

"La Gaceta de Linux... ¡haciendo a Linux un poco más divertido!"

Comunicación Segura en Linux con GnuPG

Por: [Kapil Sharma](#)

Traducción al Español por: [Walter Echarrí](#)
el día 26 de Diciembre 2000, para [La Gaceta de Linux](#).

Resumen

GnuPG es una herramienta para comunicaciones seguras y almacenamiento de datos. Se puede usar para cifrar datos y crear firmas digitales. GnuPG constituye una implementación completa y libre del PGP. Puesto que no utiliza el algoritmo patentado IDEA, puede usarse sin ningún tipo de restricciones. GnuPG usa criptografía de clave pública para que los usuarios puedan comunicarse en forma segura. En un sistema de clave pública, cada usuario posee un par de claves consistente en una clave privada y otra pública. La clave privada del usuario se mantiene en secreto y no necesita revelarse en ningún momento. La clave pública, en cambio, puede darse a cualquier persona con quien el usuario desea comunicarse

Características

- Reemplaza íntegramente al PGP.
- No emplea algoritmos patentados.
- GPLed escrito desde cero.
- Se puede usar como programa filtro.
- Completa implementación del OpenPGP.
- Mejor funcionalidad que la del PGP y algunas mejoras de seguridad del PGP 2.
- Descifra y verifica mensajes del PGP 5.x
- Soporta ElGamal (firma y encriptación), DSA, 3DES, Blowfish, Twofish, CAST5, MD5, SHA-1, RIPE-MD-160 y TIGER.
- Fácil implementación de nuevos algoritmos mediante el uso de módulos adicionales.
- Fuerza al ID del usuario a estar en el formato estándar.
- Soporta fechas de caducidad tanto de claves como de firmas.
- Soporta los idiomas inglés, danés, alemán, esperanto, francés, japonés, italiano, polaco, portugués europeo, portugués brasileño, ruso, español y sueco.
- Sistema de ayuda en línea.
- Receptores optativos de mensajes anónimos.
- Soporte integrado de servidores de claves HKP (www.keys.pgp.net).
- Posee una gran cantidad de interfaces gráficas (GUI)

Puedes encontrar todo el software relacionado con GnuPG en <http://www.gnupg.org/download.html>.

Instalación

Copia el archivo fuente gnupg en el directorio `./usr/local/` o en el que desees instalarlo y posicónate en dicho directorio mediante el comando `cd`.

```
[root@dragon local] tar xvzf gnupg-1.0.4.tar.gz
[root@dragon local]# cd gnupg-1.0.4
[root@dragon gnupg-1.0.4]# ./configure
[root@dragon gnupg-1.0.4]# make
Esto compila todos los archivos fuentes en ejecutables binarios.
[root@dragon gnupg-1.0.4]# make check
Esto ejecutará las autopruebas que vienen con el paquete.
[root@dragon gnupg-1.0.4]# make install
Esto instalará los binarios y los archivos soportados en los lugares apropiados.
[root@dragon gnupg-1.0.4]# strip /usr/bin/gpg
El comando "strip" reducirá el tamaño del binario de "gpg" para un mejor rendimiento.
```

Comandos habituales

1: Generando un par de claves nuevo

Debemos crear un par nuevo de claves por primera vez (una clave pública y otra privada). Esto se logra con la opción `--gen-key` de la línea de comandos

```
Primer paso
[root@dragon /]# gpg --gen-key
gpg (GnuPG) 1.0.2: Copyright (C) 2000 Free Software Foundation, Inc.
Este programa no ofrece NINGUN TIPO DE GARANTIAS.
Se trata de software libre y lo invitamos a redistribuirlo
bajo ciertas condiciones. Ver el archivo COPYING para mayores detalles.
```

```
gpg: /root/.gnupg: directorio creado
gpg: /root/.gnupg/options: nuevas opciones del archivo creado
```

gpg: debe ejecutar nuevamente GnuPG para que lea el archivo con las nuevas opciones

```
Segundo paso
Ejecutar nuevamente GnuPG con el siguiente comando:
[root@dragon /]# gpg --gen-key
gpg (GnuPG) 1.0.2: Copyright (C) 2000 Free Software Foundation, Inc.
Este programa no ofrece NINGUN TIPO DE GARANTIAS.
Se trata de software libre y lo invitamos a redistribuirlo
bajo ciertas condiciones. Ver el archivo COPYING para mayores detalles.
```

```
gpg:/root/.gnupg/secring.gpg: base de claves creada
gpg:/root/.gnupg/pubring.gpg: base de claves creada
```

Por favor, seleccione el tipo de clave que desea:

- (1) DSA y ElGamal (por defecto)
- (2) DSA (solamente firma)
- (4) ElGamal (firma y cifrado)

¿Su selección? 1

El par de claves DSA tendrá 1024 bits.

