Une Introduction à la Cryptographie

[traduction française: news:fr.misc.cryptologie, 1998]

PGP*, Version 6.0.2

11-98. Printed in the United States of America.

Network Associates, Inc. (408) 988-3832 main
3965 Freedom Circle
Santa Clara, CA 95054
http://www.nai.com

info@nai.com

* is sometimes used instead of the ® for registered trademarks to protect marks registered

LIMITED WARRANTY

Limited Warranty. Network Associates warrants that for sixty (60) days from the date of original purchase the media (for example diskettes) on which the Software is contained will be free from defects in materials and workmanship.

Customer Remedies. Network Associates' and its suppliers' entire liability and your exclusive remedy shall be, at Network Associates' option, either (i) return of the purchase price paid for the license, if any, or (ii) replacement of the defective media in which the Software is contained with a copy on nondefective media. You must return the defective media to Network Associates at your expense with a copy of your receipt. This limited warranty is void if the defect has resulted from accident, abuse, or misapplication. Any replacement media will be warranted for the remainder of the original warranty period. Outside the United States, this remedy is not available to the extent Network Associates is subject to restrictions under United States export control laws and regulations.

Warranty Disclaimer. To the maximum extent permitted by applicable law, and except for the limited warranty set forth herein, THE SOFTWARE IS PROVIDED ON AN "AS IS" BASIS WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. WITHOUT LIMITING THE FOREGOING PROVISIONS, YOU ASSUME RESPONSIBILITY FOR SELECTING THE SOFTWARE TO ACHIEVE YOUR INTENDED RESULTS, AND FOR THE INSTALLATION OF, USE OF, AND RESULTS OBTAINED FROM THE SOFTWARE. WITHOUT LIMITING THE FOREGOING PROVISIONS, NETWORK ASSOCIATES MAKES NO WARRANTY THAT THE SOFTWARE WILL BE ERROR-FREE OR FREE FROM INTERRUPTIONS OR OTHER FAILURES OR THAT THE SOFTWARE WILL MEET YOUR REQUIREMENTS. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, NETWORK ASSOCIATES DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT WITH RESPECT TO THE SOFTWARE AND THE ACCOMPANYING DOCUMENTATION. SOME STATES AND JURISDICTIONS DO NOT ALLOW LIMITATIONS ON IMPLIED WARRANTIES, SO THE ABOVE LIMITATION MAY NOT APPLY TO YOU. The foregoing provisions shall be enforceable to the maximum extent permitted by applicable law.

Préface

La cryptographie est un sujet de romans d'espionnage et de bandes dessinées. Autrefois, les enfants conservaient leurs emballages de chewing-gums et les renvoyaient pour recevoir "l'anneau de décodage secret du Captain Midnight". Tout le monde a vu une série T.V. ou un film mettant en scène un gentleman en costume discret portant une mallette menottée à son poignet. Le mot "espionnage" évoque des images de James Bond, de poursuites de voitures, et de balles sifflant aux alentours.

Et vous voici, assis à votre bureau, occupé à la tâche banale d'envoyer un état des ventes à un collègue, de manière à ce que personne d'autre ne puisse le lire. Vous voulez simplement être sûr que votre collègue en sera le destinataire réel et unique, et vous voulez qu'il sache que l'envoi vient bien de vous sans équivoque possible. La sécurité nationale n'est pas en jeu, mais si vos concurrents venaient à intercepter ce document, cela pourrait vous coûter cher. Comment allez-vous parvenir à vos fins?

Vous pouvez utiliser la cryptographie. Vous trouverez peut-être qu'il lui manque le sel des mots de passe chuchotés dans des venelles obscures, mais le résultat sera le même: l'information sera révélée seulement à ceux à qui elle était destinée.

Qui devrait lire cet ouvrage

Ce guide sera utile à quiconque désire se familiariser avec les principes de base de la cryptographie. Il explique la terminologie et la technologie que vous rencontrerez en utilisant les produits PGP. Vous trouverez sans doute sa lecture utile avant de commencer à travailler en utilisant la cryptographie.

Comment utiliser ce guide

Ce guide décrit comment utiliser PGP afin de sécuriser les messages de votre organisation, et le stockage des données.

Le Chapitre 1, "Les Fondements de la Cryptographie", fournit une vue d'ensemble de la terminologie et des concepts que vous rencontrerez en utilisant les produits PGP.

Le Chapitre 2, "Phil Zimmermann sur PGP", écrit par le créateur de PGP, contient un exposé à propos de la sécurité, de la protection de la correspondance privée, et des vulnérabilités inhérentes à tout système de sécurité, PGP inclus.

Pour plus d'informations

Il y a plusieurs façons de trouver plus d'informations à propos de PGP et de ses produits.

Service clients

Pour acheter des produits ou obtenir de l'information sur le produit, contactez le département clients de Network Associates.

Vous pouvez contacter la "hot line" à un des numéros suivants du lundi au vendredi entre 6:00 et 18:00, heure du Pacifique.

Téléphone	(408) 988-3832

Ou écrivez à:

Network Associates, Inc.
3965 Freedom Circle
Santa Clara, CA 95054
U.S.A.

Assistance Technique

Network Associates est célèbre pour l'attention portée à la satisfaction de ses clients. Nous poursuivons cette tradition en faisant de notre site Internet une ressource de valeur pour répondre aux questions relevant de l'assistance technique. Nous vous encourageons à l'utiliser comme votre première ressource afin d'y trouver des réponses aux questions fréquemment posées, pour les mises à jour des logiciels de Network Associates, et pour consulter les informations et les nouveautés de Network Associates en matière de cryptographie.

World Wide Web	http://www.nai.com/

L'assistance technique pour vos produits PGP est également disponible par les moyens suivants:

Téléphone	(408) 988-3832
E-mail	PGPSupport@pgp.com

Pour que nous puissions vous donner rapidement et efficacement les réponses que vous demandez, le personnel d'assistance technique de Network Associates a besoin de certaines informations concernant votre ordinateur et vos logiciels. Veuillez tenir cette information prête avant de nous appeler:

- Nom du produit PGP
- Version du produit PGP
- Type d'ordinateur et de processeur
- Taille de la mémoire RAM disponible
- Système d'exploitation, avec sa version, et type de réseau
- Texte précis de tout message d'information ou d'erreur qui serait apparu à l'écran, ou enregistré dans un fichier journal (tous les produits n'offrent pas la fonctionnalité d'un fichier-journal)
- Nom et version de l'application e-mail utilisée (si le problème provient de l'intégration de PGP avec un logiciel d'e-mail, comme par exemple le plug-in Eudora)

Lectures recommandées

Voici une liste d'ouvrages que vous pourrez trouver utiles pour approfondir vos connaissances sur la cryptographie:

Livres non techniques et techniques pour débutants

- "Cryptographie pour Internet", par Philip R. Zimmermann. Paru dans *Scientific American*, Octobre 1998. Cet article, écrit par l'auteur de PGP, est une introduction à divers protocoles et algorithmes cryptographiques, dont beaucoup sont utilisés par PGP.
- "Privacy on the Line", par Whitfield Diffie and Susan Eva Landau. *MIT Press*; ISBN: 0262041677
  Ce livre est une discussion de l'histoire et de la politique autour de la cryptographie et de la sécurité des communications. C'est une excellente lecture, même pour les débutants et les non techniciens, et il contient certaines informations inconnues même de nombreux experts.
- "The Codebreakers", par David Kahn. *Scribner*; ISBN: 0684831309
  Ce livre est une histoire des codes et des casseurs de codes depuis l'époque des Egyptiens jusqu'à la fin de la Seconde Guerre Mondiale. Kahn l'a d'abord écrit dans les années 60, et une édition révisée fut publiée en 1996. Ce livre ne vous enseignera rien sur la façon dont la cryptographie est mise en œuvre, mais il a été l'inspiration de toute la génération des cryptographes modernes.
- "Network Security: Private Communication in a Public World", par Charlie Kaufman, Radia Perlman, and Mike Spencer. *Prentice Hall*; ISBN: 0-13-061466-1
  C'est une bonne description des systèmes de sécurité des réseaux et des protocoles, incluant des descriptions de ce qui marche, ce qui ne marche pas, et pourquoi. Publié en 1995, il contient peu d'informations sur les dernières avancées technologiques, mais c'est toujours un bon livre. Il contient aussi une des plus claires descriptions jamais écrites de la façon dont marche le DES.

Livres de niveau intermédiaire

- "Applied Cryptography: Protocols, Algorithms, and Source Code in C", par Bruce Schneier. *John Wiley & Sons*; ISBN: 0-471-12845-7
  C'est une bonne introduction technique sur la façon dont beaucoup de systèmes cryptographiques fonctionnent. Si vous voulez devenir un expert, c'est ici qu'il faut commencer [ouvrage traduit en français aux Editions ITP].
- "Handbook of Applied Cryptography", par Alfred J. Menezes, Paul C. van Oorschot, et Scott Vanstone. *CRC Press*; ISBN: 0-8493-8523-7
  C'est un livre technique que vous devriez lire après celui de B. Schneier. Il y a beaucoup de mathématiques pures et dures dans ce livre, mais il est néanmoins utilisable pour ceux qui ne comprennent pas les maths.
- "Internet Cryptography", par Richard E. Smith. *Addison-Wesley Pub Co*; ISBN: 020192480
  Ce livre décrit beaucoup de protocoles de sécurité d'Internet. Surtout, il décrit comment même les systèmes les mieux conçus peuvent être battus en brèche du

fait d'une mauvaise utilisation. Ce livre est léger sur les maths et chargé d'informations pratiques.

- "Firewalls and Internet Security: Repelling the Wily Hacker", par William R. Cheswick and Steven M. Bellovin. *Addison-Wesley Pub Co*; ISBN: 0201633574
  Ce livre est écrit par deux chercheurs importants des laboratoires AT&T Bell Labs, à propos de leur expérience dans l'administration et la refonte de la connexion Internet de AT&T. Très lisible.

Livres de niveau avancé

- "A Course in Number Theory and Cryptography", par Neal Koblitz. *Springer-Verlag*; ISBN: 0-387-94293-9
  Un excellent livre du niveau d'un manuel pour diplômé en mathématiques, portant sur la théorie des nombres et la cryptographie.
- "Differential Cryptanalysis of the Data Encryption Standard", par Eli Biham and Adi Shamir. *Springer-Verlag*; ISBN: 0-387-97930-1
  Ce livre décrit la technique de la cryptanalyse différentielle telle qu'appliquée au DES. C'est un excellent livre pour apprendre des choses sur cette technique.

Table des Matières

Les Fondements de la Cryptographie 1

Quand Jules César envoyait des messages à ses généraux, il ne faisait pas confiance à ses messagers. Aussi remplaçait-il chaque A dans ses messages par un D, chaque B par un E, et ainsi de suite à travers l'alphabet. Seul quelqu'un qui connaissait la règle "décalé de 3" pouvait déchiffrer ses messages.

Et c'est ainsi que nous commençons.

Chiffrement et déchiffrement

Les données qui peuvent être lues et comprises sans mesures spéciales sont appelées *texte clair* (ou *libellé*). Le procédé qui consiste à dissimuler du texte clair de façon à cacher sa substance est appelée *chiffrement* [dans le langage courant on parle plutôt de *cryptage* et de ses dérivés: *crypter*, *décrypter*]. Chiffrer du texte clair produit un charabia illisible appelé *texte chiffré* (ou *cryptogramme*). Vous utilisez le chiffrement pour garantir que l'information est cachée à quiconque elle n'est pas destinée, même ceux qui peuvent lire les données chiffrées. Le processus de retour du texte chiffré à son texte clair originel est appelé *déchiffrement*.

La Figure 1-1 illustre ce processus.

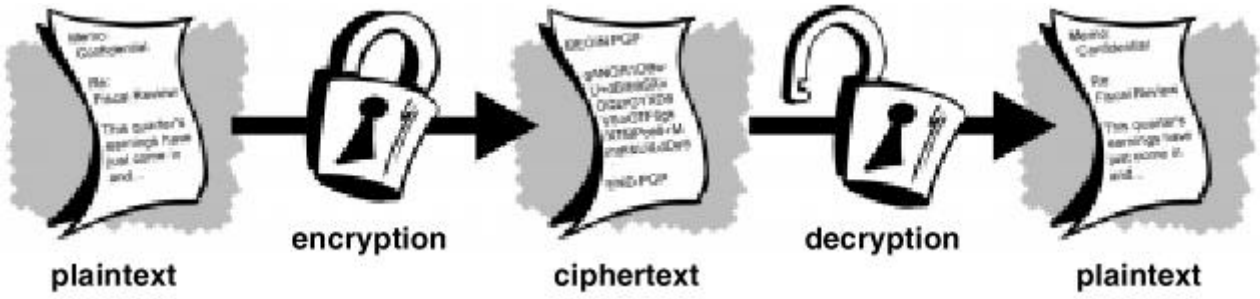

Figure 1-1. Chiffrement et déchiffrement

Qu'est-ce que la cryptographie?

La *cryptographie* est la science qui utilise les mathématiques pour chiffrer et déchiffrer des données. La cryptographie vous permet de stocker des informations sensibles ou de les transmettre à travers des réseaux non sûrs (comme Internet) de telle sorte qu'elles ne puissent être lues par personne à l'exception du destinataire convenu.

Alors que la cryptographie est la science de la sécurisation des données, la *cryptanalyse* est la science de l'analyse et du cassage des communications sécurisées. La cryptanalyse classique mêle une intéressante combinaison de raisonnement analytique, d'application d'outils mathématiques, de découverte de

redondances, de patience, de détermination, et de chance. Les cryptanalystes sont aussi appelés *attaquants*.

La *cryptologie* embrasse à la fois la cryptographie et la cryptanalyse.

La cryptographie forte

"Il y a deux sortes de cryptographie dans ce monde: la cryptographie qui empêchera votre petite sœur de lire vos fichiers, et la cryptographie qui empêchera les grands gouvernements de lire vos fichiers. Ce livre traite de la seconde."

– Bruce Schneier, Applied Cryptography: Protocols, Algorithms, and Source Code in C.

PGP traite aussi de la seconde sorte de cryptographie.

La cryptographie peut être *forte* ou *faible*, comme expliqué ci-dessus. La force de la cryptographie est mesurée par le temps et les ressources qui seraient nécessaires pour retrouver le texte clair. Le résultat de la *cryptographie forte* est un texte chiffré qui est très difficile à déchiffrer sans la possession de l'outil de déchiffrement approprié. A quel point est-ce difficile? Même en utilisant toute la puissance informatique disponible dans le monde aujourd'hui à plein temps – même avec un milliard d'ordinateurs effectuant chacun un milliard de vérifications à la seconde – il serait impossible de déchiffrer le résultat d'une cryptographie forte avant la fin de l'univers.

Certains penseront, alors, que la cryptographie forte devrait tenir plutôt bien même contre un cryptanalyste extrêmement déterminé. Qui peut le dire vraiment? Personne n'a pu prouver que le plus fort chiffrement qu'on puisse se procurer aujourd'hui tiendra devant la puissance informatique de demain. Cependant, la cryptographie forte employée par PGP est la meilleure disponible aujourd'hui. La vigilance et le conservatisme vous protégeront mieux, toutefois, que toute prétention d'impénétrabilité.

Comment fonctionne la cryptographie?

Un *algorithme cryptographique*, ou *chiffre*, est une fonction mathématique utilisée dans le processus de chiffrement et de déchiffrement. Un algorithme cryptographique fonctionne en combinaison avec une *clé* – un mot, un nombre, ou une phrase – pour chiffrer le texte clair. Le même texte clair se chiffre en un texte chiffré différent si l'on utilise des clés différentes. La sécurité des données chiffrées est entièrement dépendante de deux choses: la force de l'algorithme cryptographique et le secret de la clé.

Un algorithme cryptographique, plus toutes les clés possibles et tous les protocoles qui le font fonctionner constitue un *cryptosystème*. PGP est un cryptosystème.

Cryptographie conventionnelle

Dans la cryptographie conventionnelle, aussi appelée chiffrement à *clé secrète* ou à *clé symétrique*, une [seule et même] clé est utilisée à la fois pour le chiffrement et le déchiffrement. Le Data Encryption Standard (DES) est un exemple de cryptosystème conventionnel qui est largement employé par le Gouvernement fédéral américain. La Figure 1-2 est une illustration du processus du chiffrement conventionnel.

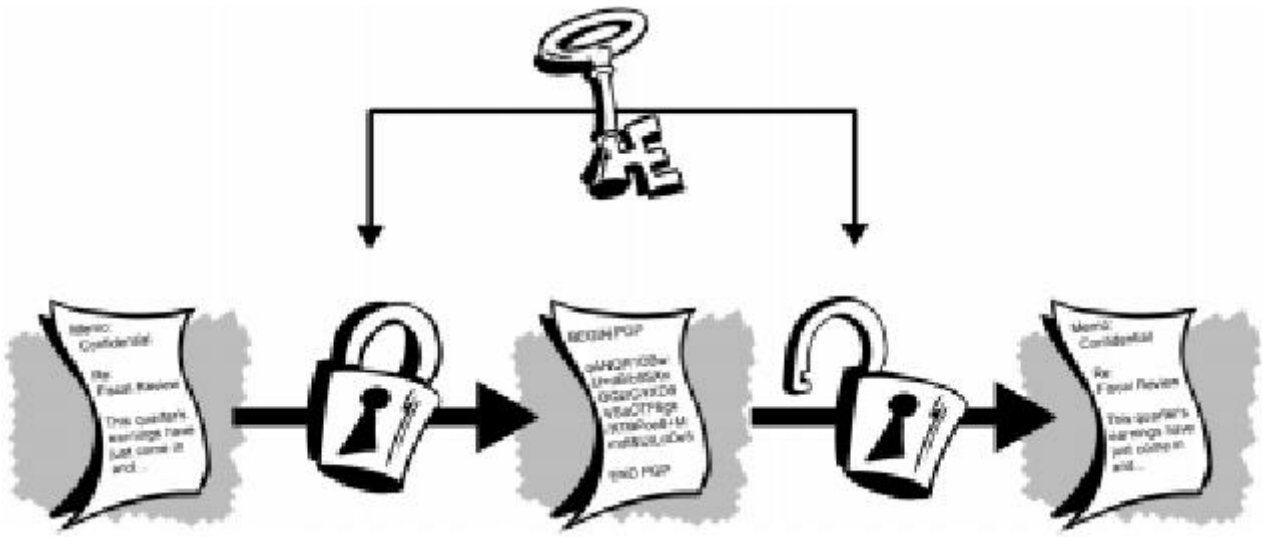

Figure 1-2. Chiffrement conventionnel

Le chiffre de César

Un exemple extrêmement simple de chiffrement conventionnel est un chiffre à substitution. Un chiffre à substitution remplace un morceau d'information par un autre. Le plus souvent, cela est effectué en décalant des lettres de l'alphabet. Deux exemples sont l'anneau décodeur secret du "Captain Midnight", que vous avez peut-être utilisé quand vous étiez enfant, et le chiffre de Jules César. Dans les deux cas, l'algorithme consiste en un décalage de l'alphabet, et la clé est le nombre de caractères à décaler. Par exemple, si nous codons le mot "SECRET" en utilisant une valeur de 3 pour la "clé de César", nous décalons l'alphabet de telle sorte que la troisième lettre en descendant (D) commence l'alphabet.

Donc en commençant avec

ABCDEFGHIJKLMNOPQRSTUVWXYZ

Et en décalant le tout de 3, vous obtenez

DEFGHIJKLMNOPQRSTUVWXYZABC

où D=A, E=B, F=C, et ainsi de suite.

En utilisant ce schéma, le texte clair, "SECRET" se chiffre comme "VHFUHW". Pour permettre à quelqu'un d'autre de lire le texte chiffré, vous lui dites que la clé est 3.

Evidemment, c'est de la cryptographie excessivement faible au regard des normes actuelles, mais bon, cela marchait pour César, et cela illustre aussi comment fonctionne la cryptographie conventionnelle.

Gestion de clé et chiffrement conventionnel

Le chiffrement conventionnel a des avantages. Il est très rapide. Il est particulièrement utile pour chiffrer des données qui ne vont *aller* nulle part. Cependant, le chiffrement conventionnel seul en tant que moyen de transmission de données sécurisées peut être assez onéreux simplement en raison de la difficulté de la distribution sécurisée de la clé.

Rappelez-vous un personnage de votre film d'espionnage préféré: l'homme qui porte une mallette menottée à son poignet. Qu'y a-t-il dans la mallette, justement? Ce n'est probablement pas le code de lancement du missile / la formule de la biotoxine / le plan d'invasion lui-même. C'est la *clé* qui déchiffrera les données secrètes.

Pour qu'un expéditeur et un destinataire communiquent de façon sûre en utilisant un chiffrement conventionnel, ils doivent se mettre d'accord sur une clé et la garder secrète entre eux. S'ils sont dans des lieux géographiques différents, ils doivent faire confiance à un messager, au Bat Phone, ou à un autre moyen de communication sûr pour empêcher la divulgation de la clé secrète pendant la transmission. Quiconque a entendu par hasard ou intercepté la clé en transit peut plus tard lire, modifier, et contrefaire toutes les informations chiffrées ou authentifiées avec cette clé. Du DES à l'anneau décodeur secret du "Captain Midnight", le problème continuel avec le chiffrement conventionnel est la *distribution de la clé*: comment donnerez-vous la clé au destinataire sans que personne ne puisse l'intercepter?

La cryptographie à clé publique

Les problèmes de distribution de clé sont résolus par la *cryptographie à clé publique*, dont le concept fut inventé par Whitfield Diffie et Martin Hellman en 1975. (Il y a maintenant des preuves que les Services secrets britanniques l'inventèrent quelques années avant Diffie et Hellman, mais en conservèrent le secret militaire – et ne firent rien avec.)[1]

La cryptographie à clé publique repose sur un schéma asymétrique qui utilise une *paire* de clés pour le chiffrement: une *clé publique*, qui chiffre les données, et une *clé privée* correspondante, aussi appelée *clé secrète*, qui sera utilisée pour le déchiffrement. Vous publiez largement votre clé publique, tout en gardant votre clé privée secrète. Toute personne en possession d'une copie de votre clé publique peut ensuite chiffrer des informations que vous seul pourrez lire. Même des gens que vous n'avez jamais rencontrés.

[1] J H Ellis, The Possibility of Secure Non-Secret Digital Encryption, CESG Report, Janvier 1970. [le CESG est l'Autorité nationale britannique pour l'utilisation officielle de cryptographie.]

Il est mathématiquement impossible de déduire la clé privée de la clé publique. Quiconque a une clé publique peut chiffrer des informations mais ne peut pas les déchiffrer. Seule la personne qui a la clé privée correspondante peut déchiffrer les informations.

