

Lightning Talk

OpenPGP & GPG

LouviLUG
28 octobre 2010

OpenPGP & GPG

- À quoi ça sert ?
 - Chiffrement
 - Signature
- Comment ça marche ?
 - Clé publique et clé privée
 - Web of Trust
- En pratique...
 - Création d'une clé
 - Utilisation dans Évolution (chiffrement & signature)

OpenPGP: À quoi ça sert ?

- À chiffrer des informations
 - Documents, mots de passe, e-mails, ...
 - De plus en plus important de nos jours
- À signer des messages
 - Pour prouver l'identité de l'expéditeur et la validité du message
 - Fonctionnalité clé pour une série d'applications
 - identité électronique

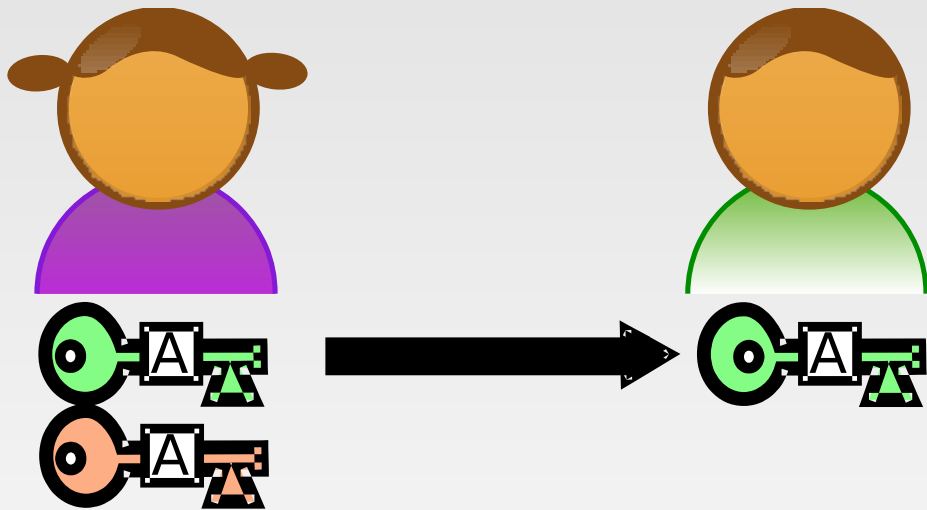
OpenPGP: Comment ça marche ?

- Cryptographie à clé publique (ou asymétrique)
 - Chaque utilisateur génère 2 clés
 - **Clé publique** : tout le monde peut la voir, elle n'a rien de secret
 - **Clé privée** : secrète.
Indispensable pour toutes les opérations « utiles » avec GPG

OpenPGP: Comment ça marche ?

- Propriétés principales des clés asymétriques
- Si on **chiffre** avec la clé publique, on peut **déchiffrer** avec la clé privée correspondante.
- Si on **chiffre** avec la clé privée, on peut **déchiffrer** avec la clé publique.

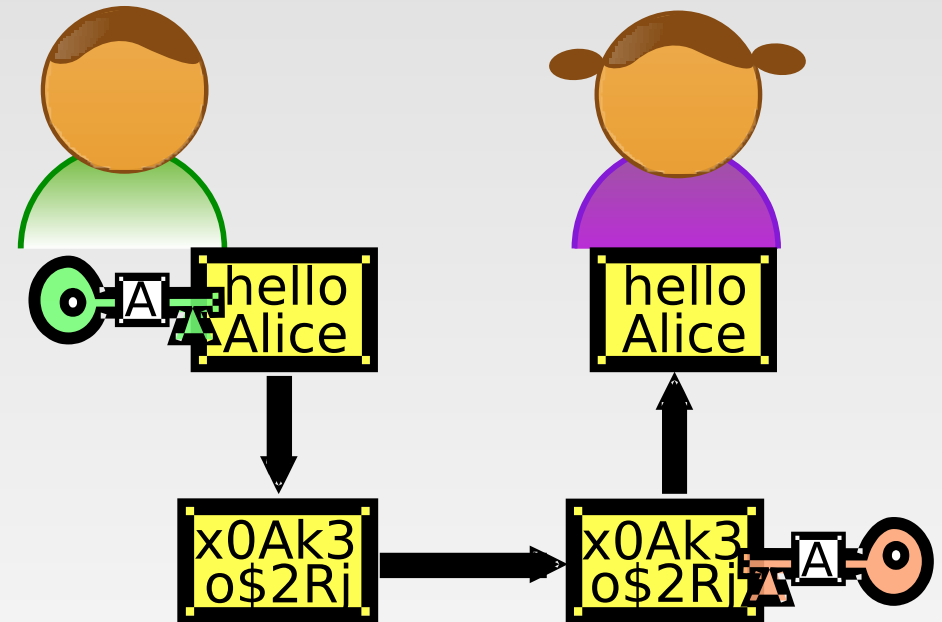
Chiffrement



- Alice a une clé publique (verte) qu'elle envoie à Bob
- Elle a aussi une clé privée qu'elle garde secrète (rouge)

