Manuale GNU sulla privacy

Mike Ashley, versione originale inglese

Lorenzo Cappelletti, traduzione italiana

Manuale GNU sulla privacy

Mike Ashley, versione originale inglese

Lorenzo Cappelletti, traduzione italiana

Pubblicato $Date: 2002/03/17 10:45:19 $

Si prega di rivolgere domande, segnalazioni di errori o suggerimenti riguardanti questo manuale al manutentore Mike Ashley (<`jashley@acm.org`>) o al traduttore Lorenzo Cappelletti (<`L.Cappelletti@mail.com`>). Riferendosi al manuale, si prega di specificare di quale versione si è in possesso utilizzando, per il documento in lingua originale la stringa di revisione `$Revision: 1.11 $`, per quello presente tradotto in italiano la stringa `$Revision: 1.2 $`.

Hanno contribuito a questo manuale Matthew Copeland, Joergen Grahn, and David A. Wheeler. J Horacio MG ha tradotto il manuale in spagnolo, Lorenzo Cappelletti in italiano.

Diario delle Revisioni

Revisione 1.2 2002/03/17 Revisionato da: LC

We eventually got a CVS access: - Spelling corrections thanks to Fabio Bonelli. - All closing angle brakets are coded now. - Dead linker

Revisione 1.1 2001/03/07 Revisionato da: LC

This revision logs come from my own RCS log, which I used before I was given a CVS access to cvs.gnupg.org's repository. 2001-03-0

Sommario

Lista delle Figure

Capitolo 1. Incominciamo

GnuPG è uno strumento per comunicare in modo sicuro. Questo capitolo è una breve guida riguardante il nocciolo funzionale di GnuPG; include la creazione di coppie di chiavi, scambio e verifica di chiavi, cifratura e decifratura di documenti e autenticazione di documenti con firme digitali. Non spiega invece nel dettaglio i concetti che stanno dietro alla crittografia a chiave pubblica, alla cifratura e alle firme digitali. Tali argomenti, infatti, vengono trattati nel capitolo 2. Il presente capitolo non spiega nemmeno come usare GnuPG con una certa cognizione di causa; ciò fa parte dei capitoli 3 e 4.

GnuPG utilizza la crittografia a chiave pubblica per permettere a coloro che lo utilizzano di comunicare in sicurezza. In un sistema a chiave pubblica ogni utente ha una coppia di chiavi consistenti in una *chiave privata* e una *chiave pubblica*. La chiave privata di una persona viene tenuta segreta; non deve mai essere rivelata. La chiave pubblica può essere data a tutti coloro con i quali l'utente vuole comunicare. GnuPG utilizza uno schema in qualche modo più sofisticato per il quale un utente possiede una coppia di chiavi primaria e zero o più coppie di chiavi subordinate addizionali. La coppia di chiavi primaria e quelle subordinate sono raggruppate assieme per facilitare la gestione delle chiavi e il mazzo così ottenuto può spesso essere considerato semplicemente come un'unica coppia di chiavi.

Generare una nuova coppia di chiavi

L'opzione a linea di comando `--gen-key` è utilizzata per creare una nuova coppia di chiavi primaria.

```
alice% gpg --gen-key
gpg (GnuPG) 0.9.4; Copyright (C) 1999 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

Per favore scegli che tipo di chiave vuoi:
   (1) DSA e ElGamal (default)
   (2) DSA (firma solo)
   (4) ElGamal (firma e cifra)
Cosa scegli?
```

GnuPG è in grado di creare diversi tipi di coppie di chiavi, ma una chiave primaria deve essere capace di fare firme. Ci sono pertanto solo tre opzioni. L'opzione 1 crea in realtà due coppie di chiavi: una coppia di chiavi di tipo DSA che rappresenta la coppia di chiavi primaria ed è utilizzabile solo per firmare; una coppia di chiavi subordinata di tipo ElGamal, usata per criptare. L'opzione 2 è simile alla precedente ma crea solo una coppia di chiavi DSA. L'opzione 4[1]crea una singola coppia di chiavi ElGamal utilizzabile sia per firmare che per criptare. In tutti i casi è possibile in un secondo momento creare sotto-chiavi addizionali per cifrature e firme.

È necessario anche scegliere la dimensione della chiave. La dimensione di una chiave DSA deve essere compresa fra 512 e 1024 bit mentre una chiave ElGamal può essere di qualsiasi dimensione. GnuPG, però, richiede che le chiavi non siano più piccole di 768 bit. Accade quindi che, se si è scelta l'opzione 1 e successivamente si sceglie una dimensione per la chiave maggiore di 1024 bit, la chiave ElGamal avrà la dimensione richiesta, mentre la chiave DSA sarà di 1024 bit.

```
Sto per generare una nuova coppia di chiavi ELG-E.
              la dimensione minima è  768 bit
         la dimensione predefinita è 1024 bit
```

```
  la dimensione massima consigliata è 2048 bit
Di che dimensioni vuoi la chiave? (1024)
```

Più lunga è la chiave maggiore è la sicurezza contro attacchi a forza bruta, anche se in pratica per un utilizzo comune la dimensione di default della chiave è adeguata. Con una chiave lunga, infatti, diventa più economico aggirare la cifratura piuttosto che provare a romperla. Inoltre cifratura e decifratura sono più lente e una dimensione maggiore della chiave può influenzare negativamente la lunghezza della firma. Una volta scelta, la dimensione della chiave non può più essere modificata.

Infine è necessario scegliere una data di scadenza. Se è stata scelta l'opzione 1, la data di scadenza verrà utilizzata sia per la coppia di chiavi ElGamal che per quella DSA.

```
Per favore specifica per quanto la chiave sarà valida.
         0 = la chiave non scadrà
      <n>  = la chiave scadrà dopo n giorni
      <n>w = la chiave scadrà dopo n settimane
      <n>m = la chiave scadrà dopo n mesi
      <n>y = la chiave scadrà dopo n anni
Chiave valida per? (0)
```

Per la maggior parte degli utenti una chiave che non scade risulta adeguata. Il tempo di scadenza, in caso contrario, dovrebbe essere scelto con cura in quanto, anche se è possibile cambiare la data di scadenza dopo che la chiave è stata creata, potrebbe risultare dificile comunicare un cambiamento alle persone che possiedono quella chiave pubblica.

È necessario fornire un identificativo utente[2] in aggiunta ai parametri della chiave. Lo User ID viene utilizzato per associare la chiave che si sta creando ad una persona reale.

```
Ti serve uno User ID per identificare la tua chiave; il software costruisce l'user id a partire da Nome e Cogn
    "Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Nome e Cognome:
```

Solamente uno User ID viene creato nel momento in cui si genera una nuova chiave. È comunque possibile aggiungere ulteriori User ID in seguito nel caso in cui si desiderasse utilizzare la chiave in due o più contesti diversi, come ad esempio sul lavoro e all'interno della propria sezione di partito. Uno User ID deve essere creato con cura in quanto non può più essere modificato.

GnuPG necessita di una "frase d'ordine"[3] per proteggere le chiavi primarie e subordinate che si possiedono.

```
Ti serve una passphrase per proteggere la tua chiave segreta.

Inserisci la passphrase:
```

Non ci sono limiti alla lunghezza della passphrase, la quale dovrebbe essere scelta con attenzione. Dal punto di vista della sicurezza, la passphrase usata per sbloccare la chiave privata è uno dei punti più deboli di GnuPG (così come di altri sistema di crittografia a chiave pubblica), in quanto è l'unica protezione che si possiede nel caso in cui un'altra persona entri in possesso della propria chiave privata. Idealmente la passphrase non dovrebbe utilizzare parole prese da un dizionario e dovrebbe usare tanto caratteri minuscoli e minuscoli quanto caratteri non-alfabetici. Una buona passphrase è cruciale per un uso sicuro di GnuPG.

Generare un certificato di revoca

Una volta che la propria coppia di chiavi è stata creata, si dovrebbe immediatamente generare un certificato di revoca per la chiave pubblica primaria utilizzando l'opzione `--gen-revoke`. Se ci si dimentica la passphrase o se la propria chiave privata viene compromessa o persa, questo certificato di revoca può essere pubblicato per segnalare ad altri che la chiave pubblica non deve più essere usata. Una chiave pubblica revocata può comunque ancora essere utilizzata per verificare firme fatte in passato, ma non può più essere usata per cifrare futuri messaggi. Inoltre la revoca non influisce sulla propria capacità di decifrare messaggi spediti in passato, se si possiede ancora l'accesso alla chiave privata.

```
alice% gpg --output revoca.asc --gen-revoke mia_chiave
[...]
```

L'argomento **`mia_chiave`** deve essere uno *specificatore di chiave*, cioè o l'ID della propria coppia primaria di chiavi o una qualsiasi altra parte dello User ID che identifica la propria coppia di chiavi. Il certificato generato verrà riposto nel file *`revoca.asc`*. Se l'opzione `--output` è omessa, il risultato verrà stampato sullo standard output. Poiché il certificato è breve, si può pensare di stamparne una copia e tenerlo al sicuro da qualche parte, ad esempio nella propria cassetta di sicurezza. Il certificato non dovrebbe venir riposto in luoghi dove altri possono aver accesso in quanto chiunque può pubblicare il certificato di revoca e rendere la chiave pubblica corrispondente inutile.

Scambiarsi le chiavi

Per comunicare con altre persone è necessario scambiarsi le chiavi pubbliche. Per elencare le chiavi presenti nel proprio portachiavi pubblico utilizzare l'opzione a linea di comando `--list-keys`.

```
alice% gpg --list-keys
/users/alice/.gnupg/pubring.gpg
---------------------------------------
pub  1024D/BB7576AC 1999-06-04 Alice (giudice) <alice@cyb.org>
sub  1024g/78E9A8FA 1999-06-04
```

Esportare una chiave pubblica

Per spedire una chiave pubblica ad un corrispondente è necessario prima esportarla. A questo scopo si usa l'opzione a linea di comando `--export`. Essa necessita di un ulteriore argomento che identifichi la chiave pubblica da esportare. Così come con l'opzione `--gen-revoke`, sia l'ID della chiave che ogni altra parte dello User ID possono servire per identificare la chiave da esportare.

```
alice% gpg --output alice.gpg --export alice@cyb.org
```

La chieve è esportata in un formato binario, ma ciò può risultare sconveniente quando la chiave viene spedita per posta elettronica o pubblicata in una pagina web. GnuPG supporta perciò l'opzione a linea di comando `--armor`[4] che forza l'output ad essere generato in un formato protetto da un'armatura ASCII[5] simile ai documenti codificati con uuencode. In generale, qualsiasi output di GnuPG, cioè chiavi, documenti cifrati e firme, possono essere ASCII-armored aggiungendo l'opzione `--armor`.

```
alice% gpg --armor --export alice@cyb.org
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v0.9.7 (GNU/Linux)
Comment: For info see http://www.gnupg.org

[...]
-----END PGP PUBLIC KEY BLOCK-----
```

Importare una chiave pubblica

Una chiave pubblica può essere aggiunta al proprio mazzo di chiavi mediante l'opzione `--import`.

```
alice% gpg --import blake.gpg
gpg: chiave 9E98BC16: chiave pubblica importata
gpg: numero totale esaminato: 1
gpg:               importate: 1
alice% gpg --list-keys
/users/alice/.gnupg/pubring.gpg
---------------------------------------
pub  1024D/BB7576AC 1999-06-04 Alice (giudice) <alice@cyb.org>
sub  1024g/78E9A8FA 1999-06-04

pub  1024D/9E98BC16 1999-06-04 Blake (esecutore) <blake@cyb.org>
sub  1024g/5C8CBD41 1999-06-04
```

Una volta che la chiave è stata importata deve venir convalidata. GnuPG utilizza un potente e flessibile modello basato sulla fiducia che non richiede all'utente di convalidare personalmente ogni chiave che viene importata. Può comunque risultare necessaria la convalida personale di alcune chiavi. Una chiave viene convalidata verificando l'impronta digitale[6] della chiave stessa e successivamente firmando la chiave per certificarla come chiave valida. L'impronta digitale di una chiave può essere velocemente visualizzata con l'opzione a linea di comando `--fingerprint`, ma, allo scopo di certificare la chiave, è necessario editarla.

```
alice% gpg --edit-key blake@cyb.org

pub  1024D/9E98BC16  created: 1999-06-04 expires: never      trust: -/q
sub  1024g/5C8CBD41  created: 1999-06-04 expires: never
(1)  Blake (esecutore) <blake@cyb.org>

Comando> fpr
pub  1024D/9E98BC16 1999-06-04 Blake (esecutore) <blake@cyb.org>
             Impronta digitale: 268F 448F CCD7 AF34 183E  52D8 9BDE 1A08 9E98 BC16
```

L'impronta digitale di una chiave va verificata con il possessore di quella chiave. Ciò può essere fatto di persona, per telefono o attraverso un qualsiasi altro mezzo con il quale sia possibile garantire che si sta comunicando con il vero possessore della chiave. Se l'impronta digitale che si riceve è la stessa impronta digitale che il possessore della chiave detiene, allora si può essere sicuri di possedere una corretta copia della chiave.

Dopo aver controllato l'impronta digitale, si può procedere alla firma in modo da convalidarla. Poiché la verifica di una chiave rappresenta un punto debole nella crittografia a chiave pubblica, è necessario essere estremamente attenti è controllare *sempre* un'impronta digitale di una chiave con il possessore prima di firmare la chiave stessa.

```
Comando> sign

pub  1024D/9E98BC16  created: 1999-06-04 expires: never      trust: -/q
             Fingerprint: 268F 448F CCD7 AF34 183E  52D8 9BDE 1A08 9E98 BC16

     Blake (esecutore) <blake@cyb.org>

Sei davvero sicuro di volere firmare questa chiave
con la tua chiave: "Alice (giudice) <alice@cyb.org>"

Firmo davvero?
```

Una volta firmato è possibile controllare la chiave listando le firme ad essa applicate e rilevare la firma che si è appena aggiunta. Ogni User ID avrà sulla chiave una o più autofirme e una firma per ogni utente che ha convalidato la chiave.

```
Comando> check
uid  Blake (esecutore) <blake@cyb.org>
sig!       9E98BC16 1999-06-04   [autofirma]
sig!       BB7576AC 1999-06-04   Alice (giudice) <alice@cyb.org>
```

Cifrare e decifrare documenti

Chiave pubblica e privata hanno ognuna uno specifico ruolo nella codifica e decodifica di documenti. Una chiave pubblica può essere vista come una cassaforte aperta. Quando un corrispondente cripta un documento utilizzando una chiave pubblica, quel documento viene messo nella cassaforte, la cassaforte viene chiusa ed il lucchetto a combinazione fatto girare diverse volte. La chiave privata corrispondente è la combinazione che può riaprire la cassaforte e recuperare il documento. In altre parole, solo la persona che detiene la chiave privata può recuperare un documento cifrato utilizzando la chiave pubblica corrispondente.

