Le Chiffrement RSA sous Thunderbird

Par JMV
Association LinuxMaine
Pôle Coluche, 72000 LE MANS

Table des matières

Préambule

Ce document se veut une aide à la compréhension du chiffrement dit RSA et à son utilisation avec le client de messagerie Thunderbird, il est réalisé dans le cadre d'actions d'information parrainées par l'association mancelle LinuxMaine.

Dans un premier temps nous présenterons les grandes lignes du chiffrement, l'objectif étant de permettre à un lecteur non spécialiste d'en comprendre les grands principes et de pouvoir l'utiliser dans un client de messagerie.

Ensuite, nous choisirons *Thunderbird* comme client de messagerie et présenterons les outils mis en œuvre pour utiliser le chiffrement RSA.

A. Le Chiffrement RSA.

Le **chiffrement RSA** doit son nom aux initiales de ses trois inventeurs : Rivest, Shamir et Adleman.
C'est un algorithme de cryptographie qu'on peut considérer comme universellement répandu.
il s'agit de construire des messages qui ne puissent être lus **que** par l'expéditeur et son destinataire.

Notre objectif n'est pas de faire un historique du sujet ni même d'en dresser un état de l'art, **signalons cependant** l'url ci-dessous qui fournit un bilan très documenté en matière de cryptographie. Au prix d'une visite approfondie, le lecteur intéressé y trouvera de quoi situer le sujet dans son contexte:
https://www.bibmath.net/crypto/index.php?action=affiche&quoi=moderne/quantique

A.1. Comment ça marche ?

L'algorithme se fonde sur l'utilisation d'une *paire* de clefs, l'une d'elles est dite *clef privée* et l'autre c*lef publique*. Il s'agit d'une paire de clefs en ce sens qu'elles interviennent toutes les deux dans le process de chiffrement/déchiffrement. Utiliser une clef d'une autre paire ne fonctionnerait pas.
Pour communiquer de manière chiffrée, en plus de sa paire de clefs personnelle, on doit disposer des clefs publiques de ses correspondants.
Évidemment :

- la clef privée **doit** rester privée, il y va du bon fonctionnement du chiffrement.
- Par contre, on peut même considérer du point de vue de la sécurité que les clefs publiques sont connues du monde entier.
- On doit s'assurer que les clefs publiques de nos correspondants sont les bonnes, faute de quoi les communications risquent d'être interceptées par des tiers malveillants.

Les explications que l'on trouve sur le web qui parlent du chiffrement RSA sont souvent très compliquées, en fait il suffit de comprendre que :

Ce que l'on chiffre avec une clef,
on le déchiffre avec l'autre.

A.2. Un exemple : Alice veut envoyer un courrier chiffré à Bob[1].

Chacun dispose de sa paire de clefs personnelle et de la clef publique de l'autre (Alice et Bob se sont échangés leurs clefs publiques au préalable).
Voici un possible process de chiffrement/déchiffrement pour un message donné, disons M :

- Alice chiffre le message M avec sa clef privée, résultat MC_1 (message chiffré 1). Cette étape **identifie** Alice comme expéditrice du message, elle est la seule à être capable de produire le message MC_1. C'est une forme de **signature** numérique pour Alice.

1 Alice écrit à Bob, c'est l'exemple couramment utilisé dans tous les tutoriels. C'est plus convivial qu'un austère « A écrit à B ». C'est qu'on aime rigoler dans le monde de la cryptographie !

- Alice chiffre le message MC_1 avec la clef publique de Bob. Résultat MC_2 (message chiffré 2). Ainsi **seul Bob sera capable de lire le message** puisqu'il faut la clef privée de Bob pour le déchiffrer.
- Alice envoie le message MC_2 à Bob.
- Bob reçoit le message chiffré MC_2, il le déchiffre avec sa clef privée, résultat MC_1.
- Bob déchiffre le message MC_1 avec la clef publique d'Alice, résultat : message M.

Remarque : Alice pourrait commencer par chiffrer avec la clef publique de bob et seulement ensuite signer numériquement ou même ne chiffrer que des parties du message. Le tout est une question de protocole.

