

## PRÁCTICA 2: CRIPTOGRAFÍA EN UNIX

EDUARDO MORA GONZÁLEZ

### Ejercicio 1: Estudiar el manual de la llamada al sistema crypt(3).

```
emorag@emorag:~$ man crypt
```

La llamada crypt es la función de cifrado de contraseña basado en el algoritmo estándar de cifrado de datos.

### Ejercicio 2: Intentar localizar diferentes posibilidades de comprobación de contraseñas en nuestra máquina y tras entender su utilización por medio del man, probarlas en un entorno seguro.

```
emorag@emorag:~$ echo "administrador" | cracklib-check
administrador: OK
emorag@emorag:~$ echo "emorag" | cracklib-check
emorag: Está basada en su nombre de usuario.
emorag@emorag:~$ echo "12345" | cracklib-check
12345: Es demasiado corta.
```

```
emorag@emorag:~$ pwgen 10 -c -n -y -B -v
```

### Ejercicio 3: Instalar de igual forma el paquete mcrypt (Sustitución del comando crypt) y probar su utilización.

```
emorag@emorag:~/Escritorio$ mcrypt prueba
Enter the passphrase (maximum of 512 characters)
Please use a combination of upper and lower case letters and numbers.
Enter passphrase:
Enter passphrase:

File prueba was encrypted.
emorag@emorag:~/Escritorio$
```

### Ejercicio 4. Utilizar la orden crypt para cifrar y descifrar ficheros.

```
emorag@emorag:~/Escritorio$ crypt Clave1 < prueba > prueba.crypt
Unix crypt(1) emulation program using mcrypt(1).

Use crypt -h for more help.
Warning: It is insecure to specify keywords in the command line
Stdin was encrypted.

emorag@emorag:~/Escritorio$ crypt Clave1 <prueba.crypt>salida
```

**Ejercicio 5. Crear un guion en shell que combine el cifrado y la compresión de textos para mejorar la orden crypt.**

```
1  #!/bin/bash
2
3
4  # Comprobar que se ingresa el archivo
5
6  if [ $# -le 0 ];
7  then
8      echo "Hay que introducir un archivo."
9      exit 1
10  fi
11
12
13  mcript $@
14  tar -cvf $@.tar $@.nc
15
16  rm $@.nc
17
18  echo "Archivo ' $@ ' encriptado y comprimido"
```

```
emorag@emorag:~/Escritorio$ ls
ejercicio5.sh prueba
emorag@emorag:~/Escritorio$ ./ejercicio5.sh prueba
Enter the passphrase (maximum of 512 characters)
Please use a combination of upper and lower case letters and numbers.
Enter passphrase:
Enter passphrase:

File prueba was encrypted.
prueba.nc
Archivo ' prueba ' encriptado y comprimido
emorag@emorag:~/Escritorio$ ls
ejercicio5.sh prueba prueba.tar
emorag@emorag:~/Escritorio$
```

**Ejercicio 6. Crear un directorio llamado .pgp en el directorio \$HOME y asegurarse de que están instalados todos los archivos y librerías relacionados con pgp (YAST2 ... buscar ... pgp).**

```
emorag@emorag:/home$ sudo mkdir .pgp
emorag@emorag:/home$ ls
emorag
emorag@emorag:/home$ ls -la
total 16
drwxr-xr-x  4 root    root    4096 abr 24 11:08 .
drwxr-xr-x 22 root    root    4096 abr 23 10:02 ..
drwxr-xr-x 15 emorag  emorag  4096 abr 24 10:59 emorag
drwxr-xr-x  2 root    root    4096 abr 24 11:08 .pgp
emorag@emorag:/home$
```

**Ejercicio 7. Definir una variable PGPPATH apuntando al directorio .pgp**

```
emorag@emorag:/home$ PGPPATH=/home/.pgp/
emorag@emorag:/home$ $PGPPATH/
bash: /home/.pgp//: Es un directorio
emorag@emorag:/home$
```