La procedura per criptare e decriptare documenti è banale con questo modello mentale. Se si desidera cifrare un messaggio per Alice, lo si cripta utilizzando la chiave pubblica di Alice e lei lo decripterà con la sua chiave privata. Se Alice vuole spedirvi un messaggio, lo cripterà utilizzando la vostra chiave pubblica e voi lo decripterete con la vostra chiave privata.

Per cifrare un documento viene utilizzata l'opzione `--encrypt`. È necessario possedere le chiavi pubbliche dei destinatari a cui si intende spedire il messaggio. Il programma si aspetta il nome del documento da cifrare come ingresso; se omesso, legge lo standard input. Il risultato cifrato è stampato sullo standard output oppure dove specificato con l'opzione `--output`. Il documento, oltre ad essere criptato, viene compresso per ragioni di maggior sicurezza.

```
alice% gpg --output doc.gpg --encrypt --recipient blake@cyb.org doc
```

L'opzione `--recipient` viene utilizzata una sola volta per ogni destinatario e richiede un argomento extra che specifichi con quale chiave pubblica debba essere criptato il documento. Tale documento può essere decriptato solo da qualcuno in possesso di una chiave privata che complementi una delle chiave pubbliche dei destinatari. In particolare non è possibile decifrare un documento criptato da voi stessi, a meno che non abbiate incluso la vostra chiave pubblica nella lista dei destinatari.

Per decriptare un messaggio si usa l'opzione `--decrypt`. È necessario possedere la chiave privata con la quale era stato cifrato il messaggio. Analogamente al processo di cifratura, il documento da decifrare è l'ingresso e quello decifrato è l'uscita.

```
blake% gpg --output doc --decrypt doc.gpg

Ti serve una passphrase per sbloccare la chiave segreta
dell'utente: "Blake (esecutore) <blake@cyb.org>"
chiave ELG-E di 1024 bit, ID 5C8CBD41, creata il 1999-06-04 (key ID principale 9E98BC16)

Inserisci la passphrase:
```

I documenti posso anche essere criptati senza l'utilizzo della crittografia a chiave pubblica. Al suo posto è possibile utilizzare un algoritmo di crittografia simmetrico per cifrare il documento. La chiave fornita all'algoritmo simmetrico viene derivata da una frase d'ordine fornita al momento in cui il documento viene criptato e, per una buona sicurezza, non dovrebbe essere la stessa passphrase utilizzata per proteggere la propria chiave privata. La cifratura simmetrica è utile per rendere sicuri i propri documenti quando non è necessario comunicare ad altri la parola d'ordine utilizzata. Un documento può essere criptato con un algoritmo simmetrico utilizzando l'opzione `--symmetric`.

```
alice% gpg --output doc.gpg --symmetric doc
Inserisci la passphrase:
```

Fare e verificare firme

Una firma digitale certifica e appone la data ad un documento. Se il documento viene successivamente modificato in qualsiasi modo, una verifica della firma fallirà. Una firma digitale può servire allo stesso scopo per il quale si utilizza una firma fatta a mano con l'ulteriore beneficio di essere a prova di manomissione. La distribuzione dei sorgenti di GnuPG, per esempio, è firmata in modo tale da permettere agli utenti di verificare che il codice sorgente non sia stato modificato dal momento in cui è stato creato il pacchetto.

La creazione e la verifica di firme utilizzano la coppia di chiavi pubblica/privata in modo differente da quanto fanno le operazioni di cifratura e decifratura. Una firma è fatta utilizzando la chiave privata di colui che firma. La firma viene verificata utilizzando la corrispondete chiave pubblica. Per esempio Alice userebbe la propria chiave privata per firmare digitalmente il suo ultimo lavoro per la rivista di chimica inorganica. Il corrispondente editore nel pubblicare il lavoro userebbe la chiave pubblica di Alice per controllare la firma e verificare che il pezzo sia effettivamente stato mandato da Alice e che non sia stato modificato dal momento in cui Alice l'ha spedito. Una conseguenza dell'utilizzo di firme digitali consiste nel fatto che è difficile negare di aver fatto una firma digitale in quanto ciò implicherebbe che la propria chiave privata era stata compromessa.

L'opzione a linea di comando `--sign` viene usata per fare firme digitali. Il documento da firmare è l'ingresso, quello firmato è l'uscita.

```
alice% gpg --output doc.sig --sign doc

Ti serve una passphrase per sbloccare la chiave segreta
dell'utente: "Alice (giudice) <alice@cyb.org>"
chiave DSA di 1024 bit, ID BB7576AC, creata il 1999-06-04
```

```
Inserisci la passphrase:
```

Il documento viene compresso prima di essere firmato e l'output è in formato binario.

Dato un documento firmato, è possibile sia controllare la firma che controllare la firma e recuperare il documento originale. Per controllare la firma si utilizza l'opzione `--verify`. Per verificare la firma ed estrarre un documento si usa l'opzione `--decrypt`. Il documento firmato da verificare e recuperare è l'ingresso, mentre il documento recuperato è l'uscita.

```
blake% gpg --output doc --decrypt doc.sig
gpg: Firma fatta ven 04 feb 1999 12:02:38 CDT usando la chiave DSA con ID BB7576AC
gpg: Firma valida da "Alice (giudice) <alice@cyb.org>"
```

Documenti firmati in chiaro

Un uso comune di firme digitali consiste nel firmare messaggi per Usenet o messaggi di posta elettronica. In tali situazioni non è desiderabile comprimere il documento quando lo si firma. L'opzione `--clearsign` avvolge il documento in una firma ASCII-armored ma non lo modifica in nessun altro modo.

```
alice% gpg --clearsign doc

Ti serve una passphrase per sbloccare la chiave segreta
dell'utente: "Alice (giudice) <alice@cyb.org>"
chiave DSA di 1024 bit, ID BB7576AC, creata il 1999-06-04

-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

[...]
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v0.9.7 (GNU/Linux)
Comment: For info see http://www.gnupg.org

iEYEARECAAYFAjdYCQoACgkQJ9S6ULt1dqz6IwCfQ7wP6i/i8HhbcOSKF4ELyQB1
oCoAoOuqpRqEzr4kOkQqHRLE/b8/Rw2k
=y6kj
-----END PGP SIGNATURE-----
```

Firme distaccate

Un documento firmato ha un'utilità limitata. Gli altri utenti devono recuperare il documento originale dalla versione firmata e, anche con documenti firmati in chiaro, il documento firmato deve essere editato per poter recuperare l'originale. Esiste perciò un terzo metodo per firmare un documento. Con tale metodo viene creata una firma distaccata in un file separato. Una firma distaccata viene creata utilizzando l'opzione `--detach-sig`.

```
alice% gpg --output doc.sig --detach-sig doc

Ti serve una passphrase per sbloccare la chiave segreta
dell'utente: "Alice (giudice) <alice@cyb.org>"
```

```
chiave DSA di 1024 bit, ID BB7576AC, creata il 1999-06-04

Inserisci la passphrase:
```

Sia il documento che la firma distaccata sono necessarie per verificare la firma stessa. L'opzione `--verify` può essere utilizzata per controllare la firma.

```
blake% gpg --verify doc.sig doc
gpg: Firma fatta ven 04 feb 1999 12:38:46 CDT usando la chiave DSA con ID BB7576AC
gpg: Firma valida da "Alice (giudice) <alice@cyb.org>"
```

Note

1. L'opzione 3 serve a generare una coppia di chiavi ElGamal che non è utilizzabile per fare firme.
2. D'ora in poi *User ID* per rispettare la traduzione del programma.
3. D'ora in poi *passphrase* come nel testo originale.
4. Numerose opzioni a linea di comando di uso frequente possono essere impostate in un file di configurazione.
5. D'ora in poi *ASCII-armored*.
6. Si noti che qui l'aggettivo *digitale* può assumere due significati, entrambi validi. Il primo è quello che deriva dalla traduzione della parola inglese originaria *fingerprint*, impronta delle dita. Il secondo significato è quello di "prodotto con l'ausilio di un computer", che è la macchina digitale per eccellenza.

Capitolo 2. Concetti

GnuPG fa uso di diversi concetti di crittografia come *algoritmi simmetrici*, *algoritmi a chiave pubblica*, e *hashing a senso unico*. È possibile utilizzare le funzioni di base di GnuPG senza comprendere appieno tali concetti, ma, se si vuole usarlo con cognizione di causa, una loro comprensione è necessaria.

Questo capitolo introduce i concetti di base della crittografia utilizzati in GnuPG. Si possono trovare altri libri che trattano questi argomenti più dettagliatamente. Un buon testo per approfondire ulteriormente gli studi è "Applied Cryptography" (http://www.counterpane.com/applied.html) di Bruce Schneier (http://www.counterpane.com/schneier.html).

Algoritmi simmetrici

Un algoritmo simmetrico è un algoritmo che utilizza la stessa chiave sia per criptare che per decriptare. Due parti che comunicano sfruttando un algoritmo simmetrico devono innanzi tutto mettersi d'accordo sulla chiave. Una volta d'accordo, il mittente cifra un messaggio utilizzando la chiave, lo spedisce al destinatario e questi decripta il messaggio usando la stessa chiave. Per esempio, il tedesco Enigma è un algoritmo simmetrico per il quale venivano distribuite delle chiavi giornaliere sotto forma di libri di codici. Ogni giorno un operatore radio, fosse esso un trasmittente o un ricevente, consultava la propria copia del libro codici per trovare la chiave di quel giorno. Il traffico radio di quel giorno veniva quindi criptato e decriptato utilizzando la quella chiave. Esempi moderni di algoritmi simmetrici includono 3DES, Blowfish e IDEA.

Un buon algoritmo di cifratura racchiude completamente la sicurezza nella chiave senza lasciare nulla nell'algoritmo. In altre parole, non dovrebbe essere di alcun aiuto per un malintenzionato conoscere il tipo di algoritmo utilizzato. Solo se ottenesse la chiave la conoscenza dell'algoritmo sarebbe necessaria. L'algoritmo usato in GnuPG possiede tale proprietà.

Poiché tutta la sicurezza è riposta nella chiave, è importante che sia veramente difficile indovinare la chiave stessa. Detto altrimenti, l'insieme di chiavi possibili, cioè lo *spazio delle chiavi*, deve essere grande. A Los Alamos, Tichard Feynman era famoso per la sua abilità nell'aprire casseforti. Per incoraggiare l'alone di mistero che lo circondava egli portava con sé perfino un serie di attrezzi, compreso un vecchio stetoscopio. In realtà egli usava una varietà di trucchi per ridurre il numero di combinazione che doveva provare e poi semplicemente tirava ad indovinare finché trovava la giusta combinazione. In altre parole, egli riduceva la dimensione dello spazio di chiavi.

La Gran Bretagna, durante la Seconda Guerra Mondiale, usò delle macchine per cercare di indovinare le chiavi usate dagli avversari. Il tedesco Enigma, infatti, possedeva uno spazio di chiavi veramente ampio, ma la Gran Bretagna costruì dei motori di calcolo specializzati, i Bombes, per provare meccanicamente le chiavi finché la chiave del giorno non veniva trovata. Questo significa che a volte riuscivano a trovare la chiave di quel giorno in capo a poche ore dal momento in cui una nuova chiave veniva usata, ma significa anche che alcuni volte non riuscissero affatto a trovare la chiave giusta. I Bombes non erano computer multi-funzione ma erano comunque i precursori dei nostri moderni elaboratori.

Oggigiorno i computer possono indovinare una chiave molto rapidamente e questo è il motivo per cui la dimensione della chiave è un requisito importante per i moderni sistemi di crittografia. L'algoritmo DES usa una chiave da 56 bit. Ciò significa che ci sono 2^{56} chiavi possibili. 2^{56} sono 72,057,594,037,927,936 chiavi. Un sacco di chiavi, ma un computer non specializzato può controllarle tutte in una manciata di giorni. Un computer specializzato in poche ore. D'altro canto, algoritmi sviluppati più recentemente, come il 3DES, il Blowfish e IDEA, usano tutti chiavi da 128 bit. Ciò implica che ci sono 2^{128} possibili chiavi. Queste sono molte, molte di più e, anche se tutti i computer della terra cooperassero, sarebbe ancora necessario più tempo di quello rappresentato dall'età dell'universo per trovare la chiave corretta.

Algoritmi a chiave pubblica

Il problema principale con gli algoritmi simmetrici non risiede nella loro sicurezza, ma nello scambio della chiave. Una volta che il mittente ed il destinatario si sono scambiati la chiave, quella chiave può essere usata per comunicare in sicurezza. Ma quale canale sicuro è stato utilizzato per comunicare la chiave stessa? In particolare sarebbe probabilmente più semplice per un malintenzionato cercare di intercettare la chiave piuttosto che provare tutte le chiavi possibili dello spazio di chiavi. Un altro problema consiste nel numero di chiavi necessarie. Se ci sono n persone che vogliono comunicare privatamente fra loro, allora servono $n(n-1)/2$ chiavi per ogni coppia di persone. Ciò può andar bene per una ristretta cerchia di persone, ma il numero diventa rapidamente enorme per un gruppo largo.

Gli algoritmi a chiave pubblica furono inventati per aggirare completamente il problema dello scambio di chiavi. Un algoritmo a chiave pubblica utilizza una coppia di chiavi per spedire messaggi, entrambi appartenenti alla persona che riceve il messaggio. Una chiave è detta *chiave pubblica* e può essere data a chiunque. L'altra chiave è detta *chiave privata* e viene mantenuta segreta dal suo possessore. Il mittente cifra un messaggio usando la chiave pubblica e, una volta criptato, il messaggio può essere decifrato solo con la chiave privata.

