

Criptografía

Adrián Gómez Lois

Contenido

1.	Obxectivos.....	3
2.	Scripts de cifrado.....	4
3.	Cifrado simétrico	7
4.	Cifrado de datos y particiones.....	15
5.	Funcións resumen	19
6.	Cifrado asimétrico	24
7.	Firma digital de un documento	38
8.	Correo electrónico. Sinatura e cifrado	63
9.	Esteganografía.....	70
10.	Conclusíóns.....	72

1. Obxectivos

O obxectivo destas tarefas e poder entender os mecanismos de cífrados da información dixital, para poder conseguir unha verazidade dos datos e dos axentes que transmiten dita información.

Para iso veremos en que consisten e como utilizar os cífrados simétricos e asimétricos, o cífrado de datos en volúmenes, a comprobación da integridade dos arquivos, o firmado e cífrado dos documentos para a súa transmisión a través de mensaxería SMTP e finalizarse coa ocultación de información orixinal emascarada noutra información o que se conoce como a esteganografía.

2. Scripts de cifrado

Para a realización desta práctica faremos uso do comando “tr” en Linux o cal permitirános manipular cadeas de texto, podelas sustituir por outras.

Creamos un ficheiro “textoPlano” o cal contén o seguinte texto:

```
GNU nano 2.2.6          Archivo: textoPlano

Esto e unha proba de texto.
Para comprobar o cifrado cesar.
```

Como podemos ver na seguinte captura mostramos o conido orixinal en plano do ficheiro creado anteriormente, a continuación concatenando e pasando con “cat” o resultado do ficheiro en cuestión ao comando “tr” (empregando o estilo de algoritmo de cifrado César) e indicando un ficheiro de saída (>). Vemos que este xa fixo as sustitucións solicitadas, cambiando así as letras as súas tres posicións seguintes no abecedario e tanto minúsculas como mayúsculas.

Para descifralo usamos o proceso inverso, e indicamos un ficheiro de saída.

```
root@seadserver:/cifrado# ls
textoPlano  transpcion.sh
root@seadserver:/cifrado# cat textoPlano
Esto e unha proba de texto.
Para comprobar o cifrado cesar.
root@seadserver:/cifrado# cat textoPlano | tr '[a-z]' '[d-zabc]' | tr '[A-Z]' '[D-ZABC]' > textoCesar
root@seadserver:/cifrado# ls
textoCesar  textoPlano  transpcion.sh
root@seadserver:/cifrado# cat textoCesar
Hwur h xqkd sured gh whawr.
Sdud frpsuredu r fliudgr fhvdu.
root@seadserver:/cifrado# cat textoCesar | tr '[d-zabc]' '[a-z]' | tr '[D-ZABC]' '[A-Z]' > textoDescifrado
root@seadserver:/cifrado# cat textoDescifrado
Esto e unha proba de texto.
Para comprobar o cifrado cesar.
root@seadserver:/cifrado# ls
textoCesar  textoDescifrado  textoPlano  transpcion.sh
root@seadserver:/cifrado#
```

Segundo o mesmo proceso agora faremos que se sustituian polos signos “;.-_” da seguinte forma. Para descifralo usaremos o proceso inverso, e decir na orden da instrucción do comando invertiremos os signos polas letras.

```
root@seadserver:/cifrado# cat textoPlano | tr '[a-z][A-Z]' '[;.-_]' > textoSignos
root@seadserver:/cifrado# cat textoSignos
L>?: 0 @93: ;=::: /0 ?0C?::.
W:=: .:8;=::,:= : .41=:/: .0>:=.
root@seadserver:/cifrado# ls
textoCesar    textoPlano    textoUpper
textoDescifrado  textoSignos  transposicion.sh
root@seadserver:/cifrado# _
```

No seguinte exemplo, dado o texto inicial pasaremos todo o texto a mayúsculas. Polo que de entrada definiremos con tr os caracteres “lower” (minúsculas) e posteriormente definiremos “upper” para convertilo en mayúsculas. Para descifralo igual que os anteriores exercicios faremos o proceso inverso.

```
root@seadserver:/cifrado# cat textoPlano | tr '[:lower:]' '[:upper:]' > textoUpper
root@seadserver:/cifrado# cat textoUpper
ESTO E UNHA PROBA DE TEXTO.
PARA COMPROBAR O CIFRADO CESAR.
root@seadserver:/cifrado# ls
textoCesar  textoDescifrado  textoPlano  textoUpper  transposicion.sh
root@seadserver:/cifrado# _
```

Agora a través dun script que contén o código necesario para realizar unha transposición de caracteres. Vamos ejecutar dito script.sh (chamado trasnposition.sh) para que realice a transposición o ficheiro en textoPlano.

Como estas prácticas desembólvense en VMs, primeiro pasamos o script da máquina anfitrión (real) a máquina huesped (virtual). Neste caso coa tarjeta de rede en modo NAT da máquina virtual, compartimos dende a máquina real unha carpeta, a continuación montámola con mount nun directorio local da máquina virtual (creado previamente).

```
root@seadserver:~# sudo mount -o username=adrian //10.0.0.4/COMPARTIDA /home/montaje
Password for adrian@//10.0.0.4/COMPARTIDA:
root@seadserver:~# ls /home/montaje
Thumbs.db  transposicion.sh
root@seadserver:~# mkdir /cifrado
root@seadserver:~# cp /home/montaje/transposicion.sh /cifrado
root@seadserver:~# ls -l /cifrado
total 4
-rwxr-xr-x 1 root root 539 Apr  1 00:46 transposicion.sh
root@seadserver:~#
```

Co comando bash ou tamén bastaría con ./transposicion.sh. Poderemos ejecutar o script sobre o ficheiro a manipular (textoPlano). O final deste script este xenera un ficheiro nomeado igual que o orixinal pero engadindo “_transposicion”.

Pódese ver na seguinte captura todo o proceso.

```
root@seadserver:/cifrado# ls -la
total 32
drwxr-xr-x  2 root root 4096 abr  4 21:48 .
drwxr-xr-x 24 root root 4096 abr  3 23:27 ..
-rw-r--r--  1 root root   60 abr  1 09:52 textoCesar
-rw-r--r--  1 root root   60 abr  1 10:09 textoDescifrado
-rw-r--r--  1 root root   60 abr  1 09:48 textoPlano
-rw-r--r--  1 root root   62 abr  4 21:34 textoPlano_transposicion
-rw-r--r--  1 root root   60 abr  1 10:58 textoSignos
-rw-r--r--  1 root root    0 abr  1 10:31 textoUpper
-rw-r--r--  1 root root  514 abr  4 21:33 transposicion.sh
root@seadserver:/cifrado# bash transposicion.sh textoPlano
root@seadserver:/cifrado# ls
textoCesar      textoPlano          textoSignos  transposicion.sh
textoDescifrado textoPlano_transposicion  textoUpper
root@seadserver:/cifrado# cat textoPlano
Esto e unha proba de texto.
Para comprobar o cifrado cesar.
root@seadserver:/cifrado# cat textoPlano_transposicion
otsE e ahmu aborp ed .otxet
araP raborp moc o odarf ic .rasec
otsE e ahmu aborp ed .otxet
araP raborp moc o odarf ic .rasec
root@seadserver:/cifrado#
```

3. Cifrado simétrico

Cifraremos con chave simétrica un ficheiro que contén unha mensaxe. Farémolo con GPG (*GNU Privacy Guard*).

Comprobamos a versión de gpg instalada actualmente, con echo creamos unha mesaxe que a gardaremos nun ficheiro chamado “mensaxe.txt”. Agora con “gpg -c” ciframos de forma simétrica o ficheiro pasando a ser unha extensión .gpg (o cifrado simétrico por defecto de gpg e CAST5). Pedirános unha contrasinal para o cifrado e o posterior descifrado do arquivo.

```
root@seadserver:/# gpg --version
gpg (GnuPG) 1.4.16
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://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: ~/.gnupg
Algoritmos disponibles:
Clave pública: RSA, RSA-E, RSA-S, ELG-E, DSA
Cifrado: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH,
          CAMELLIA128, CAMELLIA192, CAMELLIA256
Resumen: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
Compresión: Sin comprimir, ZIP, ZLIB, BZIP2
root@seadserver:/# mkdir gpg
root@seadserver:/# cd gpg
root@seadserver:/gpg# echo "Esto e unha proba de texto, tarefa sead" > mensaxe.txt
root@seadserver:/gpg# cat mensaxe.txt
Esto e unha proba de texto, tarefa sead
root@seadserver:/gpg# gpg -c mensaxe.txt
gpg: el agente gpg no esta disponible en esta sesión
root@seadserver:/gpg# ls -a
. .... mensaxe.txt mensaxe.txt.gpg
root@seadserver:/gpg# cat mensaxe.txt.gpg
vJ[*****.***Q-=]S** f$>t[HCC***pL*
'4*6*I*
!***P***E***x*root@seadserver:/gpg# _
```

Cos modificadores “-c -a” estaremos cifrando o ficheiro e generando un ficheiro .asc ASCII.

```
root@seadserver:/gpg# gpg -c -a mensaxe.txt
gpg: el agente gpg no esta disponible en esta sesión
root@seadserver:/gpg# ls -a
. .... mensaxe.txt mensaxe.txt.asc mensaxe.txt.gpg
root@seadserver:/gpg# cat mensaxe.txt.asc
-----BEGIN PGP MESSAGE-----
Version: GnuPG v1

jA0EAwMCZPrpprIbaJBgyUmXa1T5B8vp+lpZ6immi.j0sfAh5hdKmxH9awD8eYt4T
f62YbB0guL+ITxu/npNXv2ta+nkgsYAoG3z5bxUIROhzm22MQAYh75EC
=uab4
-----END PGP MESSAGE-----
root@seadserver:/gpg# _
```

Co modificador “-d” desciframos a mensaxe tanto para ficheiros .asc como para .gpg.

```
root@seadserver:/gpg# ls -la
total 20
drwxr-xr-x 2 root root 4096 abr  3 23:31 .
drwxr-xr-x 24 root root 4096 abr  3 23:27 ..
-rw-r--r-- 1 root root   40 abr  3 23:29 mensaxe.txt
-rw-r--r-- 1 root root  201 abr  3 23:31 mensaxe.txt.asc
-rw-r--r-- 1 root root   90 abr  3 23:30 mensaxe.txt.gpg
root@seadserver:/gpg# gpg -d mensaxe.txt.asc
gpg: datos cifrados CAST5
gpg: el agente gpg no esta disponible en esta sesion
gpg: cifrado con 1 frase contraseña
Esto e unha proba de texto, tarefa sead
gpg: AVISO: la integridad del mensaje no esta protegida
root@seadserver:/gpg# gpg -d mensaxe.txt.gpg
gpg: datos cifrados CAST5
gpg: el agente gpg no esta disponible en esta sesion
gpg: cifrado con 1 frase contraseña
Esto e unha proba de texto, tarefa sead
gpg: AVISO: la integridad del mensaje no esta protegida
root@seadserver:/gpg# _
```

Agora con lftp (cliente lixeiro que se instalou manualmente na máquina virtual de Ubuntu Server) subimos o ficheiro a un servidor FTP, que neste caso estaría na máquina anfitrión. Neste caso deixo a proba de cocepto xa que para este punto da tarefa non me encontraba na clase para poder envialo o servidor FTP en cuestión, pero dase a ver un exemplo de como sería.

```
root@seadserver:/gpg# lftp -u adrian 10.0.0.4
Clave:
lftp adrian@10.0.0.4:~> ls
-rw-r--r-- 1 ftp ftp          6 Apr 04 00:02 ficheroprueba.txt
lftp adrian@10.0.0.4:>/ cat ficheroprueba.txt
prueba6 bytes transferidos.
lftp adrian@10.0.0.4:>/ mput /gpg/mensaxe.txt.gpg
90 bytes transferidos.
lftp adrian@10.0.0.4:>/ mput /gpg/mensaxe.txt.asc
201 bytes transferidos.
lftp adrian@10.0.0.4:>/ ls
-rw-r--r-- 1 ftp ftp          6 Apr 04 00:02 ficheroprueba.txt
-rw-r--r-- 1 ftp ftp          201 Apr 03 23:31 mensaxe.txt.asc
-rw-r--r-- 1 ftp ftp          90 Apr 03 23:30 mensaxe.txt.gpg
lftp adrian@10.0.0.4:>/ quit
root@seadserver:/gpg#
```

Descargamos e instalamos gpg para Windows. A través do intérprete de comandos comprobamos a versión de gpg. Tamén crearemos un documento (mensaxe.txt) e cifrarémola de igual forma que como se fixo no exercicio anterior en Ubuntu. Perdirános unha contrasinal para o cifrado e o descifrado do arquivo.

```
C:\Windows\system32\cmd.exe - gpg -c mensaxe.txt

C:\Users\adrian\Documents>gpg --version
gpg (GnuPG) 2.0.29 (Gpg4win 2.3.0)
libgcrypt 1.6.4
Copyright (C) 2015 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://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: C:/Users/adrian/AppData/Roaming/gnupg
Algoritmos disponibles:
Clave pública: RSA, RSA, RSA, ELG, DSA
Cifrado: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH,
          CAMELLIA128, CAMELLIA192, CAMELLIA256
Resumen: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
Compresión: Sin comprimir, ZIP, ZLIB, BZIP2

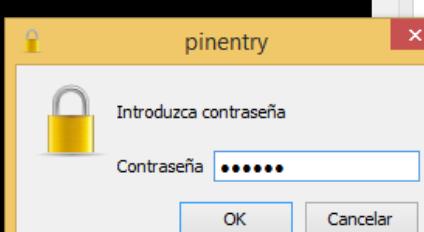
C:\Users\adrian\Documents>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: E8B6-C31A

Directorio de C:\Users\adrian\Documents

04/04/2016  09:32    <DIR>          .
04/04/2016  09:32    <DIR>          ..
04/04/2016  09:32                32 mensaxe.txt
25/02/2016  14:35           259.022 sethc.exe
              2 archivos        259.104 bytes
              2 dirs   1.970.417.664 bytes libres

C:\Users\adrian\Documents>type mensaxe.txt
"Proba de contido de arquivo"

C:\Users\adrian\Documents>gpg -c mensaxe.txt
gpg: almacén 'C:/Users/adrian/AppData/Roaming/gnupg/pubring.gpg' creado
```

A screenshot of the pinentry dialog box, which is a graphical interface for entering a GPG passphrase. The window title is "pinentry". It contains a yellow padlock icon and the text "Introduzca contraseña". Below that is a password input field containing "*****" and two buttons at the bottom right labeled "OK" and "Cancelar". The dialog is overlaid on the terminal window.

Con “-d” ou “--decrypt” (sería o mesmo, un no seu método abreviado e outro no extendido), descriframos a mensaxe.txt cifrada, pediranos de novo a contrasinal establecida.

```
C:\Windows\system32\cmd.exe

C:\Users\adrian\Documents>dir
El volumen de la unidad C no tiene etiqueta.
El n mero de serie del volumen es: E8B6-C31A

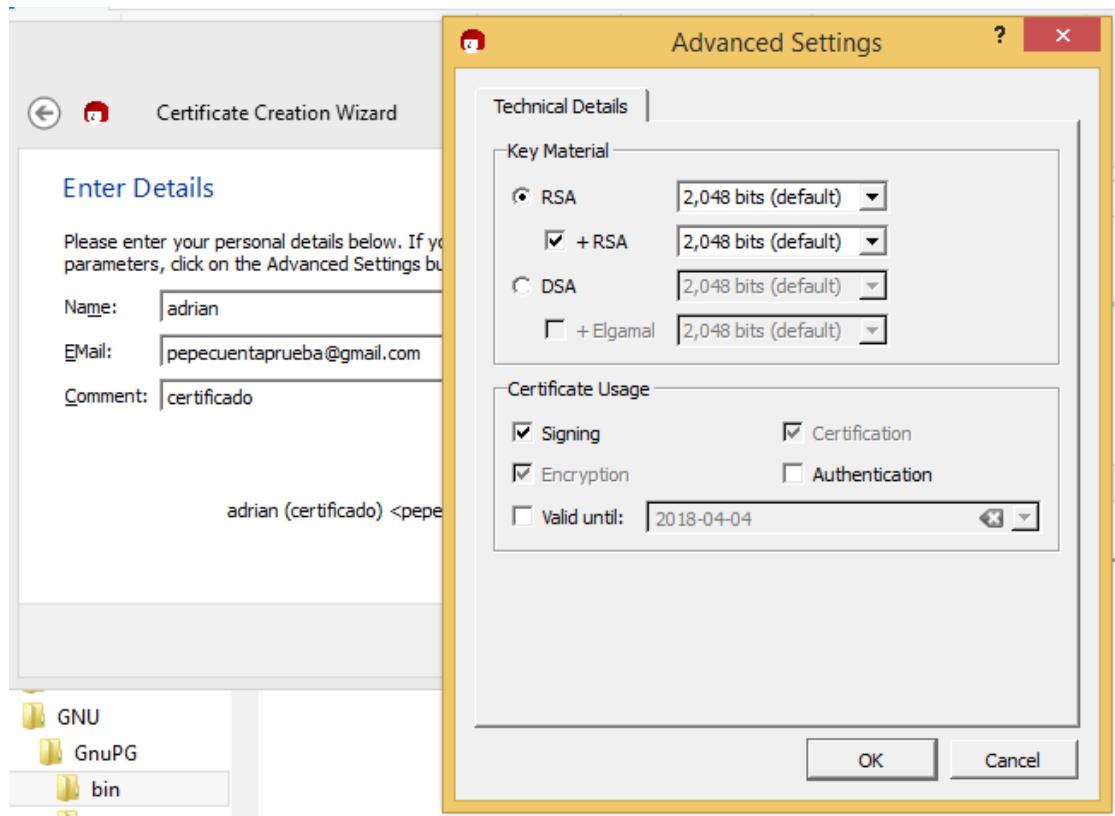
Directorio de C:\Users\adrian\Documents

04/04/2016  09:34    <DIR>          .
04/04/2016  09:34    <DIR>          ..
04/04/2016  09:32                32 mensaxe.txt
04/04/2016  09:34                80 mensaxe.txt.gpg
25/02/2016  14:35           259.072 sethc.exe
                           3 archivos   259.184 bytes
                           2 dirs     1.968.865.280 bytes libres

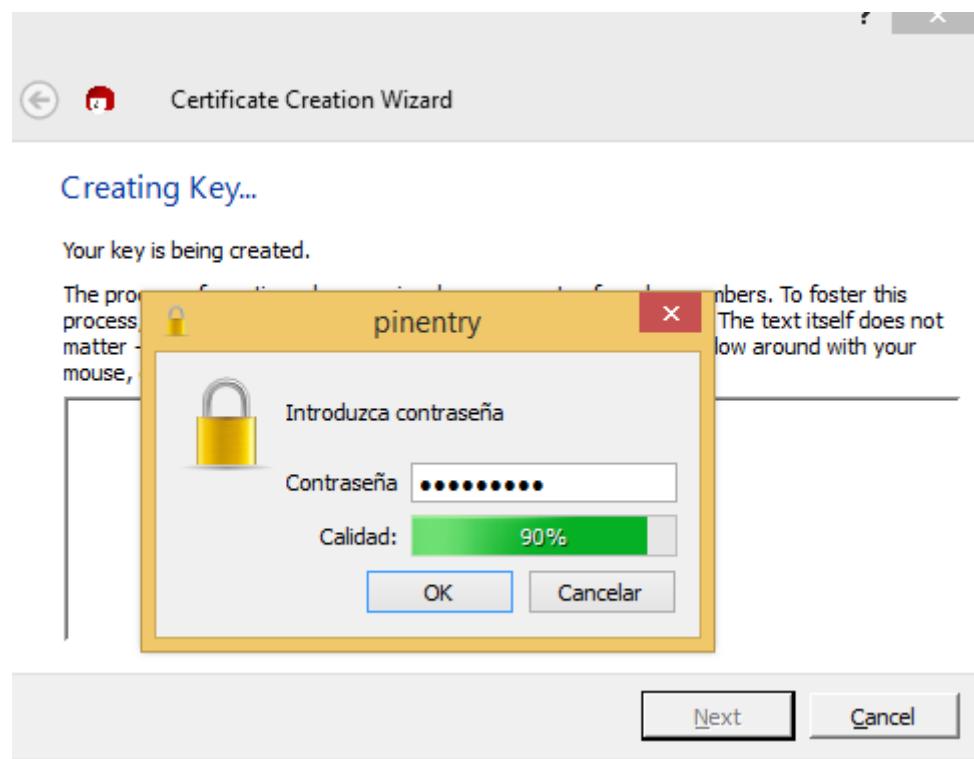
C:\Users\adrian\Documents>type mensaxe.txt.gpg
gpg: Signature made on Mon Feb 25 14:35:25 2016 -0300
gpg: using RSA key 5K:m1F%g0tJMjT=s4Qh=qly+4eCfOR+oJ7oI1uE:
C:\Users\adrian\Documents>
C:\Users\adrian\Documents>
C:\Users\adrian\Documents>
C:\Users\adrian\Documents>gpg --decrypt mensaxe.txt.gpg
gpg: datos cifrados CAST5
gpg: cifrado con 1 contraseña
"Proba de contenido de arquivo"
gpg: ATENCI N: la integridad del mensaje no est  protegida

C:\Users\adrian\Documents>_
```

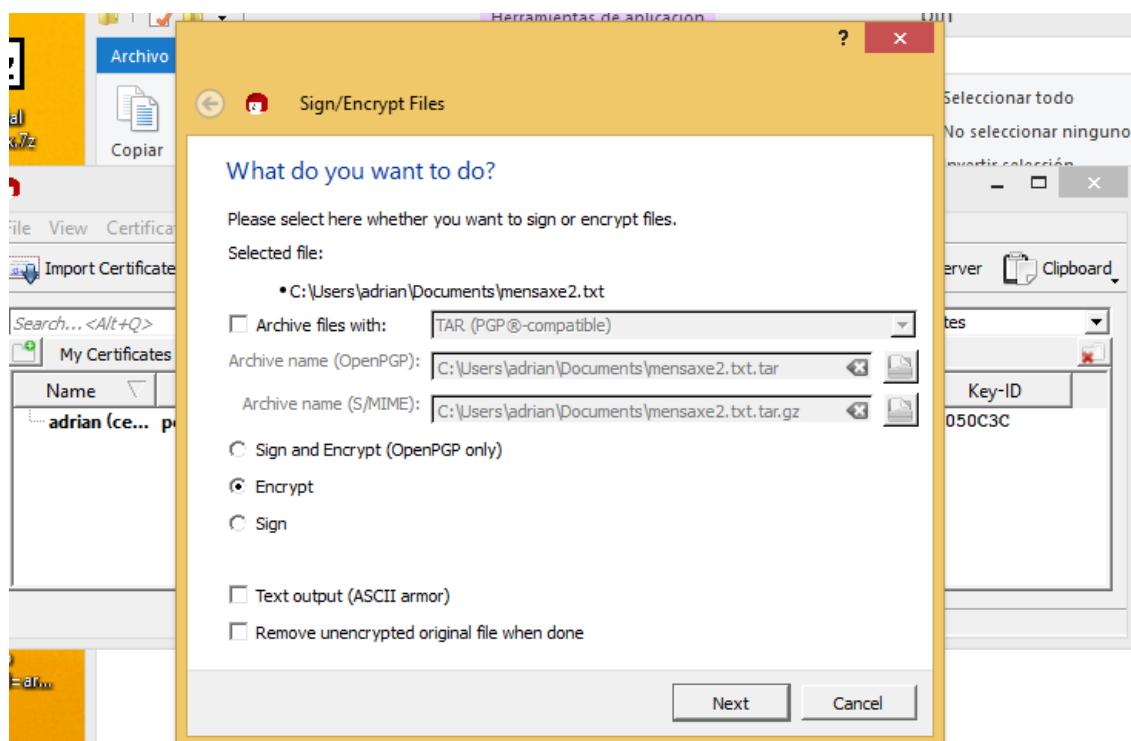
Tamén o podemos facer de forma gráfica, cifrando un arquivo con un certificado. Creamos o certificado.



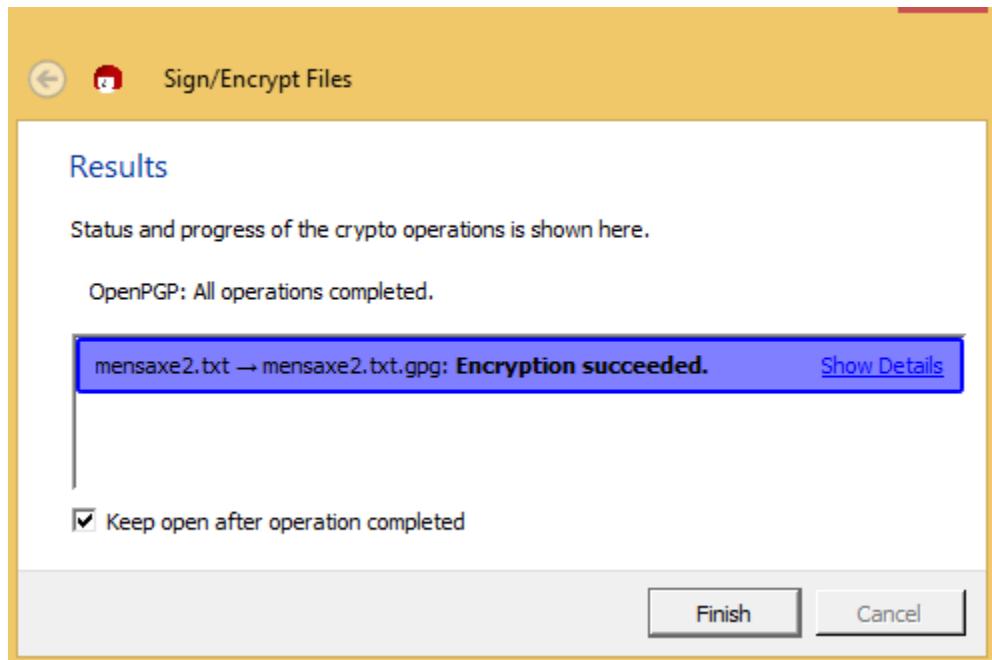
Establecemos unha contrasinal o certificado.



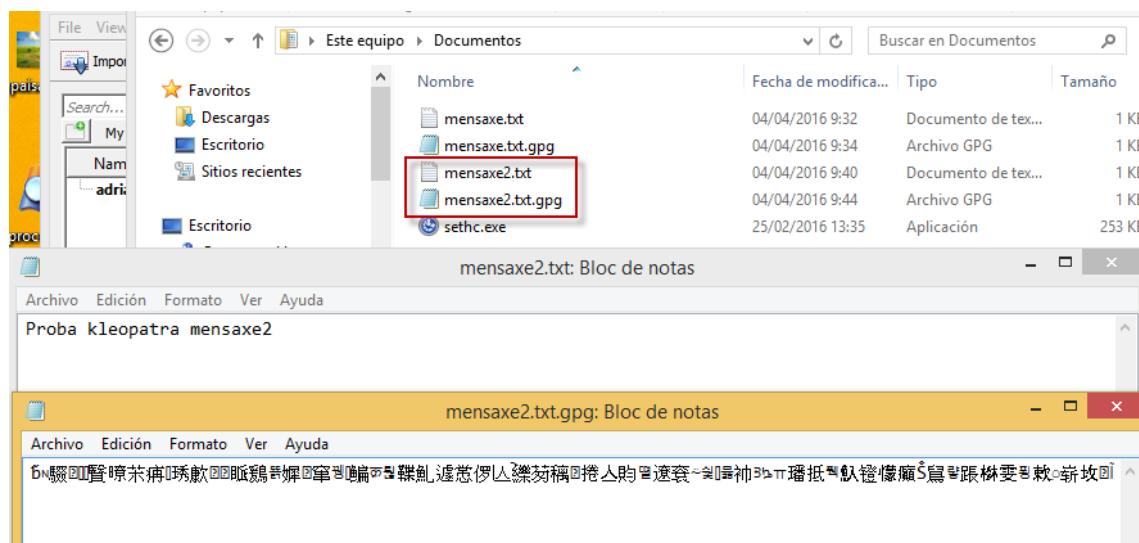
Co certificado creado, só nos quedará cifrar o documento co certificado creado.



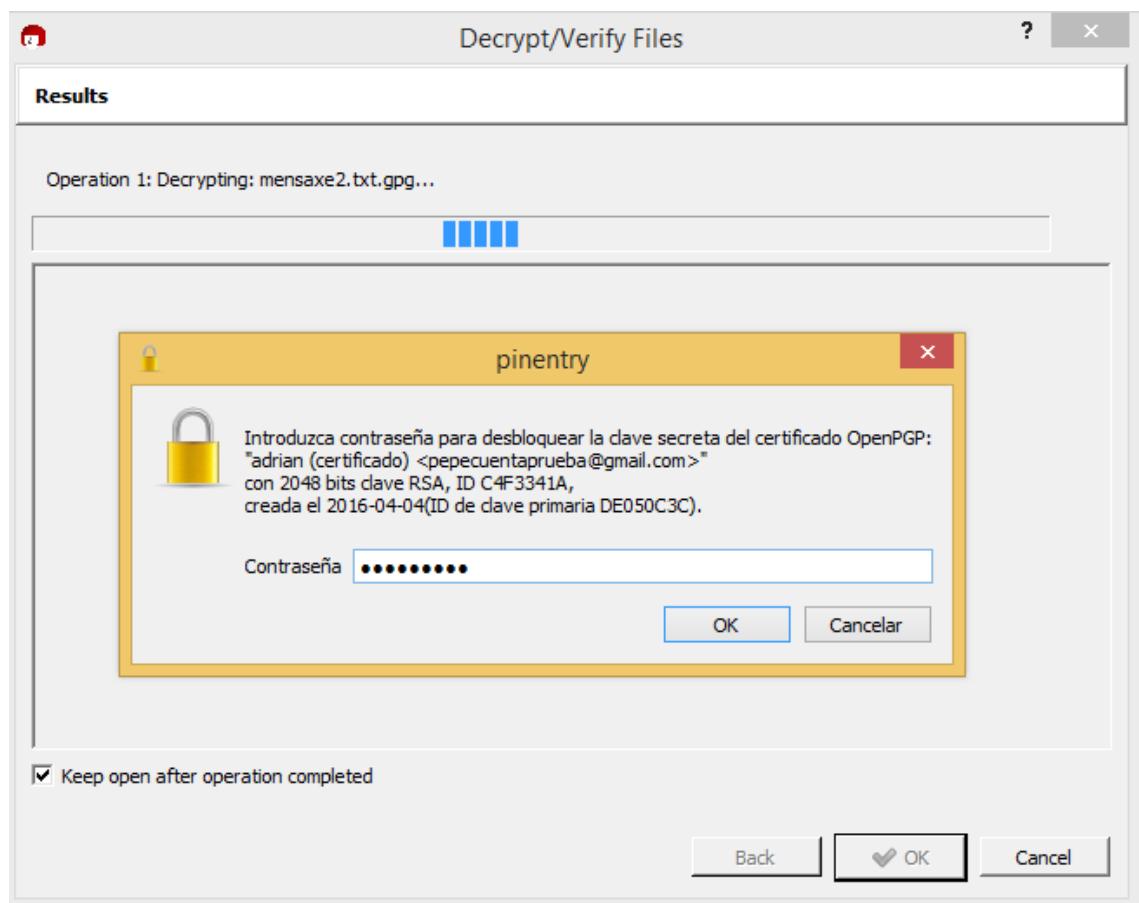
O arquivo cifrouse correctamente.