Chiffrement

- Bob **chiffre** un message en utilisant la **clé publique de Alice**
- Alice est la seule à pouvoir **déchiffrer** le message grâce à **sa clé privée**



Signature

- Bob **calcule** un **hash du message** et le **chiffre** avec sa clé privée
- Alice **déchiffre** le hash avec la clé publique de Bob et **calcule le hash** du message puis le compare avec celui déchiffré.
- Si c'est bien Bob qui a chiffré le hash, celui qu'Alice a calculé sera bien le même que celui qu'elle a déchiffré
- Alice est **assurée que le message vient de Bob.**
Ou pas ?

Web of Trust

- Pas de preuve de l'identité réelle :(
- Solution: chaque utilisateur peut affirmer sa confiance en l'identité de quelqu'un en **signant sa clé**
- Un réseau de confiance se crée...
 - Système décentralisé

OpenPGP: En pratique...

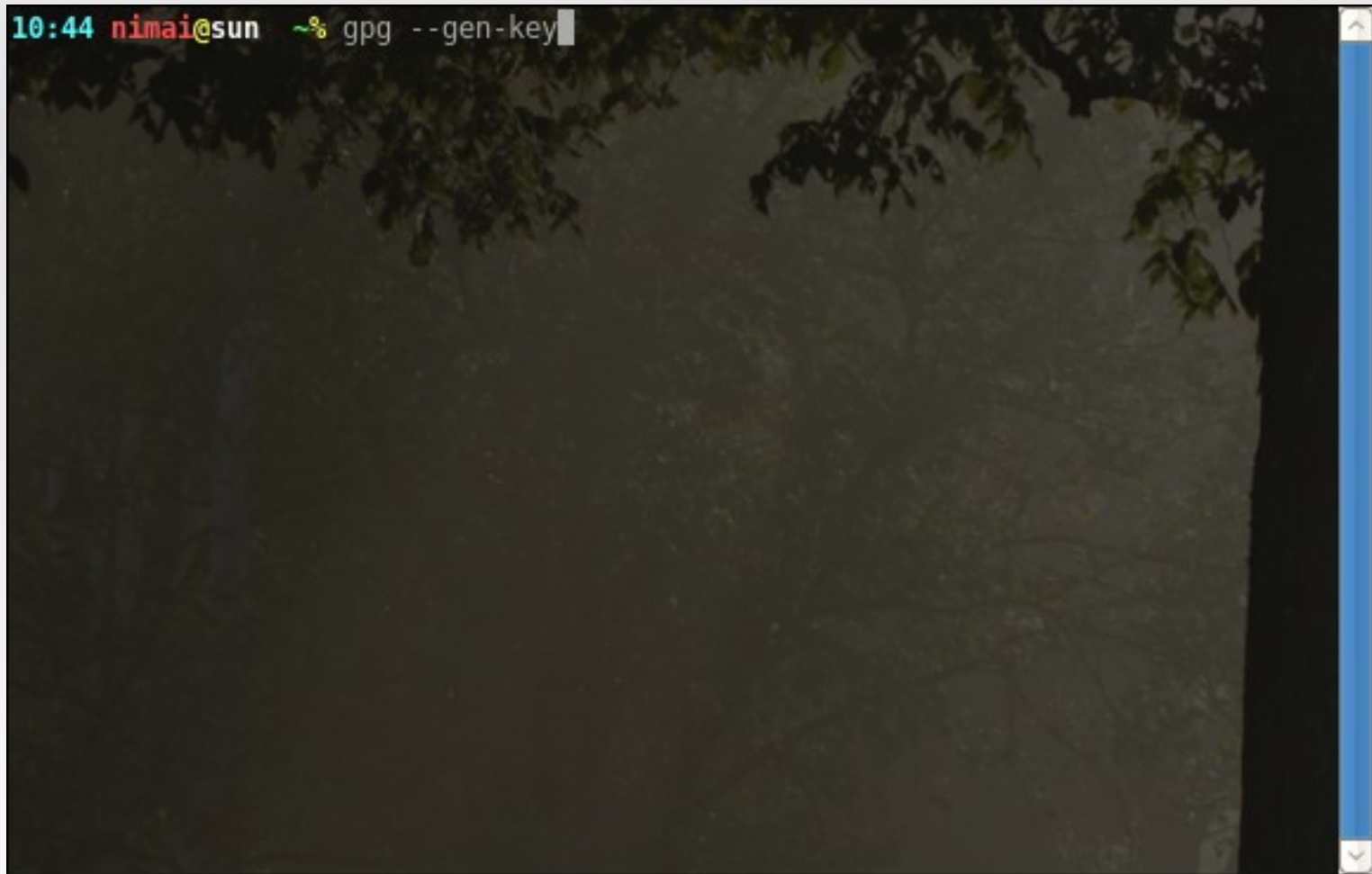
En pratique, on utilise GnuPG
parce que...

