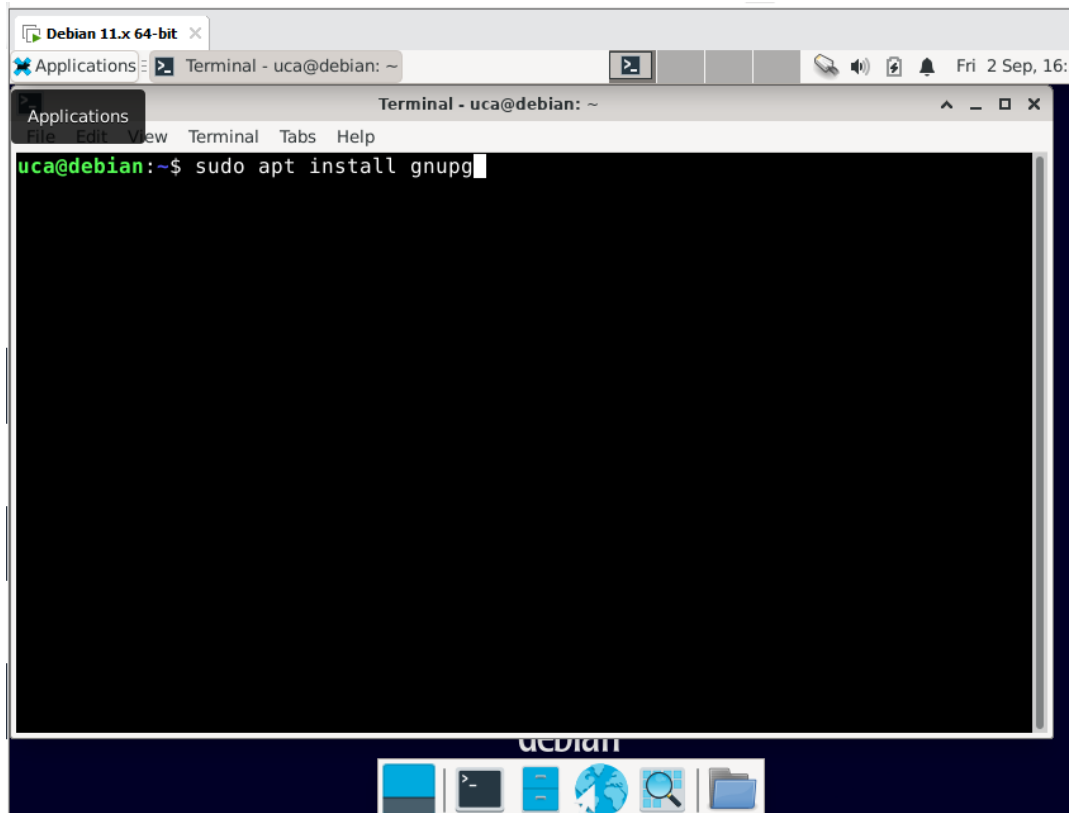
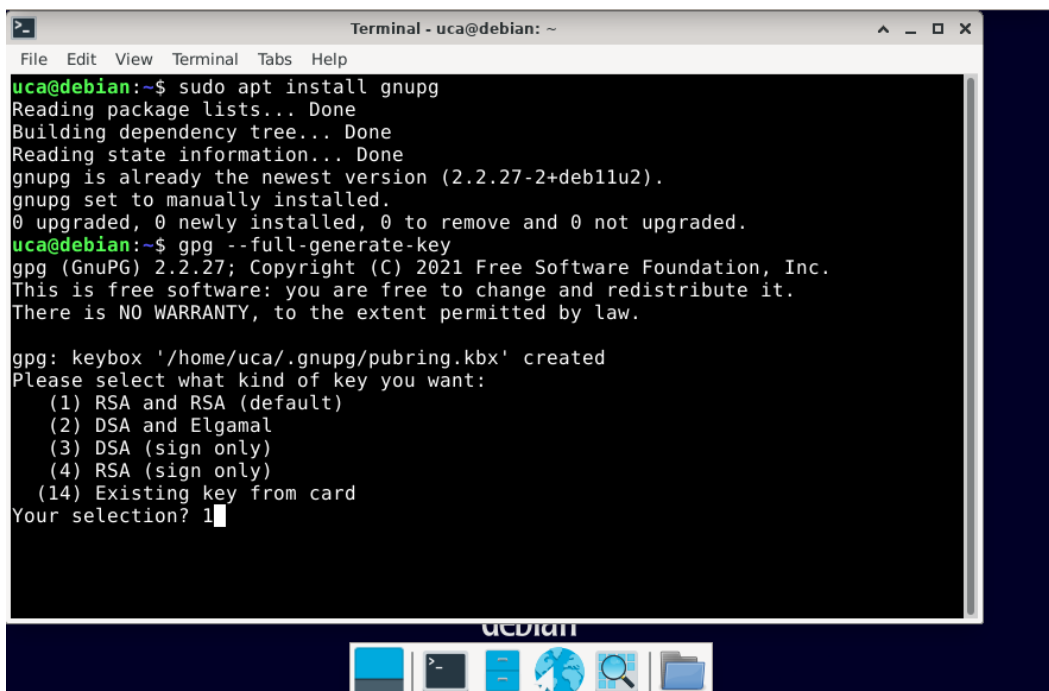


