

«standard de chiffrement de données»), triple-DES, CAST, IDEA et *Skipjack*. Les algorithmes par blocs sont au centre de la plupart des recherches actuelles en cryptographie.

La clé est la clé

En cryptographie, l'étape la plus sensible est la création des clés. Pour qu'un système soit le plus sûr possible, les clés doivent être des nombres réellement aléatoires, imprévisibles par un éventuel pirate. Les séquences pseudo-aléatoires déterministes que les ordinateurs engendrent pour les jeux ou les simulations sont inutilisables, car ils sont insuffisamment aléatoires. La seule manière d'engendrer de vrais nombres aléatoires est d'utiliser un «bruit» physique, telle la désintégration radioactive.

À l'aide d'un ordinateur, on fabrique difficilement des nombres aléatoires aussi bons que ceux issus de la physique. On obtient toutefois des résultats raisonnables quand on utilise l'intervalle de temps, en microsecondes, qui sépare la frappe de deux touches d'un clavier : cette valeur est impossible à prédire. Les données obtenues ainsi ne sont pas assez aléatoires pour fabriquer directement des clés, mais peuvent être transformées pour accroître le désordre.

La seule méthode cryptographique dont on ait prouvé la parfaite sécurité est le chiffrement de Vernam, où la longueur de la clé est égale à la longueur du message. Dans cette méthode, on utilise une séquence aléatoire pour chiffrer le message bit par bit, c'est-à-dire que le 34^e bit de la clé code le 34^e bit du message. La clé doit être parfaitement aléatoire, elle ne peut pas être une séquence pseudo-aléatoire issue d'un algorithme déterministe, sinon le code pourrait facilement être cassé. Le chiffrement de Vernam est rarement utilisé, en raison de difficultés pratiques : la clé, qui est aussi longue que le message, doit être envoyée au destinataire par un canal de transmission sûr. De surcroît, on ne peut utiliser chaque clé qu'une seule fois, sinon un pirate pourrait la décrypter.

Si la taille des clés détermine l'efficacité des systèmes cryptographiques, la conception des systèmes est également très importante. Considérons un simple code de substitution où tous les A seraient transformés en W, tous les B en K, tous les C en Q, et ainsi de suite. Le nombre de façons différentes d'arranger l'alphabet est égale à 26! (ce nombre, qui se lit «factorielle 26», est égal à $26 \times 25 \times 24 \dots 3 \times 2 \times 1$), soit environ 4×10^{26} (un nombre de 26 chiffres)! On ne peut tester toutes les combinaisons une à une, mais, lorsque j'étais enfant, je m'amusais à casser ce type de code avec un papier et un crayon : je regardais simplement la lettre la plus fréquente et je supposais qu'il s'agissait d'un E ; puis je cherchais la deuxième lettre la plus fréquente et je supposais que c'était un S, etc. Ainsi, malgré un espace des clés très grand, ce type de code est très simple.

En fait, la sécurité d'un système cryptographique n'est pas proportionnelle à la taille de la clé. Pour les systèmes par blocs, par exemple, la relation est plutôt exponentielle : un bit de plus à la clé oblige un pirate éventuel à doubler le nombre d'essais. L'effort nécessaire est élevé au carré quand la taille de la clé est doublée. Pour casser un système qui utilise une clé de longueur 128 bits, il faut en moyenne 2^{127} (soit $1,7 \times 10^{38}$) opérations.

