



Título de la actividad: Tarea llaves de encriptación

Nombre: Eric Carmen Soto

Fecha de realización: 13/11/2025

Semestre: 9no

Unidad de Aprendizaje (UA): Sistemas Operativos

Periodo escolar: 2025B

Institución: Centro Universitario UAEM
Zumpango

Reporte Técnico: Intercambio de Llaves y Desencriptación de Mensaje con GPG

Objetivo

Documentar el proceso completo de generación de llaves GPG, intercambio con un compañero, cifrado de mensaje y desencriptación del mensaje recibido, validando cada paso mediante comandos reproducibles y evidencia visual.

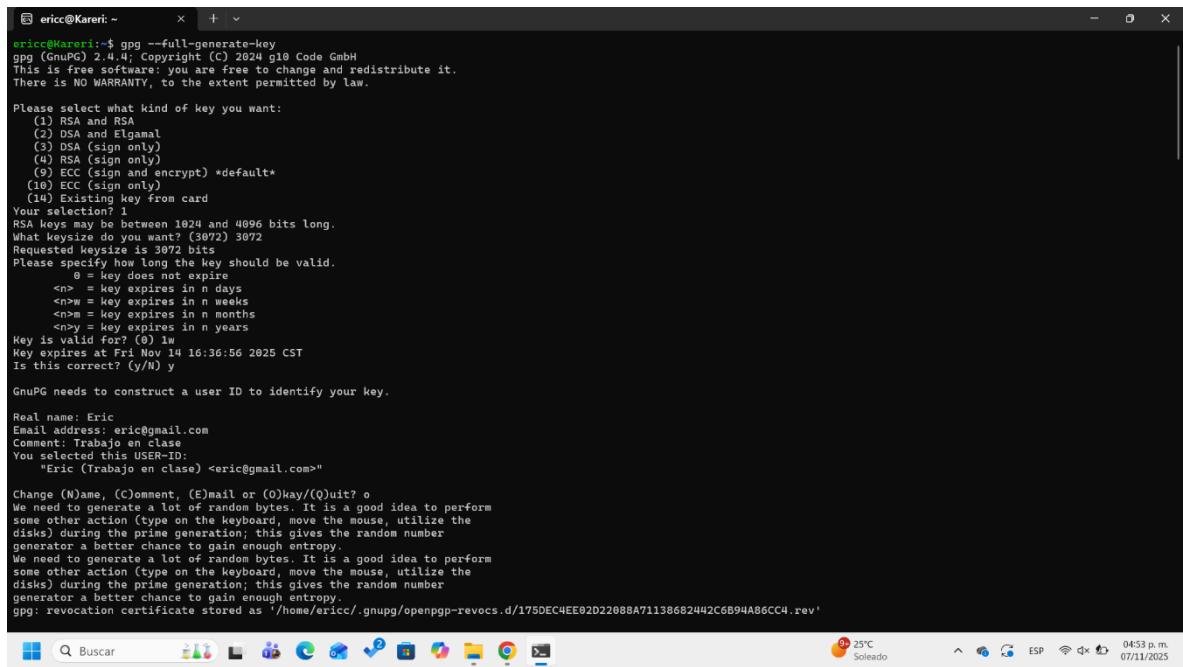
1. Generación de Llave GPG Personal

Se inicia el proceso con el comando:

Opciones seleccionadas:

- Tipo de llave: RSA y RSA (opción 1)
- Tamaño: 3072 bits
- Vigencia: 1 año
- Identidad: "Eric (Trabajo en clase) <eric@gmail.com>"

Se genera entropía adicional para fortalecer la seguridad criptográfica y se guarda el certificado de revocación en el directorio correspondiente.



```
eric@Karei: ~      x + v
eric@Karei:~$ gpg --full-generate-key
gpg (GnuPG) 2.4.4; Copyright (C) 2024 g10 Code GmbH
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Please select what kind of key you want:
 (1) RSA and RSA
 (2) DSA and ElGamal
 (3) RSA (sign only)
 (4) RSA (sign only)
 (9) ECC (sign and encrypt) *default*
 (10) ECC (sign only)
 (14) Existing key from card
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 3072
Requested keysize is 3072 bits
Please specify how long the key should be valid.
    0 = key does not expire
    <n>  = key expires in n days
    <n>w = key expires in n weeks
    <n>m = key expires in n months
    <n>y = key expires in n years
Key is valid for? (0) 1w
Key expires at Fri Nov 14 16:36:56 2025 CST
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.

Real name: Eric
Email address: eric@gmail.com
Comment: Trabajo en clase
You selected this USER-ID:
  "Eric (Trabajo en clase) <eric@gmail.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? o
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: revocation certificate stored as '/home/ericc/.gnupg/openpgp-revocs.d/175DEC4EE02D22088A71138682442C6B94A86CC4.rev'
```