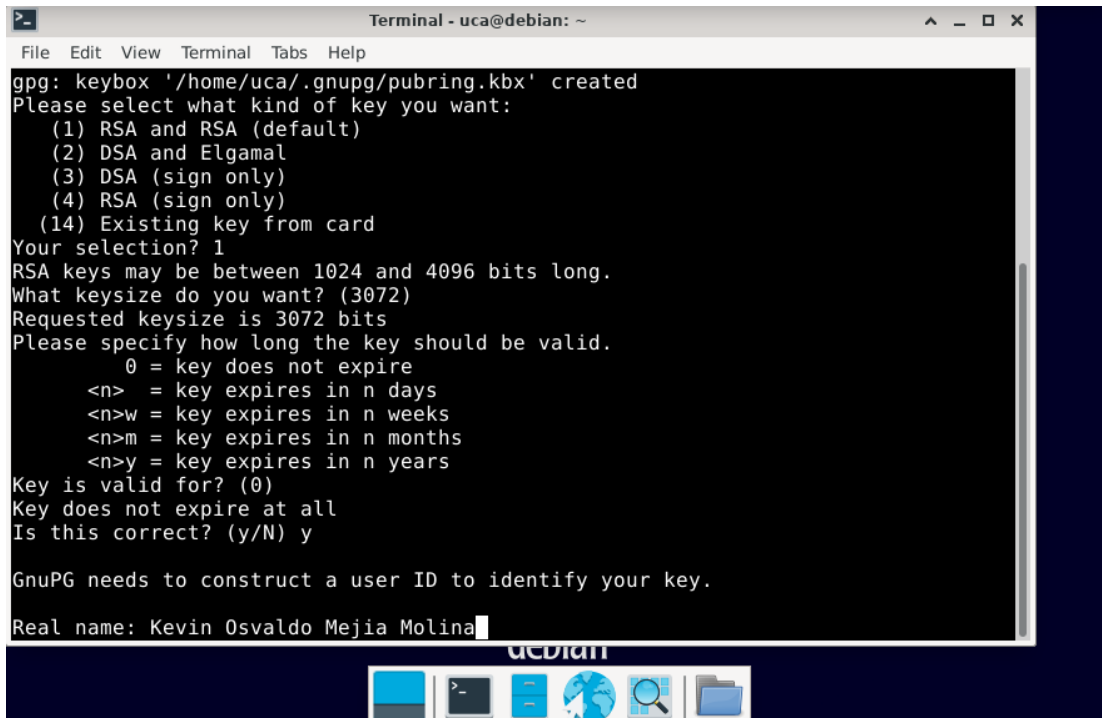
Como primer paso instalamos GNUPG



Segundo paso generamos nuestra clave de firma



Luego seleccionamos el tamaño de nuestra “key” y el tiempo de vigencia, en el caso que escogimos fue de 3070 bits y una clave que no vence e ingresamos nuestro nombre..

