmentation de l'expansion du pixel et un ajout d'informations inutiles dans notre schéma de codage.

Cette augmentation est observée dans les cas où n est relativement important ; ainsi toutes les méthodes présentent des restrictions et des limites. C'est pour cela que nous avons proposé d'autres alternatives à travers des cas particuliers plus appropriés.

En conclusion, disons que le choix d'une méthode de cryptage visuel doit être approprié à la construction voulue ainsi qu'au type de l'image à encrypter. Pour cela, plusieurs méthodes existent et chacune d'elle correspond à une optimisation partielle.

Ce projet de fin d'étude a été entrepris dans le but d'apporter deux contributions : la première relative à la présentation et à la description théorique du paradigme du cryptage visuel et à l'étude de ses protocoles de cryptage ainsi qu'aux constructions des schémas qui y participent. La deuxième contribution consiste à mettre en œuvre une implémentation logicielle exécutant quelques unes des constructions de cryptage visuel, exposées en théorie.

Tout d'abord, la partie théorique a été entamée par une approche essentielle de généralités, afin d'introduire quelques notions de cryptographie classique étant donné que ce projet représente un premier thème dans la discipline de la Cryptologie, entrepris au sein de L'Ecole Nationale Polytechnique d'Alger en collaboration avec le C.E.R.I.S.T.

Dans cette partie introductive nous avons également présenté la notion de partage de secret qui a fait ses preuves en cryptographie conventionnelle et qui figure en tant que concept de base de la cryptographie visuelle. Cette notion répond aux exigences de la société d'information actuelle puisqu'elle est au service des structures de groupe d'organisations et d'entreprises. Les cryptosystèmes de partage de secret sont principalement à seuil mais ils s'étendent aussi à des structures plus générales et satisfont même aux structures prioritaires.

Le deuxième concept de base de la cryptographie visuelle, qui lui permet de la distinguer de la cryptographie classique est le principe de l'exploitation du système visuel de l'homme. Cela favorise la vulgarisation de la cryptographie visuelle, facilite son utilisation et fait d'elle une solution économique de cryptage. Parmi ses avantages figurent :

- Un calculateur classique au cryptage,
- Aucun calculateur n'est exigé pour le décryptage, étant donné que le système visuel humain constitue l'outil de décryptage
- Economie d'apprentissage, étant donné qu'aucun calcul ni pré-requis ne sont exigés au décryptage
- Technologie low-tech, ce qui réduit le coût du cryptosystème et accroît la sécurité si l'utilisateur construit lui-même ses transparents.
- Réduction du temps de décryptage, la perception de l'œil s'exécute instantanément.

Si l'on considère le dernier point et que l'on compare le temps de décryptage du système visuel humain à celui d'une machine exécutant la même opération, on observera que l'homme effectuera instantanément ce qui est équivalent à de nombreuses instructions programmées à l'exécution par la machine et qui nécessiterait des dizaines de minutes ou plus.

On aborde ainsi le problème de la complexité qui ne s'évalue pas par rapport au nombre d'opérations élémentaires comme pour la machine en général. D'ailleurs l'évaluation de la complexité du système visuel n'est pas encore déterminée avec précision, elle fait partie d'un ensemble de capacités visuelles que l'on n'a pas encore réussi à quantifier. Car la sécurité des systèmes de cryptage visuel repose sur la vérification des exigences physiques imposées à l'utilisateur et non pas sur la confidentialité d'une clé ou d'un algorithme. Seulement, la vérification et la quantification de ces grandeurs physiques sont pour la plupart empiriques.

Dans le cas de la complexité, les exigences physiques sont linéaires avec la taille du message et logarithmiques avec la probabilité d'échec du cryptosystème.