C'est libre !

Démonstration

OpenPGP: En pratique...

Générer une paire de clés



```
10:44 nimai@sun ~% gpg --gen-key
```

The image shows a terminal window with a dark background and a blue title bar. The prompt shows the time as 10:44, the user as nimai@sun, and the current directory as ~%. The command gpg --gen-key is entered at the prompt. The terminal content is partially obscured by a dark, blurry image of a tree trunk and leaves.

OpenPGP: En pratique...

Générer une paire de clés

```
10:44 nimai@sun ~% gpg --gen-key
gpg (GnuPG) 1.4.10; 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.

Please select what kind of key you want:
  (1) RSA and RSA (default)
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (2048)
Requested keysize is 2048 bits
Please specify how long the key should be valid.
  0 = key does not expire
  <n> = key expires in n days
  <n>w = key expires in n weeks
  <n>m = key expires in n months
  <n>y = key expires in n years
Key is valid for? (0) █
```

OpenPGP: En pratique...

Générer une paire de clés

```
What keysize do you want? (2048)
Requested keysize is 2048 bits
Please specify how long the key should be valid.
  0 = key does not expire
  <n> = key expires in n days
  <n>w = key expires in n weeks
  <n>m = key expires in n months
  <n>y = key expires in n years
Key is valid for? (0) 2
Key expires at Sat 30 Oct 2010 10:50:10 AM CEST
Is this correct? (y/N) y

You need a user ID to identify your key; the software constructs the user ID
from the Real Name, Comment and Email Address in this form:
  "Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Real name: Truc Machin
Email address: lala@louvilug.be
Comment: Clé de démonstration
You are using the `utf-8' character set.
You selected this USER-ID:
  "Truc Machin (Clé de démonstration) <lala@louvilug.be>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? 0
```

OpenPGP: En pratique...

Générer une paire de clés

```
    0 = key does not expire
    <n> = key expires in n days
    <n>w = key expires in n weeks
    <n>m = key expires in n months
    <n>y = key expires in n years
Key is valid for? (0) 2
Key expires at Sat 30 Oct 2010 10:50:10 AM CEST
Is this correct? (y/N) y

You need a user ID to identify your key; the software constructs the user ID
from the Real Name, Comment and Email Address in this form:
    "Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Real name: Truc Machin
Email address: lala@louvilug.be
Comment: Clé de démonstration
You are using the `utf-8' character set.
You selected this USER-ID:
    "Truc Machin (Clé de démonstration) <lala@louvilug.be>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? 0
You need a Passphrase to protect your secret key.

Enter passphrase: █
```

OpenPGP: En pratique...

Générer une paire de clés

```
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
.....+++++
..+++++
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
.....+++++
...+++++
gpg: /home/nimai/.gnupg/trustdb.gpg: trustdb created
gpg: key 34751C4D marked as ultimately trusted
public and secret key created and signed.

gpg: checking the trustdb
gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: next trustdb check due at 2010-10-30
pub 2048R/34751C4D 2010-10-28 [expires: 2010-10-30]
    Key fingerprint = 98B2 853B FB99 8118 EB90 1815 91A3 4DC4 3475 1C4D
uid                               Truc Machin (Clé de démonstration) <lala@louvilug.be>
sub 2048R/0228778C 2010-10-28 [expires: 2010-10-30]

11:01 nimai@sun ~% █
```


OpenPGP: En pratique...