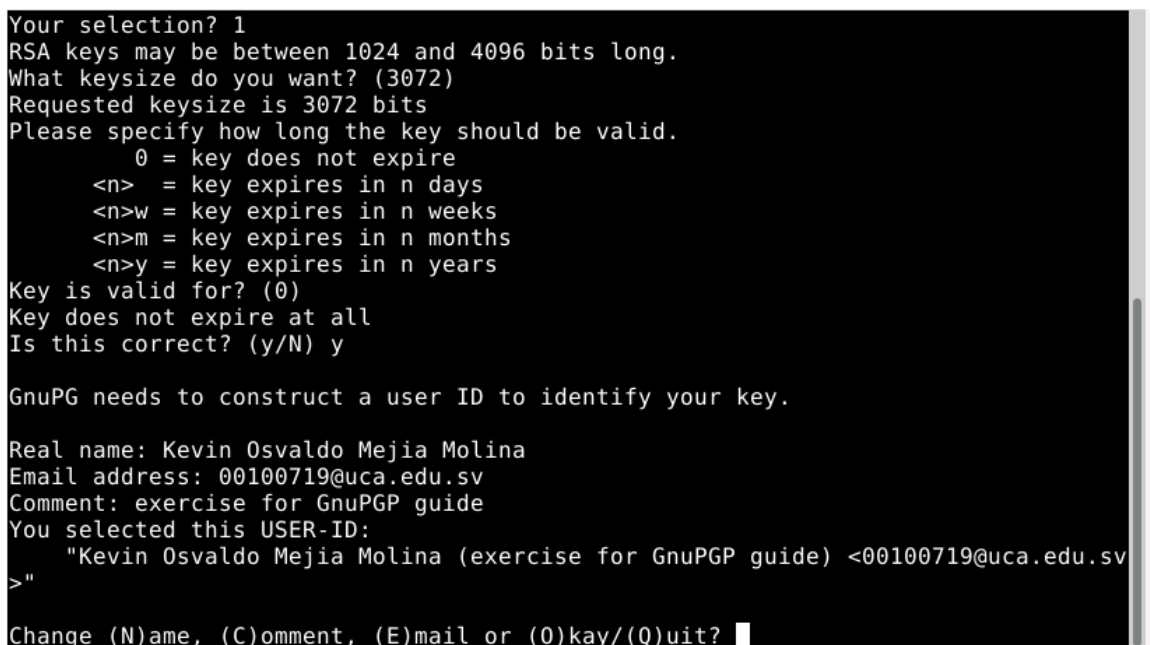


```
Terminal - uca@debian: ~
File Edit View Terminal Tabs Help
gpg: keybox '/home/uca/.gnupg/pubring.kbx' created
Please select what kind of key you want:
  (1) RSA and RSA (default)
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (14) Existing key from card
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072)
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)
Key does not expire at all
Is this correct? (y/N) y

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

Real name: Kevin Osvaldo Mejia Molina
```

Luego ingresamos nuestro correo electrónico y un comentario siempre en nuestro ID de nuestra “key”



