Cryptographie et Stockage (*EFS*)

Nawfel LAHRICHI
Christophe GROSMAIRE

EPITA - Promotion 2003

12 mai 2002

Résumé

Le sujet portant sur la cryptologie et le stockage, et au vu du fait que nos connassances sur ces sujets étant relativement limitées, nous avons dans un premier temps éffectué de nombreuses recherches sur différents sites web afin d'être à même de mieux comprendre ce dont on allait parler. Ces recherches nous ont donc permis d'étudier de façon générale la cryptologie et le stockage lié à des systèmes de fichiers. Pour ce rapport, nous nous sommes donc inspiré des nombreux articles que nous avons trouvé durant nos recherches et en avons fait une synthèse.

Table des matières

Première partie

Introduction à la Cryptologie

Chapitre 1

Cryptographie

1.1 Introduction à la Cryptographie

1.1.1 Pourquoi la cryptographie ?

L'homme a toujours ressenti le besoin de dissimuler des informations, bien avant même l'apparition des premiers ordinateurs et de machines à calculer.

Depuis sa création, le réseau Internet a tellement évolué qu'il est devenu un outil essentiel de communication. Cependant cette communication met de plus en plus en jeux des problèmes d'économie des entreprises présentes sur le Web. Les transactions faites à travers le réseau peuvent être interceptées, d'autant plus que les lois ont du mal à se mettre en place sur Internet, il faut donc garantir la sécurité de ces informations, c'est la cryptographie qui s'en charge.

1.1.2 Qu'est ce que la cryptographie ?

Le mot **cryptographie** est un terme générique désignant l'ensemble des techniques permettant de chiffrer des messages, c'est-à-dire permettant de les rendre inintelligibles sans une action spécifique.

La cryptologie est essentiellement basée sur l'arithmétique : Il s'agit dans le cas d'un texte de transformer les lettres qui composent le message en une succession de chiffres (sous forme de bits dans le cas de l'informatique car le fonctionnement des ordinateurs est basé sur le binaire), puis ensuite de faire des calculs sur ces chiffres pour :

- d'une part les modifier de telle faon à les rendre incompréhensibles. Le résultat de cette modification (le message chiffré) est appelé **cryptogramme**
- faire en sorte que le destinataire saura les déchiffrer

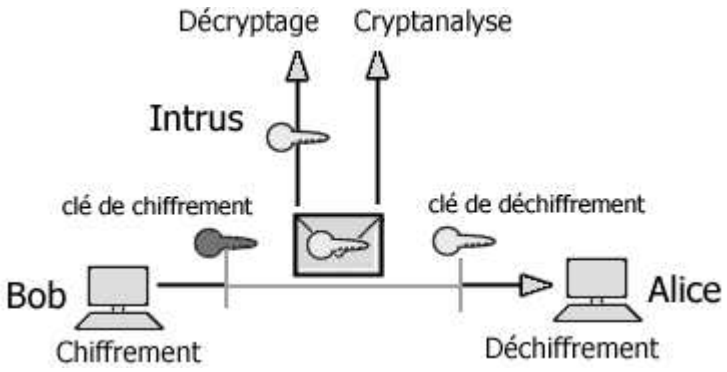

FIG. 1.1 – Chiffrement - Déchiffrement

Le fait de coder un message de telle faon à le rendre secret s'appelle *chiffrement*. La méthode inverse, consistant à retrouver le message original, est appelé *déchiffrement*.

Le chiffrement se fait généralement à l'aide d'une *clef de chiffrement*, le déchiffrement nécessite quant à lui une *clef de déchiffrement*. On distingue généralement deux types de clefs :

- *Les clés symétriques* : il s'agit de clés utilisées pour le chiffrement ainsi que pour le déchiffrement. On parle alors de chiffrement symétrique ou de chiffrement à clé secrète.
- *Les clés asymétriques* : il s'agit de clés utilisées dans le cas du chiffrement asymétrique (aussi appelé chiffrement à clé publique). Dans ce cas, une clé différente est utilisée pour le chiffrement et pour le déchiffrement

On appelle décryptement (le terme de décryptage peut éventuellement être utilisé également) le fait d'essayer de déchiffrer illégitimement le message (que la clé de déchiffrement soit connue ou non de l'attaquant). Lorsque la clef de déchiffrement n'est pas connue de l'attaquant on parle alors de cryptanalyse (on entend souvent aussi le terme plus familier de cassage). La cryptologie est la science qui étudie les aspects scientifiques de ces techniques, c'est-à-dire qu'elle englobe la cryptographie et la cryptanalyse.

1.1.3 Les fonctions de la cryptographie

La cryptographie est traditionnellement utilisée pour dissimuler des messages aux yeux de certains utilisateurs. Cette utilisation a aujourd'hui un intérêt d'autant plus grand que les communications via Internet circulent dans des infrastructures dont on ne peut garantir la fiabilité et la confidentialité. Désormais, la cryptographie sert non seulement à préserver la confidentialité des données mais aussi à garantir leur intégrité et leur authenticité.

1.1.4 La confidentialité

La confidentialité consiste à rendre l'information inintelligible à d'autres personnes que les acteurs de la transaction.

1.1.5 L'intégrité

Vérifier l'intégrité des données consiste à déterminer si les données n'ont pas été altérées durant la communication (de manière fortuite ou intentionnelle).

1.1.6 L'authentification

L'authentification consiste à assurer l'identité d'un utilisateur, c'est-à-dire de garantir à chacun des correspondants que son partenaire est bien celui qu'il croit être. Un contrle d'accès peut permettre (par exemple par le moyen d'un mot de passe qui devra être crypté) l'accès à des ressources uniquement aux personnes autorisées.

1.1.7 La non-répudiation

La non-répudiation de l'information est la garantie qu'aucun des correspondants ne pourra nier la transaction.

Chapitre 2

Chiffrement par substitution

2.1 Cryptage simple

2.1.1 Le chiffrement par substitution

Le chiffrement par substitution consiste à remplacer dans un message une ou plusieurs entités (généralement des lettres) par une ou plusieurs autres entités.

On distingue généralement plusieurs types de cryptosystèmes par substitution :

- La **substitution monoalphabétique** consiste à remplacer chaque lettre du message par une autre lettre de l'alphabet
- La **substitution polyalphabétique** consiste à utiliser une suite de chiffres monoalphabétique réutilisée périodiquement
- La **substitution homophonique** permet de faire correspondre à chaque lettre du message en clair un ensemble possible d'autres caractères
- La **substitution de polygrammes** consiste à substituer un groupe de caractères (polygramme) dans le message par un autre groupe de caractères

2.1.2 Le chiffrement de César

Ce code de chiffrement est un des plus anciens, dans la mesure o Jules César l'aurait utilisé. Le principe de codage repose sur l'ajout d'une valeur constante à l'ensemble des caractères du message, ou plus exactement à leur code ASCII.

Il s'agit donc simplement de décaler l'ensemble des valeurs des caractères du message d'un certain nombre de positions, c'est-à-dire en quelque sorte de sub-

stituer chaque lettre par une autre. Par exemple, en décalant le message "COMMENT CA MARCHE" de 3 positions, on obtient "FRPPHQW FD PDUFKH". Lorsque l'ajout de la valeur donne une lettre dépassant la lettre Z, il suffit de continuer en partant de A, ce qui revient à effectuer un modulo 26. A titre d'exemple, dans le film L'odyssée de l'espace, l'ordinateur porte le nom de HAL. Ce surnom est en fait IBM décalé de 1 position vers le bas...

On appelle clé le caractère correspondant à la valeur que l'on ajoute au message pour effectuer le cryptage. Dans notre cas la clé est C, car c'est la 3^{eme} lettre de l'alphabet.