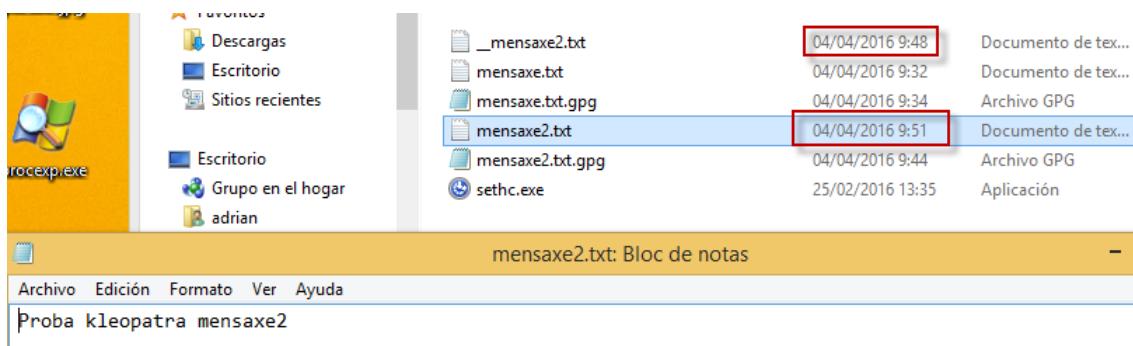
Vemos como e resultado e o mesmo, establecendo o ficheiro con extensión .gpg a cal non está en texto plano. O tipo de cifrado estaría definido en base o que hubesmos elixido na creación do certificado.



Para descifralo faremos o mesmo paso que para cifralo en Kleopatra, pero eliximos a opción “Decrypt”. Pedirános o contrasinal establecido no certificado creado.



Vemos como o descifrado funciona correctamente, como tería que ser nun cifrado simétrico.



¿Preguntas?

Podemos descifrar un arquivo cifrado, se lle cambiamos o nome ou a extensión?

Sí. Os cambios de nome non alteran o cifrado do ficheiro.

Podése comprobar na seguinte captura.

```
root@seadserver:/# gpg# cp mensaxe.txt.asc mensa.ttt
root@seadserver:/# cp mensaxe.txt.gpg mensa2.bbb
root@seadserver:/# ls -la
total 28
drwxr-xr-x 2 root root 4096 abr  4 00:07 .
drwxr-xr-x 24 root root 4096 abr  3 23:27 ..
-rw-r--r--  1 root root   90 abr  4 00:07 mensa2.bbb
-rw-r--r--  1 root root  201 abr  4 00:06 mensa.ttt
-rw-r--r--  1 root root   40 abr  3 23:29 mensaxe.txt
-rw-r--r--  1 root root  201 abr  3 23:31 mensaxe.txt.asc
-rw-r--r--  1 root root   90 abr  3 23:30 mensaxe.txt.gpg
root@seadserver:/# gpg -d mensa.ttt
gpg: datos cifrados CAST5
gpg: el agente gpg no esta disponible en esta sesion
gpg: cifrado con 1 frase contraseña
Esto e unha proba de texto, tarefa sead
gpg: AVISO: la integridad del mensaje no esta protegida
root@seadserver:/# gpg -d mensa2.bbb
gpg: datos cifrados CAST5
gpg: el agente gpg no esta disponible en esta sesion
gpg: cifrado con 1 frase contraseña
Esto e unha proba de texto, tarefa sead
gpg: AVISO: la integridad del mensaje no esta protegida
root@seadserver:/# gpg#
```

**Podemos especificar, no cifrado simétrico, algún outro algoritmo de encriptado distinto?
Como sería?**

Si. Co modificador “--cipher-algo”. Un exemplo sería:

gpg --cipher-algo {TipoCifrado} -c {Arquivo}

```
root@seadserver:/gpg# gpg --version
gpg (GnuPG) 1.4.16
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://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: ~/.gnupg
Algoritmos disponibles:
Clave pública: RSA, RSA-E, RSA-S, ELG-E, DSA
Cifrado: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH,
          CAMELLIA128, CAMELLIA192, CAMELLIA256
Resumen: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
Compresión: Sin comprimir, ZIP, ZLIB, BZIP2
root@seadserver:/gpg# gpg --cipher-algo AES256 -c mensaxe2.txt
gpg: el agente gpg no está disponible en esta sesión
root@seadserver:/gpg# ls
mensa2.bbb  mensaxe2.txt      mensaxe.txt      mensaxe.txt.gpg
mensa.ttt  mensaxe2.txt.gpg  mensaxe.txt.asc
root@seadserver:/gpg# cat mensaxe2.txt.gpg
♦      %***@**`♦iM***U*y*u**0e*gzU♦-P8*D**&X*P'F♦,*****pP**y**S*C****D*****  4
♦b*dt*HM*8bV*^R*-"
♦[♦root@seadserver:/gpg# _
```

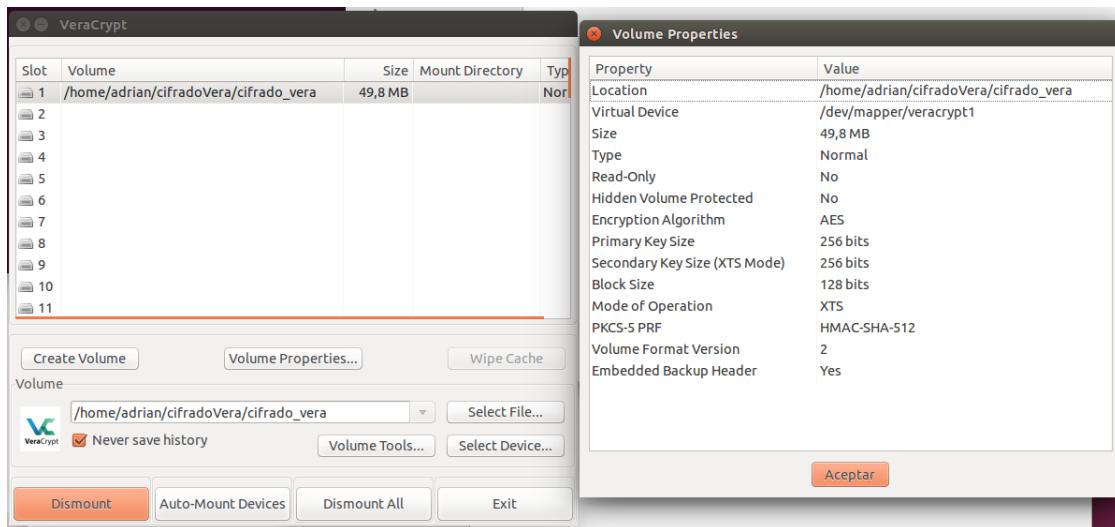
Cal é o algoritmo de cifrado aplicado por defecto dun cifrado simétrico para gpg?

CAST5.

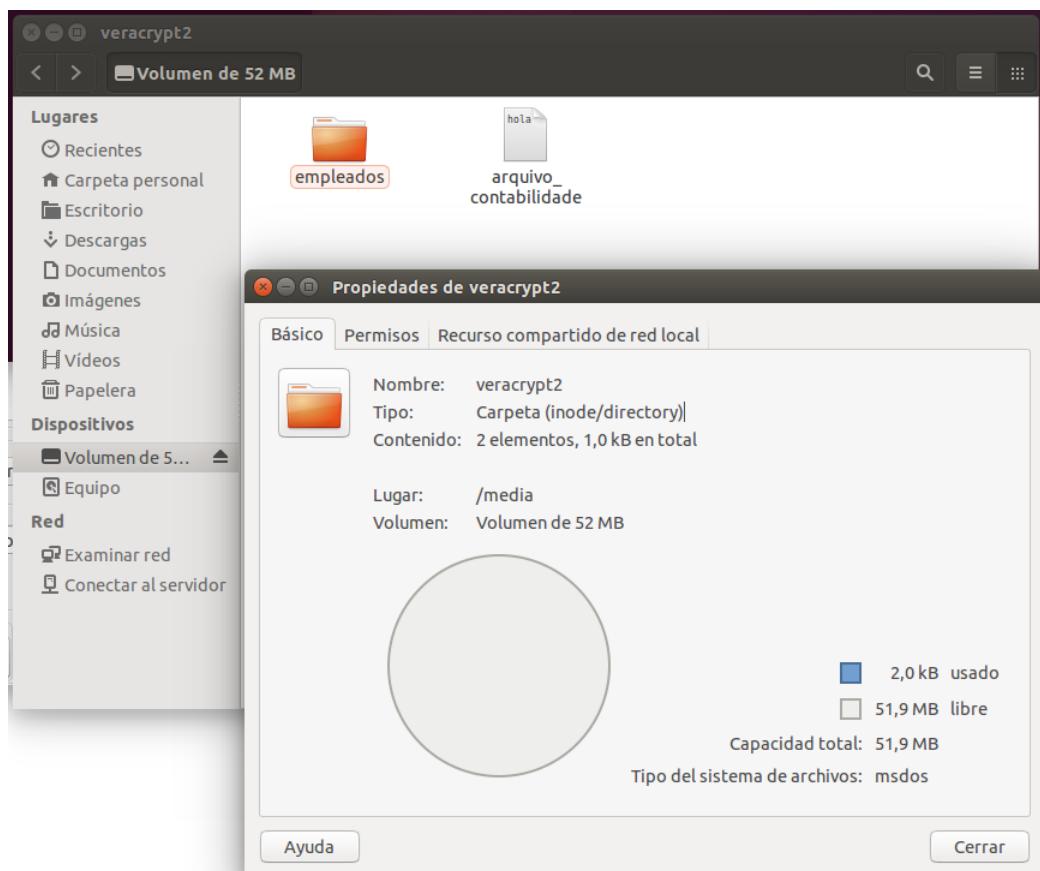
4. Cifrado de datos y particiones

Creamos un archivo que será o noso contenedor cifrado. Dito arquivo asignámoslle un tamaño de 50MB, por exemplo.

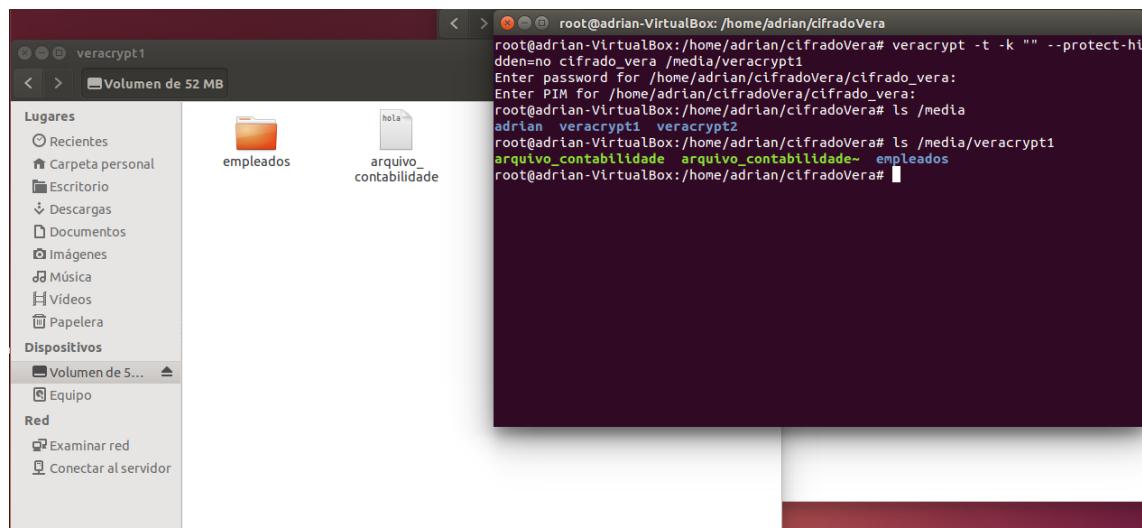
E montámolo con VeraCrypt. Para poder montalo pediranos unha contrasinal a cal establecimos previamente na súa elaboración. Estas capturas son simplemente os resultados finais.



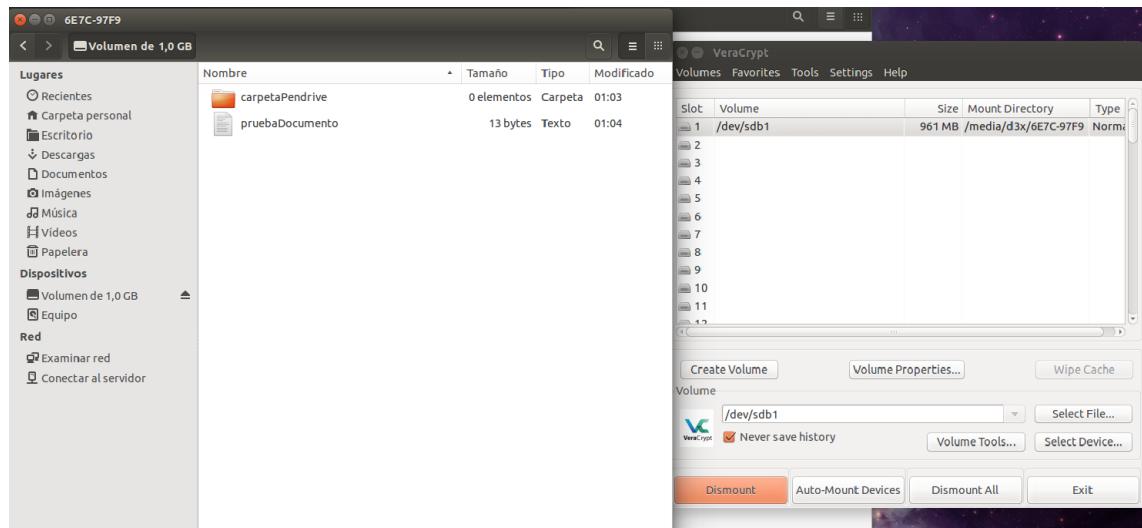
Unha vez montado podemos ver o seu contido.



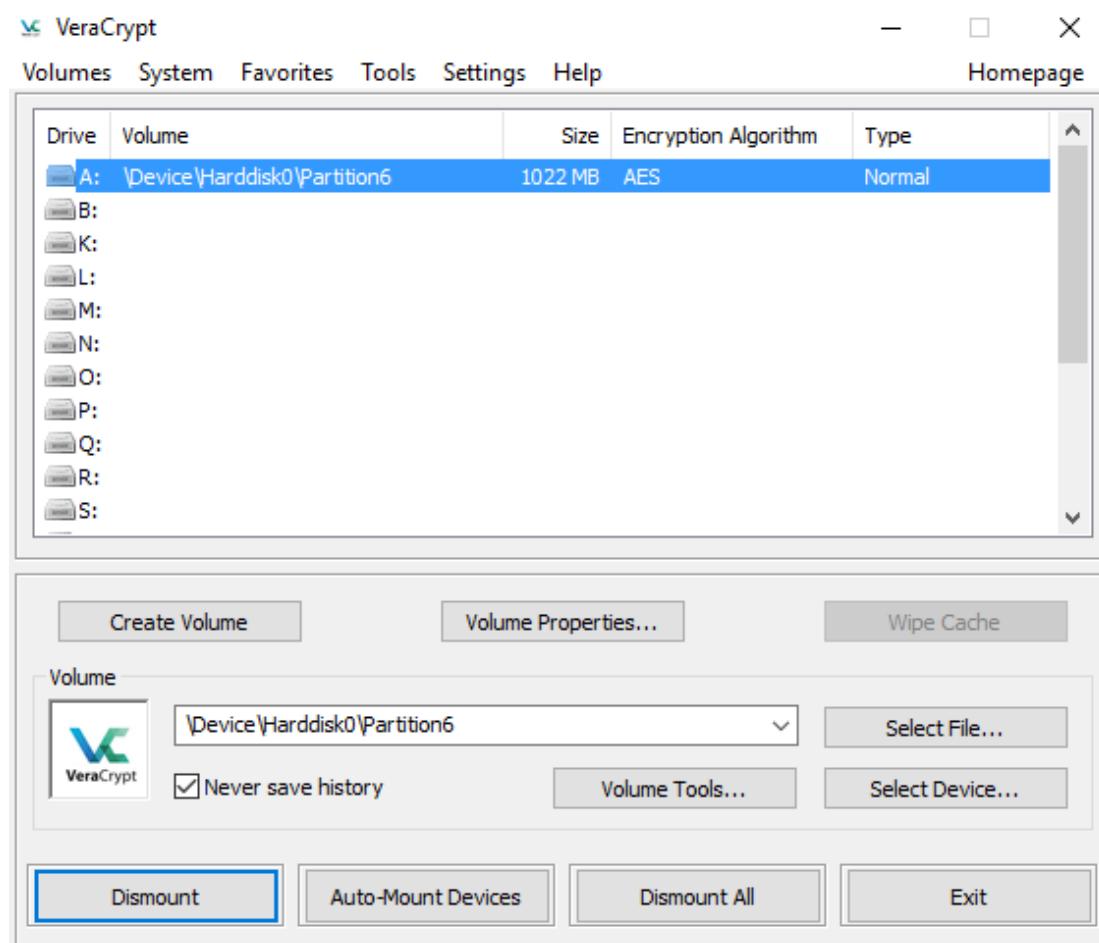
Podémolo montar a través de liñas de comandos, moi útil se non temos interfaz gráfica.



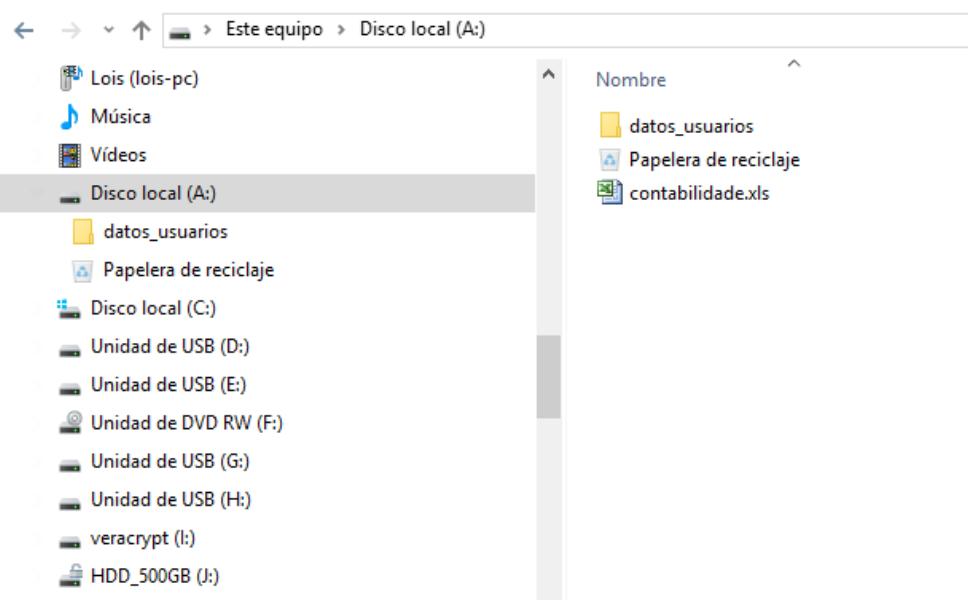
A continuación facemos o mesmo, pero en vez de crear un “archivo-contenedor”, farémolo cun **pendrive de 1GB** cifrando todo o pendrive. Esta sería unha captura final dun volume de tamaño dun 1GB montado como volume.



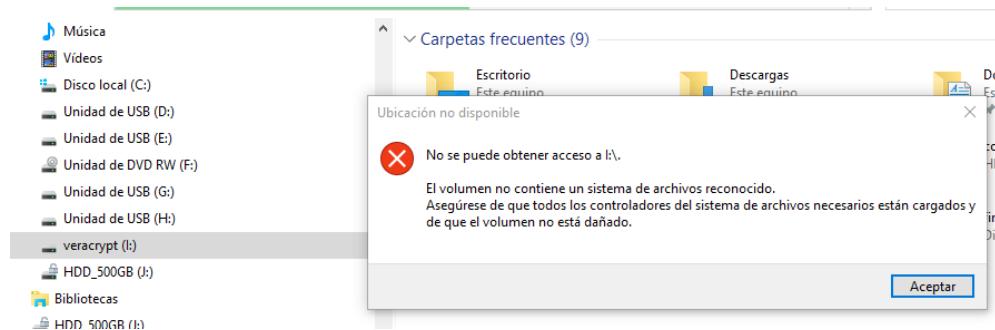
Faremos o mesmo en Windows que co pendrive pero en vez de ser un pendrive será un partición dun 1GB aprox. A cal destinaremos como contenedor cifrado. A cal montaremos e desmontaremos do noso sistema a través dunha contrasinal.



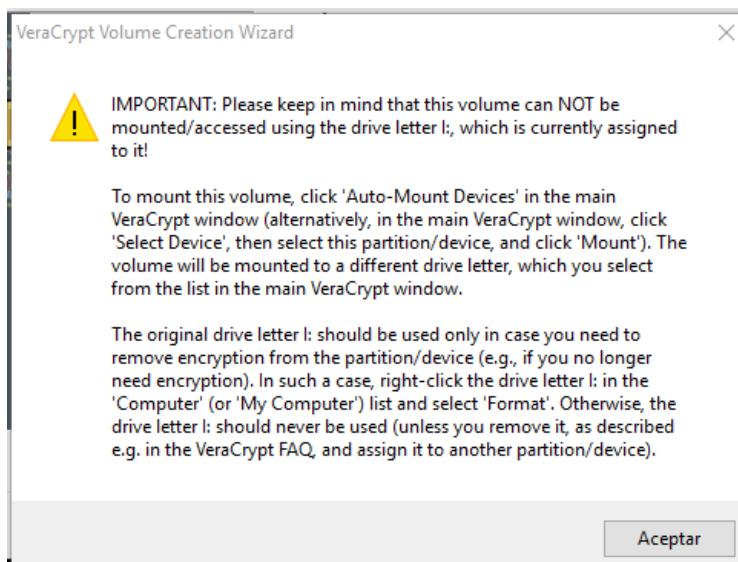
Así veríase nun explorador de Windows, sería igual que ter esa partición ca diferencia de que estaría cifrada e se podería desmontar e montar sempre que se queira con VeraCrypt.



Si non montamos a partición en VeraCrypt esto e que nos mostraría si tentamos acceder a ela.



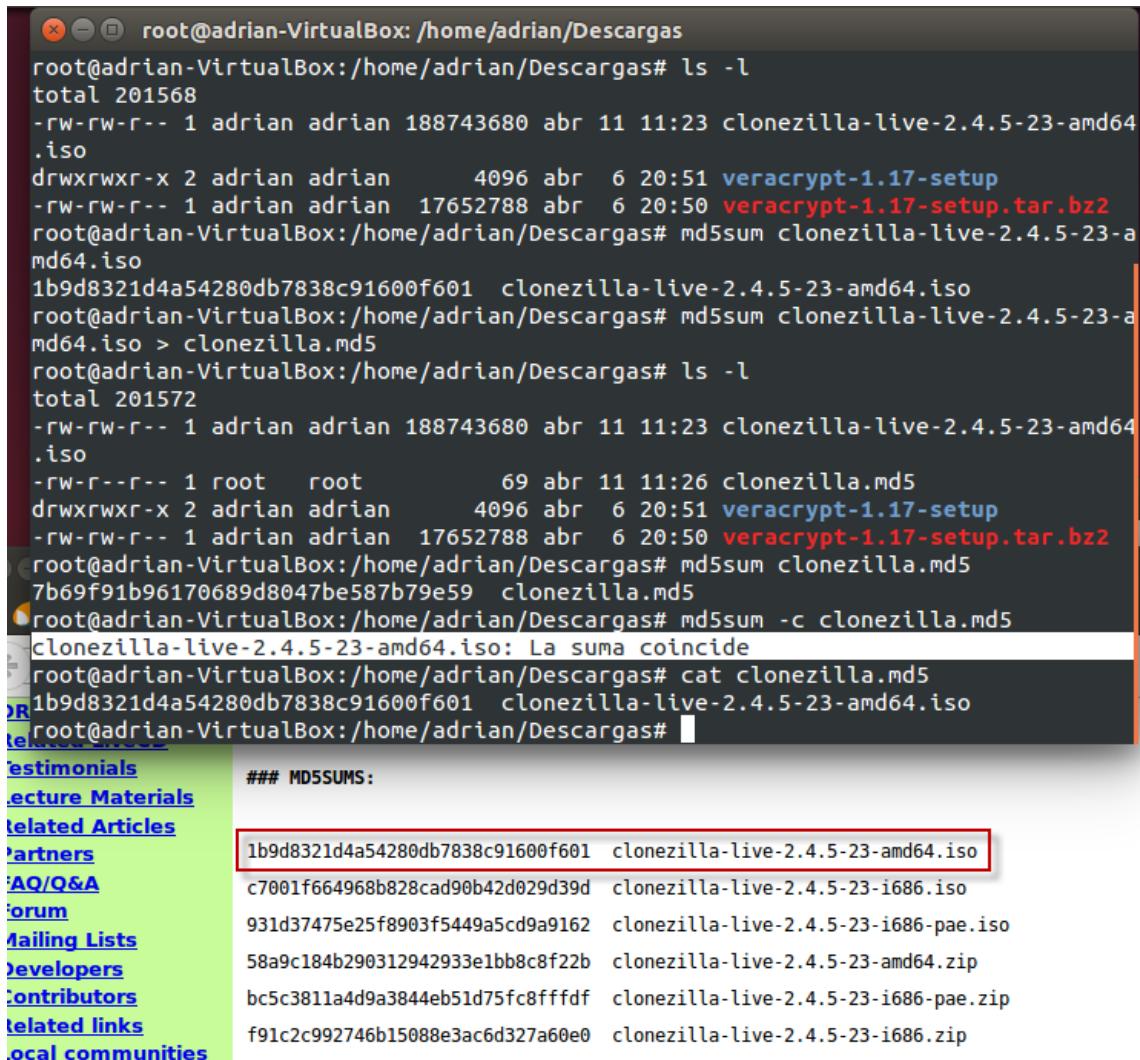
Como detalle aclarar que a letra de asignación non debe usarse a mesma para montar o volume de veracrypt. Para borrar o cifrado teremos que formatear o volume orixinal xa que a virtualizada non permite formateo.



5. Funcións resumen

Comprobando as funcións resumen nos diferentes hashes: MD5, SHA1 e SHA256.

md5sum



```
root@adrian-VirtualBox: /home/adrian/Descargas
root@adrian-VirtualBox:/home/adrian/Descargas# ls -l
total 201568
-rw-rw-r-- 1 adrian adrian 188743680 abr 11 11:23 clonezilla-live-2.4.5-23-amd64.iso
drwxrwxr-x 2 adrian adrian 4096 abr 6 20:51 veracrypt-1.17-setup
-rw-rw-r-- 1 adrian adrian 17652788 abr 6 20:50 veracrypt-1.17-setup.tar.bz2
root@adrian-VirtualBox:/home/adrian/Descargas# md5sum clonezilla-live-2.4.5-23-amd64.iso
1b9d8321d4a54280db7838c91600f601 clonezilla-live-2.4.5-23-amd64.iso
root@adrian-VirtualBox:/home/adrian/Descargas# md5sum clonezilla-live-2.4.5-23-amd64.iso > clonezilla.md5
root@adrian-VirtualBox:/home/adrian/Descargas# ls -l
total 201572
-rw-rw-r-- 1 adrian adrian 188743680 abr 11 11:23 clonezilla-live-2.4.5-23-amd64.iso
-rw-r--r-- 1 root root 69 abr 11 11:26 clonezilla.md5
drwxrwxr-x 2 adrian adrian 4096 abr 6 20:51 veracrypt-1.17-setup
-rw-rw-r-- 1 adrian adrian 17652788 abr 6 20:50 veracrypt-1.17-setup.tar.bz2
root@adrian-VirtualBox:/home/adrian/Descargas# md5sum clonezilla.md5
7b69f91b96170689d8047be587b79e59 clonezilla.md5
root@adrian-VirtualBox:/home/adrian/Descargas# md5sum -c clonezilla.md5
clonezilla-live-2.4.5-23-amd64.iso: La suma coincide
root@adrian-VirtualBox:/home/adrian/Descargas# cat clonezilla.md5
1b9d8321d4a54280db7838c91600f601 clonezilla-live-2.4.5-23-amd64.iso
root@adrian-VirtualBox:/home/adrian/Descargas#
```

MD5SUMS:

1b9d8321d4a54280db7838c91600f601	clonezilla-live-2.4.5-23-amd64.iso
c7001f664968b828cad90b42d029d39d	clonezilla-live-2.4.5-23-i686.iso
931d37475e25f8903f5449a5cd9a9162	clonezilla-live-2.4.5-23-i686-pae.iso
58a9c184b290312942933e1bb8c8f22b	clonezilla-live-2.4.5-23-amd64.zip
bc5c3811a4d9a3844eb51d75fc8ffdf	clonezilla-live-2.4.5-23-i686-pae.zip
f91c2c992746b15088e3ac6d327a60e0	clonezilla-live-2.4.5-23-i686.zip

sha1sum e sha256sum

```

root@adrian-VirtualBox:/home/adrian/Descargas
root@adrian-VirtualBox:/home/adrian/Descargas# ls -l
total 201572
-rw-rw-r-- 1 adrian adrian 188743680 abr 11 11:23 clonezilla-live-2.4.5-23-amd64
.iso
-rw-r--r-- 1 root root 69 abr 11 11:26 clonezilla.md5
drwxrwxr-x 2 adrian adrian 4096 abr 6 20:51 veracrypt-1.17-setup
-rw-rw-r-- 1 adrian adrian 17652788 abr 6 20:50 veracrypt-1.17-setup.tar.bz2
root@adrian-VirtualBox:/home/adrian/Descargas# sha1sum clonezilla-live-2.4.5-23-
amd64.iso
09ff8db76992ea82568ecf71c31ba0e4b1d0700d clonezilla-live-2.4.5-23-amd64.iso
root@adrian-VirtualBox:/home/adrian/Descargas# sha256sum clonezilla-live-2.4.5-2
3-amd64.iso
21810e30f566e28f18cf08037d8dd8c35f6c0606384889eeb43fc52d8805a384 clonezilla-liv
e-2.4.5-23-amd64.iso
root@adrian-VirtualBox:/home/adrian/Descargas#

```

SHA1SUMS:

```

09ff8db76992ea82568ecf71c31ba0e4b1d0700d clonezilla-live-2.4.5-23-amd64.iso
dbb2e850e153e6d44a85177e626a74ef57bd3921 clonezilla-live-2.4.5-23-i686.iso
9cd6e5d43a6cce82b9d7d4a34750a2ba6936e5d0 clonezilla-live-2.4.5-23-i686-pae.iso
77e0babfd165de828afc1f6ebd12e54f0f98f161 clonezilla-live-2.4.5-23-amd64.zip
a6c07a535ca9058a7aae6786653e9466c4a6db4c clonezilla-live-2.4.5-23-i686-pae.zip
62516884e0aa85155c14ff398fb042e310011256 clonezilla-live-2.4.5-23-i686.zip

```

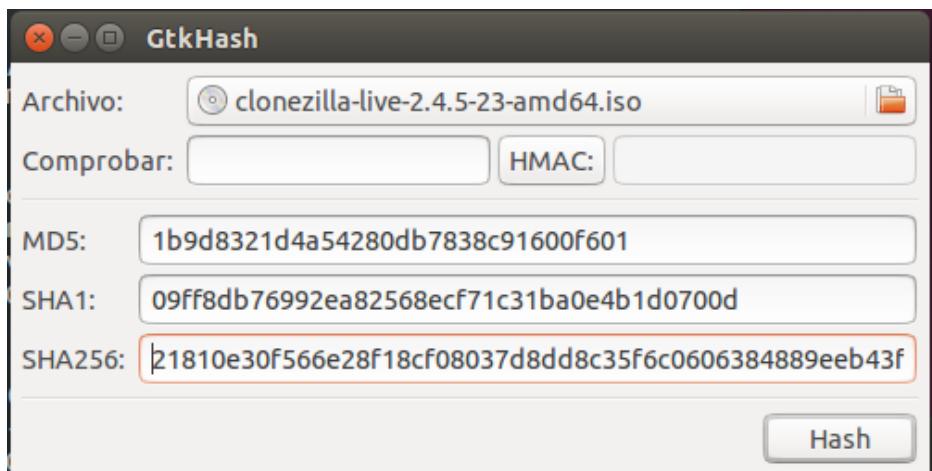
SHA256SUMS:

```

21810e30f566e28f18cf08037d8dd8c35f6c0606384889eeb43fc52d8805a384 clonezilla-live-2.4.5-23-amd64.iso
c0d28bcd02c0e2f3e7903c3548d3d11038b775a684711d9e1ada5aa8b523c39 clonezilla-live-2.4.5-23-i686.iso
a1fd3a86cbc66d6c2829e9843e60b460e6e3740026eb94197c7f883291efc2dc clonezilla-live-2.4.5-23-i686-pae.iso
2dc64bf5bdac48bea2188b12c1a3e6f5df8e7ac1b528683765dd4a2c7d69b3b6 clonezilla-live-2.4.5-23-amd64.zip
009e044e606c6c2270caca4913730a396ec0871e1422a8fe05502f486b49fc3b clonezilla-live-2.4.5-23-i686-pae.zip
fe9a6d9ee656f4600130a202b817066a60012035b8d659c7282a4baflee74669 clonezilla-live-2.4.5-23-i686.zip

```

Funcións resumen con interfaz gráfica, GtkHash. Esta sxtela utilidade móstranos os tres hashes más usados para a integridade dun ficheiro



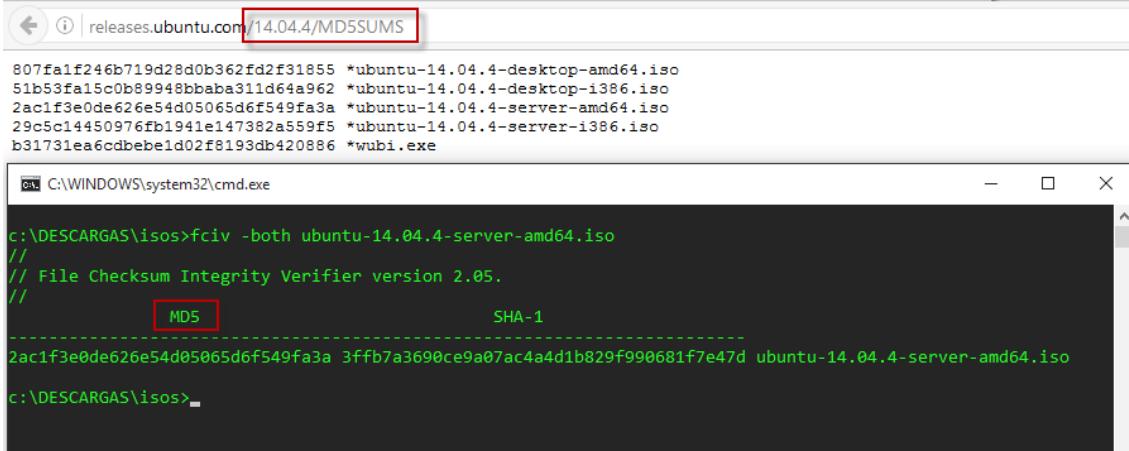
No caso de Windows temos distintas ferramentas para a comprobación de integridade dos arquivos.