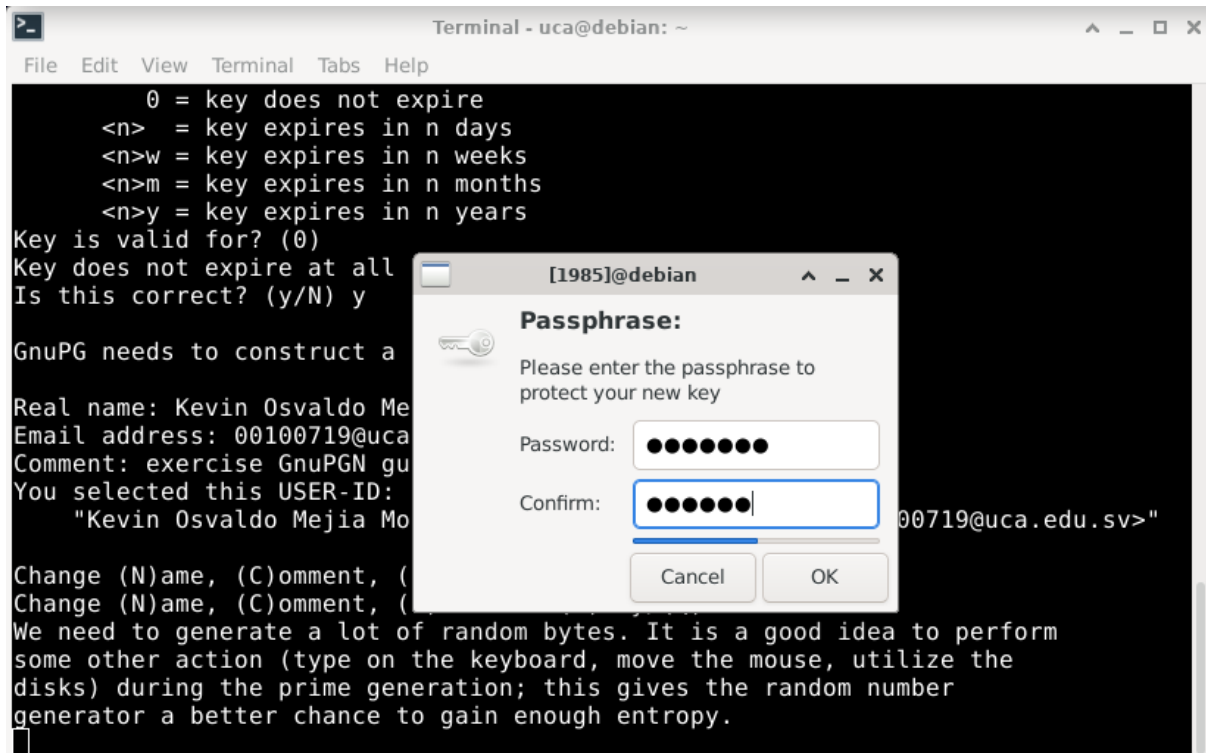
```
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072)
Requested keysize is 3072 bits
Please specify how long the key should be valid.
  0 = key does not expire
  <n> = key expires in n days
  <n>w = key expires in n weeks
  <n>m = key expires in n months
  <n>y = key expires in n years
Key is valid for? (0)
Key does not expire at all
Is this correct? (y/N) y

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

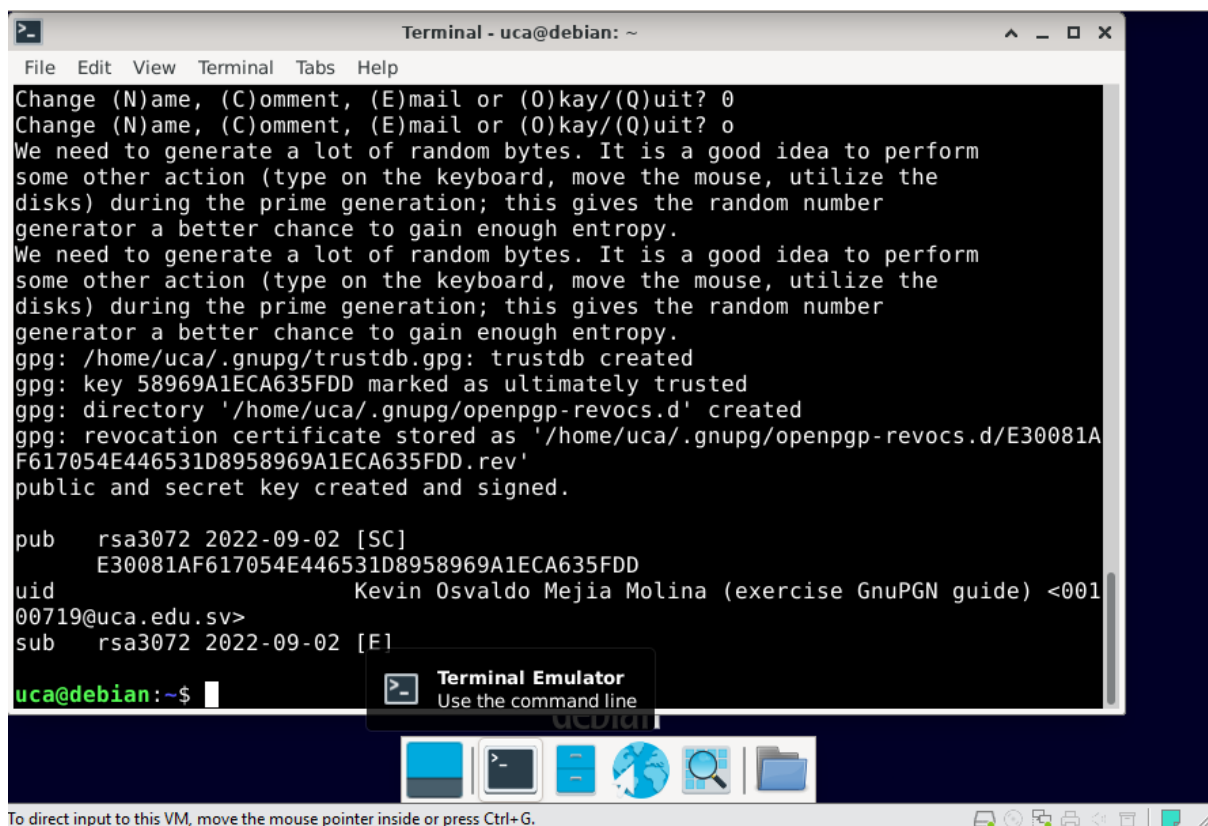
Real name: Kevin Osvaldo Mejia Molina
Email address: 00100719@uca.edu.sv
Comment: exercise for GnuPG guide
You selected this USER-ID:
  "Kevin Osvaldo Mejia Molina (exercise for GnuPG guide) <00100719@uca.edu.sv>"
>

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? 
```