Ce système de cryptage est certes simple à mettre en oeuvre, mais il a pour inconvénient d'être totalement symétrique, cela signifie qu'il suffit de faire une soustraction pour connatre le message initial. Une méthode primaire peut consister à une bête soustraction des nombres 1 à 26 pour voir si l'un de ces nombres donne un message compréhensible. Une méthode plus évoluée consiste à calculer les fréquences d'apparition des lettres dans le message codé (cela est d'autant plus facile à faire que le message est long). Effectivement, selon la langue, certaines lettres reviennet plus couramment que d'autres (en franais, par exemple, la lette la plus utilisée est la lettre E), ainsi la lettre apparaissant le plus souvent dans un texte crypté par le chiffrage de César correspondra vraisemblablement à la lettre E, une simple soustraction donne alors la clé de cryptage...

2.1.3 Le chiffrage ROT13

Dans le cas spécifique du chiffrement de Jules César o la clé de cryptage est N (13^{eme} lettre de l'alphabet), on appelle ce cryptage ROT13 (le nombre 13, la moitié de 26) a été choisi pour pouvoir crypter et décrypter facilement les messages...).

2.2 Cryptage par tranposition

Les méthodes de chiffrement par transposition consistent à réarranger les données à crypter de telle faon à les rendre incompréhensibles. Il s'agit généralement de réarranger géométriquement les données pour les rendre visuellement inexploitables.

2.2.1 La technique assyrienne

Cette technique de cryptage est vraisemblablement la première preuve de l'utilisation de moyens de chiffrement en Grèce dès 600 avant Jésus Christ pour dissimuler des messages écrits sur des bandes de papyrus.

La technique consistait à :
– enrouler une bande de papyrus sur un cylindre appelé scytale

– écrire le texte longitudinalement sur la bandelette ainsi enroulée (le message dans l'exemple ci-dessus est "comment a marche")

Le message une fois déroulé n'est plus compréhensible ("cecaeonar mt c m mh "). Il suffit au destinataire d'avoir un cylindre de même rayon pour pouvoir déchiffrer le message. en réalité un casseur (il existait des casseurs à l'époque...) peut déchiffrer le message en essayant des cylindres de diamètre successifs différents, ce qui revient à dire que la méthode peut être cassée statistiquement (il suffit de prendre les caractères un à un, éloignés d'une certaine distance).

Chapitre 3

Chiffrement symétrique

3.1 Systèmes à clef secrète

3.1.1 Le chiffrement symétrique

Le **chiffrement symétrique** (aussi appelé *chiffrement à clé privée* ou *chiffrement à clé secrète*) consiste à utiliser la même clef pour le chiffrement que pour le déchiffrement.

Le chiffrement consiste alors à effectuer une opération entre la clé privée et les données à chiffrer afin de rendre ces dernières inintelligibles. Ainsi, le moindre algorithme (tel qu'un OU exclusif) peut rendre le système quasiment inviolable (la sécurité absolue n'existant pas).

Toutefois, dans les années 40, *Claude Shannon* démontra que pour être totalement sr, les systèmes à clefs privées doivent utiliser des clefs d'une longueur au moins égale à celle du message à chiffrer. De plus le chiffrement symétrique impose d'avoir un canal sécurisé pour l'échange de la clé, ce qui dégrade sérieusement l'intérêt d'un tel système de chiffrement.

Ainsi, dans les années 20, *Gilbert Vernam* et *Joseph Mauborgne* mirent au point la méthode du one time pad (traduisez méthode du masque jetable), basée sur une clé privé générée aléatoirement, utilisée une et une seule fois, puis détruite. Ainsi à la même époque le Kremlin et la maison blanche étaient reliés

Fig. 3.1 – Clé privée

par le fameux *téléphone rouge*, c'est-à-dire un téléphone dont les communications étaient cryptées par une clé privée selon la méthode du masque jetable. La clé privée était alors échangée grce à la valise diplomatique (jouant le rle de canal sécurisé).

3.1.2 Les limites du chiffrement symétrique

Le principal inconvénient d'un cryptosystème à clefs secrètes provient de l'échange des clés. En effet, le chiffrement symétrique repose sur l'échange d'un secret (les clés). Ainsi, se pose le problème de la distribution des clés :

Pour un groupe de n personnes utilisant un cryptosystème à clés secrètes, il est nécessaire de distribuer $\frac{n \times (n-1)}{2}$ clés.

Chapitre 4

Chiffrement asymétriqe

4.1 Système à clef publique

4.1.1 le principe du chiffrement à clé publique

Le paradigme de chiffrement à clés publiques est apparu en 1976, avec la publication d'un ouvrage sur la cryptographie par *Whitfield diffie* et *Martin Hellman*.

Dans un cryptosystème asymétrique (aussi appelé cryptosystème à clés publiques), les clés existent par paires (on parle souvent de *bi-clés*) :

- Une clé publique pour le chiffrement
- Une clé secrète pour le déchiffrement

Ainsi, dans un système de chiffrement à clé publique, les utilisateurs choisissent une clé aléatoire dont ils sont seuls connaisseurs (il s'agit de la clé privée). A partir de cette clé, ils déduisent chacun automatiquement un algorithme (il s'agit de la clé publique). Les utilisateurs s'échangent cette clé publique au travers d'un canal non sécurisé.

Lorsqu'un utilisateur désire envoyer un message à un autre utilisateur, il lui suffit de chiffrer le message à envoyer au moyen de la clé publique du destinataire (qu'il trouvera par exemple dans un serveur de clés tel qu'un annuaire). Ce dernier sera en mesure de déchiffrer le message à l'aide de sa clé privée (qu'il est seul à connatre).

Ce système est basé sur une fonction facile à calculer dans un sens (appelée *fonction à trappe* à sens unique ou en anglais *one-way trapdoor function*), mais qui est mathématiquement très difficile à inverser sans la clé privée (appelée *trappe*).

Pour faire une image avec le "monde réel", il s'agit pour un utilisateur de

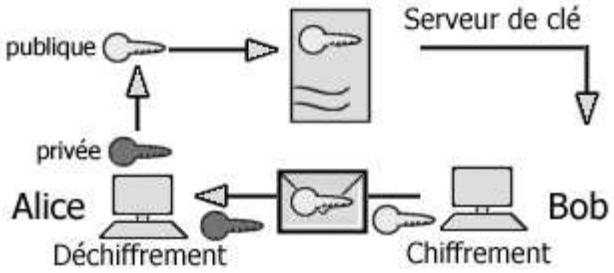

FIG. 4.1 – Clé publique

créer une clé aléatoire (la clé privée), puis de fabriquer un grand nombre de cadenas (clé publique) qu'il dispose dans un casier accessible par tous (le casier joue le rle de canal non sécurisé). Pour lui faire parvenir un document, chaque utilisateur peut prendre un cadenas (ouvert), fermer une valisette contenant le document grce à ce cadenas, puis d'envoyer la valisette au propriétaire de la clé publique (le cadenas). Seul le propriétaire sera alors en mesure d'ouvrir la valisette avec sa clé privée.

Le problème consistant à se communiquer la clé de déchiffrement n'existe plus, les clés publiques pouvant être envoyées librement. Le chiffrement par clés publiques permet donc à des personnes de s'échanger des messages cryptés sans pour autant possèder de secret en commun.

4.2 Notion de clé de session

4.2.1 Intérêt d'une clé de session

Les algorithmes asymétriques (entrant en jeu dans les cryptosystèmes à clé publique) permettent de s'affranchir de problèmes liés à l'échange de clé via un canal sécurisé. Toutefois, ces derniers restent beaucoup moins efficaces (en terme de temps de calcul) que les algorithmes symétriques.

Ainsi, la notion de clé de session est un compromis entre le chiffrement symétrique et asymétrique permettant de combiner les deux techniques.

Le principe de la clé de session est simple : il consiste à générer aléatoirement une clé de session de taille raisonnable, et de chiffrer celle-ci à l'aide d'un algorithme de chiffrement à clef publique (plus exactement à l'aide de la clé publique du destinataire).

