GNU Privacy Guard - GnuPG/GPG

Guida operativa ad un software libero per la crittografia

Matteo Mardegan

MontelLUG - Montebelluna Linux User Group

25 marzo 2016 – Montebelluna

Licenza d'utilizzo

Sommario

1. Cos'è GnuPG
   - Informazioni su `GnuPG`
   - Come funziona
   - Compatibilitá e frontend disponibili
   - Versioni e algoritmi disponibili
2. Breve cenno alla crittografia asimmetrica
3. Installazione ed uso di GnuPG
   - Comandi di GnuPG
4. Il Key Signing Party
5. Biografia

Cos'è GnuPG - Informazioni su GnuPG

GnuPG permette di **cifrare e firmare** dati e informazioni e quindi puó esser utile per:

- mantenere riservati dati ed informazioni scambiati tra due soggetti
- assicurare ad un destinatario che determinati dati provengono da un determinato mittente
- archiviare in modo i sicuro i propri file cifrati in supporti non sicuri (penne usb, hard disk esterni) o archivi cloud non sicuri (Google Drive, Gmail ed equivalenti)
- verificare l'autenticità di pacchetti scaricati da internet

GnuPG è un'implementazione completa e libera dello standard OpenPGP definito dalla **RFC4880** noto anche come PGP.
È un software libero nel senso che rispetta la libertá dell'utente essendo, distribuito sotto i termini della GNU General Public License: puó essere pertanto liberamente usato, modificato e distribuito.

Cos'è GnuPG - Come funziona

GnuPG è uno strumento a riga di comando che si puó integrare con numerose altre applicazioni.
GPG cifra messaggi o file utilizzando una coppia di chiavi (pubblica e privata) generate dall'utente[1].
Le chiavi pubbliche possono essere scambiate tra gli utenti in vari modi, principalmente email e keyserver, cioè dei server pubblici che raccolgono e distribuiscono chiavi pubbliche.
La versione 2 di GnuPG fornisce anche il supporto per S/MIME e Secure Shell (SSH).

[1] Attenzione che se si utilizza GnuPG per cifrare le proprie mail vengono *nascosti* solo i dati contenuti nel corpo del testo; rimangono in chiaro mittente, destinatario, l'oggetto e molte altre informazioni (altrimenti non vi sarebbe modo di far arrivare a destinazione la mail)

Cos'è GnuPG - compatibilitá e frontend disponibili

GNU Privacy Guard è un software stabile e maturo ed è disponibile per i seguenti sistemi operativi:

- GNU/Linux, FreeBSD, OpenBSD e NetBSD
- Microsoft Windows[2]
- OS X[3]
- Android[4]

Per Linux sono disponibili numerose applicazioni frontend, ad esempio:

- Seahorse per GNOME
- KGP, Kleopatra per KDE

ed inoltre è integrato in molti programmi di posta elettronica (es. Evolution, Mutt, K-9 Mail di Android; in Thunderbird/Icedove con il plug-in Enigmail)

[2] con il programma Gpg4win https://www.gpg4win.org/

[3] Mac GPG

[4] OpenKeyChan, distribuito su F-Droid

Cos'è GnuPG - Versioni e algoritmi disponibili

GnuPG è attualmente disponibile in tre versioni:

- **stabile** 2.0.29 è la versione suggerita per la maggior parte degli utenti
- **ultima** 2.1.11 con il supporto per ECC e molte altre nuove funzionalitá
- **classica** l'ultima è la versione 1.4.20

Installando il programma si possono trovare quindi `gnupg` e `gnupg2`

Algoritmi disponibili in GnuPG

GPG non fa utilizzo di algoritmi brevettati. Sono invece disponibili i seguenti algoritmi:

