

Práctica

Cifrado Asimétrico GnuPG

Franco Larrea

2º SMR-A
(Prof. Fernando Albert González)
Instituto IES SAN VICENTE

Índice

Teoría.....	Pag. 3
Tareas y cuestiones.....	Pag. 4-12
Problemas encontrados.....	Pag. 13
Fuentes.....	Pag. 13
Alumnos participantes.....	Pag. 13

Teoría

Para realizar esta practica es conveniente saber algunos conceptos:

Cifrado Asimétrico

¿Que es?

La criptografía asimétrica es el método criptográfico que usa un par de claves para el envío de mensajes.

Funcionamiento

Las dos claves pertenecen a la misma persona que recibirá el mensaje. Una clave es pública y se puede entregar a cualquier persona, la otra clave es privada y el propietario debe guardarla de modo que nadie tenga acceso a ella. Además, los métodos criptográficos garantizan que esa pareja de claves sólo se puede generar una vez, de modo que se puede asumir que no es posible que dos personas hayan obtenido casualmente la misma pareja de claves.

GnuPG

¿Que es?

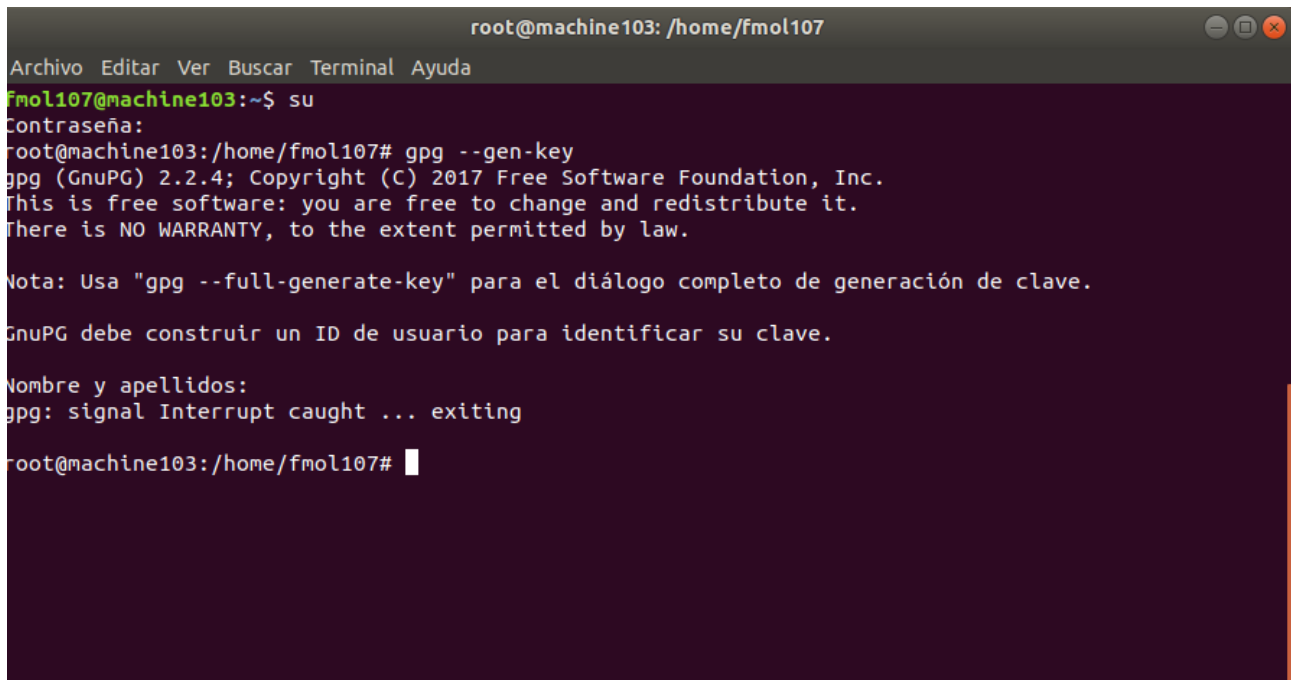
GNU Privacy Guard (GnuPG o GPG) es una herramienta de cifrado y firmas digitales desarrollado por Werner Koch. Es software libre licenciado bajo la GPL.

GPG utiliza el estándar del IETF denominado OpenPGP.

Mediante el comando **gpg** de linux vamos a realizar la comunicación mediante criptografía asimétrica.