2. Exportación de Llave Propia e Importación de Llave del Compañero

Se exporta la llave pública generada con:

gpg --export -a Eric > Eric.asc

Luego se intenta importar la llave del compañero. El primer intento falla por error en el nombre del archivo (KeyYes.asc), pero se corrige con:

gpg --import keyYes.asc

Se importa la llave publica de Yesenia, confirmando que el intercambio fue exitoso.

3. Edición y Cifrado del Mensaje Propio

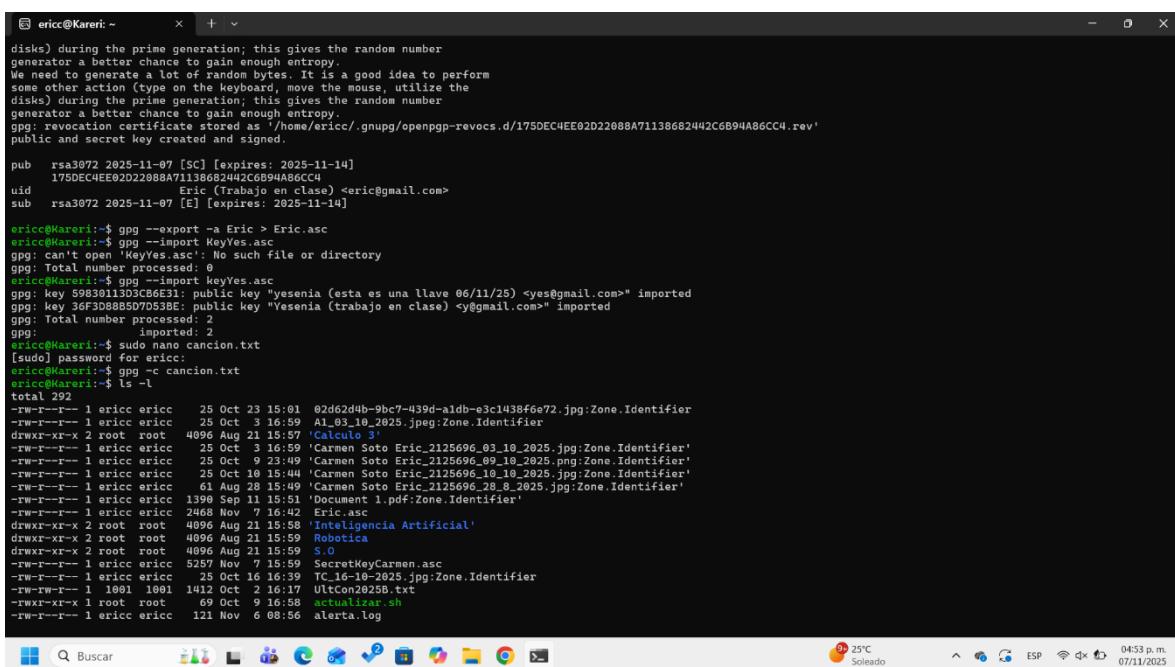
Se crea y edita el archivo cancion.txt con el mensaje a enviar, utilizando nano:

sudo nano cancion.txt

Luego se cifra el archivo con una frase de contraseña mediante cifrado simétrico:

gpg -c cancion.txt

Esto genera el archivo **cancion.txt.gpg**, que puede ser enviado de forma segura.



```
eric@Kareni: ~          x + v
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: revocation certificate stored at '/home/ericc/.gnupg/openpgp-revocs.d/175DEC4EE02D22088A71138682442C6B94A86CC4.rev'
public and secret key created and signed.

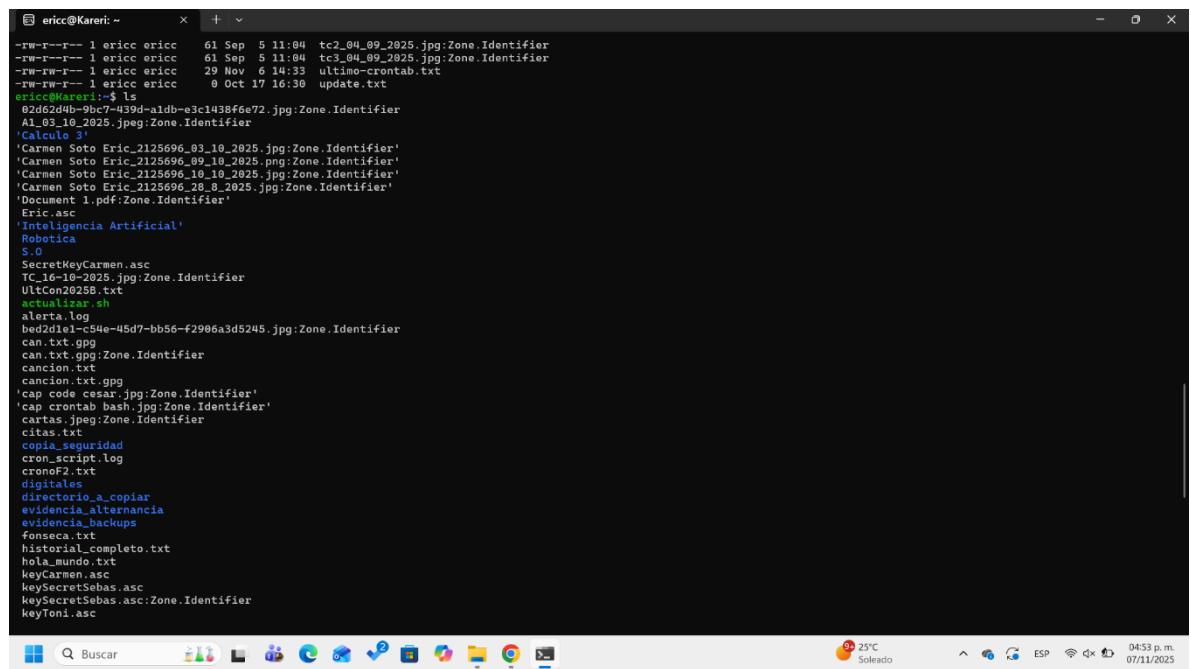
pub    rsa3072 2025-11-07 [SC] [expires: 2025-11-14]
      175DEC4EE02D22088A71138682442C6B94A86CC4
uid          Eric (Trabajo en clase) <eric@gmail.com>
sub    rsa3072 2025-11-07 [E] [expires: 2025-11-14]

eric@Kareni: ~$ gpg --export -a Eric > Eric.asc
eric@Kareni: ~$ gpg --import KeyYes.asc
gpg: can't open 'KeyYes.asc': No such file or directory
gpg: Total number processed: 0
eric@Kareni: ~$ gpg --import KeyYes.asc
gpg: key 59830113D3CB6E1: public key "yesenia (esta es una llave 06/11/25) <yes@gmail.com>" imported
gpg: key 36F3D8885D5D9E85: public key "yesenia (trabajo en clase) <y@gmail.com>" imported
gpg: Total number processed: 2
gpg:              imported: 2
eric@Kareni: ~$ sudo nano cancion.txt
[sudo] password for ericc:
eric@Kareni: ~$ gpg -c cancion.txt
eric@Kareni: ~$ ls -l
total 292
-rw-r--r-- 1 ericc ericc  25 Oct 23 15:01 02d62d4b-9bc7-439d-aldb-e3c1438f6e72.jpg:Zone.Identifier
-rw-r--r-- 1 ericc ericc  25 Oct  3 16:59 A1_03_10_2025.jpeg:Zone.Identifier
drwxr-xr-x  2 root  root  4096 Aug 21 15:57 'Calculo 3'
-rw-r--r-- 1 ericc ericc  25 Oct  9 23:49 Carmen Soto Eric_2125696_03_10_2025.jpg:Zone.Identifier
-rw-r--r-- 1 ericc ericc  25 Oct  9 23:49 Carmen Soto Eric_2125696_09_10_2025.png:Zone.Identifier
-rw-r--r-- 1 ericc ericc  25 Oct 28 15:44 (Compartido) Eric_2125696_10_10_2025.jpg:Zone.Identifier
-rw-r--r-- 1 ericc ericc  61 Aug 28 15:49 'Carmen Soto Eric_2125696_28_8_2025.jpg:Zone.Identifier'
-rw-r--r-- 1 ericc ericc 1390 Sep 11 15:51 'Document 1.pdf:Zone.Identifier'
-rw-r--r-- 1 ericc ericc 2468 Nov  7 16:42 Eric.asc
drwxr-xr-x  2 root  root  4096 Aug 21 15:58 'Inteligencia Artificial'
drwxr-xr-x  2 root  root  4096 Aug 21 15:59 Robótica
drwxr-xr-x  2 root  root  4096 Aug 21 15:59 S.O
-rw-r--r-- 1 ericc ericc 5287 Nov  7 15:59 SecretKeyCarmen.asc
-rw-r--r-- 1 ericc ericc  25 Oct 18 16:39 TC_16-10-2025.jpg:Zone.Identifier
-rw-r--r-- 1 1001 1001 1412 Oct  2 16:17 UltCon2025B.txt
-rwxr-xr-x  1 root  root   69 Oct  9 16:58 actualizar.sh
-rw-r--r-- 1 ericc ericc 121 Nov  6 08:56 alerta.log
```