DSA, RSA, ElGamal, CAST5, Triple DES (3DES), AES e Blowfish

Breve cenno alla crittografia asimmetrica

Come detto GnuPG cifra i messaggi utilizzando una coppia di chiavi (pubblica e privata) generate dall'utente.
Questo metodo di crittografia viene definita *asimmetrica*, conosciuta anche come crittografia *a coppia di chiavi* o *a chiave pubblica/privata* o anche solo crittografia a chiave pubblica ed è un tipo di crittografia dove ad ogni soggetto coinvolto nella comunicazione è associata una (indovina???) coppia di chiavi informatiche[5], ovviamente:

la chiave pubblica che deve essere distribuita

la chiave privata che deve rimanere segreta

[5] che altro non è che un set piú o meno lungo di caratteri e numeri

Esempio di una chiave GnuPG

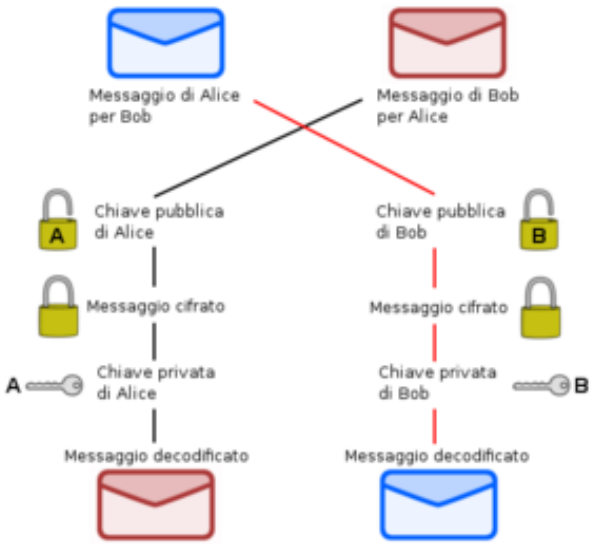
Segue un estratto di una chiave GnuPG:

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v2

mQINBE1QZ78BEACdoNoYP9uIvFBJJ4Tg5Vn2
gSFRDuKbj0qkifBdGT8co5Lfdgg2UnCximfJ
Uxkul82RBu75AJ/VU6vgDvgnP4m5vhY+f/vV
lscMeZbKzlJuW9YAIHqeCK2W1zEVfx4OqxdG
pFWiE4q0M74FDRkYMSQzb3FQWpVI2enmJCvG
ck0H16s0GflcNeQVeZ/fiUzQ65W0In7NWwc/
9YLH+fL7D+axL/Pd5cX6B0dy7mvVzBOPA0LL
myJg16Bq6J0rvtQ1TxEChQYYkV0mPXcpC5xB

-----END PGP PUBLIC KEY BLOCK-----
```

Breve cenno alla crittografia asimmetrica

Come vanno usate queste chiavi? Senza entrare nei tecnicismi:

La chiave pubblica deve esser usata per cifrare i dati destinati al proprietario della stessa, in quanto solo esso ha a disposizione la chiave privata per decriptarli. **La chiave privata deve esser usata per firmare** i dati destinati ai terzi in possesso della chiave pubblica, tramite la quale è possibile verificare l'autenticitá degli stessi.



Installazione di GnuPG

Non mi dilungo molto sul tema. Per l'installazione rinvio alla guida facilmente è possibile trovare per ciascuna distribuzione.
Per Debian ad esempio i passaggi saranno:

```
# apt-get install gnupg2
```

e poi ad esempio

```
# apt-get install kleopatra
```

con cui verranno installate tutte le dipendenze necessarie, quali ad esempio `gnupg-agent`, `pinetry-qt4` (o `pinetry-gtk`) e numerose altre.

Comandi di GnuPG

creare una chiave

digitare il comando

`gpg -gen-key`

Verranno chieste una serie di informazioni

- l'algoritmo: RSA ad esempio
- la sua lunghezza: 3072 ad esempio
- la scadenza: 2y ad esempio
- il nome e la mail
- la password da non dimenticare

creare una chiave di revoca

opzionale, ma caldamente suggerito, perchè non vi è altro modo per revocare una chiave

`gpg -output revoke-key.asc -gen-revoke your-id-key`

Il certificato va conservato in un luogo sicuro (salvato su disco, su carta, ecc.)

Comandi di GnuPG

esportare o importare una chiave

Per esportare la chiave pubblica