Chiffrement d'un mail

The image shows a screenshot of the Evolution mail client interface. The main window displays the 'Evolution Preferences' dialog, specifically the 'Account Editor' for the account 'valombre@gmail.com'. The 'Security' tab is selected, showing options for signing and encrypting outgoing messages. The 'PGP/GPG Key ID' field is filled with '34751C4D'. Below this, there are checkboxes for 'Always sign outgoing messages when using this account', 'Always encrypt to myself when sending encrypted messages', and 'Always trust keys in my keyring when encrypting'. There are also fields for 'Signing certificate' and 'Encryption certificate', each with a 'Select...' button and a 'Clear' button.

In the background, a terminal window is open, showing the output of the command `gpg --list-public-keys /home/nimai/.gnupg/pubring.gpg`. The output lists two public keys:

```
pub 2048R/34751C4D 2010-10-28 [expires: 2010-10-30]
uid Truc Machin (Clé de démonstration) <lala@louvilug.be>
sub 2048R/0228778C 2010-10-28 [expires: 2010-10-30]
```

The terminal prompt is `nimai@sun: ~`. The terminal window is titled `nimai@sun: ~`.

The Evolution Preferences dialog also shows a list of accounts in the background:

Enabled	Account Name	Protocol
<input checked="" type="checkbox"/>	valombre@gmail.com	Default imap

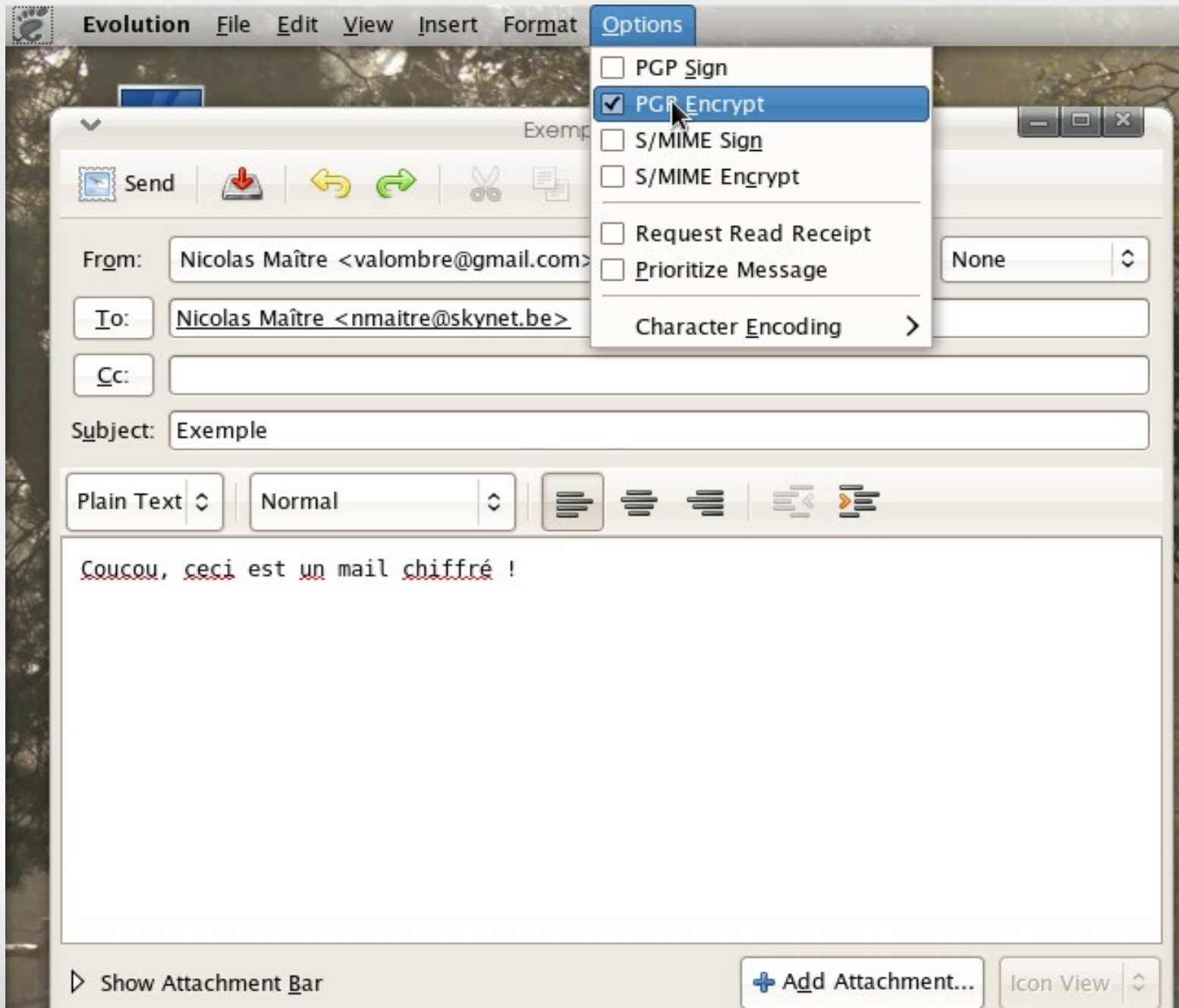
OpenPGP: En pratique...