Primero tendremos que generar nuestro par de claves. Para ello ejecutaremos desde una terminal:

- **gpg --gen-key.**

A terminal window titled 'root@machine103: /home/fmol107' with a menu bar (Archivo, Editar, Ver, Buscar, Terminal, Ayuda). The prompt is 'fmol107@machine103:~\$'. The user enters 'su', followed by 'gpg --gen-key'. The terminal displays the GnuPG version (2.2.4), copyright (C) 2017 Free Software Foundation, Inc., and a disclaimer: 'This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.' It then shows a note: 'Nota: Usa "gpg --full-generate-key" para el diálogo completo de generación de clave.' and a message: 'GnuPG debe construir un ID de usuario para identificar su clave.' followed by 'Nombre y apellidos:'. The user enters 'gpg: signal Interrupt caught ... exiting'. The prompt returns to 'root@machine103:/home/fmol107#'.

```
root@machine103: /home/fmol107
Archivo Editar Ver Buscar Terminal Ayuda
fmol107@machine103:~$ su
Contraseña:
root@machine103:/home/fmol107# gpg --gen-key
gpg (GnuPG) 2.2.4; Copyright (C) 2017 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Nota: Usa "gpg --full-generate-key" para el diálogo completo de generación de clave.

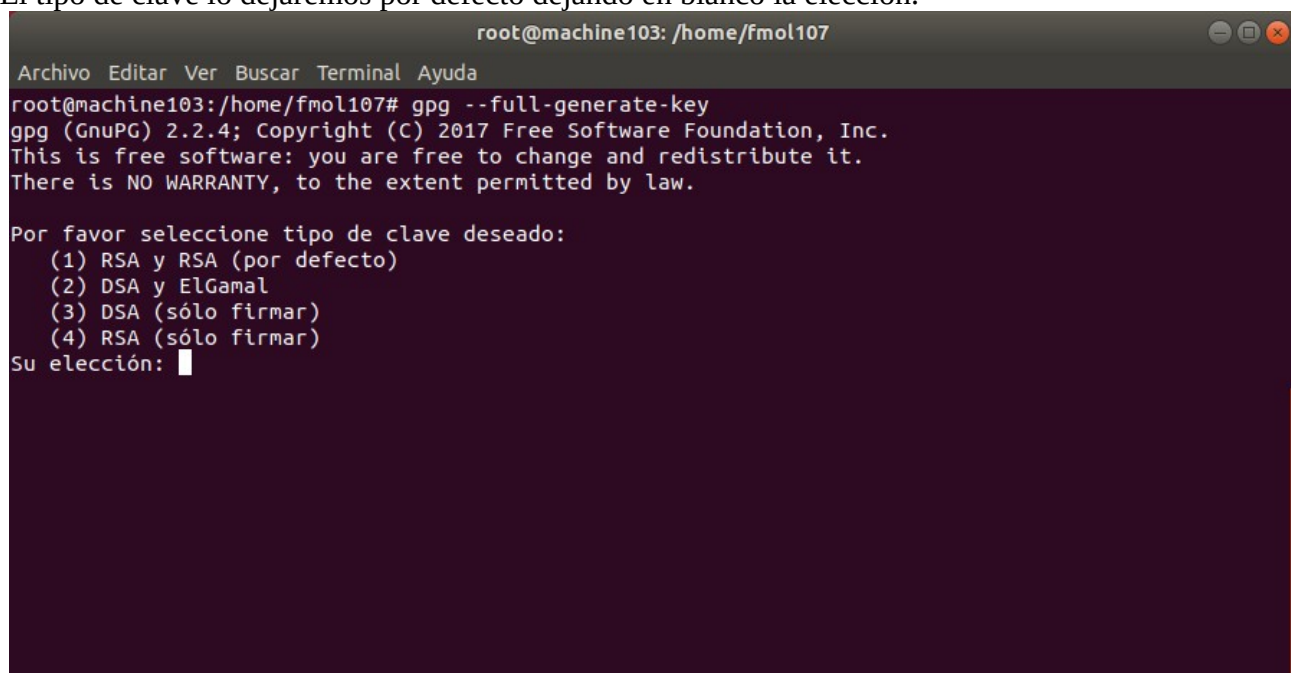
GnuPG debe construir un ID de usuario para identificar su clave.

Nombre y apellidos:
gpg: signal Interrupt caught ... exiting
root@machine103:/home/fmol107#
```

Con el anterior comando no nos salía el dialogo completo de generación de clave. Así que cancelamos y ejecutamos el siguiente comando:

- **gpg --full-generate-key**

El tipo de clave lo dejaremos por defecto dejando en blanco la elección.