Questo protocollo risolve il problema dello scambio di chiavi intrinseco agli algoritmi simmetrici. Non c'è bisogno che mittente e destinatario si mettano d'accordo su una chiave comune. Tutto ciò che serve è che, qualche tempo prima della effettiva comunicazione segreta, il mittente entri in possesso di una copia della chiave pubblica del destinatario. Inoltre una sola chiave pubblica può essere utilizzata da chiunque desideri comunicare con il destinatario. Così solo n coppie di chiavi sono sufficienti a permettere ad n persone di comunicare segretamente una con l'altra.

Gli algoritmi a chiave pubblica sono basati sulle funzioni a difficilmente invertibili con trapdoor[1] o, più brevemente, funzioni trapdoor. Una funzione difficilmente invertibile è una funzione facile da computare, ma la cui inversa è di difficile calcolo. Per esempio, è facile moltiplicare assieme due numeri primi per ottenere un numero composto, ma è difficile fattorizzare un numero composto nelle sue componenti prime. Una funzione trapdoor è simile, ma possiede una scappatoia: se si conosce una parte dell'informazione, diventa facile calcolarne l'inversa. Per esempio, se si considera un numero composto da due fattori primi, allora, conoscendo uno dei due fattori, risulta facile calcolare l'altro. Dato un algoritmo a chiave pubblica basato sulla fattorizzazione in numeri primi, la chiave pubblica contiene un numero composto formato da due fattori primi elevati e l'algoritmo di cifratura usa questo numero composto per criptare il messaggio. L'algoritmo per decriptare il messaggio richiede la conoscenza dei due fattori primi. Così, possedendo la chiave privata che contiene uno dei due fattori, è facile decifrare il messaggio, mentre è estremamente difficile se non si conosce la chiave privata.

Così come accade per gli algoritmi simmetrici, anche per gli algoritmi a chiave pubblica tutta la sicurezza risiede nella chiave. Perciò la dimensione della chiave è una misura della sicurezza del sistema, anche se non è possibile paragonare la dimensione della chiave di un algoritmo simmetrico con quella di di un algoritmo a chiave pubblica per misurare il loro grado di sicurezza. In un attacco a forza bruta contro un algoritmo simmetrico con una chiave da 80 bit, un malintenzionato deve contare al massimo 2^{80} chiavi per trovare quella giusta. In un attacco a forza bruta contro un algoritmo a chiave pubblica con una dimensione della chiave pari a 512 bit, lo stesso malintenzionato deve fattorizzare un numero composto codificato in 512 bit (fino a 155 cifre decimali). Il carico di lavoro per il malintenzionato è fondamentalmente differente a seconda dell'algoritmo che viene attaccato. Mentre 128 bit sono sufficienti per un algoritmo simmetrico, data la tecnologia odierna di fattorizzazione, sono raccomandate chiavi da 1024 bit per la maggior parte degli scopi.

Algoritmi ibridi

Gli algoritmi a chiave pubblica non sono una panacea. Molti algoritmi simmetrici sono più forti dal punto di vista della sicurezza; le operazioni di criptazione e decriptazione a chiave pubblica sono più costose delle corrispondenti operazioni dei sistemi simmetrici. Ciò nonostante, gli algoritmi a chiave pubblica rappresentano uno strumento efficace per distribuire le chiavi degli algoritmi simmetrici e per questo vengono usati in sistemi di crittografia ibridi.

Un algoritmo ibrido utilizza sia un sistema simmetrico che uno a chiave pubblica. In particolare esso funziona utilizzando un algoritmo a chiave pubblica per condividere una chiave per il sistema simmetrico. Il messaggio effettivo è quindi criptato usando tale chiave e successivamente spedito al destinatario. Poiché il metodo di condivisione della chiave è sicuro, la chiave simmetrica utilizzata è differente per ogni messaggio spedito. Per questo viene detta a volte chiave di sessione.

Sia PGP che GnuPG usano algoritmi ibridi. La chiave di sessione, criptata utilizzando l'algoritmo a chiave pubblica, e il messaggio da spedire, cifrato con l'algoritmo simmetrico, sono automaticamente combinati in un solo pacchetto. Il destinatario usa la propria chiave privata per decifrare la chiave di sessione che viene poi usata per decifrare il messaggio.

Un algoritmo ibrido non è mai più forte del più debole algoritmo utilizzato, sia esso quello a chiave pubblica o quello simmetrico. In PGP e GnuPG l'algoritmo a chiave pubblica è probabilmente il più debole dei due. Fortunatamente, però, se un malintenzionato dovesse decifrare una chiave di sessione, egli sarebbe in grado di leggere solo un messaggio, quello criptato con quella chiave di sessione. Il malintenzionato dovrebbe ricominciare di nuovo e decifrare un'altra chiave di sessione per poter leggere un altro messaggio.

Firme digitali

Una funzione hash è una funzione da molti a uno che mappa i suoi valori di ingresso in un valore appartenente ad un insieme finito. Tipicamente questo insieme è un intervallo di numeri naturali. Una semplice funzione hash è $f(x) = 0$ per tutti gli interi x. Una funzione hash più interessante è $f(x) = x \mathit{mod} 37$, che mappa tutti gli x al resto della divisione tra x e 37.

La firma digitale di un documento è il risultato dell'applicazione di una funzione hash al documento stesso. Per essere utile, però, la funzione hash deve soddisfare a due importanti proprietà. Primo, dev'essere difficile trovare due documenti che possiedono la stessa valore di hash; secondo, dato un valore di hash deve essere difficile recuperare il documento che ha prodotto quel valore.

Alcuni algoritmi a chiave pubblica[2] possono venire usati per firmare documenti. Colui che firma cripta il documento con la propria chiave *privata*. Chiunque voglia controllare la firma e vedere il documento usa semplicemente la chiave pubblica del firmatario per decifrare il documento. Questo algoritmo effettivamente soddisfa alle due proprietà richieste da una buona funzione hash, ma, in pratica, è troppo lento per risultare utilizzabile.

Un'alternativa consiste nel'utilizzare funzioni di hash pensate specificamente per soddisfare a queste due importanti proprietà. SHA e MD5 sono due esempi di tali algoritmi. Utilizzando un algoritmo di questi, un documento viene firmato applicando la funzione di hash ed il valore restituito rappresenta la firma. Un'altra persona può controllare la firma applicando la stessa funzione di hash alla propria copia del documento e confrontando il valore di hash ottenuto con quello del documento originale. Se coincidono, può essere praticamente certo che i documenti sono identici.

Ovviamente ora il problema consiste nell'usare una funzione di hash per firme digitali senza permettere ad un malintenzionato di interferire con il controllo della firma. Se documento e firma sono spediti in chiaro, un malintenzionato potrebbe infatti modificare il documento e generare la corrispondente firma senza che il destinatario ne venga a conoscenza. Se solo il documento è cifrato, un malintenzionato potrebbe manomettere la firma e provocare un fallimento del controllo sulla firma. Una terza possibilità consiste nell'usare una cifratura a chiave pubblica ibrida per criptare sia la firma che il documento. Il firmatario usa la propria chiave privata e chiunque può adoperare la corrispondente chiave pubblica per controllare la firma ed il documento. Quest'ultimo procedimento sembra corretto, ma in effetti non ha senso. Se tale algoritmo mettesse veramente al sicuro il documento, esso sarebbe anche al sicuro da eventuali manomissioni e non ci sarebbe bisogno di alcuna firma. Il problema più serio, comunque, consiste nel fatto che tutto ciò non protegge da possibili manomissioni né la firma né il documento. Con il nostro algoritmo, infatti, solo la chiave di sessione per l'algoritmo simmetrico viene criptata usando la chiave privata del firmatario. Chiunque è in grado di usare la chiave pubblica per recuperare la chiave di sessione. Perciò sarebbe banale per un malintenzionato recuperare tale chiave di sessione e usarla per criptare documenti modificati e firme da spedire ad altri in nome del mittente.

Un algoritmo valido è quello che usa un algoritmo a chiave pubblica per cifrare solo la firma. In particolare, il valore di hash viene criptato usando la chiave privata del firmatario permettendo a chiunque di controllare la firma usando la corrispondente chiave pubblica. Il documento firmato può essere spedito usando qualsiasi altro algoritmo di cifratura, compreso nessuno se si tratta di un documento pubblico. Se il documento venisse modificato, il controllo della firma fallirebbe, ma ciò è quello a cui serve il controllo della firma. Il Digital Signature Standard[3] (DSA) è un algoritmo per la firma a chiave pubblica che funziona come appena descritto. Il DSA è l'algoritmo principale usato da GnuPG per firmare documenti.

Note

1. *One-way trapdoor function* nel testo originale. Qui il termine *trapdoor* potrebbe essere tradotto con botola, scappatoia. Tali espressioni però non sono utilizzate nella pratica.
2. L'algoritmo deve possedere la proprietà che l'effettiva chiave pubblica o privata possa essere usata dall'algoritmo di cifratura come chiave pubblica. L'RSA è un esempio di tale algoritmo, mentre ElGamal non possiede tale proprietà.
3. Lo standard per la firma digitale.

Capitolo 3. Gestione delle chiavi

La manomissione delle chiavi è una delle principali debolezze per quanto concerne la sicurezza della crittografia a chiave pubblica. Uno spione potrebbe manomettere il mazzo di chiavi di un utente o creare la chiave pubblica di qualcuno e postarla affinché altri la scarichino e la utilizzino. Per esempio, si supponga che Chloe voglia tenere sotto controllo i messaggi che Alice spedisce a Blake. Potrebbe instaurare un attacco detto *uomo nel mezzo*. In questo tipo di attacco Chloe crea un nuova coppia di chiavi pubblica/privata e rimpiazza la copia della chiave pubblica di Blake in possesso di Alice con la nuova chiave pubblica. Successivamente si mette ad intercettare i messaggi che Alice spedisce a Blake. Ogni messaggio intercettato viene decriptato utilizzando la nuova chiave privata e recriptato usando la vera chiave pubblica di Blake, spedendo poi tale messaggio a Blake. Tutti i messaggi spediti da Alice a Blake possono ora essere letti da Chloe.

Una buona gestione delle chiavi è cruciale se si desidera assicurare non solo l'integrità del proprio mazzo di chiavi, ma anche l'integrità dei mazzi di chiavi di altri utenti. Il cuore della gestione delle chiavi presente in GnuPG è la nozione di chiavi di firma[1]. La firma di una chiave ha due principali obiettivi: permettere di rilevare eventuali manomissioni al proprio mazzo di chiavi e permettere di certificare che una chiave appartenga veramente alla persona riportata dallo User ID della chiave. Le firme di una chiave vengono anche usate in uno schema conosciuto come *rete della fiducia*, la quale estende la certificazione delle chiavi non firmate di proprio pugno, ma da qualcun'altro di cui ci si fida. Utenti responsabili che operano una buona gestione delle chiavi possono vanificare le manomissioni delle chiavi praticate come attacco contro sistemi comunicazione sicura quali GnuPG.

Amministrare la propria coppia di chiavi

Una coppia di chiavi è composta da una chiave pubblica e da una privata. Una chiave pubblica consiste in una parte della chiave di firma generale, una parte delle sottochiavi subordinate di firma e cifratura e un insieme di User ID utilizzati per associare una chiave pubblica ad una persona reale. Ogni componente contiene delle informazioni circa se stesso. Per una chiave tale informazioni includono l'ID della chiave, quando è stata creata, quando scadrà, etc. Per uno User ID queste informazioni includono il nome della persona reale che la identifica, un commento opzionale e un indirizzo di posta elettronica. La struttura della chiave privata è simile, tranne per il fatto che essa contiene solo la parte privata delle chiavi e non ci sono informazioni sullo User ID.

L'opzione a linea di comando `--edit-key` può venir usata per visualizzare una coppia di chiavi. Per esempio

```
chloe% gpg --edit-key chloe@cyb.org
È disponibile una chiave segreta.

pub  1024D/26B6AAE1  creata il: 1999-06-15 scade: mai      fiducia: -/u
sub  2048g/0CF8CB7A  creata il: 1999-06-15 scade: mai
sub  1792G/08224617  creata il: 1999-06-15 scade: 2002-06-14
sub   960D/B1F423E7  creata il: 1999-06-15 scade: 2002-06-14
(1)  Chloe (giullare) <chloe@cyb.org>
(2)  Chloe (plebeo) <chloe@tel.net>
Comando>
```

La chiave pubblica viene visualizzata assieme ad un'indicazione circa la disponibilità di una chiave privata. Quindi vengono listate le informazioni disponibili per ogni componente della chiave pubblica. La prima colonna indica il tipo di chiave. La parola `pub` sta ad indicare la chiave pubblica principale di firma, mentre la parola `sub` sta ad indicare una chiave pubblica subordinata. La seconda colonna indica la lunghezza della chiave in bit, il tipo e l'ID. Il tipo può essere `D` per una chiave DSA, `g` per una chiave ElGamal di sola cifratura e `G` per una chiave ElGamal che può essere usata sia per cifrare che per firmare. Le date di creazione e di scadenza sono date dalle colonne tre e quattro. Le chiavi sono seguite dagli User ID.

Informazioni più dettagliate sulla chiave possono essere ottenute con comandi interattivi. Il comando **toggle** commuta fra le componenti pubbliche e quelle private della coppia di chiavi se queste sono effettivamente entrambi disponibili.

```
Comando> toggle

sec  1024D/26B6AAE1  creata il: 1999-06-15 scade: mai
sbb  2048g/0CF8CB7A  creata il: 1999-06-15 scade: mai
sbb  1792G/08224617  creata il: 1999-06-15 scade: 2002-06-14
sbb   960D/B1F423E7  creata il: 1999-06-15 scade: 2002-06-14
(1)  Chloe (giullare) <chloe@cyb.org>
(2)  Chloe (plebeo) <chloe@tel.net>
```

Le informazioni visualizzate sono simili a quelle fornite per la componente pubblica. La parola `sec` sta ad indicare che la chiave è privata e di firma principale, mentre la parola `sbb` sta ad indicare che le chiavi sono private e subordinate. Vengono listati per convenienza anche gli User ID della chiave pubblica.