Luego procedemos a ingresar nuestro Passphrase



Luego nos confirmara que generamos nuestra "Key"



luego generamos nuestro certificado de revocación con el siguiente comando

```
uca@debian:~$ gpg --output my_revocation_certificate.asc --gen-revoke 58969A1ECA635FDD
```

Luego verificamos nuestra lista de "keys"

```
uca@debian:~$ gpg --list-keys
gpg: checking the trustdb
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
/home/uca/.gnupg/pubring.kbx
-----
pub   rsa3072 2022-09-02 [SC]
      E30081AF617054E446531D8958969A1ECA635FDD
uid   [ultimate] Kevin Osvaldo Mejia Molina (exercise GnuPGN guide) <00100719@uca.edu.sv>
sub   rsa3072 2022-09-02 [E]
```

luego hacemos un export de nuestra firma

```
uca@debian:~$ gpg --export-secret-keys --armor 00100719@uca.edu.sv > ./kevin.asc
```

creamos una firma nueva para agregarlas a nuestras "keys"

```
uca@debian:~$ nano kevin.asc
uca@debian:~$ nano nestor.asc
```

luego procedemos a importar nuestras "keys" en la lista

```
uca@debian:~$ gpg --import miguel.asc
gpg: key 22FD98109A7AB1C3: public key "miguel rivas (clave for uca sei) <00087518@uca.edu.sv>" imported
gpg: Total number processed: 1
gpg:      imported: 1
```

luego confirmamos que agregamos correctamente todas nuestras "keys"

```
uca@debian:~$ gpg --list-keys
/home/uca/.gnupg/pubring.kbx
-----
pub   rsa3072 2022-09-02 [SC]
      E30081AF617054E446531D8958969A1ECA635FDD
uid   [ultimate] Kevin Osvaldo Mejia Molina (exercise GnuPGN guide) <00100719@uca.edu.sv>
sub   rsa3072 2022-09-02 [E]

pub   rsa3072 2022-08-17 [SC] [expires: 2022-10-16]
      FE42FA6AEB275595B093D4F322FD98109A7AB1C3
uid   [ unknown] miguel rivas (clave for uca sei) <00087518@uca.edu.sv>
sub   rsa3072 2022-08-17 [E] [expires: 2022-10-16]

pub   rsa3072 2022-08-17 [SC]
      9EE66B446C0E7BC1B74E6DE9CB584318958A9202
uid   [ unknown] Nestor Santiago Aldana Rodriguez (GnuPGP Guide - For UCA in SED) <naldana@uca.edu.sv>
sub   rsa3072 2022-08-17 [E]
```

Procedemos a hacer la validación de la claves que importamos

```
uca@debian:~$ gpg --edit-key 00087518@uca.edu.sv
gpg (GnuPG) 2.2.27; Copyright (C) 2021 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.

pub rsa3072/22FD98109A7AB1C3
   created: 2022-08-17   expires: 2022-10-16   usage: SC
   trust: unknown      validity: unknown
sub rsa3072/04C91A3A0839EA12
   created: 2022-08-17   expires: 2022-10-16   usage: E
[ unknown] (1). miguel rivas (clave for uca sei) <00087518@uca.edu.sv>

gpg> fpr
pub rsa3072/22FD98109A7AB1C3 2022-08-17 miguel rivas (clave for uca sei) <00087518@uca.edu.sv>
Primary key fingerprint: FE42 FA6A EB27 5595 B093 D4F3 22FD 9810 9A7A B1C3

gpg> sign

pub rsa3072/22FD98109A7AB1C3
   created: 2022-08-17   expires: 2022-10-16   usage: SC
   trust: unknown      validity: unknown
Primary key fingerprint: FE42 FA6A EB27 5595 B093 D4F3 22FD 9810 9A7A B1C3

   miguel rivas (clave for uca sei) <00087518@uca.edu.sv>

This key is due to expire on 2022-10-16.
Are you sure that you want to sign this key with your
key "Kevin Osvaldo Mejia Molina (exercise GnuPGN guide) <00100719@uca.edu.sv>" (58969A1ECA635FDD)

Really sign? (y/N) y

gpg> quit
Save changes? (y/N) y
```