A.3. Comment Alice communique-t-elle sa clef publique ?

Alice envoie sa clef publique à Bob dans un message non chiffré qu'elle signe numériquement (avec sa clef privée). La fonctionnalité est disponible dans Thunderbird (voir page 7). Bob peut déchiffrer la signature d'Alice avec la clef publique qu'il vient de recevoir. Si cette clef est corrompue, le déchiffrement ne fonctionnera pas.

A.4. Les attaques dites de *l'homme du milieu*.

C'est la seule vulnérabilité connue du chiffrement RSA. L'homme du milieu est un tiers malveillant que nous appellerons Xavier[2] dans ce document, qui dispose de sa propre paire de clefs personnelle. Il transmet sa clef publique à Alice en lui faisant croire que c'est celle de Bob et il transmet cette même clef publique à Bob en lui faisant croire que c'est la clef publique d'Alice. Xavier peut alors intercepter les communications entre Alice et Bob.
C'est facile à réaliser pour une organisation qui aurait la main sur les serveurs de mail d'une zone géographique.

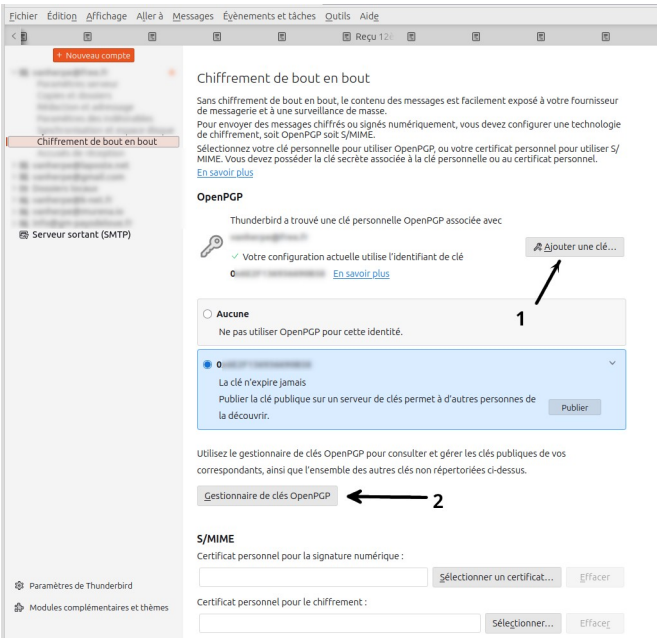
A.5. La parade à l'attaque de l'homme du milieu.

L'idée est qu'Alice et Bob s'échangent leurs clefs publiques **par un autre média** que par leur canal de communication, un média que Xavier ne maîtrise pas, par exemple une rencontre conviviale du genre barbecue saucisses-merguez chez Alice.
De fait, l'échange des clefs en tant que telles n'est pas très réaliste vu que les clefs en question sont codées en général sur 2048 bits voire plus, ce qui donne, même en base 16 des nombres à plus de 100 chiffres !
En réalité l'échange porte sur l'*empreinte* des clefs, une version de celles-ci à échelle humaine . L'algorithme RSA assure que l'empreinte d'une clef **identifie** celle-ci de manière quasi unique.

2 Rien à voir avec Xavier Niel, quoique ...

B. Le chiffrement RSA avec Thunderbird

Dans ses versions récentes, Thunderbird propose un outil spécifique accessible par : *Paramètres des comptes/Compte de messagerie concerné/Chiffrement de bout en bout.*[3]