Il serait intéressant d'explorer plus en détail cette évaluation de capacités, afin de pouvoir chercher des moyens pour casser le codage, c'est-à-dire effectuer une cryptanalyse du système. Il est normal que le cryptage visuel ait besoin d'être mieux caractérisé pour pouvoir apprécier son degré de confidentialité. En fonction des résultats de la cryptanalyse et de la recherche, des méthodes de codage de deuxième génération, plus complexes, pourraient être développées.

L'exploitation du système visuel humain figure dans la série de recherches visant à explorer les systèmes biologiques et sensoriels de l'homme, afin de les exploiter, les reproduire pour s'en inspirer dans les divers systèmes actuels.

Dans le cas précis de la cryptographie, d'autres voies ont été abordées succédant à la cryptographie visuelle. La première s'intitule cryptographie cérébrale où le décryptage est effectué par le cerveau humain en utilisant la perception 3D du système visuel humain. Quant à la deuxième nouvelle direction, elle s'intitule cryptographie audio, dont le message caché est la parole et les parties du secret sont des morceaux musicaux qui permettent à la fois la cryptographie et la stéganographie ou le data hiding de l'information.

Ainsi, on constate que le cryptage, aujourd'hui, se répand de plus ne plus et son étude reste ouverte au développement et au perfectionnement.

Dans la deuxième partie de notre document, on a essayé de répondre au problème qui vise à établir une relation entre les schémas de cryptage visuel (nombre de participants et seuil) et la construction des parts du secret(expansion de pixel). A travers toutes les constructions décrites, on déduit que le paramètre le plus influent qui décide

de la construction du transparent (expansion de pixel et distribution des subpixels) comme de la qualité de l'image reconstruite (contraste) est le seuil k.

Quant aux caractéristiques des images reconstruites, des efforts sont à chaque fois fournis à travers les schémas développés dans le but de les améliorer. Cependant, le contraste reste, incontestablement, la caractéristique qui nécessite un travail de fond pour, d'abord, le définir et l'évaluer, ensuite, étudier et comparer les schémas proposés pour l'optimiser.

Les techniques de construction des schémas de cryptage visuel sont dotées d'un caractère théorique et possèdent l'abstraction propre aux techniques de codage de l'algèbre et de la géométrie. Elles sont prépondérantes dans nos descriptions des méthodes de construction. Les deux méthodes sélectionnées ont prouvé leur faisabilité à l'implémentation. Il reste à prouver la faisabilité des autres techniques dans de prochaines implémentations.

Nous avons réussi à appliquer quatre techniques de cryptage visuel à seuil, décrites en théorie - deux cas généraux et deux cas particuliers - dans la réalisation d'un logiciel permettant de construire un cryptosystème visuel. Les performances du système sont encore modestes mais le principe existe et a été démontré expérimentalement.

Cette implémentation nous a permis de mieux évaluer les difficultés, les problèmes et les inconvénients du cryptage visuel dans la réalité pratique.

Parmi ces inconvénients figurent la perte en résolution et en contraste ainsi que la distorsion. Cette dernière a été évitée dans notre implémentation par un arrondi, cependant les paramètres caractéristiques sont parfois antagonistes et éliminer un inconvénient peut engendrer un autre. C'est pourquoi le choix de la méthode de construction doit être approprié au schéma visé comme à l'ordre de grandeur des paramètres de la construction k et n ; car chaque méthode offre une alternative d'optimisation différente aux paramètres caractéristiques coexistants.

Un autre inconvénient à prendre en considération concerne l'importante augmentation de la taille de l'image encryptée et de l'image reconstruite par rapport à l'image originale. En effet, l'image est divisée en n transparents où chaque pixel est multiplié par l'expansion de pixel (m). Si l'image originale possède une taille x, la taille du message encrypté sera de $n \times m \times x$. Cela est moins grave que dans la cryptographie classique où le temps de calcul au décryptage dépend de la taille de l'image, car la complexité du système visuel humain ne s'évalue pas de la même manière. En conséquence, on relèvera une augmentation des exigences des capacités physiques qui sont linéaires avec la taille du message. C'est d'ailleurs pour appuyer la différence entre les durées de décryptage de l'homme et de la machine que nous avons implémenté la fonction de décryptage.