Confirmamos que recibieron la validación

```
uca@debian:~$ gpg --list-keys
gpg: checking the trustdb
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: depth: 0 valid: 1 signed: 2 trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: depth: 1 valid: 2 signed: 0 trust: 2-, 0q, 0n, 0m, 0f, 0u
gpg: next trustdb check due at 2022-10-16
/home/uca/.gnupg/pubring.kbx
-----
pub rsa3072 2022-09-02 [SC]
   E30081AF617054E446531D8958969A1ECA635FDD
uid [ultimate] Kevin Osvaldo Mejia Molina (exercise GnuPGN guide) <00100719@uca.edu.sv>
sub rsa3072 2022-09-02 [E]

pub rsa3072 2022-08-17 [SC] [expires: 2022-10-16]
   FE42FA6AEB275595B093D4F322FD98109A7AB1C3
uid [ full ] miguel rivas (clave for uca sei) <00087518@uca.edu.sv>
sub rsa3072 2022-08-17 [E] [expires: 2022-10-16]

pub rsa3072 2022-08-17 [SC]
   9EE66B446C0E78C1B74E6DE9CB584318958A9202
uid [ full ] Nestor Santiago Aldana Rodriguez (GnuPG Guide - For UCA in SED) <naldana@uca.edu.sv>
sub rsa3072 2022-08-17 [E]
```

ahora procedemos a cifrar un documento de texto creamos el documento y revisamos su contenido

```
uca@debian:~$ history > history.txt
uca@debian:~$ cat history.txt
 1 su
 2 sudo apt install git
 3 brew install gnupg
 4 clear
 5 sudo apt install gnupg
 6 gpg --full-generate-key
 7 gpg --output my_revocation_certificate.asc --gen-revoke <58969A1ECA635FDD>
 8 gpg --output my_revocation_certificate.asc --gen-revoke <58969A1ECA635FDD>
 9 gpg --output my_revocation_certificate.asc --gen-revoke <58969A1ECA635FDD>
10 gpg --output my_revocation_certificate.asc --gen-revoke 58969A1ECA635FDD
11 gpg --output my_revocation_certificate.asc --gen-revoke 58969A1ECA635FDD
12 gpg --list-keys
13 gpg --list-secret-keys
14 gpg --output nextor.gpg --export naldana@uca.edu.sv
15 gpg --output nextor.gpg --export 00100719@uca.edu.sv
16 gpg --armor --export 00100719@uca.edu.sv
17 nano kevin.asc
18 nano nestor.asc
19 gpg --list-keys
20 gpg --output nextor.gpg --export naldana@uca.edu.sv
21 gpg --armor --export naldana@uca.edu.sv
22 gpg --export-secret-keys --armor 00100719@uca.edu.sv > ./kevin.asc
23 nano miguel.asc
24 gpg --list-keys
25 gpg --import ~/miguel.gpg
26 gpg --import miguel.gpg
27 gpg --import ~/Downloads/miguel.gpg
28 gpg --import /miguel.gpg
29 gpg --import miguel.gpg
30 gpg --import miguel.asc
31 gpg --import nestor.asc
32 gpg --list-keys
33 gpg --edit-key 00087518@uca.edu.sv
34 gpg --edit-key naldana@uca.edu.sv
35 gpg --list-keys
36 history > history.txt
```

Luego procedemos a cifrar nuestro documento de texto y nos solicitará ingresar un passphrase luego de ingresarla podemos verificar que nuestro texto se cifró correctamente

```
uca@debian:~$ gpg --output history.txt.gpg --symmetric history.txt
uca@debian:~$ cat history.txt.gpg
0
00000000c06000000010.000Z0601
2C0v0000RS+g;00'00S0v0t0t00uD060B0070660SD37T0 0
Z00LMr000000z00FCyPt0m , 00@0glS06a0i ; 00q0RS7_50
00xERUjsEKECs0b06S2Pq({ 0S!0QA 00H0b00
0000-00N0(000000T00000000h00000003=00000000UzD0SY0I|0000Hy0000z
ad0K0k0 00H090000AhX000000L00S000G)J000ErT0S00-00000kR0G'X00jmx000000
0000 1000000000Y00000000uca@debian:~$
```

Luego procedemos a crear un cifrado asimétrico el cual se procede a crear de esta forma

```
uca@debian:~$ history > historyPublicKey.txt
uca@debian:~$ gpg --output historyPublicKey.txt.gpg --encrypt --recipient 00100719@uca.edu.sv historyPublicKey.txt
```

luego verificamos que nuestro cifrado se realizo correctamente

[illegible]