Le destinataire est en mesure de déchiffrer la clé de session à l'aide de sa clé privée. Ainsi, expéditeur et destinataires sont en possession d'une clé commune dont ils sont seuls connaisseurs. Il leur est alors possible de s'envoyer des

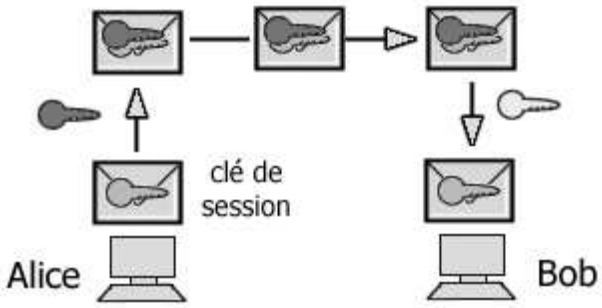

FIG. 4.2 – Session

document chiffrés à l'aide d'un algorithme de chiffrement symétrique.

4.2.2 Algorithme de Diffie-Hellman

L'algorithme de *Diffie-Hellman* (du nom de ses inventeurs Diffie et Hellman) a été mis au point en 1976 afin de permettre l'échange de clés à travers un canal non sécurisé. Il repose sur la difficulté du calcul du logarithme discret dans un corps fini.

4.2.3 Attaque Man-in-the-middle

L'algorithme de *Diffie-Hellman* est cependant sensible à l'attaque "*Man in the middle*" (traduit parfois en "*attaque de l'intercepteur*" ou "*attaque par le milieu*".

4.3 La signature électronique

4.3.1 Introduction à la notion de signature électronique

Le paradigme de signature électronique (appelé aussi *signature numérique*) est un procédé permettant de garantir l'authenticité de l'expéditeur (fonction d'*authentification*), ainsi que de vérifier l'intégrité du message reu.

La signature électronique assure également une fonction de non-répudiation, c'est-à-dire qu'elle permet d'assurer que l'expéditeur a bien envoyé le message (autrement dit elle empêche l'expéditeur de nier avoir expédié le message).

4.3.2 Qu'est-ce qu'une fonction de hachage ?

Une fonction de hachage (parfois appelée *fonction de condensation*) est une fonction permettant d'obtenir un condensé (appelé aussi *haché*) d'un texte, c'est-à-dire une suite de caractères assez courte représentant le texte qu'il condense.

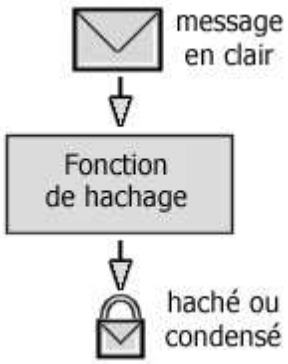

FIG. 4.3 – Fonction de hachage

La fonction de hachage doit être telle qu'elle associe un et un seul haché à un texte en clair (cela signifie que la moindre modification du document *entraine* la modification de son haché). D'autre part, il doit s'agir d'une fonction à sens unique (*one-way function*) afin qu'il soit impossible de retrouver le message original à partir du condensé.

Ainsi, le haché représente en quelque sorte l'*empreinte digitale* (en anglais *finger print*) du document.

Les algorithmes de hachage les plus utilisés actuellement sont :

- **MD5** (*MD* signifiant *Message Digest*), créant une empreinte digitale de 128 bits. Il est courant de voir des documents en téléchargement sur Internet accompagnés d'un fichier MD5, il s'agit du condensé du document permettant de vérifier l'intégrité de ce dernier)
- **SHA** (pour *Secure Hash Algorithm*, pouvant être traduit par *Algorithme de hachage sécurisé*) créant des empreintes d'une longueur de 160 bits

4.3.3 Utilité d'une fonction de hachage

En expédiant un message accompagné de son haché, il est possible de garantir l'intégrité d'un message, c'est-à-dire que le destinataire peut vérifier que le message n'a pas été altéré (intentionnellement ou de manière fortuite) durant la communication.

Lors de la réception du message, il suffit au destinataire de calculer le haché du message reu et de le comparer avec le haché accompagnant le document. Si le message (ou le haché) a été falsifié durant la communication, les deux empreintes ne correspondront pas.

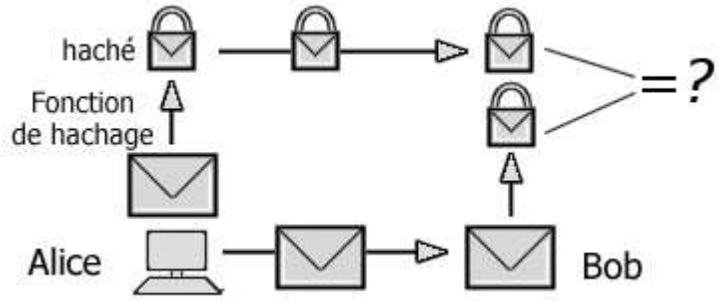

FIG. 4.4 – Utilité d'une fonction de hachage

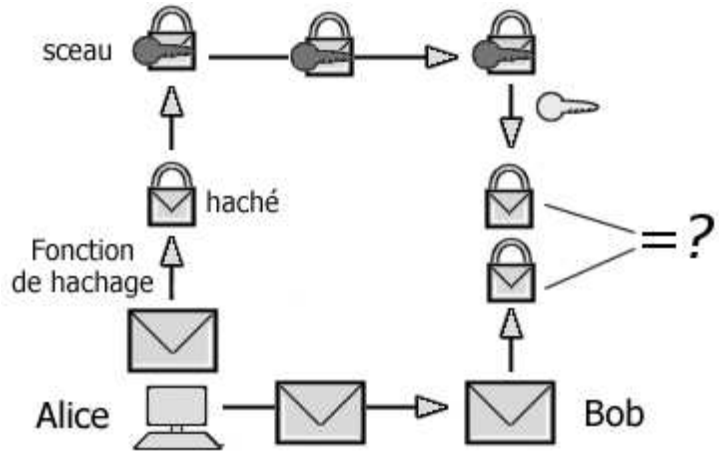

FIG. 4.5 – Scellement des données

4.3.4 Le scellement des données

L'utilisation d'une fonction de hachage permet de vérifier que l'empreinte correspond bien au message reu, mais rien ne prouve que le message a bien été envoyé par celui que l'on croit être l'expéditeur.

Ainsi, pour garantir l'authentification du message, il suffit à l'expéditeur de chiffrer (on dit généralement *signer*) le condensé à l'aide de sa clé privée (*le haché signé* est appelé *sceau*) et d'envoyer le sceau au destinataire.

A réception du message, il suffit au destinataire de déchiffrer le sceau avec la clé publique de l'expéditeur, puis de comparer le haché obtenu avec la fonction de hachage au haché reu en pièce jointe. Ce mécanisme de création de sceau est appelé *scellement*.

Chapitre 5

Certificats - *Public Key Infrastructure (PKI)*

5.1 Introduction aux certificats

5.1.1 Introduction à la notion de certificat

Les algorithmes de chiffrement asymétrique sont basé sur le partage entre les différents utilisateurs d'une clé publique. Généralement le partage de cette clé se fait au travers d'un annuaire électronique (généralement au format LDAP) ou bien d'un site web.

Toutefois ce mode de partage a une grande lacune : rien ne garantit que la clé est bien celle de l'utilisateur a qui elle est associée. En effet un pirate peut corrompre la clé publique présente dans l'annuaire en la remplaant par sa clé publique. Ainsi, le pirate sera en mesure de déchiffrer tous les messages ayant été chiffrés avec la clé présente dans l'annuaire.

Ainsi un certificat permet d'associer une clé publique à une entité (une personne, une machine, ...) afin d'en assurer la validité. Le certificat est en quelque sorte la carte d'identité de la clé publique, délivré par un organisme appelé autorité de certification.