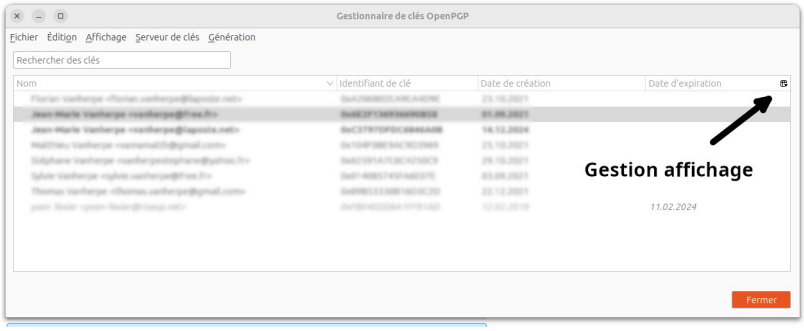


Il devient alors possible de :

1. Ajouter une clef au compte de messagerie
2. Accéder au gestionnaire de clefs OpenPGP

B.1. Le gestionnaire de clefs OpenPGP

Toute la gestion des clefs personnelles[4] se fait via cet outil.



Les lignes sont en gras lorsque le gestionnaire dispose d'une clef privée, les autres sont des clefs publiques d'autres identités.

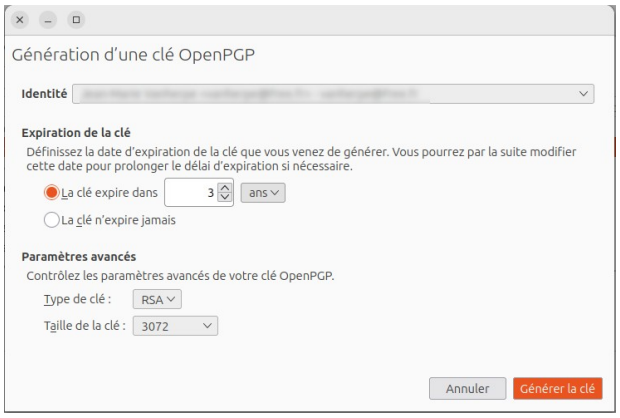
Le gestionnaire de clefs permet entre autres :

- La génération de paires de clefs associées à une adresse mail
- L'export de clefs publiques ou privées vers un fichier (une fois une identité sélectionnée) à partir de l'entrée de menu *Fichier* du gestionnaire.
- L'import de clefs depuis un fichier
  à partir de l'entrée de menu *Fichier* du gestionnaire.
- La vérification d'empreintes
- D'accorder un certain niveau de confiance aux clés publiques enregistrées

3 Nous n'évoquerons pas ici le chiffrement par certificat S/MIME
4 Clef personnelle : c'est le jargon utilisé par Thunderbird pour la paire de clefs publique,privée

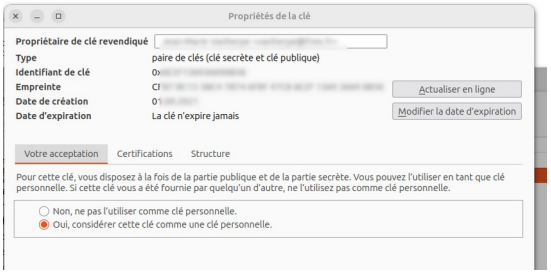
B.1.a. La génération d’une paire de clefs

Il suffit de remplir un formulaire et de patienter …
Observons une fois de plus que la paire de clefs est associée à une identité (adresse de messagerie).



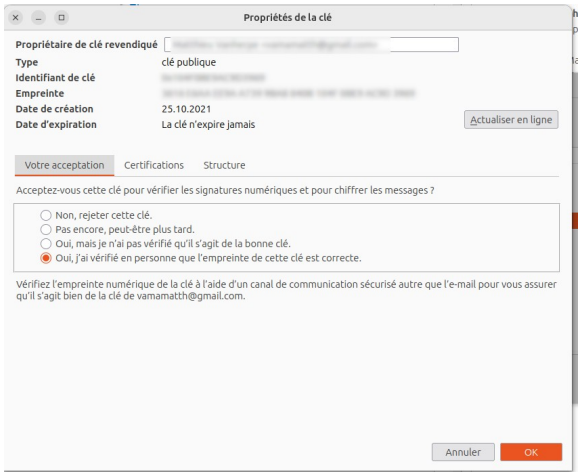
B.1.b. La vérification d’empreintes

Rappelons que la vérification d’empreintes de clef publique est la parade à l’attaque de l’homme du milieu. Elle peut se faire en consultant les propriétés de la clef :



Il suffit alors d’en contrôler la cohérence avec l’empreinte obtenue **via un autre média**.

