

P.G.P.

Pretty Good Privacy

“Un système de chiffrement sûr enfin à la portée de tous.”

Ollivier Robert

Hervé Schauer Consultants

<roberto@hsc.fr.net>

1. Pourquoi PGP ?

PGP a été créé dans un but précis : offrir à tout le monde un moyen de préserver la confidentialité d'une information, quelle qu'elle soit, personnelle, professionnelle ou autre.

Ces informations peuvent être des messages du courrier électronique (l'usage le plus fréquent de PGP), des fichiers que l'on souhaite archiver ou encore des documents dont on souhaite garantir le contenu.

2. Rappels sur la cryptographie

On distingue actuellement deux sortes de système de chiffrements :

1. les systèmes à clé privée
2. les systèmes à clé publique

2.1 Systèmes à clé privée

Ces systèmes utilisent une clé unique pour chiffrer et déchiffrer. Ce sont les systèmes les plus répandus.

Mathématiquement parlant : Soit une clé K et un texte clair P (pour *plaintext*), on obtient le texte chiffré C (pour *cyphertext*) par l'application de l'algorithme f

$$C = f_K(P)$$

Pour rétablir le texte clair, le correspondant applique soit un deuxième algorithme (système asymétrique) soit le même algorithme avec la même clé :

$$P = f'_K(C)$$

avec éventuellement $f = f'$

Le cas le plus fréquent pour les systèmes électroniques est l'utilisation d'un système symétrique car il est plus simple à implanter.

La sécurité de ces systèmes réside entièrement dans le secret de la clé et nécessite donc un canal sécurisé de transmission de la clé.

Un des problèmes que l'on peut rencontrer est le volume représenté par les clés, leur production et leur acheminement. Ce problème est souvent évoqué pour le système appelé *clé aléatoire une fois*.

Cela dit, ce dernier est le seul qui soit théoriquement incassable.

Les systèmes électroniques les plus connus actuellement sont DES (*Data Encryption Standard*), Triple DES, Lucifer, Loki et plus récemment *Skipjack*, un système mis au point par la NSA aux Etats-Unis.

Skipjack est classifié de manière à empêcher quiconque de produire un clone sans certaines caractéristiques particulières liées aux écoutes.

Les deux algorithmes les plus utilisés aujourd'hui sont DES, qui opère sur des blocs de 64 bits avec une clé de 56 bits et IDEA qui opère aussi sur des blocs de 64 bits mais qui utilise une clé de 128 bits.

2.2 Systèmes à clé publique

Nous avons vu le principal inconvénient des systèmes à clé privée : la sécurité de ladite clé.

Conscients de ce problème, deux chercheurs américains, Whitfield Diffie et Martin Hellman, ont inventé un nouveau concept : celui dans lequel, le système n'utilise plus **une** mais **deux** clés. L'une sert au chiffrement et l'autre au déchiffrement.

La clé de chiffrement est publique - voire publiée dans une sorte d'annuaire - celle qui sert au déchiffrement est secrète. Le principe est celui de la boîte aux lettres : tout le monde peut déposer des lettres dans la boîte mais seule une personne, le propriétaire, peut les en retirer.

Ils présentèrent leur concept à la **National Computer Conference** en 1976 et leur papier “*New directions in Cryptography*” apparut quelques mois plus tard dans les papiers “*IEEE Transactions on Information Theory*”.

Mathématiquement, on obtient ceci : soit le texte clair P , une clé privée pr , une clé publique pu et l’algorithme f , nous avons

$$C = f_{pu}(P)$$

et

$$P = f_{pr}(C)$$

Il existe bien entendu une relation mathématique entre les deux clés mais cette relation est sensée être dans un sens, c’est-à-dire qu’il est simple à partir de la clé privée de générer la clé publique mais l’inverse doit être considéré comme très difficile ou “avec les moyens actuels, impraticable”.

Le système de ce type le plus connu est RSA (du nom de ses auteurs Rivest Shamir et Adleman), né en 1978. Il est basé sur la difficulté de factorisation de très grands nombres, problème qui reste à ce jour considéré comme impraticable pour des nombres de plus de 512 bits.