Integrità della chiave

Quando si distribuisce la propria chiave pubblica, si rendono note le componenti pubbliche della propria chiave principale e di quelle subordinate assieme allo User ID. Distribuire solamente questo materiale, comunque, è un rischio per la sicurezza in quanto è possibile per un malintenzionato manomettere la chiave. La chiave pubblica, infatti, può essere modificata aggiungendo o sostituendo altre chiavi, oppure aggiungendo o cambiando lo User ID. Modificando lo User ID, un malintenzionato potrebbe cambiare l'indirizzo di posta elettronica dello User ID reale per fare in modo che arrivino al proprio indirizzo i messaggi dell'utente ignaro. Cambiando anche una delle chiavi di cifratura, il malintenzionato sarebbe perfino capace di decifrare i messaggi a lui reindirizzati.

L'utilizzo della firma digitale rappresenta una soluzione a questo problema. Quando delle informazioni sono firmate con una chiave privata, la corrispondente chiave pubblica è legata alle informazioni firmate. In altre parole, solo la corrispondente chiave pubblica può essere usata per verificare la firma e assicurare che quelle informazioni non siano state modificate. Una chiave pubblica può essere protetta contro la manomissione utilizzando la corrispondente chiave privata principale per firmare le componenti della chiave pubblica e lo User ID, in modo tale da legare tali componenti alla chiave pubblica principale. Firmare le componenti della chiave pubblica con la corrispondente chiave privata principale di firma è un'operazione che prende il nome di *autofirma*[2] e la chiave pubblica legata a degli User ID autofirmati prende il nome di *certificato*.

Per fare un esempio, si supponga che Chloe abbia due User ID e tre sottochiavi. Le firme degli User ID possono essere controllati con il comando **check** dal menù di modifica delle chiavi.

```
chloe% gpg --edit-key chloe
È disponibile una chiave segreta.
```

```
pub  1024D/26B6AAE1  creata il: 1999-06-15 scade: mai      fiducia: -/u
sub  2048g/0CF8CB7A  creata il: 1999-06-15 scade: mai
sub  1792G/08224617  creata il: 1999-06-15 scade: 2002-06-14
sub   960D/B1F423E7  creata il: 1999-06-15 scade: 2002-06-14
(1)  Chloe (giullare) <chloe@cyb.org>
(2)  Chloe (plebeo) <chloe@tel.net>

Comando> check
uid  Chloe (giullare) <chloe@cyb.org>
sig!     26B6AAE1 1999-06-15  [autofirma]
uid  Chloe (plebeo) <chloe@tel.net>
sig!     26B6AAE1 1999-06-15  [autofirma]
```

Come ci si aspettava, la chiave di firma di ogni User ID è la chiave di firma principale con ID `0x26B6AAE1`. Le autofirme delle sottochiavi sono presenti nella chiave pubblica, ma non vengono mostrate dall'interfaccia di GnuPG.

Aggiungere e togliere componenti alle chiavi

Sia nuove sottochiavi che nuovi User ID possono venir aggiunti alla propria coppia di chiavi dopo che questa è stata creata. Uno User ID viene aggiunto usando il comando **adduid**. Vengono richiesti il nome proprio, l'indirizzo email e un commento, proprio come se si stesse creando una coppia di chiavi iniziali. Una sottochiave viene aggiunta usando il comando **addkey**. L'interfaccia è simile a quella utilizzata durante la creazione di una coppia di chiavi iniziale. La sottochiave può essere una chiave di firma DSA, una chiave di sola cifratura ElGamal oppure una di firma e cifratura ElGamal. Quando viene generata una sottochiave o uno User ID, questi vengono autofirmati con la propria chiave di firma principale. Ecco il motivo per cui è necessario fornire la passphrase durante la generazione della chiave.

Ulteriori User ID ritornano utili quando si ha bisogno di più d'una personalità. Per esempio si può possedere un'identità per il proprio lavoro ed una per la propria attività politica. I colleghi di lavoro saranno a conoscenza dello User ID di lavoro, i colleghi di partito conosceranno quello politico. Poiché però questi gruppi di persone non dovrebbero sovrapporsi, ogni gruppo potrebbe non fidarsi dell'altro User ID. Entrambi gli User ID sono quindi necessari.

Anche avere più di una sottochiave può essere utile. Gli User ID associati alla propria chiave pubblica principale vengono convalidati dalle persone con le quali si comunica e cambiare la chiave primaria può quindi necessitare una ricertificazione. Ciò potrebbe risultare difficile e dispendioso se si comunica con molte persone. D'altro canto è opportuno cambiare periodicamente le chiavi di cifratura. Se infatti una chiave viene compromessa, tutti i dati criptati con quella chiave saranno vulnerabili. Cambiando chiave invece, solo i dati cifrati con la sola chiave compromessa saranno rivelabili.

Le sottochiavi e gli User ID possono anche venir cancellati. Per togliere una sottochiave o uno User ID è necessario innanzi tutto selezionarli usando rispettivamente il comando **key** oppure **uid**. Questi comandi funzionano in modo alternato. Per esempio, il comando **key** ***2*** seleziona la seconda sottochiave, mentre invocando nuovamente il comando **key** ***2*** la si deseleziona. Se nessun argomento opzionale viene fornito, tutte le sottochiavi o tutti gli User ID vengono deselezionati. Una volta che lo User ID che si desidera cancellare viene selezionato, il comando **deluid** cancella effettivamente lo User ID dalla propria chiave. In modo analogo, il comando **delkey** cancella tutte le sottochiavi selezionati sia dalla propria chiave pubblica che da quella privata.

Quando si amministra il proprio mazzo di chiavi locali, cancellare delle componenti di chiavi rappresenta un buon modo per ridurre la quantità di dati inutili presenti nelle chiavi pubbliche di altre persone. Cancellare gli User ID e le sottochiavi dalla propria chiave, però, non sempre è saggio in quanto si complica la distribuzione della chiave. Per default, infatti, quando un utente importa una chiave pubblica aggiornata, tale chiave verrà unita alla copia vecchia eventualmente presente nel suo mazzo di chiavi. Le componenti della chiave nuova e di quella vecchia vengono combinate nell'unione e questo effettivamente ripristina ogni componente precedentemente cancellata. Per aggiornare nel modo corretto la chiave, l'utente deve prima cancellare la vecchia copia della chiave pubblica e poi importare la nuova versione. Ciò obera di ulteriore lavoro le persone con le quali si comunica. Inoltre, se si spedisce la propria chiave ad un server di chiavi, l'unione avverrà in ogni caso e chiunque la scarichi non vedrà mai le cancellazioni apportate. Di conseguenza, per aggiornare la propria chiave è meglio revocarne le componenti al posto di cancellarle.

Revocare le componenti di una chiave

Per revocare una sottochiave è necessario prima selezionarla. Una volta selezionata può essere revocata con il comando **revkey**. La chiave è revocata aggiungendo una revoca autofirmata alla chiave stessa. A differenza di quanto accade per l'opzione a linea di comando `--gen-revoke`, l'effetto della revoca di una sottochiave è immediato.

```
Comando> revkey
Vuoi davvero revocare questa chiave? s

Ti serve una passphrase per sbloccare la chiave segreta
dell'utente: "Chloe (giullare) <chloe@cyb.org>"
chiave DSA di 1024 bit, ID B87DBA93, creata il 1999-06-28


pub  1024D/B87DBA93  creata il: 1999-06-28 scade: mai      fiducia: -/u
sub  2048g/B7934539  creata il: 1999-06-28 scade: mai
sub  1792G/4E3160AD  creata il: 1999-06-29 scade: 2000-06-28
rev! la sottochiave è stata revocata il: 1999-06-29
sub   960D/E1F56448  creata il: 1999-06-29 scade: 2000-06-28
(1)  Chloe (giullare) <chloe@cyb.org>
(2)  Chloe (plebeo) <chloe@tel.net>
```

Uno User ID viene revocato in modo diverso. Normalmente uno User ID raccoglie le firme che attestano che lo User ID descrive la persona che effettivamente possiede la chiave associata. In teoria uno User ID descrive una persona per sempre in quanto quella persona non verrà mai cambiata. In pratica, invece, gli elementi che compongono lo User ID, come l'indirizzo di posta elettronica e il commento, possono cambiare nel tempo, così da invalidare lo User ID.

Le specifiche per l'OpenPGP

* *Primo riferimento all'OpenPGP*

non supportano la revoca dello User ID, ma in pratica ciò può essere fatto revocando l'autofirma che accompagna lo User ID. Per i motivi di sicurezza descritti precedentemente, gli altri corrispondenti non si fideranno di uno User ID senza una valida autofirma.

Una firma è revocata mediante il comando **revsig**. Poiché è possibile aver firmato un numero qualsiasi di User ID, l'interfaccia utente richiederà di decidere per ogni firma se essa debba essere revocata o meno.

```
Comando> revsig
```

```
Hai firmato questi user ID:
     Chloe (giullare) <chloe@cyb.org>
   signed by B87DBA93 at 1999-06-28
     Chloe (plebeo) <chloe@tel.net>
   signed by B87DBA93 at 1999-06-28
user ID: "Chloe (giullare) <chloe@cyb.org>"
firmata con la tua chiave B87DBA93 il 1999-06-28
Creare un certificato di revoca per questa firma? (s/N)n
user ID: "Chloe (plebeo) <chloe@tel.net>"
firmata con la tua chiave B87DBA93 il 1999-06-28
Creare un certificato di revoca per questa firma? (s/N)s
Stai per revocare queste firme:
     Chloe (plebeo) <chloe@tel.net>
   firmata da B87DBA93 il 1999-06-28
Creare davvero i certificati di revoca? (s/N)s

Ti serve una passphrase per sbloccare la chiave segreta
dell'utente: "Chloe (giullare) <chloe@cyb.org>"
chiave DSA di 1024 bit, ID B87DBA93, creata il 1999-06-28


pub  1024D/B87DBA93  creata il: 1999-06-28 scade: mai      fiducia: -/u
sub  2048g/B7934539  creata il: 1999-06-28 scade: mai
sub  1792G/4E3160AD  creata il: 1999-06-29 scade: 2000-06-28
rev! la sottochiave è stata revocata il: 1999-06-29
sub   960D/E1F56448  creata il: 1999-06-29 scade: 2000-06-28
(1)  Chloe (giullare) <chloe@cyb.org>
(2)  Chloe (plebeo) <chloe@tel.net>
```

Uno User ID revocato viene indicato dalla firma di revoca che accompagna lo User ID quando vengono listate le firme degli User ID delle chiavi.

```
Comando> check
uid  Chloe (giullare) <chloe@cyb.org>
sig!    B87DBA93 1999-06-28  [autofirma]
uid  Chloe (plebeo) <chloe@tel.net>
rev!    B87DBA93 1999-06-29  [revoca]
sig!    B87DBA93 1999-06-28  [autofirma]
```

Revocando sia le sottochiavi che le autofirme di uno User ID si aggiungono delle autofirme di revoca alla chiave. Poiché le firme vengono aggiunte e nulla viene cancellato, una revoca sarà sempre visibile agli altri quando la propria chiave aggiornata verrà distribuita e unita con le copie precedenti. La revoca, quindi, garantisce che chiunque avrà una copia consistente della vostra chiave.

Aggiornare una data di scadenza di una chiave

La data di scadenza di una chiave può essere aggiornata con il comando **expire** dal menù di modifica delle chiavi. Se non è selezionata nessuna chiave, verrà aggiornato il tempo di scadenza della chiave primaria. Viceversa verrà aggiornata la data della chiave subordintata selezionata.

La data di scadenza di una chiave è associata all'autofirma della chiave stessa. La data viene aggiornata cancellando la vecchia autofirma e aggiungendone una nuova. Poiché gli altri corrispondenti non avranno cancellato la vecchia autofirma, essi vedranno un'ulteriore autofirma sulla chiave quando aggiorneranno la loro copia di chiave. L'ultima autofirma, però, prende la precedenza cosicché tutti i corrispondenti sapranno senza ambiguitià la data di scadenza della chiave.

Convalidare le altre chiavi del proprio mazzo

Nel capitolo 1 è stata fornita una procedura per convalidare le chiavi pubbliche delle persone con le quali si comunica: la chiave di un corrispondente è convalidata controllando personalmente l'impronta digitale della chiave e quindi firmando la sua chiave pubblica con la propria chiave privata. Controllando personalmente l'impronta digitale si può essere certi che la chiave appartiene realmente a quella persona e, poiché la chiave viene poi firmata, si può essere sicuri di riuscire a rilevare ogni tentativo di manomissione futuro. Sfortunatamente questa procedura risulta antipatica quando, per qualche motivo, si debba convalidare un gran numero di chiavi o comunicare con persone che non si conoscono direttamente.

GnuPG risolve questo problema con un meccanismo comunemente conosciuto come *rete della fiducia*[3]. Nel modello della rete della fiducia la responsabilità di convalidare le chiavi pubbliche è delegata a persone di cui ci si fida. Per esempio, si supponga che

- Alice abbia firmato la chiave di Blake e
- Blake abbia firmato quella di Chloe e Dharma.

Se Alice è convinta che Blake sia capace di convalidare propriamente le chiavi che firma, allora Alice può dedurre che le chiavi di Chloe e Dharma sono valide senza dover necessariamente eseguire alcun controllo. Semplicemente lei usa la propria copia convalidata della chiave pubblica di Blake per controllare che le firme di Blake apposte alle chiavi di Chloe e Dharma siano buone. In generale, assumendo che Alice sia completamente convinta che chiunque sia capace di convalidare propriamente le chiavi che firmano, allora ogni chiave firmata da una chiave valida può essa stessa essere considerata valida. La radice è la chiave di Alice che si considera valida per definizione.