Le problème de la multiplication de la taille se poserait dans le cas où l'on voudrait appliquer le cryptage visuel dans un cadre temps réel, comme par exemple transmettre à un débit vidéo en live. Il faudrait transmettre beaucoup plus vite qu'avec un débit vidéo normal.

Quant à la perte en résolution, elle est proportionnelle au nombre de transparents créés. Dans le cas du cryptage visuel des images colorées, la perte en résolution est proportionnelle au nombre de couleurs, ce qui réduit le nombre de couleurs à encrypter.

Les constructions détaillées dans ce document, ainsi que les méthodes implémentées ne concernent que les images en noir et blanc ; cependant le cryptage visuel s'offre à des ouvertures intéressantes pour les images en couleur et en niveaux de gris tout en proposant des alternatives dans la stéganographie (cf. partie II. Chapitre 5).

Ainsi, il serait intéressant de construire une boite à outil, où d'autres techniques et alternatives pourraient être implémentées et rajoutées à celles déjà disponibles.

Une fois le projet finalisé, il pourra être optimisé davantage et écrit dans un langage évolué, afin de réduire le temps d'exécution. De même que sa réalisation pourra être engagée et sa commercialisation envisagée pour vulgariser la cryptographie visuelle, car la cryptographie fait partie, aujourd'hui, des pratiques privées.

Ce travail nous a permis de confirmer que la cryptographie se trouve au confluent de plusieurs disciplines. Le défi à relever consiste alors à trouver les passages et interconnexions des unes vers les autres en assurant la projection de la théorie vers la pratique et vice versa.

LISTE DES FIGURES

PARTIE I

Historique et état de l'art

Chapitre 1

Chapitre 2

PARTIE II

Chapitre 1

Chapitre 2

Chapitre 4

Chapitre 5

PARTIE III

Chapitre 2

Chapitre 3

Bibliographie

[1] Partage d'un logiciel cryptographique de Linux vers Windows NT, C.Pourcelot, Mémoire d'Ingéniorat en informatique au C.N.A.M, 1999.

[2]Cryptologie :Etude des protocoles d'authentification utilisant des précalculs, Gildas Avoine ;DEA Intelligence Artificielle et Algorithmique

[3] La cryptographie Appliquée au Web, Sébastien Hervet, DEA Informatique

[4] Handbook of Applied Cryptography, A.Menezes, P.van Oorschot, S.Vanstone, CRC Press, 1996. Voir www.cacr.math.uwaterloo.ca/hac.

[5]Cryptographie Appliquée, Bruce Schneier ; édition 1996

[6] Key Management , Walter Fumy ; Siemens AG, Dept AUT 961

[7] Visual Cryptographi and Thresold Scheme . (Taking a Look at Secret Sharing), D.Stinson, Doctor Dobb's Journal, April 1998

[8] Visual Authentication and Identification. M. Naor and B.Pinkas,1997,in « advances in cryptology-CRYPTO'97 », B.S.Kaliski Jr.Ed.Vol.1294 of « lecture Notes in Computer Science », Spring Verlag, Berlin, pp.322 336, 1997. Available at Theory of Cryptography Library as ftp://theory.lcs.mit.edu/pub/tcryptol/97-13.ps.

[9] Visual cryptography, Moni Naor and Adi Shamir ; « Advances in Cryptology-Eurocrypt'94 », A.De Santis ED., Vol.950 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, pp.1-12,1995.

[10] Visual Cryptography for General Access Structures ; G.Ateniese,C.Blundo, A .De Santis, and R.Stinson. ICALP'1996,

[11] Contrast Optimal Threshold Visual Cryptography Schemes ; C.Blundo,P.D'Arco, A.De Santis, D.R.Stinson ;1998 ; disponible du Electronic Colloquium on Computational Complexity (TR96-012), via WWW à l'adresse : http://www.eccc.uni-trier.de/eccc/.