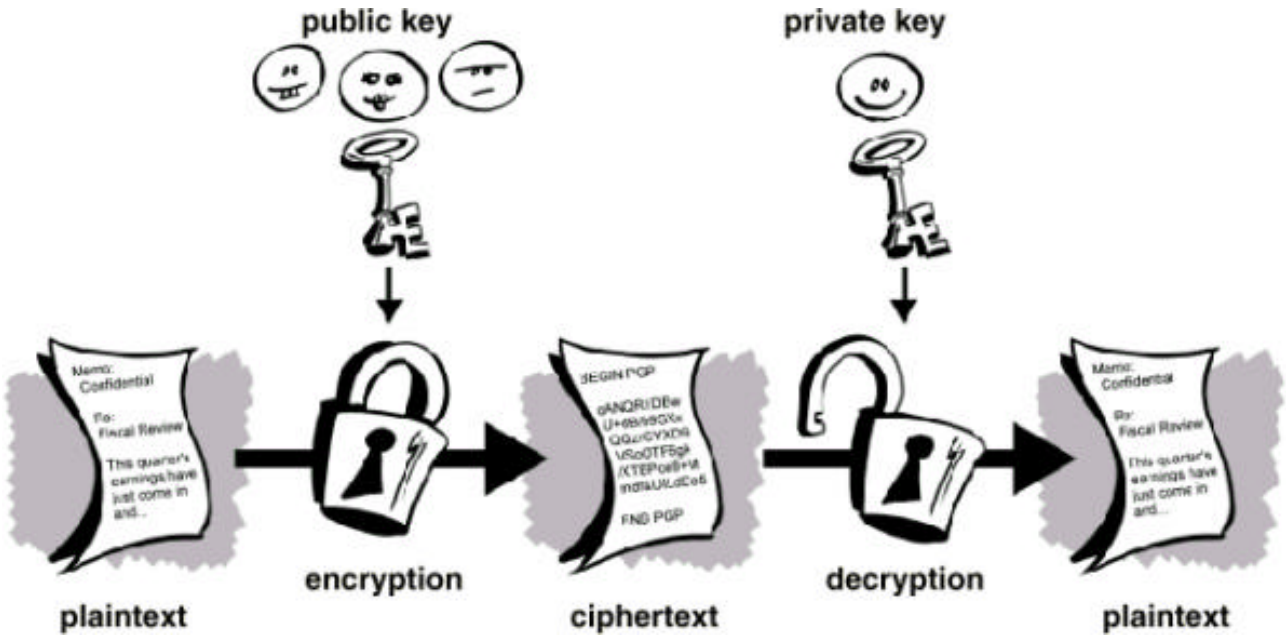

Figure 1-3. Chiffrement à clé publique

Le principal avantage de la cryptographie à clé publique est qu'elle permet à des gens qui n'ont pas d'accord de sécurité préalable d'échanger des messages de manière sûre. La nécessité pour l'expéditeur et le destinataire de partager des clés secrètes via un canal sûr est éliminée; toutes les communications impliquent uniquement des clés publiques, et aucune clé privée n'est jamais transmise ou partagée. Des exemples de cryptosystèmes à clé publique sont Elgamal (du nom de son inventeur, Taher Elgamal), RSA (du nom de ses inventeurs, Ron Rivest, Adi Shamir, et Leonard Aldeman), Diffie-Hellman (nommé ainsi, vous l'avez deviné, à cause de ses inventeurs), et DSA, l'Algorithme de Signature Digitale (inventé par David Kravitz).

Parce que la cryptographie conventionnelle était autrefois le seul moyen disponible pour transmettre des informations secrètes, le coût des canaux sûrs et de la distribution des clés réservait son utilisation uniquement à ceux qui pouvaient se l'offrir, comme les gouvernements et les grandes banques (ou les petits enfants avec leurs anneaux décodeurs secrets). Le chiffrement à clé publique est la révolution technologique qui permet aux masses d'accéder à la cryptographie forte. Vous vous souvenez du messager avec la mallette menottée à son poignet? Le chiffrement à clé publique le met à la retraite (probablement à son grand soulagement).

Comment fonctionne PGP

PGP combine à la fois les meilleures fonctionnalités de la cryptographie conventionnelle et de la cryptographie à clé publique. PGP est un *cryptosystème hybride*.

Quand un utilisateur chiffre du texte clair avec PGP, PGP compresse d'abord le texte clair. La compression de données économise du temps de transmission par modem et de l'espace disque et, ce qui est plus important, renforce la sécurité cryptographique. La plupart des techniques de cryptanalyse exploitent les redondances trouvés dans le texte clair pour craquer le texte chiffré. La compression réduit ces redondances dans le texte clair, ce qui augmente grandement la résistance à la cryptanalyse. (Les fichiers qui sont trop petits pour être compressés ou qui ne se compressent pas bien ne sont pas compressés.)

PGP crée ensuite une *clé de session*, qui est une clé secrète qui ne sert qu'une fois. Cette clé est un nombre aléatoire généré à partir des mouvements aléatoires de votre souris et des touches du clavier sur lesquelles vous tapez. Cette clé de session fonctionne avec un algorithme de chiffrement conventionnel très sûr et rapide qui chiffre le texte clair; le résultat est le texte chiffré. Une fois que les données sont chiffrées, la clé de session est elle-même chiffrée avec la clé publique du destinataire. Cette clé de session chiffrée par la clé publique est transmise avec le texte chiffré au destinataire.

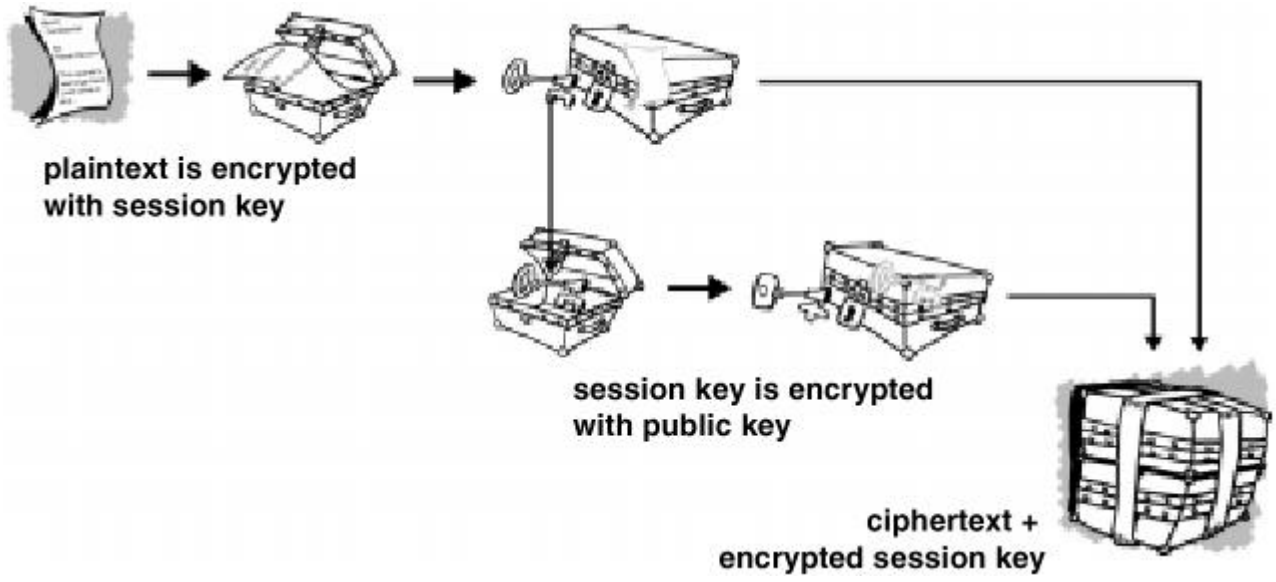

Figure 1-4. Comment fonctionne le chiffrement de PGP

Le déchiffrement fonctionne de la manière inverse. La copie de PGP du destinataire utilise la clé privée de celui-ci pour retrouver la clé de session temporaire, que PGP utilise ensuite pour déchiffrer le texte chiffré de manière conventionnelle.

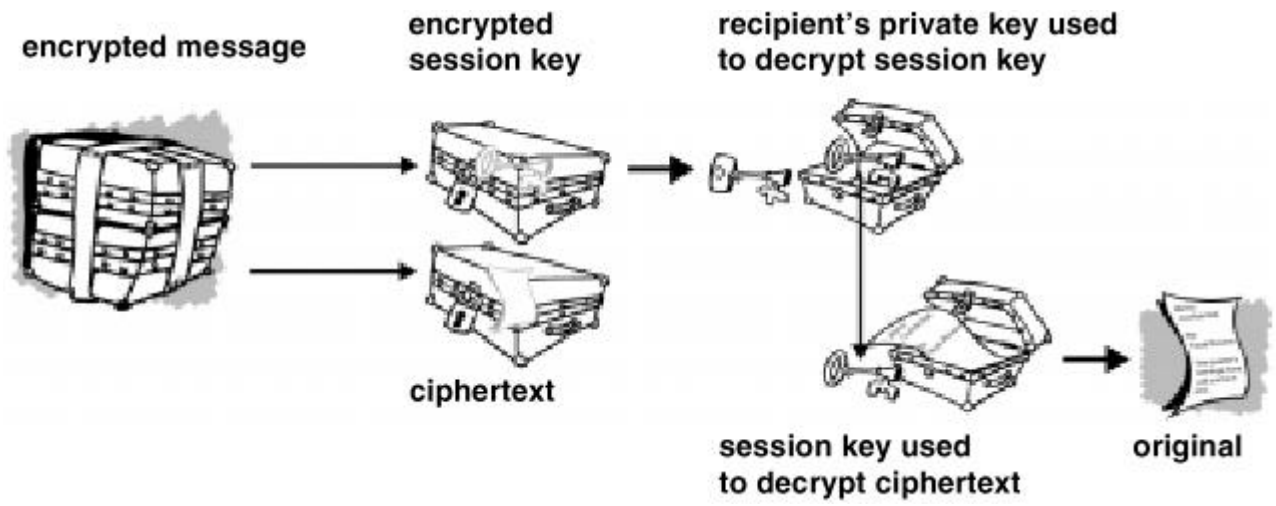

Figure 1-5. Comment fonctionne le déchiffrement de PGP

La combinaison des deux méthodes de chiffrement associe la commodité du chiffrement à clé publique avec la vitesse du chiffrement conventionnel. Le chiffrement conventionnel est environ 1000 fois plus rapide que le chiffrement à clé publique. Le chiffrement à clé publique fournit quant à lui une solution aux problèmes de distribution de la clé et de transmission des données. Utilisées toutes les deux, la performance et la distribution de la clé sont améliorées sans aucun sacrifice sur la sécurité.

Les clés

Une clé est une valeur qui est utilisée avec un algorithme cryptographique pour produire un texte chiffré spécifique. Les clés sont, à la base, de très, très, très grands nombres. La taille d'une clé se mesure en bits; et le nombre qui peut être représenté par une clé de 1024 bits est vraiment immense. En matière de cryptographie à clé publique, plus la clé est grande, plus le chiffrement est sûr.

Cependant, la taille d'une clé publique n'est absolument pas comparable avec la taille des clés secrètes employées en cryptographie conventionnelle. Une clé conventionnelle de 80 bits offre une sécurité équivalente à celle d'une clé publique de 1024 bits. Une clé conventionnelle de 128 bits équivaut à une clé publique de 3000 bits. Encore une fois, plus grande est la clé, plus grande est la sécurité, mais les algorithmes utilisés pour chaque type de cryptographie sont très différents, et donc, comparer les tailles de clés revient à comparer des pommes et des oranges.

Bien que la clé publique et la clé privée soient liées, il est très difficile de déduire la clé privée en partant de la seule clé publique; toutefois, il est toujours possible de déduire la clé privée si l'on dispose de suffisamment de temps et de puissance de calcul. Il est donc très important de choisir une clé d'une taille convenable; assez grande pour être sûre, mais suffisamment petite pour pouvoir être utilisée relativement rapidement. De surcroît, vous devez prendre en considération la nature des attaquants susceptibles de vouloir lire vos fichiers, leur détermination, le temps dont ils disposent, et leurs ressources éventuelles.

Les clés les plus grandes resteront cryptographiquement sûres pour une plus longue période. Si ce que vous chiffrez doit rester caché de nombreuses années,

vous voudrez sans doute utiliser une clé très grande. Qui peut dire en effet combien de temps il faudra pour casser votre clé en utilisant les ordinateurs du futur, plus rapides et plus efficaces? Il fut un temps où une clé symétrique de 56 bits était considérée comme extrêmement sûre.

Les clés sont stockées sous forme chiffrée. PGP stocke les clés dans deux fichiers sur votre disque dur; l'un pour les clés publiques et l'autre pour les clés privées. Ces fichiers sont appelés des *trousseaux de clés*. Quand vous utiliserez PGP, vous ajouterez les clés publiques de vos correspondants à votre trousseau public. Vos clés privées sont stockées dans votre trousseau privé. Si vous perdez votre trousseau privé, vous serez incapable de déchiffrer tout message destiné à l'une des clés qui était dans ce trousseau.

Signatures numériques

Un des avantages majeurs de la cryptographie à clé publique est qu'elle procure une méthode permettant d'utiliser des *signatures numériques*. Les signatures numériques permettent à la personne qui reçoit une information de contrôler l'authenticité de son origine, et également de vérifier que l'information en question est intacte. Ainsi, les signatures numériques des systèmes à clé publique permettent *l'authentification* et le *contrôle d'intégrité* des données. Une signature numérique procure également la *non répudiation*, ce qui signifie qu'elle empêche l'expéditeur de contester ultérieurement qu'il a bien émis cette information. Ces éléments sont au moins aussi importants que le chiffrement des données, sinon davantage.

Une signature numérique a le même objet qu'une signature manuelle. Toutefois, une signature manuelle est facile à contrefaire. Une signature numérique est supérieure à une signature manuelle en ce qu'elle est pratiquement impossible à contrefaire et, de plus, elle atteste le contenu de l'information autant que l'identité du signataire.

Certaines personnes utilisent les signatures plus qu'elles n'utilisent le chiffrement. Par exemple, vous pouvez vous moquer que quelqu'un puisse savoir que vous venez de déposer 10.000 F sur votre compte bancaire, mais vous voudrez être absolument certain que c'est bien avec votre banquier que vous avez traité.

La méthode de base utilisée pour créer des signatures numériques est illustrée sur la Figure 1-6. Au lieu de chiffrer l'information en utilisant la clé publique d'autrui, vous la chiffrez avec votre propre clé privée. Si l'information peut être déchiffrée avec votre clé publique, c'est qu'elle provient bien de vous.

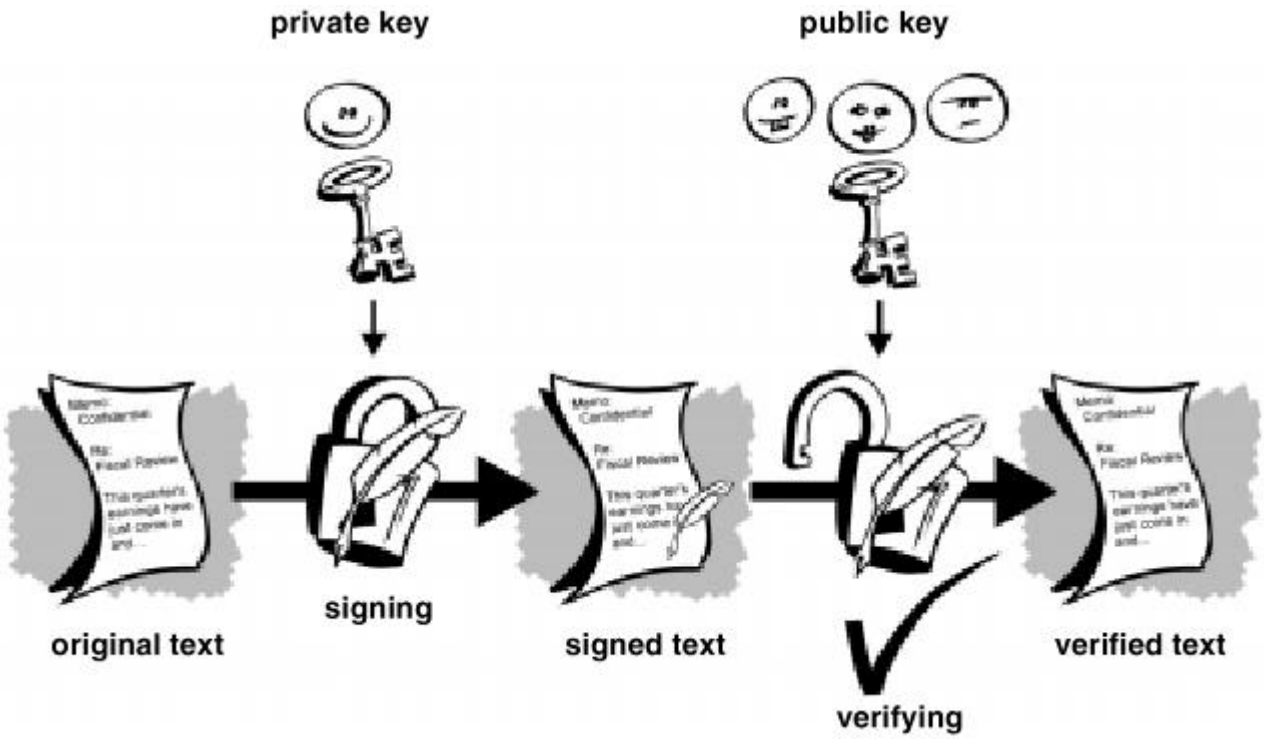

Figure 1-6. Signature numérique simple

Fonctions de hachage

Le système décrit ci-dessus comporte des inconvénients. Il est lent, et il produit un volume énorme de données – il double au minimum la taille de l'information originale. Une amélioration de ce concept est l'addition d'une *fonction de hachage* à sens unique dans le processus. Une fonction de hachage à sens unique utilise une entrée de longueur variable – dans notre cas, un message de n'importe quelle longueur, jusqu'à des milliers ou millions de bits – et produit une sortie de longueur fixe, par exemple 160 bits. La fonction de hachage assure que, si l'information était changée en quoi que ce soit – même d'un seul bit – une sortie totalement différente serait produite.

PGP applique une fonction de hachage cryptographiquement robuste, sur le texte clair que l'utilisateur veut signer. Ceci génère comme résultat une donnée de longueur fixe appelée *contraction de message*. (Encore une fois, toute modification du contenu du message produirait un condensé totalement différent.)

Ensuite, PGP utilise le condensé et la clé privée pour créer la "signature". PGP transmet la signature et le texte clair ensemble. A la réception du message, le destinataire utilise PGP pour recalculer le condensé, et le comparer avec celui reçu avec le message, ce qui permet de vérifier la signature. PGP peut chiffrer le texte clair ou non; signer un texte clair est utile si certains destinataires ne sont pas désireux de vérifier la signature, ou pas équipés pour le faire.

Tant qu'une fonction de hachage sûre est utilisée, il n'y a aucun moyen de recopier la signature de quelqu'un sur un document pour l'attacher à un autre, ni d'altérer en quoi que ce soit un document signé. Le plus petit changement dans un document signé provoquerait l'échec de la vérification de la signature.

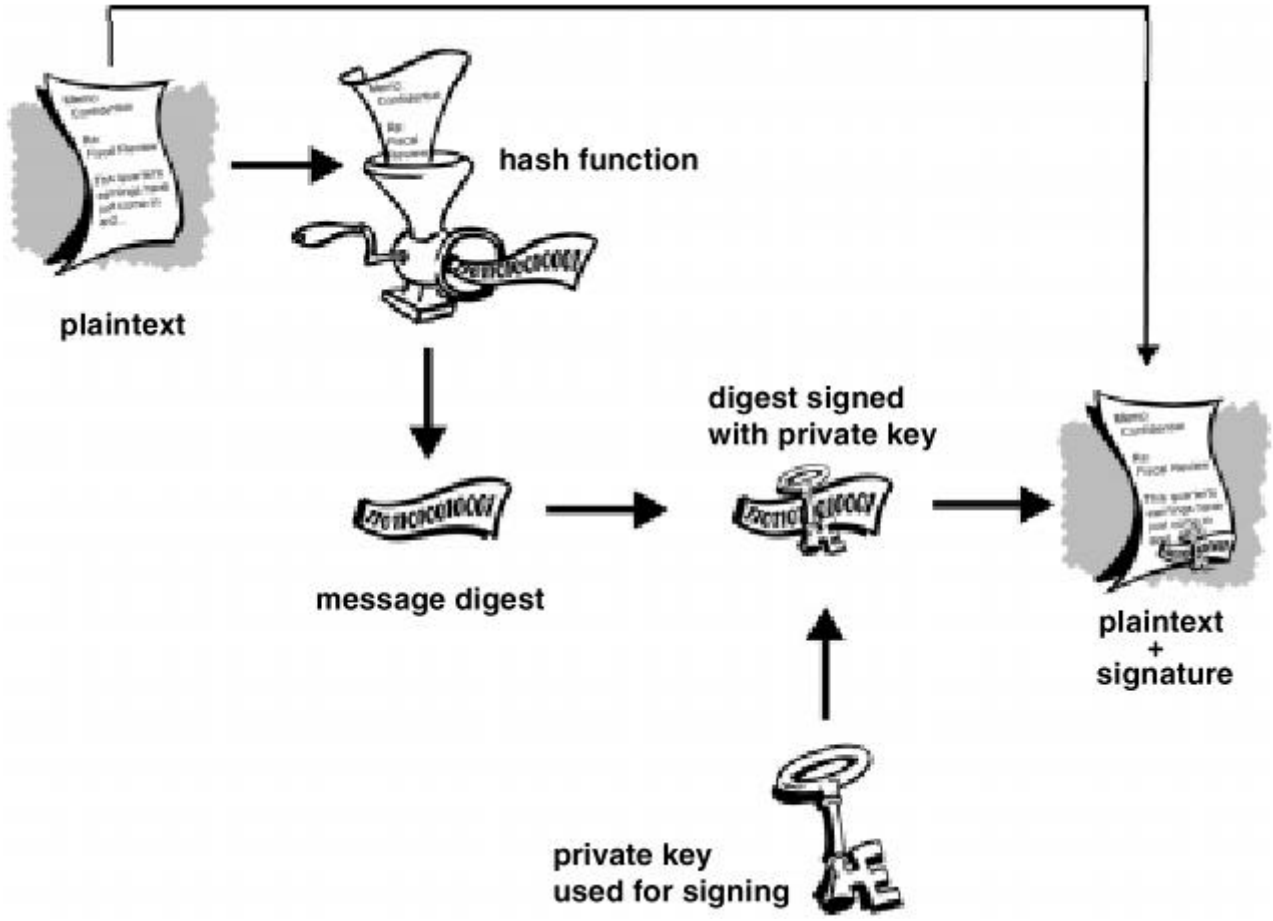

Figure 1-7. Signature numérique sécurisée

Les signatures numériques jouent un rôle majeur dans l'authentification et la *validation* des clés PGP des autres utilisateurs.

Certificats numériques

Un problème, avec les cryptosystèmes à clés publiques, est que les utilisateurs doivent être constamment vigilants pour s'assurer qu'il chiffrent leurs messages en utilisant la véritable clé de leur destinataire. Dans un environnement où l'on peut échanger des clés à travers des serveurs publics, les attaques utilisant une *personne interposée* sont un danger potentiel. Dans ce type d'attaque, un imposteur fournit une clé bidon portant le nom et l'identifiant d'utilisateur du destinataire réel des messages de l'utilisateur. Les données chiffrées pour – et interceptées par – le vrai propriétaire de cette fausse clé, seront tombées en de mauvaises mains.