```
gpg -armor -output public.key -export <user-id>
```

Per far un backup della chiave privata

```
gpg -export-secret-keys -armor <user-id> >
privkey.asc
```

Per importare una chiave pubblica

```
gpg -import public.key
```

Per inviare la chiave pubblica al keyserver

```
gpg -send-keys <key-id> -keyserver
pool.sks-keyservers.net
```

Per scaricare una chiave dal keyserver

```
gpg -recv-keys <key-id> -keyserver
pool.sks-keyservers.net
```

Comandi di GnuPG

amministrare le chiavi

Per mostrate tutte le chiavi esistenti

```
gpg -list-keys
```

Per vedere le impronte digitali in uso ad esempio per il futuro keysigningparty

```
gpg -fingerprint
```

Per vedere la lista delle chiavi private

```
gpg -list-secret-keys
```

Per cancellare una chiave pubblica

```
gpg -delete-key <user-id>
```

Per cancellare una chiave privata

```
gpg -delete-secret-key
```

Per modificare alcune caratteristiche della chiave (ad esempio la data di scadenza o aggiungere un UID)

```
gpg -edit-key <user-id>
```

Comandi di GnuPG

cifrare un file

digitare il comando

```
gpg -encrypt -r
<user-id> -armor
secret.txt
```

È consigliabile firmare qualsiasi cosa si voglia cifrare. Inoltre per far si che l'informazione cifrata sia leggibile anche dal mittente, questo deve cifrare la chiave di sessione anche con la propria chiave pubblica

firmare un file

Per firmare dati con la propria chiave, si usa il comando

```
gpg -clearsign dati
```

verificare una firma

È possibile verificare le firme con il comando

```
gpg -verify dati
```

decifrare un file

digitare il comando

```
gpg -decrypt
secret.txt.asc
```

Key Signing Party - Creare una rete di fiducia

Il problema di tutti i sistemi di crittografia asimmetrica è la certificazione dell'autenticitá della chiave.
Questa puó avvenire:

- con la presenza di una Certification Authority che compie diverse operazioni di autenticazione e validazione del richiedente fino al rilascio di un certificato digitale: in tal caso si usa lo standard S/MIME
- con la presenza di una rete di fiducia (detto sistema web of trust) mediante il quale sono altri utenti che ne certificarno l'effettiva autenticitá di una chiave: in questo caso lo standard è PGP (citato all'inizio)

Key Signing Party - Creare una rete di fiducia

Un key signing party è una riunione di persone che usano il sistema di crittografia PGP, durante la quale ogni partecipante ha la possibilitá di firmare la chiave di altri partecipanti, certificandone l'identità e instaurando una rete di fiducia.
I key signing party aiutano a estendere la propria rete della fiducia
Affinchè non venga compromessa la rete di fiducia bisogna prestare particolare attenzione alla corrispondenza tra chiave e (presunta) identitá: la verifica avviene mediante un documento di riconoscimento valido.

Key Signing Party - Come funziona

In breve le cose da fare sono:

- avere o creare una propria chiave
- rendere disponibile la propria chiave pubblica su un keyserver ad esempio
- creare un elenco di partecipanti e dei dati delle chiavi (`gpg -fingerprint`)
- recarsi al party con un documento d'identitá valido
- verificare l'UID delle chiavi dei partecipanti mediante il loro documento d'identitá
- firmare ciascuna chiave pubblica verificata al party
- restituire la chiave pubblica firmata al partecipante o pubblicarla sul keyserver (previa autorizzazione espressa del proprietario)

Spazio alle domande

DOMANDE???

Letture consigliate per approfondire...

- Sito del progetto GnuPG: https//www.gnupg.org/
- Mini Howto in Italiano: https://www.gnupg.org/howtos/it/GPGMiniHowto.html
- Guida all'uso delle smartcard: https://www.gnupg.org/howtos/card-howto/en/smartcard-howto.html
- Guida operativa sul sito di ArchLinux: https://wiki.archlinux.org/index.php/GnuPG