Chiffrement d'un mail

```
11:22 nimai@sun ~% gpg --search-keys nmaitre@skynet.be
gpg: searching for "nmaitre@skynet.be" from hkp server pgp.mit.edu
(1)   Nicolas Maitre <valombre@gmail.com>
      Nicolas Maitre <nmaitre@skynet.be>
      Nicolas Maitre (nimai's key) <nimai@skynet.be>
      1024 bit DSA key 31F5C05E, created: 2007-04-29
Keys 1-1 of 1 for "nmaitre@skynet.be".  Enter number(s), N)ext, or Q)uit > 1
gpg: requesting key 31F5C05E from hkp server pgp.mit.edu
gpg: key 31F5C05E: public key "Nicolas Maitre (nimai's key) <nimai@skynet.be>" i
mported
gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: next trustdb check due at 2010-10-30
gpg: Total number processed: 1
gpg:         imported: 1
11:22 nimai@sun ~% █
```

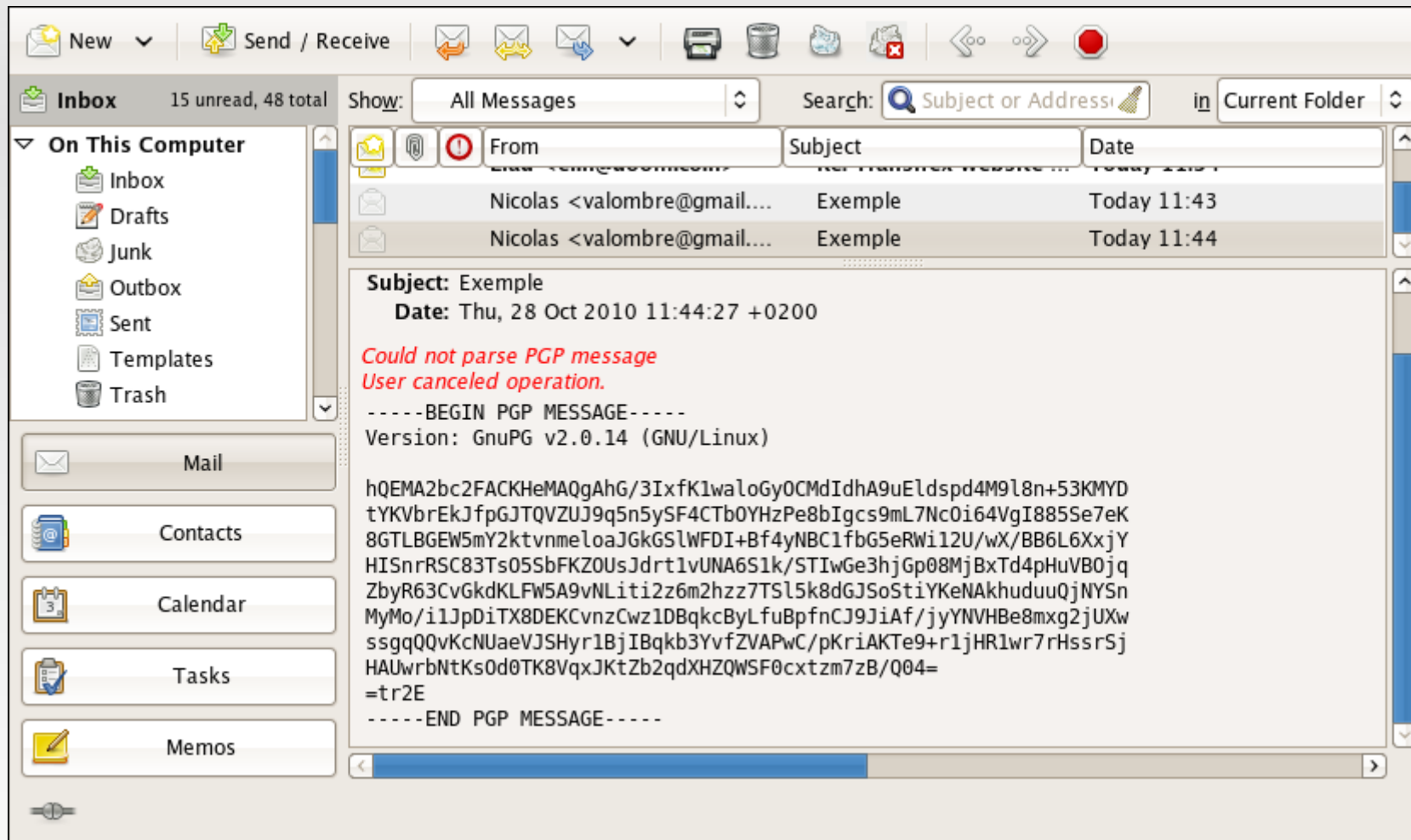
OpenPGP: En pratique...

