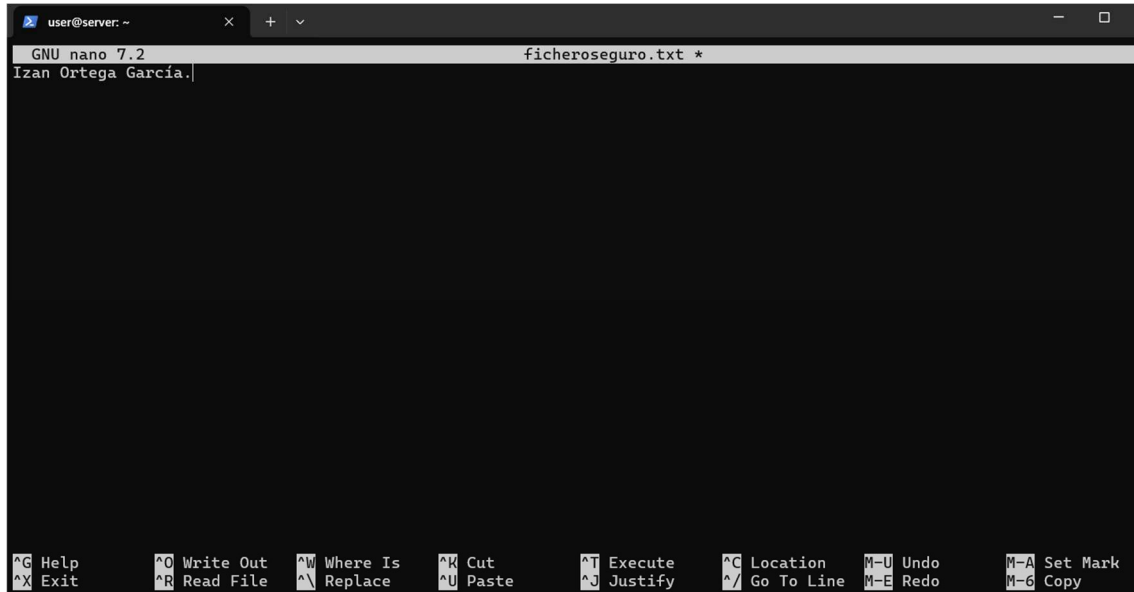


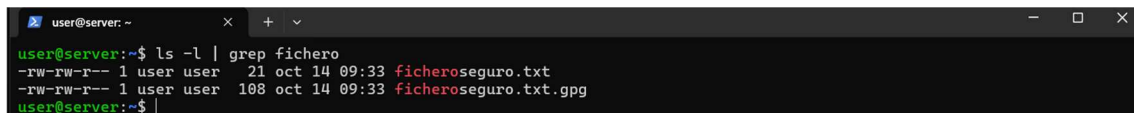
Para la realización de este ejercicio, se procede a la creación de un fichero llamado: “ficheroseguro.txt”. Esto lo haremos con el comando “nano ficheroseguro.txt”. Pondremos nuestro nombre y apellidos. Con el atajo “Ctl + O - Enter” guardamos el fichero.



```
GNU nano 7.2 ficheroseguro.txt *
Izan Ortega García.
```

Con el comando “gpg -c ficheroseguro.txt” procedemos a cifrar nuestro fichero. Se creará otro fichero llamado “ficheroseguro.txt.gpg”.

