Crear claves GpG pública y privada.

Autor Administrator

Nunca habeis querido enviar un mail con datos confidenciales y con miedo a que alguien li viera o os lo interceptara?

O simplemente creeis que los mails son privados y solo el que los envía y quien los recibe deben leerlos?

Pues bien, os voy a mostrar cómo hacer esto, y es tan sencillo como firmar o cifrar los mails con el sistema GnuPG (o GpG).

Las ventajas, que podeis cifrar vuestros mails y solamente quien tenga vuestra clave pública los podrá leer (también se pueden cifrar archivos, fotos, documentos word...etc).

Y para hacerlo solo necesitais lo siguiente: ir al synaptic e instalar GnuPG. Asi de sencillo.

GnuPG usa un sistema de claves públicas lo que quiere decir que cada usuario tiene una clave privada y una clave pública.

La clave privada es la que se usa para desencriptar aquello que nos envían encriptado con nuestra clave pública, La clave privada es una clave que solo ha de conocer el propietario ya que si alguien más la conociese podría desencriptar lo que nos mandan encriptado.

La clave pública es la que se da a la gente para que nos manden cosas encriptadas y usaran para encriptar aquello que nos quieran pasar.

Creación de claves

Lo primero que hay que hacer una vez que se tiene GnuPG instalado es crear nuestra clave pública y privada. Para hacerlo hay usar el comando gpg --gen-key.

Al ser la primera vez que se ejecuta nos crea un directorio en el que guardara el fichero de configuración así como los archivos secring.gpg y pubring.gpg. En el primero se almacenaran las claves privadas y en el segundo las claves públicas.

La primera pregunta que hace es que tipo de clave queremos. Lo normal suele ser seleccionar la primera opción (DSA and ElGamal) que nos permite encriptar y firmar.

La siguiente pregunta es el tamaño de las claves que se puede elegir entre 1024 y 4096 bits. Por defecto se recomienda 2048, a mayor tamaño más segura es la clave. También a mayor tamaño más tiempo lleva encriptar y desencriptar.

La siguiente pregunta es cuanto tiempo de validez queremos que tenga la clave. La periocidad se puede poner que no caduque nunca, que dure ciertos semanas, meses o años. En el caso de poner que caduque al cabo de cierto tiempo habrá que volver a generar las claves y volver a mandar la nueva clave pública a aquellos que usaban la que ha caducado. Por defecto viene la opción 0 que es que no caduque nunca.

Ahora pregunta nuestro nombre y apellidos, dirección de correo y un comentario para la llave. Una vez introducidos todos los datos nos muestra cual es nuestro ID de usuario que lo crea a partir de los datos que le hemos introducido antes. Luego pregunta si queremos cambiar algún dato o si están bien los datos. Si estan correctos respondemos "V" y sigue adelante el proceso.

Por último se pregunta cual va a ser el password para nuestra clave privada. Al introducir el password no se ve nada de lo que se escribe ni se ve avanzar el cursror. Despu´s de introducirla nos vuelve a preguntar el password y si coincide con el primer password comienza la generación de las claves. Cuando se produce el proceso de generación de las claves

es buena idea reproducir mp3, mover el ratón ... para que se generen números aleatorios y se creen antes las claves. Ver claves públicas disponibles

Para ver las claves públicas que tenemos disponibles hay que hacerlo con el comando gpg --list-keys. Esto lo que haces listar las claves que hay disponibles dentro del fichero pubring.gpg.

El identificador de las claves es lo que hallamos metido en el nombre, en el apellido, en la dirección de correo o el número que aparece después del 1024D al hacer gpg --list-keys. Si en algún caso coincide el ID se mostrarán los que coinciden.

Ver claves privadas disponibles

Para ver las claves privadas que tenemos disponibles hay que hacerlo con el comando gpg --list-secret-keys. Esto lo que haces listar las claves que hay disponibles dentro del fichero secring.gpg.

