2021

Cifrado simétrico

FRANCISCO JAVIER LÓPEZ CALDERRÓN

UT 6. Criptografía y Sistemas Identificación

P6.1 – Cifrado Simétrico en Linux GPG

Objetivo

Comprender las características de los cifrados de clave simétrica. Se utilizará la herramienta **GnuPG (*GNU Privacy Guard*)** para cifrar documentos y así poder asegurar la **confidencialidad** de dichos documentos.

Consideraciones previas

GPG o Gnu Privacy Guard es una herramienta para cifrado y firmas digitales que utiliza una serie de algoritmos como ElGamal, CAST5, 3DES, AES y Blowfish.

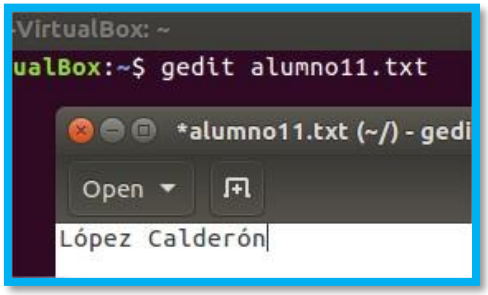
La aplicación GPG genera un archivo de salida (extensión. gpg) en el mismo directorio donde se ubica el archivo origen. El archivo de salida es binario.

Las opciones que vamos a utilizar son:

- **-c**: la orden -c (--symetric), cifra utilizando clave simétrica o privada; para ello nos solicita una contraseña **passphrase**, que se emplea en el cifrado y **descifrado** (clave simétrica). Se genera un archivo binario con extensión. gpg.

Desarrollo de la práctica

1. Crear un archivo **alumnosXX.txt** en el Escritorio que contenga los nombres y los apellidos de los componentes del grupo.
   - Para crear un fichero podemos utilizar **gedit alumnoxx.txt**
   - Podemos utilizar **touch alumnoxx.txt**, etc



Para poder utilizar el cifrado simétrico se requiere el **Fortune-mod**

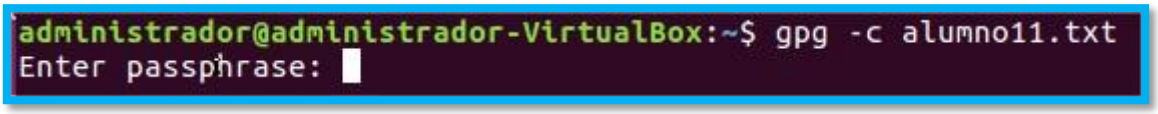
```
root@administrador-VirtualBox:/home/administrador# apt-get install fortune-mod
Reading package lists... Done
Building dependency tree
```

Ahora procederemos con el cifrado simétrico.

2. Cifrar el archivo desde la línea de comandos utilizando el comando.

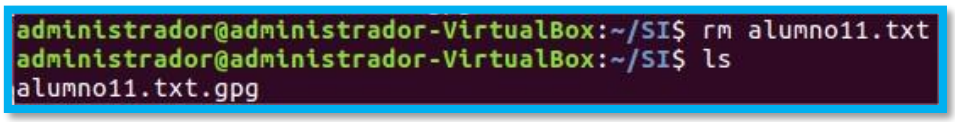
```
gpg -c nombreDelArchivo
```

Se pedirá una contraseña y la confirmación de esta.



Podemos comprobar que se ha generado un archivo simétrico y cifrado terminado en ".gpg"

3. Para descifrar el archivo y escribir la salida en un archivo nuevo ejecutad el siguiente comando:

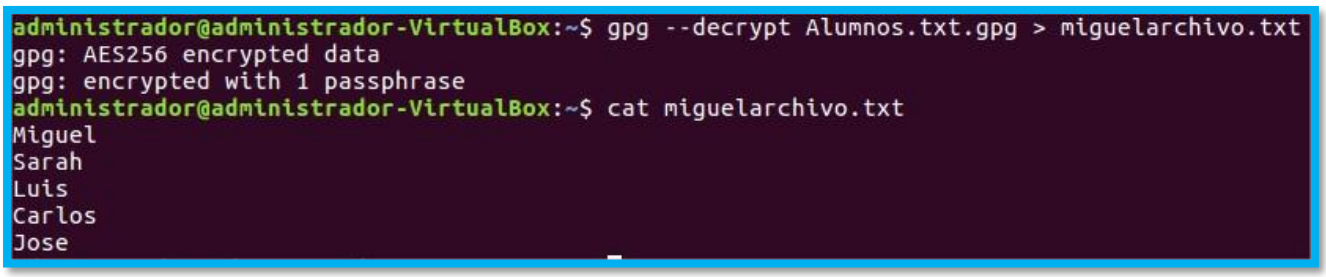


Se hará el descifrado en un archivo nuevo llamado pruebagpg.txt

Introduciendo la contraseña anteriormente utilizada se realizará el traspaso.

4. Enviar el archivo cifrado a un compañero por email y verificar el funcionamiento del proceso de encriptado/desencriptado.

Utilizamos gpg -d o –decrypt para descryptar el archivo del compañero Miguel



5. Los ficheros binarios. gpg no siempre son adecuados. No sirven para incluirlos dentro de un texto (por ejemplo, en un script o un correo electrónico).

   - Para resolverlo utilizamos el parámetro **–a**, que genera un fichero cifrado, pero solo compuesto de caracteres ASCII. Estos ficheros ya no tienen extensión. gpg si no .asc.
     Dentro está el contenido cifrado y alrededor un par de cabeceras informativas
     el comando es: **gpg –a --symmetric NombreFichero**
     Lo vemos ls –l

Otra forma sería utilizando: Por ejemplo: **gpg -a -c nuevo.txt** (otro fichero diferente al anterior)
Creará un fichero en código ascii

Para aplicar cifrado ASCII también se requiere añadir una clave

```
administrador@administrador-VirtualBox:~/SI$ gpg -c -a alumnonuevo11.txt
administrador@administrador-VirtualBox:~/SI$ ls
alumno11.txt       alumnonuevo11.txt      pruebagpg.txt
alumno11.txt.gpg   alumnonuevo11.txt.asc
```

- El fichero .asc ofrece las mismas garantías que el .gpg y se utiliza igual. Para descifrar sería: **gpg --decrypt nuevo.txt.asc**

```
root@alumno-virtual-machine:/home/alumno/docs# gpg nuevo.txt.asc
gpg: datos cifrados CAST5
gpg: el agente gpg no esta disponible en esta sesión
Introduzca contraseña:
```

- La herramienta por defecto utiliza el algoritmo de cifrado CAST5(en pantalla nos lo muestra al descifrar).
- Podemos cambiarlo con el **parámetro cipher-alg**. Por ejemplo, para utilizar AES ejecutaríamos:
- Gpg –a --**symmetric --cipher-alg** AES –o mensaje.aes mensaje
- En este ejemplo se ha utilizado –a para tener el fichero en ASCII

Se requerirá la clave anteriormente creada

```
administrador@administrador-VirtualBox:~/SI$ gpg --decrypt alumnonuevo11.txt.asc
gpg: AES encrypted data
gpg: encrypted with 1 passphrase
```