L'autorité de certification est chargée de délivrer les certificats, de leur assigner une date de validité (équivalent à la date limite de péremption des produits alimentaires), ainsi que de révoquer éventuellement des certificats avant cette date en cas de compromission de la clé (ou du propriétaire).

5.1.2 Qu'est-ce qu'un certificat ?

Les certificats sont des petits fichiers divisés en deux parties :

- La partie contenant les informations
- La partie contenant la signature de l'autorité de certification

La structure des certificats est normalisée par le standard X.509 de l'UIT, qui définit les informations contenues dans le certificat :

- Le nom de l'autorité de certification
- Le nom du propriétaire du certificat
- La date de validité du certificat
- L'algorithme de chiffrement utilisé
- La clé publique du propriétaire

L'ensemble de ces informations est signé par l'autorité de certification, cela signifie qu'une fonction de hachage crée une empreinte de ces informations, puis ce condensé est chiffré à l'aide de la clé publique de l'autorité de certification, clé publique ayant été préalablement largement diffusée ou elle même signée par une autorité de niveau supérieur.

Lorsqu'un utilisateur désire communiquer avec une autre personne, il lui suffit de se procurer le certificat du destinataire. Ce certificat contient le nom du destinataire, ainsi que sa clé publique et est signé par l'autorité de certification. Il est donc possible de vérifier la validité du message en appliquant d'une part la fonction de hachage aux informations contenues dans le certificat, en déchiffrant d'autre part la signature de l'autorité de certification avec la clé publique de cette dernière et en comparant ces deux résultats.

Chapitre 6

Cryptosystème

6.1 Le chiffrement de Vigenère

6.1.1 Le chiffrement de Vigenère

Le chiffrement de Vigenère est un cryptosystème symétrique, ce qui signifie qu'il utilise la même clé pour le chiffrement et le déchiffrement. Le chiffrement de Vigenère ressemble beaucoup au chiffrement de César, à la différence près qu'il utilise une clef plus longue afin de pallier le principal problème du chiffrement de César : le fait qu'une lettre puisse être codée d'une seule faon. Pour cela on utilise un mot clef au lieu d'un simple caractère.

On associe dans un premier temps à chaque lettre un chiffre correspondant.

A	B	C	D	E	F	G	H	I	J	K	L	M
1	2	3	4	5	6	7	8	9	10	11	12	13
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
14	15	16	17	18	19	20	21	22	23	24	25	26

Il consiste à coder un texte avec un mot en ajoutant à chacune de ses lettres la lettre d'un autre mot appelé clé. La clé est ajoutée indéfiniment en vis-à-vis avec le texte à chiffrer, puis le code ASCII de chacune des lettres de la clé est ajouté au texte à crypter. Par exemple le texte "rendezvousamidi" avec la clé bonjour sera codé de la manière suivante :

Texte original :

r	e	n	d	e	z	v	o	u	s	a	m	i	d	i
18	5	14	4	5	26	22	15	21	19	1	13	10	4	10

Clé :

b	o	n	j	o	u	r
2	15	14	10	15	21	18

Texte cypté :

r+b	e+o	n+n	d+j	e+o	z+u	v+r	
18+2	5+15	14+14	4+10	5+15	26+21	22+18	
o+b	u+o	s+n	a+j	m+o	i+u	d+r	i+b
15+2	21+15	19+14	1+10	13+15	10+21	4+18	10+2

Pour déchiffrer ce message il suffit d'avoir la clé secrète et faire le déchiffrement inverse, à l'aide d'une soustraction.

Bien que ce chiffrement soit beaucoup plus sr que le chiffrement de César, il peut encore être facilement cassé. En effet, lorsque les messages sont beaucoup plus longs que la clef, il est possible de repérer la longueur de la clef et d'utiliser pour chaque séquence de la longueur de la clef la méthode consistant à calculer la fréquence d'apparition des lettres, permettant de déterminer un à un les caractères de la clef...

Pour éviter ce problème, une solution consiste à utiliser une clef dont la taille est proche de celle du texte afin de rendre impossible une étude statistique du texte crypté. Ce type de système de chiffrement est appelé *système à clé jetable*. Le problème de ce type de méthode est la longueur de la clé de cryptage (plus le texte à crypter est long, plus la clef doit être volumineuse), qui empêche sa mémorisation et implique une probabilité d'erreur dans la clé beaucoup plus grande (une seule erreur rend le texte indéchiffrable...).

6.2 Introduction à RSA

6.2.1 le système RSA

Le premier algorithme de chiffrement à clef publique a été développé par R.Merckle et M.Hellman en 1977. Il fut vite obsolète grce aux travaux de Shamir, Zippel et Herlestman, de célèbres cryptanaliseurs.

En 1978, l'algorithme à clé publique de Rivest Shamir et Adelman (RSA) apparat. Il sert encore à l'aube de l'an 2000 à protéger les codes nucléaires de l'armée américaine et soviétique...

6.2.2 fonctionnement de RSA

Le fonctionnement du cryptosystème RSA est basé sur la difficulté de factoriser deux entiers.

Soit deux nombres premiers p et q, et d un entier premier avec $p-1$ et $q-1$. Le triplet (p, q, d) constitue ainsi la clé privée.

La clé publique est alors le doublet (n, e) créé à l'aide de la clé privée par les transformations suivantes :

$$n = p \times q \tag{6.1}$$

$$e = \frac{1}{d} \bmod ((p-1)(q-1)) \tag{6.2}$$

Si un utilisateur désire envoyer un message M à un destinataire, il lui suffit de se procurer la clé publique (n, e) de ce dernier, puis de calculer le message chiffré c :

$$c = m^e \bmod (n) \tag{6.3}$$

L'utilisateur envoie ensuite le message chiffré c au destinataire, qui est capable de le déchiffrer à l'aide de sa clé privé (p, q, d) :

$$m = m^{ed} \bmod n = c^d \bmod n \tag{6.4}$$

6.3 Introduction à PGP (*Pretty Good Privacy*)

6.3.1 Introduction à PGP

PGP est un cryptosystème (système de chiffrement) inventé par Philip zimmermann, un analyste informaticien. Philip Zimmermann a travaillé de 1984 à 1991 sur un programme permettant de faire fonctionner RSA sur des ordinateurs personnels (PGP).

Cependant, étant donné que celui-ci utilisait RSA sans l'accord de ses auteurs, cela lui a valu des procès pendant 3 ans, il est donc vendu environ 150$ depuis 1993...

Il est très rapide et sr ce qui le rend quasiment impossible à cryptanalyser.

6.3.2 Le principe de PGP

PGP est un cryptosystème hybride, c'est-à-dire qu'il combine des fonctionnalités de la cryptographie de clef publique et de la cryptographie conventionnelle.

Lorsqu'un utilisateur chiffre du texte en clair avec PGP, ces données sont tout d'abord compressées à l'aide d'une fonction de hachage, afin de réduire le

temps de transmission par modem, d'économiser l'espace disque et surtout de renforcer la sécurité cryptographique.

La plupart des cryptanalystes exploitent les modèles trouvés dans le texte en clair pour casser le chiffrement. La fonction de hachage réduit ces modèles dans le texte en clair, améliorant par conséquent considérablement la résistance à la cryptanalyse. Cette méthode de chiffrement associe la facilité d'utilisation du chiffrement à clef publique à la vitesse du chiffrement symétrique. Les cryptosystèmes conventionnels sont environ 1 000 fois plus rapides que les systèmes de chiffrement à clef publique.

De plus, le chiffrement à clef publique résoud non seulement le problème de la distribution des clés, mais également de la transmission des données. Utilisées conjointement, ces deux méthodes améliorent la performance et la distribution des clefs, sans pour autant compromettre la sécurité.

6.3.3 Les étapes du chiffrement

L'opération de chiffrement se fait en deux étapes principales :

- PGP crée une clé secrète IDEA de manière aléatoire, et chiffe les données avec cette clé.
- PGP chiffre la clé secrète IDEA précédemment créee au moyen de la clé RSA publique du destinataire.