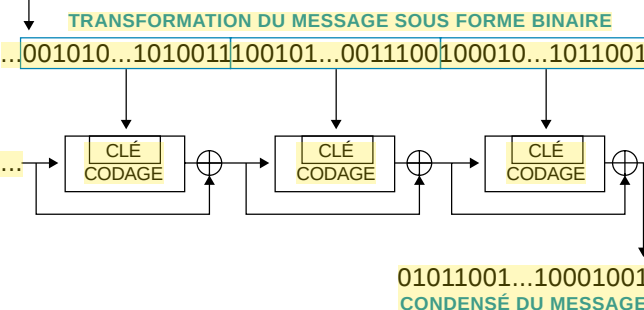
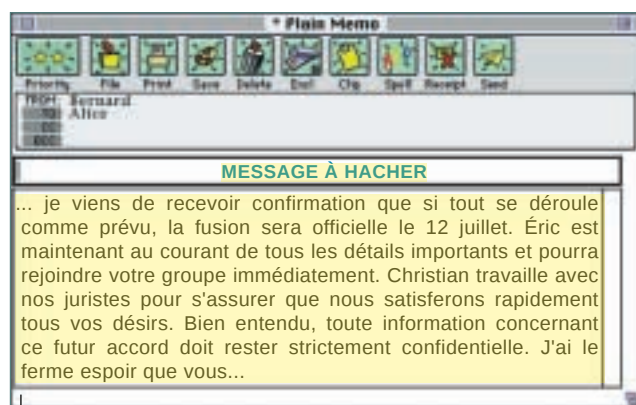
Pour les algorithmes à clé publique, la difficulté augmente plus lentement. L'espace des clés augmente moins vite qu'une loi exponentielle, mais plus vite qu'une loi de

puissance ; le doublement de la longueur de la clé n'élève pas au carré l'effort à fournir pour casser le code, mais le travail reste considérable. Par exemple, dans le système RSA, les algorithmes modernes de factorisation peuvent faire beaucoup mieux que d'essayer toutes les combinaisons de nombres premiers possibles pour factoriser un nombre de grande taille. Le code de Diffie et Hellman est, lui aussi, «sous-exponentiel». Par exemple, une clé de 3 000 bits pour RSA ou Diffie-Hellman est aussi difficile à décrypter qu'une clé de 128 bits d'un système de chiffrement par blocs.

Le chiffrement par blocs n'est toutefois pas invulnérable. En 1998, la Société *Electronic Frontier Foundation* a construit, pour moins de 1,5 million de francs, un ordinateur massivement parallèle qui a cassé un message chiffré par DES en explorant toutes les clés à 56 bits en moins d'une semaine.

La force brute n'est pas la seule façon de percer un code. Des méthodes mathématiques élaborées et des outils statistiques raccourcissent le travail. Certaines méthodes recherchent, par exemple, un motif particulier dans le message chiffré. Ces méthodes statistiques sont classées en trois catégories selon ce qui est connu du texte clair et du texte chiffré.

Quand le pirate connaît seulement le texte chiffré, il n'a rien pour le guider dans sa recherche de la clé, si bien que même un système simple de chiffrement résiste aux



2. UNE FONCTION DE HACHAGE crée un condensé de message, semblable à une empreinte digitale utilisée pour détecter les contrefaçons. Le texte d'un message est d'abord converti en chiffres binaires (la lettre A peut être représentée par 00000, la lettre B par 00001, le C par 00010, etc.) ; puis la suite de caractères qui en résulte est découpée en blocs de taille fixe. Chacun des blocs sert alors de clé dans un système de codage. La sortie finale est le condensé du message original. Quelle que soit la longueur du message initial, le condensé a toujours la même taille. C'est une opération à sens unique, car il est quasi impossible de retrouver le message original à partir de son condensé. De surcroît, avec cet algorithme, deux messages donnent presque toujours des condensés différents et il est impossible de trouver un autre message ayant le même condensé : ce condensé est une «empreinte digitale» du message.

assauts. En revanche, quand le pirate connaît au moins une partie du message, ses chances de succès augmentent. Par exemple, s'il sait qu'il commence par «Cher Monsieur Dupond», il peut d'abord chercher une clé qui décrypte la partie «Cher Monsieur Dupond» du texte clair, puis poursuivre le décryptage à l'aide des informations trouvées.

Même quand le pirate ne connaît que le langage utilisé dans le texte clair, russe, français ou COBOL, il peut exploiter cette information. Si le message est en français, par exemple, alors le mot le plus fréquemment utilisé est sans doute «le» ou «la». Pour déjouer de telles attaques, certains systèmes cryptographiques compriment le message en clair,