A punto de generar un nuevo par de claves ELG-E.

La longitud mínima de la clave es de 768 bits

La longitud por defecto es de 1024 bits

La longitud máxima sugerida es de 2048 bits

¿Qué longitud desea? (1024) 2048

¿Realmente necesita una longitud tan grande? y

La longitud solicitada es de 2048 bits

Por favor, especifique durante cuánto tiempo debe ser válida la clave.

0 = la clave no caduca

<n> d = la clave caduca en n días

<n> w = la clave caduca en n semanas

<n> m = la clave caduca en n meses

<n> y = la clave caduca en n años

La clave es válida ¿hasta..? (0) 0

La clave no caduca en absoluto

¿Es correcto (y/n)? y

Necesita un ID de usuario para identificar su clave. El programa creará el ID del usuario de acuerdo con el Nombre Verdadero, los Comentarios y la Dirección electrónica de la siguiente manera:

```
Nombre verdadero: Kapil sharma
Dirección electrónica: kapil@linux4biz.net
Comentarios: consultor Unix/Linux
Ha seleccionado el siguiente ID del usuario:
"Kapil Sharma (consultor Unix/Linux) <kapil@linux4biz.net> "
```

Cambiar (N)ombre, (C)omentarios, (E)mail o (O)kay/(Q)uit (Salir)? o

Necesita una frase de contraseña para proteger su clave secreta.

Ingrese frase de contraseña: [escriba una contraseña]

Es necesario generar gran cantidad de bytes aleatorios. Es una buena idea realizar algún tipo de acción durante la primera generación (usar el teclado, mover el ratón, utilizar los discos) esto brinda una mejor oportunidad al generador de números aleatorios para crear suficiente entropía `+++++++^++++^`
Las claves pública y secreta han sido creadas y firmadas.

A continuación explicaré los datos que se piden durante la generación del par de claves.

- Por favor, seleccione el tipo de clave que desea:

(1) DSA y ElGamal (por defecto)

(2) DSA (solamente firma)

(4) ElGamal (firma y encriptación)

¿Su selección?

GnuPG es capaz de crear diferentes tipos de pares de claves . Existen tres opciones:

El par de claves DSA se emplea únicamente para crear firmas. El par de claves subordinado ElGamal se usa para cifrar.

La opción 2 es similar pero crea sólo el par de claves DSA.

La opción 4[1] crea un único par de claves ElGamal válidas tanto para crear firmas como para cifrar.

Para la mayoría de los usuarios la opción por defecto es la adecuada.

- Debe elegir la longitud de la clave. El tamaño de una clave DSA debe estar comprendida entre los 512 y los 1024 bits. La clave ElGamal puede tener cualquier longitud.

A punto de generar un nuevo par de claves ELG-E.

La longitud mínima de la clave es de 768 bits

La longitud por defecto es de 1024 bits

La longitud máxima sugerida es de 2048 bits

¿Qué longitud desea? (1024)

Existen ventajas y desventajas al elegir una clave muy larga. Las ventajas son: 1) Cuanto más larga es la clave más segura resulta ser contra los ataques de fuerza bruta.

Las desventajas son: 1) el cifrado y descifrado resultarán más lentos conforme aumenta el tamaño de la clave 2) Una longitud de clave grande puede afectar la longitud de la firma

La longitud predefinida de la clave es la adecuada para casi todos los propósitos y no se puede modificar después de haber hecho la selección.

- Por último, debe seleccionar una fecha de caducidad. Si se elige la Opción 1, la fecha de caducidad se usará tanto para los pares de claves ElGamal como para el DSA

Por favor, especifique durante cuánto tiempo debe ser válida la clave.

```
0 = la clave no caduca
<n> d = la clave caduca en n días
<n> w = la clave caduca en n semanas
<n> m = la clave caduca en n meses
<n> y = la clave caduca en n años
La clave es válida ¿hasta..? (0) 0
```

Para la mayoría de los usuarios una clave que no caduca es la adecuada. El tiempo de caducidad debe elegirse con cuidado aunque es posible cambiarlo posteriormente a la creación de la clave. Sin embargo, puede resultar difícil comunicar un cambio a los usuarios que tienen su clave pública.