Deixo unha ligazón de referencia a un artículo relacioanado con isto:

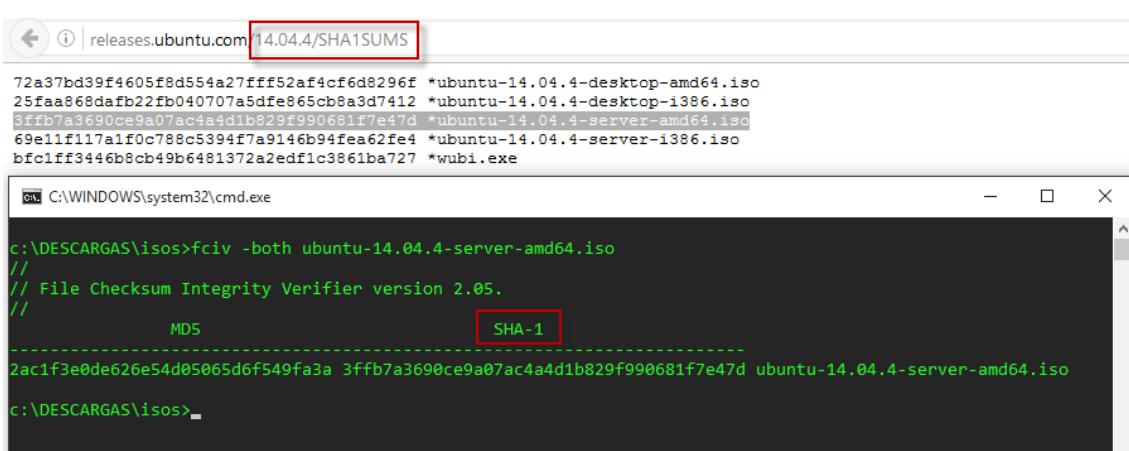
<http://www.zonasytem.com/2012/07/calcular-o-comprobar-los-checksum-crc.html>

Windows probaremos con FCIV (*File Checksum Integrity Verifier*) onde o seu nome xa o dice todo.

Comprobaremos os checksum MD5 e SHA1. (usaremos “-both” para que nos mostre o resultado dos dous a vez).



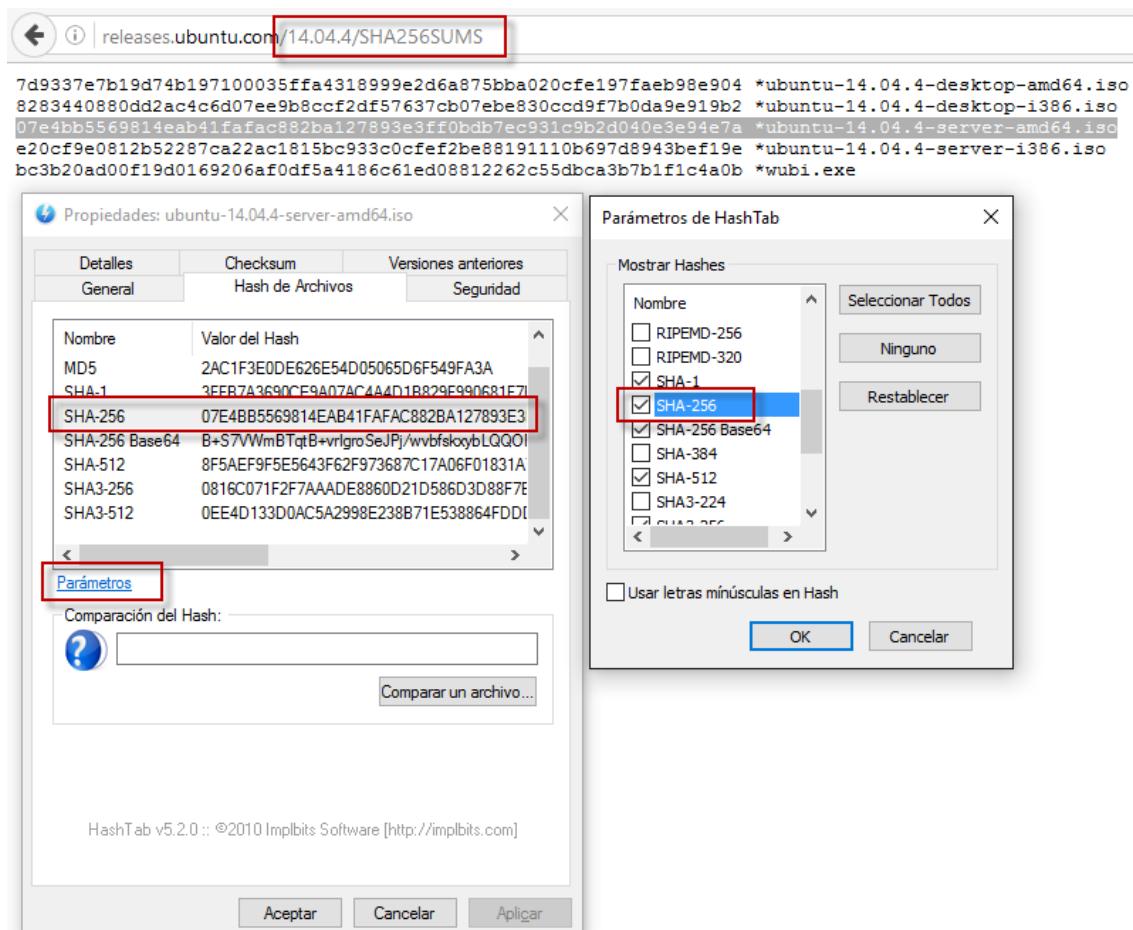
```
c:\DESCARGAS\isos>fciv -both ubuntu-14.04.4-server-amd64.iso
// File Checksum IntegrityVerifier version 2.05.
//          MD5           SHA-1
2ac1f3e0de626e54d05065d6f549fa3a 3ffb7a3690ce9a07ac4a4d1b829f990681f7e47d  ubuntu-14.04.4-server-amd64.iso
c:\DESCARGAS\isos>
```

```
c:\DESCARGAS\isos>fciv -both ubuntu-14.04.4-server-amd64.iso
// File Checksum IntegrityVerifier version 2.05.
//          MD5           SHA-1
2ac1f3e0de626e54d05065d6f549fa3a 3ffb7a3690ce9a07ac4a4d1b829f990681f7e47d  ubuntu-14.04.4-server-amd64.iso
c:\DESCARGAS\isos>
```

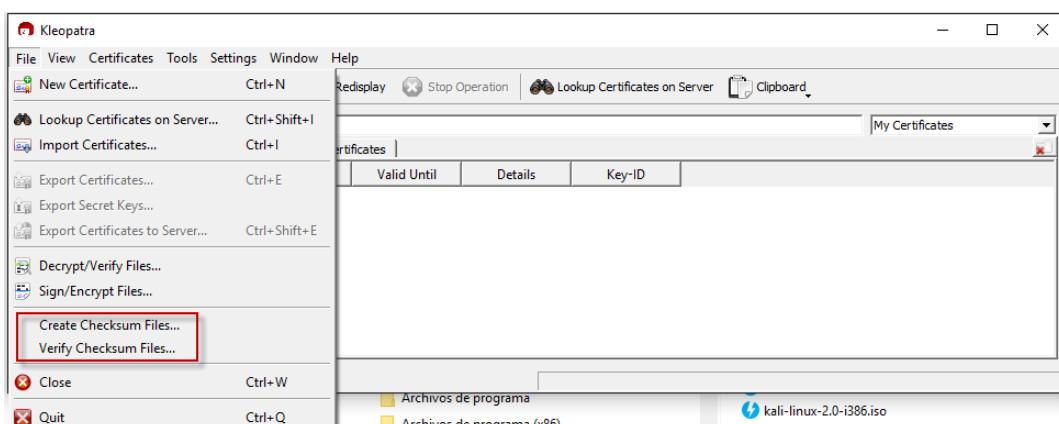
Que problemas presenta o fciv? Como o solucionamos?

So nos mostra o MD5 e o SHA-1., non hai posibilidade que nos mostre o SHA-2, neste caso podemos solucionar con HashTab que se engande como unha solapa no menú de propiedades dun arquivo e seleccionar SHA256 ou SHA-3, para ter dispoñibles estas funcións resumen.

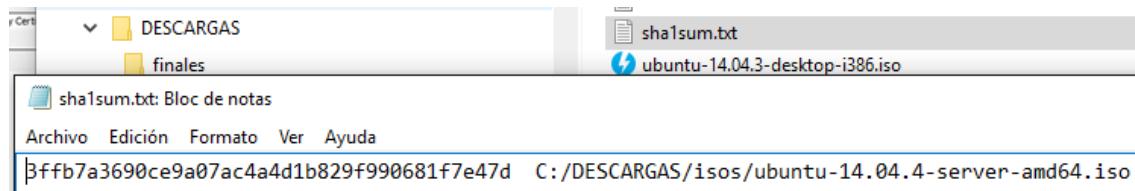


Kleopatra permite realizar e/ou verificar tamén as sumas de comprobación?

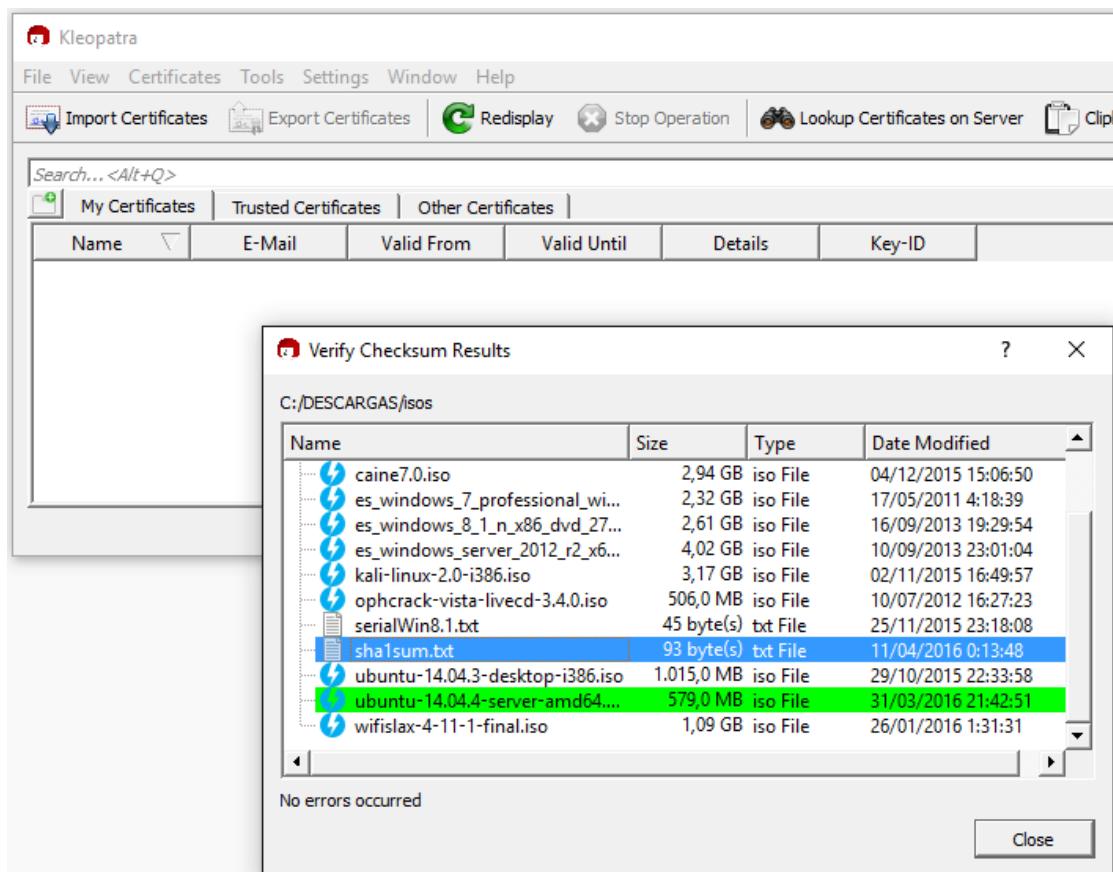
Si, permítenos verificar un checksum a si lle damos un ficheiro de texto cos resumes correctos, este simplemente verificarán que dito arquivo que lle pasamos coincide cos hashes do ficheiro que conteñen as funcións resumen.



Tendo o ficheiro creado co checksum.



Pasámoslo a Kleopatra para que o compare, neste caso ca .iso de ubuntu. Si non se produce ningún erro e se marca en cor verde, entonces e que hai total integridade nese ficheiro polo cal indica que non haia sido modificado por terceiros.



6. Cifrado asimétrico

Primeiro de nada vamos proceder a instalar o paquete “rng-tools”, o cal axudaranos a xenerar máis entropía e tardaremos moito menos na creación de chaves con gpg.

```
GNU nano 2.2.6          Archivo: /etc/default/rng-tools          Modificado

# Configuration for the rng-tools initscript
# $Id: rng-tools.default,v 1.1.2.5 2008-06-10 19:51:37 hmh Exp $

# This is a POSIX shell fragment

# Set to the input source for random data, leave undefined
# for the initscript to attempt auto-detection. Set to /dev/null
# for the viapadlock and tpm drivers.
#HRNGDEVICE=/dev/hwrng
#HRNGDEVICE=/dev/null
HRNGDEVICE=/dev/urandom

# Additional options to send to rngd. See the rngd(8) manpage for
# more information. Do not specify -r/--rng-device here, use
# HRNGDEVICE for that instead.
#RNGOPTIONS="--hrng=intelfwh --fill-watermark=90% --feed-interval=1"
#RNGOPTIONS="--hrng=viakerne1 --fill-watermark=90% --feed-interval=1"
#RNGOPTIONS="--hrng=viapadlock --fill-watermark=90% --feed-interval=1"
#RNGOPTIONS="--hrng=tpm --fill-watermark=90% --feed-interval=1"

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y RePág. ^K Cortar Tex^C Pos actual
^X Salir ^J Justificar^W Buscar ^U Pág. Sig. ^U PegarTxt ^T Ortografía
```

Iniciamos o servicio de rng-tools.

```
root@seadserver:~# /etc/init.d/rng-tools start
Starting Hardware RNG entropy gatherer daemon: rngd.
root@seadserver:~#
```

Creanse o par de chaves con gpg.

gpg --gen-key

Gracias a rng-tools este proceso púdose axilizar bastante.

```
Es necesario generar muchos bytes aleatorios. Es una buena idea realizar
algunas otras tareas (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.
+++++
.....
gpg: clave 1A08F3BD marcada como de confianza absoluta
claves pública y secreta creadas y firmadas.

gpg: comprobando base de datos de confianza
gpg: 3 dudosas(s) necesarias, 1 completa(s) necesarias,
modelo de confianza 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: 2016-04-30
pub 2048R/1A08F3BD 2016-04-20 [[caducada: 2016-04-30]]
    Huella de clave = 6C55 7E3B 3472 BA55 C957 E4FE 7347 8E75 1A08 F3BD
uid                  Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>
sub 2048R/3CC20BBE 2016-04-20 [[caducada: 2016-04-30]]

root@seadserver:~#
```

Comprobamos que a chave pública están creada.

gpg --list-keys ou gpg -k

```
root@seadserver:~/.gnupg# gpg --list-keys
/root/.gnupg/pubring.gpg

pub  2048R/1A08F3BD 2016-04-20 [[caduca: 2016-04-30]]
uid            Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>
sub  2048R/3CC20BBE 2016-04-20 [[caduca: 2016-04-30]]

root@seadserver:~/.gnupg# _
```

Comprobamos que a chave privada está creada.

gpg --list-secret-keys

```
root@seadserver:/home/adrian# gpg --list-secret-keys
/root/.gnupg/secring.gpg

sec  2048R/1A08F3BD 2016-04-20 [[caduca: 2016-04-30]]
uid            Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>
ssb  2048R/3CC20BBE 2016-04-20

root@seadserver:/home/adrian# _
```

Exportamos tanto a chave pública como a privada

```
root@seadserver:/home/adrian# gpg --list-keys
/root/.gnupg/pubring.gpg

pub  2048R/1A08F3BD 2016-04-20 [[caduca: 2016-04-30]]
uid            Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>
sub  2048R/3CC20BBE 2016-04-20 [[caduca: 2016-04-30]]

root@seadserver:/home/adrian# gpg --list-secret-keys
/root/.gnupg/secring.gpg

sec  2048R/1A08F3BD 2016-04-20 [[caduca: 2016-04-30]]
uid            Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>
ssb  2048R/3CC20BBE 2016-04-20

root@seadserver:/home/adrian# gpg -a --output /home/adrian/clavePub.asc --export
1A08F3BD
root@seadserver:/home/adrian# gpg -a --output /home/adrian/claveSec.asc --export
--secret-key 1A08F3BD
root@seadserver:/home/adrian# ls
clavePub.asc  claveSec.asc
root@seadserver:/home/adrian# _
```

Creamos un certificado de revogación da chave

```
root@seadserver:/home/adrian# gpg --output /home/adrian/CertRevogacion_adrian.asc --gen-revoke 1A08F3BD

sec 2048R/1A08F3BD 2016-04-20 Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>

¿Crear un certificado de revocación para esta clave? (s/N) s
Por favor elija una razón para la revocación:
 0 = No se dio ninguna razón
 1 = La clave ha sido comprometida
 2 = La clave ha sido reemplazada.
 3 = La clave ya no está en uso
 Q = Cancelar
(Probablemente quería seleccionar 1 aquí)
¿Su decisión? 0
Introduzca una descripción opcional; acábelo con una línea vacía:
> Certificado de revocación
>
Razones de la revocación: No se dio ninguna razón
Certificado de revocación
¿Es correcto? (s/N) s

Necesita una frase contraseña para desbloquear la clave secreta
del usuario: "Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>""
clave RSA de 2048 bits, ID 1A08F3BD, creada el 2016-04-20

gpg: el agente gpg no esta disponible en esta sesión
Introduzca frase contraseña: _

Necesita una frase contraseña para desbloquear la clave secreta
del usuario: "Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>""
clave RSA de 2048 bits, ID 1A08F3BD, creada el 2016-04-20

gpg: el agente gpg no esta disponible en esta sesión
se fuerza salida con armadura ASCII.
Certificado de revocación creado.

Por favor consérvelo en un medio que pueda esconder; si alguien consigue
acceso a este certificado puede usarlo para inutilizar su clave.
Es inteligente imprimir este certificado y guardarlo en otro lugar, por
si acaso su medio resulta imposible de leer. Pero precaución: ¡el sistema
de impresión de su máquina podría almacenar los datos y hacerlos accesibles
a otras personas!
root@seadserver:/home/adrian# _
```

Enviamos por FTP o certificado de revogación.

```
root@seadserver:/home/adrian# ftp 10.10.100.100
Connected to 10.10.100.100.
220-FileZilla Server 0.9.53 beta
220-written by Tim Kosse (tim.kosse@filezilla-project.org)
220 Please visit https://filezilla-project.org/
Name (10.10.100.100:adrian): admin
331 Password required for admin
Password:
230 Logged on
Remote system type is UNIX.
ftp> put CertRevogacion_adrian.ase
local: CertRevogacion_adrian.ase remote: CertRevogacion_adrian.ase
200 Port command successful
150 Opening data channel for file upload to server of "/CertRevogacion_adrian.ase"
226 Successfully transferred "/CertRevogacion_adrian.ase"
589 bytes sent in 0.00 secs (15977.6 kB/s)
ftp> ls
200 Port command successful
150 Opening data channel for directory listing of "/"
-rw-r--r-- 1 ftp ftp 589 Apr 20 21:26 CertRevogacion_adrian.ase
-rw-r--r-- 1 ftp ftp 3648 Apr 19 21:57 secretJorgeBlanco.asc
226 Successfully transferred "/"
ftp> quit
421 No-transfer-time exceeded. Closing control connection.
root@seadserver:/home/adrian#
```

Enviamos a chave pública o servidor de chaves pgp.rediris.es. Comprobamos facendo unha búsqueda interna na web pgp.rediris.es a través da ChavelD ou e-mail, podemos ver como a chave está subida e existe no servidor.

```
root@seadserver:/home/adrian# gpg --send-keys --keyserver pgp.rediris.es 1A08F3BD
gpg: enviando clave 1A08F3BD a hkp servidor pgp.rediris.es
root@seadserver:/home/adrian# gpg --keyserver pgp.rediris.es --search-keys 1A08F3BD
gpg: buscando «1A08F3BD» de hkp servidor pgp.rediris.es
(1) Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>
     2048 bit RSA key 1A08F3BD, creado: 2016-04-20, [caduca: 2016-04-30]
Keys 1-1 of 1 for "1A08F3BD". Introduzca número(s), Otro, o F)in >
gpg: Interrupt caught ... exiting

root@seadserver:/home/adrian# gpg --keyserver pgp.rediris.es --search-keys pepepruebacuenta@gmail.com
gpg: buscando «pepepruebacuenta@gmail.com» de hkp servidor pgp.rediris.es
(1) Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>
     2048 bit RSA key 1A08F3BD, creado: 2016-04-20, [caduca: 2016-04-30]
Keys 1-1 of 1 for "pepepruebacuenta@gmail.com". Introduzca número(s), Otro, o F)in >
```

Como no caso anterior agora buscaremos a chave pública do seguinte e-mail, solicitado na tarefa.

```
root@seadserver:/home/adrian# gpg --keyserver pgp.rediris.es --search-keys julyo
v@gmail.com
gpg: buscando «julyov@gmail.com» de hkp servidor pgp.rediris.es
(1) Julio Mosquera González <julyov@gmail.com>
    4096 bit RSA key 88E3026E, creado: 2016-03-16, [caduca: 2021-03-15]
Keys 1-1 of 1 for "julyov@gmail.com". Introduzca número(s), O)tro, o F)in >
Introduzca número(s), O)tro, o F)in > 1
gpg: solicitando clave 88E3026E de hkp servidor pgp.rediris.es
gpg: clave 88E3026E: «Julio Mosquera González <julyov@gmail.com>» sin cambios
gpg: Cantidad total procesada: 1
gpg:           sin cambios: 1
root@seadserver:/home/adrian#
```

Descargamos a chave pública do e-mail solicitado o equipo local e listamos as chaves actuales.

```
root@seadserver:/home/adrian# gpg --keyserver pgp.rediris.es --recv-keys 88E3026
E
gpg: solicitando clave 88E3026E de hkp servidor pgp.rediris.es
gpg: clave 88E3026E: «Julio Mosquera González <julyov@gmail.com>» sin cambios
gpg: Cantidad total procesada: 1
gpg:           sin cambios: 1
root@seadserver:/home/adrian# gpg -k
/root/.gnupg/pubring.gpg

pub  2048R/1A08F3BD 2016-04-20 [[caduca: 2016-04-30]]
uid            Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>
sub  2048R/3CC20BBE 2016-04-20 [[caduca: 2016-04-30]]

pub  4096R/88E3026E 2016-03-16 [[caduca: 2021-03-15]]
uid            Julio Mosquera González <julyov@gmail.com>
sub  4096R/2B8E9D2B 2016-03-16 [[caduca: 2021-03-15]]

root@seadserver:/home/adrian#
```

Agora revogamos a chave persoal personal tanto no equipo local.

```
root@seadserver:/home/adrian# ls -l
total 12
-rw-r--r-- 1 root root 576 abr 20 21:23 CertRevogacion_adrian.ase
-rw-r--r-- 1 root root 1751 abr 20 21:00 clavePub.asc
-rw-r--r-- 1 root root 3618 abr 20 21:01 claveSec.asc
root@seadserver:/home/adrian# gpg --import CertRevogacion_adrian.ase
gpg: clave 1A08F3BD: "Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>" certificado de revocación importado
gpg: Cantidad total procesada: 1
gpg:   nuevas revocaciones de claves: 1
gpg: 3 dudosa(s) necesarias, 1 completa(s) necesarias,
modelo de confianza 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: 2016-04-30
root@seadserver:/home/adrian# gpg -k
/root/.gnupg/pubring.gpg

pub  2048R/1A08F3BD 2016-04-20 [revocada: 2016-04-20]
uid            Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>

pub  4096R/88E3026E 2016-03-16 [[caduca: 2021-03-15]]
uid            Julio Mosquera González <julyov@gmail.com>
sub  4096R/2B8E9D2B 2016-03-16 [[caduca: 2021-03-15]]

root@seadserver:/home/adrian# _
```

Revogamos a chave persoal do servidor o que se subiu (pgp.rediris.es). Comprobamos de que se revogou correctamente.

```
root@seadserver:/home/adrian# gpg --keyserver pgp.rediris.es --send-keys 1A08F3BD
gpg: enviando clave 1A08F3BD a hkp servidor pgp.rediris.es
root@seadserver:/home/adrian# gpg --keyserver pgp.rediris.es --search-keys 1A08F3BD
gpg: buscando «1A08F3BD» de hkp servidor pgp.rediris.es
(1) Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>
      2048 bit RSA key 1A08F3BD, creado: 2016-04-20, [caduca: 2016-04-30] ([revocada])
Keys 1-1 of 1 for "1A08F3BD". Introduzca número(s), Otro, o Fin >
Introduzca número(s), Otro, o Fin > 1
gpg: solicitando clave 1A08F3BD de hkp servidor pgp.rediris.es
gpg: clave 1A08F3BD: «Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>»
      sin cambios
gpg: Cantidad total procesada: 1
gpg:           sin cambios: 1
root@seadserver:/home/adrian# _
```

Podemos comprobar a través da propia web pgp.rediris.es, facendo unha búsqueda por email ou por ID de chave, como esta a sido correctamente revogada.

pgp.rediris.es:11371/pks/lookup?search=pepepruebacuenta%40gmail.com&op=index

Search results for 'pepepruebacuenta gmail com'

Type	bits/keyID	Date	User ID
pub	2048R/ 1A08F3BD	2016-04-20	*** KEY REVOKED *** [not verified] Paquito Sead (Proba con gpg) <pepepruebacuenta@gmail.com>

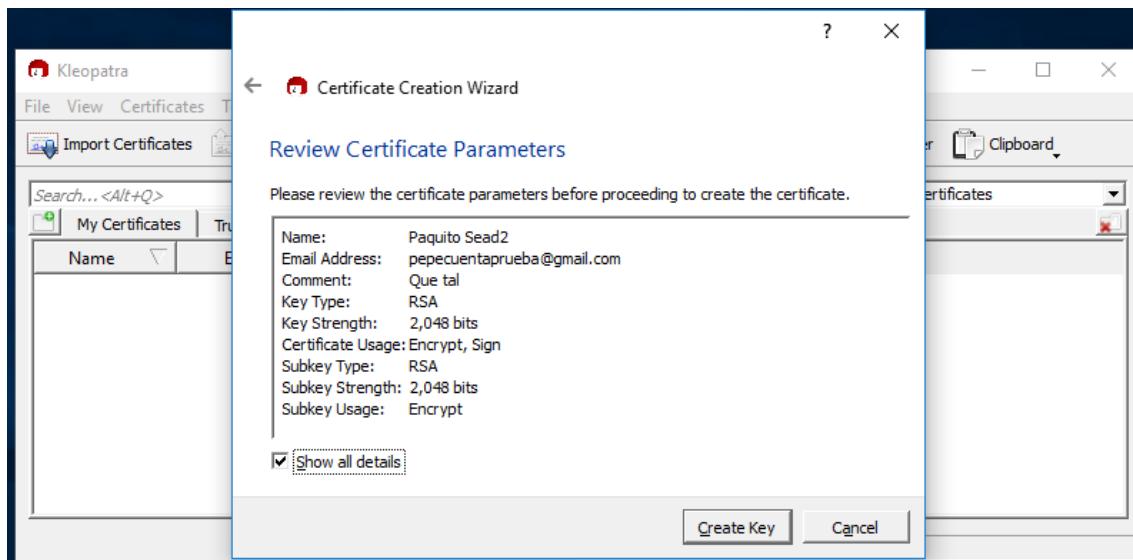
pgp.rediris.es:11371/pks/lookup?op=vindex&search=0x73478e751a08f3bd

Search results for '0x73478e751a08f3bd'

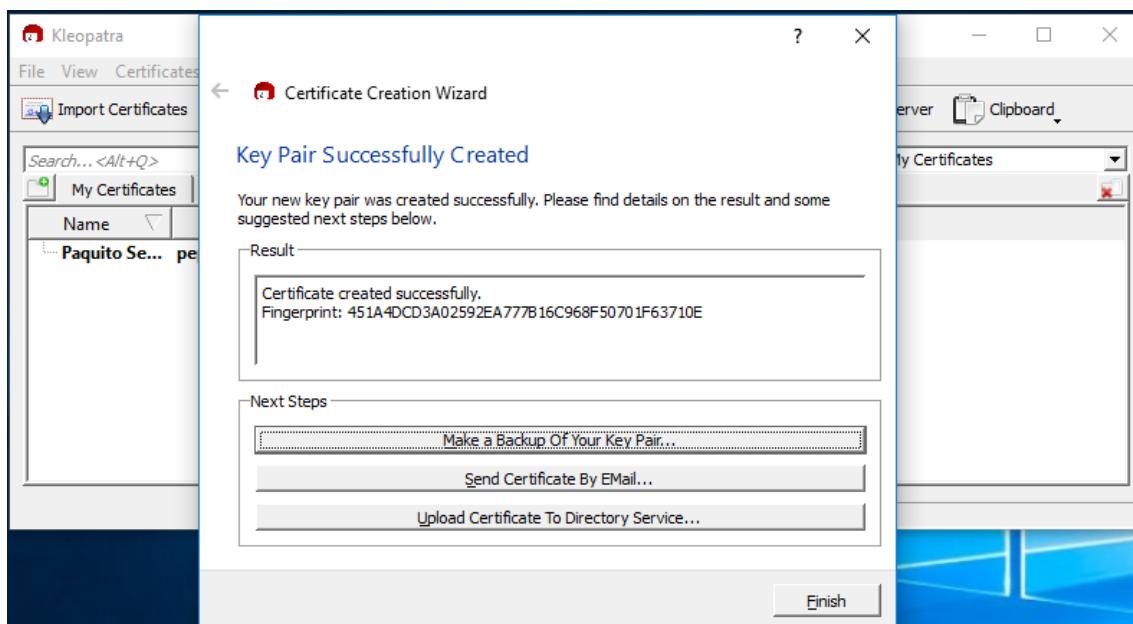
Type	bits/keyID	cr. time	exp time	key expir
pub	2048R/ 1A08F3BD	2016-04-20		
sig	revok 1A08F3BD	2016-04-20		[selfsig]
uid	Paquito Sead (Proba con gpg) < pepepruebacuenta@gmail.com >			
sig	sig3 1A08F3BD	2016-04-20	2016-04-30	[selfsig]
sub	2048R/3CC20BBE	2016-04-20		
sig	sbind 1A08F3BD	2016-04-20	2016-04-30	[]

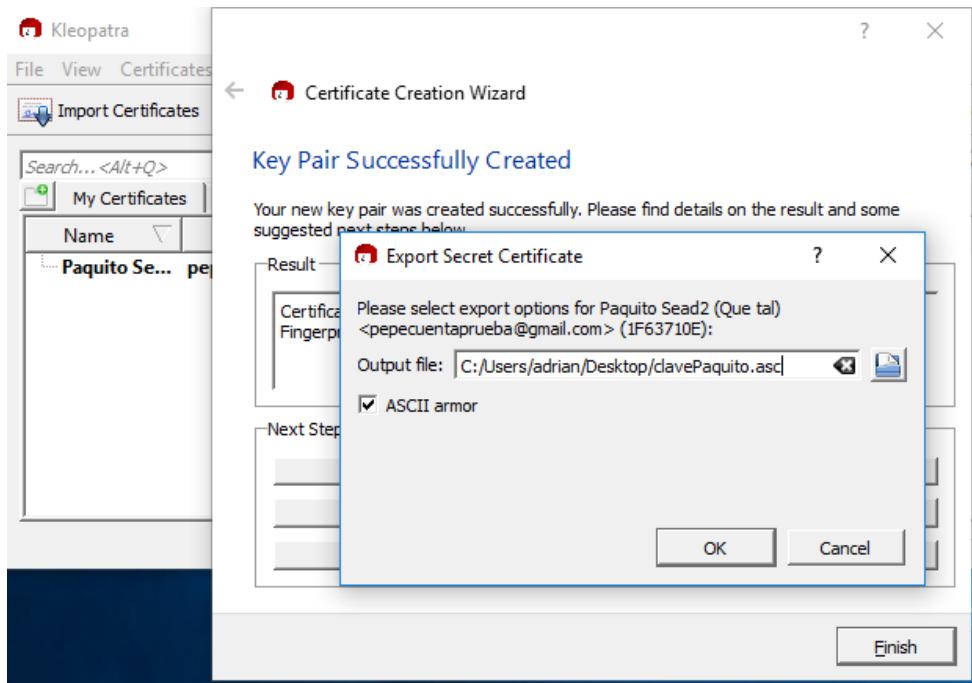
Agora faremos o mesmo pero en Windows coa interfaz gráfica Kleopatra.