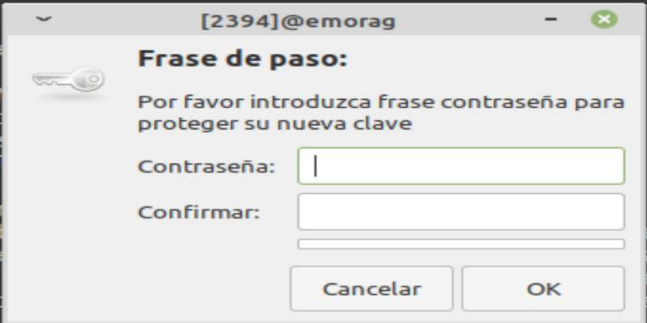
**Ejercicio 13. Consultar las opciones -kg en el manual de pgp y utilizarlas para generar tus claves.**

```
emorag@emorag:~/Escritorio$ pgp -kg
pgp (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 "pgp --full-generate-key" para el diálogo completo de generación
ave.

GnuPG debe construir un ID de usuario.
Nombre y apellidos: Eduardo Mora
Dirección de correo electrónico: emg1015@alu.ubu.es
Ha seleccionado este ID de usuario: "Eduardo Mora <emg1015@alu.ubu.es>"
¿Cambia (N)ombre, (D)irección de correo electrónico?
Es necesario generar muchos bytes aleatorios. Es una buena idea realizar
alguna otra tarea (trabajar en otra ventana/consola, 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.

```



```
Es necesario generar muchos bytes aleatorios. Es una buena idea realizar
alguna otra tarea (trabajar en otra ventana/consola, 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.
pgp: clave 533022DA4CBD3620 marcada como de confianza absoluta
pgp: creado el directorio '/home/emorag/.gnupg/openpgp-revocs.d'
pgp: certificado de revocación guardado como '/home/emorag/.gnupg/openpgp-revocs
.d/5E4C5B4A7F5ABFE952056530533022DA4CBD3620.rev'
claves pública y secreta creadas y firmadas.

pub   rsa3072 2020-04-27 [SC] [caduca: 2022-04-27]
      5E4C5B4A7F5ABFE952056530533022DA4CBD3620
uid           Eduardo Mora <emg1015@alu.ubu.es>
sub   rsa3072 2020-04-27 [E] [caduca: 2022-04-27]

emorag@emorag:~/Escritorio$
```