Fiducia nel possessore di una chiave

In pratica la fiducia è soggettiva. Per esempio, la chiave di Blake è valida per Alice in quanto è stata lei a firmarla, ma Alice potrebbe non fidarsi di Blake e della sua capacità di convalidare propriamente le chiavi che egli firma. In questo caso Alice non considererebbe valide le chiavi di Chloe e Dharma basandosi solo sulle firme di Blake. Il modello della rete della fiducia tiene in cosiderazione questo carattere soggettivo associando ad ogni chiave pubblica presente nel proprio mazzo un'indicazione di quanto ci si fidi del possessore di quella chiave. Ci sono quattro livelli di fiducia.

sconosciuto

Non c'è nessuna informazione sul giudizio del possessore nella chiave di firma. Le chiavi del proprio mazzo che non siano le proprie hanno inizialmente questo livello di fiducia.

nessuna

Si sa che il possessore non firma opportunamente le chiavi degli altri.

marginale

Il possessore capisce le implicazioni che comporta firmare una chiave ed è capace di convalidare le chiavi propriamente prima di firmarle.

piena

Il possessore ha un'eccellente comprensione di ciò che comporta firmare una chiave e la sua firma su una chiave è tanto valida quanto la propria.

Un livello di fiducia per la chiave è qualcosa che si assegna da soli alla chiave ed è considerata un'informazione privata. Non viene inclusa con la chiave quando questa è esportata; viene perfino salvata separatamente dal proprio mazzo di chiavi in un elenco a sé stante.

L'editor delle chiavi di GnuPG può essere usato per impostare la fiducia che si possiede verso il possessore di una chiave. Il comando è **trust**. In questo esempio Alice modifica la propria fiducia verso Blake e quindi aggiorna il database della fiducia per ricalcolare quali chiavi siano valide con questa sua nuova fiducia verso Blake.

```
alice% gpg --edit-key blake

pub  1024D/8B927C8A  creata il: 1999-07-02 scade: mai      fiducia: q/f
sub  1024g/C19EA233  creata il: 1999-07-02 scade: mai
(1)  Blake (esecutore) <blake@cyb.org>

Comando> trust
pub  1024D/8B927C8A  creata il: 1999-07-02 scade: mai      fiducia: q/f
sub  1024g/C19EA233  creata il: 1999-07-02 scade: mai
(1)  Blake (esecutore) <blake@cyb.org>

Per favore, decidi quanta fiducia hai che questo utente verifichi
correttamente le chiavi di altri utenti (guardando il loro passaporto,
controllando le impronte digitali prese da diverse fonti, etc.)?

 1 = Non lo so
 2 = NON mi fido
 3 = Mi fido marginalmente
 4 = Mi fido completamente
 s = mostrami ulteriori informazioni
 m = torna al menù principale

Cosa hai deciso? 3

pub  1024D/8B927C8A  creata il: 1999-07-02 scade: mai      fiducia: m/f
sub  1024g/C19EA233  creata il: 1999-07-02 scade: mai
(1)  Blake (esecutore) <blake@cyb.org>

Comando> quit
[...]
```

La fiducia nel possessore della chiave e la validità della chiave sono indicate sulla destra quando viene mostrata la chiave. Prima viene la fiducia nella persona e poi la validità della chiave[4]. I quattro livelli di fiducia/validità sono abbreviati con: sconosciuto (`q`), nessuna (`n`), marginale (`m`) e piena (`f`). In questo caso la chiave di Blake è pienamente valida in quanto è stata Alice stessa a firmarla. Inizialmente Alice ha una fiducia non nota nella capacità di Blake di firmare altre chiavi, ma decide di concedergli una fiducia marginale.

Utilizzare la fiducia per convalidare le chiavi

La rete della fiducia permette di usare un algoritmo più elaborato per convalidare una chiave. In precedenza una chiave veniva considerata valida solo se era stata firmata di persona. Ora può venir utilizzato un algoritmo più flessibile: una chiave *K* è considerata valida se soddisfa a due condizioni:

1. è firmata da un numero sufficiente di chiavi valide, cioé
   - è stata firmata di persona, oppure
   - è stata firmata da una chiave di cui ci si fida pienamente, oppure
   - è stata firmata da tre chiavi con fiducia marginale; inoltre
2. il percorso delle chiavi firmate che risale dalla chiave *K* alla propria chiave è al massimo di cinque passi.

La lunghezza del percorso, il numero delle chiavi con fiducia marginale richiesto e il numero di chiavi con fiducia piena possono essere modificati. I numeri dati precedentemente sono quelli preimpostati in GnuPG.

La Figura 3-1 mostra una rete della fiducia con radice Alice. Il grafico illustra chi è stato a firmare la chiave di una data persona. La tabella mostra le chiavi che Alice considera valide basandosi sulla sua fiducia negli altri membri della rete.

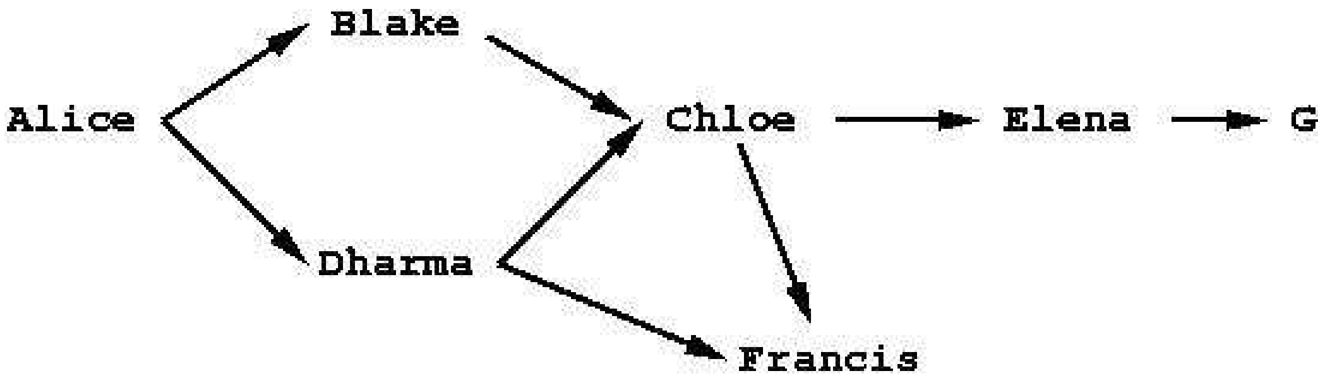
* *Potenziale baco: l'opzione a linea di comando* `--completes-needed` *sembra sia ignorata quando la si combina con* `--update-trustdb`. *Comunque il valore preso è corretto se l'opzione è posta nel file di configurazione.*

Questo esempio assume che siano necessarie due chiavi con fiducia marginale o una chiave con fiducia piena per convalidare un'altra chiave. La lunghezza massima del percorso è tre.

Quando vengono calcolate le chiavi valide nell'esempio, quelle di Blake e Dharma sono sempre considerate pienamente valide in quanto sono state firmate direttamente da Alice. La validità delle altre chiavi dipende dalla fiducia. Nel primo caso, la fiducia in Dharma è piena e ciò implica che le chiavi di Chloe e Francis saranno considerate valide. Nel secondo esempio, la fiducia in Blake e Dharma è marginale. Poiché sono necessarie due chiavi con fiducia marginale per convalidare pienamente un'altra chiave, la chiave di Chloe verrà considerata valida, mentre quella di Francis verrà considerata solo marginalmente valida. In caso in cui sia stata riposta una fiducia marginale in Chloe e Dharma, la chiave di Chloe sarà considerata marginalmente valida in quanto la chiave di Dharma è pienamente valida. D'altro canto, anche la chiave di Francis verrà considerata marginalmente valida in quanto solo una chiave con fiducia piena può venir usata per convalidare altre chiavi e la chiave di Dharma è l'unica chiave pienamente valida che è stata usata per firmare la chiave di Francis. Aggiungendo una fiducia marginale in Blake, la chiave di Chloe diventa pienamente valida e può quindi essere utilizzata per convalidare pienamente la chiave di Francis e marginalmente quella di Elena. Infine, anche riponendo piena fiducia in Blake, Chloe ed Elena, la chiave di Geoff non può essere convalidata in quanto la lunghezza massima del percorso di certificazione è tre, mentre la lunghezza del percorso da Geoff ad Alice è quattro.

Il modello della rete della fiducia è un approccio flessibile al problema dello scambio sicuro di chiavi pubbliche. Esso permette di regolare GnuPG in modo da riflettere l'uso che se ne fa. Da una parte si può insistere su percorsi multipli e brevi che, partendo dalla propria chiave ed arrivando alla chiave *K*, la convalidino. Dall'altra parte si può essere soddisfatti con percorsi più lunghi e magari un solo percorso che colleghi la propria chiave con l'altra chiave *K*. Specificando percorsi multipli e brevi si ottiene una forte garanzia che la chiave *K* appartenga effettivamente a chi si pensa debba appartenere. Il prezzo da pagare, ovviamente, consiste nel fatto che è più difficile convalidare le chiavi in quanto è necessario firmare personalmente più chiavi di quante ne sarebbero necessarie se si accettassero percorsi più brevi ed in numero inferiore.

Figura 3-1. Un'ipotetica rete della fiducia



fiducia		validità	
marginale	**piena**	**marginale**	**piena**
	Dharma		Blake, Chloe, Dharma, Francis
Blake, Dharma		Francis	Blake, Chloe, Dharma
Chloe, Dharma		Chloe, Francis	Blake, Dharma
Blake, Chloe, Dharma		Elena	Blake, Chloe, Dharma, Francis
	Blake, Chloe, Elena		Blake, Chloe, Elena, Francis

Distribuire le chiavi

Idealmente la propria chiave viene distribuita dandola di persona al proprio corrispondente. In pratica, però, le chiavi vengono spesso distribuite per posta elettronica o attraverso qualche altro mezzo di comunicazione elettronico. La distribuzione per e-mail è una buona pratica quando si hanno solo pochi corrispondenti e, qualora si avessero anche numerosi corrispondenti, è possibile distribuirla con mezzi alternativi, ad esempio pubblicandola nella homepage del proprio sito. Ciò risulta però inacettabile se le persone che hanno bisogno di quella chiave non sanno dove trovarla.

Per risolvere questo problema si usano dei server che raccolgono e distribuiscono chiavi pubbliche. Una chiave pubblica depositata su un server viene o aggiunta al database del server o unita alla chiave esistente qualora presente. Quando al server perviene una richiesta, il server consulta il suo database e restitutisce la chiave pubblica cercata se trovata.

Un server di chiavi è prezioso anche quando molte persone firmano in continuazione chiavi di altri. Senza un server di chiavi, dopo che Blake ha firmato la chiave di Alice, egli dovrebbe spedire una copia della chiave pubblica di Alice ad Alice stessa, cosicché Alice possa aggiornare il proprio mazzo di chiavi e distribuire la sua nuova copia a tutti i suoi corrispondenti. Così facendo è responsabilità di Alice e di Blake verso la comunità intera mantenere una stretta rete di fiducia e migliorare così la sicurezza di GnuPG. Questo meccanismo può diventare fastidioso se la firma di chiavi avviene con frequenza.

L'uso di un server di chiavi rende il processo in qualche modo più semplice. Dopo che Blake ha firmato la chiave di Alice, egli la invia al server di chiavi. Il server aggiunge la firma di Blake alla copia della chiave di Alice in suo possesso. Le persone interessate nell'aggiornare la propria copia della chiave di Alice possono quindi consultare di propria iniziativa il server ed ottenere la chiave aggiornata. Non c'è bisogno che Alice venga coinvolta nella distribuzione e, per aggiornare la sua chiave, può semplicemente prelevare le nuove firme apposte interrogando il server.

* *`--keyserver` deve venire prima dell'opzione `--send-key` o `--recv-key`. Questo sembra essere un baco.*

È possibile spedire una o più chiavi ad un server utilizzando l'opzione a linea di comando `--send-keys`. L'opzione richiede uno o più specificatori di chiave e la sua azione consiste nello spedire le chiavi indicate al server di chiavi. Il server al quale inviare le chiavi è specificato con l'opzione a linea di comando `--keyserver`. In modo analogo l'opzione `--recv-keys` viene usata per ottenere delle chiavi da un server, ma l'opzione `--recv-keys` richede che venga specificato un ID di chiave. Nel seguente esempio Alice aggiorna la propria chiave con le nuove firme dal server di chiavi *`certserver.pgp.com`* e successivamente spedisce la propria copia della chiave pubblica di Blake allo stesso server per contribuire con una qualche nuova firma che potrebbe aver aggiunto.

```
alice% gpg --keyserver certserver.pgp.com --recv-key 0xBB7576AC
gpg: richiesta chiave BB7576AC da certserver.pgp.com ...
gpg: chiave BB7576AC: 1 nuova firma

gpg: numero totale esaminato: 1
gpg:          nuove firme: 1
alice% gpg --keyserver certserver.pgp.com --send-key blake@cyb.org
gpg: successo nell'invio a 'certserver.pgp.com' (status=200)
```

Esistono diversi server di chiavi in uso in giro per il mondo. I server principali si sincronizzano a vicenda cosicché è sufficiente scegliere quello più vicino ed usarlo regolarmente per spedire e ricevere chiavi.

Note

1. Dal testo originale *signing key*. Una chiave di firma è quel tipo chiave che può essere usata per apporre una firma.
2. *Self-signing* nel testo originale.
3. *Web of trust* nel testo originale.
4. GnuPG estende il significato della parola "fiducia" intendendola come la fiducia riposta in una persona e quella riposta in una chiave. Ciò può essere fonte di confusione. A volte ci si riferisce alla fiducia in un possessore, usando esplicitamente *fiducia nel possessore*, per distringuerla dalla fiducia in una chiave. In questo manuale, comunque, "fiducia" è usato per indicare la fiducia nel possessore di una chiave e "validità" per indicare la fiducia che si possiede nel fatto che una chiave appartenga ad un essere umano associato all'ID della chiave.

Capitolo 4. Uso quotidiano di GnuPG

GnuPG è uno strumento complesso che coinvolge questioni tecniche, sociali e legali. Da un punto di vista tecnico è stato progettato per venir usato in situazioni con requisiti di sicurezza notevolmente differenti, fatto che ha complicato la gestione delle chiavi. Da un punto di vista sociale, l'uso di GnuPG non è una decisione strettamente personale. Per utilizzare veramente GnuPG entrambe le parti coinvolte nella comunicazione devono usarlo effettivamente. Infine, a partire all'incirca dal 1999, le leggi riguardanti la criptazione digitale, in particolare la legalità dell'uso di GnuPG, variano da stato a stato e sono tutt'ora dibattute dai governi di molte nazioni.