Creamos o certificado que contén o par de chaves.

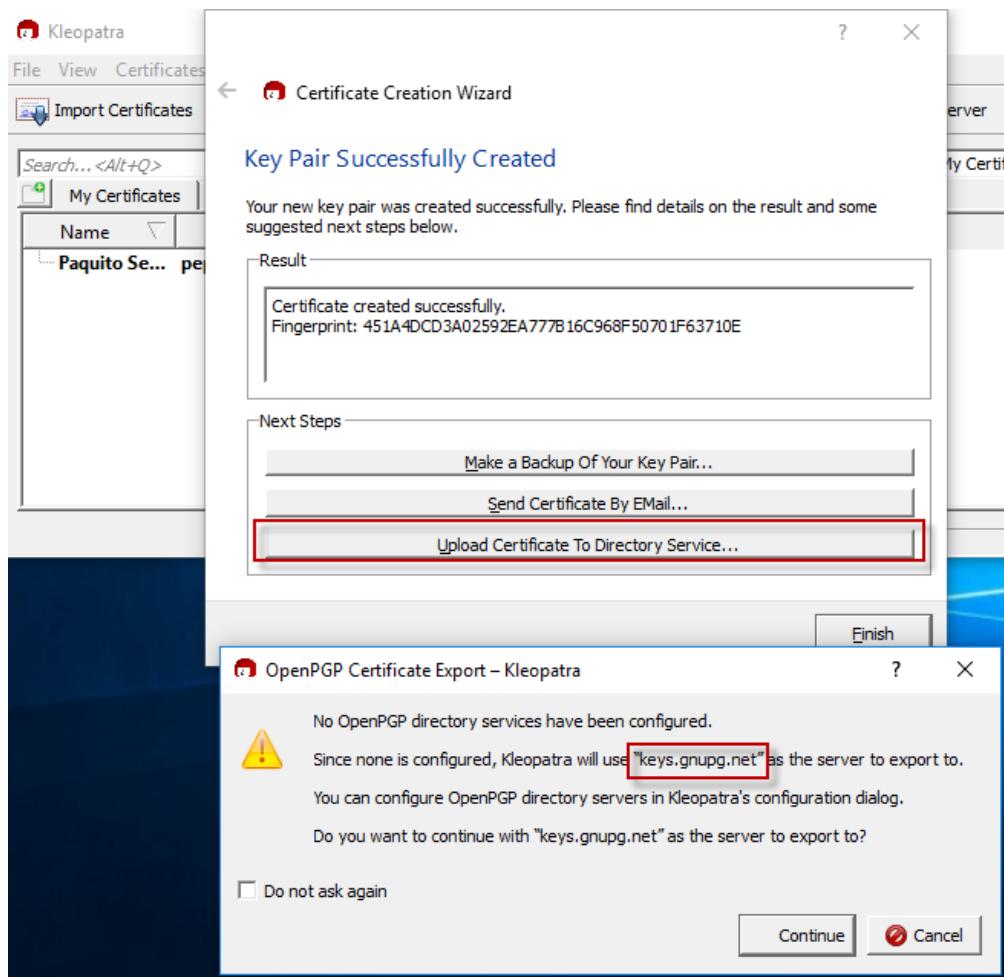


Temos as opcións de exportala a un directorio local.

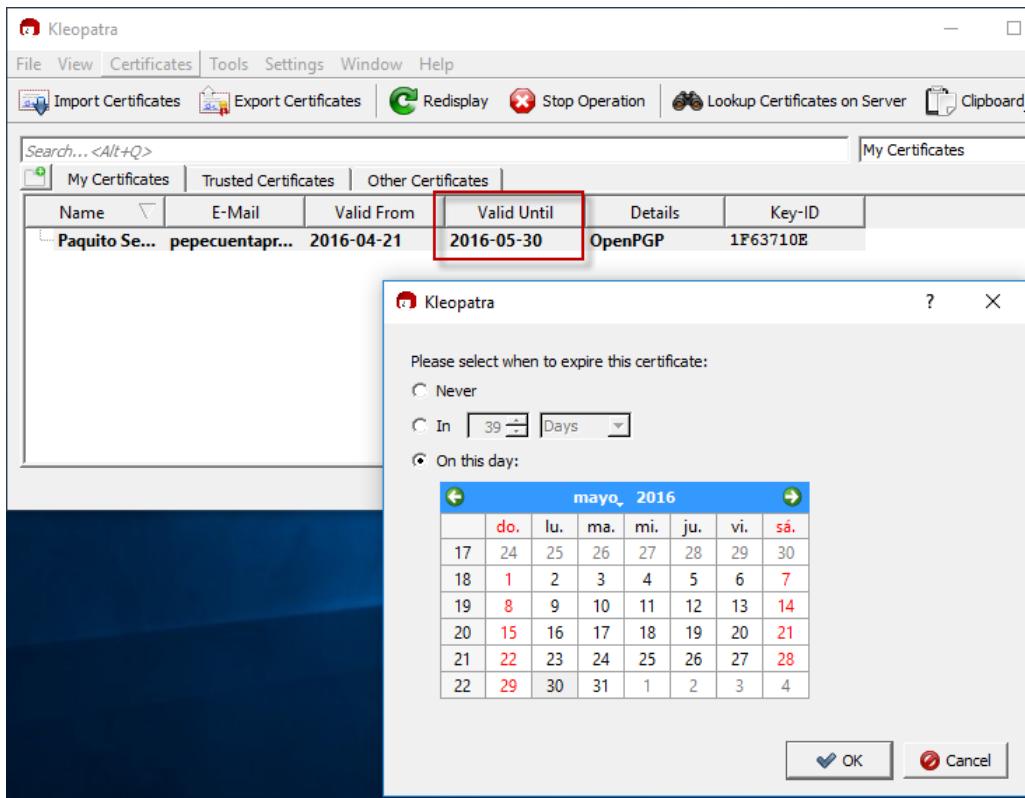




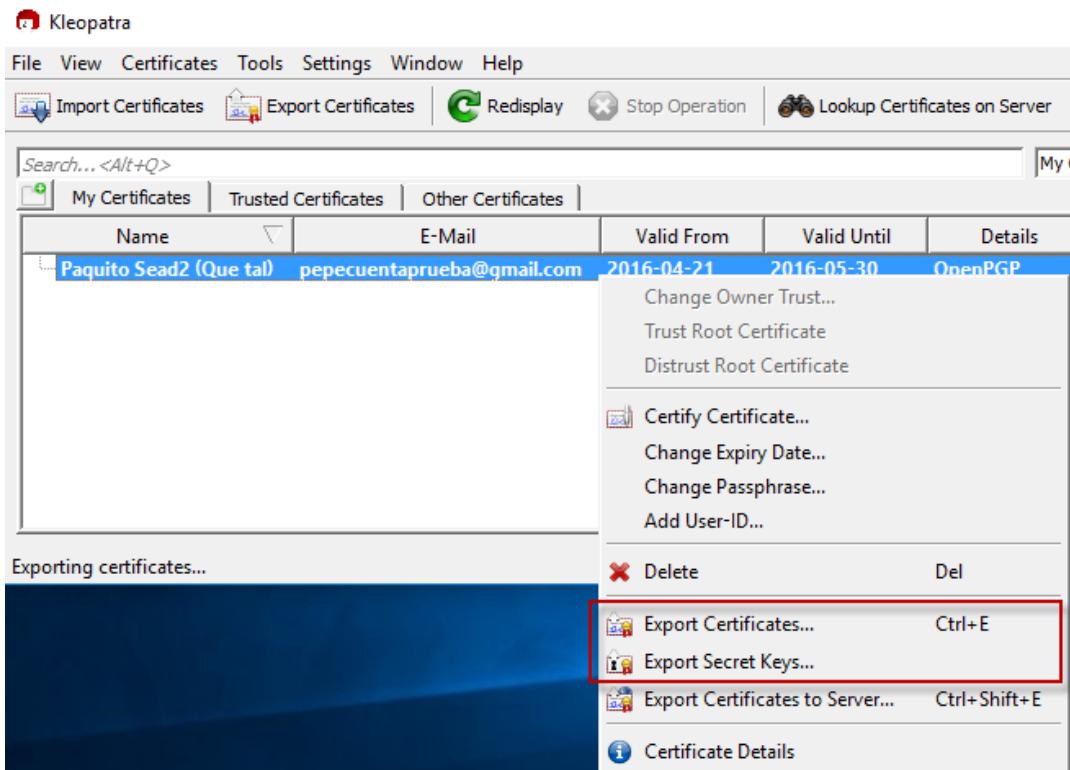
Aproveitamos para subir a chave pública a un servidor de chaves, por defecto: "keys.gnupg.net"



Por defecto o certificado crease sen fecha de expiración, pero podemos establecer un fecha de caducidade do certificado.



Exportamos a chave privada.





The screenshot shows the Kleopatra interface with a file named "clavePaquito.asc" selected. The main window displays the contents of the file as a PGP PRIVATE KEY BLOCK. The text is a long string of base64-encoded binary data.

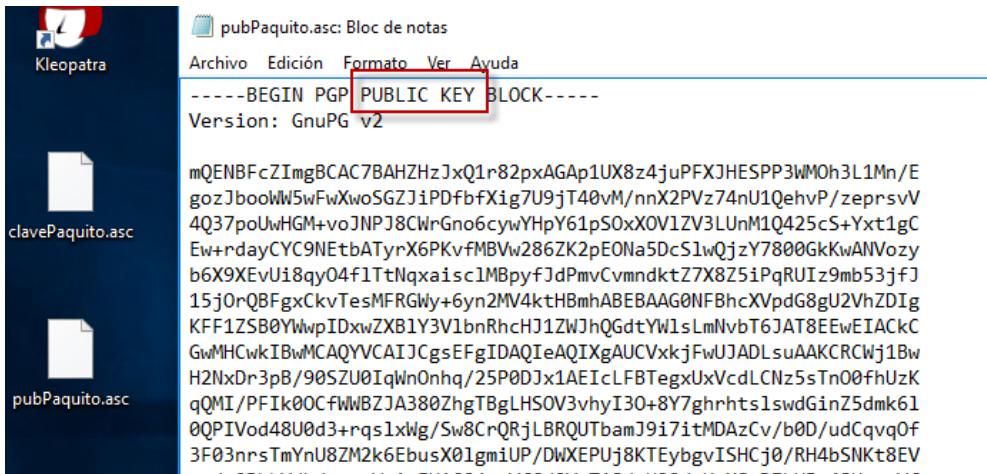
```

-----BEGIN PGP PRIVATE KEY BLOCK-----
Version: GnuPG v2

1QO+BFcZImgBCAC7BAHZHzJxQ1r82pxAGAp1UX8z4juPFXJHESPP3WM0h3L1Mn/E
gozJboolW5wFwXwoSGZJiPDfbfXig7U9jT40vM/nX2PVz74nU1QehvP/zeprsvV
4Q37poUwHGM+voJNPJ8CWrGno6cywYHpY61pS0xXOV1ZV3LUm1Q425cS+Yxt1gC
Ew+rdayCYC9NEtbATyrX6PKvfMBVw286ZK2pEONa5DcS1wQjzY7800GkKwANVozy
b6X9XEvUi8qy04f1TtNqxaisc1MBpyfJdPmvCvmdktZ7X8Z5iPqRUIz9mb53jfJ
15j0rQBFGxCkvTesMFRGwly+6yn2MV4ktHBmhABEBAAH+AwMCd9DBpArhmBbAfrbf
cut134+wk6xvM0wOeBxH/ivtsMM4A/Z2QcJtOoQQ03+NTL1lq7eH02kB2zBoqbAN
dxFqtqVPec7p/QBZbutKrr2KJ50uyCv6UrLKzaUFUfjCI6qiy4vpBAi/erZSp2bUC
eELgr2rrAhU9bXaUdMFIRz15gvgj9XB3S3RFUQRaUdgR+UGyncBTa+0wm7XYGwE
pw3yKBCBygdusXBRwsM7LZpvLbAy7dyP2B2hETTDXm9+rnm6ffX+B6oVDkTeBVO
Ap00hkkIe+Rw1kx711azZj8EGjhJ8hv3UimH0Uv2e7tizGpAk8emFQvQ2XEVH3
X0g1WIfoqNqFg0e40ZoVmzLAc6cDStmmpedqZnhEIC2LLqcArHuSS50xKwMSkhp

```

Exportamos a chave pública.



The screenshot shows the Kleopatra interface with a file named "clavePaquito.asc" selected. The main window displays the contents of the file as a PGP PUBLIC KEY BLOCK. The text is a long string of base64-encoded binary data.

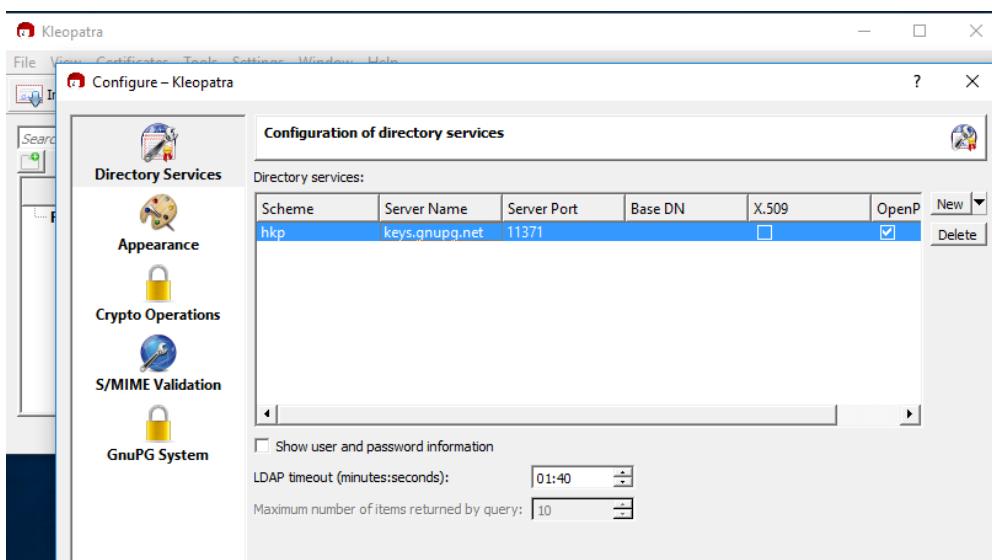
```

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v2

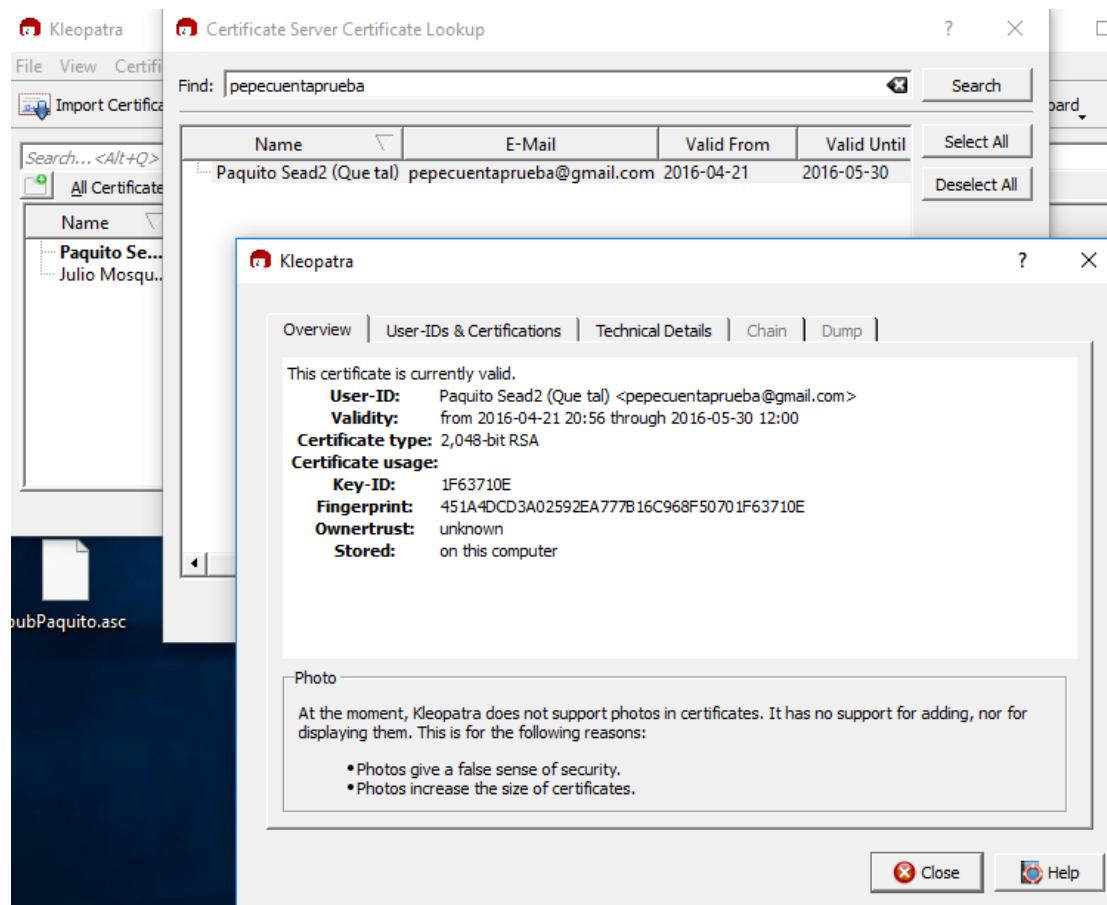
mQENBFcZImgBCAC7BAHZHzJxQ1r82pxAGAp1UX8z4juPFXJHESPP3WM0h3L1Mn/E
gozJboolW5wFwXwoSGZJiPDfbfXig7U9jT40vM/nX2PVz74nU1QehvP/zeprsvV
4Q37poUwHGM+voJNPJ8CWrGno6cywYHpY61pS0xXOV1ZV3LUm1Q425cS+Yxt1gC
Ew+rdayCYC9NEtbATyrX6PKvfMBVw286ZK2pEONa5DcS1wQjzY7800GkKwANVozy
b6X9XEvUi8qy04f1TtNqxaisc1MBpyfJdPmvCvmdktZ7X8Z5iPqRUIz9mb53jfJ
15j0rQBFGxCkvTesMFRGwly+6yn2MV4ktHBmhABEBAAG0NFhcxVpdG8gU2VhZDIg
KFF1ZSB0YWwpIDxwXZB1Y3V1bnRhchJ1ZWJhQGdtYWlsLmNvbT6JAT8EEwEIACK
GwMHCKwIBwMCAQYVCAIJCgsEFgIDAQIeAQIXgAUCVxkjFwUJADLsuAAKCRCWj1Bw
H2NxDr3pB/90SZU0IqWhOnhq/25P0DJx1AEIcLFBTegxUxVcdLCNz5sTn0fhuZK
qQMI/PFIk0OCfWBZJA380ZhgTBgLHSOV3vhvI30+8Y7ghrhsts1swdGinZ5dmk61
0QPIVod48U0d3+rqs1xWg/Sw8CrQRjLBRQUTbamJ9i7itMDAzCv/b0D/udCqvq0f
3F03nnrsTmYnU8ZM2k6EbusX01gmiUP/DWXEPUj8KTEybqvISHCj0/RH4bSNKt8EV

```

Agregamos un servidor de búsquedas por defecto na configuración de Kleopatra para poder encontrar as chaves subidas a ese servidor (por defecto) de forma que podamos facer a búsqueda internamente na aplicación de Kleopatra.



Podemos buscar a través de Kleopatra a nosa chave pública subda anteriormente os servidor de "keys.gnupg.net".



Tamén podes buscar no servidor público en cuestión.

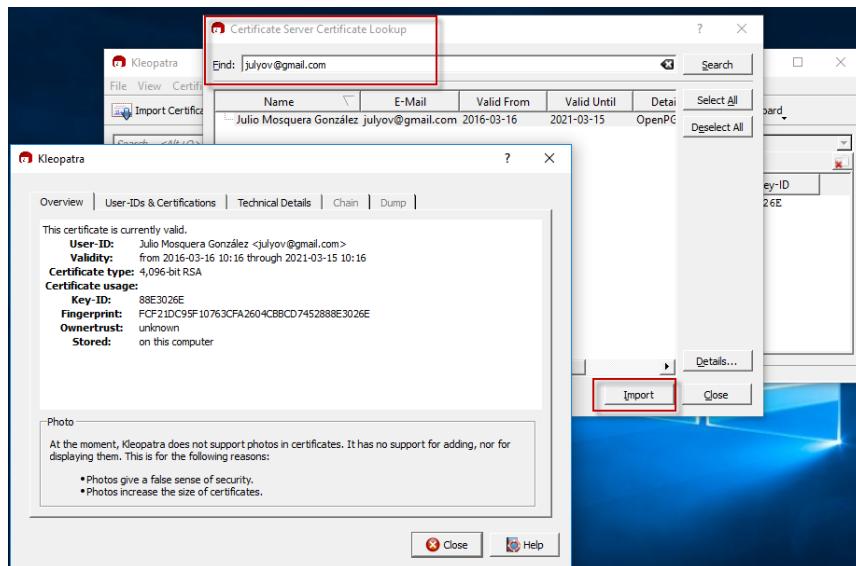
Search results for '0x968f5...' [Close](#) [Help](#)

← | sks.spodhuis.org/pks/lookup?op=vindex&search=0x968F50701F63710E

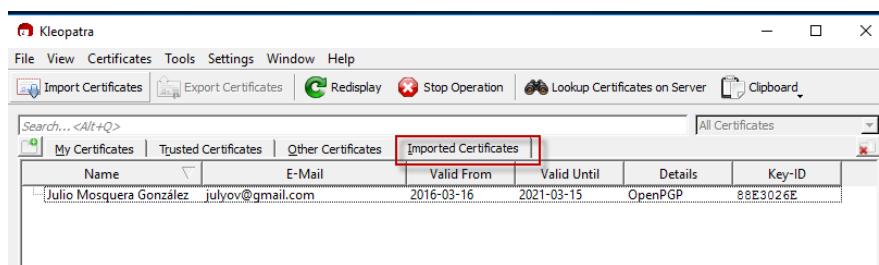
Search results for '0x968f50701f63710e'

Type	bits/keyID	cr. time	exp time	key	expir
pub	2048R/ 1F63710E	2016-04-21			
uid	Paquito Sead2 (Que tal) <pepecuentaprueba@gmail.com>				
sig	sig3 1F63710E	2016-04-21			[selfsig]
sig	sig3 1F63710E	2016-04-21	2016-05-30		[selfsig]
sub	2048R/1B5B2460	2016-04-21			
sig	sbind 1F63710E	2016-04-21			[1]

Buscamos a chave de "julyov@gmail.com".



A cal importamos ao noso equipo local.

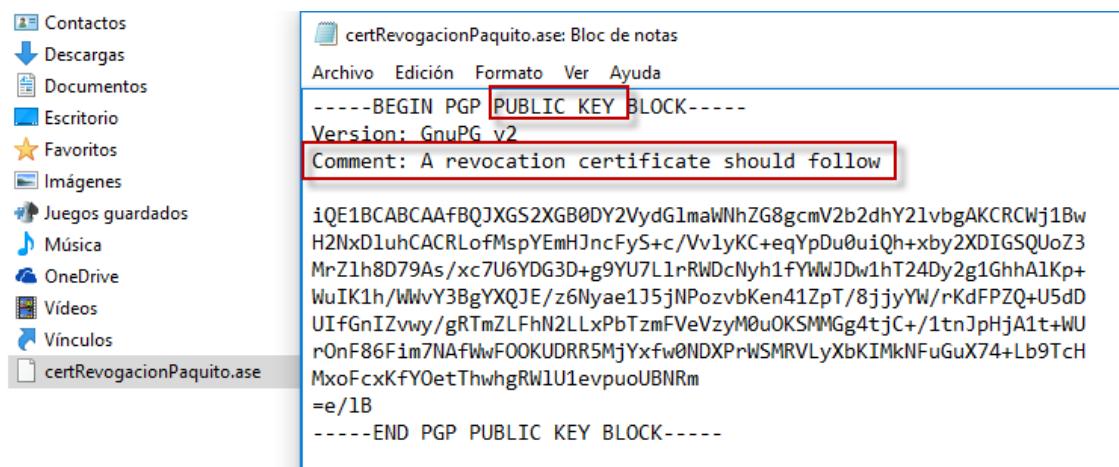


Creamos unha chave de revogación para o noso certificado, non encontrei forma gráfica para esta tarefa polo que finalmente decidiuse crear o certificado de revogación a través de consola con gpg.

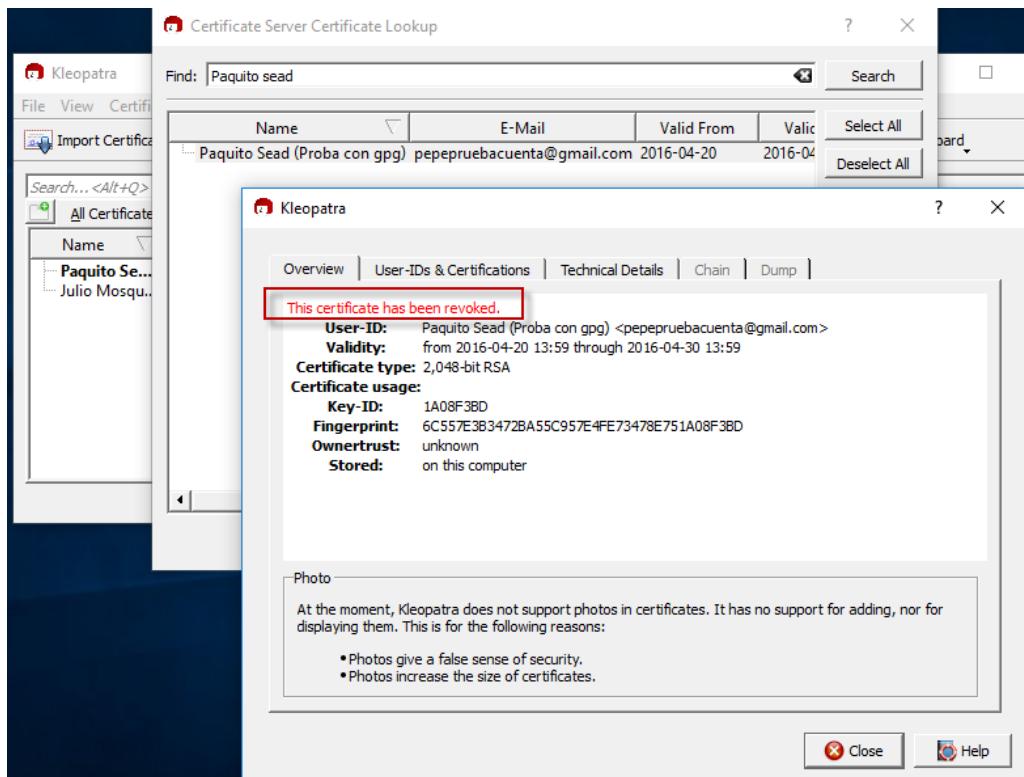
```
C:\Windows\system32\cmd.exe - gpg -o certRevocacionPaquito.ase --gen-revoke 1F63710E
::\Users\adrian\gpg -o certRevocacionPaquito.ase --gen-revoke 1F63710E
sec 2048R/1F63710E 2016-04-21 Paquito Sead2 (Que tal) <pepecuentaprueba@gmail.com>
Crear un certificado de revocaci n para esta clave? (s/N) s
Elija una raz n para la revocaci n:
0 = No se dio ninguna raz n
1 = La clave ha sido comprometida
2 = La clave ha sido reemplazada.
3 = La clave ya no est  en uso
Q = Cancelar
Probablemente quer a seleccionar 1 aqu y)
Su decis n? 3
Introduzca una descripci n opcional; ac bela con una
certificado revocaci n
Raz n para la revocaci n: La clave ya no est  en uso
Certificado revocaci n
Es correcto? (s/N) s
Necesita una contrase a para desbloquear la clave secreta
del usuario: "Paquito Sead2 (Que tal) <pepecuentaprueba@gmail.com>" 
clave RSA de 2048 bits, ID 1F63710E, creada el 2016-04-21
```

A dialog box titled 'pinentry' is shown, asking for a password to unlock the OpenPGP key. The password '*****' is entered in the 'Contraseña' field.