B.1.c. Accorder un certain niveau de confiance aux clés publiques enregistrées

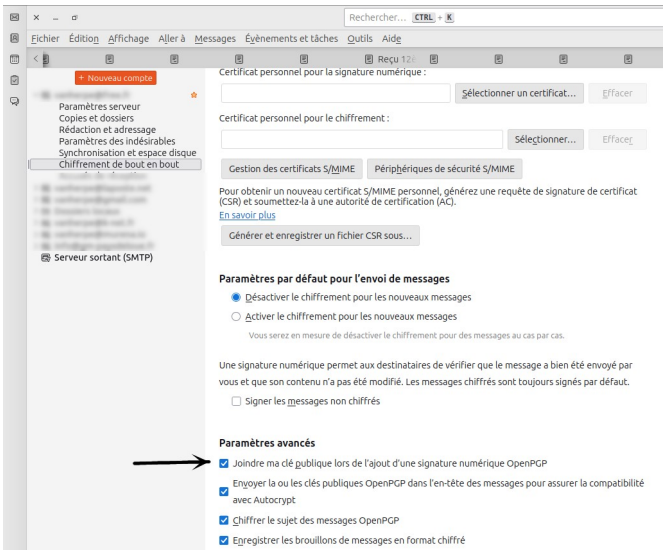


Voir le paragraphe B.3.b page 8 pour savoir ce qu’il faut en penser.

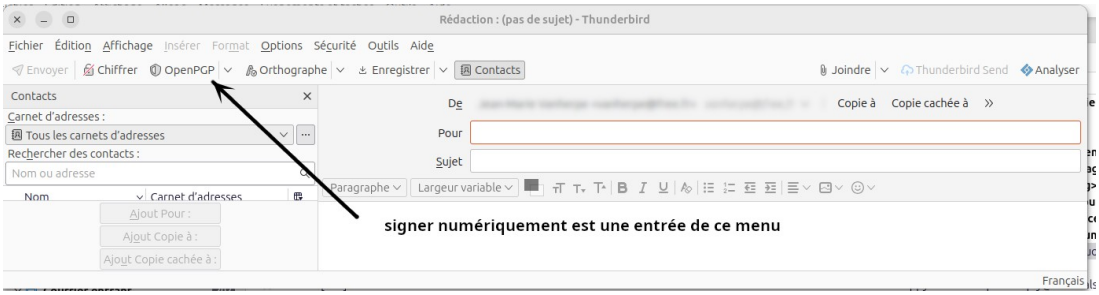
B.2. L’envoi de message

En matière de chiffrement l’expéditeur d’un message a plusieurs choix :

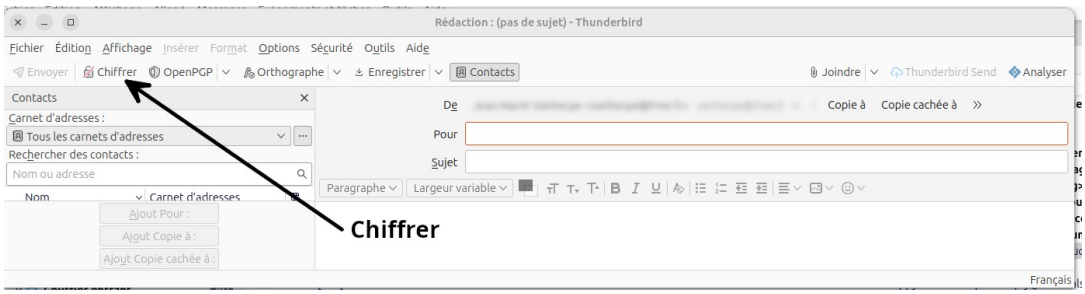
1. **Signer numériquement** son message. Le message en lui-même n’est pas chiffré, il est juste signé avec la clef privée de l’expéditeur. On peut y joindre aussi la clef publique de l’expéditeur.
   Cela suppose d’avoir au préalable correctement paramétré le chiffrement de bout en bout.