A terminal window titled 'root@machine103: /home/fmol107' with a menu bar (Archivo, Editar, Ver, Buscar, Terminal, Ayuda). The prompt is 'root@machine103:/home/fmol107#'. The user enters 'gpg --full-generate-key'. The terminal displays the GnuPG version (2.2.4), copyright (C) 2017 Free Software Foundation, Inc., and a disclaimer: 'This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.' It then prompts: 'Por favor seleccione tipo de clave deseado:' followed by a list: '(1) RSA y RSA (por defecto)', '(2) DSA y ElGamal', '(3) DSA (sólo firmar)', and '(4) RSA (sólo firmar)'. The user enters 'Su elección:'.

```
root@machine103:/home/fmol107# gpg --full-generate-key
gpg (GnuPG) 2.2.4; Copyright (C) 2017 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Por favor seleccione tipo de clave deseado:
  (1) RSA y RSA (por defecto)
  (2) DSA y ElGamal
  (3) DSA (sólo firmar)
  (4) RSA (sólo firmar)
Su elección:
```

La longitud la dejaremos también por defecto.

```
las claves RSA pueden tener entre 1024 y 4096 bits de longitud.  
¿De qué tamaño quiere la clave? (3072)  
El tamaño requerido es de 3072 bits
```

Pondremos un periodo de validez de 1 mes. Para ello ponemos "1m".

```
Por favor, especifique el período de validez de la clave.  
  0 = la clave nunca caduca  
  <n> = 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  
¿Validez de la clave (0)? 1m  
La clave caduca sáb 28 dic 2019 11:29:00 CET  
¿Es correcto? (s/n) s
```

Introduciremos "s" para indicar que SI es correcto.

Ahora escribiremos nuestro nombre y apellidos.

```
GnuPG debe construir un ID de usuario para identificar su clave.  
Nombre y apellidos: Franco Larrea
```

Franco Matias Oscar Larrea

Nos pedirá que pongamos un comentario, yo he puesto "Claves de Franco".

Por ultimo introduciremos nuestro correo electrónico.

```
Dirección de correo electrónico: francolarrea02@gmail.com
```

Así nos quedarían los datos solicitados para la clave.

```
Ha seleccionado este ID de usuario:  
  "Franco Larrea (Claves de Franco) <francolarrea02@gmail.com>"  
¿Cambia (N)ombre, (C)omentario, (D)irección o (V)ale/(S)alir? V
```

Introduciremos "V" para la opción "Vale".

A continuación tendremos que introducir una contraseña. Esta contraseña será nuestra **clave privada**. La cual no deberemos compartir con NADIE.

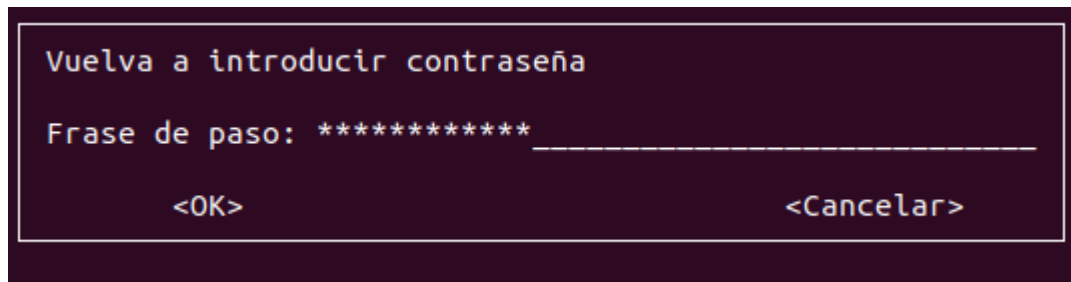
```
Por favor introduzca frase contraseña para  
proteger su nueva clave
```

```
Frase de paso: _____
```

```
<OK>
```

```
<Cancelar>
```

Tendremos que confirmar la clave privada.



El ultimo paso nos pedirá que hagamos cosas aleatorias en nuestro ordenador para recoger suficiente entropía. Esto sirve para generar números aleatorios y crear un par de claves más seguras.