Chiffrement d'un mail



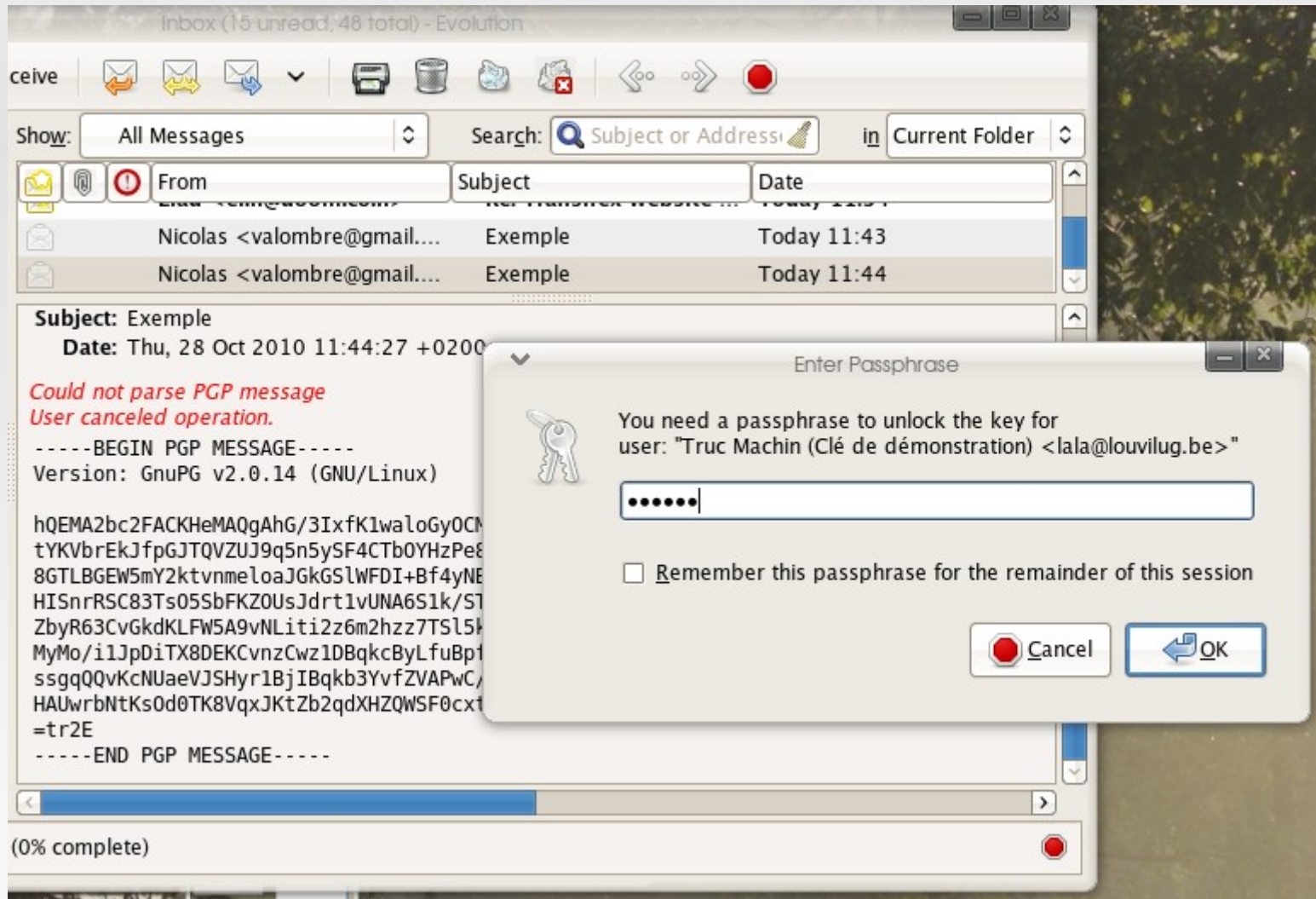
OpenPGP: En pratique...