Il presente capitolo affronta queste tematiche. Esso fornisce pratici consigli su come usare GnuPG per soddisfare alle proprie esigenze di sicurezza. Suggerisce anche dei modi per promuovere l'uso di GnuPG come strumento per comunicazioni sicure fra sé ed i propri colleghi, quando questi non lo utilizzano già. Infine, viene sottolineato lo status legale di GnuPG all'interno dello stato corrente delle leggi sulla criptazione nel mondo.

Definire i propri requisiti di sicurezza

GnuPG è uno strumento per proteggere la propria privacy. La propria privacy è protetta se è possibile corrispondere con altri senza che nessuno possa intromettersi e legga i vostri messaggi.

Il modo in cui si dovrebbe usare GnuPG dipende dalla determinazione e dalla ricchezza di risorse di coloro i quali vogliono leggere i vostri messaggi criptati. Un ficcanaso può essere un amministratore di sistema senza scrupoli che legge a caso la posta altrui; può essere una spia industriale che sta provando a rubare i segreti industriali della vostra compagnia; può essere un'agenzia per il rispetto della legge che cerca di accusarvi. L'utilizzo di GnuPG per proteggersi contro spioni casuali sarà differente dall'uso di GnuPG per proteggersi contro avversari determinati. L'obiettivo, in ultima analisi, consiste nel rendere più costoso il recupero dei dati non criptati di quanto valgano i dati stessi.

La personalizzazione del proprio uso di GnuPG orbita attorno a quattro punti:

- la scelta della dimensione della chiave della propria coppia di chiavi pubblica/privata,
- la protezione della propria chiave privata,
- la scelta delle date di scadenza e dell'uso di sottochiavi e
- la gestione della propria rete della fiducia.

Una dimensione della chiave scelta bene protegge dagli attacchi a forza bruta sferrati contro i messaggi criptati. Proteggere la propria chiave privata evita che un malintenzionato possa semplicemente usare la propria chiave privata per decifrare messaggi criptati e firmare messaggi a vostro nome. La corretta gestione della propria rete della fiducia evita che dei malintenzionati fingano di essere le persone con le quali usualmente si comunica. Infine, decidere per questi elementi di personalizzazione rispettando i propri requisiti di sicurezza è il modo in cui si bilancia il lavoro extra dovuto all'uso di GnuPG con la privacy che esso offre.

Scegliere la dimensione della chiave

La scelta della dimensione di una chiave dipende dalla chiave. In OpenPGP una coppia di chiavi pubblica/privata possiede, normalmente, più di una chiave. Al minimo possiede una chiave di firma principale e probabilmente una o più sotto-chiavi addizionali di cifratura. Utilizzando i parametri preimpostati per la generazione di chiavi con GnuPG, la chiave principale sarà di tipo DSA e le sotto-chiavi di tipo ElGamal.

Per lo standard DSA sono previste chiavi fino a 1024 bit. Questo valore non è particolarmente alto data la tecnologia di fattorizzazione di oggi, ma è ciò che lo standard specifica. Senza esitazioni si dovrebbero quindi usare chiavi DSA da 1024 bit.

Le chiavi ElGamal, d'altro canto, possono essere di qualsiasi dimensione. Poiché GnuPG è un sistema a chiave pubblica ibrido, la chiave pubblica viene usata per criptare una chiave di sessione da 128 bit, mentre la chiave privata è usata per decriptarla. Ciò nonostante, la dimensione della chiave influenza la velocità di cifratura e decifratura in quanto il costo di questi algoritmi cresce esponenzialmente con il crescere della dimensione della chiave. Chiavi grandi, inoltre, richiedono più tempo ad essere generate e più spazio per essere salvate. Infine, c'è una riduzione del ritorno di sicurezza che una chiave grande fornisce. Dopo tutto, se la chiave è tanto grande da resistere ad un attacco a forza bruta, uno spione semplicemente cambierebbe metodo e cercherebbe di ottenere i dati in chiaro in un altro modo. Esempi di tali metodi alternativi sono il furto in appartamento e la rapina. 1024 bit è perciò la dimensione di chiave raccomandata. Se si ha sinceramente bisogno di una dimensione di chiave maggiore, allora probabilmente si conoscono già tutte queste problematiche e ci si dovrebbe rivolgere pertanto ad un esperto in sicurezza di dati.

Proteggere la propria chiave privata

Proteggere la propria chiave privata è il lavoro più importante che si deve considerare quando si vuole utilizzare GnuPG correttamente. Se qualcuno ottiene la vostra chiave privata, allora tutti i dati criptati con quella chiave privata possono essere decifrati e firme fatte in vostro nome. Se si perde la propria chiave privata, allora non sarà più possibile decriptare documenti cifrati personalmente in futuro o nel passato e non sarà più possibile fare alcuna firma. La sola perdita della propria chiave privata è un evento catastrofico.

In qualsiasi modo si utilizzi GnuPG si dovrebbe riporre in un posto sicuro il certificato di revoca della propria chiave pubblica e salvare una copia della propria chiave privata su un supporto protetto da scrittura. Per esempio, si potrebbe masterizzare un CD-ROM e riporlo nella cassetta di sicurezza della propria banca in una busta sigillata. In alternativa, si potrebbe salvare il tutto su un dischetto e nasconderlo da qualche parte in casa propria. Qualsiasi cosa si pensi di fare, chiave privata e certificato di revoca dovrebbero essere messi su un supporto che garantisca il mantenimento dei dati per tanto tempo quanto ci si aspetta di utilizzare la chiave e si dovrebbe salvarli con più attenzione di quanta se ne faccia per la propria chiave privata di uso quotidiano.

Nel tentativo di salvaguardare la propria chiave, GnuPG non salva su disco la chiave privata così com'è. Essa viene invece criptata utilizzando un algoritmo di cifratura simmetrico. Questo è il motivo per cui si necessita di una passphrase per poter accedere alla chiave. In questo modo ci sono due barriere che un malintenzionato deve superare per poter accedere alla vostra chiave privata: deve aver effettivamente accessso alla chiave e deve riuscire a superare la cifratura.

Il salvataggio sicuro della propria chiave privata è importante, ma c'è un costo da sostenere. Idealmente si dovrebbe tenere la chiave privata su un disco rimovibile e protetto in scrittura, come un floppy, e lo si dovrebbe utilizzare su una macchina con un unico utente non connessa ad alcuna rete. Ciò può risultare sconveniente o impossibile. Per esempio può accadere che non si possieda un computer ma ci ritrovi costretti ad utilizzarne uno al lavoro o a scuola. Oppure potrebbe succedere che ci si vedrebbe costretti a scollegare il proprio computer da una connessione permanente ogni volta che si voglia usare GnuPG.

Tutto questo non significa che non si possa o non si debba usare GnuPG. Piuttosto si è deciso che i dati da proteggere sono abbastanza importanti da venir cifrati, ma non così importanti da richiedere un impegno ulteriore affinché la prima barriera risulti più robusta. È una questione di scelta.

Una buona passphrase è assolutamente cruciale nell'uso di GnuPG. Qualsiasi malintenzionato che riesca a guadagnare l'accesso alla propria chiave privata deve oltrepassare la cifratura della chiave privata stessa. Invece di indovinare brutalmente la chiave, il malintenzionato cercherà quasi certamente di indovinare la passphrase.

Il motivo per cui si prova prima con la passphrase consiste nel fatto che la maggior parte delle persone sceglie una passphrase più facile da indovinare di una chiave casuale a 128 bit. Se la passphrase è una parola, è molto più economico provare tutte le parole presenti nei dizionari delle lingue del mondo. Anche se i caratteri della parola sono permutati, per esempio "putcomter" invece di "computer", è comunque più semplice provare con delle parole da dizionario a cui sono applicate delle regole di permutazione. Lo stesso problema si applica a citazioni. Generalmente le passphrase basate su frasi del linguaggio naturale sono poveri esempi di passphrase, in quanto esiste poca casualità e molta ridondanza nel linguaggio naturale.

Una buona passphrase è quella passphrase che si riesce a ricordare, ma che altri difficilmente possono indovinare. Essa dovrebbe includere caratteri presi da tutto lo spettro dei caratteri stampabili presenti sulla tastiera. Ciò include i caratteri alfabetici maiuscoli, numeri e caratteri speciali come `}` e `|`. Bisonga essere creativi e spendere un po' di tempo nel considerare la propria passphrase; una buona scelta è importante per assicurare la propria privacy.

Scegliere la data di scadenza e usare le sotto-chiavi.

Per default, quando si crea una nuova coppia di chiavi, vengono generate una chiave di firma principale DSA ed una sotto-chiave di cifratura ElGamal. Ciò risulta inconveniente in quanto i ruoli giocati dalle due chiavi sono differenti e si potrebbe perciò volere che le chiavi abbiano una differente durata. La chiave di firma principale viene usata per fare le firme digitali e raccoglie le firme delle altre persone che hanno confermato la vostra identità. La chiave di cifratura viene usata solo per decifrare i documenti criptati che si ricevono. Tipicamente una firma digitale possiede un tempo di vita lungo, cioè eterno, in quanto non si vuole perdere le firme applicate alla propria chiave per la raccolta delle quali si è lavorato duramente. D'altro canto, la sotto-chiave di cifratura dovrebbe essere cambiata periodicamente per un'ulteriore sicurezza, poiché, se una chiave di cifratura viene violata, il malintenzionato può leggere tutti i documenti cifrati con quella chiave sia nel futuro che nel passato.

Accade quasi sempre che non si desidera avere una data di scadenza per la propria chiave principale. Ci sono due ragioni per le quali si potrebbe scegliere il contrario. La prima è che si intende usare la chiave per un periodo limitato. Per esempio la si potrebbe usare per un evento, come una campagna politica, dopo il quale non sarebbe più utile. Un'altra ragione consiste nel fatto che se si dovesse perdere il controllo della chiave senza possedere un certificato di revoca con il quale revocarla, avendo un data di scadenza sulla chiave principale ci si assicura che alla fine essa cadrà comunque in disuso.

Cambiare le sotto-chiavi di cifratura è semplice ma può risultare sconveniente. Se si genera una nuova coppia di chiavi con una data di scadenza per la sotto-chiave, alla fine tale chiave scadrà. Appena prima della scadenza si aggiungerà allora una nuova sottochiave e si renderà nota la propria chiave pubblica aggiornata. Chi desidera corrispondere con voi dovrà, una volta che la sotto-chiave è scaduta, recuperare la chiave aggiornata in quanto non sarà più in grado di cifrare con la chiave scaduta. Ciò può essere sconveniente a seconda di come si distribuisce la chiave. Fortunatamente, però, non saranno necessarie altre firme in quanto la nuova sotto-chiave sarà stata firmata mediante la vostra chiave di firma principale, la quale, verosimilmente, è già stata convalidata dai vostri corrispondenti.

Questa sconvenienza può o meno valere la sicurezza aggiunta. Proprio come è possibile fare di persona, un malintenzionato può tuttavia leggere tutti i documenti criptati con la sotto-chiave scaduta [se riuscisse ad entrarne in possesso, *ndt*]. Cambiare le sotto-chiavi protegge solo i documenti futuri. Per leggere i documenti cifrati con la nuova sotto-chiave, un malintenzionato avrebbe bisogno di cominciare un nuovo attacco utilizzando una qualsiasi delle tecniche adottate la prima volta nei vostri confronti.

Per concludere, ha senso avere solo una sotto-chiave di cifratura valida in un mazzo. Non c'è nessun guadagno in termini di sicurezza nell'avere due o più sotto-chiavi attive. Può certamente esserci un numero qualsiasi di chiavi scadute in un mazzo, in modo tale che documenti cifrati in passato possano ancora essere decifrati, ma è sufficiente che solo una sotto-chiave sia attiva in un dato momento.

Gestire la propria rete della fiducia

Così come per la protezione della propria chiave, anche la gestione della propria rete della fiducia rappresenta un altro aspetto dell'uso di GnuPG che richiede la ponderazione di sicurezza a sfavore della semplicità di utilizzo. Se si sta utilizzando GnuPG per proteggersi contro spioni e falsari casuali, allora ci si può permettere di avere relativamente una buona fiducia delle firme di altre persone. Viceversa, se si è preoccupati che possa esserci un malintenzionato ben determinato nel voler invadere la propria privacy, allora si dovrebbe avere molta meno fiducia nelle firme altrui e si dovrebbe spendere molto più tempo nella verificarle personale.

Qualsiasi siano i propri requisiti di sicurezza, però, si dovrebbe *sempre prestare attenzione* quando si firmano le chiavi degli altri. È egoistico firmare una chiave possedendo la sola confidenza nella validità della chiave sufficiente a soddisfare i propri requisiti di sicurezza. Altre persone, con richieste di sicurezza più stringenti, potrebbero voler dipendere dalla propria firma. Se non possono dipendere da voi, ciò indebolisce la rete della fiducia e rende più difficoltosa la comunicazione per gli utenti di GnuPG. Si utilizzi la stessa cura nel firmare chiavi che si vorrebbero altri usassero quando si dipende dalle loro firme.

In pratica gestire la propria rete della fiducia si riduce a concedere fiducia agli altri e nell'aggiustare le opzioni `--marginals-needed` e `--completes-needed`. Qualsiasi chiave si firmi personalmente verrà considerata valida, ma, tranne che per piccoli gruppi, non sarebbe pratico firmare di persona la chiave di ogni corrispondente con i quali si desideri comunicare. Si dovrà pertanto concedere della fiducia agli altri.