- Además de los parámetros claves se debe proporcionar el ID de un usuario. Esta información se usa para asociar a una persona determinada la clave creada.

Necesita un ID de usuario para identificar su clave. El programa creará el ID del usuario de acuerdo con el Nombre Verdadero, los Comentarios y la Dirección electrónica de la siguiente manera:
"Kapil Sharma (consultor Linux) <kapil@linux4biz.net> "

Nombre verdadero: *Escriba aquí su nombre*
Dirección electrónica: *Escriba aquí su dirección de correo electrónico*
Comentarios: *Escriba aquí cualquier comentario*

- GnuPG necesita una frase de contraseña para proteger las claves privadas primaria y subordinada que debe conservar en su poder.

Se necesita una frase de contraseña para proteger su clave secreta.

Ingrese frase de contraseña:

No existe límite en la longitud de la frase de contraseña aunque debe elegirse cuidadosamente. Desde el punto de vista de la seguridad, la frase de contraseña para desbloquear la clave privada es uno de los puntos más débiles del GnuPG (y de otros sistemas de encriptación de clave pública) puesto que es la única protección que uno tiene si otro individuo obtiene nuestra clave privada. Idealmente la frase de contraseña no debería contener palabras del diccionario y deberían mezclarse caracteres alfabéticos en mayúsculas y minúsculas así como caracteres no alfabéticos. Una buena frase de contraseña es crucial para el uso seguro del GnuPG.

2: Generando un certificado de revocación

Luego de haber creado el par de claves se debe generar inmediatamente un certificado de revocación para la clave pública primaria mediante la opción --gen-revoke. Si se olvida la frase de contraseña o si la clave privada se ve comprometida o se extravía se puede publicar este certificado de revocación para notificar a los demás que no debe usarse más la clave.

```
[root@dragon /]# gpg --output revoke.asc --gen-revoke mi_clave
```

En este caso mi_clave debe ser un especificador de la clave; ya sea el ID del par de claves primario o cualquier parte del ID del usuario que identifique su par de claves. El certificado generado quedará en el archivo revoke.asc. El certificado no debe guardarse en un lugar donde puedan acceder otras personas pues cualquiera podría publicar el certificado de revocación y, de esta manera, inutilizar la correspondiente clave pública.

3: Listado de claves

Para listar las claves de su archivo de claves públicas usar la opción --list-keys de la línea de comandos.

```
[root@dragon /]# gpg --list-keys
/root/.gnupg/pubring.gpg
```

```
pub 1024D/020C9884 2000-11-09 Kapil Sharma (consultor Unix/Linux) <kapil@linux4biz.net>
sub 2048g/555286CA 2000-11-09
```

4: Exportando una clave pública

Se puede exportar la clave pública a nuestra página personal o a un servidor de claves disponible en Internet o mediante algún otro método. Para enviar la clave pública a un destinatario se debe ante todo exportarla. Para hacer esto se debe usar la opción --export de la línea de comandos que lleva un argumento adicional para identificar la clave pública a exportar.

- Para exportar su clave pública en formato binario, use el siguiente comando:

```
[root@dragon /]# gpg --output kapil.gpg --export kapil@linux4biz.net
```

- Para exportar su clave pública dotándola de una armadura ASCII, use el siguiente comando:

```
[root@dragon /]# gpg --export-armor > kapil-key.asc
Dónde "--export" es para extraer la clave pública de su archivo encriptado de la base de claves, "-armor" es para crear una armadura ASCII para enviarla por correo electrónico o para publicarla en una página de Internet y "> kapil-key.asc" es para poner el resultado en un archivo.
```

- Para exportar y ver su clave pública con una armadura ASCII, use el siguiente comando:

```
[root@dragon /]# gpg --export-armor
----INICIO BLOQUE CLAVE PUBLICA PGP----
Versión: GnuPG v1.0.2 (GNU/Linux)
Comentarios: Para mayor información dirigirse a http://www.gnupg.org
```

```
[...]
-----FIN BLOQUE CLAVE PUBLICA PGP-----
```