**Ejercicio 15. Escribir un mensaje en un fichero y cifrarlo mediante este método.**

```
emorag@emorag:~/Escritorio/practica$ pgp -e fichero.text
No ha especificado un ID de usuario (puede usar "-r")

Destinatarios actuales:

Introduzca ID de usuario. Acabe con una línea vacía: emg1015@alu.ubu.es
pgp: comprobando base de datos de confianza
pgp: marginals needed: 3  completes needed: 1  trust model: pgp
pgp: nivel: 0  validez: 4  firmada: 0  confianza: 0-, 0q, 0n, 0m, 0f, 4u
pgp: siguiente comprobación de base de datos de confianza el: 2022-04-27

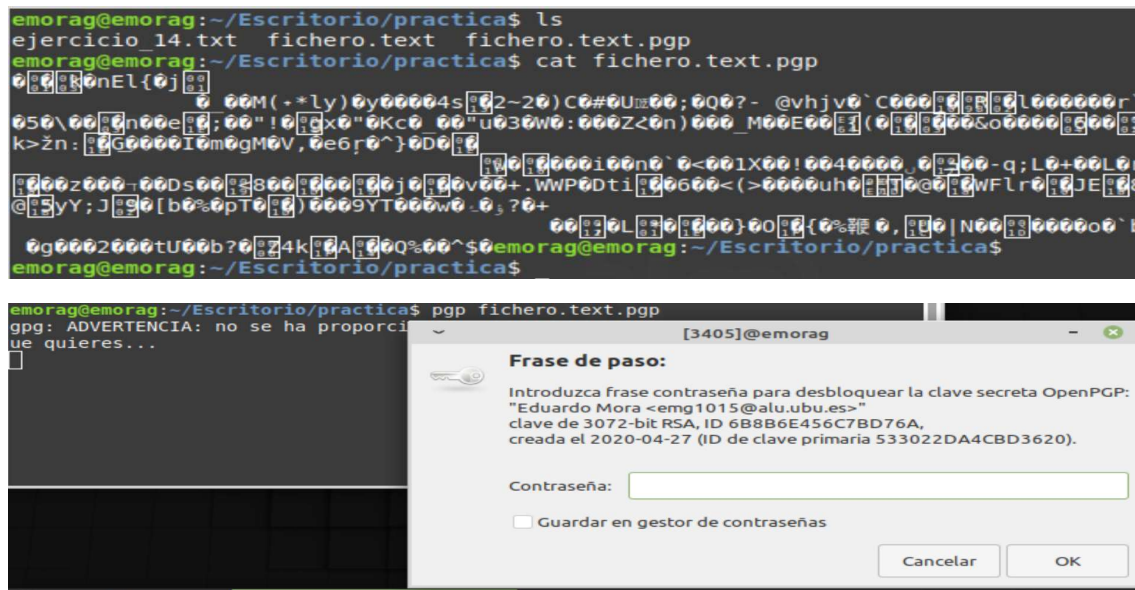
Destinatarios actuales:
rsa3072/6B8B6E456C7BD76A 2020-04-27 "Eduardo Mora <emg1015@alu.ubu.es>"

Introduzca ID de usuario. Acabe con una línea vacía: Eduardo Mora
pgp: omitida: clave pública ya establecida

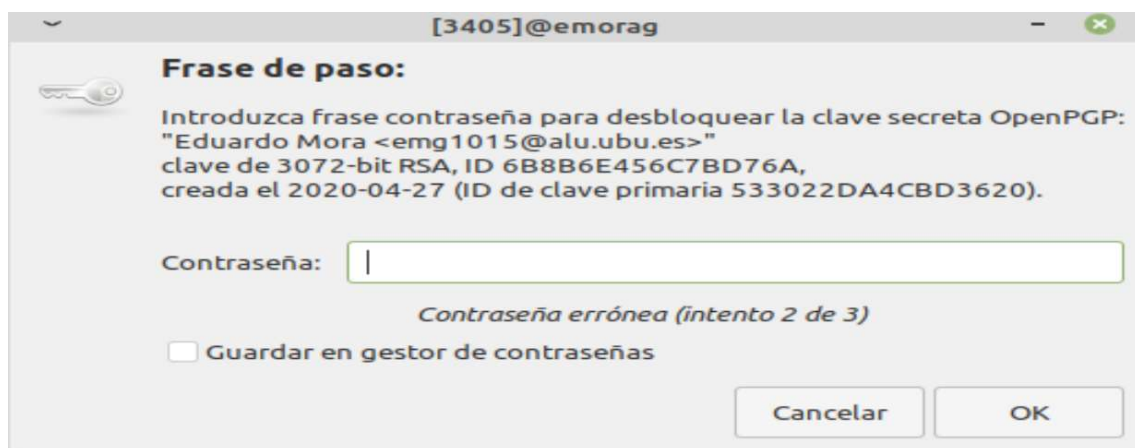
Destinatarios actuales:
rsa3072/6B8B6E456C7BD76A 2020-04-27 "Eduardo Mora <emg1015@alu.ubu.es>"

emorag@emorag:~/Escritorio/practica$ ls
ejercicio_14.txt  fichero.text  fichero.text.pgp
```

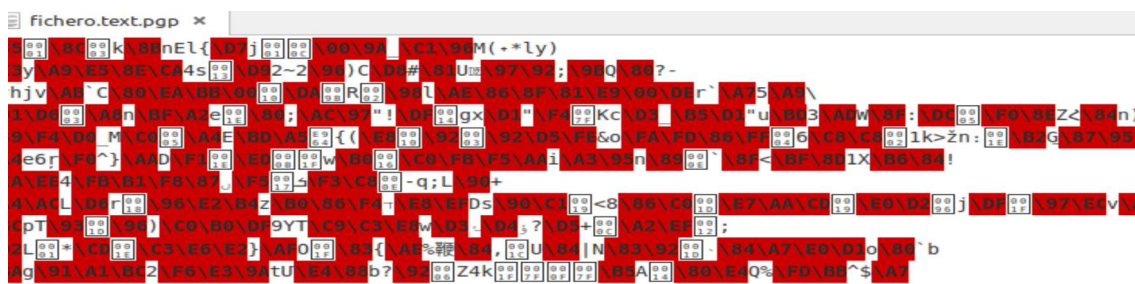
### Ejercicio 16. Comprobar el contenido del fichero resultante y descifrarlo.