2.3 Principe des signatures

L’un des grands intérêts des systèmes à clé publique a été de permettre la création de signatures irrévocables et sûres de documents. Le principe utilise la dualité des clés pour sécuriser le document.

Pour cela et afin d’optimiser les performances, le texte entier est passé à un algorithme appelé *Message Digest* qui, à partir d’un texte donné, génère une valeur (sur un nombre fixe de bits) qui identifie de manière sûre le document. C’est cette valeur qui est ensuite chiffrée et envoyée avec le message. L’intérêt réside dans le fait que quelle que soit la longueur du texte clair, la signature reste de taille fixe, ce qui est pratique.

Mathématiquement, soit M un message de taille arbitraire et H une fonction appelée *one-way hash function*, h est le résultat de l’application de H sur M

$$h = H(M)$$

avec h de longueur m .

Ces fonctions ont les caractéristiques suivantes, garantissant ainsi la sécurité de l’ensemble :

⇒ Si nous avons M , il est simple d’obtenir h .

⇒ Si nous avons h , il est compliqué d’obtenir M .

⇒ Si nous avons M , il est compliqué d’obtenir un autre message $M^{e'}$ tel que $H(M) = H(M^{e'})$.

Les plus connus des *message digests* sont MD2, MD4 et MD5, inventés par Don Rivest ainsi que SHA, récemment adopté par l’Administration Américaine comme le *message digest* officiel et utilisé par le système DSA (*Digital Signature Algorithm*).

Pour permettre de vérifier facilement la signature, le programme va générer un *message digest* du message et le chiffrer avec la clé *privée* de l’utilisateur. Cela permet aux correspondants qui possèdent la clé publique de l’expéditeur, de déchiffrer le *message digest* et ainsi vérifier l’authenticité du message.

3. L’auteur de PGP: Philip Zimmermann

Philip Zimmermann, un programmeur américain, s’est intéressé à RSA dès 1977 mais n’a réellement commencé à écrire PGP qu’en 1984. Son but est de donner au public un moyen de protéger ses données en utilisant RSA.

PGP 1.0, qui ne tournait à l’époque que sur PC avec MS-DOS, fut initialement mis sur un BBS et de là va faire le tour du monde. En moins de temps qu’il n’en faut pour le dire, PGP devient le programme de chiffrement le plus diffusé. Ce qui attire sur lui les foudres de la firme RSA Data Security, Inc. qui possède des brevets sur tout l’ensemble des systèmes à clé publique.

Pour éviter le problème des lois sur l’exportation de matériel cryptographique des USA, les versions suivantes de PGP sont écrites en dehors des USA par des programmeurs qui partagent le même idéal que P. Zimmermann.

Zimmermann est actuellement sous le coup d’une enquête par les Douanes Américaines sur l’exportation de PGP. L’affaire évolue lentement...

4. PGP, le programme

PGP existe sur de multiples plate-formes UNIX, Amiga, Atari, VMS, OS/2, MS-DOS et même VM/CMS. Toutes les versions sont interopérables entre elles évidemment. La seule qui soit quelque peu différente est la version Mac, puisqu’elle est intégrée à l’environnement Macintosh alors que les autres ne sont que des versions “ligne de commande”.

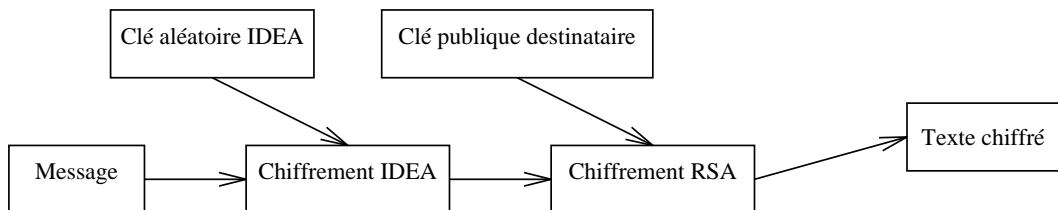


Figure 1. Processus de chiffrement utilisé par PGP

4.1 Principes de fonctionnement

PGP, comme tous les programmes actuellement de ce type, utilise *deux* systèmes de chiffrement pour des raisons de performances et de taille de message.