5: Importando una clave pública

Una vez creadas las claves, se las puede guardar en la base de claves públicas confiables provenientes de terceros de modo de poder usarlas para futuros cifrados y autenticación de comunicaciones. Se puede agregar nuestra clave pública a nuestra base de claves mediante la opción import.

```
[root@dragon /]# gpg --import <nombre_archivo>
Aquí "nombre_archivo" es el nombre de nuestra clave pública importada.
```

Por ejemplo:

```
[root@dragon /]# gpg --import mandrake.asc
gpg: clave :9B4A4024: clave pública importada
gpg: /root/.gnupg/trustdb.gpg: trustdb creada
gpg: Número total procesado: 1
gpg: importada: 1
```

En el ejemplo precedente hemos importado el archivo de la clave pública "mandrake.asc" desde la compañía Linux Mandrake, que se puede obtener desde el sitio de Internet de Mandrake, a nuestra base de claves

6: Validación de la clave

Una vez importada la clave se la debe validar. Una clave se valida verificando su huella digital y firmándola para certificar que se trata de una clave válida. La huella digital de la clave se puede visualizar rápidamente con la opción --fingerprint de la línea de comandos.

```
[root@dragon /]# gpg --fingerprint <UID>
```

Por ejemplo:

```
[root@dragon /]# gpg --fingerprint mandrake
pub 1024D/9B4A4024 2000-01-06 MandrakeSoft (claves oficiales de MandrakeSoft) <mandrake@mandrakesoft.com>
Huella digital = 63A2 8CBD A7A8 387E 1A53 2C1E 59E7 0DEE 9B4A 4024
sub 1024g/686FF394 2000-01-06
```

En el ejemplo anterior hemos verificado la huella digital de Mandrake. La huella digital de una clave se confirma con el propietario de la misma. Esto se puede realizar personalmente, en forma telefónica o mediante otros medios siempre y cuando uno pueda garantizar que la comunicación es con el verdadero propietario de la clave. Si la huella digital obtenida es la misma que la proporcionada por el propietario de la clave entonces uno puede confiar que se tiene una copia correcta de la misma.

7: Firmando la clave

Después de importar y verificar las claves que ha almacenado en su base de claves públicas, las puede empezar a firmar. El firmar una clave certifica que uno conoce al dueño de la misma. Se deben firmar las claves únicamente cuando uno está 100% seguro de su autenticidad.

- Para firmar una clave para la compañía Mandrake que hemos agregado a nuestra base de claves, utilizar el siguiente comando:

```
[root@dragon /]# gpg --sign-key <UID>
Por ejemplo:
[root@dragon /]# gpg --sign-key <UID>
pub 1024D/9B4A4024 creada: 2000-01-06 caduca: nunca confiabilidad: -/q
sub 1024g/686FF394 creada: 2000-01-06 caduca: nunca
(1) MandrakeSoft (claves oficiales de MandrakeSoft) <mandrake@mandrakesoft.com>
```

```
pub 1024D/9B4A4024 creada: 2000-01-06 caduca: nunca confiabilidad: -/q
Huella digital: 63A2 8CBD A7A8 387E 1A53 2C1E 59E7 0DEE 9B4A 4024
```

```
MandrakeSoft (claves oficiales de MandrakeSoft) <mandrake@mandrakesoft.com>
```

¿Está seguro de querer firmar esta clave con su clave:
"Kapil Sharma (consultor Unix/Linux) <kapil@linux4biz.net>?"

¿Realmente firma? y

Necesita frase de contraseña para desbloquear la clave secreta
para el usuario: "Kapil Sharma (consultor Unix/Linux) <kapil@linux4biz.net> "

clave DSA de 1024-bits, ID 020C9884, creada 2000-11-09

Ingrese frase de contraseña:

8: Verificando las firmas

Una vez firmada la clave puede verificar la lista de firmas y comprobar la presencia de la firma agregada. Cada ID de usuario tendrá una o más firmas propias así como aquella perteneciente a cada usuario que ha validado la clave. Podemos verificar las firmas de las claves mediante la opción "--check-sigs" del pgg:

Por ejemplo:

```
[root@dragon /]# gpg --check-sigs mandrake
pub 1024D/9B4A4024 2000-01-06 MandrakeSoft (claves oficiales de MandrakeSoft) <mandrake@mandrakesoft.com>
sig! 9B4A4024 2000-01-06 MandrakeSoft (claves oficiales de MandrakeSoft) <mandrake@mandrakesoft.com>
sig! 020C9884 2000-11-09 Kapil Sharma (consultor Unix/Linux) <kapil@linux4biz.net>
sub 1024g/686FF394 2000-01-06
sig! 9B4A4024 2000-01-06 MandrakeSoft (claves oficiales de MandrakeSoft) <mandrake@mandrakesoft.com>
```