De même, l'opération de déchiffrement se fait elle aussi en deux étapes :

- PGP déchiffre la clé secrète IDEA au moyen de la clé RSA privée.
- PGP déchiffre les données avec la clé secrète IDEA précédemment obtenue.

6.3.4 Les fonctionnalités de PGP

PGP offre les fonctionnalités suivantes :

- **Signature électronique et vérification d'intégrité de messages** : fonction basée sur l'emploi simultané d'une fonction de hachage (MD5) et du système RSA. MD5 hache le message et fournit un résultat de 128 bits qui est ensuite chiffré, grce à RSA, par la clef privée de l'expéditeur.
- **Chiffrement des fichiers locaux** : fonction utilisant IDEA.
- **Génération de clefs publiques et privées** : chaque utilisateur chiffre ses messages à l'aide de clefs privées IDEA. Le transfert de clefs électroniques IDEA utilise le système RSA ; PGP offre donc des mécanismes de génération de clefs adaptés à ce système. La taille des clefs RSA est proposée suivant plusieurs niveaux de sécurité : 512, 768, 1024 ou 1280 bits.

- **Gestion des clefs** : fonction s'assurant de distribuer la clef publique de l'utilisateur aux correspondants qui souhaiteraient lui envoyer des messages chiffrés.
- **Certification de clefs** : cette fonction permet d'ajouter un sceau numérique garantissant l'authenticité des clefs publiques. Il s'agit d'une originalité de PGP, qui base sa confiance sur une notion de proximité sociale plutt que sur celle d'autorité centrale de certification.
- **Révocation, désactivation, enregistrement de clefs** : fonction qui permet de produire des certificats de révocation.

6.3.5 Le format des certificats PGP

Un certificat PGP comprend, entre autres, les informations suivantes :

- **Le numéro de version de PGP** : identifie la version de PGP utilisée pour créer la clef associée au certificat.
- **La clef publique du détenteur du certificat** : partie publique de votre paire de clefs associée à l'algorithme de la clef, qu'il soit RSA, DH (Diffie-Hellman) ou DSA (Algorithme de signature numérique).
- **Les informations du détenteur du certificat** : il s'agit des informations portant sur l'" identité " de l'utilisateur, telles que son nom, son ID utilisateur, sa photographie, etc.
- **La signature numérique du détenteur du certificat** : également appelée autosignature, il s'agit de la signature effectuée avec la clef privée correspondant à la clef publique associée au certificat.
- **La période de validité du certificat** : dates/ heures de début et d'expiration du certificat. Indique la date d'expiration du certificat.
- **L'algorithme de chiffrement symétrique** préféré pour la clef : indique l'algorithme de chiffrement que le détenteur du certificat préfère appliquer au cryptage des informations. Les algorithmes pris en charge sont CAST, IDEA ou DES triple

Le fait qu'un seul certificat puisse contenir plusieurs signatures est l'un des aspects uniques du format du certificat PGP. Plusieurs personnes peuvent signer la paire de clefs/ d'identification pour attester en toute certitude de l'appartenance de la clef publique au détenteur spécifié. Certains certificats PGP sont composés d'une clef publique avec plusieurs libellés, chacun offrant un mode d'identification du détenteur de la clef différent (par exemple, le nom et le compte de messagerie d'entreprise du détenteur, l'alias et le compte de messagerie personnel du détenteur, sa photographie, et ce, dans un seul certificat). Dans un certificat, une personne doit affirmer qu'une clef publique et le nom du détenteur de la clef sont associés. Quiconque peut valider les certificats PGP. Les certificats X. 509 doivent toujours être validés par une autorité de certification ou une personne désignée par la CA. Les certificats PGP prennent également en charge une structure hiérarchique à l'aide d'une CA pour la validation des

certificats.

Plusieurs différences existent entre un certificat X. 509 et un certificat PGP. Les plus importantes sont indiquées ci-dessous : Pour créer votre propre certificat PGP, vous devez demander l'émission d'un certificat X. 509 auprès d'une autorité de certification et l'obtenir ;

- Les certificats X. 509 prennent en charge un seul nom pour le détenteur de la clef ;
- Les certificats X. 509 prennent en charge une seule signature numérique pour attester de la validité de la clef ;

6.3.6 Les modèles de fiabilité de PGP

En règle générale, la CA (Certification authority - autorité de certification) inspire une confiance totale pour établir la validité des certificats et effectuer tout le processus de validation manuelle. Mais, il est difficile d'établir une ligne de confiance avec les personnes n'ayant pas été explicitement considérées comme fiables par votre CA. Dans un environnement PGP, tout utilisateur peut agir en tant qu'autorité de certification. Il peut donc valider le certificat de clef publique d'un autre utilisateur PGP. Cependant, un tel certificat peut être considéré comme valide par un autre utilisateur uniquement si un tiers reconnat celui qui a validé ce certificat comme un correspondant fiable. C'est-à-dire, si l'on respecte par exemple mon opinion selon laquelle les clefs des autres sont correctes uniquement si je suis considéré comme un correspondant fiable. Dans le cas contraire, mon opinion sur la validité d'autres clefs est controversée.

Supposons, par exemple, que votre trousseau de clefs contient la clef d'Alice. Vous l'avez validée et, pour l'indiquer, vous la signez. En outre, vous savez qu'Alice est très pointilleuse en ce qui concerne la validation des clefs d'autres utilisateurs. Par conséquent, vous affectez une fiabilité complète à sa clef. Alice devient ainsi une autorité de certification. Si elle signe la clef d'un autre utilisateur, cette clef apparat comme valide sur votre trousseau de clefs.

6.3.7 La révocation d'un certificat PGP

Seul le détenteur du certificat (le détenteur de sa clef privée correspondante) ou un autre utilisateur, désigné comme autorité de révocation par le détenteur du certificat, a la possibilité de révoquer un certificat PGP. La désignation d'une autorité de révocation est utile, car la révocation, par un utilisateur PGP, de son certificat est souvent due à la perte du mot de passe complexe de la clef privée correspondante. Or, cette procédure peut uniquement être effectuée s'il est possible d'accéder à la clef privée. Un certificat X. 509 peut uniquement être

révoqué par son émetteur.

Lorsqu'un certificat est révoqué, il est important d'en avertir ses utilisateurs potentiels. Pour informer de la révocation des certificats PGP, la méthode habituelle consiste à placer cette information sur un serveur de certificats. Ainsi, les utilisateurs souhaitant communiquer avec vous sont avertis de ne pas utiliser cette clef publique.

Deuxième partie

Le stockage et l'EFS (*Encrypting File System*)

Chapitre 7

Introduction

Connaissez-vous la valeur marchande de votre poste de travail ? A combien évalueriez-vous la perte liée au vol de votre portable ?

Son cot peut se décomposer comme suit :

- le coût direct Il s'agit du cot lié au matériel, périphériques, logiciels, ainsi que le cot lié à l'installation de ce matériel. Le risque lié à son détournement est assuré essentiellement par la sécurité physique de l'entreprise.
- le cot lié à la perte des données Il s'évalue essentiellement au temps passé et à l'énergie dépensée pour produire ces données. Les systèmes de sauvegardes sont prévues pour limiter ce risque.
- le cot lié à la diffusion publique de ces données Le détournement de l'information peut tre extrmement grave pour une entreprise, selon la nature mme de cette information, et le destinataire non autorisé (le concurrent ?) de ces informations. C'est cet aspect qui sera développé ici.

Afin de protéger l'accès à ses données, une entreprise peut construire un bastion physique (locaux) ou logique (réseau) autours de celles-ci. Cependant, cela ne permet de se protéger que d'effractions externes à l'entreprise, au service... Il restera toujours une population (interne à l'entreprise le plus souvent) susceptible d'approcher ou d'accéder le système à protéger.
Quand aux portables, ils sont à la merci du moindre voleur de coin de rue...