Dans un environnement de clés publiques, il est vital que vous vous assuriez que la clé publique que vous vous apprêtez à utiliser pour chiffrer un message appartient bien au destinataire désiré, et n'est pas une contrefaçon. Vous pourriez vous limiter à utiliser les clés publiques qui vous ont été remises physiquement, de la main à la main, par leur propriétaire. Mais supposez que vous deviez échanger des informations avec des gens que vous n'avez jamais rencontrés; comment vous assurer que vous en possédez les véritables clés?

Les *certificats numériques* [ou *signatures*], ou *certs* simplifient la tâche d'établir la réelle appartenance d'une clé à son propriétaire supposé.

Le dictionnaire de Webster définit un *certificat* comme "un document contenant une affirmation certifiée, spécialement quant à la véracité de quelque chose". Un certificat est une sorte de pièce d'identité. Comme par exemple votre passeport, votre carte de Sécurité Sociale, ou votre extrait de naissance. Chacun de ces certificats contient des informations vous identifiant, et la signature d'une autorité qui certifie cette identité. Certaines de ces pièces d'identité, comme votre permis de conduire, sont suffisamment importantes pour que vous preniez soin de ne pas les perdre, pour éviter que quelqu'un ne puisse usurper votre identité.

Un certificat numérique fonctionne en gros comme une pièce d'identité matérielle. Un certificat numérique est une information attachée à une clé publique, et qui permet de vérifier que cette clé est authentique, ou *valide*. Les certificats numériques sont utilisés pour contrecarrer les tentatives de substituer une clé falsifiée à la clé véritable.

Un certificat numérique comporte trois éléments:

- Une clé publique
- Une information de certification ("l'identité" de l'utilisateur, comme son nom, son adresse e-mail, etc.)
- Une ou plusieurs signatures numériques

L'objet de la signature numérique sur un certificat est de garantir que les informations de certification ont été contrôlées par une autre personne ou organisme. La signature numérique ne garantit pas l'authenticité du certificat complet, elle garantit seulement que les informations d'identité ainsi signées correspondent bien à la clé publique *à laquelle* elles sont attachées.

Bien que certains experts en sécurité considèrent qu'il n'est pas de bonne pratique de mélanger des informations d'identité personnelles et professionnelles sur une même clé, mais qu'il vaut mieux utiliser une clé séparée pour chacune, vous rencontrerez néanmoins des certificats contenant une clé publique à laquelle sont associées plusieurs identités diverses (comme par exemple, le nom complet d'une utilisatrice avec son adresse e-mail professionnelle, son surnom avec une adresse e-mail personnelle, un "nom de jeune fille" avec une adresse e-mail universitaire – le tout dans un seul et même certificat). La liste des signatures validant chacune de ces identités peut être différente; chaque signature n'atteste l'authenticité que de l'une de ces identités, mais pas forcément des trois.

Par exemple, supposez que votre collègue Alice vous demande de signer sa clé. Vous la recherchez sur le serveur, et vous vous apercevez qu'elle a deux informations d'identité distinctes attachées à cette clé. La première est "Alice Petucci <alice@societesure.com>". La seconde est "Cleopatra <cleo@cheops.org>". En fonction de votre connaissance d'Alice, vous choisirez peut-être de ne signer que son identité telle que vous la connaissez au bureau.

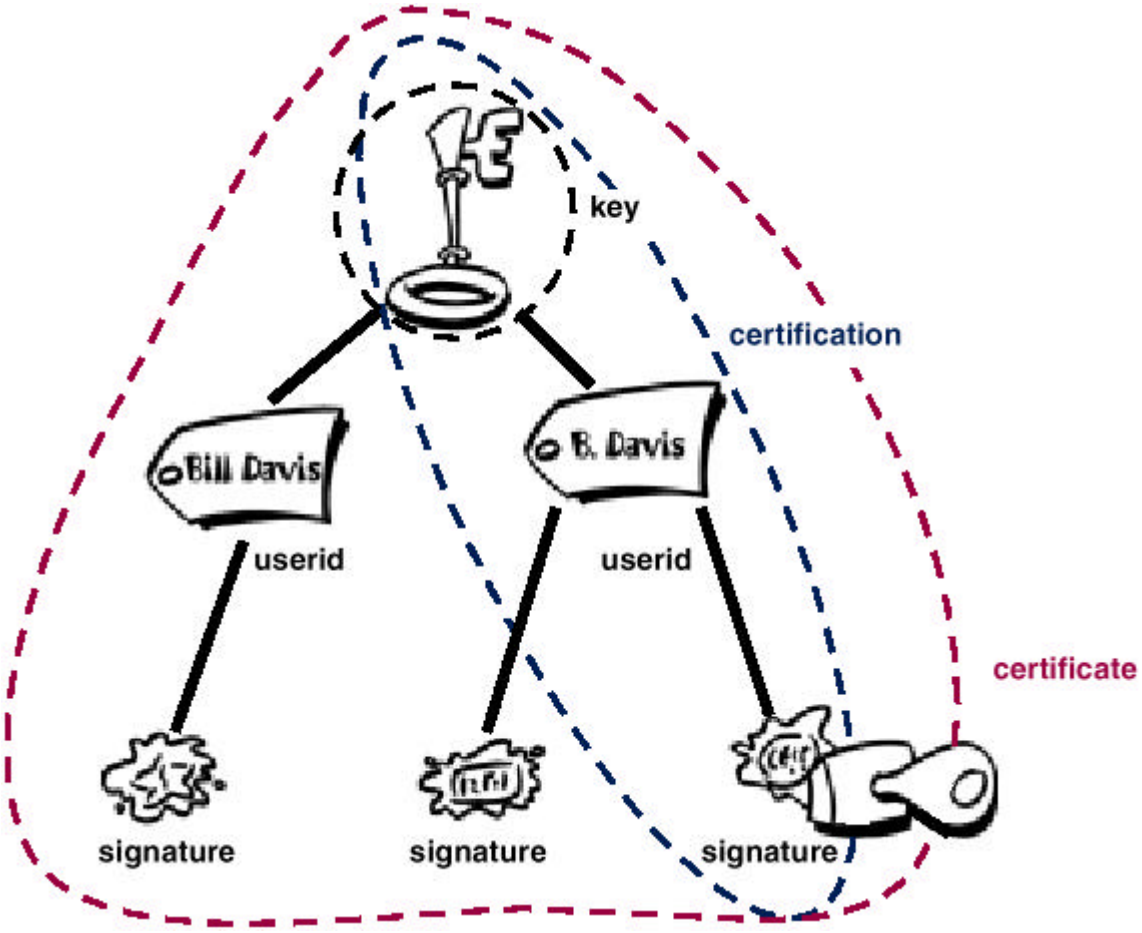

Figure 1-8. Anatomie d'un certificat

Validité et confiance

Chaque utilisateur d'un système à clés publiques court le risque de confondre une clé falsifiée (certificat) avec une véritable. La *validité* est la créance qu'une clé donnée appartient réellement à son propriétaire supposé. La validité est essentielle dans un environnement à clés publiques où l'on doit à chaque instant être en mesure d'établir l'authenticité d'un certificat donné.

Quand vous vous êtes assuré qu'un certificat qui appartient à autrui est valide, vous pouvez en signer la copie dans votre trousseau pour attester le fait que vous avez contrôlé ce certificat, et qu'il est valide. Si vous voulez que d'autres personnes puissent savoir que vous avez donné à ce certificat votre propre approbation, vous pouvez exporter la signature que vous lui avez apposée vers un serveur de certificats, afin que d'autres puissent la voir.

Certaines sociétés désignent une ou plusieurs *Autorités de Certification (A.C.)*, dont le rôle est de vérifier la validité de tous les certificats à l'intérieur de l'organisation, et de signer ceux qui sont valides. L'A.C. est le grand Manitou de la validation à l'intérieur de l'organisation, en qui tout le monde a confiance et, dans certains environnements à clés publiques, aucun certificat n'est considéré comme valide s'il n'a pas été certifié par l'A.C.

Contrôle de la validité

Une manière d'établir la validité des certificats est de suivre une procédure manuelle. Il y a plusieurs façons d'accomplir cela. Vous pourriez demander à votre futur correspondant de vous remettre physiquement, de la main à la main, une copie de sa clé publique. Mais c'est souvent impraticable et inefficace.

Une autre méthode est de contrôler manuellement l'*empreinte numérique* du certificat. De la même façon que les empreintes digitales de chaque être humain sont uniques, chaque certificat PGP possède une "empreinte numérique" unique. Cette empreinte est un hachage du certificat de l'utilisateur, et apparaît comme l'une des propriétés de ce certificat. Vous pouvez vérifier qu'un certificat est valide en appelant le propriétaire de la clé (de manière à ce que vous soyez à l'origine de l'appel) et en lui demandant de vous lire au téléphone l'empreinte numérique de sa clé, de manière à pouvoir vérifier que celle-ci correspond bien à la copie que vous avez en votre possession. Cela fonctionne si vous connaissez la voix de votre correspondant, mais comment vérifier l'identité de quelqu'un que vous ne connaissez pas? Certaines personnes font imprimer les empreintes numériques de leur clé sur leur carte de visite professionnelle pour cette raison même.

Une autre manière d'établir la validité du certificat de quelqu'un est de *faire confiance* à quelqu'un d'autre qui aura lui-même effectué le processus de validation.

Une A.C., par exemple, est tenue de s'assurer soigneusement qu'un certificat appartient bien à son propriétaire supposé, avant de lui apposer sa signature de validité. Quiconque a confiance en l'A.C. considérera automatiquement comme valides tous les certificats validés par cette A.C.

Instituer la confiance

Vous validez des *clés*, mais vous avez confiance en des *personnes*. Plus précisément, vous avez confiance en des personnes pour valider les clés d'autres personnes. Typiquement, à moins que le propriétaire d'une clé ne vous l'ait remise lui-même, vous devrez faire confiance à autrui pour déterminer si la clé est valide.

Méta-avals et avals de confiance

Dans la plupart des situations, les utilisateurs s'en remettent entièrement à leur confiance en l'A.C. pour établir la validité des certificats. Ceci signifie que chacun se repose sur l'A.C. pour effectuer la procédure manuelle complète de vérification à sa place. C'est possible dans la limite d'un certain nombre d'utilisateurs ou de lieux de travail, au delà desquels il ne sera plus possible à l'A.C. de maintenir le même niveau de qualité dans sa validation. Dans ce cas, il devient nécessaire d'ajouter des avals supplémentaires dans le système.

Une A.C. peut aussi être un *méta-aval*. Un méta-aval ne certifie pas seulement la validité des clés, mais il certifie de plus *la qualité d'autrui pour certifier des clés*. De la même manière que le roi confie ses sceaux à ses conseillers pour qu'ils puissent agir en son nom, le méta-aval délègue à d'autres la qualité d'agir comme *avals de confiance*. Ces avals de confiance peuvent à leur tour valider des clés, ce

qui leur donnera la même validité que si celles-ci avaient été directement validées par le méta-aval.

Toutefois, les avals de confiance ne peuvent pas à leur tour désigner d'autres avals.

Modèles de confiance

Dans des systèmes relativement fermés, comme à l'intérieur d'une société, il est facile de remonter l'arbre de la confiance jusqu'à sa racine: l'A.C. Toutefois, dans le monde réel, les utilisateurs doivent souvent communiquer avec des personnes qui se trouvent en dehors du cadre de leur entreprise, y compris des personnes qu'ils n'ont jamais personnellement rencontrées, comme des fournisseurs, des clients, des associés, et ainsi de suite. Etablir une ligne de confiance pour des gens qui n'ont pas été explicitement certifiés par une A.C. est difficile.

Les sociétés adoptent tel ou tel *modèle de confiance* donné, qui indique la procédure que devront suivre les utilisateurs pour établir la validité d'une clé. On trouve trois modèles différents:

- Confiance directe
- Confiance hiérarchisée
- Un réseau [ou toile d'araignée] de confiance

Confiance directe

La confiance directe est le modèle de confiance le plus simple. Dans ce modèle, un utilisateur considère qu'une clé est valide parce qu'il en connaît la provenance. Tous les cryptosystèmes utilisent cette forme de confiance d'une manière ou d'une autre. Par exemple, dans les navigateurs Web, les clés de l'Autorité de Certification "racine" sont directement crédibles parce qu'elles ont été fournies avec le programme par son éditeur. S'il existe une quelconque forme de hiérarchie, elle procède de ces certificats directement fiables.

Dans PGP, un utilisateur qui valide lui-même les clés de ses correspondants, et ne désigne jamais un autre certificat en tant qu'aval de confiance, utilise ce modèle de confiance directe.

Confiance hiérarchisée

Dans un système hiérarchisé, il y a un certain nombre de certificats "racines" à partir desquels s'étend l'arbre de la confiance. Ces certificats peuvent valider directement d'autres certificats, ou ils peuvent déléguer à d'autres certificats le pouvoir de certifier par eux-mêmes, en descendant une chaîne. Il faut voir ceci comme un grand arbre de confiance. Le certificat qui occupe la place d'une "feuille" dans cet arbre est vérifié en remontant le long d'une "branche", d'aval en aval, jusqu'à atteindre une racine de l'arbre: un certificat ultime pour lequel la confiance est directe.

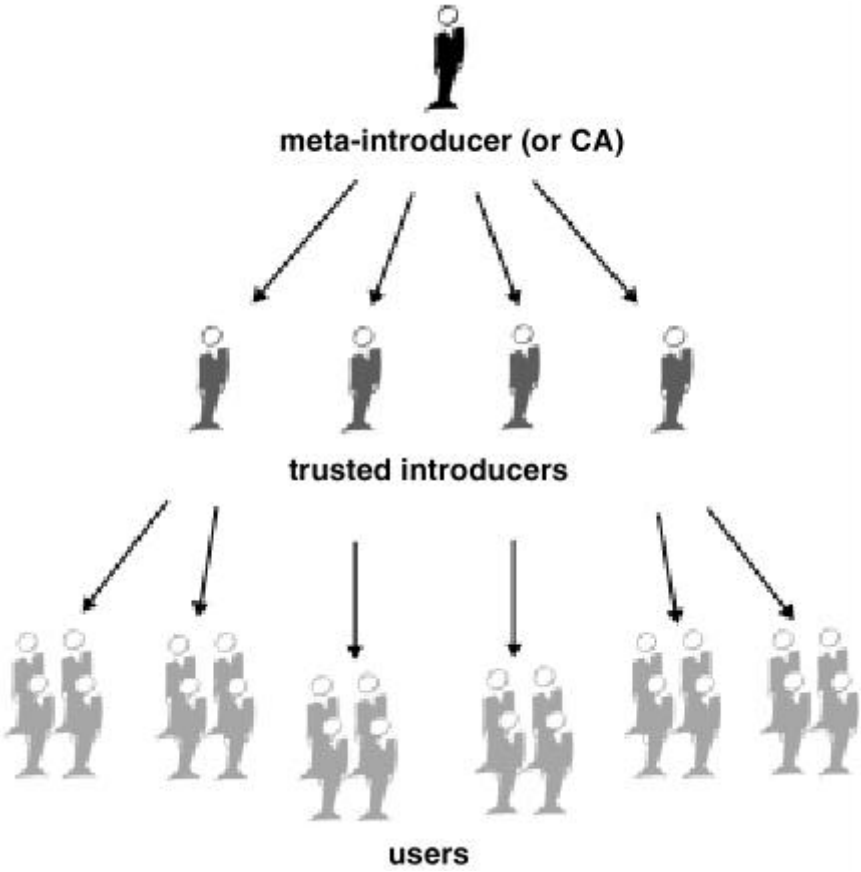

Figure 1-9. Confiance hiérarchisée

Réseau [ou toile d'araignée] de confiance

Un réseau de confiance embrasse les deux modèles précédents, en y ajoutant la notion de confiance perçue du point de vue de l'utilisateur (ce que l'on trouve dans le monde réel), et l'idée que la plus grande quantité d'information est la meilleure. C'est donc un modèle de confiance cumulatif. Un certificat peut être validé directement, ou à travers une branche remontant jusqu'à un méta-aval racine, ou par un groupe d'avals de confiance.

Peut-être avez-vous déjà entendu la formule "*six degrés de séparation*", qui suggère que toute personne dans le monde peut être mise en relation avec toute autre personne en utilisant un maximum de six autres personnes comme intermédiaires. Ceci constitue un réseau d'avals.

C'est aussi le concept de PGP à propos de la confiance. PGP utilise des signatures numériques comme forme de validation. Quand n'importe quel utilisateur signe la clé d'autrui, il devient lui-même un aval de cette clé. La continuation de ce processus aboutit à un *réseau de confiance*, ou *toile d'araignée de confiance*.

Dans un environnement PGP, *n'importe quel* utilisateur peut agir comme autorité de certification. N'importe quel utilisateur de PGP peut certifier la clé publique de n'importe quel autre utilisateur. Toutefois, cette certification n'a d'effet pour un autre utilisateur que si celui-ci considère l'aval comme fiable à ses propres yeux. (Ce qui veut dire: vous faites confiance à ma certification d'une clé seulement si

vous me considérez comme un aval de confiance. Dans le cas contraire, mon opinion quant à la validité d'une autre clé est ignorée.)

On trouve, stocké dans le trousseau public de chacun:

- L'indication de l'opinion de l'utilisateur quant à la validité de chaque clé particulière.
- L'indication de l'opinion de l'utilisateur quant à la crédibilité du propriétaire de chaque clé pour agir comme aval.

Vous indiquez, sur votre copie de ma clé, le crédit que vous accordez à mon jugement. C'est réellement un système basé sur la réputation: certaines personnes sont réputées donner des signatures crédibles, aussi de nombreuses personnes leur font-elles confiance pour attester de la validité d'autres clés.

Niveaux de confiance dans PGP

Le plus haut niveau de confiance dans une clé, la confiance *implicite*, est la confiance en votre propre paire de clés. PGP part de l'hypothèse que si vous possédez la clé privée, vous devez avoir confiance dans les actions de la clé publique qui lui est liée. Toutes les clés signées par votre clé, dont la confiance est implicite, sont donc considérées comme valides.

Il y a trois niveaux de confiance que vous pouvez assigner à la clé publique d'autrui:

- Confiance *complète*
- Confiance *marginale*
- Aucune confiance *(Untrusted)*

Pour rendre les choses encore plus confuses, il y a aussi trois degrés de validité:

- Valide
- Marginalement valide
- Invalide

Pour définir la clé d'autrui comme aval de confiance, vous:

1. Partez d'une clé valide, qui est soit:
    - Signée par vous-même
    - Signée par un autre aval de confiance

    et ensuite
2. Indiquez le niveau de confiance que vous accordez au propriétaire de cette clé.

Par exemple, supposons que votre trousseau contienne la clé d'Alice. Vous avez validé la clé d'Alice, ce que vous indiquez en signant celle-ci. Vous savez également qu'Alice est très pointilleuse avant de signer d'autres clés. Vous

assignez donc à sa clé votre confiance totale. Ceci fait d'Alice une Autorité de Certification à l'intérieur de votre trousseau. Si vous avez d'autres clés publiques signées par Alice, elles apparaîtront donc automatiquement comme valides.

PGP demande une signature de confiance complète, ou deux signatures différentes de confiance marginale, pour établir qu'une clé est valide. La méthode qu'emploie PGP en demandant deux signatures de confiance marginale est la même que celle qu'emploie un commerçant, lorsqu'il vous demande deux pièces d'identité différentes [avant d'accepter un gros chèque]. Vous pouvez considérer qu'Alice est assez crédible, et que Bob lui aussi est assez crédible. Chacun pris individuellement risque un jour par accident de signer une clé contrefaite, donc vous pouvez choisir de ne pas placer en chacun d'eux une confiance totale. Toutefois, les chances pour que ces deux personnes aient toutes deux signé la même clé contrefaite sont relativement faibles.

Qu'est-ce qu'une phrase secrète?

La plupart des gens sont familiers du fait de contrôler l'accès à un ordinateur avec un *mot de passe*, qui est une chaîne de caractères unique que l'utilisateur tape comme code d'identification.

Une *phrase secrète* est une version plus longue d'un mot de passe et, en théorie, plus sûre. Typiquement composée de plusieurs mots, une phrase secrète est moins vulnérable à des *attaques par dictionnaire* standards, où l'attaquant essaie l'ensemble des mots d'un dictionnaire lors de sa tentative de trouver votre mot de passe. Les meilleures phrases secrètes sont relativement longues et complexes et contiennent une combinaison de majuscules et de minuscules, de chiffres et de signes de ponctuation.

PGP utilise une phrase secrète pour chiffrer votre clé privée sur votre machine. Votre clé privée est chiffrée sur le disque en utilisant un hachage de votre phrase secrète comme clé. Vous utiliserez la phrase secrète pour déchiffrer votre clé privée afin de l'utiliser. Une phrase secrète doit être facile à retenir pour vous, tout en étant difficile à deviner par autrui. Cela doit être quelque chose de solidement ancré dans votre mémoire à long terme, plutôt que quelque chose que vous venez de construire à partir de rien. Pourquoi? Parce que **si vous oubliez votre phrase secrète, vous êtes coincé.** Votre clé privée est totalement et absolument inutile sans votre phrase secrète, et on ne peut absolument rien y faire. Vous souvenez-vous de la citation déjà vue dans ce chapitre? PGP est un cryptosystème capable de protéger vos fichiers contre les grands gouvernements. Il sera certainement en mesure de vous faire obstacle également. Gardez ceci en mémoire le jour où vous déciderez de changer votre phrase secrète pour la chute de cette histoire drôle dont vous ne vous souvenez jamais.

Scission de clés

Certains disent qu'un secret n'est plus un secret, dès qu'il est connu de plus d'une personne. Partager une clé privée pose le même problème. Bien que ce ne soit pas une pratique généralement recommandée, partager une clé privée est parfois

nécessaire. Les *Clés de Signature d'Entreprise*, par exemple, sont des clés privées utilisées par des sociétés pour signer – par exemple – des documents juridiques, des informations personnelles sensibles, ou des communiqués de presse pour en authentifier l'origine. Dans un tel cas, il est intéressant que plusieurs membres de l'entreprise aient accès à la clé. Toutefois, cela signifierait que chacune de ces personnes pourrait agir individuellement en engageant totalement la société.

Dans un tel cas, il est sage de *scinder* la clé en la partageant entre plusieurs personnes, de manière à ce que plus de deux personnes doivent présenter leur fragment de la clé, afin de pouvoir reconstituer celle-ci dans un état utilisable. Si trop peu de fragments de la clé sont disponibles, celle-ci sera inutilisable.

Par exemple, on peut scinder une clé en trois segments, et exiger au moins deux d'entre eux pour reconstituer la clé; ou scinder une clé en deux et exiger les deux segments. Si une connexion par réseau sécurisé est utilisée pendant la reconstitution de la clé, les dépositaires des fragments n'ont même pas besoin d'être physiquement présents pour pouvoir reconstituer celle-ci.