Le texte est chiffré à l'aide de l'algorithme IDEA. La clé utilisée est générée à chaque session aléatoirement. Comment faire dans ce cas pour que le correspondant ait la clé ?

La figure 1 montre les opérations utilisées par PGP pour chiffrer un message.

Simplement en envoyant la clé IDEA **avec** le message lui-même. La clé IDEA est chiffrée à l'aide de la clé publique RSA du destinataire.

A la réception, le correspondant déchiffre la clé IDEA à l'aide de sa clé privée et déchiffre ainsi le message. Pour garantir la sécurité de la clé secrète RSA, Zimmermann utilise une phrase clé entrée par l'utilisateur pour chiffrer cette clé secrète à l'aide d'un MD5 de la phrase clé.

Voir la figure 2 pour le schéma complet.

Le programme PGP fonctionne à l'aide d'options sur la ligne de commande (il existe des programmes pour en faciliter l'utilisation, généralement sous Windows et sur Mac). Les options sont multiples et il vaut mieux se référer à l'aide en ligne, très complète.

4.2 Encodage

Il existe plusieurs formats de sortie pour les fichiers/messages chiffrés : le format par défaut est un format binaire 8 bits classique. Il a donc l'inconvénient majeur de ne pas pouvoir être envoyé par E-mail.

Pour remédier à ce problème, PGP a adopté le même format que le standard Internet PEM (*Privacy Enhanced Mail*), format similaire dans son esprit au format *uuencode* mais plus sécurisé par rapport aux différents réseaux par lesquels les messages vont circuler.

Ce format, appelé *base64* et défini dans le RFC-1113, utilise le jeu de caractères suivant :

```

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
+ /

```

plus le caractère `=` comme caractère de *padding*.

Les caractères sont pris par groupe de trois en entrée et sortent quatre par quatre (valeurs de 0 à 63). Ces valeurs sont ensuite utilisées comme index de la table précédente.

Ce système a été adopté de par sa robustesse puisque, contrairement à *uuencode* et *base85*, les caractères sont les mêmes sur tous les systèmes, y compris ceux utilisant EBCDIC.

PGP peut chiffrer de deux manières un fichier :

1. si le fichier est destiné à être envoyé par courrier électronique, PGP demande un identificateur de la clé publique du correspondant, clé qui doit être dans le fichier `pubring.pgp`.
2. si le fichier doit juste être chiffré, il faut utiliser l'option "`-c`". Le programme demande alors une phrase clé, *indépendante* de votre clé normale, qui va servir, après passage à MD5, de clé IDEA.

4.3 La signature

La signature sert à assurer qu'un message a bien été écrit par la personne qui possède la clé publique correspondante et garantit que le message n'a pas été altéré durant le transport.