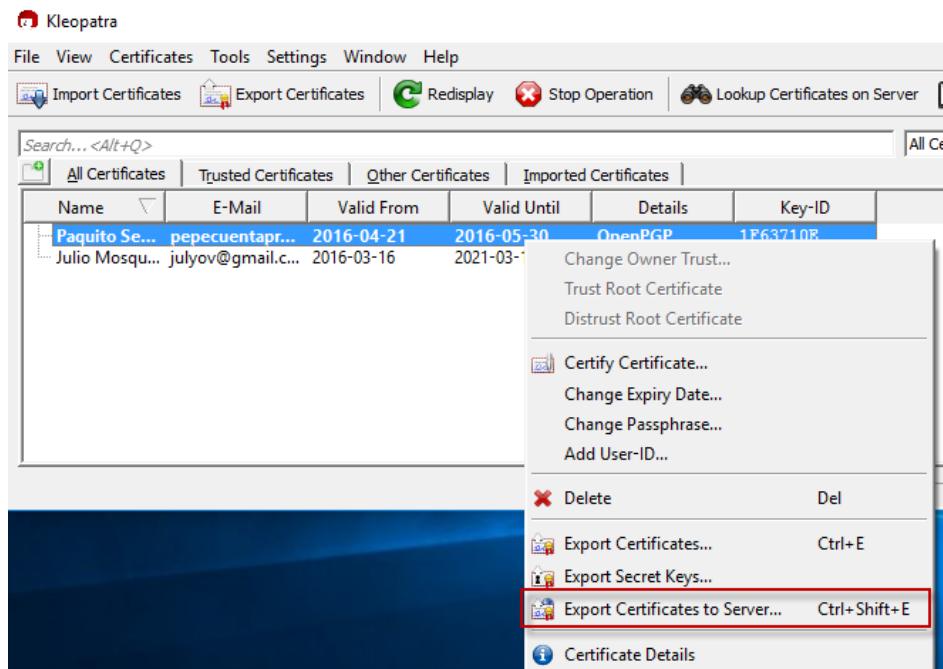
Podemos ver como se creou o certificado de revogación (.ase).



Comprobamos que o certificado foi revogado e non fixo falta importalo a local para actualizar o certificado que xa tíñamos.



Agora temos que actualizar dito certificado nos servidores de chaves. Polo que enviamos de novo o certificado xa revocado a "keys.gnupg.net", para que se complete de forma pública a revogación.



Finalmente comprobamos no servidor que o certificado foi actualizado e revogado correctamente.

Search results for '0x73478e751a08f3bd'

Type	bits/keyID	cr. time	exp time	key	expir
pub	2048R/1A08F3BD	2016-04-20			
sig	revok 1A08F3BD	2016-04-20			[selfsig]
uid	Pagquito Sead (Proba con gpg)	<pepepruebacuenta@gmail.com>			
sig	sig3 1A08F3BD	2016-04-20	2016-04-30	[selfsig]	
sub	2048R/3CC20BBE	2016-04-20			
sig	sbbind 1A08F3BD	2016-04-20	2016-04-30	[]	

Notas: Para a creación de certificados de revogación temos que ter importados no noso equipo local as duas chaves, tanto a privada como a pública (xa que de outra forma non tería sentido porque se nos fixese falta so a chave pública poderíamos revogar certificados que non son nosos e iso non tería sentido...).

O certificado de revogación invalida únicamente a chave pública, xa que unha chave pública sen unha privada o cifrado asimétrico non tería sentido.

7. Firma digital de un documento

Antes de nada importamos os respectivos certificados nos equipos. No seguinte equipo importamos as chaves pública e privada de Paquito, e a pública de Alberto e Julio.

```
root@ubuntufuhdad: /home/adrian/chaves
root@ubuntufuhdad: /home/adrian/chaves# gpg -k
/root/.gnupg/pubring.gpg
-----
pub 2048R/D9BFFFDD 2016-04-23 [[caduca: 2016-05-28]]
uid          Alberto Sanchez Almido (prueba alberto sead) <albertosanchezalmido@gmail.com>
sub 2048R/707F6A43 2016-04-23

pub 2048R/4BC96D87 2016-04-23 [[caduca: 2016-06-07]]
uid          Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>
sub 2048R/FD8D6909 2016-04-23

root@ubuntufuhdad: /home/adrian/chaves# gpg --list-secret-keys
/root/.gnupg/secring.gpg
-----
sec 2048R/D9BFFFDD 2016-04-23 [[caduca: 2016-05-28]]
uid          Alberto Sanchez Almido (prueba alberto sead) <albertosanchezalmido@gmail.com>
ssb 2048R/707F6A43 2016-04-23

root@ubuntufuhdad: /home/adrian/chaves#
```

No outro equipo importamos a chave pública e privada de Paquito. É a pública de Paquito.

```
root@ubuntusead: /home/adrian/chaves
root@ubuntusead: /home/adrian/chaves# gpg -k
/root/.gnupg/pubring.gpg
-----
pub 2048R/4BC96D87 2016-04-23
uid          Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>
sub 2048R/FD8D6909 2016-04-23

pub 2048R/D9BFFFDD 2016-04-23 [[caduca: 2016-05-28]]
uid          Alberto Sanchez Almido (prueba alberto sead) <albertosanchezalmido@gmail.com>
sub 2048R/707F6A43 2016-04-23

pub 4096R/88E3026E 2016-03-16 [[caduca: 2021-03-15]]
uid          Julio Mosquera González <julyov@gmail.com>
sub 4096R/2B8E9D2B 2016-03-16 [[caduca: 2021-03-15]]

root@ubuntusead: /home/adrian/chaves# gpg --list-secret-keys
/root/.gnupg/secring.gpg
-----
sec 2048R/4BC96D87 2016-04-23
uid          Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>
ssb 2048R/FD8D6909 2016-04-23

root@ubuntusead: /home/adrian/chaves#
```

Comprobaremos a firma que está na plataforma, e supostamente asinada por Julio. Trátase dunha firma de arquivo separada, por un lado estaría o documento orixinal que contén a información e por outro soamente a firma.

Vese que efectivamente dito arquivo foi firmado por Julio.

gpg --verify [arquivo_sinatura]

```
root@ubuntusead: /home/adrian
root@ubuntusead:/home/adrian# ls
chaves          Escritorio      ola a todos.txt      Vídeos
cifrado_para_un_alumno.txt.asc examples.desktop  ola a todos.txt.sig
Descargas        Imágenes       Plantillas
Documentos      Música        Público
root@ubuntusead:/home/adrian# cat "ola a todos.txt"
Que tal estades?root@ubuntusead:/home/adrian#
root@ubuntusead:/home/adrian# gpg --verify "ola a todos.txt.sig"
gpg: Firmado el sáb 23 abr 2016 00:00:22 CEST usando clave RSA ID 88E3026E
gpg: Firma correcta de «Julio Mosquera González <julyov@gmail.com>»
gpg: AVISO: ;Esta clave no está certificada por una firma de confianza!
gpg:                 No hay indicios de que la firma pertenezca al propietario.
Huellas dactilares de la clave primaria: FCF2 1DC9 5F10 763C FA26 04CB BCD7 452
8 88E3 026E
root@ubuntusead:/home/adrian#
```

Comprobamos o siguiente arquivo subido a platadorma o cal está firmado e cifrado todo nun mesmo arquivo.

O tentar descifralo vemos que nos aparece un erro, esto é debido a que este arquivo foi cifrado para unha persoa en particular ca chave pública de esa persoa, o cal non temos a chave privada disa persoa para poder descifralo.

gpg --output --decrypt [arquivo_cifrado]

```
root@ubuntusead: /home/adrian
root@ubuntusead:/home/adrian# ls
chaves          Escritorio      ola a todos.txt      Vídeos
cifrado_para_un_alumno.txt.asc examples.desktop  ola a todos.txt.sig
Descargas        Imágenes       Plantillas
Documentos      Música        Público
root@ubuntusead:/home/adrian# gpg --output cifradoAlumno.txt --decrypt cifrado_p
ara_un_alumno.txt.asc
gpg: cifrado con clave RSA de 4096 bits, ID 2B8E9D2B, creada el 2016-03-16
      «Julio Mosquera González <julyov@gmail.com>»
gpg: descifrado fallido: clave secreta no disponible
root@ubuntusead:/home/adrian#
```

Creamos un documento o cal firmaremos, deberá estar por un lado o arquivo orixinal e o arquivo da sinatura que fará referencia o arquivo orixinal.

--detach-sig: crea a firma do arquivo a firmar, nun ficheiro separado.

Estos arquivos xunto ca chave pública de Paquito, adxuntarase na entrega desta tarefa.

gpg --output [archivo_sinatura_de_saída] --detach-sig [archivo_orixinal]

```
root@ubuntusead:/home/adrian
root@ubuntusead:/home/adrian# echo "Ola! Documento asinado por Adrián" > asinado_adrian.txt
root@ubuntusead:/home/adrian# cat asinado_adrian.txt
Ola! Documento asinado por Adrián
root@ubuntusead:/home/adrian# gpg --output asinado_adrian.txt.sig --detach-sig asinado_adrian.txt

Necesita una frase contraseña para desbloquear la clave secreta
del usuario: "Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>"
clave RSA de 2048 bits, ID 4BC96D87, creada el 2016-04-23

gpg: el agente gpg no esta disponible en esta sesión
root@ubuntusead:/home/adrian# ls
asinado_adrian.txt          examples.desktop
asinado_adrian.txt.sig       Imágenes
chaves                         Música
cifrado_para_un_alumno.txt.asc ola_a_todos.txt
Descargas                      ola_a_todos.txt.sig
documento_importante.txt      Plantillas
documento_importante.txt.gpg   Público
Documentos                     Vídeos
Escritorio
root@ubuntusead:/home/adrian#
```

Agora crearemos un documento o cal firmaremos de tres formas distintas.

--detach-sig: crea a firma do arquivo a firmar, nun ficheiro separado.

```
gpg --output [archivo_sinfatura_saída] --detach-sig [archivo_orixinal]
```

--clearsign: crea a firma nun único arquivo e non o comprime.

```
gpg --output [archivo_sinfatura_saída] --clearsign [archivo_orixinal]
```

--sign: crea a firma nun único arquivo e si o comprime.

```
gpg --output [archivo_sinfatura_saída] --sign [archivo_orixinal]
```

```
root@ubuntusead:/home/adrian
root@ubuntusead:/home/adrian# echo "Ola Alberto, este é un documento asinado por
Adrián" > asinadoParaAlberto.txt
root@ubuntusead:/home/adrian# gpg --output asinadoParaAlberto.txt.sig --detach-s
ig asinadoParaAlberto.txt

Necesita una frase contraseña para desbloquear la clave secreta
del usuario: "Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>"  
clave RSA de 2048 bits, ID 4BC96D87, creada el 2016-04-23

gpg: el agente gpg no esta disponible en esta sesión
root@ubuntusead:/home/adrian# gpg --output asinadoTodoEnUnNoncompri.txt.asc --cl
earsign asinadoParaAlberto.txt

Necesita una frase contraseña para desbloquear la clave secreta
del usuario: "Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>"  
clave RSA de 2048 bits, ID 4BC96D87, creada el 2016-04-23

gpg: el agente gpg no esta disponible en esta sesión
root@ubuntusead:/home/adrian# gpg --output asinadoTodoEnUnSicompri.txt.sig --sig
n asinadoParaAlberto.txt

Necesita una frase contraseña para desbloquear la clave secreta
del usuario: "Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>"  
clave RSA de 2048 bits, ID 4BC96D87, creada el 2016-04-23

gpg: el agente gpg no esta disponible en esta sesión
root@ubuntusead:/home/adrian# ls
asinado_adrian.txt          entrega
asinado_adrian.txt.sig       entrega.zip
asinadoParaAlberto.txt       Escritorio
asinadoParaAlberto.txt.sig   examples.desktop
asinadoTodoEnUnNoncompri.txt.asc  Imágenes
asinadoTodoEnUnSicompri.txt.sig Música
chaves                         ola_a_todos.txt
cifrado_para_un_alumno.txt.asc  ola_a_todos.txt.sig
Descargas                      Plantillas
documento_importante.txt      Público
documento_importante.txt.gpg    Vídeos
Documentos
root@ubuntusead:/home/adrian#
```

Enviamos os arquivos asinados a Alberto.

Descargamos os arquivos asinados no equipo de Alberto.

Comprobamos a sinatura dos arquivos recibidos no equipo de Alberto.

gpg -d = gpg --decrypt: Descifra, extrae e comproba a firma dun arquivo.

Como vemos na seguinte captura, o arquivo coa firma independiente “asinadoParaAlberto.txt.sig” ainda que se indique o modificador --decrypt este comproba a firma pero NON extrae o contido, esto é normal xa que se trata de arquivos independientes, por un lado estaría o arquivo orixinal (.txt) e por outro soamente a sinatura (.txt.sig). Neste caso (de ter dous arquivos) teríase que comprobar a firma ou ben con --decrypt ou tamén e más correcto, con --verify (como se mostra o final do procedemento da seguinte captura).

Nas outras comprobacións donde o arquivo xa vai xunto ca firma vemos que con -d, podemos extraer o contido e a misma vez verificar a firma.

Pediranos a contrasinal da chave privada do emisor que firma o arquivo, neste caso Paquito.

```
root@ubuntufuhdr:/home/adrian
root@ubuntufuhdr:/home/adrian# ls
asinadoParaAlberto.txt           documento_importante.txt      Música
asinadoParaAlberto.txt.sig        documento_importante.txt.gpg  Plantillas
asinadoTodoEnUnNoncompri.txt.asc  Documentos                  Público
asinadoTodoEnUnSicomPri.txt.sig   Escritorio                 Vídeos
chaves
Descargas                         Imágenes
root@ubuntufuhdr:/home/adrian# gpg -d asinadoParaAlberto.txt.sig
gpg: Firmado el mié 04 may 2016 21:38:52 CEST usando clave RSA ID 4BC96D87
gpg: Firma correcta de «Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>»
gpg: AVISO: ;Esta clave no está certificada por una firma de confianza!
gpg:          No hay indicios de que la firma pertenezca al propietario.
Huellas dactilares de la clave primaria: 51E5 581E 5A7E EF6F E292 8F4D B928 11C
7 4BC9 6D87
root@ubuntufuhdr:/home/adrian# gpg -d asinadoTodoEnUnNoncompri.txt.asc
Ola Alberto, este é un documento asinado por Adrián
gpg: Firmado el mié 04 may 2016 21:41:27 CEST usando clave RSA ID 4BC96D87
gpg: Firma correcta de «Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>»
gpg: AVISO: ;Esta clave no está certificada por una firma de confianza!
gpg:          No hay indicios de que la firma pertenezca al propietario.
Huellas dactilares de la clave primaria: 51E5 581E 5A7E EF6F E292 8F4D B928 11C
7 4BC9 6D87
root@ubuntufuhdr:/home/adrian# gpg -d asinadoTodoEnUnSicomPri.txt.sig
Ola Alberto, este é un documento asinado por Adrián
gpg: Firmado el mié 04 may 2016 21:41:52 CEST usando clave RSA ID 4BC96D87
gpg: Firma correcta de «Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>»
gpg: AVISO: ;Esta clave no está certificada por una firma de confianza!
gpg:          No hay indicios de que la firma pertenezca al propietario.
Huellas dactilares de la clave primaria: 51E5 581E 5A7E EF6F E292 8F4D B928 11C
7 4BC9 6D87
root@ubuntufuhdr:/home/adrian# gpg --verify asinadoParaAlberto.txt.sig
gpg: Firmado el mié 04 may 2016 21:38:52 CEST usando clave RSA ID 4BC96D87
gpg: Firma correcta de «Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>»
gpg: AVISO: ;Esta clave no está certificada por una firma de confianza!
gpg:          No hay indicios de que la firma pertenezca al propietario.
Huellas dactilares de la clave primaria: 51E5 581E 5A7E EF6F E292 8F4D B928 11C
7 4BC9 6D87
root@ubuntufuhdr:/home/adrian#
```

No caso de o arquivo NON comprimido (--clearsign) vemos como móstrase o contido do documento e seguidamente a chave pública de quen o firmou.

```
root@ubuntufuhdadr:/home/adrian# ls
asinadoParaAlberto.txt          documento_importante.txt      Música
asinadoParaAlberto.txt.sig       documento_importante.txt.gpg  Plantillas
asinadoTodoEnUnNoncompri.txt.asc Documentos                  Público
asinadoTodoEnUnSicompri.txt.sig Escritorio                 Vídeos
chaves                           examples.desktop
Descargas                        imágenes
root@ubuntufuhdadr:/home/adrian# cat asinadoTodoEnUnNoncompri.txt.asc
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

Ola Alberto, este é un documento asinado por Adrián
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1

iQEcBAEBAgAGBQJXKlBnAAoJELkoEcdLyW2HUsIH/1aP4yZzyFhf4qXP6zJ85vP0
+7WYyXwM4IzSYYw++ApjN245WqgiCSRiSF1ZEQLMXokp0HYNjKp0aGSmZI1wW8QmL
yijNAhsUSX2ieBgs8dvB9Y9XBDJdExsQ0o+jXQeL59ZVbUn9FZ+/uqnCNoVvxkUy
5Jr07Tw0iqliqdyG4UWbgiQbvepHaED1IMAMc0kxjdFKQKq/LT3KZAuGLb4vbfiXF
cCNNMpD0E4FnRd1Yhh5VHYom+dqp83rFQqKFqaC4KRPFJpXWyL4KWzcHo0F5SpLg
Q0Jgm7G4BJZOD1I6rlp09DbqS+rpjTYnU/ov04CVnqxSqQZc74Bis9t+o+mtvuU=
=TLH2
-----END PGP SIGNATURE-----
root@ubuntufuhdadr:/home/adrian#
```

Si modificamos o documento que vai por separado, documento orixinal e documento da sinatura. Vemos que non verifica a firma.

Esto é debido a que cando se crea un arquivo de sinatura por separado, este comproba e compara cunha función resumen (checksum) se coincide co arquivo orixinal que se asinou nunha primeira instancia. Se se modifica o arquivo orixinal estase cambiando o checksum deste, polo tanto o arquivo de sinatura detecta dito cambio e rotura da integridade do arquivo e verifica a sinatura como incorrecta, o cal é o máis lóxico.

```
root@ubuntufuhdadr:/home/adrian# ls
asinadoParaAlberto.txt          documento_importante.txt      Música
asinadoParaAlberto.txt.sig       documento_importante.txt.gpg  Plantillas
asinadoTodoEnUnNoncompri.txt.asc Documentos                  Público
asinadoTodoEnUnSicompri.txt.sig Escritorio                 Vídeos
chaves                           examples.desktop
Descargas                        imágenes
root@ubuntufuhdadr:/home/adrian# echo "isto é unha modificación do documento de
firma separada" >> asinadoParaAlberto.txt
root@ubuntufuhdadr:/home/adrian# cat asinadoParaAlberto.txt
Ola Alberto, este é un documento asinado por Adrián
isto é unha modificación do documento de firma separada
root@ubuntufuhdadr:/home/adrian# gpg --verify asinadoParaAlberto.txt.sig
gpg: Firmado el mié 04 may 2016 21:38:52 CEST usando clave RSA ID 4BC96D87
gpg: Firma INCORRECTA de «Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>»
root@ubuntufuhdadr:/home/adrian#
```

Por último xeramos un arquivo firmado e cifrado para un determinado usuario, o cal neste caso será o noso amigo Alberto.

--recipient [chavePublica_ou_PubID]: Será a chave pública do destinario ca que se cifrará o arquivo.

```
gpg --output [arquivo_sinatura_saída] --encrypt --sign --recipient [chavePublica_ou_PubID] [arquivo_orixinal]
```

```
root@ubuntusead:/home/adrian
root@ubuntusead:/home/adrian# echo "Ola documento asinado e cifrado para un compañero" > documento_importante.txt
root@ubuntusead:/home/adrian# cat documento_importante.txt
Ola documento asinado e cifrado para un compañero
root@ubuntusead:/home/adrian# gpg --output documento_importante.txt.gpg --encrypt --sign --recipient albertosanchezalmido@gmail.com documento_importante.txt

Necesita una frase contraseña para desbloquear la clave secreta
del usuario: "Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmail.com>"
```

clave RSA de 2048 bits, ID 4BC96D87, creada el 2016-04-23

```
gpg: el agente gpg no esta disponible en esta sesión
gpg: 707F6A43: No hay seguridad de que esta clave pertenezca realmente
al usuario que se nombra

pub 2048R/707F6A43 2016-04-23 Alberto Sanchez Almido (prueba alberto sead) <albertosanchezalmido@gmail.com>
    Huella de clave primaria: DE3D 3C67 893D E640 4841  1F18 6B8B 90C1 D9BF FFDD
    Huella de subclave: 7D12 20D3 1634 55A2 FED6  0BED 1498 AF98 707F 6A43

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) s
root@ubuntusead:/home/adrian# ls
asinado_adrian.txt          examples.desktop
asinado_adrian.txt.sig       Ímágenes
chaves                         Música
cifrado_para_un_alumno.txt.asc ola_a_todos.txt
Descargas                      ola_a_todos.txt.sig
documento_importante.txt      Plantillas
documento_importante.txt.gpg   Público
Documentos                     Vídeos
Escritorio
root@ubuntusead:/home/adrian#
```

No proceso habitual, enviamos dito arquivo xa asinado e cifrado nun mesmo arquivo a Alberto.

Descargamos o arquivo no equipo de Alberto.

Comprobamos a verificación da firma e a súa vez desciframos o arquivo. Como foi cifrado coa chave pública do destinatario (Alberto), soamente este coa súa chave privada poderá descifrar, polo que lle pedirá a contrasinal.

Na seguinte captura vemos como se verifica correctamente a firma, e indicámoslle un arquivo de saída para poder ver o contido cifrado en plain-text.

```
root@ubuntufuhdadr:/home/adrian# ls
asinadoParaAlberto.txt      documento_importante.txt.gpg  Imágenes     Vídeos
asinadoParaAlberto.txt.sig   Documentos                  Música
chaves                      Escritorio                Plantillas
Descargas                   examples.desktop          Público
root@ubuntufuhdadr:/home/adrian# gpg --output documento_importante.txt --decrypt
documento_importante.txt.gpg

Necesita una frase contraseña para desbloquear la clave secreta
del usuario: "Alberto Sanchez Almido (prueba alberto sead) <albertosanchezalmido
@gmail.com>"
clave RSA de 2048 bits, ID 707F6A43, creada el 2016-04-23 (ID de clave primaria
D9BFFFDD)

gpg: el agente gpg no está disponible en esta sesión
gpg: cifrado con clave RSA de 2048 bits, ID 707F6A43, creada el 2016-04-23
    «Alberto Sanchez Almido (prueba alberto sead) <albertosanchezalmido@gmail.
com>»
gpg: Firmado el mié 04 may 2016 10:30:34 CEST usando clave RSA ID 4BC96D87
gpg: Firma correcta de «Paquito sead (prueba cuenta sead) <pepepruebacuenta@gmai
l.com>»
gpg: AVISO: ¡Esta clave no está certificada por una firma de confianza!
gpg:                 No hay indicios de que la firma pertenezca al propietario.
Huellas dactilares de la clave primaria: 51E5 581E 5A7E EF6F E292  8F4D B928 11C
7 4BC9 6D87
root@ubuntufuhdadr:/home/adrian# ls
asinadoParaAlberto.txt      documento_importante.txt.gpg  Música
asinadoParaAlberto.txt.sig   Documentos                  Plantillas
chaves                      Escritorio                Público
Descargas                   examples.desktop          Vídeos
documento_importante.txt    Imágenes
root@ubuntufuhdadr:/home/adrian# cat documento_importante.txt
Ola documento asinado e cifrado para un compañero
root@ubuntufuhdadr:/home/adrian#
```

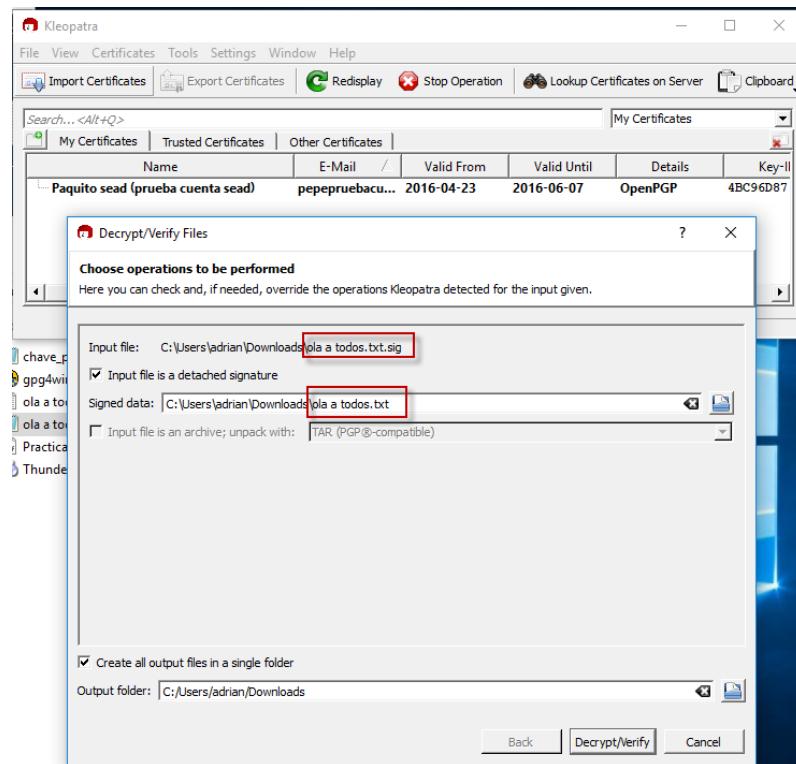
Nota: Os arquivos de saída xerados co modificador --output non sería preciso asignarlle extensións, se todas estas comprobacións facémolas a nivel gpg por consola. No caso de traballar con outras ferramentas (Kleopatra) teremos problemas de compatibilidade e recoñecemento dos arquivos tratados, por iso razón nestas tarefas asináronsele determinadas extensións.

Referencias de axuda:

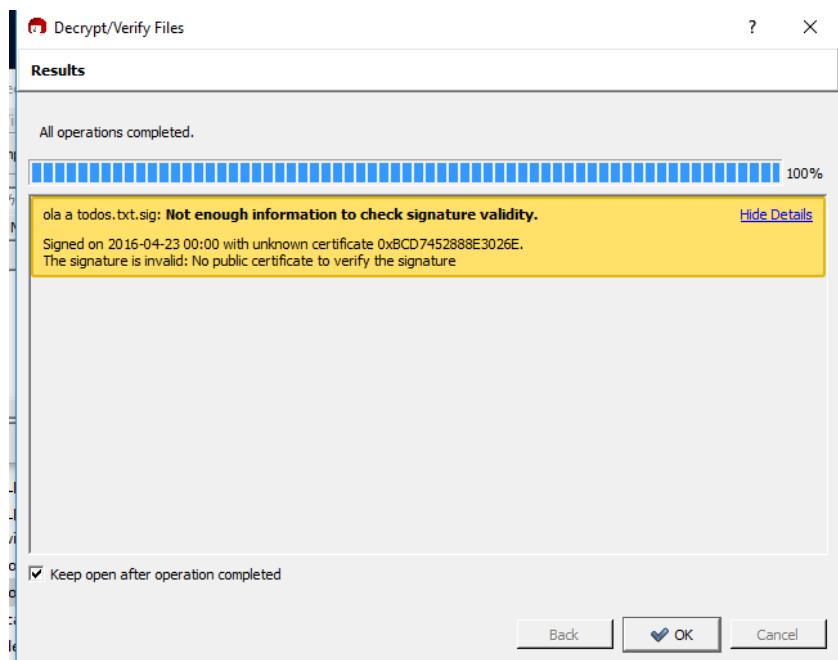
<https://www.gnupg.org/documentation/manuals/gnupg/Operational-GPG-Commands.html>
<https://www.gnupg.org/gph/es/manual.html>

Faremos as mismas probas que se fixeron para Linux pero agora para Windows, facendo uso da utilidade gráfica Kleopatra.

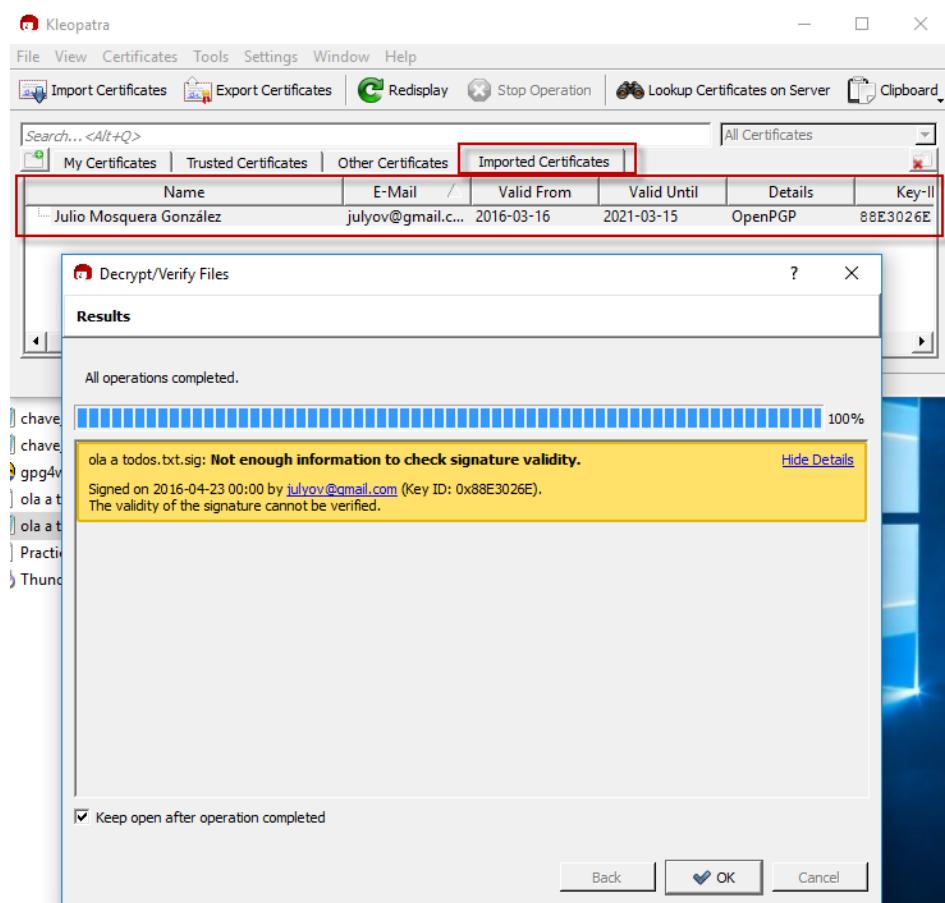
Comprobamos que o arquivo subido a plataforma está firmado por "julyov@gmail.com". Para iso cargamos (input) o arquivo .sig xunto co arquivo que contén os datos de información.