Détails techniques

Ce chapitre a fourni une introduction de haut niveau en matière de concepts et de terminologie cryptographiques. Dans le Chapitre 2, Phil Zimmermann, le créateur de PGP, conduit une discussion approfondie sur la confidentialité, les détails techniques du fonctionnement de PGP, incluant les différents algorithmes qu'il utilise, ainsi que sur la variété des attaques possibles et les moyens de s'en protéger.

Pour plus d'information sur la cryptographie, veuillez vous reporter aux ouvrages référencés dans le chapitre "Lectures recommandées" de la préface.

Phil Zimmermann sur PGP 2

Ce chapitre contient une introduction et des informations de référence à propos de la cryptographie et de PGP, écrites par Phil Zimmermann.

Pourquoi j'ai écrit PGP

" Quoi que vous ferez, ce sera insignifiant, mais il est très important que vous le fassiez." – Mahatma Gandhi

[*"Whatever you do will be insignificant, but it is very important that you do it."* – Mahatma Gandhi.]

C'est personnel. C'est privé. Et cela ne regarde personne d'autre que vous. Vous pouvez être en train de préparer une campagne électorale, de discuter de vos impôts, ou d'avoir une romance secrète. Ou vous pouvez être en train de communiquer avec un dissident politique dans un pays répressif. Quoi qu'il en soit, vous ne voulez pas que votre courrier électronique (e-mail) ou vos documents confidentiels soient lus par quelqu'un d'autre. Il n'y a rien de mal à défendre votre intimité. L'intimité est aussi fondamentale que la Constitution.

Le droit à la vie privée est disséminé implicitement tout au long de la Déclaration des Droits. Mais quand la Constitution des Etats-Unis a été élaborée, les Pères Fondateurs ne virent aucun besoin d'expliciter le droit à une conversation privée. Cela aurait été ridicule. Il y a deux siècles, toutes les conversations étaient privées. Si quelqu'un d'autre était en train d'écouter, vous pouviez aller tout simplement derrière la grange et y tenir une conversation. Personne ne pouvait vous écouter sans que vous le sachiez. Le droit à une conversation privée était un droit naturel, non pas seulement au sens philosophique, mais au sens des lois de la physique, étant donnée la technologie de l'époque.

Mais avec l'arrivée de l'âge de l'information, commençant avec l'invention du téléphone, tout cela a changé. Maintenant, la plupart de nos conversations sont acheminées électroniquement. Cela permet à nos conversations les plus intimes d'être divulguées sans que nous le sachions. Les appels des téléphones cellulaires peuvent être enregistrés par quiconque possède une radio. Le courrier électronique, envoyé à travers Internet, n'est pas plus sûr que les appels de téléphone cellulaire. L'e-mail est en train de remplacer rapidement le courrier classique, devenant la norme pour tout le monde, et non plus la nouveauté qu'il était par le passé. Et l'e-mail peut être systématiquement et automatiquement fouillé à la recherche de mots clés, sur une grande échelle, sans que cela soit détecté. C'est comme la pêche aux filets dérivants.

Peut-être pensez-vous que le courrier électronique que vous recevez est assez légitime pour que le chiffrement ne se justifie pas. Si vous êtes vraiment un citoyen au-dessus de tout soupçon, pourquoi n'envoyez-vous pas toujours votre correspondance papier sur des cartes postales? Pourquoi ne vous soumettez-vous pas aux tests de consommation de drogue sur simple demande? Pourquoi exigez-

vous un mandat de perquisition pour laisser la police fouiller votre maison? Essayez-vous de cacher quelque chose? Si vous cachez votre courrier dans des enveloppes, cela signifie-t-il que vous êtes un [élément] subversif ou un trafiquant de drogue, ou peut-être un paranoïaque aigu? Est-ce que les citoyens honnêtes ont un quelconque besoin de chiffrer leurs e-mails?

Que se passerait-il si tout le monde estimait que les citoyens honnêtes devraient utiliser des cartes postales pour leur courrier? Si un non-conformiste s'avisait alors d'imposer le respect de son intimité en utilisant une enveloppe, cela attirerait la suspicion. Peut-être que les autorités ouvriraient son courrier pour voir ce que cette personne cache. Heureusement, nous ne vivons pas dans ce genre de société car chacun protège la plupart de son courrier avec des enveloppes. Aussi personne n'attire la suspicion en protégeant son intimité avec une enveloppe. La sécurité vient du nombre. De la même manière, ce serait excellent si tout le monde utilisait la cryptographie de manière systématique pour tous ses e-mails, qu'ils soient innocents ou non, de telle sorte que personne n'attirerait la suspicion en protégeant la confidentialité de ses e-mails par la cryptographie. Voyez cela comme une forme de solidarité.

Jusqu'à aujourd'hui, si le Gouvernement désirait violer l'intimité de citoyens ordinaires, il devait consentir une certaine dépense d'argent et de travail pour intercepter, ouvrir et lire les lettres. Ou il devait écouter et si possible transcrire le contenu des conversations téléphoniques, du moins avant que la technologie de la reconnaissance vocale automatique soit disponible. Cette méthode, coûteuse en travail, n'était pas praticable sur une grande échelle. Cela était fait seulement dans les cas importants, quand cela en valait la peine.

En 1991 aux Etats-Unis, le projet de loi 266 du Sénat, un texte anti criminalité, comportait une disposition troublante cachée à l'intérieur du texte. Si cette résolution était devenue une véritable loi, cela aurait contraint les fabricants d'équipements de communications sécurisées à insérer des "portes dérobées" spéciales dans leurs produits, de telle sorte que le gouvernement puisse lire les messages chiffrés par n'importe qui. Le texte disait: "La recommandation du Sénat est que les fournisseurs de services de communications électroniques et les fabricants d'équipements de communication électronique devront s'assurer que les systèmes de communication permettent au gouvernement d'obtenir le contenu en clair des communications vocales, des données, et des autres communications dans les cas prévus par la loi". Ce fut cette loi qui me conduisit à publier PGP gratuitement sous forme électronique cette année-là, peu de temps avant que la mesure ne soit retirée après de vigoureuses protestations des groupes de défense des libertés civiles et des groupes industriels.

Le "Digital Telephony bill" de 1994 a fait obligation aux compagnies de téléphone d'installer des dispositifs d'interception à distance dans leurs commutateurs centraux, créant une nouvelle infrastructure technologique pour cette interception "pointer et cliquer", de telle sorte que les agents fédéraux n'auront plus à sortir et attacher des pinces crocodiles sur les lignes de téléphone. Maintenant, ils auront la possibilité de rester assis dans leur quartier général à Washington et d'écouter vos appels téléphoniques. Bien sûr, les lois requièrent encore une réquisition judiciaire pour une interception. Mais alors que les infrastructures techniques peuvent durer des générations, les lois et politiques changent du jour au lendemain. Une fois que

l'infrastructure des communications est optimisée pour la surveillance, une modification dans les conditions politiques peut conduire à abuser de ce pouvoir fondé sur de nouvelles bases. Les conditions politiques peuvent se modifier avec l'élection d'un nouveau gouvernement, ou peut-être même encore plus brusquement après l'attentat à la bombe contre un immeuble fédéral.

Un an après que le "Digital Telephony bill" de 1994 soit passé, le FBI dévoila des plans pour exiger des compagnies de téléphone d'intégrer dans leurs infrastructures la capacité d'intercepter simultanément 1 % de tous les appels téléphoniques dans toutes les grandes villes américaines. Cela représentait une multiplication par plus de mille du nombre d'appels qui peuvent être interceptés. Dans les années précédentes, il y avait eu seulement à peu près un millier de réquisitions d'interceptions judiciaires par an aux Etats-Unis, à la fois au niveau fédéral, au niveau des Etats et au niveau local. Il est difficile de savoir comment le gouvernement pourrait ne serait-ce qu'employer assez de juges pour signer assez d'ordres d'interception pour intercepter 1 % de tous les appels téléphoniques, encore moins embaucher assez d'agents fédéraux pour s'asseoir et écouter tout ce trafic en temps réel. La seule façon plausible de traiter toute cette quantité de trafic est une application massivement Orwellienne de la technologie de reconnaissance vocale pour passer au crible tout cela, à la recherche de mots clés intéressants ou de la voix d'un interlocuteur particulier. Si le gouvernement ne trouve pas la cible dans le premier échantillon de 1 %, les interceptions peuvent être étendues à un 1 % différent jusqu'à ce que la cible soit trouvée, ou jusqu'à ce que la ligne de téléphone de chacun ait été inspectée à la recherche de trafic subversif. Le FBI dit qu'ils ont besoin de cette capacité pour prévoir le futur. Ce plan a provoqué un tel scandale qu'il a été retiré au Congrès, en peu de temps, en 1995. Mais le simple fait que le FBI ait été jusqu'à demander ces pouvoirs élargis révèle leur programme. Et la défaite de ce plan n'est pas si rassurante quand vous considérez que le "Digital Telephony bill" de 1994 avait aussi été retiré la première fois qu'il a été introduit, en 1993.

Les avancées technologiques ne permettent pas le maintien du statu quo, à partir du moment où la vie privée est concernée. Le statu quo est instable. Si nous ne faisons rien, des nouvelles technologies donneront au gouvernement de nouvelles capacités de surveillance dont Staline n'aurait jamais pu rêver. La seule façon de préserver la vie privée à l'ère de l'information est de recourir à la cryptographie sûre.

La crainte d'abus de pouvoir du gouvernement n'est pas la seule raison pour vouloir recourir à la cryptographie. Votre correspondance d'affaires peut être interceptée par des concurrents, le crime organisé, ou des gouvernements étrangers. Plusieurs gouvernements, par exemple, admettent utiliser leurs services d'écoutes contre les compagnies d'autres pays pour donner à leurs propres sociétés un avantage sur la concurrence. L'ironie est que les restrictions du gouvernement des Etats-Unis sur la cryptographie ont affaibli les défenses des entreprises américaines contre les services de renseignement étrangers et le crime organisé.

Le gouvernement sait quel rôle pivot la cryptographie est appelée à jouer dans le rapport de force avec son peuple. En avril 1993, l'administration Clinton dévoila une audacieuse nouvelle initiative dans la politique cryptographique, qui avait été préparée à l'Agence de Sécurité Nationale ("National Security Agency" NSA)

depuis le début de l'administration Bush. La pièce centrale de ce dispositif est le microprocesseur construit par le gouvernement et appelé puce "Clipper", contenant un chiffre de la NSA classé top secret. Le gouvernement est en train d'encourager l'industrie privée à l'insérer dans leurs équipements de communications sécurisées, comme les téléphones sécurisés, les fax sécurisés, etc. AT&T insère dès à présent la "Clipper" dans ses équipements vocaux sécurisés. Ce que cela cache: au moment de la fabrication, chaque puce "Clipper" sera chargée avec sa propre clé, et le gouvernement en gardera une copie, placée entre les mains d'un tiers. Il n'y a pas à s'inquiéter, cependant: le gouvernement a promis qu'il utiliserait ces clés pour lire le trafic des citoyens uniquement dans les cas dûment autorisés par la loi. Bien sûr, pour rendre la "Clipper" complètement efficace, la prochaine étape devrait être de mettre hors-la-loi toute autre forme de cryptographie.

Le gouvernement avait déclaré au début que l'utilisation de Clipper serait volontaire, que personne ne serait forcé de l'utiliser à la place d'autres types de cryptographie. Mais la réaction du public contre le Clipper a été forte, si forte que le gouvernement a anticipé. L'industrie informatique a affirmé de manière unanime son opposition à l'usage de Clipper. Le directeur du FBI, Louis Freeh, répondit à une question lors d'une conférence de presse en 1994 en disant que si Clipper n'arrivait pas à obtenir le soutien du public, et que les interceptions du FBI étaient réduites à néant par une cryptographie non contrôlée par le gouvernement, son Bureau n'aurait pas d'autre choix que de chercher une solution législative. Plus tard, dans les suites de la tragédie d'Oklahoma City, M. Freeh témoignant devant la Commission Judiciaire du Sénat, déclara que la disponibilité publique de cryptographie sûre devait être restreinte par le gouvernement (bien que personne n'eût suggéré que la cryptographie avait été utilisée par les auteurs de l'attentat).

L'Electronic Privacy Information Center (EPIC) a obtenu des documents révélateurs par le biais du "Freedom of Information Act" [loi sur la liberté de l'information]. Dans un document de travail intitulé "Encryption: The Threat, Applications and Potential Solutions" [Chiffrement: la menace, les applications, et les solutions possibles], et envoyé au Conseil national de sécurité en février 1993, le FBI, la NSA, et le Ministère de la Justice (DOJ) concluaient que "Les solutions techniques, telles qu'elles existent, marcheront seulement si elles sont incorporées dans tous les produits de chiffrement. Pour s'assurer qu'il en sera ainsi, une loi obligeant à l'utilisation de produits de chiffrement approuvés par le Gouvernement ou l'adhésion aux critères de chiffrement du Gouvernement est requise."

Le Gouvernement a eu un comportement qui n'inspire pas confiance dans le fait qu'il n'abuseront pas de nos libertés civiles. Le programme COINTELPRO du FBI avait ciblé les groupes qui s'opposaient aux politiques du Gouvernement. Ils ont espionné les mouvements pacifistes et le mouvement des droits civils. Ils ont intercepté le téléphone de Martin Luther King Jr. Nixon avait sa liste d'ennemis. Et ensuite il y a eu la pagaille du Watergate. Le Congrès paraît maintenant prêt à faire passer des lois restreignant nos libertés civiles sur Internet. A aucun moment dans le passé la méfiance envers le Gouvernement n'a été si largement partagée sur tout le spectre politique qu'aujourd'hui.

Si nous voulons résister à cette tendance inquiétante du gouvernement pour rendre illégale la cryptographie, une mesure que nous pouvons adopter est d'utiliser la

cryptographie autant que nous le pouvons actuellement pendant que c'est encore légal. Quand l'utilisation de cryptographie sûre devient populaire, il est plus difficile pour le gouvernement de la criminaliser. Par conséquent, utiliser PGP est bon pour préserver la démocratie.

Si l'intimité est mise hors la loi, seuls les hors-la-loi auront une intimité. Les agences de renseignement ont accès à une bonne technologie cryptographique. De même les trafiquants d'armes et de drogue. Mais les gens ordinaires et les organisations politiques de base n'avaient pour la plupart pas eu accès à une technologie cryptographique de "qualité militaire" abordable. Jusqu'à présent.

PGP donne aux gens le pouvoir de prendre en main leur intimité. Il y a un besoin social croissant pour cela. C'est pourquoi je l'ai créé.

Les chiffres symétriques de PGP

PGP offre une sélection de différents chiffres à clé secrète pour chiffrer un message. Par chiffre à clé secrète, nous entendons un algorithme de chiffrement par blocs conventionnel, ou symétrique, qui utilise la même clé aussi bien pour chiffrer que pour déchiffrer. Les trois chiffres symétriques par blocs offerts par PGP sont CAST, Triple-DES, IDEA. Il ne s'agit pas de chiffres "maison". Ils furent tous développés par des équipes de cryptographes de réputation incontestable.

Pour les curieux de cryptographie, ces trois chiffres opèrent sur des blocs de 64 bits de texte clair et de texte chiffré. CAST et IDEA ont des tailles de clés de 128 bits, alors que Triple-DES utilise une clé de 168 bits. Comme le Data Encryption Standard (DES), ces trois chiffres peuvent être utilisés en mode cipher feedback (CFB) et cipher block chaining (CBC). PGP les utilise en mode CFB 64 bits.

J'ai inclus le chiffre CAST dans PGP parce qu'il s'annonce comme un bon chiffre par blocs avec une taille de clé de 128 bits, il est très rapide, et il est libre. Son nom est tiré des initiales de ses concepteurs Carlisle Adams et Stafford Tavares de Northern Telecom (Nortel). Nortel a déposé un brevet pour CAST, mais ils ont ajouté une disposition pour rendre CAST disponible à tous sans avoir à payer de royalties. CAST apparaît comme étant exceptionnellement bien conçu, par des gens jouissant d'excellentes réputations dans ce domaine. La conception est fondée sur une approche très formelle, avec un nombre d'assertions formellement démontrables qui donnent de bonnes raisons de penser qu'il exige une recherche exhaustive des clés pour casser sa clé de 128 bits. CAST n'a pas de clés faibles ou semi faibles. Il existe de solides arguments permettant de penser que CAST est complètement immunisé aussi bien contre la cryptanalyse linéaire que différentielle, les deux formes de cryptanalyse les plus puissantes dans la recherche publique, toutes deux ayant été utilisées pour craquer DES. CAST est trop récent pour que se soit développée une longue série d'études à son sujet, mais sa conception formelle et la bonne réputation de ses concepteurs attirera sans aucun doute l'attention et les tentatives d'attaques cryptanalytiques d'une partie de la communauté cryptographique universitaire. Je ne suis pas loin d'éprouver la même bonne impression au sujet de CAST qu'il y a quelques années au sujet d'IDEA, le chiffre que j'avais choisi pour l'utiliser dans les versions précédentes de PGP. A cette époque, IDEA était aussi trop récent pour avoir fait l'objet d'une série d'études, mais il a très bien tenu.

Le chiffre par blocs IDEA (International Data Encryption Algorithm) est fondé sur le concept du “mixage d’opérations depuis différents groupes algébriques”. Il a été développé au ETH à Zurich par James L. Massey et Xuejia Lai, et publié en 1990. Les premiers articles publiés sur le chiffre l’appelaient IPES (Improved Proposed Encryption Standard), mais ils ont ensuite changé le nom en IDEA. Depuis, IDEA a beaucoup mieux résisté que d’autres chiffres tels que FEAL, REDOC-II, LOKI, Snefru et Khafre. Et IDEA est plus résistant que DES à la très puissante attaque par crytanalyse différentielle de Biham et Shamir, aussi bien qu’aux attaques par cryptanalyse linéaire. Comme ce chiffre continue à attirer les attaques des plus formidables milieux du monde de la cryptanalyse, la confiance en IDEA grandit avec le temps. Malheureusement, le plus grand obstacle à ce que IDEA devienne un standard a été le fait que Ascom Systec détient un brevet sur sa conception, et à la différence de DES et de CAST, IDEA n’est pas disponible gratuitement.

En plus, PGP inclut le Triple-DES à trois clés parmi ses chiffres disponibles. Le DES a été développé par IBM au milieu des années 70. Alors qu’il est de bonne conception, sa taille de clé de 56 bits est trop petite pour les normes d’aujourd’hui. Triple-DES est très robuste, et a été bien étudié depuis plusieurs années, aussi peut-il être considéré comme un pari plus sûr que les nouveaux chiffres tels que CAST et IDEA. Triple-DES est le DES appliqué trois fois au même bloc de données, en utilisant trois clés différentes, à ceci près que la seconde opération DES est lancée en arrière-plan, en mode déchiffrement. Bien que Triple-DES soit beaucoup plus lent que CAST ou IDEA, la vitesse n’est habituellement pas déterminante pour les logiciels d’e-mail. Bien que Triple-DES utilise une taille de clés de 168 bits, il apparaît avoir une taille effective de clé d’au moins 112 bits contre un attaquant, à supposer qu’il ait la capacité de réunir d’immenses quantités de données à utiliser dans l’attaque. Selon un article présenté par Michael Weiner à Cryto96, toute quantité de données plausible pour l’attaquant permettrait une attaque qui requerrait autant de travail que de casser une clé de 129 bits. Triple-DES n’est pas encombré de brevets.

Les clés publiques PGP qui ont été générées par PGP version 5.0 ou ultérieure intègrent des informations qui indiquent à l’expéditeur quels chiffres sont reconnus par le logiciel du destinataire, de telle sorte que le logiciel de l’expéditeur sait quels chiffres peuvent être utilisés pour chiffrer. Les clés Diffie-Hellman/DSS acceptent CAST, IDEA, ou Triple-DES comme chiffres, avec CAST comme sélection par défaut. A ce jour, pour des raisons de compatibilité, les clés RSA n’offrent pas cette fonctionnalité. Seul le chiffre IDEA est utilisé par PGP pour envoyer des messages avec des clés RSA, parce que les anciennes versions de PGP ne géraient que RSA et IDEA.

A propos des routines de compression de données PGP

Normalement PGP compresse le texte clair avant de le chiffrer, parce qu’il est trop tard pour le compresser après qu’il ait été chiffré; des données chiffrées ne sont pas compressibles. La compression de données économise le temps de transmission par modem et l’espace disque et, plus important, augmente la sécurité cryptographique. De nombreuses techniques cryptanalytiques exploitent les redondances trouvées dans le texte clair pour craquer le chiffre. La compression de données réduit cette redondance dans le texte clair, et par là augmente

considérablement la résistance à la cryptanalyse. La compression du texte clair demande un temps supplémentaire, mais du point de vue de la sécurité cela en vaut la peine.

Les fichiers qui sont trop petits pour être compressés, ou qui ne se compressent pas bien, ne sont pas compressés par PGP. En plus, le programme reconnaît les fichiers produits par les programmes de compression les plus courants, tels que PKZIP, et n'essaye pas de compresser un fichier qui a déjà été compressé.

Pour les amateurs de technique, le programme utilise les routines de compression gratuites ZIP écrites par Jean-Loup Gailly, Marc Adler, et Richard B. Wales. Ce logiciel ZIP utilise des algorithmes de compression qui sont fonctionnellement équivalents à ceux utilisés par PKZIP 2.x de PKWare. Ce logiciel de compression ZIP a été sélectionné pour PGP principalement parce qu'il a un taux de compression vraiment bon et parce qu'il est rapide.

A propos des nombres aléatoires utilisés comme clés de session

PGP utilise un générateur de nombres pseudo aléatoires cryptographiquement robuste pour créer les clés de session temporaires. Si ce fichier de semence n'existe pas, il est automatiquement créé et alimenté avec de véritables nombres aléatoires dérivés par PGP de vos actions aléatoires à partir de l'intervalle entre vos frappes clavier et les mouvements de la souris.

Le générateur réalimente le fichier de semence chaque fois qu'il est utilisé, en y mélangeant un nouveau matériau partiellement issu de l'heure du jour et d'autres sources réellement aléatoires. Il utilise le chiffre conventionnel comme un moteur pour le générateur de nombres aléatoires. Le fichier de semence contient des éléments de semence aléatoires et des éléments de clés aléatoires utilisés pour alimenter le moteur de chiffrement conventionnel pour le générateur aléatoire.

Ce fichier de semence aléatoire devrait être protégé de la divulgation, pour réduire le risque qu'un attaquant puisse en déduire vos prochaines ou précédentes clés de session. L'attaquant aurait les plus grandes difficultés à tirer quoi que se soit d'utilisable en s'emparant de ce fichier de semence aléatoire, parce que le fichier est cryptographiquement blanchi avant et après chaque utilisation. Néanmoins, il semble prudent d'essayer de l'empêcher de tomber en de mauvaises mains. Si possible, faites en sorte que ce fichier ne soit identifiable que par vous. Sinon, ne laissez pas n'importe qui copier des disques depuis votre ordinateur.