Les systèmes d'authentification n'évitent pas de booter depuis un autre OS ou la réinstallation du disque sur une autre machine pour analyse ou copie des données qu'il contient. Le disque peut mme tre réinstallé proprement, sans que son propriétaire se soit jamais douté de quoi que ce soit.

C'est pour parer à cet espionnage qu'interviennent les outils de chiffrement disque. Ils permettent de créer un nouveau type de bastion à l'intérieur mme

du disque dur, une zone à l'intérieur de laquelle les données sont à l'abri des regards indésirables.

Il existe plusieurs dizaines de produits de chiffrement de disque sur le marché. Tous fonctionnent selon le mme principe : créer une zone chiffrée sur le disque, tenant lieu de stockage pour les données. La clé de chiffrement est quand à elle détenue par le propriétaire des données. Bien évidemment, la diffusion publique ou perte de cette clé fait tomber les remparts du bastion

Si le disque vient à tomber dans de mauvaises mains, il sera quasiment impossible de pénétrer à l'intérieur de ce bastion, c'est à dire de visualiser les données. Celles-ci, chiffrées, ne seront pas exploitables. Seule la robustesse de l'algorithme utilisé servira de rempart à une tentative de décryptage de l'information.

Un certain nombre de paramètres permettent de distinguer les différents outils de chiffrement.

7.1 Algorithme utilisé

Il s'agit dans tous les cas d'algorithmes à clé secrète. En effet, la majorité des produits fonctionnant à la volée, les opérations de chiffrement/déchiffrement se doivent d'tre rapides (ce qui n'est pas le cas des algorithmes à clé publique). RC4 (128 bits), DES (56 bits), Triple DES (168 bits), Blowfish (128 bits), IDEA (128 bits) et maintenant AES (128, 192 ou 256 bits) sont les plus fréquemment proposés. L'algorithme jouera sur la robustesse du chiffrement, ainsi que sur la vitesse des opérations.

La clé de chiffrement pourra tre un hashage d'un mot de passe utilisateur, ou contenu dans un porte clé logique (fichier), voire physique (stockage de la clé dans une carte à puce, une clé USB), lui-mme protégé par un mot de passe ou un code PIN. Le support physique accentue la sécurité du système, en associant possession d'un objet et connaissance d'un secret. Cette clé de chiffrement peut tre unique pour tous les fichiers, ou diversifiée à partir d'une clé matre pour chaque fichier. Dans ce dernier cas, la clé de session devra tre présente en en-tte du fichier, chiffrée avec la clé matre.

7.2 Mode de fonctionnement

Le mode de fonctionnement varie d'un produit à un autre. La zone chiffrée s'étendra, en fonction du produit, du fichier jusqu'à la partition, voire au disque, en passant par le répertoire ou la liste d'objets éparpillés (fichiers, répertoires). Les opérations de chiffrement se feront quant à eux sur demande utilisateur, sur planification horaire ou automatiquement.

Dans le premier mode, l'utilisateur devra déchiffrer le fichier avant utilisation, et ne devra pas oublier de le rechiffrer après utilisation. Le mode planifié correspond à une surcouche, automatisant le rechiffrement des fichiers en clair à heure fixe, ou lors de l'arrt du poste de travail. Ces modes de fonctionnement ne garantissent pas une confidentialité parfaite des données, les données étant parfois chiffrées, parfois déchiffrées sur le disque.

Le mode automatique fonctionne quant à lui à la volée, sur la base de répertoires dédiés. Tout accès par une application à un fichier situé dans ce répertoire se verra intercepté par la couche de chiffrement/déchiffrement, qui opérera l'opération à la volée. Les données en mémoire seront donc toujours déchiffrées, tandis qu'elles seront toujours chiffrées sur disque. Ce dernier mode apporte l'avantage de fonctionner en tche de fond, masquant à l'utilisateur final non informaticien (la secrétaire, le commercial) une partie des contraintes imposées par la politique de sécurité de l'entreprise (un produit contraignant à utiliser risquant vite d'tre abandonné par les utilisateurs).

7.3 Compatibilité avec l'OS

Les produits ayant un mode de fonctionnement à la volée se greffent au niveau des couches basses de l'OS. Ils sont donc généralement certifiés pour des versions identifiées d'OS. Toutefois, des incompatibilités peuvent survenir avec d'autres logiciels détournant les mmes couches d'accès aux fichiers (ex : antivirus)

Certains éditeurs fournissent une suite logicielle pour poste bureautique incluant anti-virus et chiffrement disque, assurant ainsi la cohérence de l'ensemble. L'existence d'un anti-virus dans l'entreprise sera une contrainte forte sur le choix du produit de chiffrement.

Les produits sont généralement accompagnés d'utilitaires indépendants permettant :

- l'effacement sécurisé d'un fichier sur disque : réecritures multiples de données aléatoires sur l'espace disque occupé par le fichier (la norme militaire américaine préconisant 7 passes).
- le chiffrement d'un fichier en vue d'une diffusion externe (disquette, mail). Dans ce cas, un auto-extractible sera créé (évitant que le correspondant possède le logiciel), le mot de passe devant tre partagé par les correspondants (transmission orale). Ce mode de fonctionnement s'apparente aux fichiers archive (zip) protégés par un mot de passe.

7.4 Travail en groupe

Le travail en groupe est pris en compte de diverses manières par les logiciels.

Certains mettent en avant la notion de propriété du fichier. Seul le propriétaire du fichier pourra y accéder. D'autres permettent de créer un notion de groupe de travail, basé sur l'utilisation commune d'une mme clé. Tous les utilisateurs ayant la clé commune dans leur porte-clé auront accès au fichier.

D'autres enfin, permettent le partage réseau des fichiers en clair, à partir du moment o le propriétaire s'est authentifié sur son poste. Ce mode de fonctionnement laisse au partage réseau la gestion des droits d'accès.

En ce qui concerne l'administration du produit, là encore, les produits se distinguent. Certains produits ne proposent aucune administration. L'utilisateur est complètement libre et autonome. D'autres proposent une administration de création des utilisateurs et de création des clés par utilisateurs. Le déploiement peut tre manuel, ou réalisé au travers d'une architecture logicielle de distribution. Dans ce dernier cas, il sera parfois possible de gérer la configuration de manière centralisée, ainsi que de mettre à jour des updates du produit.

Les fonctions d'administration sont généralement offertes dans le cadre de produits s'inscrivant dans une offre plus globale d'outils (anti-virus...).

7.5 Recouvrement des clés

Ce mode permet de déchiffrer le fichier en cas de perte du mot de passe, d'absence du propriétaire, éventuellement suite à son départ de l'entreprise Cette procédure relève autant de l'opération technique que de l'opération administrative (pourquoi l'administrateur système aurait le pouvoir de tout déchiffrer sans contrle ?).

Généralement, quand cette fonction est prévue, un administrateur est identifié et possède une clé de déverrouillage des fichiers. Cette clé, différente de la clé de chiffrement, utilise souvent un mécanisme de clé privé/publique. La clé de chiffrement est alors incluse dans le fichier chiffré, elle-mme chiffrée avec la clé publique de recovery. Ainsi, seule la personne en possession de la clé privée de recorvery sera en état d'accéder au fichier en clair.

Certains produits proposent de répartir la responsabilité de possession des clés de recovery en définissant plusieurs administrateurs et en imposant la présence d'un certain nombre au moment du recouvrement.

Bien que répondant à un mme objectif de confidentialité par chiffrement des données d'un poste bureautique, les produits restent suffisamment différents les

uns des autres. Chaque produit possède ses caractéristiques, qui se révéleront des avantages et des inconvénients en fonction de l'environnement dans lequel ils s'intégreront. Une étude spécifique permettra d'identifier le produit qui vous conviendra le mieux.