Chiffrement d'un mail



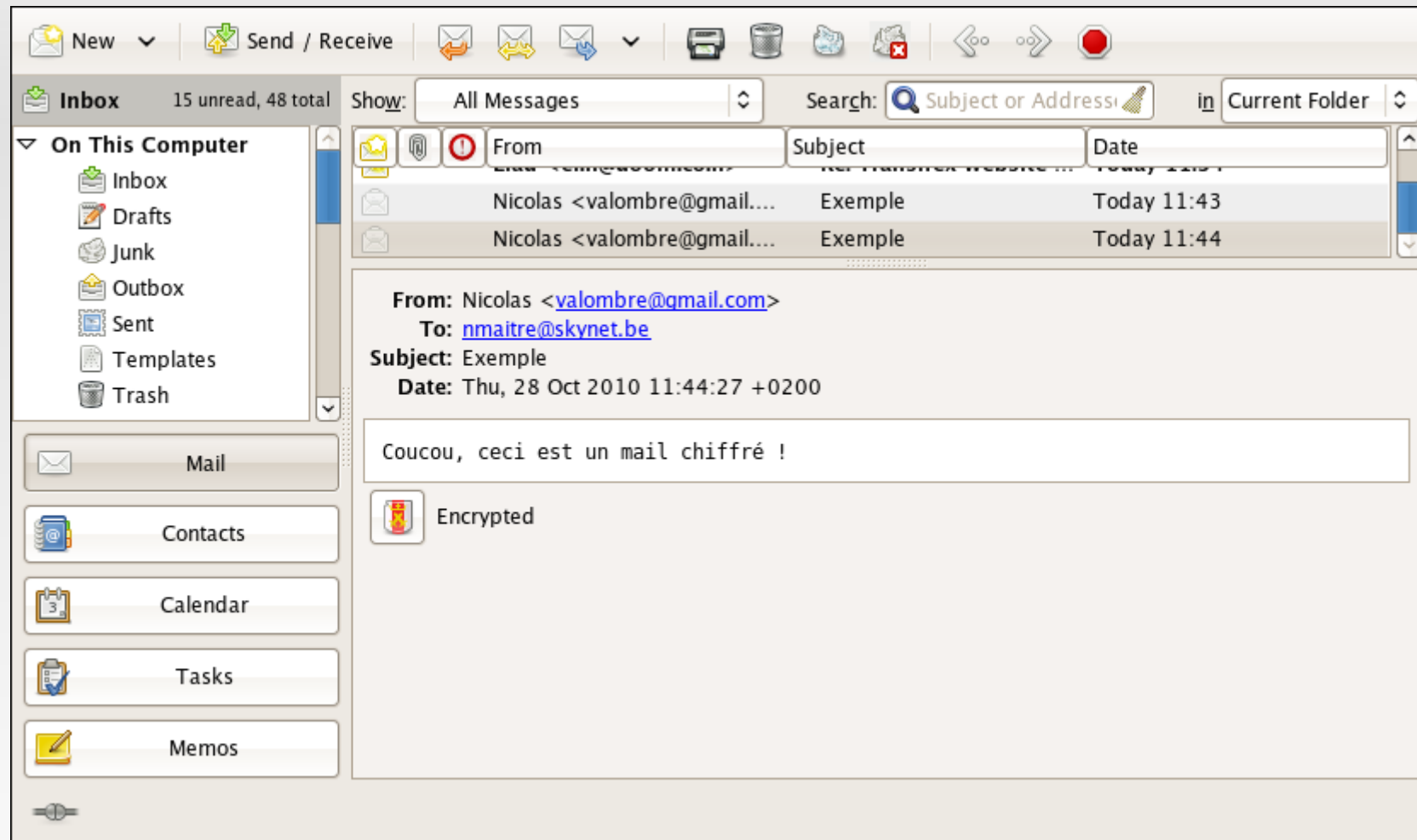
OpenPGP: En pratique...