A propos des contractions de message

La contraction de message est une "condensation" compacte (160 bits ou 128 bits) de votre message ou de la somme de contrôle de fichier. Vous pouvez aussi la voir comme une "empreinte" du message ou du fichier. La contraction de message "représente" votre message d'une manière telle que si le message était altéré en quelque façon, une contraction de message différente serait calculée à partir de lui. Cela permet de détecter tout changement apporté au message par un contrefacteur. La contraction de message est calculée par l'application d'une fonction de hachage

à sens unique, cryptographiquement robuste, au message. Il sera cryptographiquement impraticable pour un attaquant d'élaborer un message de substitution qui produirait une contraction de message identique. Sous ce rapport, une contraction de message est bien meilleure qu'une somme de contrôle, parce qu'il est facile d'élaborer un message différent qui produirait la même somme de contrôle. Mais de même qu'avec une somme de contrôle, vous ne pouvez pas déduire le message originel de la contraction de ce message.

Le chiffre de contraction de message maintenant utilisé dans PGP (version 5.0 et ultérieure) est appelé SHA, acronyme de Secure Hash Algorithm conçu par la NSA pour le National Institute of Standards and Technology (NIST). SHA est un algorithme de hachage sur 160 bits. Quelques personnes pourraient considérer tout ce qui vient de la NSA avec suspicion, parce que la NSA est en charge d'intercepter les communications et de casser les codes. Mais ne perdez pas de vue que la NSA n'a aucun intérêt à contrefaire des signatures, et que le Gouvernement tirera profit d'une bonne norme de signature numérique infalsifiable, qui empêchera quiconque de répudier sa signature. Il y a aussi des avantages distincts dans le cas de poursuites judiciaires et de la recherche de renseignements. De plus, SHA a été publié dans la littérature publique et a été intensivement examiné par la plupart des meilleurs cryptographes du monde spécialisés dans les fonctions de hachage, et l'opinion unanime est que SHA est extrêmement bien conçu. Il comporte quelques innovations de conception qui pallient aux faiblesses constatées dans les algorithmes de contraction précédemment publiés par les cryptographes universitaires. Toutes les nouvelles versions de PGP utilisent SHA en tant qu'algorithme de contraction de messages pour créer des signatures avec les nouvelles clés DSS qui sont compatibles avec le NIST Digital Signature Standard. Pour des raisons de compatibilité, les nouvelles versions de PGP utilisent toujours MD5 pour les signatures RSA, parce que les anciennes versions de PGP utilisaient MD5 pour les signatures RSA.

Le chiffre de contraction de message utilisé par les anciennes versions de PGP est le MD5 Message Digest Algorithm, placé dans le domaine public par RSA Data Security, Inc. MD5 est un algorithme de hachage sur 128 bits. En 1996, MD5 a été presque cassé par un cryptographe Allemand, Hans Dobbertin. Bien que MD5 n'ait pas été complètement cassé cette fois-là, on lui a découvert de sérieuses faiblesses, telles que personne ne devrait l'utiliser pour créer des signatures. Des travaux ultérieurs dans ce domaine pourraient le casser complètement, permettant de contrefaire des signatures. Si vous ne voulez pas qu'un jour votre signature numérique PGP figure sur de faux aveux, vous feriez bien de migrer vers les nouvelles clés PGP DSS comme méthode préférée pour créer des signatures numériques, parce que DSS utilise SHA comme algorithme de hachage.

Comment protéger les clés publiques de la falsification

Dans un cryptosystème à clé publique, vous n'avez pas à protéger les clés publiques de la divulgation. En fait, mieux vaut qu'elles soient largement diffusées. Mais il est important de protéger les clés de la falsification, pour être sûr que la clé publique appartient réellement à la personne à qui elle semble appartenir. C'est peut-être la plus importante vulnérabilité des cryptosystèmes à clé publique.

Voyons d'abord un désastre potentiel, avant de voir comment l'éviter sûrement avec PGP.

Supposons que vous vouliez envoyer un message privé à Alice. Vous téléchargez la clé publique d'Alice depuis un BBS [quelconque, ou un site Internet inconnu]. Vous chiffrez votre lettre à Alice avec cette clé publique et vous la lui envoyez par e-mail.

Malheureusement, à votre insu ou à l'insu d'Alice, un autre utilisateur appelé Charlie a infiltré le BBS et a lui-même généré une clé publique avec l'ID d'utilisateur "Alice" attaché à cette clé. Il a secrètement substitué cette fausse clé à la véritable clé d'Alice. Vous utilisez sans le savoir cette fausse clé appartenant [en réalité] à Charlie au lieu de la clé publique d'Alice. Tout semble normal parce que cette fausse clé affiche "Alice" comme ID d'utilisateur. Maintenant, Charlie peut déchiffrer le message destiné à Alice parce qu'il a la clé secrète correspondante. Il peut même chiffrer à nouveau le message préalablement déchiffré, avec la vraie clé publique d'Alice et le lui envoyer pour que personne ne se doute de la fraude. Pire encore, il peut même faire des signatures, en apparence authentiques, d'Alice avec sa [fausse] clé secrète parce que tout le monde utilisera la fausse clé publique pour vérifier la signature d'Alice.

La seule façon d'éviter ce désastre est d'empêcher que qui ce soit puisse falsifier les clés publiques. Si vous avez obtenu la clé publique d'Alice directement d'Alice, il n'y a pas de problème. Mais cela peut être difficile si Alice est à des milliers de kilomètres de là, ou si elle est actuellement injoignable.

Peut-être pourriez-vous vous procurer la clé publique d'Alice par l'intermédiaire de David, un ami commun en qui vous avez tous les deux confiance, et qui sait qu'il détient une copie authentique de la clé publique d'Alice. David pourrait signer la clé publique d'Alice, se portant ainsi garant de l'intégrité de la clé publique d'Alice. David créerait cette signature avec sa propre clé secrète.

Cela créerait une signature de la clé publique, et prouverait que la clé d'Alice n'a pas été falsifiée. Cela exige de disposer d'une copie reconnue authentique de la clé publique de David pour vérifier sa signature. Peut-être David pourrait-il aussi fournir à Alice une copie signée de votre clé publique. De cette manière, David sert d'"Aval" entre vous et Alice.

Cette signature de la clé publique d'Alice pourrait être mise en ligne par David ou Alice sur un BBS, et vous pourriez la télécharger ultérieurement. Vous pourriez alors vérifier la signature via la clé publique de David et être ainsi assuré qu'il s'agit réellement de la clé publique d'Alice. Aucun imposteur ne peut vous duper en vous faisant accepter sa propre fausse clé comme étant la clé d'Alice parce que personne ne peut contrefaire la signature créée par David.

Une personne largement reconnue comme digne de confiance pourrait même se spécialiser dans ce service [consistant à] "certifier" les utilisateurs les uns aux autres en signant leurs clés publiques. Cette personne de confiance pourrait être considérée comme une "Autorité Certifiante". On aurait l'assurance que toute clé publique portant la signature de l'Autorité Certifiante appartient réellement à la personne à qui elle semble appartenir. Tout utilisateur intéressé n'aurait dès lors besoin que d'une copie reconnue authentique de la clé publique de l'Autorité Certifiante, de sorte que les signatures de l'Autorité Certifiante puissent être

vérifiées [sur les clés publiques des utilisateurs]. Dans certains cas, l'Autorité Certifiante peut aussi faire office de serveur de clés, permettant aux utilisateurs d'un réseau de consulter des clés publiques en interrogeant le serveur de clés, mais il n'y a pas de raison pour qu'un serveur de clés doive aussi certifier des clés.

Une Autorité Certifiante centralisée fiable est particulièrement adaptée aux grandes institutions contrôlées depuis un centre unique comme les grandes entreprises ou les administrations. Quelques milieux institutionnels recourent au modèle de telles Autorités Certifiantes.

Pour des milieux plus décentralisés, permettre à tous les utilisateurs d'agir comme avals de confiance pour leurs amis se révélera probablement mieux adapté que le recours à une autorité de certification centralisée.

Une des fonctionnalités les plus séduisantes de PGP est qu'il est aussi bien adapté à un milieu centralisé avec une Autorité Certifiante qu'à un milieu plus décentralisé dans lequel des individus échangent leurs clés personnelles.

Toute cette affaire de la protection des clés publiques contre la falsification est le problème le plus délicat à résoudre pour les applications pratiques de la cryptographie à clé publique. C'est le "talon d'Achille" de la cryptographie à clé publique, et une grande partie de la complexité du logiciel est liée à la résolution de ce seul problème.

Vous ne devriez utiliser une clé publique qu'après vous être assuré qu'il s'agit d'une clé publique authentique qui n'a pas été falsifiée, et qui appartient réellement à la personne à qui la clé prétend appartenir. Vous pouvez en être sûr si vous tenez cette clé publique directement de son propriétaire, ou si elle est signée par quelqu'un en qui vous avez confiance, dont vous détenez déjà une clé publique authentique. Aussi, l'ID d'utilisateur devrait être le nom complet du propriétaire de la clé, et non pas seulement son nom de famille.

Peu importe combien vous pouvez être tenté, ne cédez *jamais* à la facilité en faisant confiance à une clé publique que vous avez téléchargée depuis un BBS, à moins qu'elle ne soit signée par quelqu'un en qui vous avez confiance. Cette clé non certifiée pourrait avoir été falsifiée, peut-être même par l'administrateur système du BBS.

Si on vous demande de signer la clé publique d'autrui, assurez-vous qu'elle appartient réellement à la personne nommée dans l'ID d'utilisateur de cette clé publique. Et cela parce que votre signature sur sa clé est votre promesse que cette clé publique lui appartient réellement. D'autres personnes qui vous font confiance accepteront sa clé parce qu'elle porte votre signature. Il peut être malavisé de se fier au ouï-dire – ne signez pas sa clé publique sauf si vous avez une connaissance indépendante et de première main qu'elle lui appartient vraiment. De préférence, vous ne devriez la signer que si vous l'obtenez directement d'elle.

Pour signer une clé publique, vous devez être encore bien plus certain de l'appartenance de cette clé que si vous vouliez simplement utiliser cette clé pour chiffrer un message. Pour être convaincu qu'une clé est d'un aloi suffisant pour être utilisée, les signatures par des avals de confiance devraient suffire. Mais pour signer une clé vous-même, vous devriez recourir à votre connaissance directe, personnelle et indépendante du propriétaire de cette clé. Peut-être pourriez-vous téléphoner au propriétaire de la clé et lui lire l'empreinte de la clé pour qu'il

confirme que la clé que vous détenez est réellement sa clé – et assurez-vous que vous parlez réellement à la bonne personne.

Gardez présent à l'esprit que votre signature sur une clé publique ne garantit pas l'intégrité de cette personne, mais seulement l'intégrité (l'appartenance) de la clé publique de cette personne. Vous ne risquez pas de compromettre votre crédibilité en signant la clé publique d'un débile mental, si vous êtes absolument sûr que la clé lui appartient réellement. D'autres personnes accepteront cette clé parce que vous l'avez signée (en admettant qu'elles vous fassent confiance), mais elles n'auront pas confiance dans le propriétaire de cette clé. Avoir confiance en une clé n'est pas la même chose que d'avoir confiance dans le propriétaire de la clé.

Ce serait une bonne idée de garder sous la main une copie de votre propre clé publique signée par de nombreux "avals", dans l'espoir que beaucoup de gens feront confiance à au moins un des avals qui se sont portés garants de la validité de votre propre clé. Vous pourriez poster votre clé avec sa collection de signatures sur divers BBS. Si vous signez la clé publique d'autres personnes, renvoyez-la leur avec votre signature de telle sorte qu'elles puissent l'ajouter à leur propre collection de garants de leur propre clé publique.

Assurez-vous que personne ne peut falsifier votre propre trousseau de clés. La vérification d'une nouvelle signature certifiant une clé publique doit dépendre en dernier ressort de l'intégrité des clés publiques certifiées qui se trouvent déjà dans votre propre trousseau de clés publiques. Gardez un contrôle physique de votre trousseau de clés publiques, de préférence sur votre propre ordinateur personnel plutôt que sur un système distant et/ou partagé, exactement comme vous le feriez pour votre clé secrète. Ceci pour le protéger de la falsification, non de la divulgation. Gardez une copie de sauvegarde fiable de vos trousseaux de clés publiques et secrètes sur un support protégé en écriture.

Dans la mesure où votre propre clé publique certifiée est utilisée comme référence pour certifier directement ou indirectement toutes les autres clés de votre trousseau, c'est celle qu'il faut protéger avec le plus grand soin de la falsification. Vous devriez en garder une copie de sauvegarde sur un support protégé en écriture.

D'une manière générale, PGP présume que vous conserverez le contrôle physique de votre système et de vos trousseaux de clés, ainsi que de votre copie de PGP elle-même. Si un intrus peut accéder à votre disque, alors en théorie il peut falsifier PGP lui-même, remettant en cause l'efficacité des dispositifs de sécurité dont dispose PGP pour détecter une falsification des clés.

Une méthode plus complexe pour protéger votre propre trousseau de toute falsification est de signer ce trousseau entier avec votre propre clé secrète. Vous pouvez le faire en créant une signature détachée du trousseau de clés publiques.

Comment PGP reconnaît-il les clés valides?

Avant de lire ce chapitre, vous devriez lire le chapitre précédent, "Comment protéger les clés publiques de la falsification".

PGP reconnaît les clés convenablement certifiées de votre trousseau de clés publiques à l'aide des signatures des avals en qui vous avez confiance. Tout ce que

vous avez à faire est de dire à PGP qui sont les gens fiables en tant qu'avals, et de certifier leurs clés avec votre propre clé la plus certifiée. PGP peut utiliser cette information, validant automatiquement toutes les autres clés qui ont été signées par les avals. Et bien sûr, vous pouvez directement signer d'autres clés vous-même.

PGP utilise deux critères bien distincts pour apprécier l'aloi d'une clé publique – ne les confondez pas:

1. La clé appartient-elle réellement à la personne à qui elle semble appartenir? En d'autres termes, a-t-elle été certifiée avec une signature fiable?
2. Appartient-elle à quelqu'un en qui vous pouvez avoir confiance pour certifier d'autres clés?

PGP peut évaluer la réponse à la première question. Pour répondre à la deuxième question, vous devez le dire explicitement à PGP. Quand vous répondez à la question 2, PGP peut ensuite évaluer la réponse à la question 1 pour les autres clés signées par l'aval que vous avez désigné comme fiable.

Les clés qui ont été certifiées par un aval de confiance sont considérées comme valides par PGP. Les clés appartenant aux avals de confiance doivent être certifiées soit par vous soit par un autre aval de confiance.

PGP offre aussi la possibilité d'établir des nuances quant au crédit que méritent les avals. Votre confiance dans les propriétaires de clés pour agir en tant qu'avals ne reflète pas seulement votre estimation de leur intégrité personnelle – cela devrait refléter également la sagacité que vous leur supposez dans la compréhension de la gestion des clés et dans celle de signer les clés à bon escient. Vous pouvez désigner à PGP une personne comme inconnue, non fiable, marginalement fiable, ou complètement fiable pour certifier les autres clés publiques. Cette information sur la fiabilité est conservée avec leurs clés dans votre trousseau, mais quand vous demandez à PGP de copier [extraire] une clé de votre trousseau, PGP ne copie pas l'information sur la fiabilité avec la clé, parce que vos opinions personnelles sur la fiabilité sont considérées comme confidentielles.

Quand PGP évalue la validité d'une clé publique, il examine le niveau de fiabilité de toutes les signatures attachées. Il calcule un résultat pondéré de la validité – deux signatures marginalement fiables sont considérées comme équivalentes à une signature complètement fiable. L'évaluation critique de PGP est modulable – par exemple, vous pouvez régler PGP pour exiger deux signatures complètement fiables ou trois signatures marginalement fiables pour décider qu'une clé est valide.

Votre propre clé est "axiomatiquement" valide pour PGP, n'ayant pas besoin de la signature d'un aval pour prouver sa validité. PGP sait quelles clés publiques sont les vôtres, en regardant la clé secrète correspondante dans le trousseau de clés secrètes. PGP présume également que vous vous considérez vous-même comme complètement fiable pour certifier d'autres clés.

Avec le temps, vous accumulerez des clés d'autres personnes que vous pouvez vouloir désigner comme avals de confiance. Chacun choisira ses propres avals de confiance. Et chacun accumulera progressivement et distribuera avec sa clé une collection de signatures d'autres personnes, dans l'espoir que parmi ceux qui en détiendront une copie, il s'en trouvera pour faire confiance à au moins une ou deux

des signatures. Cela permettra l'émergence d'un réseau de confiance décentralisé, à tolérance d'erreurs, pour toutes les clés publiques.

Cette approche originale par la base tranche nettement avec les schémas de la norme de gestion des clés publiques développés par le gouvernement et d'autres institutions centralisées, tel le "Internet Privacy Enhanced Mail" (PEM), qui sont basés sur un contrôle et une obligation de confiance centralisés. Le modèle normatif repose sur une hiérarchie d'Autorités Certifiantes qui vous dictent à qui vous devez faire confiance. La méthode probabiliste décentralisée de PGP pour déterminer l'aloi des clés publiques est la poutre maîtresse de l'architecture de son modèle de gestion des clés. PGP vous laisse choisir vous-même ceux qui méritent votre confiance, vous plaçant au sommet de votre propre pyramide personnelle de certification. PGP est destiné aux gens qui préfèrent plier eux-mêmes leur propre parachute.

Notez que si PGP tend à privilégier cette approche par la base, décentralisée, cela ne signifie pas qu'il ne soit pas aussi bien adapté à des modèles plus hiérarchisés et centralisés de gestion des clés publiques. Dans les grandes sociétés, par exemple, les utilisateurs voudront probablement avoir affaire à un seul interlocuteur, personne physique ou non, qui signera toutes les clés des employés. PGP gère ce scénario centralisé comme un sous-cas particulier de son modèle général de confiance.

Comment protéger ses clés secrètes de la divulgation

Protégez votre propre clé secrète et votre phrase secrète très soigneusement. Si jamais votre clé secrète est compromise, vous feriez mieux de le faire savoir à toutes les parties concernées avant qu'on l'utilise pour signer en votre nom. Par exemple, on pourrait l'utiliser pour créer de fausses vraies signatures, qui pourraient créer des problèmes à beaucoup de monde, surtout si votre signature est largement considérée comme fiable. Et bien sûr, une compromission de votre propre clé secrète compromettrait tous les messages qui vous sont envoyés.

Pour protéger votre clé secrète, vous pouvez commencer par la maintenir toujours sous votre contrôle physique. Il est bon de la conserver sur votre ordinateur personnel à la maison, ou sur un ordinateur portable que vous pouvez emmener avec vous. Si vous devez utiliser au bureau un ordinateur dont vous n'avez pas en permanence le contrôle physique, alors gardez vos trousseaux de clés publiques et secrètes sur une disquette protégée en écriture, et ne l'oubliez pas en quittant le bureau. Ce ne serait pas une bonne idée de conserver votre clé secrète sur un ordinateur distant et/ou partagé, comme un système de type Unix connecté en permanence. Quelqu'un pourrait intercepter la ligne de votre modem et capturer votre phrase secrète, et ensuite se procurer votre clé secrète depuis le système distant. Vous ne devriez utiliser votre clé secrète que sur une machine placée sous votre contrôle physique.

Ne conservez pas votre phrase secrète sur l'ordinateur sur lequel se trouve votre clé secrète. Conserver ensemble la clé secrète et la phrase secrète sur le même ordinateur est aussi dangereux que de garder votre code secret de carte bancaire dans le même portefeuille que la carte. Vous ne voulez pas que quelqu'un mette la main sur votre disque contenant à la fois la phrase secrète et le fichier de clé

secrète. Il serait plus sûr de simplement mémoriser votre phrase secrète et de ne pas la conserver ailleurs que dans votre cerveau. Si vous sentez que vous devez écrire votre phrase secrète, protégez-la bien, peut-être mieux encore que la clé secrète.

Et conservez des copies de sauvegarde de votre clé secrète – rappelez-vous, vous détenez l'unique exemplaire de votre clé secrète, et la perdre rendra inutilisables toutes les copies de votre clé publique que vous avez diffusées à travers le monde.

L'approche décentralisée non institutionnelle utilisée par PGP pour gérer les clés publiques a ses avantages, mais malheureusement elle signifie aussi qu'on ne peut pas compter sur une liste centralisée unique des clés compromises. Cela rend beaucoup plus difficile de limiter les dégâts causés par une compromission de clé secrète. Vous ne pouvez que le faire savoir et espérer que tout le monde en entendra parler.

Si le pire des cas survient – votre clé secrète et votre phrase secrète sont toutes les deux compromises (espérons que vous vous en apercevrez) – vous devrez émettre un certificat de "révocation de clé". Ce type de certificat est utilisé pour prévenir les gens d'arrêter d'utiliser votre clé publique. Vous pouvez utiliser PGP pour créer un tel certificat en utilisant la commande Revoke du menu PGPkeys ou bien en le faisant faire par votre Designated Revoker. Ensuite, vous devez l'envoyer à un serveur de clés de sorte que d'autres puissent le trouver. Leur propre logiciel PGP installera ce certificat de révocation dans leur trousseau de clés publiques et les empêchera automatiquement d'utiliser votre clé publique à l'avenir. Vous pouvez alors générer une nouvelle paire de clés secrète/publique et publier la nouvelle clé publique. Vous pourriez diffuser un "lot" contenant votre nouvelle clé publique et le certificat de révocation de votre ancienne clé.

Que faire si vous perdez votre clé secrète?

Normalement, si vous voulez révoquer votre propre clé secrète, vous pouvez utiliser la commande Revoke du menu PGPkeys pour émettre un certificat de révocation, signé avec votre propre clé secrète.

Mais que pouvez-vous faire si vous perdez votre clé secrète, ou si votre clé secrète est détruite? Vous ne pouvez pas la révoquer vous-même, parce que vous devez utiliser votre propre clé secrète pour la révoquer, et vous ne l'avez plus. Si vous n'avez pas de révocateur désigné pour votre clé, quelqu'un spécifié dans PGP pour révoquer la clé à votre place, vous devez demander à chaque personne qui a signé votre clé de retirer sa certification. Ainsi, quiconque essayera d'utiliser votre clé sur la foi de l'un de vos avals saura qu'il ne faut plus faire confiance à votre clé publique.

Pour plus d'explications au sujet des révocateurs désignés, voir le *Manuel de l'Utilisateur* de PGP.

Méfiez-vous de la poudre de perlimpinpin

Quand vous examinez un logiciel de cryptographie, la question revient toujours: pourquoi devriez-vous faire confiance à ce produit? Même si vous examinez vous-

même le code source, tout le monde n'a pas l'expérience cryptographique pour en apprécier la sécurité. Même si vous êtes un cryptographe expérimenté, de subtiles faiblesses dans les algorithmes peuvent toujours vous échapper.

Quand j'étais au collège, au début des années 70, j'avais conçu ce que je croyais être un schéma de chiffrement génial. Un simple flux pseudo aléatoire était ajouté au flux de texte clair pour créer un texte chiffré. Cela devait apparemment contrecarrer toute analyse de fréquence sur le texte chiffré, et être incassable même pour les services gouvernementaux de renseignement disposant des plus grandes ressources qui soient. Je me sentais tellement suffisant à propos de mon exploit.