### Ejercicio 17. Tratar de descifrar el texto con una frase de paso errónea.



### Ejercicio 18. Modificar el contenido del fichero cifrado e intentar descifrarlo.





```
emorag@emorag:~/Escritorio/practicas$ gpg fichero.text.gpg
gpg: ADVERTENCIA: no se ha proporcionado ninguna orden. Intentando adivinar lo q
ue quieres...
gpg: no se han encontrados datos OpenPGP válidos
gpg: processing message failed: Error desconocido del sistema
WARNING: Can't find the right public key-- can't check signature integrity.
emorag@emorag:~/Escritorio/practicas$
```

```
emorag@emorag:~/Escritorio/practica$ gpg --output clave.gpg --export emg1015@alu.ubu.es
emorag@emorag:~/Escritorio/practica$ ls
clave.gpg  ejercicio 14.txt  fichero.text  fichero.text.gpg
```

```
[0x0] 0L6S 0G
    " . 0 o0a00Y[0w]/0z0LSg900VBo %eF000(0oz[00,oN00[0G07700D^0000o[Pk0j0]0
Bm(0G[0])0"0[0][0]0=Ä0\00[0]0çS040f004[0aI[00[00]F00:0,v00[0]0v<+00000"#00[0]0&\[00{0}Z_00
0l50,s[0ur0uz!\0
    0<00000[0WC 0r00+00[0000Qn000 *0%W-0[0[0[0?700"0oa0\0|0!l{000y[0*)00I00:00
0[0]|n0[a000[0K0S0"n0~:(0#)#0n0CE[00[0[0ua[0000[0v0M0000[0S0y0왓 004000g0n0Sb840K.0000-00"P80[
00[00]}Fz0*D00y0[0[0]0"00
    0Yc0U0C0[0[0[0].000d0p0[0000[00#m"."n001[0]+00M00X00U0p(000z[f[0[0000[0
>a00s0A0000l00\00F0z 0q>0[F'M002F'T0N,}00U0000y000z00A0030S0000GU00b^10#C00LI:0R0 00Y!z:F
0000[A0000000t0000op00!WyB5E0k000wjD,[0JoAo00/X000o0-0H0]0"qhE[0]80[0Z00   s0oh=#R09[000[000
y0[00000*I00F0[0cY79K$0 0Y00000k==m%04Y0[0[0[0SFg[0[0[0]-00S0000p0p0[^¿00,000GT8l[0v000{00u[
00[0[0Sc[0[0]x
    0000XS$.000b cU[0][0]!Eduardo Mora <emg1015@alu.ubu.es>0[0[0][0]
->[0[0]0Z_00@0000t0u03[0[0]0"0[0][0] [0g
        [0][0]
            [0][0][0][0][0][0][0][0]
            [0u03[00N
                0M@I0"0Az0[0[0][0],Ezn0000R0@[t-0[0[0]j0[0Ed00S[0]0j00UKp00m000[0]"0E00=00Y0R00
00l
09000'-00R[0]
    000[Qa[003^000?00"000w0.7#9-[000[00P0'
                                4atP0[00u0[00] 02b[00000[0[0[0]8vm0[000A+
000000F0[0Y00?7[0i6[0:000W[0~0200[0][0]
                        000H00[00000WL0EDl0¿0kS[0[0]t'00-0000[0]0WNZ0q 0^
                                                    N0
0[xk030N0Zg0uFr1[000HZ00SD0?0V00000M[0H00000:0rc00[0vZ0[00b[0040i006x00000000p-000000SC000
```

```
emorag@emorag:~/Escritorio/practica$ gpg --import clave2.gpg
```

**Ejercicio 22. Comprueba el contenido de tu llavero público con la opción -kv.**

```
emorag@emorag:~/Escritorio/practica$ gpg -kv
/home/emorag/.gnupg/pubring.kbx
-----
pub   rsa3072 2020-04-27 [SC] [caduca: 2022-04-27]
      5E4C5B4A7F5ABFE952056530533022DA4CBD3620
uid   [ absoluta ] Eduardo Mora <emg1015@alu.ubu.es>
sub   rsa3072 2020-04-27 [E] [caduca: 2022-04-27]

pub   rsa3072 2020-04-27 [SC] [caduca: 2022-04-27]
      8C88EC99C1707AEEEE01515BE647C5DB2A6DA4FFD
uid   [ absoluta ] EDUARDO MORA <EMG@LOQUE.ES>
sub   rsa3072 2020-04-27 [E] [caduca: 2022-04-27]

pub   rsa3072 2020-05-11 [SC] [caduca: 2022-05-11]
      F7875A5FEEDD408FF78CDB74A775D8331B4616B7
uid   [ absoluta ] Eduardo Mora <emg1015@alu.ubu.es>
sub   rsa3072 2020-05-11 [E] [caduca: 2022-05-11]

pub   rsa3072 2020-05-11 [SC] [caduca: 2022-05-11]
      FC0679659349D4F279BACF597062491051E46C6B
uid   [ absoluta ] Eduardo Mora Gonzalrz <emg1015@alu.ubu.es>
sub   rsa3072 2020-05-11 [E] [caduca: 2022-05-11]
```