Chiffrement d'un mail



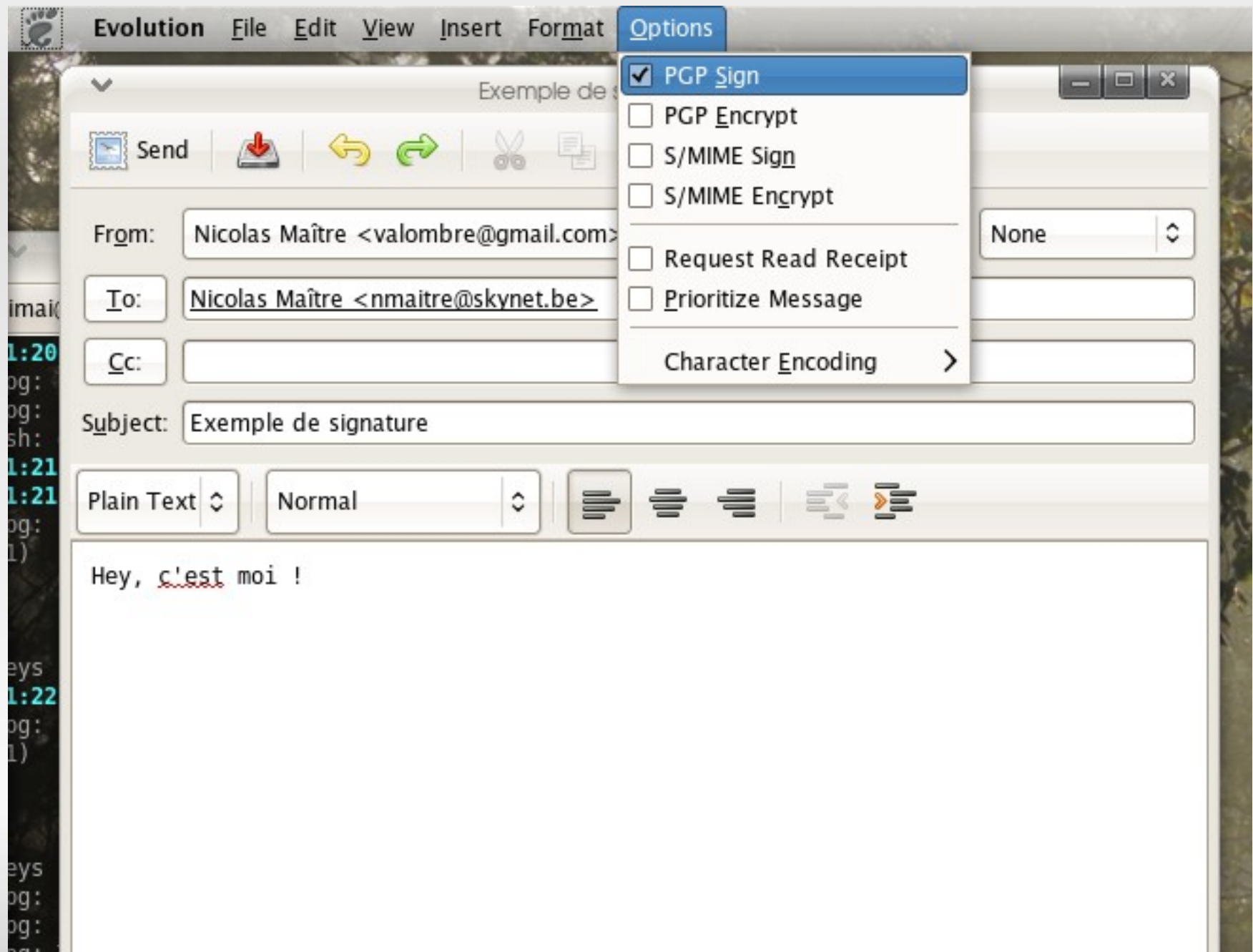
OpenPGP: En pratique...

Chiffrement d'un mail



OpenPGP: En pratique...

Signature d'un mail



OpenPGP: En pratique...

Signature d'un mail