Risulta probabilmente saggio essere precisi nel concedere la propria fiducia e quindi usare le opzioni per regolare l'attenzione che GnuPG debba porre nella convalida delle chiavi. Come esempio concreto ci si potrebbe fidare totalmente solo di alcuni stretti amici, che si sa essere attenti nella firma delle chiavi, e fidarsi solo marginalmente di tutti gli altri presenti nel proprio mazzo di chiavi. Di coseguenza si potrebbe impostare `--completes-needed` a `1` e `--marginals-needed` a `2`. Se si è più preoccupati per la sicurezza, si potrebbero scegliere dei valori pari a `1` e `3` o `2` e `3` rispettivamente. Se invece si è meno preoccupati di eventuali attacchi alla propria privacy e si desidera solo un po' di ragionevole confidenza circa la validità delle chiavi, si possono impostare i valori `1` e `1`. In generale, se i valori per queste opzioni sono grandi, ciò implica che un numero maggiore di persone dovranno cospirare contro di voi per veder convalidare una chiave che non appartiene effettivamente alla persona alla quale si pensa appartenga.

Costruire la propria rete della fiducia

Desiderare di usare GnuPG da soli non è abbastanza. Per utilizzarlo e comunicare in sicurezza con altri è necessario avere una rete della fiducia. Di primo acchito, perciò, costruire una rete della fiducia può essere scoraggiante. È necessario che le persone con le quali si comunica usino GnuPG[1] e che ci siano abbastanza chiavi che firmino in modo che tutte possano essere considerate valide. Questi non sono problemi tecnici; sono solo problemi sociali. Ciò nonostante è necessario superare queste difficoltà se si vuole usare GnuPG.

Quando si comincia ad usare GnuPG è importante osservare che non c'è bisogno di comunicare in sicurezza con qualsiasi corrispondente. Si può cominiciare con una piccola cerchia di persone, magari voi stessi e altri uno o due che vogliono esercitare il loro diritto alla privacy. Si generano le proprie chiavi e si firmano qulle degli altri. Questa è la vostra rete della fiducia iniziale. Così facendo si apprezzerà il valore di una piccola e robusta rete della fiducia e ci si comporterà in modo più cauto quando in futuro si farà crescere la propria rete della fiducia.

Oltre alle persone presenti nella propria rete della fiducia, si potrebbe voler comunicare in sicurezza con altre persone che usano GnuPG. Così facendo, però, si potrebbe provare un senso di stranezza, per due motivi: non sempre si sa quando qualcuno usa o vorrebbe usare GnuPG; se si sa chi lo usa, si potrebbero comunque avere dei problemi nel convalidare le loro chiavi. Il primo motivo capita perché le persone non sempre avvertono quando usano GnuPG. Il modo ci cambiare questo comportamento consiste nell'essere di esempio e avvertire sempre che si utilizza GnuPG. Ci sono almeno tre modi per fare questo: si può firmare i messaggi di posta che si spediscono o si postano; si può mettere la chiave pubblica sulla propria home page; se si mette la propria chiave su un server di chiavi, si può pensare di mettere il proprio ID di chiave nella firma in fondo ai propri messaggi. Se si reclamizza la propria chiave, si fa in modo che sia molto più accettabile per altri publicizzare la loro. Inoltre si fa in modo che altre persone cominicino a comunicare con voi in sicurezza più facilmente, in quanto si è presa l'iniziativa e si chiarito il fatto che si utilizza GnuPG.

La convalida delle chiavi è più difficile. Se non si conosce direttamente la persona della quale si vuole firmare la chiave, allora non è possibile firmare la chiave. Ci si deve affidare alle firme di altri e sperare di trovare una catena di firme che porti dalla chiave in questione fino a se stessi. Per avere una possibilità di trovare una catena, si deve prendere l'iniziativa e farsi firmare la chiave da altre persone al di fuori dalla propria rete della fiducia iniziale. Un modo efficace per perseguire questo obiettivo consiste nel partecipare agli incontri di firma delle chiavi. Se si sta andando ad una conferenza, si cerchi di avere il tempo per un incontro di firma delle chiavi e, se non se ne dovesse trovare uno, lo si può proporre (http://www.herrons.com/kb2nsx/keysign.html). Si può anche essere più passivi e portare con se le proprie imporonte digitali per uno scambio di chiavi estemporaneo. In una tale situazione la persona alla quale si sono date le proprie impronte digitali le verificherà e quindi firmerà la vostra chiave pubblica una volta tornata a casa.

Si tenga presente, però, che tutto questo è facoltativo. Non c'è nessun obbligo né nel dover reclamizzare la propria chiave, né nel firmare le chiavi di altre persone. La potenza di GnuPG consiste nella sua flessibilità di adattarsi ai vostri requisiti di sicurezza, qualsiasi essi possano essere. La realtà sociale, però, impone che si debba prendere l'iniziativa se si vuole far crescere la propria rete della fiducia e usare GnuPG per quanto più sia possibile.

Usare GnuPG legalmente

Lo status legale del software per la crittografia varia da stato a stato e le leggi riguardanti tale software si stanno evolvendo rapidamente. Bert-Japp Koops (http://cwis.kub.nl/~frw/people/koops/bertjaap.htm) ha un'eccellente manuale di sopravvivenza alle leggi sulla crittografia (http://cwis.kub.nl/~frw/people/koops/lawsurvy.htm) al quale ci si dovrebbe riferire per lo status legale del software di crittografia nel proprio Paese.

Note

1. In questa sezione GnuPG si riferisce tanto all'implementazione GnuPG di OpenPGP quanto ad altre implementazioni, come PGP prodotto della NAI.

Capitolo 5. Argomenti vari

Questo capitolo riguarda argomenti disparati che non hanno trovato posto in altre parti di questo manuale. Quando verranno aggiunti ulteriori paragrafi, essi potrebbero venir raccolti e ripartiti in capitoli a sé stanti. Se si desidera venga trattato un particolare argomento, si prega di mandare un suggerimento o, ancora meglio, si potrebbe volontariamente scrivere una prima bozza riguardante l'argomento suggerito!

Scrivere interfacce utente

Alma Whitten (http://www.cs.cmu.edu/~alma) e Doug Tygar (http://www.cs.berkeley.edu/~tygar) hanno affrontato uno studio (http://reports-archive.adm.cs.cmu.edu/anon/1998/abstracts/98-155.html) sull'interfaccia utente di PGP 5.0 della NAI giungendo alla conclusione che i neo-utenti di PGP lo trovano confuso e frustrante. Nel loro studio sul fattore umano, solo quattro di dodici soggetti testati sono stati in grado di spedire correttamente un messaggio di posta elettronica cifrato ai membri del proprio gruppo, e tre su dodici hanno spedito il messaggio segreto senza criptarlo. Inoltre, la metà dei soggetti sotto test avevano delle basi tecniche.

Questi risultati non sono sorprendenti. PGP 5.0 possiede una graziosa interfaccia utente che si rivela eccellente se si sa già come funziona la crittografia a chiave pubblica e si ha familiarità con il modello di gestione delle chiavi e della rete della fiducia specificato da OpenPGP. Sfortunatamente i nuovi utenti non sanno né di crittografia a chiave pubblica né di gestione di chiavi e l'interfaccia utente può aiutare ben poco.

È certamente opportuno leggere il rapporto di Whitten e Tygar se si sta per scrivere un'interfaccia utente. Vengono infatti raccolti e descritti in dettaglio i commenti specifici forniti da ognuno dei soggetti sotto test. Per esempio, si è riscontrato che una maggioranza dei soggetti riteneva necessario cifrare il messaggio da spedire alle altre persone con la chiave pubblica del soggetto stesso. Se ci si riflette un attimo si vedrà che è un errore facile da commettere. In generale gli utenti alle prime armi hanno difficoltà nel comprendere i differenti ruoli giocati dalla chiave pubblica e da quella privata durante l'uso di GnuPG. Come progettista di interfacce utente, si dovrebbe cercare sempre di chiarire quale delle due chiavi si sta usando. Si potrebbero usare dei wizard o altre comuni tecniche GUI per cercare di guidare l'utente attraverso i procedimenti più comuni, come la generazione di chiavi, dove extra passaggi, quali la generazione di un certificato di revoca per una chiave e la creazione di una copia di sicurezza, sono tutto fuorché essenziali nell'uso corretto di GnuPG. Altri commenti derivanti dallo studio sono riportati qui sotto.

- La sicurezza è normalmente un obiettivo secondario; le persone vogliono spedire messaggi, navigare e così via. Non si parta dal presupposto che gli utenti sono motivati alla lettura di manuali o alla ricerca di controlli di sicurezza.
- La sicurezza di un computer connesso in rete è forte solo quanto lo è la più debole delle parti. Gli utenti hanno bisogno di essere guidati in tutti gli aspetti della loro sicurezza e non lasciati da soli in una esplorazione casuale, come si potrebbero fare con un word processor o un foglio elettronico.
- Si utilizzino in modo consistente gli stessi termini per le stesse azioni. Non si alternino l'uso di sinonimi come "criptatura" e "cifratura".
- Si semplifichi la videata per gli utenti inesperti. Troppe informazioni nascondono le informazioni più importanti. Una configurazione della videata iniziale si potrebbe concentrare sul fornire all'utente il

corretto modello della relazione fra chiavi pubbliche e private e una chiara comprensione delle funzioni per l'acquisizione e la distribuzione delle chiavi.

La progettazione di un'interfaccia utente efficace per la gestione delle chiavi è ancora più complicata. Il modello della rete della fiducia di OpenPGP è sfortunatamente alquanto ottuso. Per esempio, le specifiche impongono tre livelli arbitrari di fiducia per l'utente: nessuno, marginale e completo. Tutti i gradi di fiducia percepiti dall'utente devono ricadere all'interno di uno di questi tre angusti livelli. L'algoritmo di convalida delle chiavi risulta inoltre difficile da comprendere per chi non è un appassionato di computer, in particolare le nozioni di "necessità marginale" e "necessità completa". Poiché il modello della rete della fiducia è ben specificato e non può essere modificato, sarà necessario fare del proprio meglio per progettare un'interfaccia utente che aiuti a chiarificare tale modello all'utente. Un sicuro miglioramento, per esempio, si otterrebbe nel generare un diagramma che rappresenti come una chiave è stata validata al momento della richiesta da parte dell'utente. Tra i commenti di un certo rilievo derivanti dal rapporto sono inclusi quelli riportati qui di seguito.

- Gli utenti sono spesso incerti su come e quanto garantire gli accessi.
- Si dia alta priorità nell'assicurarsi che gli utenti capiscano sufficientemente la loro sicurezza in modo da evitare che commettano errori potenzialmente costosi. Questa categoria di errori include la cancellazione accidentale della chiave privata, la pubblicazione accidentale di una chiave, la revoca accidentale di una chiave, il dimenticarsi della passphrase ed il fallimento nell'effettuare una copia di sicurezza del mazzo di chiavi.

Appendice A. GNU Free Documentation License

Version 1.1, March 2000

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Appendice B. GNU Free Documentation License (traduzione italiana)

Questa è semplicemente una traduzione della licenza che accompagna questo documento, e non ha valore legale. Riferirsi alla versione originale in lingua inglese, acclusa in questo documento, per ogni questione di rilievo.

Versione 1.1, Marzo 2000

0. PREAMBOLO

Lo scopo di questa licenza è di rendere un manuale, un testo o altri documenti scritti "liberi" nel senso di assicurare a tutti la libertà effettiva di copiarli e redistribuirli, con o senza modifiche, a fini di lucro o no. In secondo luogo questa licenza prevede per autori ed editori il modo per ottenere il giusto riconoscimento del proprio lavoro, preservandoli dall'essere considerati responsabili per modifiche apportate da altri.

Questa licenza è un "copyleft": ciò vuol dire che i lavori che derivano dal documento originale devono essere ugualmente liberi. È il complemento alla GNU General Public License, che è una licenza di tipo "copyleft" pensata per il software libero.

Abbiamo progettato questa licenza al fine di applicarla alla documentazione del software libero, perché il software libero ha bisogno di documentazione libera: un programma libero dovrebbe accompagnarsi a manuali che forniscano la stessa libertà del software. Ma questa licenza non è limitata alla documentazione del software; può essere utilizzata per ogni testo che tratti un qualsiasi argomento e al di là dell'avvenuta pubblicazione cartacea. Raccomandiamo principalmente questa licenza per opere che abbiano fini didattici o per manuali di consultazione.

1. APPLICABILITÀ E DEFINIZIONI

Questa licenza si applica a qualsiasi manuale o altra opera che contenga una nota messa dal detentore del copyright che dica che si può distribuire nei termini di questa licenza. Con "Documento", in seguito ci si riferisce a qualsiasi manuale o opera. Ogni fruitore è un destinatario della licenza e viene indicato con "voi".

Una "versione modificata" di un documento è ogni opera contenente il documento stesso o parte di esso, sia riprodotto alla lettera che con modifiche, oppure traduzioni in un'altra lingua.

Una "sezione secondaria" è un'appendice cui si fa riferimento o una premessa del documento e riguarda esclusivamente il rapporto dell'editore o dell'autore del documento con l'argomento generale del documento stesso (o argomenti affini) e non contiene nulla che possa essere compreso nell'argomento principale. (Per esempio, se il documento è in parte un manuale di matematica, una sezione secondaria non può contenere spiegazioni di matematica). Il rapporto con l'argomento può essere un tema collegato storicamente con il soggetto principale o con soggetti affini, o essere costituito da argomentazioni legali, commerciali, filosofiche, etiche o politiche pertinenti.

Le "sezioni non modificabili" sono alcune sezioni secondarie i cui titoli sono esplicitamente dichiarati essere sezioni non modificabili, nella nota che indica che il documento è realizzato sotto questa licenza.

I "testi copertina" sono dei brevi brani di testo che sono elencati nella nota che indica che il documento è realizzato sotto questa licenza.

Una copia "trasparente" del documento indica una copia leggibile da un calcolatore, codificata in un formato le cui specifiche sono disponibili pubblicamente, i cui contenuti possono essere visti e modificati direttamente, ora e in futuro, con generici editor di testi o (per immagini composte da pixel) con generici editor di immagini o (per i disegni) con qualche editor di disegni ampiamente diffuso, e la copia deve essere adatta al trattamento per la formattazione o per la conversione in una varietà di formati atti alla successiva formattazione. Una copia fatta in un altro formato di file trasparente il cui markup è stato progettato per intralciare o scoraggiare modifiche future da parte dei lettori non è trasparente. Una copia che non è trasparente è "opaca".