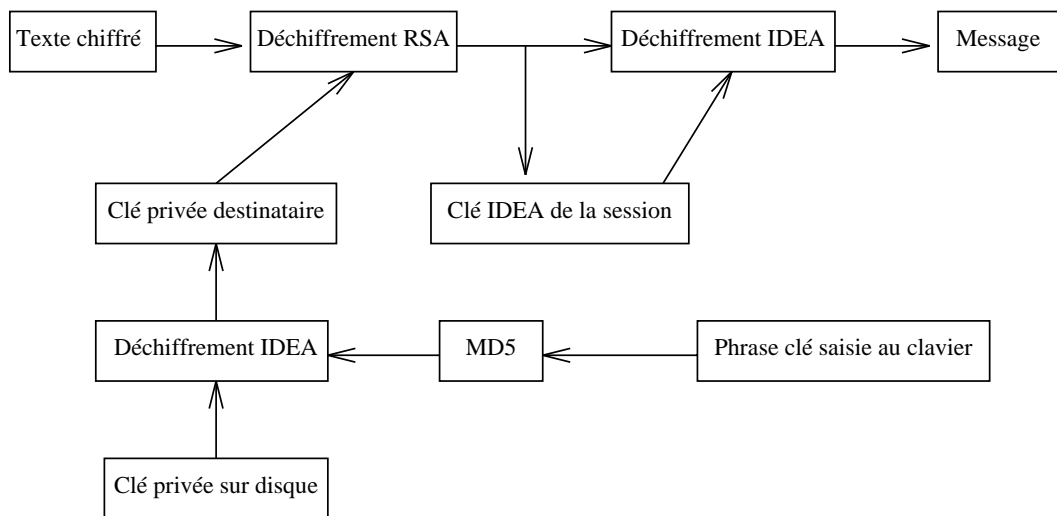


Figure 2. Processus de déchiffrement utilisé par PGP

La signature d'un fichier peut se réaliser de trois manières :

1. avec le fichier, le tout entièrement transformé en base64,
2. avec le fichier mais le texte reste en clair et la signature est en base64 (spécifier l'option `+clearsig=on`),
3. séparé du fichier (options `-b` et `-sb`).

Une chose importante à garder en mémoire : la signature ne garantit pas l'identité de la personne, n'importe qui pouvant créer une clé avec une identité quelconque. Ce n'est que par le biais de la signature *des clés* qu'une identité peut être garantie.

4.4 Gestion des clés

Un aspect important réside dans la gestion des clés, privées et publiques.

Le système adopté par PGP est appelé *Web of Trust*. Il consiste en l'établissement d'un réseau de gestion distribuée des clés.

Le principe consiste pour un utilisateur à faire signer sa clé par un certain nombre d'autres utilisateurs, qui les feront signer par d'autres et ainsi de suite. L'intérêt principal réside dans l'absence d'une autorité centrale de certification contrairement au système PEM qui oblige à avoir un système arborescent.

Le système est ainsi moins lourd à mettre en oeuvre et mieux adapté à l'Internet dans lequel les gens s'échangent les clés et les signent lors de *signing-parties*, à l'occasion de salons généralement.

L'Internet possède également un ensemble maillé de serveurs de clés, accessibles par *finger*, E-mail et *ftp*. Les serveurs s'échangent régulièrement les clés garantissant ainsi un temps de propagation rapide. Ainsi, on espère, par une diffusion rapide et étendue des clés, éviter que quelqu'un ne forge une clé.

4.5 Les différentes versions

La dernière version américaine est la 2.6.1. Elle est "légale" aux USA parce qu'elle utilise, contrairement aux versions précédentes, la librairie RSA de RSA Data Security appelée RSAREF. Elle est bien évidemment interdite d'exportation. Sa principale caractéristique est qu'elle génèrera des messages que seules des post-2.3a (comme la 2.6ui donc pas de panique) pourront lire à partir du 1^{ère} sept. 1994

Pour éviter ce problème et permettre comme avant l'interaction entre les USA et le reste du monde, un Anglais, Mathew Mantis a repris la version 2.3a, qui était la dernière "internationale", a modifié où il fallait pour la rendre compatible avec la 2.6.1 et l'a mise sur Internet sous le nom de 2.6ui, "ui" pour *Unofficial*

International.

Par cette “pirouette”, les Américains sont contents puisqu’il est maintenant légal d’utiliser PGP aux USA grâce à l’accord entre le MIT et RSAADI ; le reste du monde l’est aussi nous avons une version qui n’est pas bridée et qui reste néanmoins, et c’est le principal, compatible avec la 2.6 du MIT.

D’autre part, il existe également les versions commerciales 2.4 (basée sur la 2.3a) et 2.7 (basé sur la 2.6 du MIT) de la firme ViaCrypt. Ces versions utilisent la librairie RSAREF de RSA Data Security, Inc. et sont donc utilisables aux U.S.A. Ce sont les seules qui permettent un usage commercial de PGP.