4. Verificación del Entorno y Archivos Relacionados

Se ejecuta ls para listar los archivos del directorio personal. Se confirma la presencia de:

- Eric.asc (llave exportada)
- cancion.txt y cancion.txt.gpg (mensaje original y cifrado)
- can.txt.gpg (mensaje recibido de la compañera)



```
ericc@Karei: ~      + ~
-rw-r--r-- 1 ericc ericc  61 Sep  8 11:08 tc2_04_09_2025.jpg:Zone.Identifier
-rw-r--r-- 1 ericc ericc  61 Sep  8 11:08 tc3_04_09_2025.jpg:Zone.Identifier
-rw-rw-r-- 1 ericc ericc 29 Nov  8 14:33 ultimo-crontab.txt
-rw-rw-r-- 1 ericc ericc  8 Oct  8 16:38 update.txt
ericc@Karei:~$ ls
02662d1b-9b7c-439d-a1db-e3c1438f6e72.jpg:Zone.Identifier
A1_03_10_2025.jpg:Zone.Identifier
'Calculo 3'
'Carmen Soto Eric_2125696_03_10_2025.jpg:Zone.Identifier'
'Carmen Soto Eric_2125696_09_10_2025.png:Zone.Identifier'
'Carmen Soto Eric_2125696_10_10_2025.jpg:Zone.Identifier'
'Carmen Soto Eric_2125696_28_8_2025.jpg:Zone.Identifier'
'Datos_1.pdf:Zone.Identifier'
Eric.asc
'Inteligencia Artificial'
Robotica
S.O
SecretKeyCarmen.asc
TC_16-10-2025.jpg:Zone.Identifier
ULTCon025B.txt
actualizar.sh
alerta.log
bed2d1e1-c54e-45d7-bb56-f2906a3d5245.jpg:Zone.Identifier
can.txt.gpg
cancion.txt
cancion.txt.gpg
cap_code_ceesar.jpg:Zone.Identifier
cap_crontab_bash.jpg:Zone.Identifier
cartas.jpeg:Zone.Identifier
citas.txt
copia_seguridad
cron_script.log
cronF2.txt
digitales
directorio_a_copiar
encendido_intermitten
evidencias_backups
fonscga.txt
historial_completo.txt
holo_mundo.txt
keyCarmen.asc
keySecretSebas.asc
keySecretSebas.asc:Zone.Identifier
keyToni.asc
ericc@Karei:~$
```

5. Desencriptación del Mensaje Recibido

Se desencripta el archivo **can.txt.gpg** con:

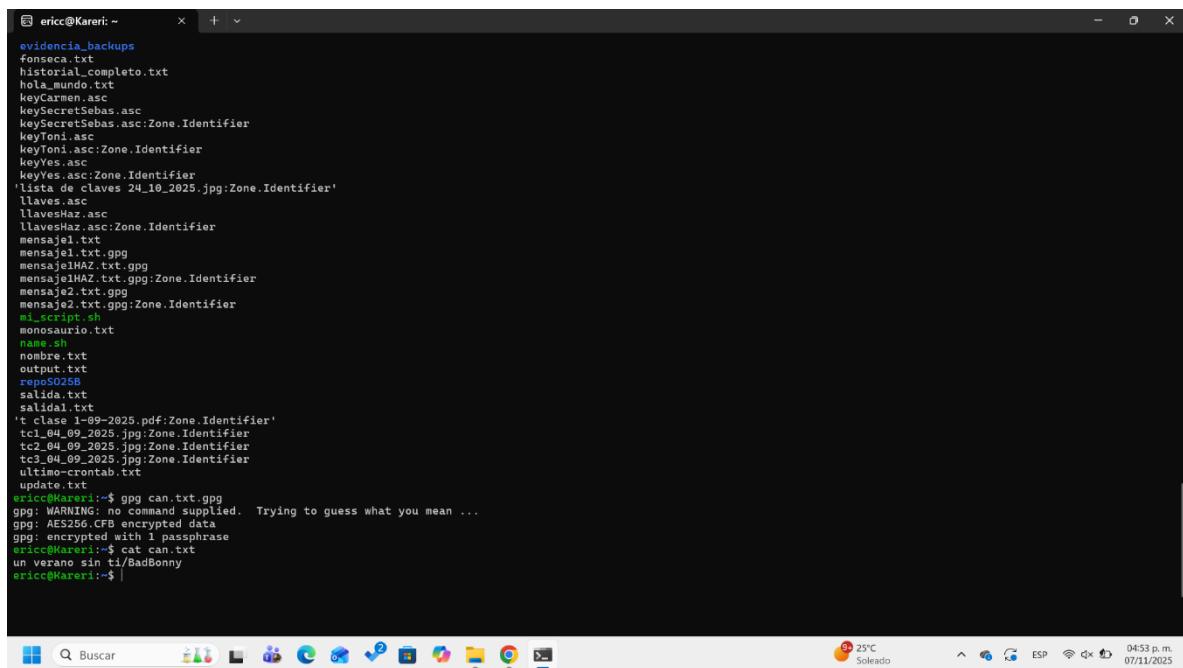
gpg can.txt.gpg

Aunque el comando se ejecuta sin parámetros, GPG lo interpreta correctamente y solicita la frase de contraseña. Luego se muestra el contenido desencriptado con:

cat can.txt

Contenido recuperado:

un verano sin ti/BadBonny



```
ericc@Karei: ~      x + v
evidencia_backups
fonseca.txt
historial_completo.txt
holo_mundo.txt
keyCarmen.asc
keySecretSebas.asc
keySecretSebas.asc:Zone.Identifier
keyToni.asc
keyToni.asc:Zone.Identifier
keyYes.asc
keyYes.asc:Zone.Identifier
'listas de claves 24_10_2025.jpg:Zone.Identifier'
llaves.txt
llavesHdaz.asc
llavesHdaz.asc:Zone.Identifier
mensaje1.txt
mensaje1.txt.gpg
mensaje1HAZ.txt.gpg
mensaje1HAZ.txt.gpg:Zone.Identifier
mensaje2.txt.gpg
mensaje2.txt.gpg:Zone.Identifier
mi_script.sh
monosauvio.txt
name.sh
nombrefoto.txt
output.txt
tempSO258
salida.txt
salida1.txt
't clase 1-09-2025.pdf:Zone.Identifier'
tc1_04_09_2025.jpg:Zone.Identifier
tc2_04_09_2025.jpg:Zone.Identifier
tc3_04_09_2025.jpg:Zone.Identifier
ultimo-crontab.txt
update.txt
ericc@Karei:~$ gpg can.txt.gpg
gpg: no passphrase supplied. Trying to guess what you mean ...
gpg: AES256 CFB encrypted data
gpg: encrypted with 1 passphrase
ericc@Karei:~$ cat can.txt
un verano sin ti/BadBonny
ericc@Karei:~$ |
```

The screenshot shows a terminal window with a dark background. It displays a list of files in the current directory, including various text files, images, and scripts. The user then runs the command 'gpg can.txt.gpg'. The terminal prompts for a passphrase, indicating that the file is encrypted. After entering the passphrase, the contents of the file are decrypted and displayed as 'un verano sin ti/BadBonny'. The terminal window has a standard Windows-style title bar at the top and a taskbar with various icons at the bottom.

Conclusión

El proceso de intercambio de llaves y desencriptación mediante GPG se completó con éxito, cumpliendo los objetivos de seguridad y trazabilidad. Se generó una llave personal, se exportó e intercambió con un compañero, y se cifró un mensaje propio de forma simétrica. Posteriormente, se recibió y desencriptó un mensaje cifrado, validando la integridad del contenido. Este procedimiento demuestra la utilidad de GPG para comunicaciones seguras en entornos académicos y técnicos.

Referencias

- [1] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 8th ed., Pearson, 2023.
- [2] B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 20th Anniversary ed., Wiley, 2015.
- [3] D. J. Bernstein, "Introduction to Public-Key Cryptography," *Lecture Notes in Computer Science*, vol. 4622, pp. 1–14, 2007.
- [4] GNU Privacy Guard, "GnuPG Documentation," [Online]. Available: <https://gnupg.org/documentation/>
- [5] A. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.
- [6] M. Bishop, *Computer Security: Art and Science*, 2nd ed., Addison-Wesley, 2018.