Des années plus tard, je découvris le même schéma dans de nombreux textes d'introduction à la cryptographie et des articles de cours. Comme c'était charmant. Les autres cryptographes avaient pensé au même schéma. Malheureusement, le schéma était présenté comme un simple devoir d'écolier sur la manière d'utiliser des techniques cryptographiques élémentaires pour les craquer simplement. Autant pour mon schéma génial.

De ma modeste expérience, j'ai appris combien il est facile de verser dans une conception erronée de la sécurité quand on conçoit un chiffre. La plupart des gens ne réalisent pas combien il est fichtrement difficile de concevoir un chiffre qui puisse résister à une attaque prolongée et déterminée par un adversaire possédant de grandes ressources. Beaucoup d'ingénieurs informaticiens sur grands systèmes ont développé des schémas de chiffrement aussi naïfs (souvent même exactement le même schéma), et certains d'entre eux ont été incorporés dans des logiciels de chiffrement commerciaux et vendus contre argent sonnant et trébuchant à des milliers d'utilisateurs ne soupçonnant rien.

C'est comme vendre des ceintures de sécurité d'automobile qui ont bonne apparence et semblent efficaces, mais s'ouvrent même au plus petit test d'accident. Compter sur elles peut être pire que de ne pas porter de ceinture du tout. Personne ne suspecte qu'elles sont mauvaises jusqu'à l'accident réel. Compter sur un logiciel de cryptographie faible peut faire mettre inconsciemment en danger des informations sensibles. Vous ne l'auriez pas fait si vous n'aviez pas eu du tout de logiciel de cryptographie. Peut-être ne découvrirez-vous jamais que vos données ont été compromises.

Parfois, les logiciels commerciaux utilisent le standard fédéral américain Data Encryption Standard (DES), un assez honnête chiffre conventionnel recommandé par le Gouvernement américain pour l'utilisation commerciale (mais pas pour l'information classée secret défense, curieusement – Hmmm). Il y a plusieurs "modes d'opération" que le DES peut utiliser, certains d'entre eux sont meilleurs que d'autres. Le Gouvernement recommande expressément de ne pas utiliser le mode le plus simple et le plus faible pour les messages, le mode Electronic Codebook (ECB). En revanche, on recommande les modes plus résistants et plus complexes Cipher Feedback (CFB) ou Cipher Block Chaining (CBC).

Malheureusement, la plupart des logiciels commerciaux de cryptographie que j'ai examinés utilisent le mode ECB. Quand j'en ai parlé aux auteurs de plusieurs de ces réalisations, ils ont dit qu'ils n'avaient jamais entendu parler des modes CBC ou CFB, et qu'ils ne savaient rien au sujet de la faiblesse du mode ECB. Le fait même qu'ils n'aient jamais étudié assez de cryptographie pour connaître ces

concepts élémentaires n'est pas rassurant. Et ils gèrent parfois leurs clés DES d'une manière inadéquate ou non sûre. De même, ces logiciels incluent souvent un second chiffre plus rapide qui peut être utilisé à la place du DES plus lent. L'auteur du logiciel pense souvent que son chiffre propriétaire plus rapide est aussi sûr que le DES, mais après l'avoir questionné je découvre habituellement que c'est juste une variation de mon génial schéma de l'époque du collège. Ou peut-être ne révélera-t-il jamais comment son schéma de chiffrement propriétaire fonctionne, mais il m'assure que c'est un schéma génial et que je devrais lui faire confiance. Je suis sûr qu'il croit que son chiffre est génial, mais comment puis-je le savoir sans le voir?

En toute justice, je dois signaler que dans la plupart des cas ces produits lamentables ne proviennent pas de sociétés qui se spécialisent dans la technologie cryptographique.

Même les très bons logiciels, qui utilisent le DES dans le mode d'opération correct présentent encore des problèmes. Le standard DES utilise une clé de 56 bits, ce qui est trop petit pour les normes actuelles, et peut maintenant être aisément cassée par des recherches exhaustives de la clé sur des machines ultra rapides spéciales. Le DES a atteint la fin de sa vie utile, et voilà pourtant encore des logiciels qui y font appel.

Il y a une société appelée AccessData (http://www.accessdata.com/) qui vend très bon marché un ensemble qui craque le schéma de chiffrement intégré utilisé par WordPerfect, Lotus 1-2-3, MS Excel, Symphony, Quattro Pro, Paradox, MS Word et PKZIP. Il ne recherche pas simplement les mots de passe – il fait vraiment de la cryptanalyse. Des gens l'achètent quand ils ont oublié leur mot de passe pour leurs propres fichiers. Les services de police judiciaire l'achètent aussi, ainsi peuvent-ils lire les fichiers qu'ils saisissent. J'ai parlé à Eric Thompson, l'auteur, et il a dit que son programme prend seulement une demi seconde pour les craquer, mais qu'il a intégré une boucle retardatrice pour le ralentir de sorte que cela ne semble pas trop facile au client.

Dans le domaine du téléphone sécurisé, vos choix sont plutôt limités. Le ténor est le STU-III (Secure Telephone Unit), fabriqué par Motorola et AT&T pour un prix de 2 à 3.000 $, utilisé par le Gouvernement pour des applications classées secret défense. Il dispose d'une cryptographie forte, mais l'achat de cette version forte est soumise à une autorisation spéciale du Gouvernement. Une version commerciale du STU-III est disponible, mais édulcorée pour le confort de la NSA, ainsi qu'une version pour l'exportation, encore plus sévèrement affaiblie. On trouve ensuite le AT&T Surity 3600, qui utilise la fameuse puce gouvernementale Clipper pour le chiffrement, avec séquestre des clés à l'usage du Gouvernement et pour le confort des intercepteurs. Ensuite, bien sûr, on trouve les brouilleurs vocaux analogiques (non numériques) qui vous pouvez acheter sur les catalogues du parfait espion, qui sont des joujoux insignifiants sur le plan cryptographique, mais qui sont vendus comme étant des équipements de communication "sécurisés" à des clients qui de toute façon ne connaissent rien de mieux.

D'un certain point de vue, la cryptographie est comme la pharmacie. Sa qualité peut être absolument cruciale. La mauvaise pénicilline a la même apparence que la bonne. Vous pouvez juger que votre tableur est mauvais, mais comment juger que votre logiciel de cryptographie est faible? Le texte chiffré produit par un chiffre

faible paraît aussi bon que le texte chiffré produit par un chiffre résistant. Il y a beaucoup de poudre de perlimpinpin là-dedans. Beaucoup de remèdes de charlatan. Contrairement aux colporteurs d'élixirs de charlatans, ces programmeurs de logiciels ne savent habituellement même pas que leur truc est de la poudre de perlimpinpin. Ils sont peut-être de bons ingénieurs informaticiens, mais ils n'ont habituellement même pas lu d'ouvrages universitaires de cryptographie. Mais ils croient quand même qu'ils peuvent écrire de bons logiciels de cryptographie. Et pourquoi pas? Après tout, cela semble intuitivement facile à faire. Et leurs logiciels semblent bien marcher.

Quiconque croit avoir inventé un schéma de chiffrement incassable est, soit un véritable génie, soit un naïf inexpérimenté. Malheureusement, j'ai quelquefois affaire à ces prétendus cryptographes qui veulent apporter des "améliorations" à PGP en lui ajoutant des chiffres de leur cru.

Je me souviens d'une conversation avec Brian Snow, un cryptographe de haut rang de la NSA. Il me dit qu'il ne ferait jamais confiance à un chiffre conçu par quelqu'un qui ne s'était pas "fait les os" en passant d'abord beaucoup de temps à casser des codes. Cela tombait sous le sens. J'observai que pratiquement personne dans le monde de la cryptographie commerciale n'était qualifié selon ce critère. "Oui", répondit-il avec un sourire entendu, "Et cela rend notre travail à la NSA tellement plus facile." Une réflexion à vous glacer le sang. Je n'en aurais pas jugé autrement.

Le Gouvernement américain a également colporté la poudre de perlimpinpin. Après la Seconde Guerre mondiale, les USA vendirent les machines à chiffrer allemandes Enigma aux gouvernements du Tiers monde. Mais ils ne leur dirent pas que les Alliés avait cassé le code Enigma pendant la guerre, un fait qui resta classé secret défense pendant de nombreuses années. Aujourd'hui encore, de nombreux systèmes Unix dans le monde entier utilisent le chiffre d'Enigma pour le chiffrement de fichiers, en partie parce que le Gouvernement a dressé des obstacles légaux contre l'utilisation de meilleurs chiffres. Ils essayèrent même d'empêcher la publication initiale de l'algorithme RSA en 1977. Et ils ont étouffé dans l'œuf toutes les tentatives [de l'industrie] pour développer des téléphones réellement sécurisés pour le grand public.

La principale activité de la NSA (National Security Agency) du Gouvernement américain consiste à recueillir des renseignements, principalement en enregistrant secrètement les communications privées des gens (voir le livre de James Bamford, *The Puzzle Palace*). La NSA a accumulé des compétences et des ressources considérables pour casser des codes. Quand les gens ne peuvent pas disposer de bonne cryptographie pour se protéger, cela rend le travail de la NSA plus facile. La NSA a également pour mission d'approuver et de recommander des chiffres. Des critiques soutiennent que c'est une source de conflits d'intérêts, comme mettre le renard à garder le poulailler. La NSA a poussé en avant un chiffre conventionnel qu'elle avait conçu (le COMSEC Endorsement Program), et elle ne dira à personne comment il fonctionne parce que c'est classé secret défense. Elle veut qu'on lui fasse confiance et qu'on l'utilise. Mais n'importe quel cryptographe vous dira qu'un chiffre bien conçu n'a pas à être classé secret défense pour rester sûr. Seules les clés auraient besoin de protection. Comment fait-on pour savoir vraiment si le chiffre classé secret défense de la NSA est sûr? Il n'est pas difficile pour la NSA de

concevoir un chiffre qu'elle seule peut craquer, si personne ne peut examiner le chiffre.

Il y a trois facteurs principaux qui ont miné la qualité des logiciels commerciaux de cryptographie aux Etats-Unis.

- Le premier est le manque virtuellement universel de compétence des programmeurs de logiciels commerciaux de cryptographie (quoique cela commence à changer depuis la sortie de PGP). Chaque ingénieur informaticien se prend pour un cryptographe, ce qui a conduit à la prolifération de logiciels de crypto vraiment mauvais.
- Le second est que la NSA a délibérément et systématiquement éliminé toutes les bonnes technologies commerciales de chiffrement, par l'intimidation légale et la pression économique. Une partie de cette pression a été portée à son maximum par les rigoureux contrôles à l'exportation sur les logiciels de cryptographie ce qui, vu l'aspect financier du marketing logiciel, a eu pour résultat d'éliminer les logiciels de chiffrement domestiques.
- La troisième méthode d'élimination consiste à concéder tous les brevets portant sur tous les algorithmes de chiffrement à clé publique à une seule société, constituant un goulot d'étranglement pour empêcher l'extension de cette technologie (cependant le monopole de cette concession est tombé fin 1995).

Le résultat tangible de tout cela est qu'avant la sortie de PGP, il n'y avait presque pas de logiciels de chiffrement de haute sécurité disponibles aux USA. Je ne suis pas aussi certain de la sécurité de PGP que je l'étais autrefois de celle de mon génial logiciel de chiffrement du collège. Si je l'étais, ce serait mauvais signe. Mais je suis à peu près sûr que PGP ne contient pas de faiblesses manifestes (bien qu'il puisse contenir des bogues). J'ai choisi les meilleurs chiffres publiés dans les milieux de la cryptographie universitaire civile. Pour la plupart, ces chiffres ont fait individuellement l'objet d'un examen approfondi étendu. Je connais beaucoup des cryptographes d'autorité mondiale, et j'ai beaucoup discuté avec certains d'entre eux des chiffres et des protocoles utilisés par PGP. Il est bien étudié, et cela a pris des années pour le réaliser. Et je ne travaille pas pour la NSA. Mais vous n'avez pas à me croire sur parole au sujet de l'intégrité cryptographique de PGP, parce que le code source est disponible pour faciliter son examen approfondi.

Encore une chose au sujet de mon engagement en faveur de la qualité cryptographique de PGP. Depuis qu'à l'origine j'ai développé et réalisé gratuitement PGP en 1991, j'ai fait l'objet pendant trois ans, d'une enquête judiciaire diligentée à la requête des Douanes américaines sous la prévention d'avoir diffusé PGP à l'étranger, avec le risque de poursuites pénales et d'années d'emprisonnement. Par comparaison, vous n'avez pas vu le Gouvernement s'émouvoir à propos d'autres logiciels cryptographiques – c'est PGP qui les a rendus furieux. N'est-ce pas là un aveu quant à la puissance de PGP? J'ai bâti ma réputation sur l'intégrité cryptographique de mes produits. Je ne trahirai pas mon engagement en faveur de notre droit au respect de l'intimité, pour lequel j'ai risqué ma liberté. Je ne suis pas près de permettre à un produit portant mon nom d'être muni d'une quelconque porte cachée.

Vulnérabilités

“Si tous les ordinateurs personnels du monde – 260 millions – étaient mis à travailler sur un seul message chiffré avec PGP, cela prendrait encore un temps estimé à 12 millions de fois l’âge de l’univers, en moyenne, pour casser un simple message.”

– William Crowell, Directeur délégué, National Security Agency, 20 Mars 1997.

Aucun système de sécurité n’est impénétrable. PGP peut être circonvenu par une variété de biais. Dans tout système de sécurité de données, vous devez vous interroger pour savoir si l’information que vous cherchez à protéger a plus de valeur pour l’attaquant que le coût de l’attaque. Cela devrait vous amener à vous protéger des attaques les moins coûteuses, tout en ne vous préoccupant pas des attaques plus onéreuses.

Des passages de la discussion qui suit peuvent paraître excessivement paranoïaques, mais une telle attitude est appropriée pour une discussion raisonnable des problèmes de vulnérabilité.

Phrase secrète et clé privée compromises

L’attaque probablement la plus simple intervient si vous laissez la phrase secrète de votre clé privée écrite quelque part. Si quelqu’un l’obtient et obtient aussi votre clé privée, il peut lire vos messages et faire des signatures en votre nom.

Voici quelques recommandations pour protéger votre phrase secrète:

1. N’utilisez pas de phrases secrètes évidentes qui peuvent être aisément devinées, comme les noms de vos enfants ou conjoint.
2. Utilisez des espaces et une combinaison de nombres et de lettres dans votre phrase secrète. Si vous ne mettez qu’un seul mot dans votre phrase secrète, elle peut être aisément devinée à l’aide d’un ordinateur qui essaie tous les mots d’un dictionnaire jusqu’à ce qu’il trouve votre mot de passe. C’est pourquoi une phrase secrète est bien meilleure qu’un mot de passe. Un attaquant plus sophistiqué peut fouiller avec son ordinateur un livre de citations connues pour trouver votre phrase secrète.
3. Soyez créatif. Utilisez une phrase secrète facile à mémoriser mais difficile à deviner; vous pouvez facilement en construire un en utilisant un dicton insensé ou une obscure citation littéraire.

La falsification de clé publique

Une vulnérabilité majeure existe si les clés publiques ont été falsifiées. Cela peut être une vulnérabilité d’une importance cruciale pour un cryptosystème à clé publique, en partie parce que la plupart des novices ne la reconnaissent pas immédiatement.

Pour résumer: quand vous utilisez une clé publique, assurez-vous qu’elle n’a pas été falsifiée. Une nouvelle clé publique ne devrait être digne de confiance que si vous l’obtenez directement de son propriétaire, ou si elle a été signée par

quelqu'un en qui vous avez confiance. Assurez-vous que personne n'a pu falsifier votre propre clé publique. Maintenez un contrôle physique à la fois sur votre trousseau de clés publiques et votre clé privée, de préférence sur votre propre ordinateur personnel plutôt que sur un système distant et/ou partagé. Conservez une copie de sauvegarde de vos deux trousseaux de clés.

Fichiers pas tout à fait effacés

Un autre problème potentiel de sécurité vient de la façon dont la plupart des systèmes d'exploitation effacent les fichiers. Quand vous chiffrez un fichier puis effacez le texte clair originel, le système d'exploitation ne détruit pas réellement les données. Il se contente de marquer ces secteurs du disque comme effacés, permettant à l'espace d'être réutilisé plus tard. C'est un peu comme de mettre des papiers sensibles dans la corbeille à papier plutôt que dans le broyeur. Les secteurs du disque contiennent encore les données sensibles originelles que vous vouliez détruire, et qui seront probablement effacées par de nouvelles données dans le futur. Si un attaquant lit ces blocs de fichier effacés peu de temps après qu'ils aient été retirés de l'espace alloué, il pourrait retrouver votre texte clair.

En fait, cela pourrait même arriver accidentellement, si quelque chose a mal fonctionné sur le disque et que des fichiers ont été accidentellement effacés ou corrompus. Un programme de récupération de disque peut être lancé pour récupérer les fichiers endommagés, mais cela signifie souvent que des fichiers précédemment effacés sont ressuscités en même temps que tout le reste. Vos fichiers confidentiels que vous pensiez partis à jamais peuvent ensuite réapparaître et être inspectés par quiconque tente de récupérer votre disque endommagé. Même pendant que vous créez le message originel avec un traitement de texte ou un éditeur de texte, l'éditeur peut créer de multiples copies temporaires de votre texte sur le disque, uniquement pour son fonctionnement interne. Ces copies temporaires de votre texte sont effacées par le traitement de texte une fois le travail effectué, mais ces fragments sensibles sont encore quelque part sur votre disque.

La seule façon d'empêcher le texte clair de réapparaître est de provoquer d'une façon ou d'une autre l'écrasement par écriture des textes clairs effacés. A moins que vous teniez pour sûr le fait que tous les secteurs de disque effacés seront bientôt réutilisés, vous devez prendre des dispositions positives pour écrire par-dessus le texte clair, et aussi tout fragment du texte clair laissé sur le disque par votre traitement de texte. Vous pouvez vous occuper de tout fragment du texte clair laissé sur le disque en utilisant les fonctions de nettoyage sécurisé et de nettoyage de l'espace libre de PGP.

Virus et chevaux de Troie

Une autre attaque pourrait impliquer un virus informatique spécialement ajusté ou un "ver" qui pourrait infecter PGP ou votre système d'exploitation. Cet hypothétique virus pourrait être conçu pour capturer votre phrase secrète ou votre clé privée ou vos messages déchiffrés, et pour écrire à la dérobée dans un fichier l'information capturée ou l'envoyer à travers un réseau au propriétaire du virus. Ou il pourrait altérer le comportement de PGP de telle sorte que les signatures ne

soient pas convenablement vérifiées. Cette attaque est moins coûteuse qu'une attaque cryptanalytique.

Se défendre contre ce type d'attaque tombe dans la catégorie de la défense contre les infections virales en général. Il y a des produits commerciaux relativement capables qui sont disponibles, et il y a des procédures prophylactiques à suivre qui peuvent réduire grandement les risques d'une infection virale. Un traitement complet de contre-mesures antivirales et anti vers sort du cadre de ce document. PGP n'a pas de défenses contre les virus, et présume que votre propre ordinateur personnel est un environnement d'exécution digne de confiance. Si un tel virus ou un ver apparaissait réellement, avec un peu de chance le monde serait aussitôt au courant.

Une attaque similaire implique quelqu'un créant une habile imitation de PGP qui se comporte comme PGP à bien des égards, mais qui ne marche pas de la façon dont il est supposé le faire. Par exemple, il pourrait être délibérément mutilé pour ne pas vérifier les signatures correctement, permettant à de fausses clés d'être acceptées. Cette version *cheval de Troie* de PGP n'est pas difficile à créer pour un attaquant, parce que le code source de PGP est largement disponible, aussi n'importe qui pourrait modifier le code source et produire un zombie lobotomisé imité de PGP qui ait l'air conforme mais qui répond aux ordres de ses maîtres diaboliques. Cette version cheval de Troie de PGP pourrait ensuite être largement distribuée, se déclarant provenir d'une source légitime. Comme c'est insidieux.

Vous devriez faire un effort pour obtenir votre copie de PGP directement de Network Associates, Inc.

Il y a d'autres façons de vérifier si PGP a été falsifié, en utilisant des signatures numériques. Vous pourriez utiliser une autre version digne de confiance de PGP pour vérifier la signature sur une version suspecte de PGP. Mais cela n'aidera pas du tout si votre système d'exploitation est infecté, ni ne le détectera si votre copie originale de pgp.exe a été malicieusement altérée d'une façon ou d'une autre pour altérer sa propre capacité à vérifier les signatures. Ce test présume aussi que vous avez une bonne copie fiable de la clé publique que vous utilisez pour vérifier la signature de l'exécutable de PGP.

Fichiers d'échange et/ou mémoire virtuelle

PGP a été développé à l'origine pour MS-DOS, un système d'exploitation primitif par rapport aux normes actuelles. Mais alors qu'il était porté vers d'autres systèmes d'exploitation plus complexes, comme Microsoft Windows et Macintosh OS, une nouvelle vulnérabilité a émergé. Cette vulnérabilité découle du fait que ces systèmes d'exploitation de connaisseurs utilisent une technique appelée *mémoire virtuelle*.

La mémoire virtuelle vous permet de lancer d'énormes programmes sur votre ordinateur qui sont plus gros que l'espace disponible dans le microprocesseur de la mémoire vive de votre ordinateur. Cela est pratique parce que les logiciels sont devenus de plus en plus hypertrophiés depuis que les interfaces graphiques adaptées à l'utilisateur sont devenues la norme et que les utilisateurs ont commencé à lancer de nombreuses grosses applications en même temps. Le système d'exploitation utilise le disque dur pour stocker des portions du logiciel

qui ne sont pas utilisées à ce moment. Cela signifie que le système d'exploitation pourrait, sans que vous le sachiez, recopier sur le disque des choses dont vous pensiez qu'elle resteraient seulement en mémoire – des choses comme des clés, des phrases secrètes, et du texte déchiffré. PGP ne garde pas cette sorte de données sensibles exposées en mémoire plus longtemps que nécessaire, mais il y a de toute façon un risque que le système d'exploitation les copie sur le disque.

Les données sont recopiées dans l'espace mémoire du disque, appelé *fichier d'échange* (ou de swap). Les données sont relues depuis ce fichier d'échange dès que c'est nécessaire, de telle sorte que seule une partie de votre programme ou de vos données est physiquement en mémoire à un moment précis. Toute cette activité est invisible pour l'utilisateur, qui voit juste le disque en train de mouliner. Microsoft Windows échange des segments de mémoire, appelés pages, en utilisant un algorithme de remplacement de page appelé "Least Recently Used" (LRU). Cela signifie que les pages qui n'ont pas été consultées depuis le plus longtemps sont les premières à être échangées vers le disque. Cette approche suggère que dans la plupart des cas le risque sera relativement faible que des données sensibles soient échangées vers le disque, parce que PGP ne les laisse pas en mémoire très longtemps. Mais nous ne garantissons rien.