También depende de la fortaleza de tu clave privada, ya que he puesto un contraseña bastante segura y este proceso ha sido bastante breve.

```
Es necesario generar muchos bytes aleatorios. Es una buena idea realizar
alguna otra tarea (trabajar en otra ventana/console, mover el ratón, usar
la red y los discos) durante la generación de números primos. Esto da al
generador de números aleatorios mayor oportunidad de recoger suficiente
entropía.
gpg: /root/.gnupg/trustdb.gpg: se ha creado base de datos de confianza
gpg: clave AE04EC98F51EF2B8 marcada como de confianza absoluta
gpg: creado el directorio '/root/.gnupg/openpgp-revocs.d'
gpg: certificado de revocación guardado como '/root/.gnupg/openpgp-revocs.d/663E0D6C60F133DEC51C61
B0AE04EC98F51EF2B8.rev'
claves pública y secreta creadas y firmadas.

pub   rsa3072 2019-11-28 [SC] [caduca: 2019-12-28]
      663E0D6C60F133DEC51C61B0AE04EC98F51EF2B8
uid           Franco Larrea (Claves Franco Larrea) <francolarrea02@gmail.com>
sub   rsa3072 2019-11-28 [E] [caduca: 2019-12-28]
```

Con **sudo gpg -k** podemos ver las claves que tenemos instaladas.

```
gpg: comprobando base de datos de confianza
gpg: marginals needed: 3  completes needed: 1  trust model: pgp
gpg: nivel: 0  validez: 1  firmada: 0  confianza: 0-, 0q, 0n, 0m, 0f, 1u
gpg: siguiente comprobación de base de datos de confianza el: 2019-12-28
/root/.gnupg/pubring.kbx
-----
pub   rsa3072 2019-11-28 [SC] [caduca: 2019-12-28]
      663E0D6C60F133DEC51C61B0AE04EC98F51EF2B8
uid           [ absoluta ] Franco Larrea (Claves Franco Larrea) <francolarrea02@gmail.com>
sub   rsa3072 2019-11-28 [E] [caduca: 2019-12-28]
```

Ahora vamos a exportar mi clave publica para poder compartirla y que el resto de gente pueda cifrar con ella.

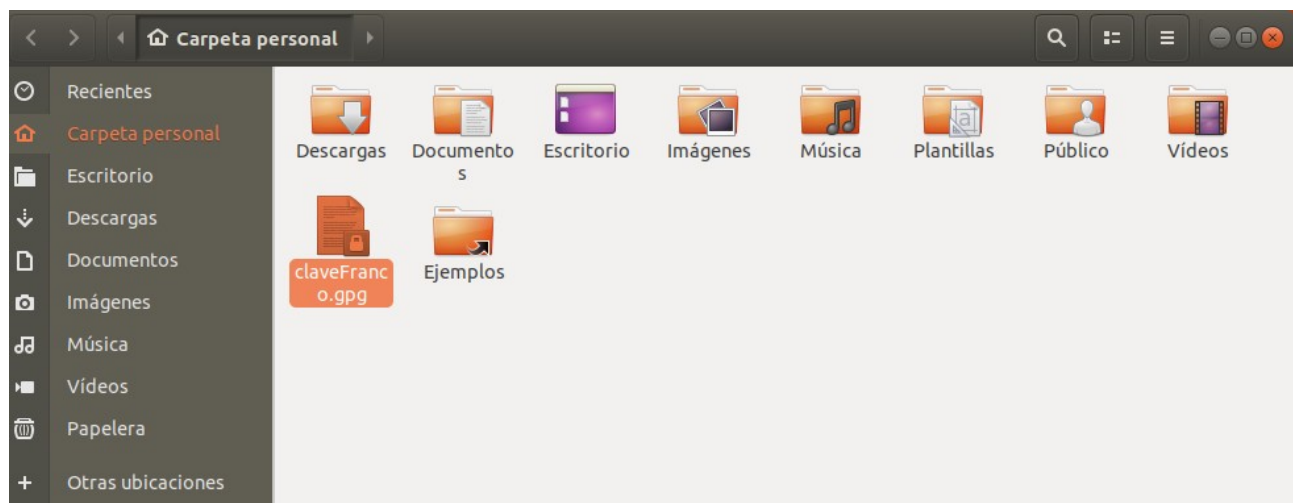
El comando para esto es el siguiente:

- **gpg --output clavepublica.gpg --export identificador**

```
root@machine103:/home/fmol107# gpg --output claveFranco.gpg --export francolarrea02@gmail.com
```

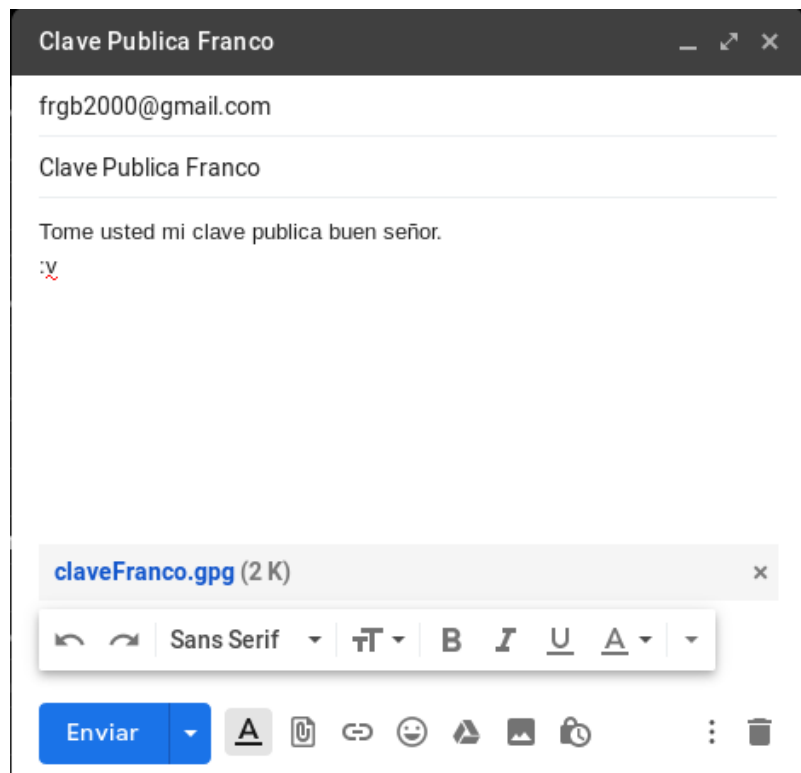
Podemos ver como se ha generado la clave publica “claveFranco.gpg”.

```
root@machine103:/home/fmol107# ls
claveFranco.gpg  Documentos  examples.desktop  Música  Público
Descargas        Escritorio  Imágenes          Plantillas  Vídeos
```



Mandaremos por correo la clave a mi compañero.

Con esta clave el podrá cifrar mensajes los cuales solo yo podré descifrar.



Mi compañero Fran me ha mandado también su clave publica.



La descargaremos.

```
root@machine103:/home/fmol107/Descargas# ls
clavefran
```

Ahora que ya tenemos la clave publica de Fran, la podemos importar. Esto lo realizamos desde el terminal con el siguiente comando:

- **gpg --import clavefran**

```
root@machine103:/home/fmol107/Descargas# gpg --import clavefran
gpg: clave 1A4E22F27CDBF63C: clave pública "Francisco Gómez Benimeli (Práctica)
<frgb2000@gmail.com>" importada
gpg: Cantidad total procesada: 1
gpg:          importadas: 1
```

Con **sudo gpg -k** podemos listar las claves que tenemos instaladas.

```
root@machine103:/home/fmol107/Descargas# gpg -k
/root/.gnupg/pubring.kbx
-----
pub   rsa3072 2019-11-28 [SC] [caduca: 2019-12-28]
      663E0D6C60F133DEC51C61B0AE04EC98F51EF2B8
uid   [ absoluta ] Franco Larrea (Claves Franco Larrea) <francolarrea02@g
mail.com>
sub   rsa3072 2019-11-28 [E] [caduca: 2019-12-28]

pub   rsa3072 2019-12-02 [SC] [caduca: 2020-01-01]
      2098BC9C4ED4F70C1C4579381A4E22F27CDBF63C
uid   [ desconocida ] Francisco Gómez Benimeli (Práctica) <frgb2000@gmail.co
m>
sub   rsa3072 2019-12-02 [E] [caduca: 2020-01-01]
```

Podemos ver que tenemos mi clave creada anteriormente y la recientemente importada de Fran.