[12] Construction and Properties of k out of n Secret Sharing Schemes. 1996. E.Verheul ; henk C.A Van Tilborg, Designs,Codes and Cryptography, Vol.11 (1997) pp.179-196.

[13] Doug Stinson's visual Cryptography Page, Available at http://www.cacr.math.uwaterloo.ca/dstinson/visual.html.

[14]Extended Capabilities for Visual Cryptography.G.Ateniese, C.Blundo, D.R Stinson, 1999, Theoretical Computer Science.

[15] Threshold Visual Cryptography Schemes With Specified Whiteness Levels of Reconstructed Pixels. P.A.Eisen, D.R.Stinson, 1999,

[16] Efficient Colour Visual Encryption or 'Shared Colors of Benetton' . V.Rijmen,B.Preneel présenté lors de l'EUROCRYPT'96 Rome Session ,
http://www.esat.kuleuven.ac.be/~rijmen/vc , disponible aussi
http://www.iacr.org/conferences/ec96/rump/preneel.ps

[17] Colorful Cryptography – A Purely Physical Secret Sharing Scheme based on Chromatic Filters. D.Naccache. Coding and Information Integrity, French- Israeli Workshop, Décembre 1994

[18] Scripting : Higher Level Programming for the 21st Century - John K Ousterhout. IEE computer magazine, March 1998

[19] La révolution du scripting - Bulletin Micro du CNRS - LMB N°75 - Alexandre Ferrieux (CNET) et Olivier Lenormand(CNRS)

[20] Constructions and Bounds for Visual Cryptography , G.Ateniese , C.Blundo , A.De Santis , D.R.Stinson , « 23rd International Colloquium on Automata, Languages and Programming » , ICALP'96 Proceedings , M.Auf Der Heide and D.Moniem eds , Volume 1099 of « Lecture notes
in computer science », Springer – Verlag, Berlin , pp.416-428, 1996

[21] On the Contrast in Visual Cryptography , C.Blundo , A. De Santis, D.R.Stinson , Journal of Cryptography , disponible aussi dans Théorie Of Cryptography Library à l'adresse
ftp://theory.lcs.mit.edu/pub/cryptol/96-13.ps

[22] Contrast Optimal k out of n Secret Sharing Scemes in Visual Cryptography , T.Hofmeister , M.Kraus , H.U.Simon , in « COCOON'97 », T.Jiang and D.T.Lee Eds. , Vol.1276 of « Lecture Notes in Computer Science », Springer – Verlag, berlin, pp. 176-185, 1997

[23]Présentation du langage Python, magazine Programmez !, novembre1999.

[23]A testimonial, Mark Lutz

[24]Glue It All Together With Python, Guido Van Rossum, Paper for the OMG-DARPA-MCC Workshop on compositional Software Architecture in Monterey, California, January 6-8 1998

[25] Questions et réponses au sujet de Python, Magazine Langages et Systèmes, Interview donnée par Olivier BERGER

[26]Comparing Python to other Languages, Guido Van rossum, http://www.python.org

[27] Introduction to tkinter, Frederick Ludth, http://www.pythonware.com, Octobre1999.

[28] The Python imaging Library, Frederick Ludth, ,Avril 2001

[29] Dive Into Python, Mark Pilgrim, 31 May 2001, http://diveintopython.org/.

[30] Notes de cours sur l'apprentissage de la programmation avec Python, Gérard Swinnen, Allen B. Downey, JeffreyElkner, w3.ann.jussieu.fr/escritor/DrAuteur.htm,http://www.gnu.org/copyleft/gpl.html,septembre2000

[31] F. Stajano, VCK: The Visual Cryptography Kit, Available at http://www.foretec.com/python/workshops/1998-11/demosessions /stajano2.html.