Le fichier d'échange peut être consulté par quiconque peut obtenir un accès physique à votre ordinateur. Si vous êtes concernés par ce problème, vous pouvez le résoudre en récupérant un logiciel spécial qui écrit par-dessus votre fichier d'échange. Un autre remède possible est de désactiver la fonction mémoire virtuelle de votre système d'exploitation. Microsoft Windows le permet, ainsi que Mac OS. Désactiver la mémoire virtuelle peut vouloir dire que vous aurez besoin de plus de mémoire physique sous forme de barrettes de RAM installée, afin de tout gérer dans la RAM.

Brèche dans la sécurité physique

Une brèche dans la sécurité physique peut permettre à quelqu'un de recueillir physiquement vos textes clairs ou vos messages imprimés. Un adversaire déterminé pourrait accomplir cela par le cambriolage, le tri des poubelles, les fouilles et saisies illégales, ou la corruption, l'ouverture du courrier, ou l'infiltration de votre équipe. Certaines de ces attaques peuvent être spécialement réalisables contre les organisations politiques de base qui dépendent dans une large mesure de volontaires.

Ne vous endormez pas dans une fausse sécurité uniquement parce que vous avez un outil cryptographique. Les techniques cryptographiques ne protègent les données que lorsqu'elles sont chiffrées – une violation directe de la sécurité physique peut encore compromettre les données en clair ou les informations écrites ou orales.

Ce type d'attaque est moins coûteux qu'une attaque cryptanalytique sur PGP.

Les attaques Tempest

Une autre sorte d'attaque qui a été utilisée par des adversaire bien équipés implique la détection à distance des signaux électromagnétiques [émis par] votre ordinateur. Cette coûteuse et parfois laborieuse attaque est probablement toujours moins coûteuse que l'attaque cryptanalytique directe. Une camionnette équipée des instruments appropriés se gare près de votre bureau et capture à distance toutes les frappes du clavier et les messages affichés sur l'écran vidéo de votre ordinateur. Cela compromettrait tous vos mots de passes, messages, etc. Cette attaque peut être contrecarrée en protégeant correctement votre équipement informatique et câblage de réseau de telle sorte qu'ils n'émettent pas ces signaux. Cette technologie de protection, appelée "Tempest," est utilisée par certaines agences gouvernementales et entreprises travaillant avec la Défense Nationale. Il y a des vendeurs d'équipements qui proposent ces boucliers Tempest.

Quelques version récentes de PGP (postérieures à la version 6.0) peuvent afficher le texte clair déchiffré en utilisant une police spécialement conçue qui peut réduire le niveau d'émission radio de l'écran de votre moniteur. Cela peut rendre plus difficile la capture des signaux à distance. Cette police spéciale est disponible dans quelques versions de PGP qui gèrent le dispositif "Secure Viewer".

Se protéger contre les fausses empreintes de date

Une vulnérabilité quelque peu obscure de PGP implique que des utilisateurs malhonnêtes créent de fausses empreintes de date sur leurs propres copies de clés publiques et signatures. Vous pouvez sauter cette partie si vous êtes un utilisateur occasionnel et n'êtes pas versé dans les obscurs protocoles de clés publiques.

Il n'y a rien à faire pour empêcher un utilisateur malhonnête de modifier les réglages de date et d'heure de l'horloge système [de son ordinateur], et de générer ses propres clés publiques et signature qui paraissent avoir été créées à une époque différente. Il peut feindre d'avoir signé quelque chose plus tôt ou plus tard qu'il ne le prétend, ou bien que sa paire de clés publique/privée a été créée plus tôt ou plus tard. Il peut y avoir des avantages juridiques ou financiers pour lui, par exemple en créant un ensemble d'échappatoires qui pourrait lui permettre de répudier sa signature.

Je pense que ce problème de la falsification de l'empreinte de date dans les signatures numériques n'est pas pire qu'il n'est déjà dans les signatures manuscrites. N'importe qui peut écrire n'importe quelle date à côté de sa signature manuscrite sur un contrat, mais personne ne semble s'alarmer de cet état de choses. Dans certains cas, une date "incorrecte" sur une signature manuscrite pourrait ne pas être associée avec la fraude en question. L'empreinte de date pourrait être celle du moment où le signataire déclare qu'il a signé le document, ou peut-être celle à laquelle il veut que la signature prenne effet.

Dans les situations où il est d'importance critique qu'une signature soit authentifiée pour une date véritable, les gens peuvent simplement utiliser des notaires pour attester et dater une signature manuscrite. L'équivalent dans les signatures numériques est d'avoir un tiers vraiment digne de confiance pour signer un certificat de signature, appliquant une empreinte de date fiable. Des protocoles

exotiques ou trop formels ne sont pas nécessaires pour cela. Des signatures témoins ont été reconnues depuis longtemps comme un moyen légitime de déterminer l'époque à laquelle un document a été signé.

Une Autorité Certifiante inspirant une large confiance ou un notaire pourraient créer des signatures notariales avec une empreinte de date fiable. Cela ne requerrait pas nécessairement une autorité centralisée. Peut-être que des avals de confiance ou des tiers désintéressés pourraient assurer cette fonction, comme le font les vrais notaires. Quand un notaire signe les signatures d'autres personnes, il créé un certificat de signature d'un certificat de signature. Cela servirait de certification de la signature de la même façon que les notaires réels certifient aujourd'hui des signatures manuscrites. Le notaire pourrait déposer le certificat de signature détaché (sans la totalité du document qui a été signé) dans un registre spécial contrôlé par le notaire. N'importe qui pourrait lire ce registre. La signature du notaire aurait une empreinte de date digne de confiance, ce qui aurait une plus grande crédibilité ou de signification légale que l'empreinte de date dans la signature originale.

Il y a une bonne analyse de ces questions dans l'article de Denning de 1983 dans IEEE Computer. Les futures améliorations de PGP pourraient inclure des fonctionnalités pour gérer facilement les signatures notariées de signatures, avec des empreintes de date fiables.

Divulgation sur des systèmes multi utilisateurs

PGP a été conçu à l'origine pour un système mono utilisateur sous votre contrôle physique direct. Si vous lancez PGP à la maison sur votre propre PC, vos fichiers chiffrés sont généralement sûrs, à moins que quelqu'un pénètre par effraction chez vous, vole votre PC et vous persuade de lui donner votre phrase secrète (ou que votre phrase secrète soit assez facile à deviner).

PGP n'est pas conçu pour protéger vos données alors qu'elles sont sous une forme lisible sur un système compromis. Il ne peut pas non plus empêcher un intrus d'utiliser des moyens sophistiqués pour lire votre clé privée pendant qu'elle est utilisée. Vous devrez reconnaître ces risques sur les systèmes multi utilisateurs, et adapter vos habitudes en conséquence. Peut-être que votre situation est telle que vous devriez envisager de ne lancer PGP que sur un système isolé et mono utilisateur sous votre contrôle physique direct.

Analyse de trafic

Même si l'attaquant ne peut pas lire le contenu de vos messages chiffrés, il peut en déduire au moins des informations utiles en observant d'où viennent les messages et où ils vont, la taille des messages, et le moment de la journée où les messages sont envoyés. Pour l'attaquant, c'est comme examiner votre facture de téléphone pour voir qui vous appelez, quand et pour combien de temps, quand bien même le contenu actuel de vos appels lui demeure inconnu. Cela s'appelle l'analyse de trafic. PGP seul ne protège pas contre l'analyse de trafic. Résoudre ce problème requerrait des protocoles de communication spécialement conçus pour réduire

l'exposition à l'analyse de trafic dans votre environnement de communication, éventuellement avec une assistance cryptographique.

Cryptanalyse

Une coûteuse et formidable attaque cryptanalytique pourrait éventuellement être montée par quelqu'un avec les ressources d'énormes super calculateurs, comme les agences de renseignement gouvernementales. Ils pourraient craquer votre clé publique en utilisant une quelconque nouvelle percée mathématique. Mais la [recherche] universitaire civile a intensément attaqué la cryptographie à clé publique sans succès depuis 1978.

Peut-être que le gouvernement possède des méthodes classées top secret de craquage des chiffres conventionnels utilisés dans PGP. C'est le pire cauchemar de tout cryptographe. Il ne peut pas y avoir de garanties absolues de sécurité dans les réalisations cryptographiques pratiques.

Tout de même, l'optimisme semble justifié. Les algorithmes de clé publique, les algorithmes de contraction de message, et les chiffres par blocs utilisés dans PGP ont été conçus par les meilleurs cryptographes du monde. Les chiffres de PGP ont subi des analyses de sécurité approfondies et des examens méticuleux de la part des meilleurs cryptographes dans le monde non classé top secret.

En outre, même si les chiffres par blocs utilisés dans PGP ont certaines faiblesses subtiles inconnues, PGP compresse le texte clair avant le chiffrement, ce qui devrait réduire considérablement ces faiblesses. Le temps de calcul pour le craquer revient largement plus cher que la valeur du message.

Si votre situation justifie de s'inquiéter d'attaques vraiment formidables de ce calibre, alors peut-être devriez-vous contacter un consultant en sécurité des données pour des approches sur mesure de la sécurité des données qui soit adaptée à vos besoins particuliers.

Pour résumer, sans une bonne protection cryptographique de vos communications de données, il peut être facile en pratique et peut-être même banal pour un adversaire d'intercepter vos messages, particulièrement ceux envoyés par un modem ou un système e-mail. Si vous utilisez PGP et prenez des précautions raisonnables, l'attaquant aura à dépenser nettement plus d'efforts et d'argent pour violer votre vie privée.

Si vous vous protégez par vous-même des attaques les plus simples, et que vous estimez que votre intimité ne va pas être violée par un attaquant déterminé et doté de grandes ressources, alors vous serez probablement en sécurité en utilisant PGP. PGP vous donne une Assez Bonne Confidentialité [en anglais: *Pretty Good Privacy*].

Glossaire

A5	Algorithme cryptographique secret utilisé dans les téléphones cellulaires européens.
Access control [Contrôle d'accès]	Méthode pour restreindre l'accès à des ressources. On n'autorise que certaines entités privilégiées.
Additional recipient request key [Clé à requête de destinataire supplémentaire imposé]	La présence de cette clé spéciale indique que tous les messages chiffrés avec sa clé de base associée doivent aussi être chiffrés avec elle. On l'appelle quelquefois par son nom commercial "clé de déchiffrement supplémentaire [imposé]".
AES (Advanced Encryption Standard) [Standard de chiffrement avancé]	Standard approuvé par le NIST, en général valable pour les 20 ou 30 prochaines années.
AKEP (Authentication Key Exchange Protocol) [Protocole d'échange de clé d'authentification]	Transport de clé basé sur du chiffrement symétrique permettant à deux parties d'échanger une clé privée partagée, sûr contre des adversaires passifs.
Algorithm (encryption) [Algorithme (de chiffrement]	Ensemble de règles mathématiques (logiques) utilisées pour le chiffrement et le déchiffrement.
Algorithm (hash) [Algorithme (de hachage)]	Ensemble de règles mathématiques (logiques) utilisées dans le processus de création d'empreinte de message et la génération de clés / de signatures.
Anonymity [Anonyme, anonymat]	D'une origine ou d'un auteur non déclaré; fait de cacher son identité.
ANSI (American National Standards Institute) [Institut National de Standards Américain]	Développe des standards via divers "comités de standardisation accrédités" [Accredited Standards Institute] (ASC). Le comité X9 s'intéresse particulièrement aux standards de sécurité pour l'industrie des services financiers.

API (Application Programming) [Interface de programmation applicative]	Fournit le moyen de tirer parti des fonctions du logiciel, en permettant à des produits logiciels différents d'interagir.
ASN.1 (Abstract Syntax Notation One) [Notation de syntaxe abstraite n° 1]	Standard ISO/IEC pour les règles de codage des certificats X.509. Deux types existent – DER (Règles de codage élaborées) [Distinguished Encoding Rules] et BER (règles de codage de base) [Basic Encoding Rules].
Asymetric keys [Clés asymétriques]	Paire de clés séparées mais unifiées, composée d'une clé publique et d'une clé privée. Chaque clé est à sens unique, ce qui signifie que la clé utilisée pour chiffrer des informations ne peut pas être utilisée pour déchiffrer les mêmes données.
Authentification [authentication]	Confirmer une identité.
Authorization certificate [Certificat d'autorisation]	Document électronique qui prouve les droits d'accès et les privilèges de quelqu'un, et qui prouve aussi qu'il est bien ce qu'il prétend être.
Authorization [Autorisation]	Transmettre une approbation officielle, un pouvoir légal ou un droit d'accès à une entité.
Blind signature [Signature aveugle]	Habilitation à signer des documents sans connaître leur contenu, similaire à un notaire.
Block cipher [Chiffre par blocs]	Chiffre symétrique opérant sur des blocs de texte clair et de texte chiffré, habituellement de 64 bits.
Blowfish	Algorithme de chiffrement par bloc de 64 bits composé d'une expansion de clé et d'un chiffrement de données; rapide, simple et compact, dans le domaine public, écrit par Bruce Schneier.
CA (Certificate Authority) [Autorité de Certification]	Tiers de confiance (TTP) qui crée des certificats composés d'assertions sur divers attributs, et les associe à une entité et/ou leurs clés publiques.
CAPI (Crypto API)	L'API de crypto de Microsoft pour les systèmes et les applications basés sur Windows.
Capstone	Puce cryptographique développée par la NSA qui met en œuvre un système de dépôt de clé du gouvernement des Etats-Unis.

CAST	Algorithme de chiffrement par blocs de 64 bits utilisant une clé de 64 bits, six boîtes S ayant 8 bits en entrée et 32 en sortie, développé au Canada par Carlisle Adams et Stafford Tavares.
CBC (Cipher Block Chaining) [Chiffrement par chaînage de blocs]	Processus consistant à combiner par OU exclusif le texte clair avec le cryptogramme précédent avant de le chiffrer, ce qui ajoute un mécanisme de rétroaction à un chiffrement par blocs.
CDK (Crypto Developer Kit) [Kit de développeur crypto]	Environnement documenté, incluant une API pour les tiers qui désirent écrire des applications sûres via une bibliothèque cryptographique propre à un vendeur.
CERT (Computer Emergency Response Team) [Equipe d'intervention d'urgence en informatique]	Bureau qui encourage la prise de conscience en matière de sécurité. CERT fournit une assistance technique 24 heures sur 24 sur les incidents en sécurité d'ordinateurs et de réseaux. Le CERT est situé au Software Engineering Institute dans l'université Carnegie Mellon University à Pittsburgh, PA.
Certificate (digital certificate) [Certificat (certificat numérique)]	Document électronique rattaché à une clé publique par un tiers de confiance, qui fournit la preuve que la clé publique appartient à un propriétaire légitime et n'a pas été compromise.
CFM (Cipher Feedback Mode) [Mode de chiffrement par rétroaction]	Chiffre par bloc utilisé dans un chiffrement de flot auto synchrone.
CDSA (Common Data Security Architecture) [Architecture commune de sécurité des données]	Intel Architecture Labs (IAL) a développé ce cadre de travail pour les problèmes de sécurité des données propres à Internet et aux intranets, pour les produits d'Intel et autres.
Certification	Approbation d'une information par une entité de confiance.
CHAP (Challenge Authentication Protocol) [Protocole d'authentification par challenge]	Mécanisme d'authentification par mot de passe à double sens, basé sur une session.
Cipher text [Texte chiffré (ou cryptogramme)]	Résultat de la manipulation de caractères ou de bits via des substitutions, transpositions, ou les deux.

Clear text [Texte clair (ou libellé)]	Caractères ou bits sous une forme lisible par un humain ou une machine (aussi appelé *plain text*).
Confidentiality [Confidentialité]	Fait de garder quelque chose privé et secret vis à vis de tout le monde sauf ceux qui sont autorisés à le voir.
Cookie	Cookie HTTP Persistant sur le Client –fichier ou information quelconque qui est envoyé par le serveur web au client (votre browser) et qui sert à vous identifier et peut enregistrer des informations personnelles comme votre identité et votre mot de passe, votre adresse e-mail, votre numéro de carte de crédit, et d'autres informations.
CRAB	Chiffrement par blocs de 1024 octets (similaire à MD5) utilisant des techniques d'une fonction de hachage à sens unique, développé par Burt Kaliski et Matt Robshaw aux RSA Laboratories.
Credentials [Référence]	Quelque chose qui fournit une base de "foi" ou de confiance.
CRL (Certificate Revocation List) [Liste de révocation de certificat]	Liste en ligne, à jour, de certificats qui ont été émis précédemment et ne sont plus valides.
Cross-certification [Certification croisée]	Deux (ou plus) organisations ou autorités de certification qui partagent le même niveau de confiance.
Cryptanalysis [Cryptanalyse]	L'art ou la science de transformer des textes chiffrés en données claires sans connaissance initiale de la clé utilisée lors du chiffrement.
CRYPTOKI	Cf. PKCS #11.
Cryptography [Cryptographie]	L'art et la science de création de messages qui sont privés, signés, non modifiés et/ou non répudiés.
Cryptosystem [Cryptosystème]	Système composé d'algorithmes cryptographiques, de tous les textes clairs possibles, de tous les textes chiffrés et de toutes les clés.
Data integrity [Intégrité des données]	Une méthode assurant que l'information n'a pas été altérée par des moyens inconnus ou non autorisés.

Decryption [Déchiffrement]	Le processus transformant le texte chiffré en texte clair.
DES (Data Encryption Standard) [Standard de chiffrement de données]	Chiffre par blocs de 64 bits, algorithme symétrique aussi appelé DEA (Algorithme de Chiffrement de Données) [Data Encryption Algorithm] par l'ANSI et DEA-1 par l'ISO. Largement utilisé depuis plus de 20 ans, le standard FIP 46 a été adopté en 1976.
Dictionnary attack [Attaque par dictionnaire]	Attaque par force brute qui révèle les mots de passe évidents et des combinaisons logiques de mots.
Diffie-Hellman	Premier algorithme à clé publique, inventé en 1976. Il utilise les logarithmes discrets sur un corps fini.
Digital cash [Argent numérique]	Argent électronique stocké et transféré via divers protocoles compliqués.
Direct trust [Confiance directe]	Mise en place de confiance point à point.
Discrete logarithm [Logarithme discret]	Problème mathématique sous-jacent aux algorithmes asymétriques comme Diffie-Hellman et les Courbes Elliptiques. C'est le problème inverse de l'exponentiation modulo N, qui est une fonction à sens unique.
DMS (Defense Messaging System) [Système de Messagerie de la Défense]	Standards conçus par le Département de la Défense américaine pour fournir une infrastructure de messagerie à grande échelle sûre et fiable pour les agences gouvernementales et militaires.
DNSSEC (Domain Name System Security Working Group) [Groupe de travail sur la sécurité du système de noms de domaines]	Brouillon (*draft*) IETF qui spécifiera des améliorations au protocole DNS pour le protéger contre des modifications illicites de données et contre la falsification de l'origine des données. Il ajoutera des mécanismes d'intégrité de données et d'authentification au DNS via les signatures numériques.
DSA (Digital Signature Algorithm)	Algorithme de signature à clé publique proposé par le NIST pour être utilisé dans DSS.
Digital signature [Signature numérique]	Identification électronique d'une personne ou chose créée par un algorithme à clé publique. Destiné à vérifier l'intégrité des données et l'identité de l'émetteur.

DSS (Digital Signature Standard) [Standard de signature numérique]	Standard (FIPS) proposé par le NIST pour les signatures numériques en utilisant DSA.
ECC (Elliptic Curve Cryptosystem) [Cryptosystème à courbes elliptiques]	Méthode unique de création d'algorithmes à clé publique basés sur des courbes mathématiques sur des corps finis ou des grands nombres premiers.
EDI (Electronic Data Interchange) [Echange Electronique de Données]	Echange direct standardisé, d'ordinateur à ordinateur, de documents d'affaires (ordres d'achats, mandats, paiements, analyses de stock, etc.) entre votre organisation et vos fournisseurs et clients.
EES (Escrowed Encryption Standard) [Standard de chiffrement à dépôt]	Standard proposé par le gouvernement américain pour le dépôt des clés privées.
Elgamal scheme [Schéma d'Elgamal]	Utilisé pour les signatures numériques et le chiffrement; basé sur les logarithmes discrets dans un corps fini; peut être utilisé par la fonction DSA.
Encryption [Chiffrement]	Processus consistant à déguiser un message de façon à cacher sa substance.
Entropy [Entropie]	Mesure mathématique de la quantité d'incertitude ou d'aléa.
FEAL	Chiffre par blocs de 64 bits avec une clé de 64 bits, conçu par A. Shimizu et S. Miyaguchi à NTT Japon.
Filter [Filtre]	Fonction, ensemble de fonctions ou combinaison de fonctions qui applique un certain nombre de transformations à son espace d'entrée, produisant en sortie un ensemble contenant seulement les éléments en entrée qui respectent certains critères. Les éléments sélectionnés peuvent être transformés ou non. Un exemple serait une fonction de recherche qui accepte des chaînes ayant une relation booléenne (comme a ou comme b mais ne contenant pas c) et force optionnellement la casse des chaînes trouvées en sortie.
Fingerprint [Empreinte (numérique)]	Identificateur de clé unique obtenu en hachant certaines portions des données de la clé.

FIPS (Federal Information Processing Standard) [Standard de traitement de données fédéral]	Standard gouvernemental américain publié par le NIST.
Firewall [Pare-feu (ou garde-barrière)]	Ensemble de matériels et de logiciels qui protège le périmètre du réseau public / privé contre certaines attaques pour atteindre un bon degré de sécurité.
GAK (Government Access to Keys) [Accès gouvernemental aux clés]	Méthode pour le gouvernement de détourner les clés privées des individus.
Gost (GOST 28147-89 en fait)	Chiffre symétrique par blocs de 64 bits utilisant une clé de 256 bits, développé dans l'ex-Union Soviétique.
GSS-API (Generic Security Services API) [API générique de services de sécurité]	API de sécurité de haut niveau basé sur le RFC 1508 qui isole le code applicatif orienté session des détails de mise en œuvre.
Hash function [Fonction de hachage]	Fonction de hachage à sens unique – une fonction qui produit un résumé (une empreinte) d'un message qui ne peut pas être inversé pour reproduire l'original.
HMAC	Fonction de hachage à sens unique dépendant de la clé, destinée à être utilisée avec MAC (voir ce mot), basée sur le RFC 2104.
Hierarchical trust [Confiance hiérarchique]	Série étagée d'entités qui accorde la confiance de façon organisée, communément utilisé dans l'ANSI X.509 aboutissant aux autorités de certification.
HTTP (HyperText Transfer Protocol) [Protocole Transfert d'hypertexte]	Protocole commun utilisé pour transférer des documents entre serveurs ou d'un serveur à un client.
IDEA (International Data Encryption Standard) [Standard international de chiffrement de données]	Chiffre symétrique par blocs de 64 bits utilisant des clés de 128 bits, basé sur des opérations de mélange dans divers groupes algébriques. Considéré comme l'un des algorithmes les plus forts.