   C’est la méthode utilisée pour fournir sa clef publique à ses correspondants.



2. **Chiffrer le message** : le message est à la fois **signé** avec la clef privée de l’expéditeur **et chiffré** avec la clef publique du destinataire. Évidemment, il est nécessaire de disposer des clefs publiques de ces destinataires



3. **Ne rien faire**, le message n’est ni signé ni chiffré

B.3. Considérations annexes

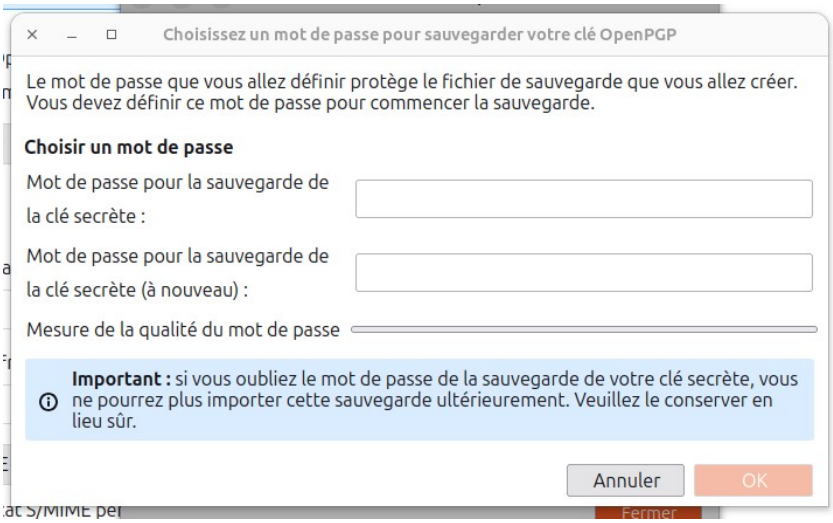
B.3.a. Utiliser la même paire de clefs sur différents périphériques

La vie moderne impose souvent de communiquer à l'aide de multiples instruments : PC fixe, PC portable, voire téléphone mobile. Il s'agit ici de pouvoir utiliser la même paire de clefs sur plusieurs périphériques.

La méthode est d'utiliser l'export puis l'import des clefs secrètes et publiques vers des fichiers.

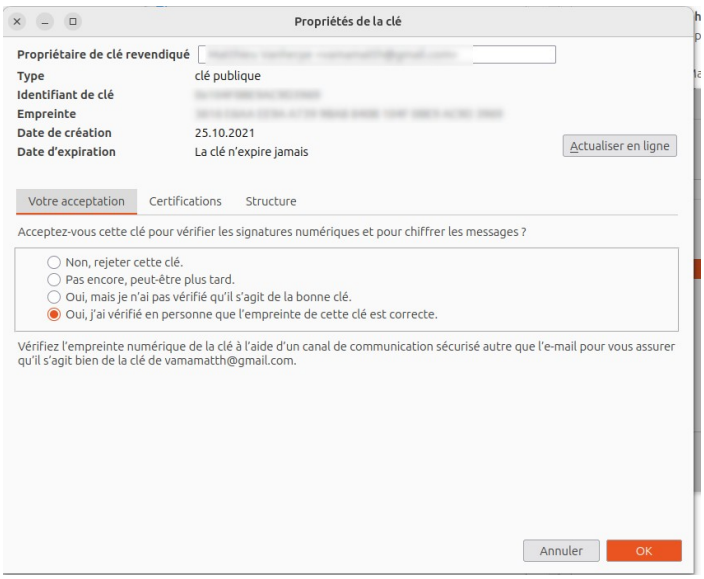
L'export, puis l'import se font par le menu *Fichier* du gestionnaire de clefs.