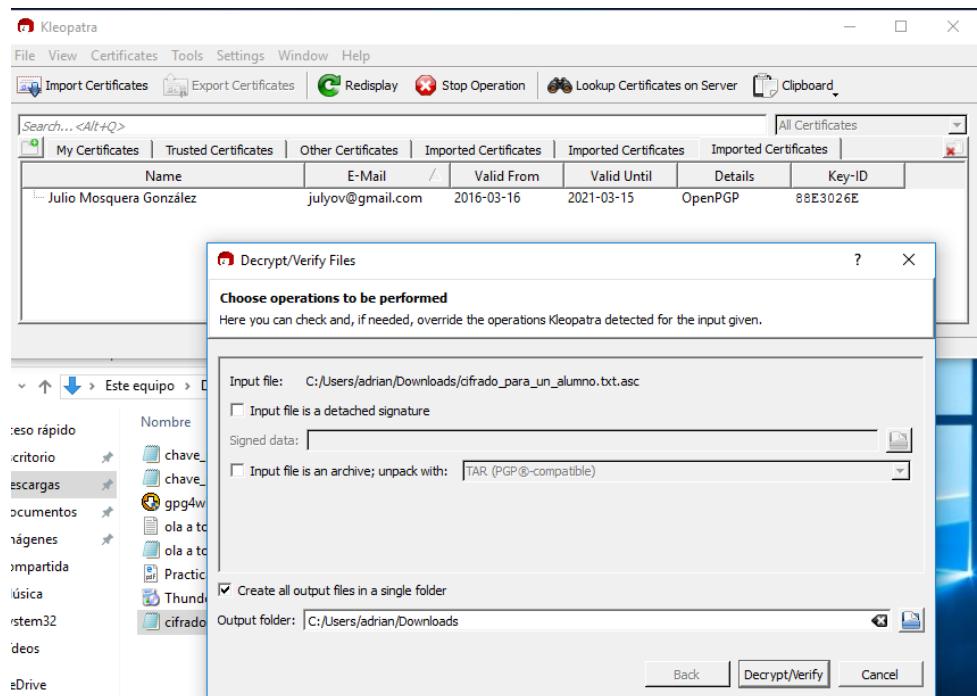
Comprobamos que nos recoñece o ID co que se firmou pero non especifica que e o emisor. Esto e debido a que non temos cargado en Kleopatra a chave pública do emisor da mensaxe.



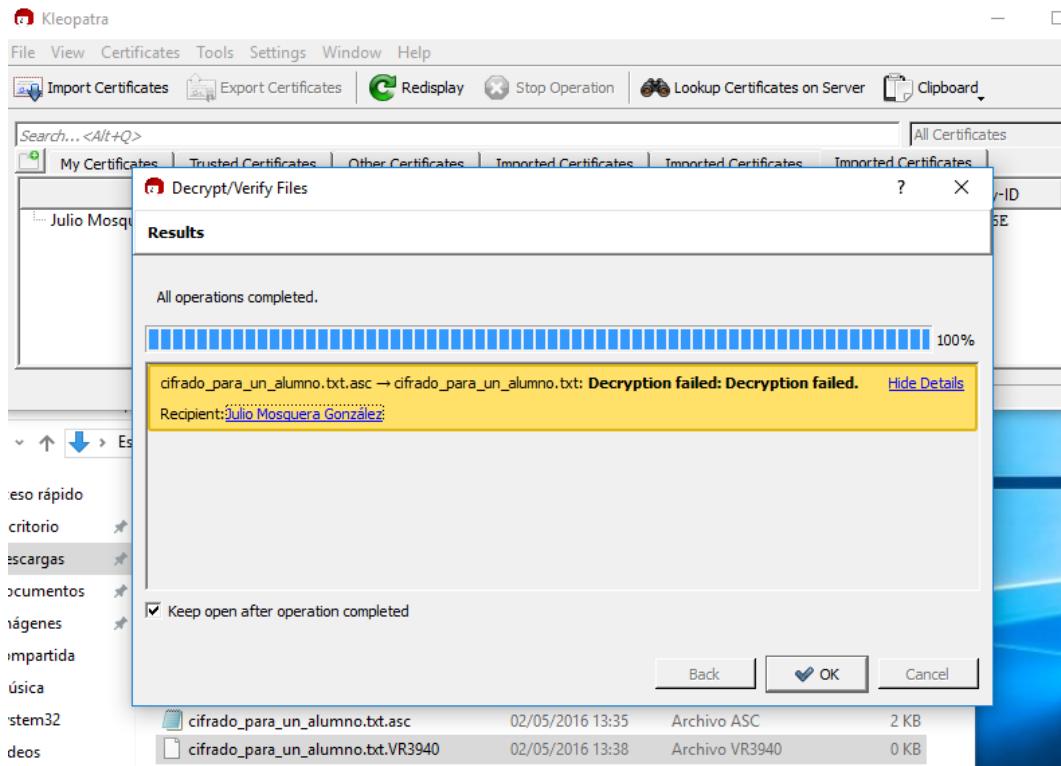
Cargamos o certificado público do emisor, e vemos como agora podemos ver o correo de quen posiblemente o firmou, pero sigue sin poder verificarlo.



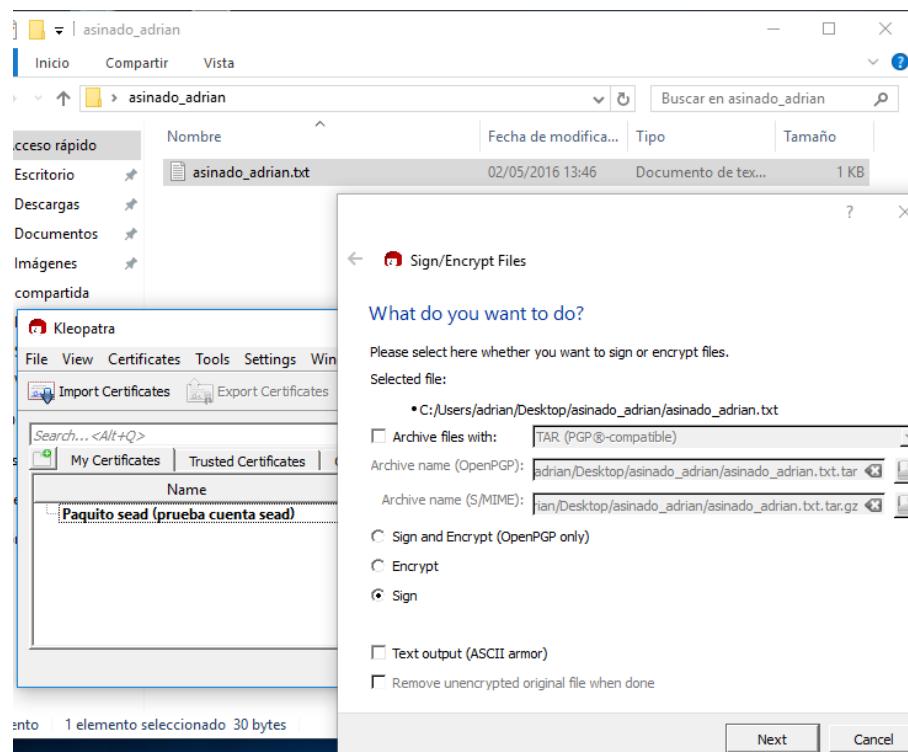
Comprobamos o arquivo asinado e cifrado subido a plataforma (todo nun arquivo .asc).



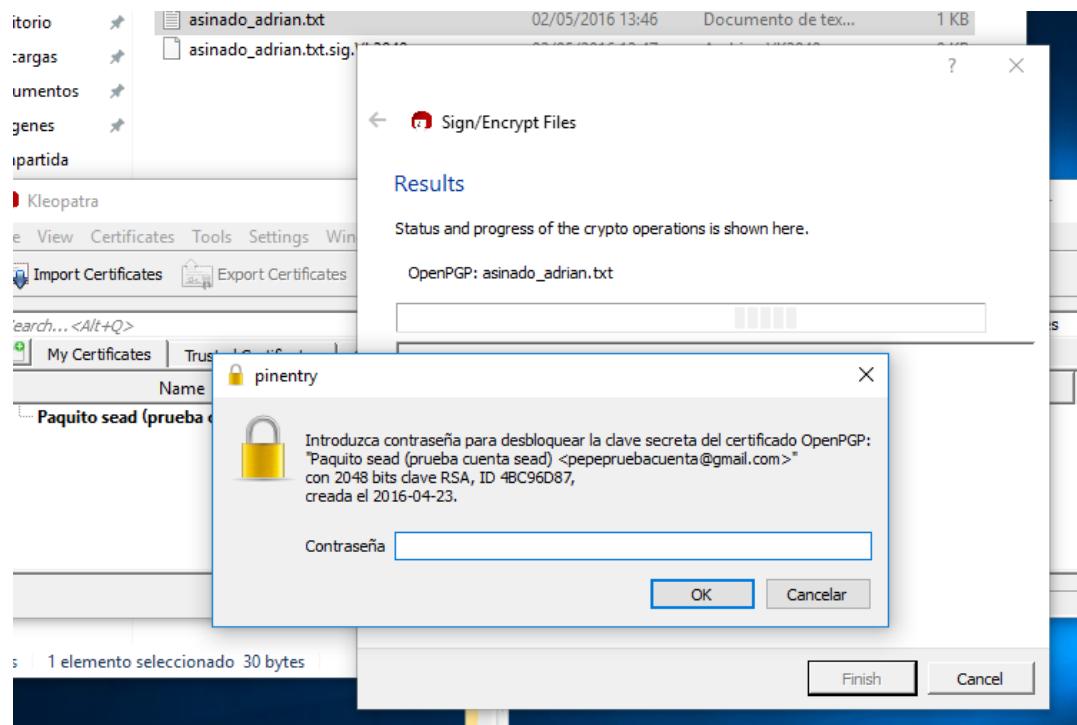
A comprobación de firma non a verifica?, o descifrado non e posible debido a que vai a un destinatario en concreto (“o propio Julio”? cifrado ca sua propia chave secreta?), o cal non temos o certificado dispoñible para poder descifralo.



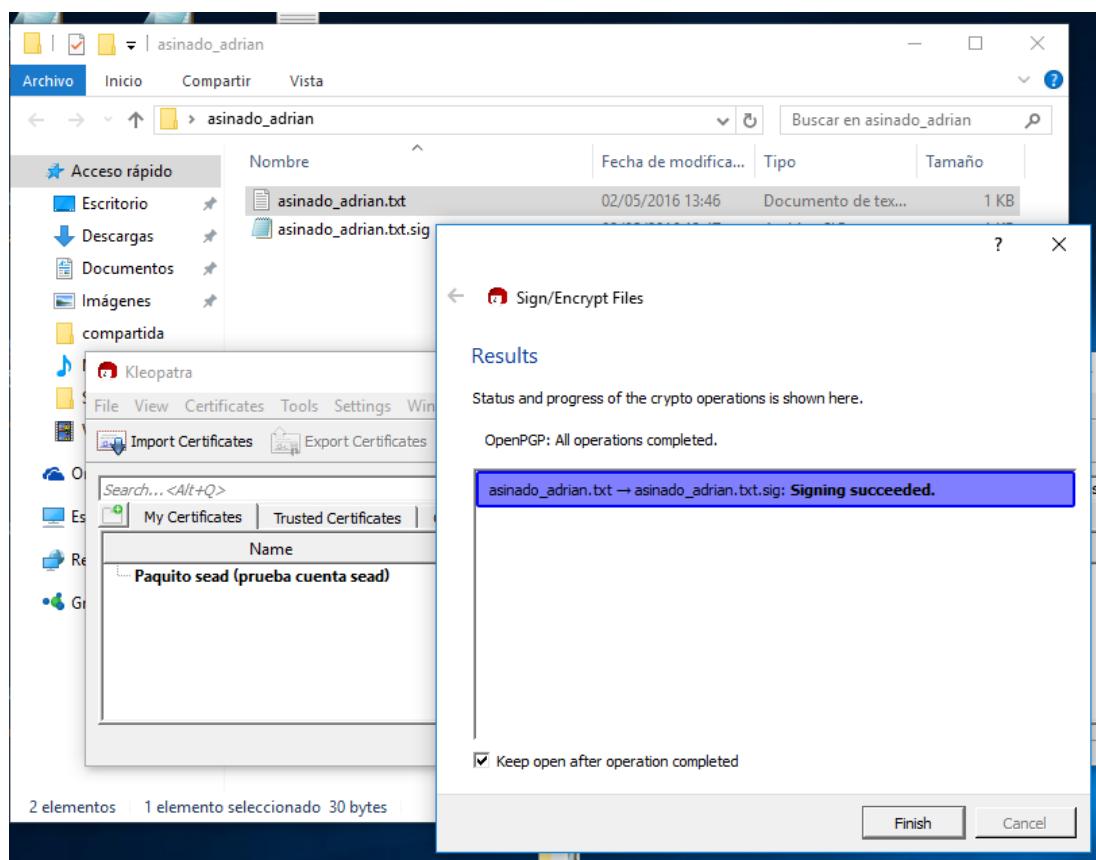
So firmamos unha mensaxe coa nosa chave privada, a cal enviarase na entrega destas tarefas aduxuntas nun arquivo comprimido en .7z.



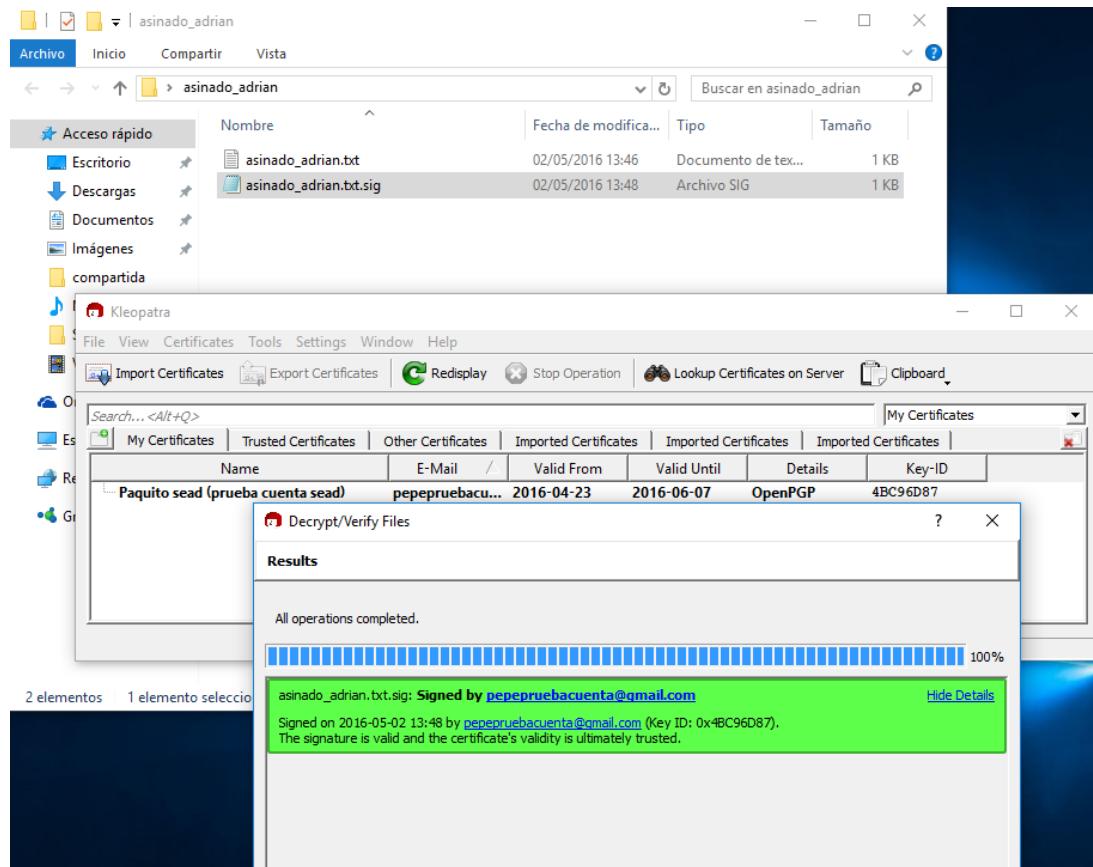
Temos a opción de facer nun mesmo arquivo en nun separado, seleccionamos a opción por separado.



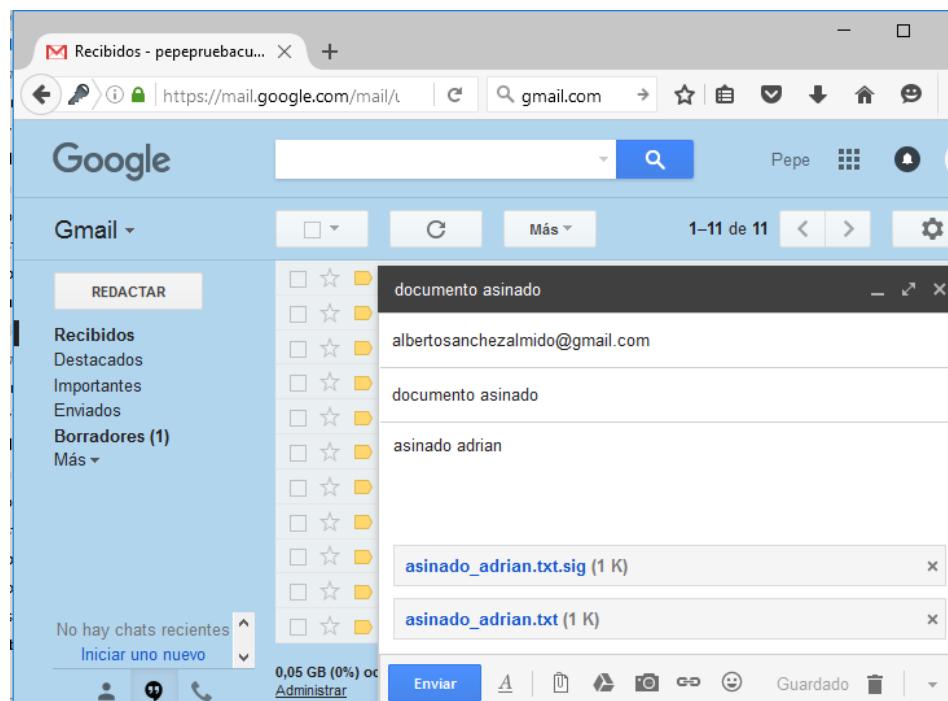
Conclue o firmado correctamente, xerando un novo arquivo .txt.sig.



Para verificar esto internamente probamos a facelo nos mesmos, co noso propio mensaxe. E vemos que verifica correctamente.



Enviamos agora un documento soamente asinado a outro compañoero.



Descargamos os arquivos e comprobamos a veracidade con Kleopatra.

The screenshot shows a Gmail inbox with the following details:

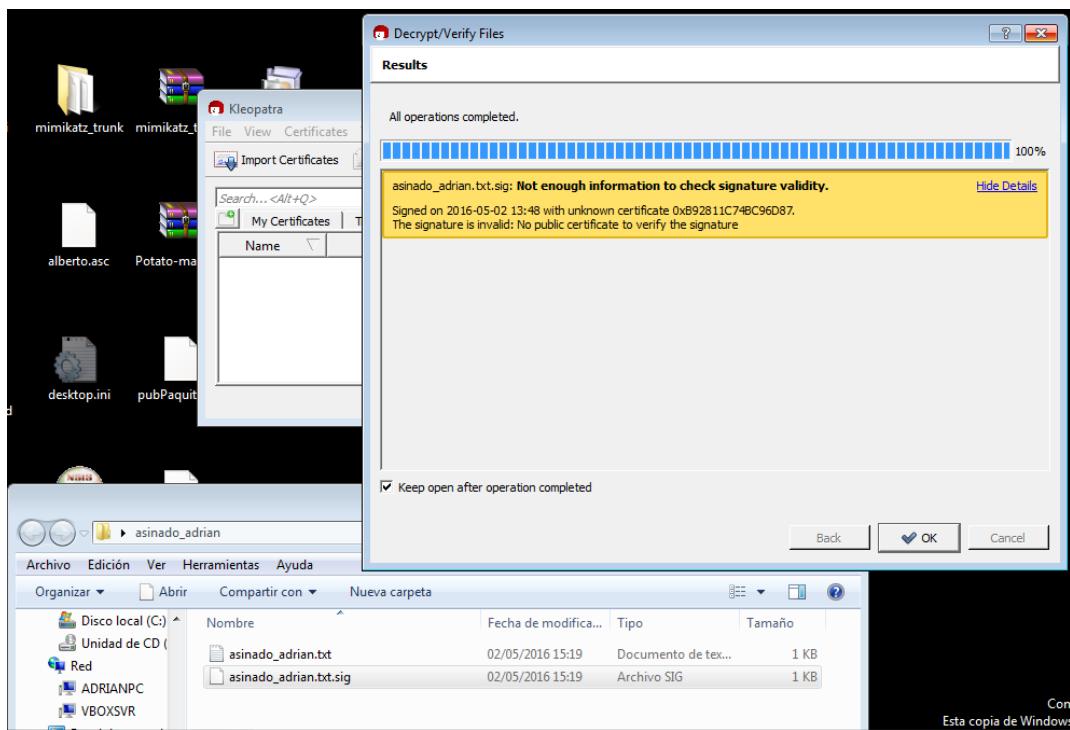
- Subject:** documento asinado
- From:** Pepe guzmán garcía <pepepruebacuenta@gmail.com>
- To:** albertosanchezalmido@gmail.com
- Message Content:** asinado adrian
- Attachments:** 2 archivos adjuntos — Explorar y descargar todos los archivos adjuntos
 - asinado_adrian.txt (1K) Ver Explorar y descargar
 - asinado_adrian.txt.sig (1K) Explorar y descargar
- Response Options:** Respuesta rápida (Para: Pepe guzmán garcía <pepepruebacuenta@gmail.com>), Más opciones de respuesta

Verificamos o arquivo .txt.sig.

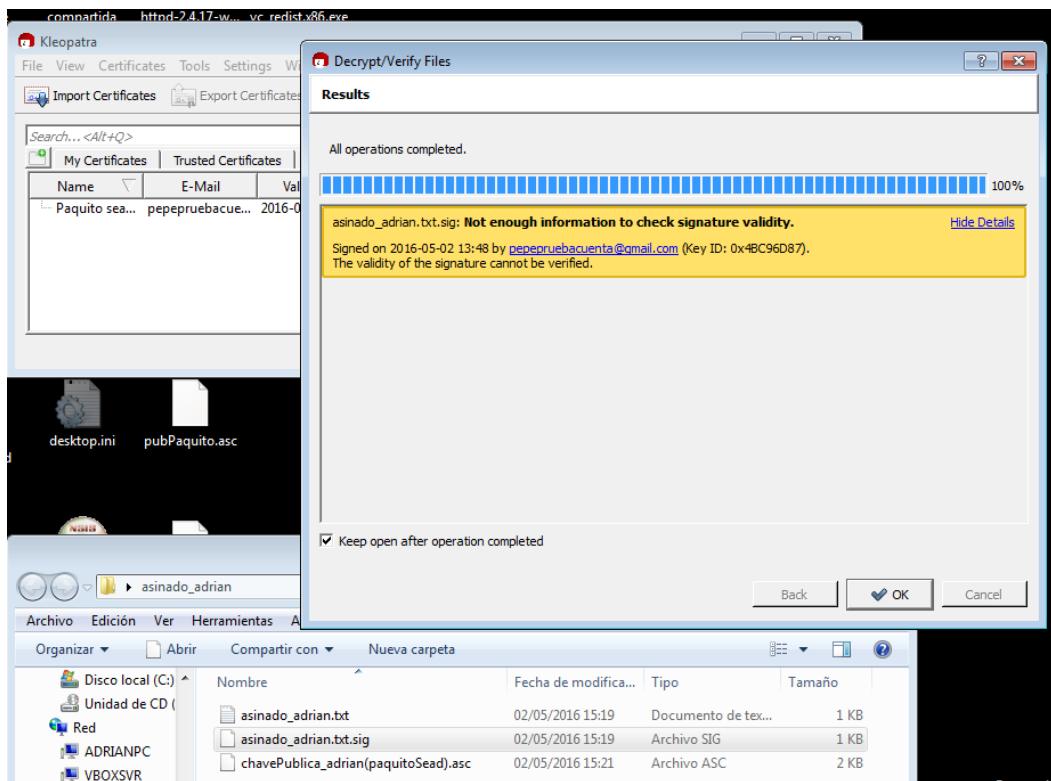
The screenshot shows a Windows desktop environment with the following components:

- A desktop background showing various icons like mimikatz_trunk, mimikatz_t, alberto.asc, Potato-ma, desktop.ini, and pubPaquit.
- An open folder window titled 'asinado_adrian' containing two files: 'asinado_adrian.txt' and 'asinado_adrian.txt.sig'.
- A Kleopatra application window titled 'Decrypt/Verify Files' with the following settings:
 - Input file: C:/Users/adrian/Desktop/asinado_adrian/asinado_adrian.txt.sig
 - Signed data: C:/Users/adrian/Desktop/asinado_adrian/asinado_adrian.txt
 - Create all output files in a single folder (checkbox checked)
 - Output folder: C:/Users/adrian/Desktop/asinado_adrian

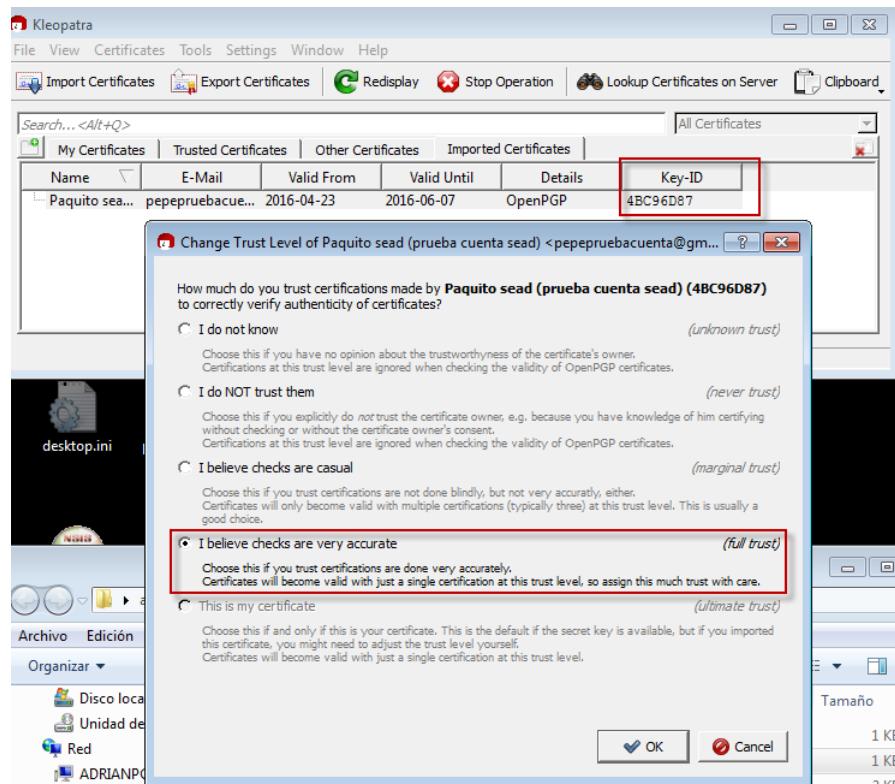
Comprobamos que ainda que vemos o ID de quen o firmou este non foi verificado.



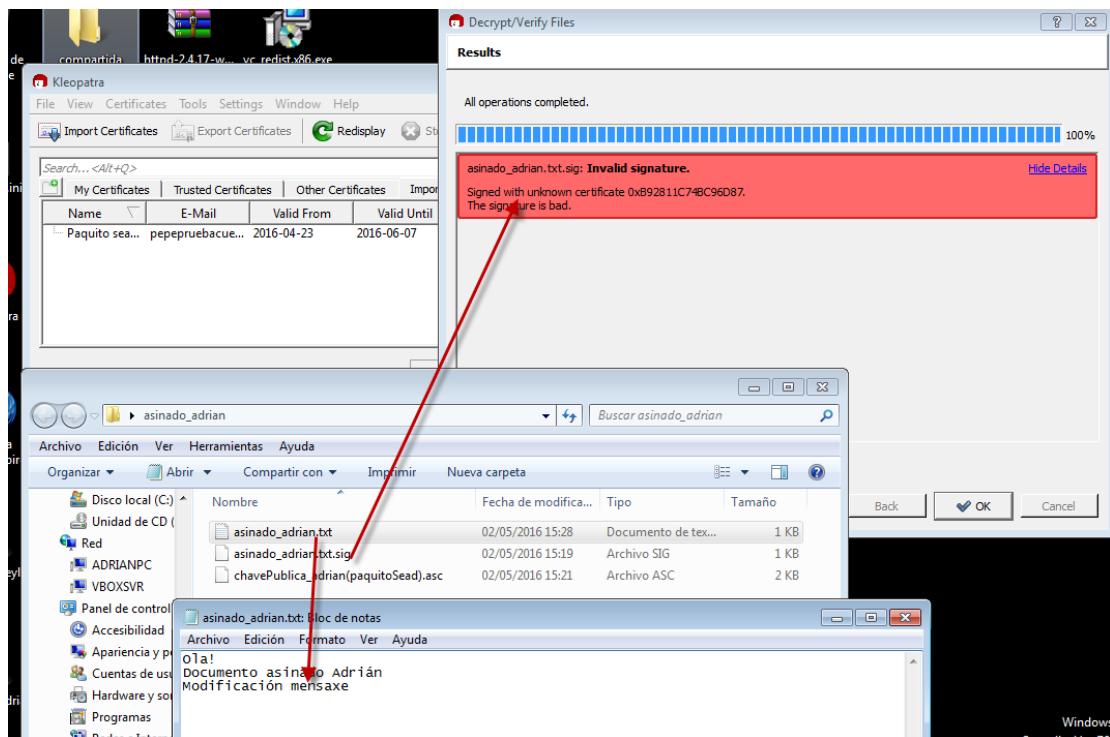
Probamos a importar a chave pública do emisor (Paquito Sead), o resultado e que xa nos verifica quen o firmou ainda que dice que non e unha firma válida.



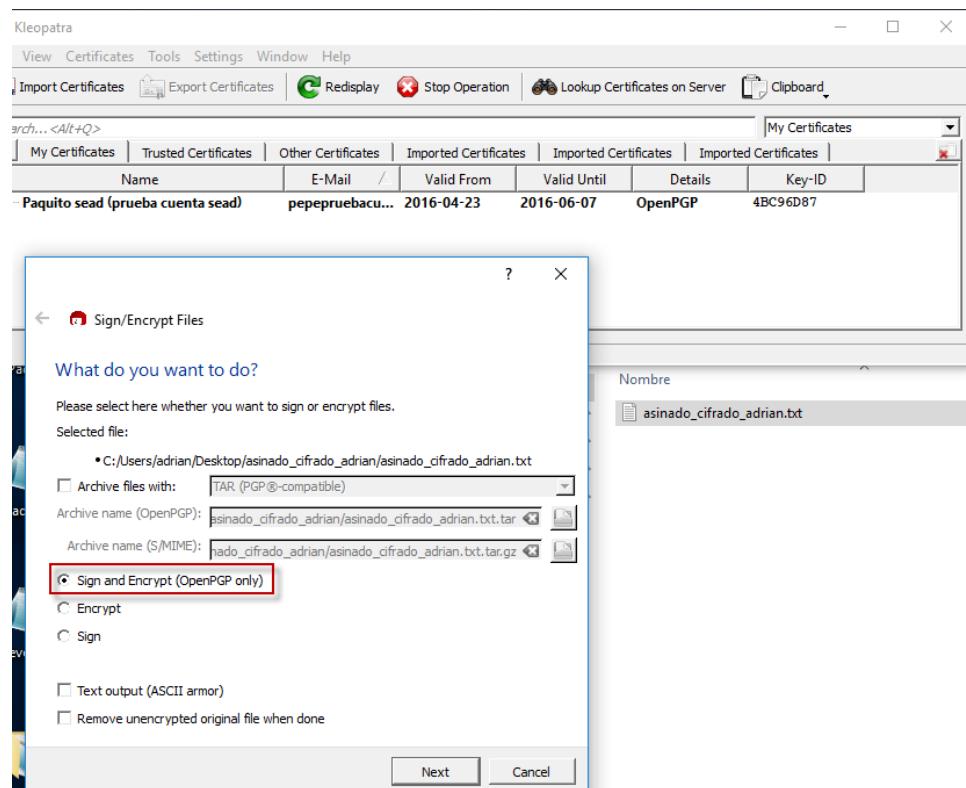
Modificamos o certificado para confiar en el, ainda así sigue recoñecendo como unha firma non válida?.