**Ejercicio 23. Cifra un mensaje con la clave pública de un compañero y guárdalo como ASCII**

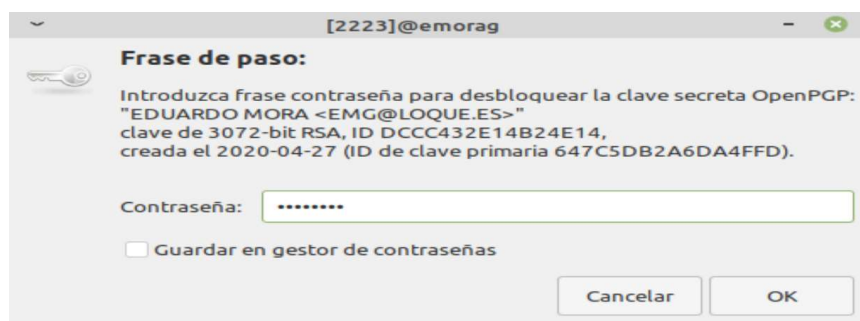
```
gpg -e fichero.text EMG@LOQUE.ES
```

**Ejercicio 24. Envíaselo por correo (mediante mail o cualquier otra aplicación existente) y que él te envíe el suyo a ti. Guarda el mensaje recibido en un fichero.**

Me la he enviado a otro correo para poder hacer el ejercicio.

**Ejercicio 25. Descifralo con tu clave privada.**

```
gpg fichero.text.pgp
```



```
gpg: cifrado con clave de 3072 bits RSA, ID DCCC432E14B24E14, creada el 2020-04-27
      "EDUARDO MORA <EMG@LOQUE.ES>"
```

**Ejercicio 26. Cifra el mismo mensaje con otra clave pública y envíaselo al mismo compañero**

```
emorag@emorag:~/Escritorio/practica$ gpg -e fichero.text emg1015@alu.ubu.es
```

**Ejercicio 27. Recibe el mensaje de tu compañero y guárdalo en un fichero.**

Me la he enviado a otro correo para poder hacer el ejercicio.

**Ejercicio 28. Intenta descifrarlo con tu clave privada.**

```
WARNING: Can't find the right public key-- can't check signature integrity.
```

**Ejercicio 29. Firma un mensaje a enviar con tu clave privada y guárdalo como ASCII.**

```
emorag@emorag:~/Escritorio/practica$ gpg --output fichero_ascii.asc --encrypt --recipient emg1015@alu.ubu.es fichero.text
```

**Ejercicio 30. Envía el fichero a un compañero que posea tu clave pública.**

Me la he enviado a otro correo para poder hacer el ejercicio.

**Ejercicio 31. Recibe algún o algunos mensajes firmados de tus compañeros y guárdalos en sendos ficheros.**

Me la he enviado a otro correo para poder hacer el ejercicio.

**Ejercicio 32. Verifica la firma de los mensajes recibidos y comprueba el contenido de los mismos.**

```
emorag@emorag:~/Escritorio/practica$ gpg fichero_ascii.asc
gpg: ADVERTENCIA: no se ha proporcionado ninguna orden. Intentando adivinar lo que quieres...
gpg: cifrado con clave de 3072 bits RSA, ID 238FB48EAB9510B6, creada el 2020-05-11
"Eduardo Mora Gonzalrz <emg1015@alu.ubu.es>"
```

**Ejercicio 33. Genera un mensaje secreto y autentico y envíaselo a un compañero.**

```
gpg -sea fichero.text EMG@LOQUE.ES -u emg1015@alu.ubu.es
```

**Ejercicio 34. Recibe algún mensaje con estas características, verifícalo y descifrarlo.**

```
gpg fichero.text.asc
```