GPG signing key

packages, apt, aptly

Afin de pouvoir signer nos dépôts APT locaux lors de leur publication, nous allons avoir besoin d'une clé GPG dédiée à cette opération.

Création de la clé GPG

Nous créons la clé

```
gpg --full-generate-key
```

Un certain nombre de question nous sont posées

```
gpg (GnuPG) 2.2.12; Copyright (C) 2018 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? 4
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 4096
Requested keysize is 4096 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) 2y
Key expires at Wed 15 Feb 2023 07:31:51 PM CET
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.

Real name: PHM APT repository
Email address: aptrepo@grohub.org
Comment:
You selected this USER-ID:
    "Grohub APT repository <aptrepo@grohub.org>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? o
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: key 6EA07D11EEA802A8 marked as ultimately trusted
gpg: directory '/var/cache/aptly/.gnupg/openpgp-revocs.d' created
gpg: revocation certificate stored as '/var/cache/aptly/.gnupg/openpgp-
revocs.d/768518BF91E8BEF5E991155F6EA07D11EEA802A8.rev'
public and secret key created and signed.

Note that this key cannot be used for encryption.  You may want to use
the command "--edit-key" to generate a subkey for this purpose.
pub   rsa4096 2021-02-15 [SC] [expires: 2023-02-15]
      768518BF91E8BEF5E991155F6EA07D11EEA802A8
uid                      Grohub APT repository <aptrepo@grohub.org>
```

Le certificat de révocation est à mettre en lieu sûr, dans le cas où la clé serait compromise.

Export / import des clés

Comme Aptly utilise openGPG v1, et que nous avons créé nos clés avec openGPG v2, nous allons devoir exporter et importer celles-ci dans le bon trousseau.

Pour lister les clés du trousseau de la v2

```
gpg --list-keys
gpg: checking the trustdb
gpg: marginals needed: 3  completes needed: 1  trust model: pgp
gpg: depth: 0  valid:   1  signed:   0  trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: next trustdb check due at 2023-02-15
/var/cache/aptly/.gnupg/pubring.gpg
---------------------------------
pub   rsa4096 2021-02-15 [SC] [expires: 2023-02-15]
      768518BF91E8BEF5E991155F6EA07D11EEA802A8
uid           [ultimate] Grohub APT repository <aptrepo@grohub.org>
```

```
gpg --export --armor EEA802A8 > EEA802A8.key
gpg --export-secret-key --armor EEA802A8 > EEA802A8.sec
```

Nous importons notre clé privée dans openGPG v1

```
gpg1 --import < EEA802A8.sec
```

Liens

- manuel GPG
- aptly publish: gpg: no default secret key: secret key not available

- ERROR: unable to initialize GPG signer: looks like there are no keys in gpg, please create one