GPG - BÁSICO

GnuPG utiliza criptografía de clave pública para que los usuarios puedan comunicarse de un modo seguro.

En un sistema de claves públicas cada usuario posee un par de claves, compuesto por una clave privada y una clave pública.

Cada usuario debe mantener su clave privada secreta; no debe ser revelada nunca.

La clave pública se puede entregar a cualquier persona con la que el usuario desee comunicarse.

1.- Generar un nuevo par de claves PRIVADA-PÚBLICA

La opción de la línea de órdenes --gen-key se usa para generar un nuevo par de claves.

```
gpg --gen-key
```

Elegir alguna opción que permite cifrar y firmar.

Nos pedirá también Nombre, E-mail y Comentario.

Seguir las instrucciones. Si nos pide mover el ratón lo haremos (el sistema necesita aleatoriedad)

Pedirá una contraseña que habrá que usar cada vez que usemos la clave privada

No hay límite para la longitud de una contraseña, y ésta debe ser escogida con sumo cuidado. Desde un punto de vista de seguridad, la contraseña que desbloquea la clave privada es uno de los puntos más débiles en GnuPG (y en otros sistemas de cifrado de clave pública), ya que es la única protección que tiene el usuario si alguien se apoderara de su clave privada.

Para una contraseña lo ideal es que no se usen palabras de un diccionario, y que se mezclen mayúsculas y minúsculas, dígitos, y otros caracteres. Una buena contraseña es crucial para el uso seguro de GnuPG.

1b.-Generar un certificado de revocación

Después de haber generado un par de claves, el usuario debe, de forma inmediata, generar un certificado de revocación para la clave pública primaria, mediante el uso de la opción --gen-revoke.

Si el usuario olvidara la contraseña, o si su clave privada estuviera en peligro o extraviada, este certificado de revocación podría ser hecho público para notificar a otros usuarios que la clave pública no debe ser usada nunca más.

Una clave pública revocada puede ser usada para verificar firmas hechas por el usuario en el pasado, pero no puede ser usada para cifrar datos. Esto tampoco afecta a la capacidad de descifrar mensajes que hayan sido cifrados con la clave antes de su revocación, siempre y cuando el usuario todavía tenga acceso a la clave privada.

```
gpg --output revocaciondesergioag.asc --gen-revoke sergioinf@hotmail.com
```

2.- Intercambiar clave pública

Para poder comunicarse con otros, el usuario debe intercambiar las claves públicas.

Para VER la lista de las claves en el fichero («anillo») de claves públicas, se puede usar la opción de la línea de órdenes --list-keys.

gpg --list-keys **ó** **gpg -k**

2.a.Exportar una clave pública

Para poder enviar una clave pública, antes hay que exportarla a un archivo.

Para ello se usará la opción de la línea de órdenes --export. Es necesario un argumento adicional para poder identificar la clave pública que se va a exportar.

Hay que usar el identificador de clave o cualquier parte del identificador de usuario para identificar la clave que se desea exportar.

```
gpg --output clavesergio.publica --export sergioinf@hotmail.com
```

La clave se exporta en formato binario, y esto puede no ser conveniente cuando se envía la clave por correo electrónico o se publica en una página web. Por tanto, GnuPG ofrece una opción de la línea de órdenes –armor que fuerza que la salida de la orden sea generada en formato armadura-ASCII

```
gpg --armor --output clavesergiopublica.asc --export sergioinf@hotmail.com
```

2b.Importar una clave pública

Por ejemplo Javier puede añadir la clave pública de Sergio a su anillo de claves públicas

```
javier:~$ gpg --import clavesergio.publica
```

Una vez que la clave haya sido importada habría que validarla, comprobando si ESTÁ FIRMADA Y VIENDO QUIÉN HA FIRMADO, pero no lo vamos a hacer

3.Cifrar y descifrar documentos

3.1.a.Javier va a cifrar un documento para Sergio con la clave pública de Sergio:

```
javier:~$ gpg --output doc1.odt.cps --encrypt --recipient sergioinf@hotmail.com doc1.odt
```

Solo Sergio podrá descifrar usando su clave privada.

Ni siquiera Javier podrá descifrar el archivo que él mismo ha cifrado para Sergio (si quisiera hacerlo debería firmar también con su clave pública)

3.1.b.Sergio, para descifrar el mensaje se usa la opción --decrypt. Para ello es necesario poseer la clave privada para la que el mensaje ha sido cifrado.

```
sergio:~$  gpg --output doc1.odt --decrypt doc1.odt.cps
```

El sistema pedirá a Sergio la contraseña con la que él protege a su clave privada.

3.2.También es posible cifrar documentos usando una clave de cifrado simétrico.

La clave que se usa para el cifrado simétrico deriva de la contraseña dada en el momento de cifrar el documento (no debería ser la misma contraseña que se esté usando para proteger la clave privada)

Para cifrar con una clave simétrica usar la opción --symmetric.

```
gpg --output doc1.odt.sim --symmetric doc1.odt
```
(Pedirá contraseña)

4.- Firmar y verificar firmas

Se genera una firma con la clave privada del firmante. La firma se verifica por medio de la clave pública correspondiente.

La opción de línea de órdenes --sign se usa para generar una firma digital. El documento que se desea firmar es la entrada, y la salida es el documento firmado.

```
sergio:~$ gpg --output doc1.odt.sig --sign doc1.odt
```

El documento se comprime antes de ser firmado, y la salida es en formato binario.

Con un documento con firma digital el usuario puede llevar a cabo dos acciones:

- comprobar sólo la firma: Para comprobar la firma se usa la opción --verify.
- comprobar la firma y recuperar el documento original al mismo tiempo. Para verificar la firma y extraer el documento se usa la opción --decrypt. El documento con la firma es la entrada, y el documento original recuperado es la salida.

```
javier:~$ gpg --verify doc1.odt.sig
javier:~$ gpg --output doc1.odt --decrypt doc1.odt.sig
```

b) Documentos con firmas ASCII

Las firmas digitales suelen usarse a menudo para firmar mensajes de correo electrónicos o en los grupos de noticias. En estas situaciones no se debe comprimir el documento al firmarlo, ya que para aquellos que no dispongan de un sistema para procesarlo sería ininteligible.

```
sergio:~$ gpg --output doc1.odt.ascii-sig --clearsign doc1.odt
```

c) Firmas acompañantes

Un documento firmado tiene una utilidad limitada. Los otros usuarios deben recuperar la versión original del documento de la versión firmada, y aun en el caso de los documento firmados en ASCII, el documento firmado debe ser editado para poder recuperar el original.

Por tanto, existe un tercer método para firmar un documento, que genera una firma acompañante. Para generar una firma acompañante se usa la opción --detach-sig.

```
sergio:~$ gpg --output doc1.odt.solosig --detach-sig doc1.odt
```

Tanto el documento como la firma acompañante son necesarios para poder verificar la firma. La opción --verify se usará para comprobar la firma.

```
javier:~$  gpg --verify doc1.odt.solosig doc1.odt
```

Nota: Nos estará dando un aviso, de que "***sí, todo va bien, pero***" falta la garantía de que la clave pública con la que comprobamos firmas y ciframos pertenezca a quien dice que pertenece. Luego veremos cómo garantizarlo.