IETF (Internet Engineering Task Force) [groupe spécial d'ingénierie d'Internet]	Grande communauté internationale ouverte de concepteurs de réseaux, d'opérateurs, de vendeurs et de chercheurs intéressés par l'évolution de l'architecture et le fonctionnement sans heurt d'Internet. Elle est ouverte à tout individu intéressé.
Identity certificate [Certificat d'identité]	Formulaire signé qui lie une clé au nom d'un individu avec l'intention de déléguer l'autorité de cet individu à la clé publique.
Initialization vector (IV) [Vecteur d'initialisation]	Bloc de données arbitraire qui sert de point de départ pour les chiffres par blocs qui utilisent un mécanisme de chaînage ou de rétroaction (voir CBC)
Integrity [Intégrité]	Assurance que les données n'ont pas été modifiées (par des personnes non autorisées) pendant le stockage ou la transmission.
IPSec	Couche TCP/IP de chiffrement en cours d'examen par l'IETF.
ISA/KMP (Internet Security Association, Key Mgt. Protocol) [association pour la sécurité d'Internet, protocole de gestion de clés]	Définit les procédures pour l'authentification des deux parties d'une communication, la création et la gestion d'associations de sécurité [Security Associations], les techniques de génération de clés, et l'atténuation des menaces, par exemple le blocage de service et les attaques par rejeu.
ISO (International Organization for Standardization) [Organisation de standardisation internationale]	Responsable d'une large gamme de standards, comme le modèle OSI et les relations internationales avec l'ANSI sur le X. 509.
ITU-T (International Telecommunication Union-Telecommunication) [Union internationale pour les télécommunication]	Anciennement le CCITT (Comité consultatif pour le télégraphe et le téléphone internationaux) [Consultative Committee for International Telegraph and Telephone], une organisation mondiale de standardisation de technologies de télécommunications.
Kerberos	Protocole d'authentification basé sur un tiers de confiance, développé au MIT.
Key [Clé]	Moyen d'obtenir ou de refuser un accès, une possession ou un contrôle, représenté par une valeur parmi un grand nombre.

Key escrow/recovery [Dépôt de clé]	Mécanisme qui permet à un tiers de récupérer des clés cryptographiques utilisées pour la confidentialité de données, dans le but de déchiffrer ces données cryptées.
Key exchange [Echange de clé]	Mécanisme permettant à deux ou plusieurs nœuds de transférer une clé de session secrète sur un canal non sûr.
Key length [Longueur de clé]	Nombre de bits représentant la taille de la clé; plus c'est long et plus c'est robuste.
Key management [Gestion de clé]	Processus et procédure pour stocker et distribuer de façon sûre des clés cryptographiques; l'ensemble du processus de génération et de distribution des clés aux destinataires autorisés de façon sûre.
Key splitting [Scission de clé]	Processus pour répartir des segments d'une même clé entre plusieurs parties, aucune n'ayant la possibilité de reconstituer l'ensemble de la clé.
LDAP (Lightweight Directory Access Protocol) [Protocole léger d'accès à répertoire]	Simple protocole qui supporte des opérations d'accès et de recherche dans des répertoires contenant des renseignements comme des noms, des numéros de téléphones, et des adresses, entre des systèmes qui seraient sinon incompatibles sur tout Internet.
Lexical section [Section lexicale]	Paragraphe distinct d'un message qui contient une classe de données spécifique, par exemple des données en clair signées, des données chiffrées, et des données de clé.
MAA (Message Authenticator Algorithm) [algorithme d'authentification de message]	Standard ISO qui produit une empreinte de 32 bits, conçu pour les mainframes IBM.
MAC (Message Authentication Code) [code d'authentification de message]	Fonction de hachage à sens unique dépendant d'une clé, nécessitant l'utilisation d'une clé identique pour vérifier le hachage.
MD2 (Message Digest 2)	Fonction de hachage à sens unique conçue par Ron Rivest, dépendant d'une permutation aléatoire d'octets.
MD4 (Message Digest 4)	Fonction de hachage à sens unique de 128 bits conçue par Ron Rivest, utilisant un ensemble simple de manipulations de bits sur des opérandes de 32 bits.

MD5 (Message Digest 5)	Version améliorée de MD4 et plus complexe, produisant toujours une empreinte de 128 bits.
Message digest [Résumé de message]	Nombre dérivé d'un message. Changez un seul caractère dans le message et l'empreinte sera différente.
MIC (Message Integrity Check) [Vérification d'intégrité de message]	Défini à l'origine dans PEM pour l'authentification via MD2 ou MD5. Micalg (calcul d'intégrité de message) [message integrity calculation] est utilisé dans MIME sécurisé.
MIME (Multipurpose Internet Mail Extensions) [extensions à buts variés du courrier Internet]	Ensemble librement disponible de spécifications offrant un moyen d'échanger du texte dans des langues avec des jeux de caractères différents, et d'envoyer du courrier électronique multimédia entre des systèmes informatiques différents.
MMB (Modular Multiplication-based Block) [bloc basé sur une multiplication modulaire]	John Daemen a développé cet algorithme de chiffrement par blocs de 128 bits avec clés de 128 bits basé sur IDEA, non utilisé à cause de sa sensibilité à la cryptanalyse linéaire.
MOSS (MIME Object Security Service) [Service de sécurité des objets MIME]	Défini dans le RFC 1848, il facilite le chiffrement et les services de signature pour MIME, y compris la gestion des clés basée sur des techniques asymétriques (pas très répandu).
MSP (Message Security Protocol) [protocole de sécurité de message]	Equivalent militaire de PEM, un protocole de niveau applicatif compatible avec X.400 pour sécuriser le courrier électronique, développé par la NSA à la fin des années 80.
MTI	Protocole d'agrément en une passe par Matsumoto, Takashima, et Imai qui fournit l'authentification mutuelle de clés sans confirmation de clé ni entité d'authentification.
NAT (Network Address Translator) [Traducteur d'adresse réseau]	RFC 1631, un routeur connectant deux réseaux ensemble; l'un désigné comme l'intérieur est adressé soit avec une adresse privée soit avec une adresse obsolète qui doit être convertie en adresse légale avant que les paquets soient envoyés à l'autre réseau (désigné comme l'extérieur).
NIST (National Institute for Standards and Technology) [institut national pour les standards et la technologie]	Division du département américain du commerce qui publie des standards ouverts d'interopérabilité appelés FIPS.

Non-repudiation [Non répudiation]	Empêche le reniement d'actions ou de messages anciens.
Oakeley	L'"échange de clés de sessions de Oakley" fournit un échange de clé de session de Diffie Hellman hybride destiné à être utilisé dans un cadre ISA/KMP. Oakley amène la propriété importante de Perfect Forward Secrecy [secret parfait en avant]
One-time pad [Masque jetable (ou clé aléatoire une fois)]	Grand ensemble de lettres réellement aléatoire, non répétitif, considéré comme le seul système de chiffrement parfait, inventé par le Major J. Mauborgne et G. Vernam en 1917.
One-way hash [Hachage à sens unique]	Fonction d'une chaîne variable pour créer une valeur de longueur fixe représentant l'original, appelé résumé de message, empreinte, vérification d'intégrité de message (MIC).
Orange Book [Livre orange]	Livre du National Computer Security Center [Centre National de Sécurité des Ordinateurs] intitulé *Department of Defense Trusted Computer Systems Evaluation Criteria* [Critères d'évaluation des Systèmes informatiques certifiés par le département de la Défense] qui définissent les besoins de sécurité.
PAP (Password Authentication Protocol) [protocole d'authentification par mot de passe]	Protocole d'authentification qui permet à des "pairs" PPP de s'identifier l'un l'autre, n'empêche pas l'accès non autorisé mais identifie juste l'autre bout.
Passphrase [Phrase secrète]	Phrase facile à retenir utilisée pour une meilleure sécurité qu'un simple mot de passe; le broyage de clé la converti en une clé aléatoire.
Password [Mot de passe]	Séquence de caractères ou mot qu'un sujet donne à un système pour authentification, validation ou vérification.
PCT (Private Communication Technology) [technologie de communication privée]	Protocole développé par Microsoft et Visa pour des communications sûres sur Internet.
PEM (Privacy Enhanced Mail) [Courrier à confidentialité améliorée]	Protocole qui fournit du courrier sûr sur Internet (RFC 1421-1424), y compris des services de chiffrement, authentification, intégrité de messages, et gestion de clés. PEM utilise des certificats X.509.

Perfect forward secrecy [Secret parfait en avant]
Cryptosystème dans lequel le texte chiffré ne fournit aucune information sur le texte clair si ce n'est la longueur, éventuellement.

Primitive filter [Filtre primitif]
Fonction qui applique une transformation simple à son espace d'entrée, produisant un ensemble de sortie ne contenant que les membres de l'ensemble d'entrée qui satisfont au critère. Un exemple serait une fonction de recherche qui n'accepterait qu'une chaîne simple et produisant en sortie les numéros de lignes où cette chaîne a été trouvée.

PGP (Pretty Good Privacy) [Assez bonne confidentialité]
Application et protocole (RFC 1991) pour le courrier électronique sécurisé et le chiffrement de fichiers développé par Phil R. Zimmermann. Diffusé gratuitement à l'origine, le code source a toujours été disponible et inspecté. PGP utilise divers algorithmes comme IDEA, RSA, DSA, MD5, SHA-1 pour le chiffrement, l'authentification, l'intégrité des messages et la gestion de clés. PGP est basé sur le modèle de la "toile d'araignée de confiance" et est utilisé dans le monde entier.

PGP/MIME
Standard IETF (RFC 2015) qui fournit authentification et confidentialité via les types de contenus de sécurité de MIME décrit dans le RFC 1847, mis en œuvre actuellement dans PGP 5.0 et les versions suivantes.

PKCS (Public Key Crypto Standards) [Standards de crypto à clé publique]
Ensemble de standards de fait pour la cryptographie à clé publique développé en coopération avec un consortium informel (Apple, DEC, Lotus, Microsoft, MIT, RSA, et Sun) qui comprend des standards de mise en œuvre spécifiques à des algorithmes ou indépendants de tout algorithme. Les spécifications définissant la syntaxe des messages et d'autres protocoles sont contrôlés par RSA Data Security Inc.

PKI (Public Key Infrastructure) [Infrastructure à clé publique]
Système de certificats largement disponible et accessible pour obtenir la clé publique d'une entité, avec une bonne probabilité que vous ayez la "bonne" clé et qu'elle n'ait pas été révoquée.

Plain text (or clear text) [Texte clair (ou libellé)]
Données ou message lisible par un humain avant chiffrement.

Pseudo-random number [Nombre pseudo aléatoire]
Nombre résultant d'algorithmes de confusion de l'entrée dérivée de l'environnement de l'ordinateur, comme les coordonnées de la souris. Voir *random number* [nombre aléatoire].

Private key [Clé privée]
Composant gardé "secret" d'une paire de clés asymétriques, souvent appelé "clé de déchiffrement".

Public key [Clé publique]	Composant disponible publiquement d'une paire de clés asymétriques, souvent appelé "clé de chiffrement".
RADIUS (Remote Authentication Dial-In User Service) [Service utilisateur d'authentification par téléphone]	Protocole IETF (développé par Livingston, Enterprise), pour la sécurité distribuée qui verrouille les accès réseau à distance et les services réseau contre les accès non autorisés. RADIUS est composé de deux parties – le code serveur d'authentification et les protocoles client.
Random number [Nombre aléatoire]	Aspect important de bien des cryptosystèmes, et élément nécessaire pour générer une clé unique que l'adversaire ne puisse pas prédire. Les vrais nombres aléatoires sont habituellement dérivés de sources analogiques, et nécessitent du matériel spécial.
RC2 (Rivest Cipher 2)	Chiffre symétrique par blocs de 64 bits et clé de longueur variable. Secret commercial de RSA, DSI.
RC4 (Rivest Cipher 4)	Chiffrement de flux à clé de longueur variable, algorithme propriétaire de RSA Data Security Inc pendant un temps.
RC5 (Rivest Cipher 5)	Chiffre par blocs à arguments, taille de blocs, taille de clé et nombre de rondes variables.
RIPE-MD	Algorithme développé par le projet européen RIPE, conçu pour résister aux attaques cryptanalytiques connues et produire un hachage de 128 bits, une variante de MD4.
REDOC	Algorithme de chiffrement par bloc breveté aux USA, développé par M. Wood, utilisant une clé de 160 bits et des blocs de 80 bits.
Revocation	Désaveu d'un certificat ou d'une autorisation.
RFC (Request for Comment) [Demande de Commentaire]	Dcument IETF, soit de la série FYI [for your information] (pour votre information) soit de la série STD qui spécifie des standards Internet. Chaque RFC a un numéro qui permet de le retrouver (www.ietf.org)
ROT-13 (Rotation Cipher) [Chiffre par rotation]	Chiffre par substitution simple (César), déplaçant chaque lettre de 13.

RSA

Abrégé de RSA Data Security, Inc.; ou bien ses principaux responsables – Ron Rivest, Adi Shamir et Len Aldeman; ou bien l'algorithme qu'ils ont inventé. L'algorithme RSA est utilisé pour la cryptographie à clé publique et est basé sur le fait qu'il est facile de multiplier deux grands nombres premiers mais difficile de factoriser le produit.

SAFER (Secure And Fast Encryption Routine) [routine de chiffrement sûre et rapide]

Chiffrement par blocs de 64 bits. Il n'est pas breveté, est disponible gratuitement sans licence, et a été développé par Massey, qui a aussi développé IDEA.

Salt [Sel]

Chaîne aléatoire concaténée aux mots de passe (ou aux nombres aléatoires) avant de les passer à travers une fonction de hachage à sens unique. Cette concaténation allonge et obscurcit le mot de passe, rendant le texte chiffré moins sensible aux attaques par dictionnaire.

SDSI (Simple Distributed Security Infrastructure) [Infrastructure de sécurité distribuée simple]

Nouvelle proposition de PKI par Ronald L. Rivest (MIT) et Butler Lampson (Microsoft). Elle fournit un moyen de définir des groupes et des appartenances à ces groupes, des listes de contrôle d'accès, et des politiques de sécurité. La conception de SDSI insiste sur des espaces de noms locaux liés plutôt que sur un espace de nom global hiérarchisé.

SEAL (Software-optimized Encryption ALgorithm) [Algorithme de chiffrement optimisé pour le logiciel]

Algorithme de chiffrement de flux rapide sur les machines 32 bits conçu par Rogaway et Coppersmith.

Secret key [Clé secrète]

"Clé privée" d'un algorithme à clé publique (ou asymétrique), ou bien "clé de session" dans les algorithmes symétriques.

Secure channel [Canal sûr]

Moyen de communication entre deux entités tel qu'un adversaire ne puisse pas changer l'ordre, supprimer, insérer ou lire de l'information (exemples: SSL, IPsec, chuchoter dans l'oreille de quelqu'un).

Self-signed key [Clé auto signée]

Clé publique qui a été signée par la clé privée correspondante comme preuve de propriété.

SEPP (Secure Electronic Payment Protocol) [protocole de paiement électronique sûr]

Spécification ouverte pour des transactions bancaires sûres sur Internet. Développé par IBM, Netscape, GTE, Cybercash et MasterCard.

SESAME (Secure European System for Applications in a Multi-vendor Environment) [système européen sûr pour des applications en environnement multi vendeurs]	Projet européen de recherche et développement qui étend Kerberos en ajoutant des services d'autorisation et d'accès.
Session key [Clé de session]	Clé secrète (symétrique) utilisée pour chiffrer chaque jeu de données dans un système de transaction. Une clé de session différente est utilisée pour chaque session de communication.
SET (Secure Electronic Transaction) [Transaction électronique sûre]	Fournit un échange sûr de numéros de cartes de crédit sur Internet.
SHA-1 (Secure Hash Algorithm) [Algorithme de hachage sûr]	Version de 1994 du SHA, développé par le NIST (FIPS 180-1); utilisé avec le DSS il fournit une empreinte de 160 bits, similaire à MD4, très populaire et largement utilisé.
Single sign-on [Engagement simple]	Une seule connexion (log-on) permet d'accéder à toutes les ressources du réseau.
SKIP (Simple Key for IP) [clé simple pour IP]	Gestion de clé simple pour les protocoles Internet, développé par Sun Microsystems, Inc.
Skipjack	L'algorithme de chiffrement avec clés de 80 bits contenu dans les puces Clipper de la NSA.
SKMP (Secure key Management Protocol) [Protocole de gestion de clé sûr]	Architecture de récupération de clé proposé par IBM, qui utilise une technique d'encapsulation de clé qui fournit la clé (et la possibilité de déchiffrer le message) à un tiers de confiance.
S/MIME (Secure Multipurpose Mail Extension) [MIME sûr]	Standard proposé par Deming Software et RSA Data Security pour chiffrer et/ou authentifier des données MIME. S/MIME définit un format pour les données MIME, les algorithmes qui doivent être utilisés pour l'interopérabilité (RSA, RC2, SHA-1) et les problèmes opérationnels supplémentaires comme les certificats ANSI X.509 et le transport via Internet.

SNAPI (Secure Network API) [API de réseau sûre]
Interface de programmation lancée par Netscape pour les services de sécurité. Protège des ressources contre des utilisateurs non authentifiés, chiffre et authentifie des communications, et vérifie l'intégrité de l'information.

SPKI (Simple Public Key Infrastructure) [Infrastructure de clé publique simple]
Brouillon de standard (*draft*) de format de certificats à clé publique, des signatures associées et d'autres formats, ainsi que le protocole d'acquisition de clé, proposé à l'IETF (par Ellison, Franz et Thomas). Récemment fusionné avec la proposition SDSI de Ron Rivest.

SSH (Secure Shell) [shell sûr]
Protocole proposé par l'IETF pour sécuriser la couche transport en fournissant chiffrement, authentification cryptographique de l'hôte et protection de l'intégrité.

SSH (Site Security Handbook) [livret de sécurité de site]
Le groupe de travail (WG) de l'IETF a planché depuis 1994 pour produire deux documents destinés à éduquer la communauté Internet sur la sécurité. Le premier document est une refonte complète du RFC 1244 et vise les administrateurs système et réseau ainsi que les décideurs (hiérarchie intermédiaire).

SSL (Secure Socket Layer) [Couche de sockets sûres]
Développé par Netscape pour fournir sécurité et confidentialité sur Internet. Supporte l'authentification du serveur et du client et assure la sécurité et l'intégrité du canal de transmission. Opère au niveau de la couche de transport et imite la "bibliothèque des sockets", permettant d'être indépendant de l'application. Chiffre complètement le canal de communication et ne supporte pas les signatures numériques au niveau du message.

SST (Secure Transaction Technology) [technologie de transactions sûres]
Protocole de paiement sûr développé par Microsoft et Visa comme "compagnon" du protocole PCT.

Stream cipher [Chiffrement de flux]
Classe de chiffre à clé symétrique où la transformation peut être changée à chaque symbole de texte clair, utile pour les équipements qui ont peu de mémoire tampon.

STU-III (Secure Telephone Unit) [unité de téléphone sûre]
Téléphone sécurisé conçu par la NSA pour la voix et des communications de données à basse vitesse pour le Département américain de la Défense et ses fournisseurs.

Substitution cipher [Chiffre par substitution]
Les caractères du texte clair sont changés en d'autres caractères pour former le texte chiffré.

S/WAN (Secure Wide Area Network) [réseau géographiquement étendu sûr]	Spécifications de mise en oeuvre d'IPsec lancées par RSA Data Security, Inc., pour assurer l'interopérabilité des gardes-barrières et produits TCP/IP. Le but de S/WAN est d'utiliser IPsec pour permettre à des entreprises de mélanger des gardes-barrières et des piles TCP/IP pour construire des Virtual Private Networks (VPNs) [réseaux privés virtuels] basés sur Internet.
Symetric algorithm [Algorithme symétrique]	Algorithmes conventionnels, à clé secrète, ou à clé unique; les clés de chiffrement et de déchiffrement sont identiques ou peuvent être facilement calculées l'une à partir de l'autre. Deux sous catégories existent – par bloc ou par flux.
TACACS+ (Terminal Access Controller Access Control System) [système de contrôle d'accès / contrôleur d'accès terminal]	Protocole qui fournit des services d'authentification d'accès distant, d'autorisation, et la comptabilité et la journalisation, utilisé par Cisco Systems.
Timestamping [Horodatage]	Enregistrement de la date de création ou d'existence d'une information.
TLS ([Transport Layer Security) [sécurité de la couche transport]	Brouillon IETF, la version 1 est basée sur la version 3.0 du protocole SSL, et assure la confidentialité des communications sur Internet.
TLSP (Transport Layer Security Protocol) [protocole de sécurité de la couche transport]	ISO 10736, brouillon de standard international.
Transposition cipher [Chiffre par transposition]	Le texte clair reste le même mais l'ordre des caractères est modifié.
Triple DES	Configuration de chiffrement dans lequel l'algorithme DES est utilisé trois fois de suite avec trois clés différentes.
Trust [Confiance]	Certitude ou confiance dans l'honnêteté, l'intégrité, la rectitude et/ou la fiabilité d'une personne, entreprise ou toute autre entité.
TTP (Trust Third-Party) [Tiers de confiance]	Tiers responsable sur lequel tous les participants se mettent d'accord par avance, pour fournir un service ou une fonction comme la certification, en associant une clé publique à une entité, l'horodatage, ou le dépôt de clé.

UEPS (Universal Electronic Payment System) [système de paiement électronique universel]	Application bancaire basée sur des cartes à puces (carte de crédit sécurisée) développée pour l'Afrique du Sud où les téléphones de mauvaise qualité rendent la vérification en ligne impossible.
Validation	Moyen de fournir une autorisation pour utiliser ou manipuler de l'information ou des ressources.
Verification	Authentifier, confirmer ou établir la véracité.
VPN (Virtual Private Network) [réseau privé virtuel]	Etend des réseaux privés de l'utilisateur final à la passerelle de son choix, tel l'intranet de son entreprise, via un réseau public (Internet).
WAKE (Word Auto Key Encryption) [Chiffrement autoclave par mot]	Fournit un flux de mots de 32 bits, qui peuvent être combinés par XOR (ou exclusif) avec le texte clair pour former un texte chiffré, inventé par David Wheeler.
Web of Trust [Toile d'araignée de confiance]	Modèle de confiance distribuée utilisé par PGP pour valider la propriété d'une clé publique – le niveau de confiance est une somme basée sur les connaissances individuelles des "avals".
W3C (World Wide Web Consortium) [consortium de la toile d'araignée mondiale]	Consortium industriel international fondé en 1994 pour développer des protocoles communs pour l'évolution du web.
XOR [Ou exclusif]	Opération ou exclusif; une façon mathématique de représenter des différences.
X.509v3	Certificat numérique ITU-T qui est un document électronique reconnu à l'échelle internationale pour prouver l'identité et la propriété d'une clé publique sur un réseau. Il contient le nom de celui qui l'émet, son information d'identification, sa signature numérique, ainsi que d'autres extensions possibles en version 3.
X9.17	Spécification ANSI qui détaille la méthodologie de génération de nombres aléatoires et pseudo aléatoires.

Index

D

E

F

I

L

M

N

P

R

S

T

V

Z