Coopération entre Kmail et GnuPG

Quand le dialogue ne passe plus, sous Kubuntu

INDEXATION DU DOCUMENT			
	TITRE : Coopération entre Kmail et GnuPG		
ACTION	*NOM*	*DATE*	*SIGNATURE*
RÉDIGÉ PAR	Asdrad TORRES	6/12/2009	

SUIVI DU DOCUMENT			
INDICE	DATE	MODIFICATIONS	NOM

Table des matières

Résumé

Réparer les facéties d'Unbuntu et GnuPG pour pouvoir, dans Kmail, signer et crypter ses messages avec ses clés PGP.

Quand le dialogue ne passe plus, sous Kubuntu

1 Problème lié à KDE 4.2.4

Lorsqu'on veut chiffrer un message, Kmail indique le mot passe (passphrase) protégeant la clé que l'on utilise n'est pas correcte, alors qu'il ne nous a même pas proposé de taper ce mot de passe. On récolte un message du style "Signing failed: Bad passphrase" / "Échec de la signature : Mauvaise phrase de passe". C'est un bug de la distribution Ubuntu.

La solution donnée pour la KDE 4.1 fonctionne également pour KDE 4.2.4 :

1. créer un fichier ~/.kde/env/start-gpg-agent.sh contenant :

```
#!/bin/bash
eval "$(/usr/bin/gpg-agent --daemon --default-cache-ttl 1200)"
```

2. créer .kde/shutdown/gpg-agent.sh contenant :

```
#!/bin/bash
[ -n "${GPG_AGENT_INFO}" ] &&
    kill $(echo "${GPG_AGENT_INFO}" | cut -d ':' -f 2)
```

3. donner à ces fichiers les droits d'exécution (chmod u+x nomfichier)
4. relancer la session KDE 4.2

2 Problème lié à GnuPG

Le changement de version de Kmail lié au passage de KDE 3 à KDE 4 introduit un contrôle plus rigoureux de la validité des clés utilisées pour chiffrer ou simplement signer ses messages. Deux symptômes majeurs apparaissent :

– lorsqu'on envoie un fichier, même non signé et non chiffré, Kmail indique un problème de validité sur les clés,
– lorsqu'on on veut vérifier et "mettre à jour" les clés utilisées pour l'identité qui pose problème, on peut sélectionner la clé pour signer mais pas pour chiffrer.

Kmail n'est pas en cause. La raison vient d'un dysfonctionnement dans l'auto-validation de nos clés. On le vérifie dans un terminal :

```
moi@local:~$ gpg --edit-key Moi
gpg (GnuPG) 1.4.9; Copyright (C) 2008 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

La clé secrète est disponible.

pub  1024D/131321G4  créé: 2009-01-05  expire: jamais       utilisation: SCA
                     confiance: ultime        validité: inconnue
...
```

Note
On peut également vérifier avec un outil graphique de gestion des clés mais l'information fournie est généralement plus confuse qu'ici.

La validité de la clé est marquée "inconnue" et le restera même si l'on a, évidemment, auto-signé sa clé et que, comme on le voit dans l'exemple, on se fait à soit-même une confiance "ultime". En toute logique, une auto-signature avec une clé à confiance ultime devrait donner une clé de validité "ultime" (absolue).

Le seul moyen réellement efficace de corriger le problème est d'indiquer à GnuPG, par un autre moyen, que notre clé est valide. On obtient ce résultant en :

– éditant le fichier ~/.gnupg/trustlist.txt (avec l'éditeur de texte de votre choix)

– et ajoutant l'empreinte à 16 caractères hexadécimaux de notre clé, à la fin du fichier

Le fichier ressemble alors à ça :

```
....
# or '*'. Additional data, delimited by white space, is ignored.
#
# NOTE: You should give the gpg-agent a HUP after editing this file.

# Ajout pour compatibilite descendante
# (bootstrap du reseau de confiance)
# A.T. 2009-12-06
46B7987D987E987F
```

Si on ne connaît pas l'empreinte demandée (qui n'est pas l'identifiant), on pourra la récupérera avec la commande :

```
moi@local:~$ gpg --list-keys --with-colons 0x131321G4
```

en remplaçant le dernier paramètre par l'identifiant (obtenu par la commande précédente) de la clé, préfixé par "0x" (le "0" est un zéro). Dans le résultat affiché, on s'intéressera à la ligne contenant la clé que l'on veut déclarer valide :

```
pub:u:1024:17:46B7987D987E987F:2009-01-05:::u:Moi <moi@monmail.org>::scaESCA:
```

Note
On peut également retrouver l'empreinte avec un outil graphique de gestion des clés.

Il ne reste qu'à fermer puis rouvrir sa session KDE[1].

Une petite vérification pour s'assurer que GnuPG reconnaît enfin notre clé comme valide :

```
moi@local:~$ gpg --edit-key Moi
...

pub  1024D/131321G4  créé: 2009-01-05  expire: jamais       utilisation: SCA
                     confiance: ultime        validité: ultime
...
```

YES !!! La validité est `ultime`.

On peut alors retourner dans Kmail et configurer les clés associées aux identités, comme on s'attendait à pouvoir le faire.

A Quelques sources d'info

Le manuel de GnuPG

La FAQ de GnuPG

[1] Il existe certainement une manière plus élégante de "give the gpg-agent a HUP", mais je ne connais pas.