Chapitre 8

Le cryptage du système de fichier à partir Windows 2000

8.1 Introduction

Le système ESF apporte une solution à tous les problèmes inhérents au montage d'une partition par des outils indépendants du système d'exploitation propriétaire de cette dernière et à bien d'autres encore. Les quatre sections suivantes en font une présentation détaillée et expliquent la technologie de cryptage à laquelle il fait appel, son emplacement sur le système, son interaction avec l'utilisateur et la récupération des données.

Le système EFS repose sur le cryptage par clé publique et tire parti de l'architecture CryptoAPI de Windows. Chaque fichier est crypté au moyen d'une clé générée de faon aléatoire, qui est indépendante de la paire clé publique/clé privée d'un utilisateur, étouffant ainsi dans l'uf les nombreuses formes d'attaques reposant sur l'analyse de la cryptographie.

Le cryptage des fichiers peut utiliser tous les algorithmes symétriques. La première version du système EFS utilise l'algorithme de cryptage DES. Les versions futures permettront d'autres schémas de cryptage.

Le système EFS prend également en charge le cryptage et le décryptage des fichiers stockés sur des serveurs de fichiers distants. Remarquons que dans ce cas, seul le cryptage des données sur le disque est pris en charge, les données transitant de faon non protégée sur le réseau. Windows fournit des protocoles de réseau, tels que SSL/PCT, permettant le cryptage de l'accès aux données sur le réseau.

Le système EFS est étroitement intégré à NTFS. Lorsque des fichiers temporaires sont créés, les attributs du fichier d'origine sont copiés dans des fichiers temporaires, à condition que tous les fichiers se trouvent sur un volume NTFS. Si vous cryptez un fichier, EFS crypte aussi ses copies temporaires. Le système EFS réside dans le noyau du système d'exploitation et stocke les clés de cryptage des fichiers dans la réserve non paginée, de sorte que ces clés ne se trouvent jamais dans le fichier de pagination.

La configuration par défaut du système EFS permet aux utilisateurs de crypter des fichiers sans aucun effort administratif. EFS génère automatiquement une paire de clés publiques permettant le cryptage des fichiers, sauf s'il en existe déjà une.

Le cryptage et le décryptage des fichiers sont en charge au niveau du fichier ou du répertoire. Le cryptage des répertoires s'effectue de faon transparente. Tous les fichiers (et les sous répertoires) créés dans un répertoire marqué pour le cryptage sont automatiquement cryptés. Chaque fichier possède une clé de cryptage unique, ce qui sécurise le processus de changement de nom des fichiers. Lorsque vous renommez un fichier d'un répertoire crypté dans un répertoire non crypté d'un mme volume, le fichier reste crypté. Les services de cryptage et de décryptage sont disponibles dans l'Explorateur Windows. Les outils de ligne de commande et les interfaces d'administration du système s'adressent, quant à eux, aux utilisateurs confirmés et aux agents de récupération qui peuvent ainsi tirer pleinement parti des fonctionnalités proposées.

Il n'est pas nécessaire de décrypter un fichier pour pouvoir l'utiliser, le cryptage et le décryptage sont transparents lors de l'échange des données sur le disque. Le système EFS détecte automatiquement un fichier crypté et localise la clé de l'utilisateur dans le magasin de clés du système. Comme le mécanisme de stockage des clés repose sur l'architecture CryptoAPI, les utilisateurs ont la possibilité de conserver des clés sur des supports sécurisés tels que les cartes à puce.

Dans sa version initiale, le système EFS ne prend pas en compte le partage des fichiers. Toutefois, son architecture prévoie le partage des fichiers entre un nombre illimité de personnes, moyennant l'utilisation de leurs clés publiques. Les utilisateurs peuvent alors décrypter leurs fichiers individuellement en se servant de leurs clés privées. De nouveaux utilisateurs peuvent facilement tre ajoutés (s'ils ont une paire clé publique/clé privée configurée) ou supprimés d'un groupe d'utilisateurs détenteurs d'une autorisation d'accès en partage.

Le système EFS comporte un système de récupération des données intégré. L'infrastructure de la sécurité de Windows 2000 fait appel à la configuration des clés de récupération des données. Vous ne pouvez utiliser le cryptage des données que si le système est configuré avec une ou plusieurs clés de récupération. Le

système EFS autorise les agents de récupération à configurer des clés publiques servant à activer la récupération des fichiers. La clé de récupération ne permet de retrouver que la clé de cryptage du fichier qui a été générée de faon aléatoire mais en aucun cas la clé privée de l'utilisateur. Aussi l'agent de récupération ne peut-il récupérer aucune autre information privée.

La récupération des données concerne de nombreuses entreprises désireuses de pouvoir récupérer des données chiffrées, suite au départ d'un employé ou à la perte des clés de cryptage. La stratégie de récupération peut se définir au niveau du contrleur de domaine d'un domaine Windows. Elle s'applique alors à tous les ordinateurs du domaine. Les administrateurs de domaine, qui en ont généralement la charge, peuvent la déléguer à des comptes d'administrateur de sécurité des données, désignés grce aux fonctions de délégation du Service d'annuaire de Windows. Le contrle des utilisateurs autorisés à récupérer des données cryptées s'en trouve ainsi amélioré et bénéficie d'une plus grande souplesse. Le système EFS prend également en charge plusieurs agents de récupération : il est en effet possible de disposer de plusieurs configurations de clés de récupération, ce qui assure aux entreprises souplesse et redondance dans la mise en uvre de leurs procédures de récupération.

Le système EFS s'utilise également dans un environnement familial. Il génère automatiquement des clés de récupération qu'il enregistre en tant que clés machine, en l'absence de domaine Windows. Les utilisateurs ont là encore la possibilité d'utiliser l'outil de ligne de commande pour récupérer les données, en passant par le compte administrateur. Leur charge administrative s'en trouve ainsi allégée.

Le menu contextuel comporte les fonctions EFS suivantes :

- **Cryptage** : cette option permet à l'utilisateur de crypter le fichier sélectionné. Dans le cas d'un répertoire, elle laisse l'utilisateur crypter tous les fichiers (et sous-répertoires) qui le composent et marque le répertoire comme étant crypté.
- **Décryptage** : inverse de l'option cryptage, cette option permet à l'utilisateur de décrypter le fichier sélectionné. Dans le cas d'un répertoire, elle autorise l'utilisateur à décrypter tous les fichiers qui le composent et marque le répertoire comme étant décrypté.
- **Configuration** : cette fonction permet aux utilisateurs de générer, d'exporter, d'importer et de gérer les clés publiques servant au cryptage des fichiers à base de système EFS. La configuration est intégrée aux autres paramètres de sécurité utilisateur. Elle s'adresse aux utilisateurs avancés qui souhaitent gérer leurs propres clés. En règle générale, les utilisateurs n'ont aucune configuration à créer. EFS génère automatiquement, pour les utilisateurs qui en sont dépourvus, les clés de cryptage des fichiers.

8.2 Récupération du cryptage

Avant toute utilisation, le système EFS nécessite la mise en place d'une stratégie de récupération des données au niveau du domaine (ou localement si l'ordinateur n'appartient pas à un domaine). La stratégie de récupération est définie par les administrateurs de domaine (ou par des délégués connus sous le nom d'agents de récupération), qui gèrent les clés de récupération de tous les ordinateurs d'un domaine.

En cas de perte de sa clé privée, un utilisateur peut récupérer son fichier protégé en l'exportant et en l'envoyant par courrier électronique à l'un des agents de récupération. L'agent importe le fichier sur un ordinateur sécurisé disposant des clés de récupération privées et utilise l'outil de récupération au niveau de la ligne de commande pour le décrypter. Il restitue ensuite à l'utilisateur le fichier en texte simple. En l'absence de domaine, comme dans les petites entreprises ou dans un contexte familial, la récupération peut se faire sur l'ordinateur autonome utilisé.