Con el comando “ls -l | grep fichero” nos muestra el fichero que hemos creado con el “nano” y el fichero “.gpg”.



```
user@server:~$ ls -l | grep fichero
-rw-rw-r-- 1 user user  21 oct 14 09:33 ficheroseguro.txt
-rw-rw-r-- 1 user user 108 oct 14 09:33 ficheroseguro.txt.gpg
user@server:~$
```

Con el comando “cat ficheroseguro.txt.gpg” nos muestra el fichero con lenguaje cifrado.



```
user@server:~$ cat ficheroseguro.txt.gpg
M00[l&q00y0u08W000user@server:~$ |
```

Probamos a desencriptar el fichero previamente cifrado. Si lo hacemos desde nuestra terminal, lo desencripta sin problema.



```
bEA0zuser@gpg --decrypt ficheroseguro2.txt.gpg
gpg: AES256.CFB encrypted data
gpg: encrypted with 1 passphrase
Izan Ortega García.
```

Sin embargo, si lo hacemos desde otra terminal, nos pide la contraseña.

```
user@server: ~  
lqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqk  
x Please enter the passphrase for decryption. x  
x x x  
x Passphrase: |_____ x  
x x x  
x <OK> <Cancel> x  
mqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqj
```

Con el comando “gpg –version” podemos ver la versión de “GPG” que tenemos. Soporta varias técnicas de cifrado, de hecho, se puede usar para funciones de “hashing”. Se puede hacer con otras versiones de GPG.

```
user@server:~$ gpg --version  
gpg (GnuPG) 2.4.4  
libgcrypt 1.10.3  
Copyright (C) 2024 g10 Code GmbH  
License GNU GPL-3.0-or-later <https://gnu.org/licenses/gpl.html>  
This is free software: you are free to change and redistribute it.  
There is NO WARRANTY, to the extent permitted by law.  
  
Home: /home/user/.gnupg  
Supported algorithms:  
Pubkey: RSA, ELG, DSA, ECDH, ECDSA, EDDSA  
Cipher: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH,  
CAMELLIA128, CAMELLIA192, CAMELLIA256  
Hash: SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224  
Compression: Uncompressed, ZIP, ZLIB, BZIP2  
user@server:~$ |
```

Ahora, procederemos a crear un fichero llamado “ficheroseguro3.txt” el cual, procederemos a cifrar con TWOFISH.

```
GNU nano 7.2 ficheroseguro3.txt *  
Izan Ortega García.Z|
```

Con el siguiente comando, ciframos el fichero en TWOFISH: “gpg -c --cipher-algo TWOFISH ficheroseguro3.txt”. El parámetro que permite seleccionar el algoritmo con el que queremos cifrar es “—cipher-algo TWOFISH”.

```
user@server:~$ gpg -c --cipher-algo TWOFISH ficheroseguro3.txt|
```

Comprobamos que se ha creado un fichero “.gpg” y que está cifrado.

```
user@server:~$ ls -l | grep ficheroseguro3  
-rw-r--r-- 1 root root 22 oct 14 10:09 ficheroseguro3.txt  
-rw-rw-r-- 1 user user 110 oct 14 10:11 ficheroseguro3.txt.gpg  
user@server:~$ |
```

Con el comando “cat” comprobamos que el fichero está cifrado.

```
user@server:~$ cat ficheroseguro3.txt.gpg
U+rB?l+mH+M3柳 user@server:~$ |K&G{d+17>^ pR+g+z*I+
```

Para el siguiente paso, procederemos a instalar las “Guest Additions”.

Desencriptación del fichero “secreto.txt”.

```
user@server:~$ gpgconf --kill gpg-agent
user@server:~$ gpg -d secreto.txt
gpg: invalid armor header: jA0EAgMCoIhF64oHhe7SyYCPl+qxMpXgCiACPttlUaY+kGGLLkRuvi5Q/yan3ojF\n
gpg: 3DES.CFB encrypted data
gpg: encrypted with 1 passphrase
La criptografía simétrica es muy rápida es la que se utiliza para
encriptar contenidos.

Operación: 4-7
gpg: WARNING: message was not integrity protected
gpg: Hint: If this message was created before the year 2003 it is
likely that this message is legitimate. This is because back
then integrity protection was not widely used.
gpg: Use the option '--ignore-mdc-error' to decrypt anyway.
gpg: decryption forced to fail!
user@server:~$ |
```