5. La concurrence

PGP possède plusieurs concurrents :

RIPEM est une implémentation gratuite du système décrit dans les RFC 1113 à 1115 (*Privacy Enhanced Mail* ou PEM). Il utilise RSAREF et supporte les certificats X.509. Les fonctionnalités sont équivalentes à celles de PGP mais beaucoup de gens préfèrent le système du *Web of Trust* à celui de PEM, la plupart ne faisant pas confiance à RSAADI pour son rôle d’Autorité Certificatrice. Une différence majeure avec PGP réside dans le fait que RIPEM utilise DES (et éventuellement Triple-DES) comme moyen de chiffrement conventionnel.

TIS/PEM La firme TIS, connue pour ses logiciels de *Firewall*, a aussi un produit compatible RFC 1113 et X.509. C’est un produit commercial assez cher. Il n’est bien évidemment pas disponible en dehors des U.S.A. ce qui rend son analyse difficile.

Un des inconvénients les plus souvent discuté du système PEM est qu’il est vulnérable à l’analyse de trafic (beaucoup d’informations sont en clair comme l’adresse de l’expéditeur) alors que PGP, lorsqu’il chiffre un message, incorpore ces mêmes informations à l’intérieur du message.

Un article récent dans le groupe `alt.security.pgp` disait que PEM était plutôt un système d’authentification avec le chiffrement en prime alors que PGP est l’inverse : un système de chiffrement qui incorpore également de l’authentification.

Il est impossible par exemple, d’utiliser un *remailer* anonyme¹ avec RIPEM/PEM puisque la signature est visible.

6. La situation en France

Enfin, rappelons que l’usage de tout matériel cryptographique en France est soumis à l’autorisation préalable de la part du SCCSI. Je pense personnellement qu’il soit impossible actuellement qu’un individuel obtienne ce droit, est regrettable.

7. Conclusion

PGP est un outil très pratique, bien écrit et qui répond à un besoin de plus en plus croissant de sécurité sur Internet. L’idéal serait que tout le monde utilise un système de ce genre de manière à rendre le courrier chiffré aussi commun que l’usage des enveloppes pour le courrier classique.

Enfin, voici pour tous ceux qui croiraient qu’en empêchant l’usage de la cryptographie au plus grand nombre, la “Sécurité Nationale” et la sécurité individuelle, notamment en facilitant les écoutes, pourraient être mieux préservées, la phrase célèbre de Zimmermann :

“If privacy is outlawed, only outlaws will have privacy.”

8. Bibliographie

Il existe beaucoup de livres et publications sur le sujet, dont la FAQ de

1. Un *remailer* anonyme est un programme qui, à la réception d’un message, réécrit les en-têtes de manière à masquer l’expéditeur du message. Permet d’envoyer ou de poster des messages anonymement.

`alt.security.pgp` mais la référence principale pour la cryptographie électronique récente est le livre suivant :

Applied Cryptography,
Protocols, Algorithms, and Source Code in C,
Bruce Schneier,
Wiley and Sons, 1993.

Pour ce qui est de la cryptographie en générale et son usage dans l'Histoire, je recommande le livre suivant, souvent cité comme référence :

La Guerre des Codes Secrets
"The Codebreakers"
David Kahn
Interéditions, 1981

La documentation de PGP est également une bonne référence, remarquable sur tous les aspects techniques et politiques.

9. Sites FTP & Usenet

La liste des sites FTP est longue dans le monde, aussi je me limiterai aux sites *hors USA* de manière à ne pas violer les lois US et possédant la 2.6ui.

Site	Catalogue
ftp.demon.co.uk	/pub/pgp
ftp.dsi.unimi.it	/pub/security/crypt
ftp.informatik.uni-hamburg.de	/pub/virus/crypto
ftp.ee.und.ac.za	/pub/crypto/pgp
ftp.demon.co.uk	/pub/amiga/pgp
	/pub/archimedes
	/pub/pgp
	/pub/mac/MacPGP
ftp.funet.fi	/pub/pc/crypt
ftp.dsi.unimi.it	/pub/security
src.doc.ic.ac.uk (Amiga)	/aminet
	/amiga-boing
black.ox.ac.uk	/src/security (DOS)

D'autre part, on discute beaucoup de PGP dans `alt.security` et évidemment dans le groupe consacré à PGP : `alt.security.pgp`.