8.3 Architecture du système EFS

Le système EFS met en uvre le cryptage et le décryptage des données au moyen d'un schéma à clé publique. Les données des fichiers sont cryptées au moyen d'un algorithme symétrique rapide avec une clé de cryptage des fichiers (FEK, File Encryption Key). La clé FEK est une clé générée de manière aléatoire dont la longueur est fixée soit par l'algorithme lui-mme, soit par la législation si l'algorithme prend en charge des longueurs de clé variables. Les aspects liés à l'exportation du système ESF font l'objet d'un document séparé.

La clé FEK est cryptée au moyen d'une ou de plusieurs clés publiques afin de générer une liste de FEK cryptées. La partie publique de la paire de clés de l'utilisateur sert à crypter les FEK. La liste des FEK cryptées est stockée avec le fichier crypté dans un attribut EFS spécial appelé champ de décryptage des données (DDF, Data Decryption Field). Les informations de cryptage du fichier sont étroitement liées au fichier. La partie privée de la paire de clés de l'utilisateur sert au décryptage. La clé FEK est décryptée à l'aide de la partie privée de la paire de clés. Elle peut tre conservée à part, sur une carte à puce ou sur tout autre moyen de stockage sécurisé.

Une clé d'utilisateur peut aussi tre cryptée à l'aide d'un algorithme symétrique, comme une clé dérivée d'un mot de passe. EFS n'utilise pas cette méthode peu fiable en raison du risque d'attaques par dictionnaire.

La clé FEK est aussi cryptée au moyen d'une ou de plusieurs clés publiques de cryptage de clés de récupération. Là encore, la partie publique de chaque paire de clés sert à crypter les FEK. La liste des FEK cryptées est également stockée

avec le fichier dans un attribut EFS spécial appelé champ de récupération des données (DRF, Data Recovery Field). Seules les parties publiques des paires de clés de récupération sont nécessaires au cryptage de la FEK dans le champ DRF. Ces clés publiques de récupération doivent tre présentes en permanence sur un système EFS pour effectuer les opérations classiques sur le système de fichiers. La récupération proprement dite doit, en toute logique, demeurer une opération rare, se révélant nécessaire uniquement lorsque des utilisateurs quittent une entreprise ou perdent leurs clés. C'est pourquoi les agents de récupération stockent séparément les parties privées des clés sur des cartes à puce ou sur tout autre support de stockage sécurisé. Remarque : La figure montre un seul utilisateur et un seul agent de récupération, -mais il pourrait tout aussi bien y avoir une liste d'utilisateurs et une liste d'agents de récupération avec des clés indépendantes. La première version du système EFS prend en charge un seul utilisateur et plusieurs agents de récupération.

Ce schéma très simple débouche sur une technologie de cryptage puissante qui permet à plusieurs utilisateurs de partager un fichier crypté ou à plusieurs agents de récupération de récupérer un fichier, si nécessaire. Ce schéma étant entièrement algorithmique, tous les algorithmes de cryptographie sont exploitables lors des différentes phases de cryptage. Il s'agit là d'un point qui se révélera capital lorsque des algorithmes plus performants auront vu le jour.

Le système d'exploitation de Windows 2000 englobe les composants ESF suivants :

- **Pilote EFS**. Le pilote EFS est placé en couche au-dessus de NTFS. Il communique avec le service EFS pour demander des clés de cryptage des fichiers, des champs DDF, des champs DRF et d'autres services de gestion des clés. Il transmet ces informations à la bibliothèque d'exécution du système de fichiers (FSRTL, File System Run-Time Library) pour procéder à diverses opérations sur le système de fichiers (ouverture, lecture, écriture et ajout) de faon transparente.
- M**odule FSRTL**. Il s'agit d'un module du pilote EFS qui met en uvre les appels NTFS de gestion des différentes opérations du système de fichiers telles que les lectures, écritures et ouvertures sur les fichiers et répertoires cryptés, ainsi que des opérations de cryptage, de décryptage et de récupération des données d'un fichier lorsque celui-ci est lu ou écrit sur le disque. Bien que le pilote EFS et le module FSRTL soient mis en uvre en tant que composant unique, ils ne communiquent jamais directement. Ils utilisent le mécanisme d'appel de contrle de fichiers NTFS pour échanger des messages, garantissant ainsi la participation de NTFS à toutes les opérations de fichiers. Les opérations effectuées à l'aide des mécanismes de contrle des fichiers comprennent l'écriture des données d'attributs EFS (DDF et DRF) en tant qu'attribut de fichier et la communication de la FEK calculée dans le service EFS à FSRTL afin qu'elle soit définie dans le contexte du fichier ouvert. Ce contexte de fichier sert ensuite au cryp-

tage et au décryptage transparent des écritures et lectures des données du fichier sur disque.
- **Service EFS**. Le service EFS fait partie du sous-système de sécurité. Il utilise le port de communication LPC existant entre l'autorité de sécurité locale (LSA) et le contrleur de référence de sécurité en mode noyau pour communiquer avec le pilote EFS. En mode utilisateur, il s'interface avec CryptoAPI pour fournir les clés de cryptage et générer les DDF et les DRF. Il prend également en charge les API Win32 en vue du cryptage, du décryptage, de la récupération, de l'importation et de l'exportation de fichiers.
- API Win32. Les API Win32 fournissent des interfaces de programmation pour crypter des fichiers en texte simple, décrypter ou récupérer des fichiers en texte chiffré, ainsi que pour importer et exporter des fichiers cryptés (sans décryptage préalable). Ces API sont prises en charge dans une DLL système standard, Advapi32.dll.

8.4 Débat sur l'exportation du système EFS

Le système EFS permet à des agents de récupération de récupérer des données. L'architecture de récupération des données procède de l'effort consenti par Microsoft pour se mettre en conformité avec l'actuelle réglementation en matière d'exportation de méthodes de cryptage et pour offrir à sa clientèle internationale un cryptage plus puissant que celui sur 40 bits. Dans ce cadre, le système EFS utilise l'algorithme DES standard avec une clé de cryptage sur 56 bits. Le système EFS est conu pour prendre en charge des algorithmes dotés de clés de puissance variable, en vue des évolutions futures.

8.5 Conclusion

Le système EFS de Windows 2000 permet aux utilisateurs de crypter des fichiers NTFS individuels ou des répertoires entiers à l'aide d'un schéma cryptographique par clé publique très puissant.

- Le système EFS prend en charge le cryptage des fichiers distants accessibles par partage de fichiers.
- Le système EFS donne aux entreprises les moyens de mettre en place des stratégies de récupération des données efficaces.
- La stratégie de récupération des données est intégrée à la stratégie globale de sécurité Windows. Sa gestion peut tre déléguée à des agents de récupération.
- La récupération des données est une opération bien matrisée. Elle n'autorise l'accès qu'aux données récupérées et en aucun cas à la clé privée de l'utilisateur qui a servi à crypter le fichier.
- Le décryptage et le recryptage des fichiers à chaque utilisation ne sont

pas nécessaires. Ces opérations sont transparentes pour les lectures et les écritures sur le disque.
- Le système EFS prend en charge les fonctions d'exportation et d'importation qui permettent d'enregistrer, de restaurer et de transférer des fichiers cryptés sans décryptage préalable.
- Le système EFS est intégré au système d'exploitation pour éviter les fuites d'informations sur les clés à partir des fichiers de pagination et garantir que toutes les copies temporaires d'un fichier crypté le sont aussi.
- La version nord-américaine du système EFS utilise l'algorithme DES avec une clé codée sur 56 bits. La version internationale du système ESF utilise également l'algorithme DES mais avec une clé codée sur 40 bits seulement.

Troisième partie

Bibliographie

Grard Florin - *Les Techniques de Cryptographie*

CNRS - *De l'usage de la Cryptographie*

Introduction to Cryptography - the PGP 6.5.1 documentation

`http://www.althes.fr/Technologie/`
`http://www.microsoft.com/france/technet/produits/Win2000S/info/nt5efs.html`
`http://www.geocities.com/SiliconValley/Bay/9648/crypto.htm`