L'export de la clef privée vers un fichier (menu *Fichier/Sauvegarder …*) requiert un mot de passe



B.3.b. Sur le niveau de confiance des clefs publiques stockées.

Lorsque Thunderbird reçoit une clef publique, comme jointe à un courrier par exemple, il l'enregistre dans le gestionnaire de clefs avec un certain niveau de confiance qui est ensuite modifiable.



Attention : Le niveau de confiance ne dispense pas de la vérification d'empreinte.

Le niveau de confiance indique seulement à quel point vous faites confiance à une personne pour certifier l'identité d'autres personnes. Il ne garantit pas que la clef publique appartient réellement à la bonne personne si vous ne l'avez jamais vérifiée vous-même.

Le niveau de confiance devient utile après la vérifications initiale, notamment dans un réseau de confiance où Alice vérifie Bob, Bob vérifie Charlie, et vous faites confiance à Alice pour certifier l'identité de Charlie.
En résumé : La vérification de l'empreinte est l'étape critique qui établit l'authenticité initiale. Le niveau de confiance ne fait qu'étendre cette sécurité dans un réseau.

B.3.c. Sur les serveurs de clés

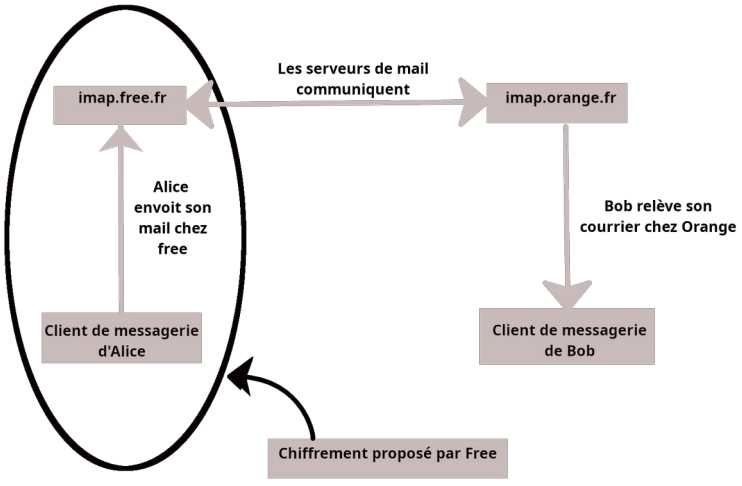
L'idée est de déposer sa clef publique sur un serveur de clef pour la rendre vraiment publique. Cependant ce type de serveurs a pu être corrompu dans le passé (https://www.01net.com/actualites/un-reseau-mondial-de-serveurs-de-cles-pgp-victime-d-un-sabotage-irremediable-1725460.html) laissant ouvertes des attaques de type homme du milieu. La méthode d'échange d'empreintes autour d'un BBQ convivial paraît nettement meilleure.

B.3.d. Sur l'expression « chiffrement de bout en bout ».

Prenons toujours l'exemple d'Alice qui veut communiquer de manière chiffrée avec Bob. Imaginons que l'adresse mail d'Alice est alice@free.fr et que celle de bob est bob@orange.fr. Les logiciels/serveurs impliqués dans l'opération sont :

- le client de messagerie d'Alice
- le serveur de mail de chez free (imap.free.fr)
- le serveur de mail d'orange (imap.orange.fr)
- le client de messagerie de Bob.

Il y a quelques années *free* proposait le chiffrement des courriers électroniques. En fait Xavier[5] ne proposait **que** le chiffrement de la connexion entre le client de messagerie (d'Alice) et son serveur imap de messagerie (imap.free.fr).



On n'est pas très loin de l'attaque de l'homme du milieu, Xavier profitait de la difficulté du sujet et de l'ignorance des gens pour les induire en erreur. Les utilisateurs avec un mail de la forme @free.fr qui se fient à ce chiffrement pouvaient penser que leurs courriers étaient totalement chiffrés alors que le cela s'arrêtait au serveur mail de chez free. L'expression *chiffrement de bout en bout* prend ici tout son sens.

5 Xavier Niel pour le coup.