Ahora con el programa “nano” vamos a crear un archivo de texto.

```
GNU nano 2.9.3 mensajerandomdeFrancoLarrea.txt
Los psicoactivos son malos Fran.
```

Este mismo archivo vamos a cifrarlo con la clave publica de Fran.

Esto lo realizamos con el siguiente comando:

- **gpg --output nombrearchivoencriptado.gpg --encrypt --recipient identificador archivoadesencriptar**

nombrearchivoencriptado.gpg → Es el nombre que queramos poner al archivo que se creará encriptado.

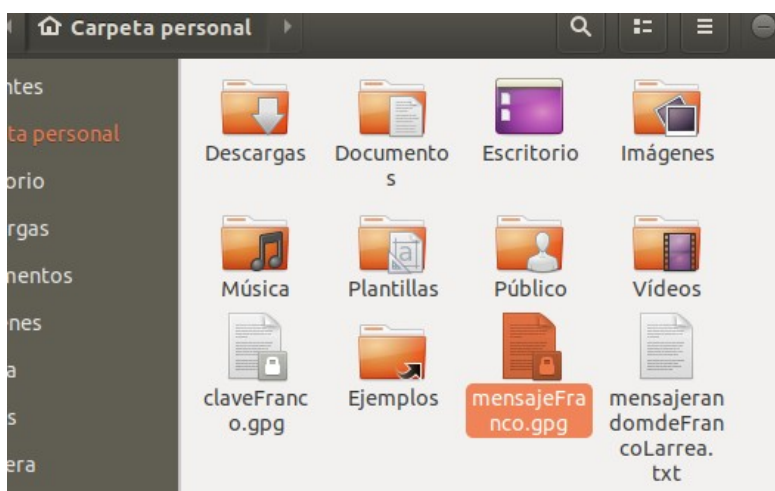
identificador → Clave publica con la que encriptamos el archivo. (La de mi compañero.)

archivoadesencriptar → Archivo que queremos desencriptar.

```
root@machine103:/home/fmol107# sudo gpg --output mensajeFranco.gpg --encrypt --
recipient frgb2000@gmail.com mensajerandomdeFrancoLarrea.txt
gpg: 691E15718550307C: No hay seguridad de que esta clave pertenezca realmente
al usuario que se nombra
sub rsa3072/691E15718550307C 2019-12-02 Francisco Gómez Benimeli (Práctica) <f
rgb2000@gmail.com>
Huella clave primaria: 2098 BC9C 4ED4 F70C 1C45 7938 1A4E 22F2 7CDB F63C
Huella de subclave: 8D5E 5CD2 ED4E D2D9 3F60 3C47 691E 1571 8550 307C

No es seguro que la clave pertenezca a la persona que se nombra en el
identificador de usuario. Si *realmente* sabe lo que está haciendo,
puede contestar sí a la siguiente pregunta.

¿Usar esta clave de todas formas? (s/N) █
```



Aquí podemos ver el archivo cifrado que se ha generado.

Este archivo solo puede ser descifrado por Fran, quien tiene la clave privada.

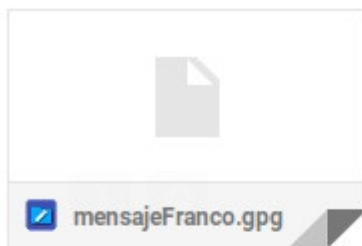
Le mandaremos este archivo cifrado a Fran por correo.

Mensaje encriptado by Franco ➤



Franco Larrea <francolarrea02@gmail.com>
para frgb2000 ▾

Tome este mensaje godofrederik.



Responder



Reenviar

Fran también nos ha mandado un mensaje encriptado.

Mensaje encriptado ➤ Recibidos x



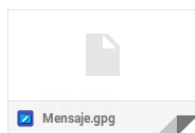
BR

TheFunnyLows
para mí ▾

📧 lun., 2 dic. 12:46 (hace 19 horas) ☆ ↩ ⋮

🌐 inglés ▾ > español ▾ Traducir mensaje

Desactivar para: inglés x



Responder



Reenviar

Descargaremos el mensaje.



```
fmol107@machine103:~$ cd Descargas/  
fmol107@machine103:~/Descargas$ ls  
clavefran  Mensaje.gpg
```

Fran ha encriptado este mensaje con mi clave publica, ahora con mi clave privada solo yo podré descifrar este mensaje y entender su contenido.