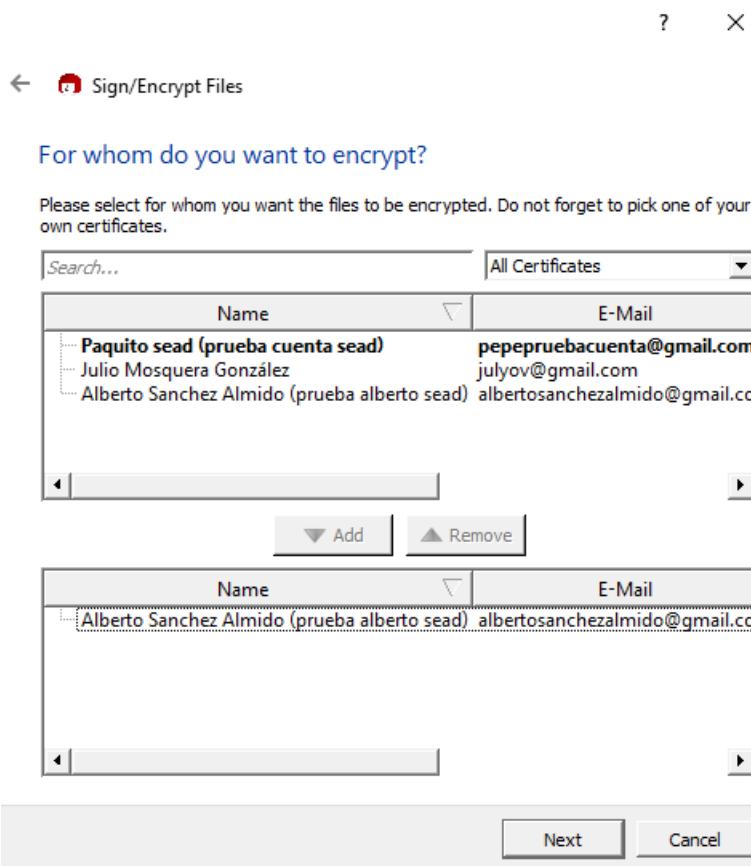
Modificamos a mensaxe orixinal en plain/text (.txt) por exemplo, añadindo unha liña de texto. E comprobamos si a verificación é correcta. Como vemos agora xa non está verificando correctamente a mensaxe, polo que rompeu a sua integridade.



Por último asinaremos unha nova mensaxe a vez que a ciframos, todo nun mesmo arquivo.



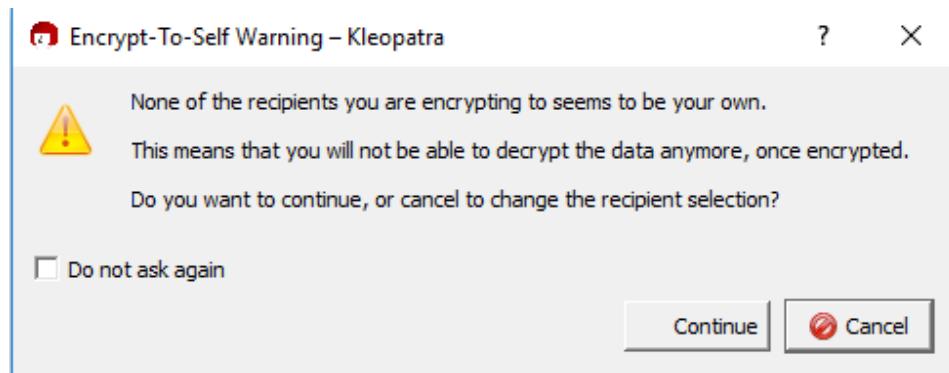
Ciframos a mensaxe ca chave pública de que o quermeos que o reciba, neste caso “Alberto”.



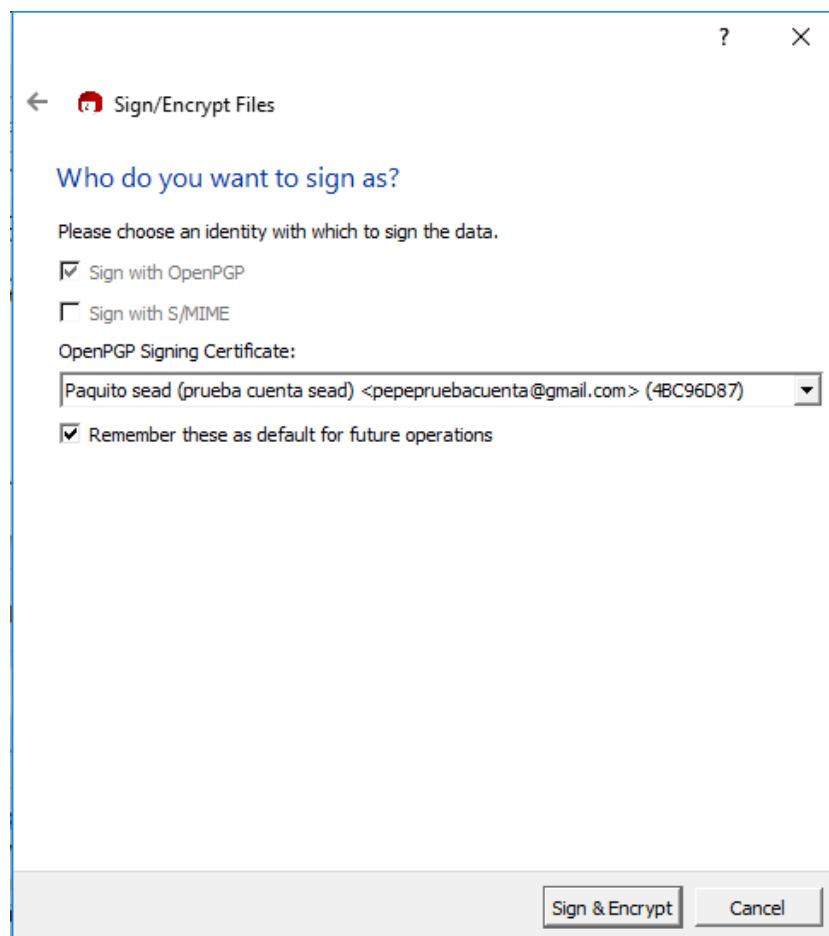
Saltaranos a seguinte alerta, a cal dinos:

- Ningún dos destinatarios parece ser o noso.
- Pode ser que non sexamos capaces de descifrar o noso arquivo coa nosa chave.
- Continuar ou cambiar a selección do receptor?.

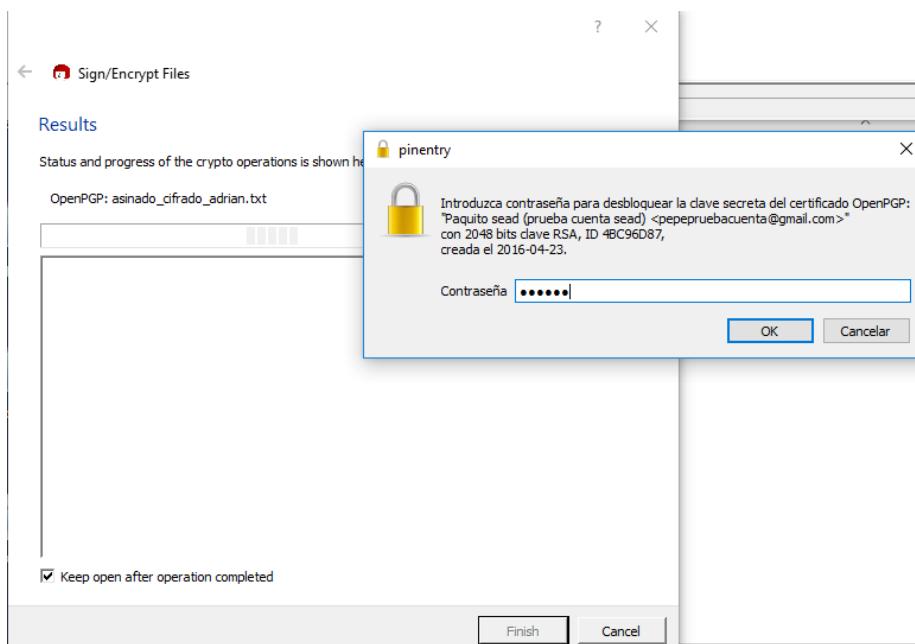
Continuamos.



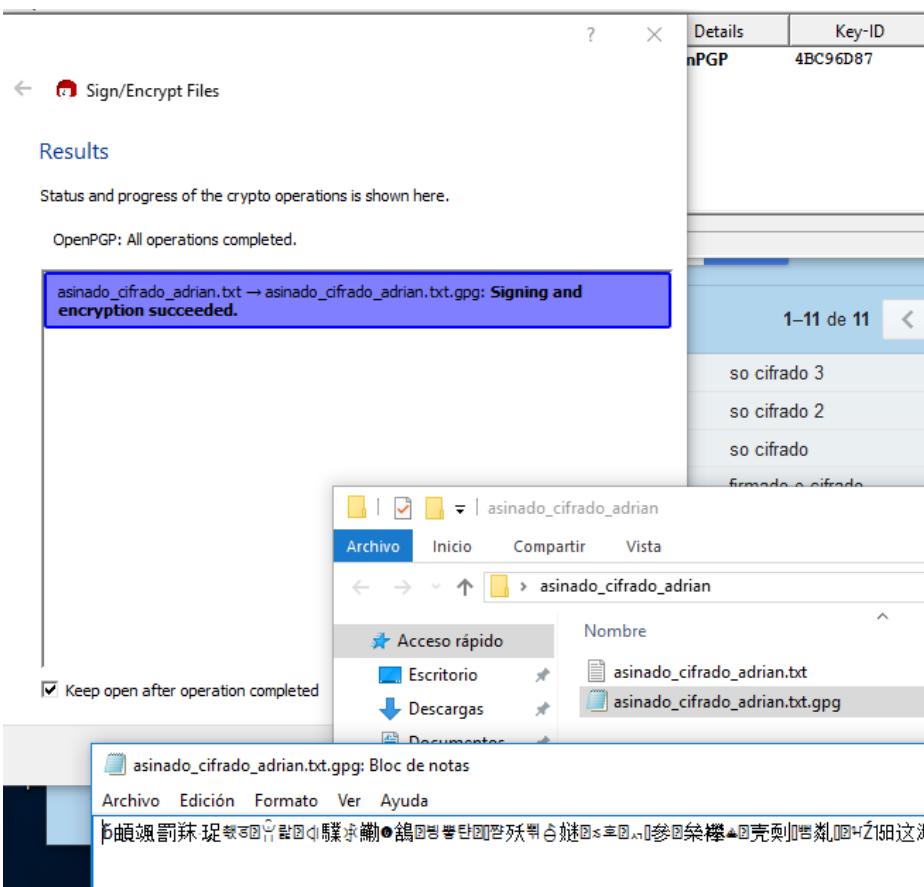
Neste paso cifraremos e firmaremos a mensaxe coa nosa chave privada.



Pediranos a contrasinal da chave privada de que firma o arquivo.



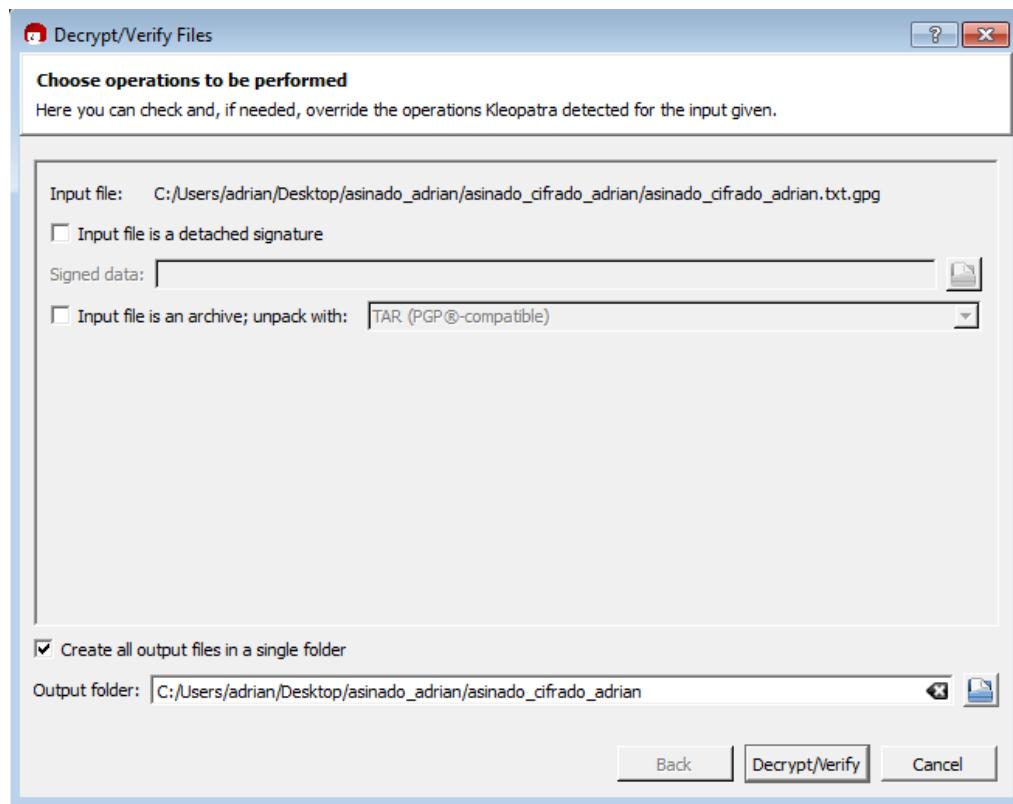
Finalizamos, e vemos que se creou un único arquivo (.txt.gpg) a partir do documento orixinal (.txt) o cal está ilexible a simple vista.



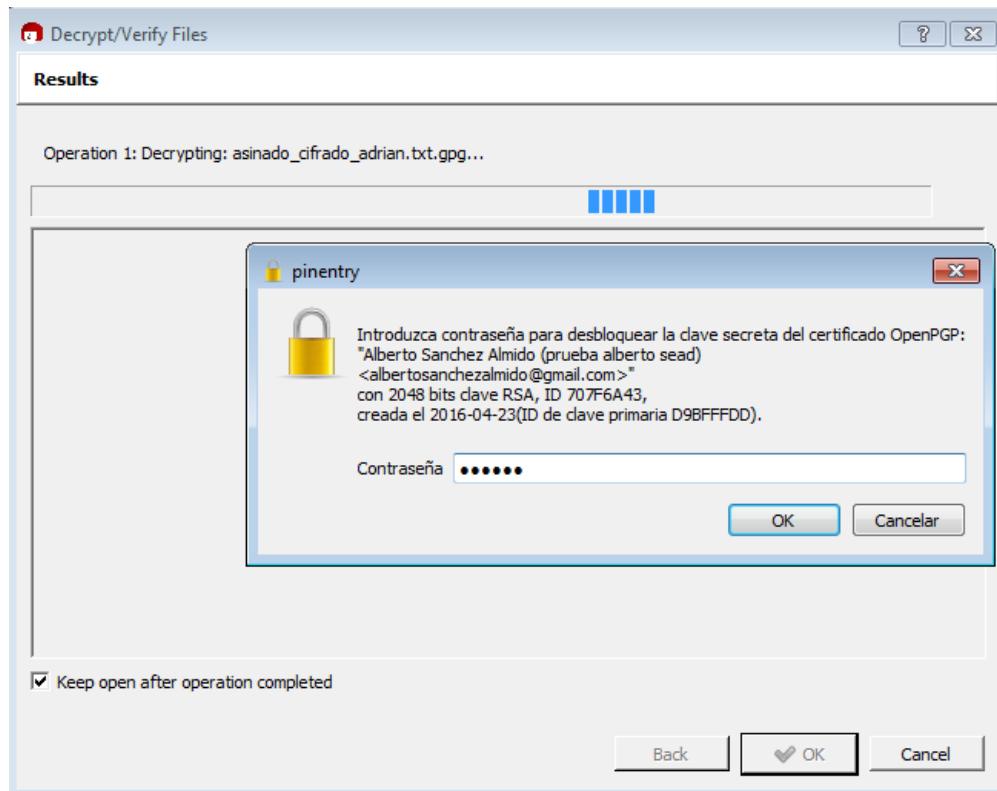
Enviamos este arquivo o destinario que queremos que descifre e verifique a firma do arquivo.

Descargamos o arquivo no equipo do destinatario.

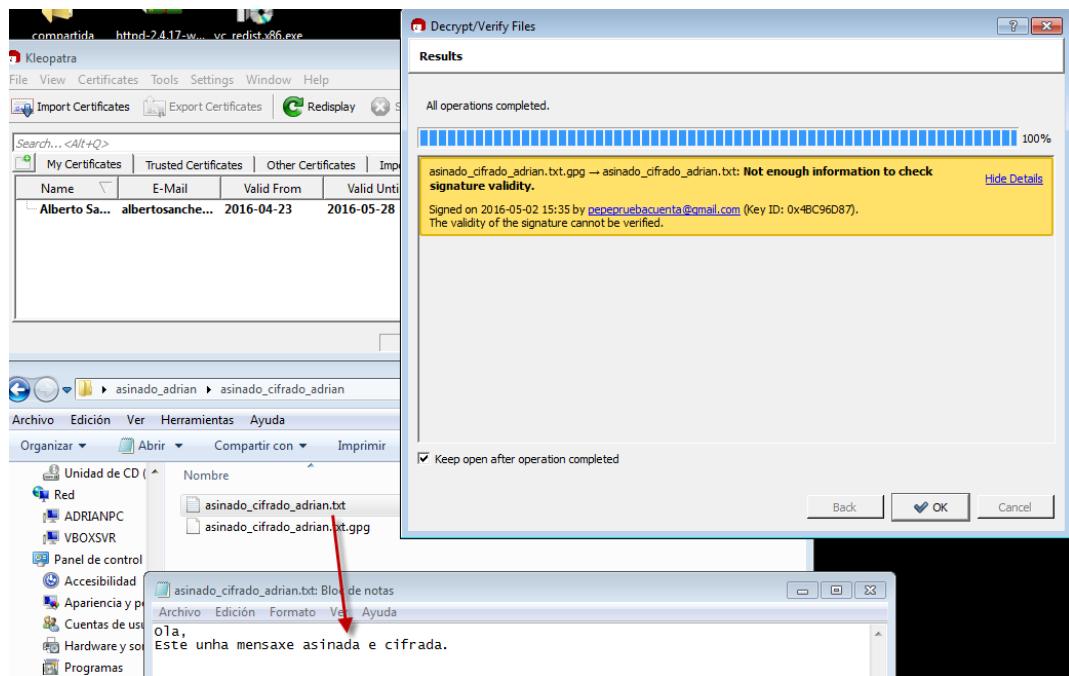
Tentamos descifralo e verificala firma.



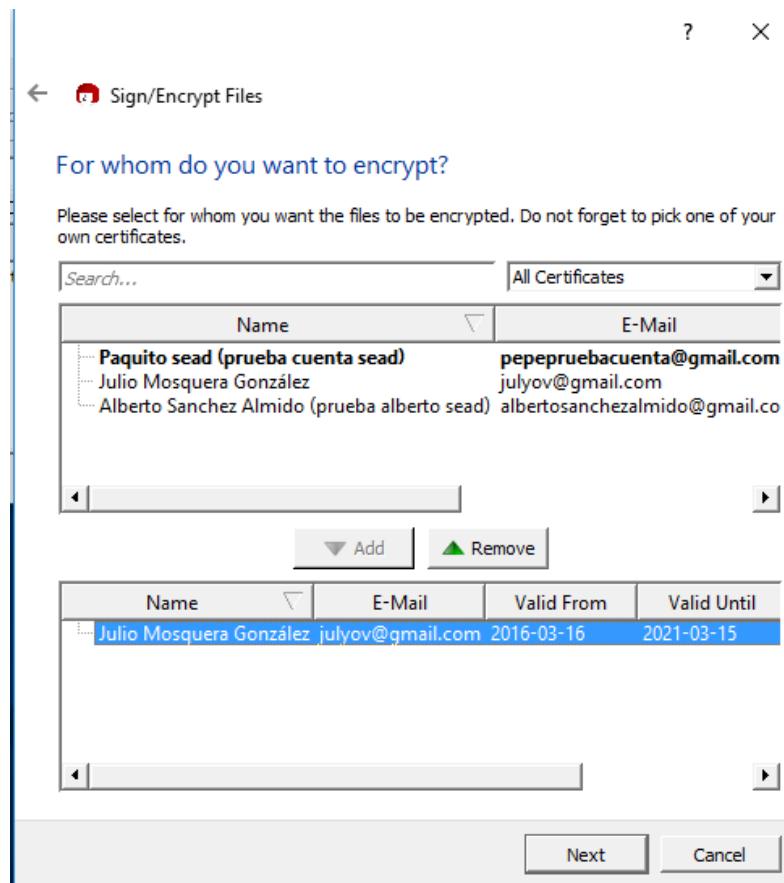
Pediranos a chave privada de “Alberto”, xa que foi cifrado orixinalmente ca súa chave pública o cal quere decir que soamente el poderá descifralo.



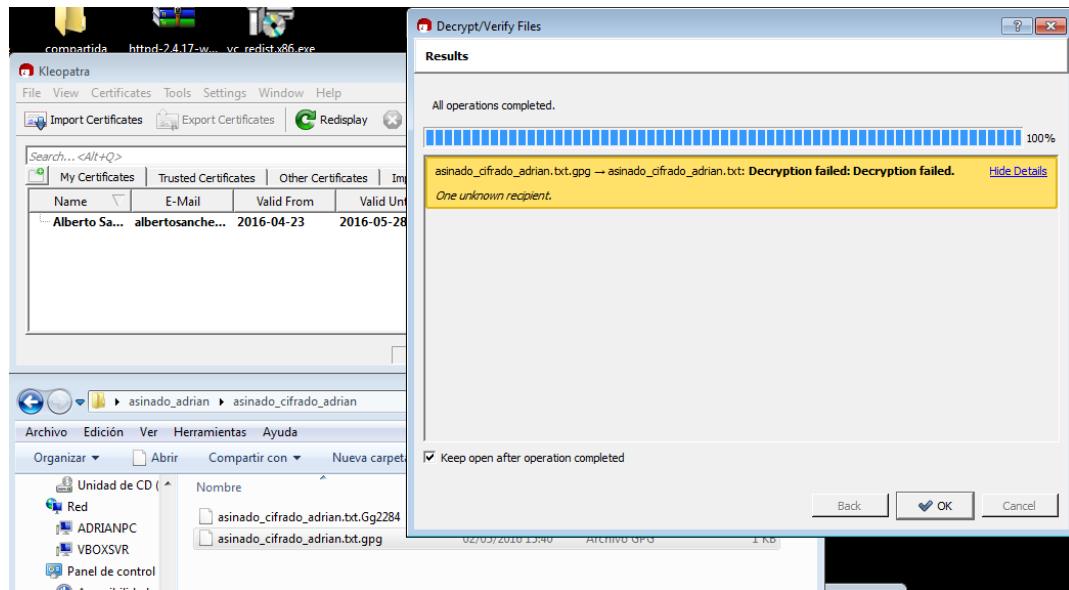
Vemos como a mensaxe se descifrou correctamente, e a verificación de firma está correcta, exceptuando o cor amerelo, o cal e causa da confianza que temos sobre ise certificado emisor.



Se probamos a cifrar a mensaxe con **outra chave pública** (por exemplo: "julyov@gmail.com").



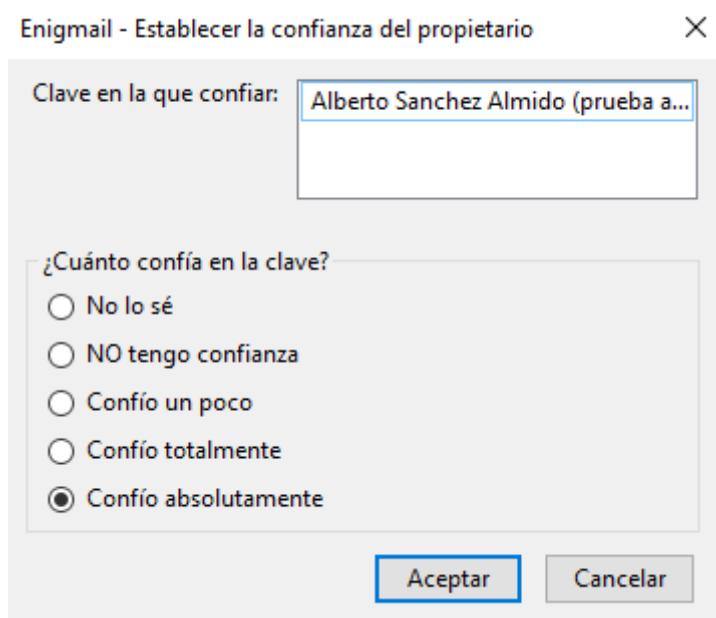
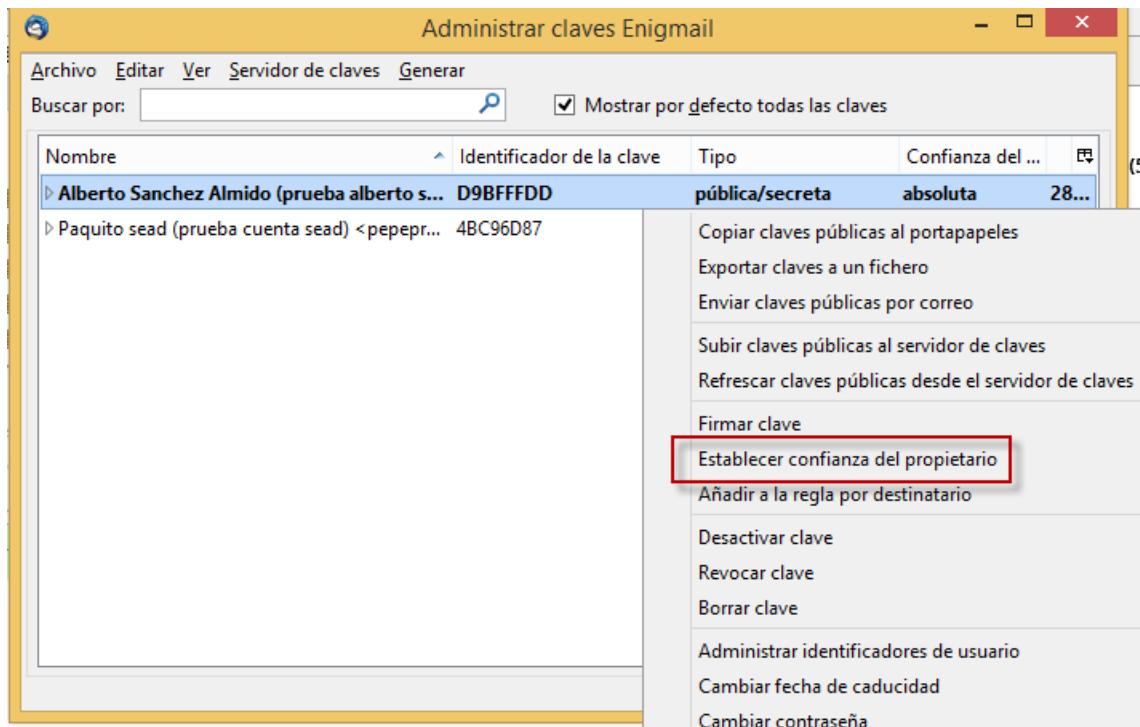
E tentamos descifralo co usuario Alberto veremos que non vamos poder descifrar dita mensaxe, xa que non sería el quen tería que recibir este arquivo, se non "julyov@gmail.com", que era o que nun principio foi o elexido para abrilo.



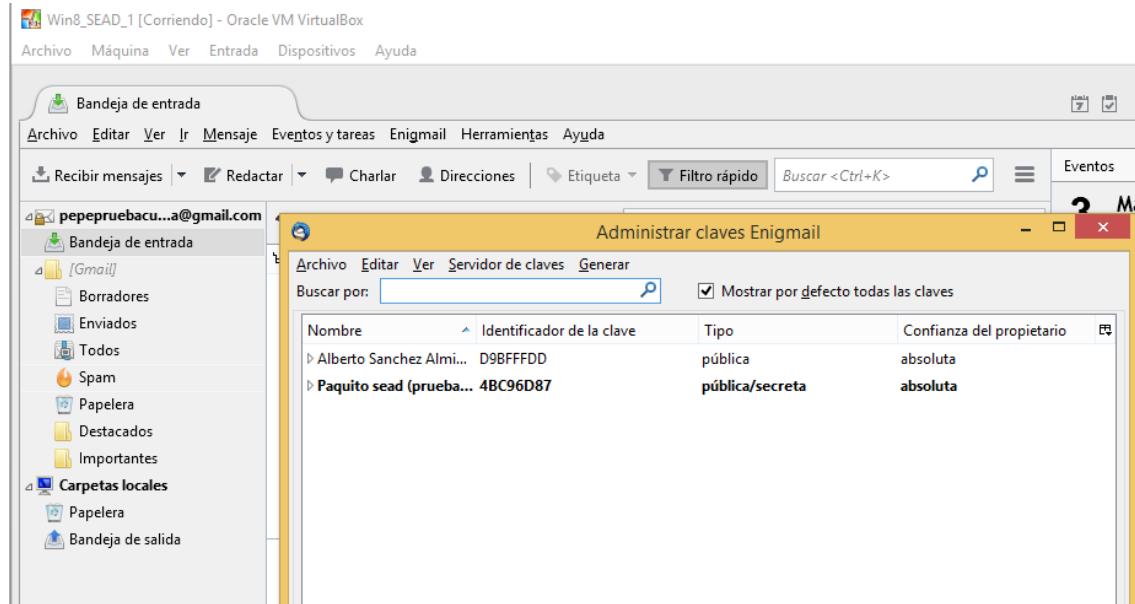
8. Correo electrónico. Sinatura e cifrado

Unha vez instalamos o cliente de correo para escritorio Mozilla Thunderbird máis o complemento para poder firmar e cifrar as mensaxes entre emisor/receptor “Enigmail”.

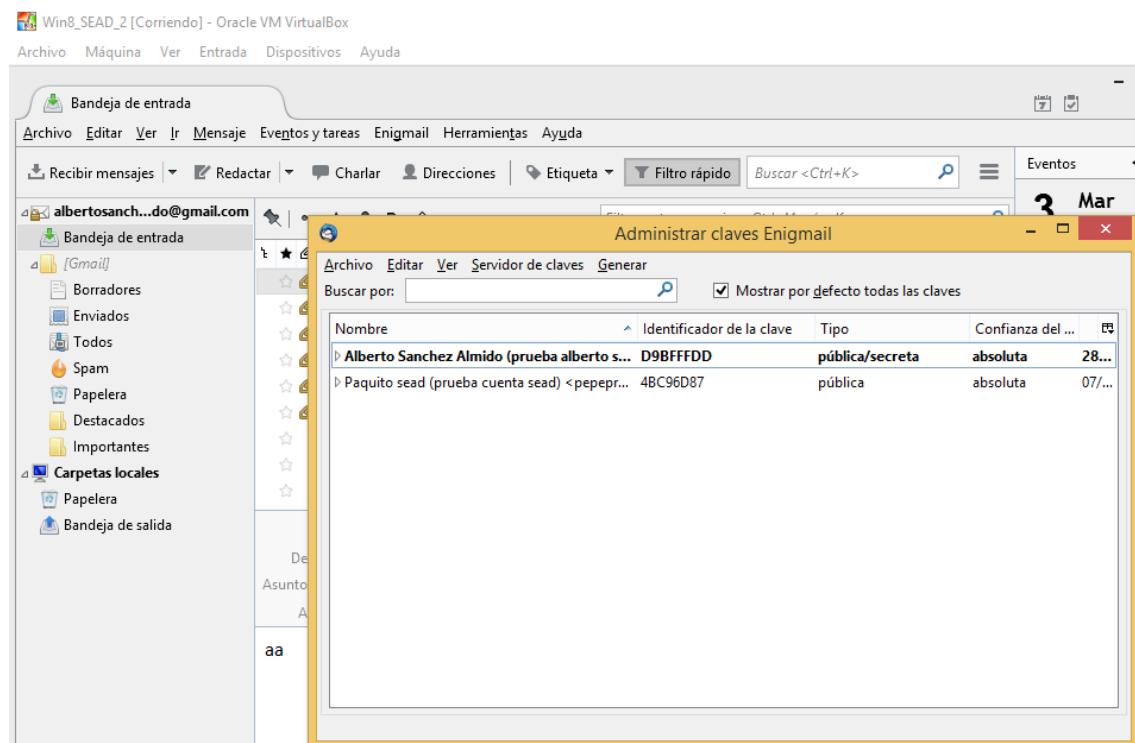
Engadimos os respectivos certificados no administrador de chaves de Enigmail, así como establecemos a súa confianza neles, verificando que realmente o ID dos certificados están correctos e son de quen dicen ser (non repudio).