Borrar claves de los anillos

Se llama anillos a los archivos en los que se guardan las claves públicas y las privadas. Generalmente donde se guardan las claves públicas es el archivo pubring.gpg y en el que se guardan las claves secretas secring.gpg. Si se quiere borrar alguna clave primero hay que borrar la clave privada y después la pública. Si se intenta borrar primero la clave pública y esta tiene asociada una clave privada da un mensaje de error.

Para borrar claves privadas se hace con el comando gpg --delete-secret-key ClaveID
Para borrar claves públicas se hace con el comando gpg --delete-key ClaveID

Ver huella de clave

Las claves están identificadas por lo que se llama huella. La huella es una serie de números que se usa para verificar si una clave pertenece realmente al propietario. Si se recive una clave podemos ver cual es su huella y luego pedirle a su propietario que nos diga su huella. Si ambas coinciden la clave es correcta y no ha sido manipulada. Si no fuese igual es que ha sido modificada. La huella es como el md5 que verifica que un archivo no ha sido manipulado.

Exportar claves

Las claves se pueden exportar a ficheros para que las podamos distribuir entre la gente que queremos que nos encripte o firme cosas o bien porque vamos a formatear el equipo y necesitamos salvarlas.

Para exportar la clave publica se hace poniendo gpg --armor --output fichoeDeSalida --export ClaveID

Para exportar la clave privada se haría poniendo gpg --armor --output fichoeDeSalida --export-secret-key ClaveID

Si quisieramos salvar todas las claves que tenemos valdría con copiar los archivos pubring.gpg y secring.gpg y luego cuando vayamos al nuevo equipo poneralas en el directorio de GnuPG.

Importar claves

Si se quiere importar claves nuevas porque por ejemplo hemos formateado el equipo y queremos volver a tener nuestras claves las importamos con el comando gpg --import ClaveID. En el apartado anterior se han salvado las claves publica y privada pues ahora vamos a importarlas. Primero importamos la publica y luego la privada.

Ahora si queremos importar la clave de una amigo pues se haría igual.

Encriptar mensajes

Si se quiere encriptar mensajes se puede hacer poniendo gpg --armor --recipient ClaveID --encrypt mensaje. Si por ejemplo queremos encriptar el archivo a.txt habría que poner gpg --armor --recipient prueba@prueba.com --encrypt a.txt

También se puede encriptar a un fichero en concreto con la opción --output nombreFichero

Si en las opciones no se le pasa el parámetro --armor lo que se encripta lo deja en un archivo de tipo binario. Al poner la opción --armor transforma lo que se encripta en texto ASCII con el mensaje encriptado.

Desencriptar mensajes

Para desencriptar el mensaje que hemos encriptado antes hay que poner gpg --decrypt archivo. Para el caso anterior sería gpg --decrypt a.txt.asc.

Cuando desencriptamos algo se pide la password de nuestra clave para poder desencriptarlo. Para nuestro caso tenemos el archivo a.txt.asc encriptado al desencriptarlo nos deja el archivo a.txt y nos muestra su contenido.

Firmar mensajes

Firmar mensajes sirve para que cuando a alguien le llegue un mensaje que hemos firmado la persona que lo ha recibido verifique con GnuPG que la firma es buena y que entonces hemos sido nosotros quien le ha enviado el mensaje. Por ejemplo vamos a firmar el archivo a.txt para ello se pondría gpg --clearsign a.txt. Esto nos creara el arhivo a.txt.asc con el contenido que se ve en la imagen.

Para firmar algo se pide la contraseña para poder firmarlo. Como se ve en la imagen lo que se ha hecho en el fichero firmado es añadir unas líneas que contienen la firma.

A la hora de fimar si se firma con el parámetro --sign en lugar de --clearsign nos generara un fichero de salida en binario con extensión .gpg. Para validar la firma y ver el contenido hay desencriptarlo con la opción --decrypt.

La firma también se puede hacer que se muestre en un fichero aparte con la opción -b. Esta opción se suele usar para firmar archivos binarios.

Verificar mensajes firmados

Para verificar mensajes firmados se hace poniendo gpg --verify mensaje. Para el caso anterior seria poner gpg --verify a.txt.asc

Si la firma no fuese correcta podríamos ver un mensaje como el siguiente:

FUENTE: lostscene.com