Para descifrar el siguiente mensaje utilizaré este comando:

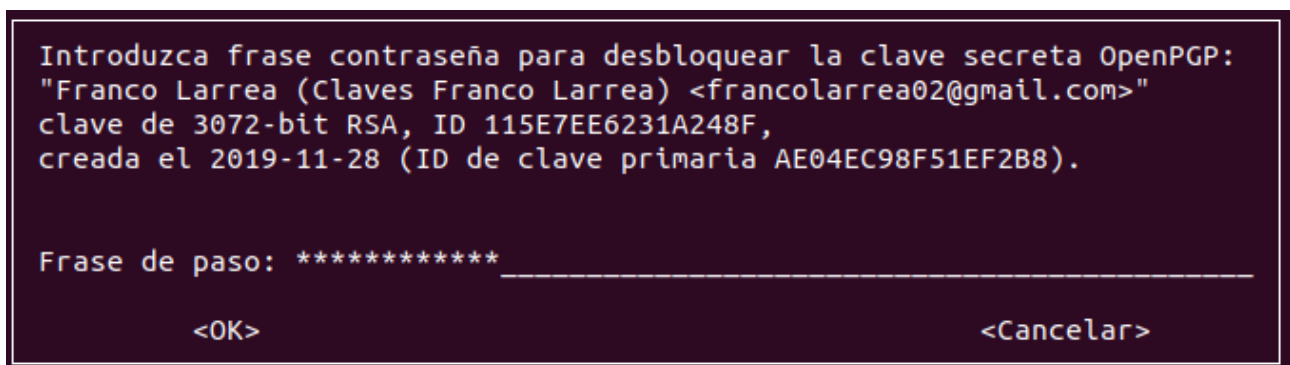
- **gpg --output mensajedesencriptado --decrypt Mensaje.gpg**

mensajedesencriptado → Es el nombre que queramos poner al archivo que se creará desencriptado.

Mensaje.gpg → Es el mensaje que queremos desencriptar.

```
root@machine103:/home/fmol107/Descargas# gpg --output mensajedesencriptado --de  
crypt Mensaje.gpg
```

Al ejecutar el comando nos solicitará una “Frase de paso”. Aquí introduciremos nuestra clave privada.

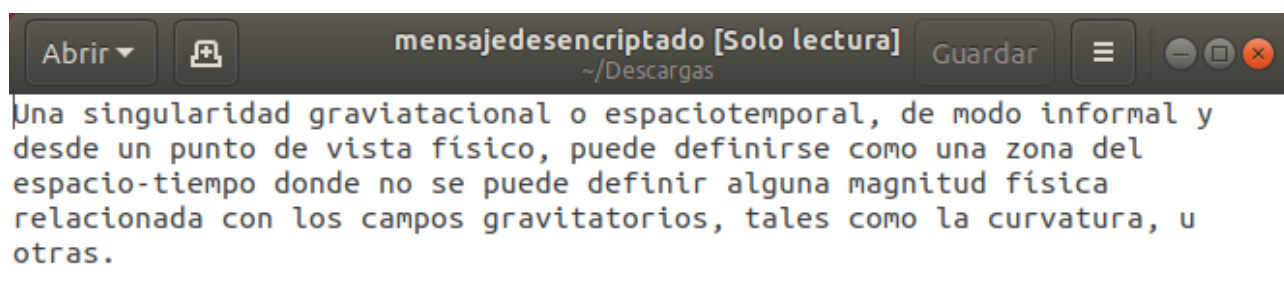


Después de ingresar la clave privada nos creará un archivo descifrado.

```
gpg: cifrado con clave de 3072 bits RSA, ID 115E7EE6231A248F, creada el 2019-11-28  
"Franco Larrea (Claves Franco Larrea) <francolarrea02@gmail.com>"
```



Al abrir el archivo podemos ver que su contenido es entendible.



Problemas encontrados:

- Para que nos salga el dialogo completo de generación de clave deberemos usar el comando **gpg --full-generate-key**, no el comando **gpg --gen-key** .

Fuentes:

- https://es.wikipedia.org/wiki/GNU_Privacy_Guard
- https://es.wikipedia.org/wiki/Criptograf%C3%ADa_asim%C3%A9trica

Alumnos participantes:

- Franco Matias Oscar Larrea
- Francisco Gomez Benimeli