Esempi di formati adatti per copie trasparenti sono l'ASCII puro senza markup, il formato di input per Texinfo, il formato di input per LaTex, SGML o XML accoppiati ad una DTD pubblica e disponibile, e semplice HTML conforme agli standard e progettato per essere modificato manualmente. Formati opachi sono PostScript, PDF, formati proprietari che possono essere letti e modificati solo con word processor proprietari, SGML o XML per cui non è in genere disponibile la DTD o gli strumenti per il trattamento, e HTML generato automaticamente da qualche word processor per il solo output.

La "pagina del titolo" di un libro stampato indica la pagina del titolo stessa, più qualche pagina seguente per quanto necessario a contenere in modo leggibile, il materiale che la licenza prevede che compaia nella pagina del titolo. Per opere in formati in cui non sia contemplata esplicitamente la pagina del titolo, con "pagina del titolo" si intende il testo prossimo al titolo dell'opera, precedente l'inizio del corpo del testo.

2. COPIE ALLA LETTERA

Si può copiare e distribuire il documento con l'ausilio di qualsiasi mezzo, per fini di lucro e non, fornendo per tutte le copie questa licenza, le note sul copyright e l'avviso che questa licenza si applica al documento, e che non si aggiungono altre condizioni al di fuori di quelle della licenza stessa. Non si possono usare misure tecniche per impedire o controllare la lettura o la produzione di copie successive alle copie che si producono o distribuiscono. Però si possono ricavare compensi per le copie fornite. Se si distribuiscono un numero sufficiente di copie si devono seguire anche le condizioni della sezione 3.

Si possono anche prestare copie e con le stesse condizioni sopra menzionate possono essere utilizzate in pubblico.

3. COPIARE IN NOTEVOLI QUANTITÀ

Se si pubblicano a mezzo stampa più di 100 copie del documento, e la nota della licenza indica che esistono uno o più testi copertina, si devono includere nelle copie, in modo chiaro e leggibile, tutti i testi copertina indicati: il testo della prima di copertina in prima di copertina e il testo di quarta di copertina in quarta di copertina. Ambedue devono identificare l'editore che pubblica il documento. La prima di copertina deve presentare il titolo completo con tutte le parole che lo compongono egualmente visibili ed evidenti. Si può aggiungere altro materiale alle copertine. Il copiare con modifiche limitate alle sole copertine, purché si preservino il titolo e le altre condizioni viste in precedenza, è considerato alla stregua di copiare alla lettera.

Se il testo richiesto per le copertine è troppo voluminoso per essere riprodotto in modo leggibile, se ne può mettere una prima parte per quanto ragionevolmente può stare in copertina, e continuare nelle pagine immediatamente seguenti.

Se si pubblicano o distribuiscono copie opache del documento in numero superiore a 100, si deve anche includere una copia trasparente leggibile da un calcolatore per ogni copia o menzionare per ogni copia opaca un indirizzo di una rete di calcolatori pubblicamente accessibile in cui vi sia una copia trasparente completa del documento, spogliato di materiale aggiuntivo, e a cui si possa accedere anonimamente e gratuitamente per scaricare il documento usando i protocolli standard e pubblici generalmente usati. Se si adotta l'ultima opzione, si deve prestare la giusta attenzione, nel momento in cui si inizia la distribuzione in quantità elevata di copie opache, ad assicurarsi che la copia trasparente rimanga accessibile all'indirizzo stabilito fino ad almeno un anno di distanza dall'ultima distribuzione (direttamente o attraverso rivenditori) di quell'edizione al pubblico.

È caldamente consigliato, benché non obbligatorio, contattare l'autore del documento prima di distribuirne un numero considerevole di copie, per metterlo in grado di fornire una versione aggiornata del documento.

4. MODIFICHE

Si possono copiare e distribuire versioni modificate del documento rispettando le condizioni delle precedenti sezioni 2 e 3, purché la versione modificata sia realizzata seguendo scrupolosamente questa stessa licenza, con la versione modificata che svolga il ruolo del "documento", così da estendere la licenza sulla distribuzione e la modifica a chiunque ne possieda una copia. Inoltre nelle versioni modificate si deve:

A. Usare nella pagina del titolo (e nelle copertine se ce ne sono) un titolo diverso da quello del documento, e da quelli di versioni precedenti (che devono essere elencati nella sezione storia del documento ove presenti). Si può usare lo stesso titolo di una versione precedente se l'editore di quella versione originale ne ha dato il permesso.

B. Elencare nella pagina del titolo, come autori, una o più persone o gruppi responsabili in qualità di autori delle modifiche nella versione modificata, insieme ad almeno cinque fra i principali autori del documento (tutti gli autori principali se sono meno di cinque).

C. Dichiarare nella pagina del titolo il nome dell'editore della versione modificata in qualità di editore.

D. Conservare tutte le note sul copyright del documento originale.

E. Aggiungere un'appropriata licenza per le modifiche di seguito alle altre licenze sui copyright.

F. Includere immediatamente dopo la nota di copyright, un avviso di licenza che dia pubblicamente il permesso di usare la versione modificata nei termini di questa licenza, nella forma mostrata nell'addendum alla fine di questo testo.

G. Preservare in questo avviso di licenza l'intera lista di sezioni non modificabili e testi copertina richieste come previsto dalla licenza del documento.

H. Includere una copia non modificata di questa licenza.

I. Conservare la sezione intitolata "Storia", e il suo titolo, e aggiungere a questa un elemento che riporti al minimo il titolo, l'anno, i nuovi autori, e gli editori della versione modificata come figurano nella pagina del titolo. Se non ci sono sezioni intitolate "Storia" nel documento, createne una che riporti il titolo, gli autori, gli editori del documento come figurano nella pagina del titolo, quindi aggiungete un elemento che descriva la versione modificata come detto in precedenza.

J. Conservare l'indirizzo in rete riportato nel documento, se c'è, al fine del pubblico accesso ad una copia trasparente, e possibilmente l'indirizzo in rete per le precedenti versioni su cui ci si è basati. Questi possono essere collocati nella sezione "Storia". Si può omettere un indirizzo di rete per un'opera pubblicata almeno quattro anni prima del documento stesso, o se l'originario editore della versione cui ci si riferisce ne dà il permesso.

K. In ogni sezione di "Ringraziamenti" o "Dediche", si conservino il titolo, il senso, il tono della sezione stessa.

L. Si conservino inalterate le sezioni non modificabili del documento, nei propri testi e nei propri titoli. I numeri della sezione o equivalenti non sono considerati parte del titolo della sezione.

M. Si cancelli ogni sezione intitolata "Riconoscimenti". Solo questa sezione può non essere inclusa nella versione modificata.

N. Non si modifichi il titolo di sezioni esistenti come "miglioria" o per creare confusione con i titoli di sezioni non modificabili.

Se la versione modificata comprende nuove sezioni di primaria importanza o appendici che ricadono in "sezioni secondarie", e non contengono materiale copiato dal documento, si ha facoltà di rendere non modificabili quante sezioni si voglia. Per fare ciò si aggiunga il loro titolo alla lista delle sezioni immutabili nella nota di copyright della versione modificata. Questi titoli devono essere diversi dai titoli di ogni altra sezione.

Si può aggiungere una sezione intitolata "Riconoscimenti", a patto che non contenga altro che le approvazioni alla versione modificata prodotte da vari soggetti--per esempio, affermazioni di revisione o che il testo è stato approvato da una organizzazione come la definizione normativa di uno standard.

Si può aggiungere un brano fino a cinque parole come Testo Copertina, e un brano fino a 25 parole come Testo di Retro Copertina, alla fine dell'elenco dei Testi Copertina nella versione modificata. Solamente un brano del Testo Copertina e uno del Testo di Retro Copertina possono essere aggiunti (anche con adattamenti) da ciascuna persona o organizzazione. Se il documento include già un testo copertina per la stessa copertina, precedentemente aggiunto o adattato da voi o dalla stessa organizzazione nel nome della quale si agisce, non se ne può aggiungere un altro, ma si può rimpiazzare il vecchio ottenendo l'esplicita autorizzazione dall'editore precedente che aveva aggiunto il testo copertina.

L'autore/i e l'editore/i del "documento" non ottengono da questa licenza il permesso di usare i propri nomi per pubblicizzare la versione modificata o rivendicare l'approvazione di ogni versione modificata.

5. UNIONE DI DOCUMENTI

Si può unire il documento con altri realizzati sotto questa licenza, seguendo i termini definiti nella precedente sezione 4 per le versioni modificate, a patto che si includa l'insieme di tutte le Sezioni Invarianti di tutti i documenti originali, senza modifiche, e si elenchino tutte come Sezioni Invarianti della sintesi di documenti nella licenza della stessa.

Nella sintesi è necessaria una sola copia di questa licenza, e multiple sezioni invarianti possono essere rimpiazzate da una singola copia se identiche. Se ci sono multiple Sezioni Invarianti con lo stesso nome ma contenuti differenti, si renda unico il titolo di ciascuna sezione aggiungendovi alla fine e fra parentesi, il nome dell'autore o editore della sezione, se noti, o altrimenti un numero distintivo. Si facciano gli stessi aggiustamenti ai titoli delle sezioni nell'elenco delle Sezioni Invarianti nella nota di copiright della sintesi.

Nella sintesi si devono unire le varie sezioni intitolate "storia" nei vari documenti originali di partenza per formare una unica sezione intitolata "storia"; allo stesso modo si unisca ogni sezione intitolata "Ringraziamenti", e ogni sezione intitolata "Dediche". Si devono eliminare tutte le sezioni intitolate "Riconoscimenti".

6. RACCOLTE DI DOCUMENTI

Si può produrre una raccolta che consista del documento e di altri realizzati sotto questa licenza; e rimpiazzare le singole copie di questa licenza nei vari documenti con una sola inclusa nella raccolta, solamente se si seguono le regole fissate da questa licenza per le copie alla lettera come se si applicassero a ciascun documento.

Si può estrarre un singolo documento da una raccolta e distribuirlo individualmente sotto questa licenza, solo se si inserisce una copia di questa licenza nel documento estratto e se si seguono tutte le altre regole fissate da questa licenza per le copie alla lettera del documento.

7. RACCOGLIERE INSIEME A LAVORI INDIPENDENTI

Una raccolta del documento o sue derivazioni con altri documenti o lavori separati o indipendenti, all'interno di o a formare un archivio o un supporto per la distribuzione, non è una "versione modificata" del documento nella sua interezza, se non ci sono copiright per l'intera raccolta. Ciascuna raccolta si chiama allora "aggregato" e questa licenza non si applica agli altri lavori contenuti in essa che ne sono parte, per il solo fatto di essere raccolti insieme, qualora non siano però loro stessi lavori derivati dal documento.

Se le esigenze del Testo Copertina della sezione 3 sono applicabili a queste copie del documento allora, se il documento è inferiore ad un quarto dell'intero aggregato i Testi Copertina del documento possono essere piazzati in copertine che delimitano solo il documento all'interno dell'aggregato. Altrimenti devono apparire nella copertina dell'intero aggregato.

8. TRADUZIONI

La traduzione è considerata un tipo di modifica, e di conseguenza si possono distribuire traduzioni del documento seguendo i termini della sezione 4. Rimpiazzare sezioni non modificabili con traduzioni richiede un particolare permesso da parte dei detentori del diritto d'autore, ma si possono includere traduzioni di una o più sezioni non modificabili in aggiunta alle versioni originali di queste sezioni immutabili. Si può fornire una traduzione della presente licenza a patto che si includa anche l'originale versione inglese di questa licenza. In caso di discordanza fra la traduzione e l'originale inglese di questa licenza la versione originale inglese prevale sempre.

9. TERMINI

Non si può applicare un'altra licenza al documento, copiarlo, modificarlo, o distribuirlo al di fuori dei termini espressamente previsti da questa licenza. Ogni altro tentativo di applicare un'altra licenza al documento, copiarlo, modificarlo, o distribuirlo è deprecato e pone fine automaticamente ai diritti previsti da questa licenza. Comunque, per quanti abbiano ricevuto copie o abbiano diritti coperti da questa licenza, essi non ne cessano se si rimane perfettamente coerenti con quanto previsto dalla stessa.

10. REVISIONI FUTURE DI QUESTA LICENZA

La Free Software Foundation (http://www.gnu.org/fsf/fsf.html) può pubblicare nuove, rivedute versioni della Gnu Free Documentation License volta per volta. Qualche nuova versione potrebbe essere simile nello spirito alla versione attuale ma differire in dettagli per affrontare nuovi problemi e concetti. Si veda http://www.gnu.org/copyleft/ (http://www.gnu.org/copyleft).

Ad ogni versione della licenza viene dato un numero che distingue la versione stessa. Se il documento specifica che si riferisce ad una versione particolare della licenza contraddistinta dal numero o "ogni versione successiva", si ha la possibilità di seguire termini e condizioni sia della versione specificata che di ogni versione successiva pubblicata (non come bozza) dalla Free Software Foundation. Se il documento non specifica un numero di versione particolare di questa licenza, si può scegliere ogni versione pubblicata (non come bozza) dalla Free Software Foundation.

Come usare questa licenza per i vostri documenti

Per applicare questa licenza ad un documento che si è scritto, si includa una copia della licenza nel documento e si inserisca il seguente avviso di copiright appena dopo la pagina del titolo:

> Copyright (c) ANNO VOSTRO NOME.
>
> È garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili ELENCARNE I TITOLI, con i Testi Copertina ELENCO, e con i Testi di Retro Copertina ELENCO. Una copia della licenza è acclusa nella sezione intitolata "GNU Free Documentation License".

Se non ci sono Sezioni non Modificabili, si scriva "senza Sezioni non Modificabili" invece di dire quali sono non modificabili. Se non c'è Testo Copertina, si scriva "nessun Testo Copertina" invece di "il testo Copertina è ELENCO"; e allo stesso modo si operi per il Testo di Retro Copertina.

Se il vostro documento contiene esempi non banali di programma in codice sorgente si raccomanda di realizzare gli esempi contemporaneamente applicandovi anche una licenza di software libero di vostra scelta, come ad esempio la GNU General Public License (http://www.gnu.org/copyleft/gpl.html), al fine di permetterne l'uso come software libero.