9: Cifrando y descifrando

El procedimiento para cifrar y descifrar documentos es muy sencillo. Si se desea cifrar un mensaje dirigido a Mandrake, se debe hacer uso de la clave pública de Mandrake y sólo Mandrake podrá descifrar el archivo con su clave privada. Si Mandrake le desea enviar un mensaje, lo cifra usando su clave pública y uno lo descifra con nuestra clave privada.

Para cifrar y firmar datos para el usuario Mandrake que hemos agregado a nuestra base de claves se debe usar el siguiente comando (debe conocerse la clave pública del destinatario):

```
[root@dragon /]# gpg -sear <UID de la clave pública > <archivo>
```

Por ejemplo:

```
[root@dragon /]# gpg -sear Mandrake documento.txt
Necesita una frase de contraseña para desbloquear la clave secreta del
usuario: "Kapil Sharma (consultor Unix/Linux) <kapil@linux4biz.net> "
clave DSA de 1024-bits, ID 020C9884, creada 2000-11-09
```

Ingrese frase de contraseña:

Aquí la "s" significa firmar (signing), la "e" cifrar (encrypting), la "a" crear una armadura ASCII (armored) (los ".asc" están listos par ser enviados por correo electrónico), "r" cifrar el id del usuario y <archivo> hace referencia a los datos que se desean cifrar. Para descifrar los datos, usar el siguiente comando:

```
[root@dragon /]# gpg -d <archivo>
```

Por ejemplo:

```
[root@dragon /]# gpg -d documentforkapil.asc
Necesita una frase de contraseña para desbloquear la clave secreta para el
usuario: "Kapil Sharma (consultor Unix/Linux) <kapil@linux4biz.net> "
clave DSA de 1024-bits, ID 020C9884, creada 2000-11-09
Ingrese frase de contraseña:
```

Aquí el parámetro "d" es para descifrar los datos y <archivo> hace referencia a los datos que se desean descifrar.

[Nota: se debe poseer en la base de claves la clave pública del emisor del mensaje para poder descifrarlo]

10: Verificar la firma

Una vez extraída y exportada la clave pública cualquiera puede comprobar, con ayuda de la opción -verify de GnuPG, si los datos cifrados provenientes de una determinada persona están firmados por la misma persona.

- Para verificar la firma de los datos encriptados usar el siguiente comando:

```
[root@dragon /]# gpg --verify <Datos>
Aquí la opción "--verify" sirve para verificar la firma y la etiqueta "<Datos>" hace referencia a los datos/archivos encriptados que se desean verificar.
```

Algunos usos del programa GnuPG

- 1: Enviar mensajes de correo electrónico cifrados.
- 2: Encriptar archivos y documentos.
- 3: Transmitir archivos cifrados y documentos importantes a través de la red.

Aquí hay una lista de algunas interfaces gráficas y programas para el GnuPG

GPA apunta a ser la interfaz gráfica estándar de GnuPG. Posee una GUI muy agradable.
GnomePGP es una herramienta del escritorio GNOME para el control de GnuPG.
Geheimniss es la interfaz gráfica de GnuPG para KDE.
pgp4pine es un filtro de Pine para gestionar mensajes PGP.
MagicPGP es otro conjunto de scripts para usar GnuPG con Pine.
PinePGP es otro filtro de Pine para GnuPG.

Más información

<http://www.gnupg.org/docs.html>

Conclusión

Cualquier persona que sea consciente del tema de seguridad debe usar GnuPG. Es uno de los mejores programas de código abierto que tiene todas las funciones para cifrar y descifrar datos seguros y que puede usarse sin ningún tipo de restricciones ya que está bajo la Licencia Pública General GNU. Puede emplearse para enviar correo electrónico, archivos y documentos encriptados. También puede usarse para transmitir archivos y documentos importantes de forma segura a través de la red.

Copyright © 2000, Kapil Sharma.
 Licencia de copia <http://www.linuxgazette.com/copying.html>
 Publicado en el número 60 de *Linux Gazette*, Diciembre del 2000