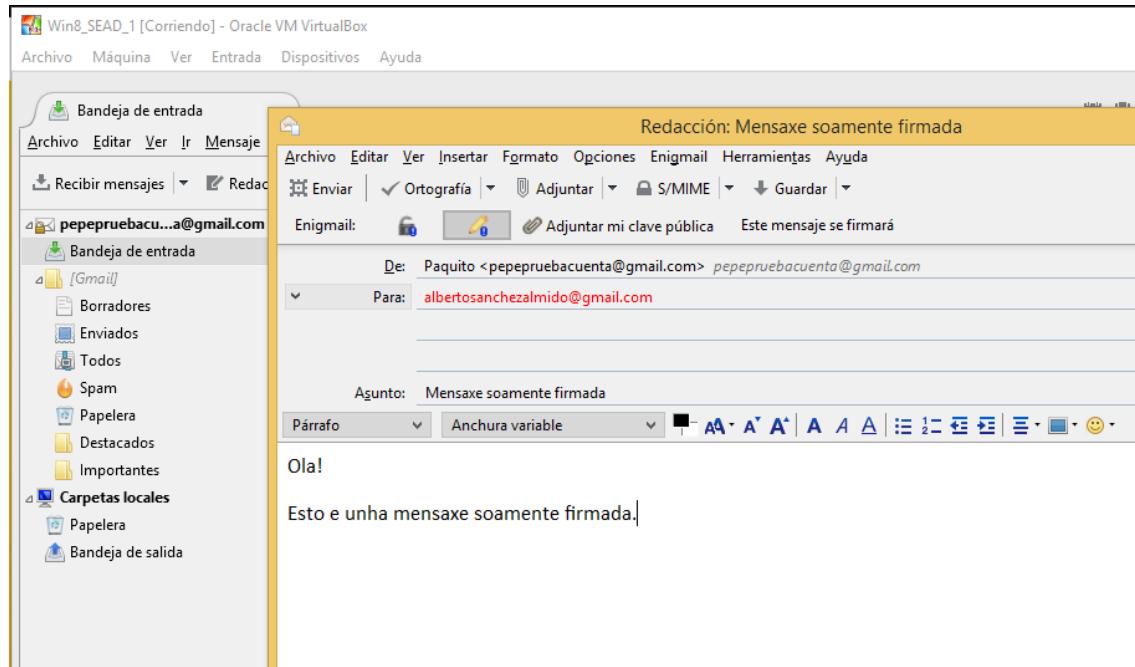
Configuración de administración de certificados no correo de “Paquito Sead”.



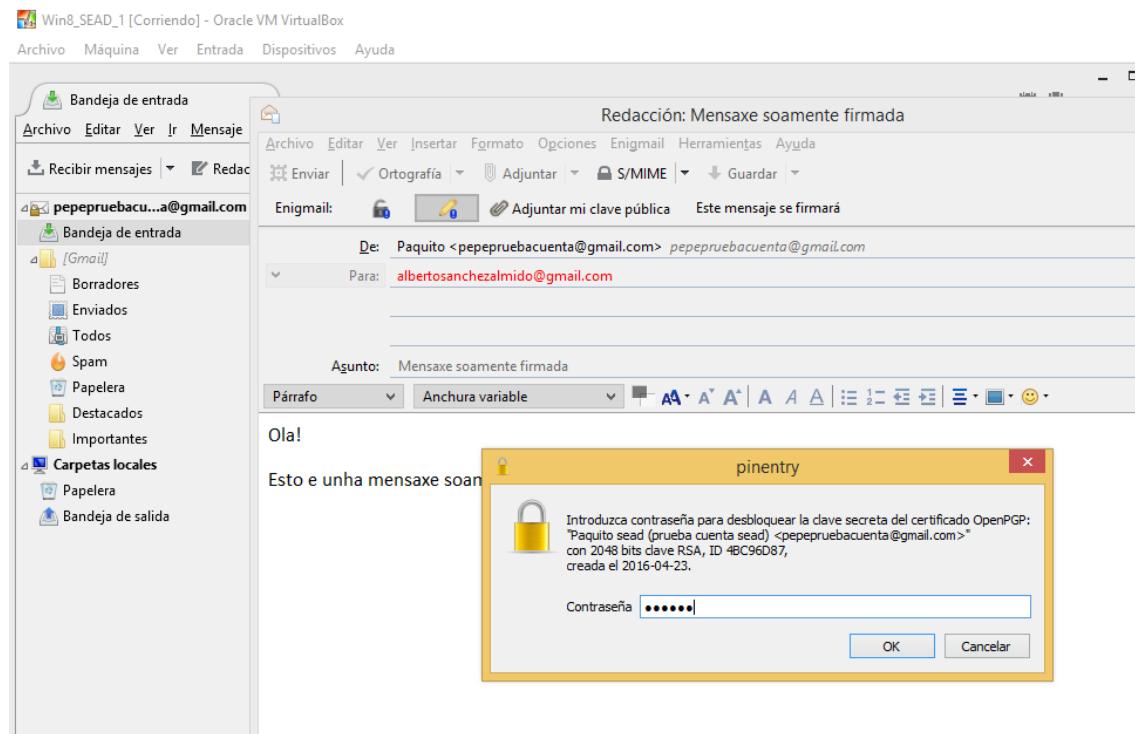
Configuración de administración de certificados no correo de “Alberto”.



Dende a conta de “Paquito” enviamos unha mensaxe soamente firmada a conta de “Alberto”.

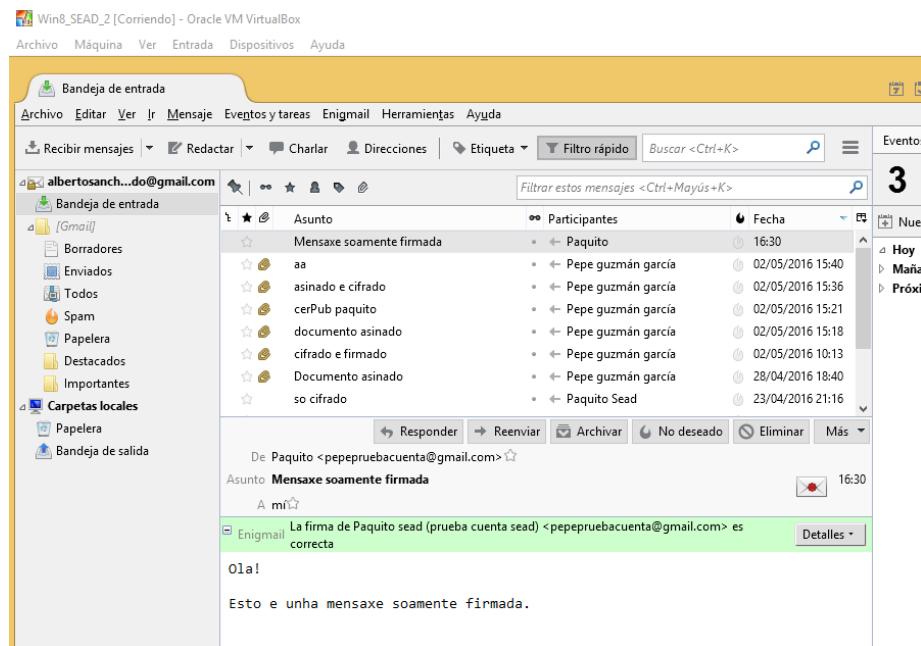


Para verificar a autoridade desta mensaxe pedirános a contrasinal da chave privada de quen o envía, neste caso “Paquito”.

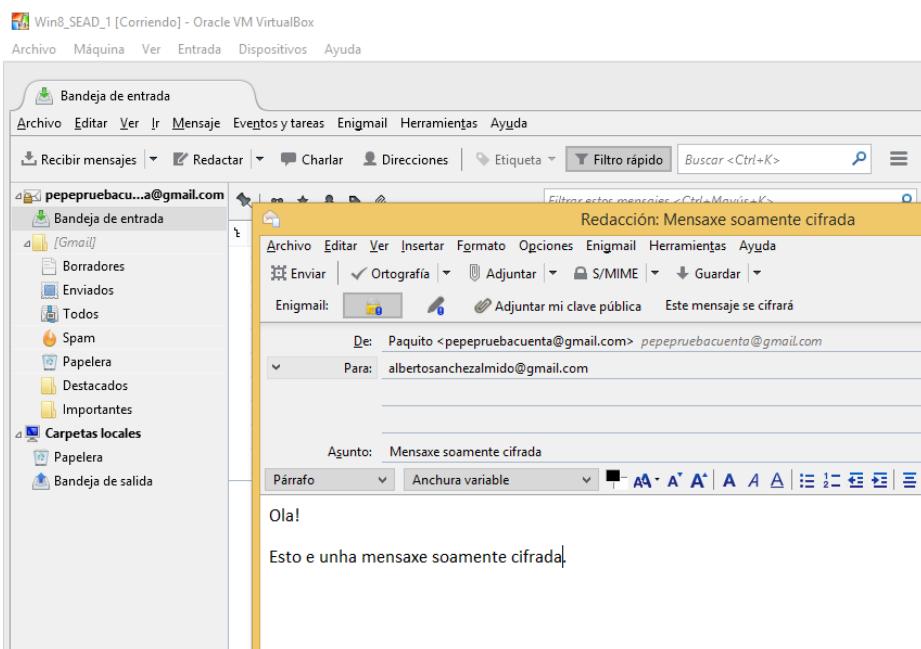


Recibimos a mensaxe na conta de “Alberto”, e verificamos que realmente foi firmada por “Paquito”. Apaerecerá en cor amarela si non temos a chave pública de Paquito como un certificado de confianza (este paso xa o configuramos anteriormente).

Non nos pedirá ninguna contrasinal para poder verificar a firma do mensaxe.

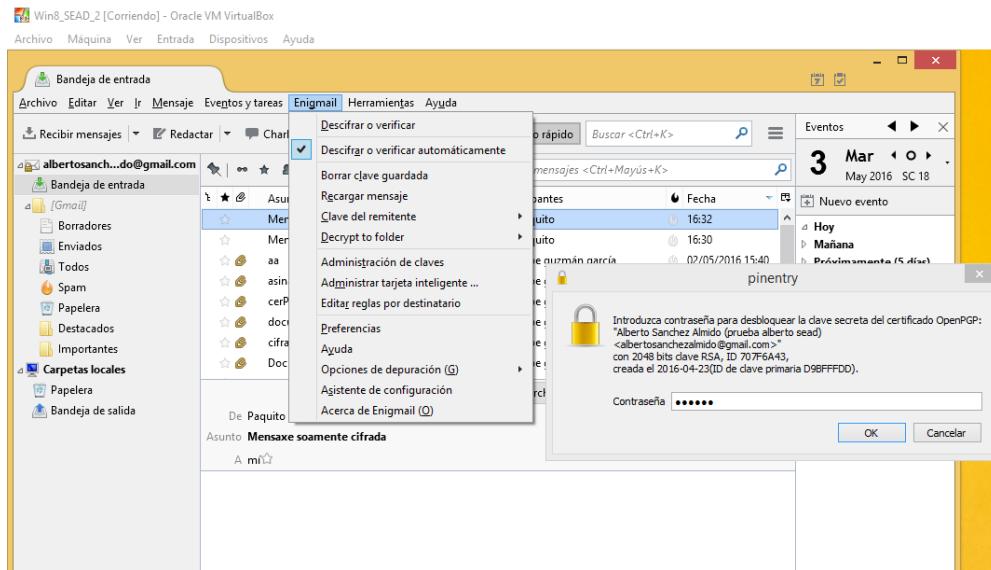


Agora enviamos unha mensaxe soamente cifrada co mesmo procedemento. Esta vez o non firmar a mensaxe non nos pedirá a contrasinal do emisor xa que non sería necesario verificar a identidade. Para poder enviar unha mensaxe será necesario ter a chave pública do receptor “Alberto” (xa que esta cifrarase ca súa chave pública), para poder así cifrar a mensaxe que soamente vai destinada a él, e que él ca súa chave privada poda descrifrar dita mensaxe como se comprobará posteriormente.

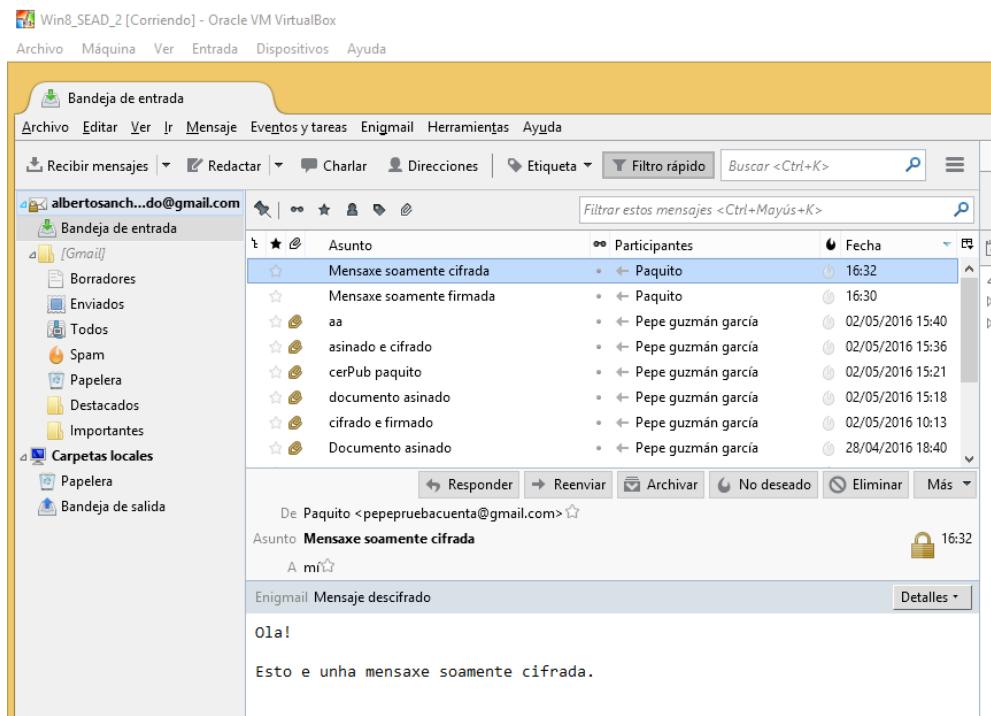


Vemos que “Alberto” xa recibiu a mensaxe soamente cifrada de “Paquito”, neste caso temos a opción (marcada por defecto en enigmail) de poder descifrar automaticamente tódalas mensaxes que procedan deste emisor (esta contrasinal quedará almacenada en enigmail ata que nos decidamos eliminar esa “caché”).

O contrasinal gardada sería a da chave privada do receptor que recibe a mensaxe para podela descifrar, que soamente nola preguntará a primeira vez no caso de que teñamos marcado ese checkbox, no caso de que o teñamos descamardo sempre, entón teremos que introducir a contrasinal da chave privada para descifrar todas as mensaxes recibidas xa sexa dun mesmo emisor ou de outro distinto.

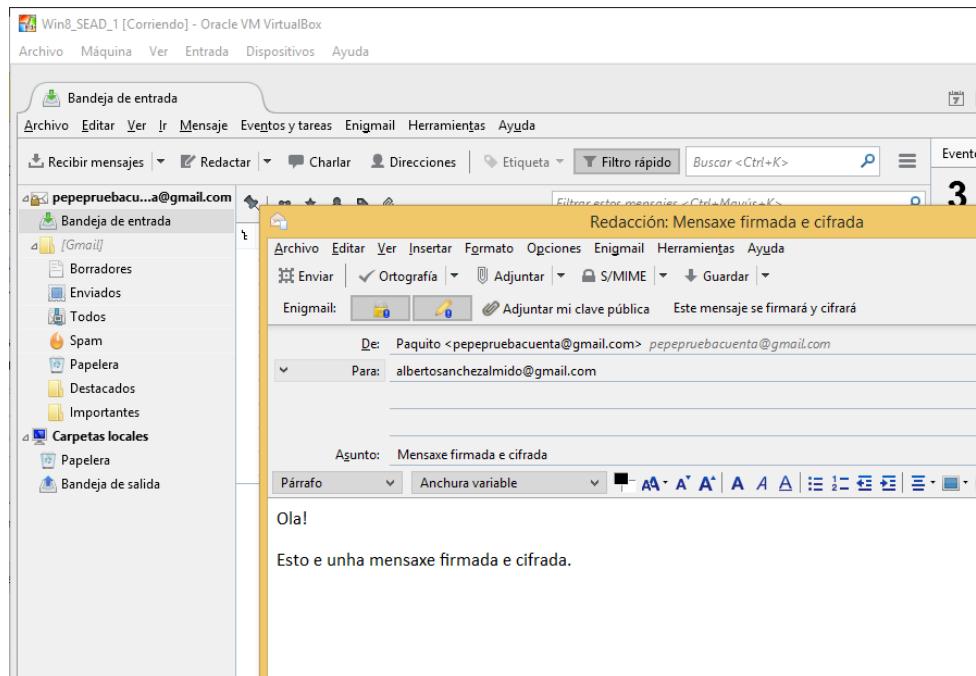


Comprobamos que despois de descifrala vemos a mensaxe cifrada recibida.

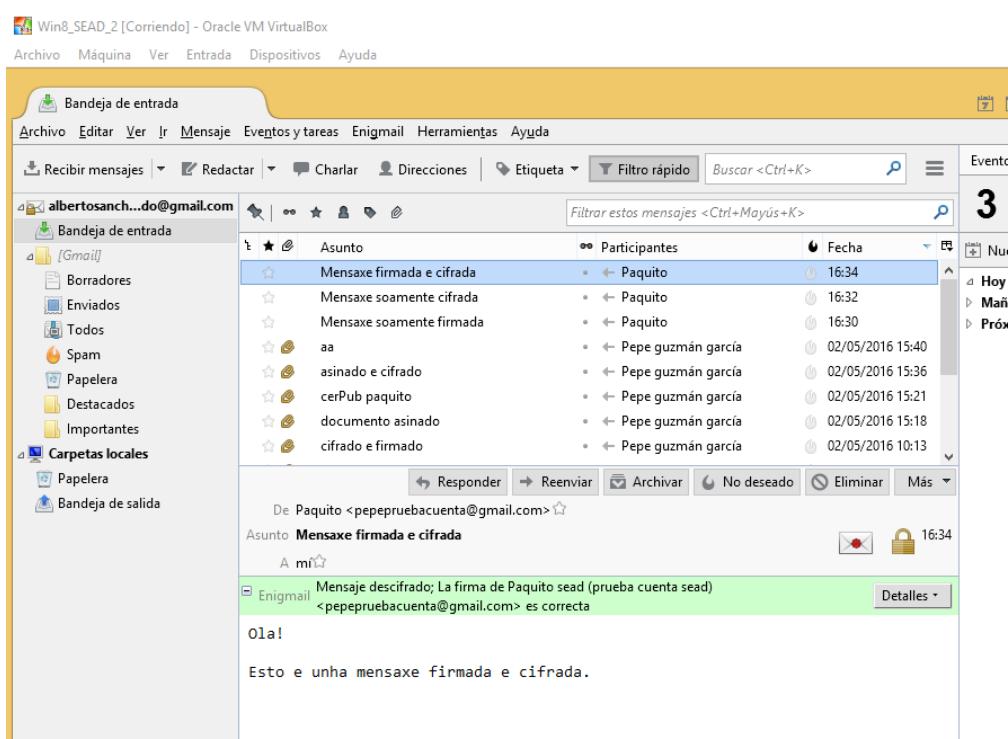


Por último faremos o mesmo procedemento pero esta vez firmando e cifrando a mesma mensaxe. Como xa se comentou antes, no caso de que non teñamos marcado o checkbox de alamacenaxe de contrasinais, pediríamos de novo a contrasinal da chave privada de quen quere enviar a mensaxe (neste caso “Paquito”) no caso de ter marcado dito checkbox no nos pedirá dita contrasinal.

Esta contrasinal sería como no primer caso visto, para firmar a mensaxe (non para cifrala).

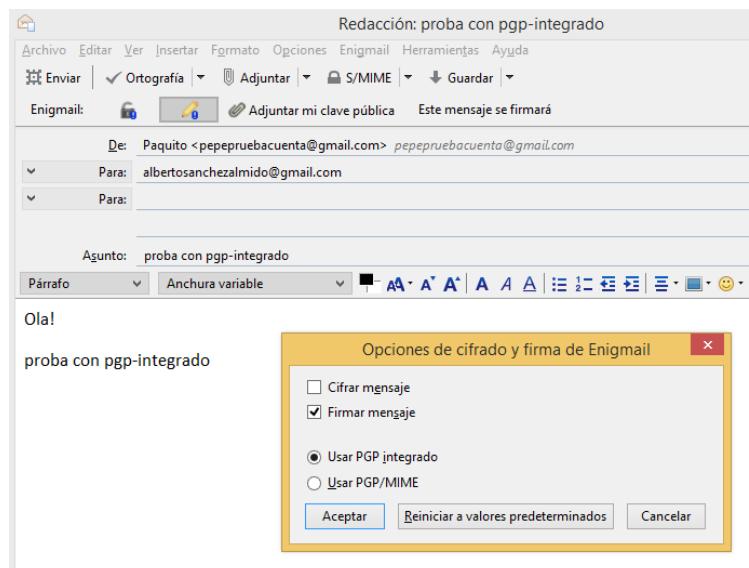


O recibir a mensaxe na conta de correo de “Alberto” vemos como esta a pode descifrar e verificar a procedencia da autoridade da firma, asinada por “Paquito”.



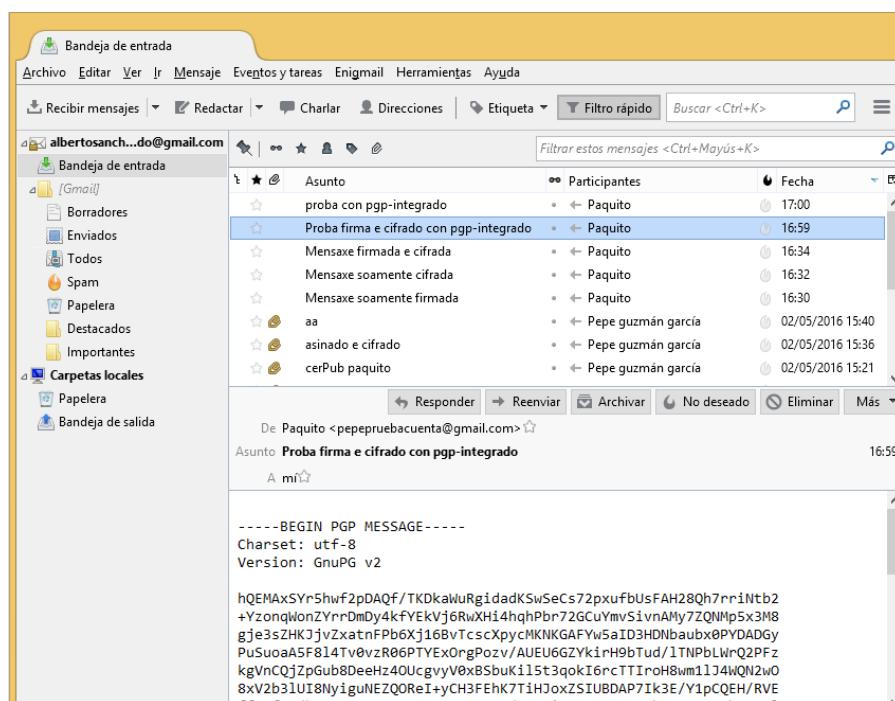
Como última nota añado a funcionalidade que nos permite Enigmail de poder usar: “PGP Integrado” ou “PGP/MIME”, MIME (*Multipurpose Internet Mail Extensions*) é unha especificación a cal permitiríanos “ocultar” a propia chave pública ca que foi firmada unha mensaxe. E decir, que a parte do contenido de informaicon da propia mensaxe añadese o final dita chave pública en plain-text. MIME permítenos ocultar o contenido de texto da chave pública xa que non queda “moi vistoso” e pode ser incómodo para os usuarios finais ter que ver esa información a maiores.

No caso de firmala con “PGP Integrado”.



A mensaxe que vería o receptor sería algo así.

NOTA: Enigmail, igualmente sigue ocultando este texto ainda que si cambiamos rápido dunha mensaxe a outra no buzón de entrada do correo, temos un retardo de tempo no que si que o mostra.



9. Esteganografía

Para esta tarefa faremos uso da utilidade steghide a cal instalaremos nun Ubuntu.

Deixo unha ligazón sobre este tema:

<http://www.zonasystem.com/2016/02/steghide-ocultar-informacion-dentro-de.html>

Neste exemplo pídense ocultar unha ficheiro que teña contido sensible nunha imaxe, e decir, enmascarar dito ficheiro.

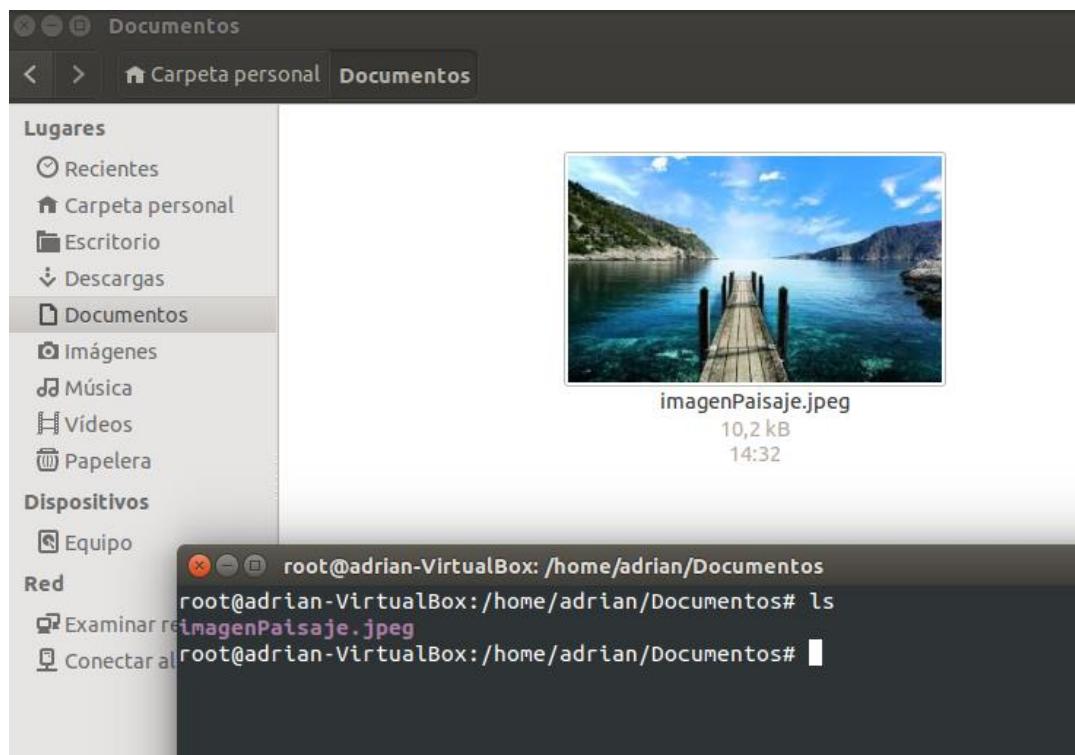
A sintaxis do comando sería:

steghide embed -cf {ficheiro_imaxe} -ef {ficheiro_a_ocultar}

Pediranos unha contrasinal.

```
root@adrian-VirtualBox: /home/adrian/Documentos
root@adrian-VirtualBox: /home/adrian/Documentos# ls
contrasinais  imagenPaisaje.jpeg
root@adrian-VirtualBox: /home/adrian/Documentos# cat contrasinais
Aqui hai contrasinais salvagardadas
root@adrian-VirtualBox: /home/adrian/Documentos# steghide embed -cf imagenPaisaje
.jpeg -ef contrasinais
Anotar salvoconducto:
Re-ingresar salvoconducto:
adjuntando "contrasinais" en "imagenPaisaje.jpeg"...
hecho
root@adrian-VirtualBox: /home/adrian/Documentos#
```

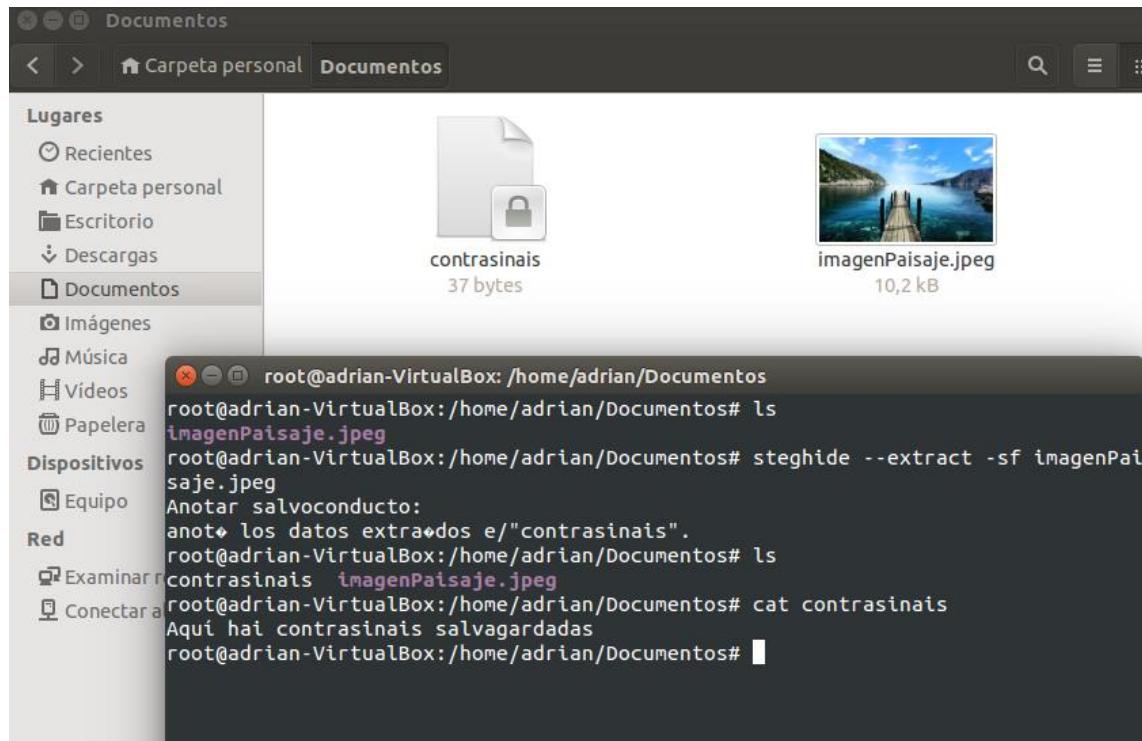
Vemos como so se mostra un arquivo final, o cal contén tamén o arquivo con información sensible, neste caso.



Para descifralo simplemente usaremos a seguinte sintaxis:

steghide --extract -sf {ficheiro_imaxe}

Pediranos a contrasinal establecida, e listo, xa tendremos separados de novo os dous archivos.



10. Conclusóns

Como conclusión de todas estas tarefas sabemos que hai mecanismos, técnicas e ferramentas que nos proporcionan a autenticidade, integridade, ocultación de información e cifrado e firmado de documentos.

Algunhas destas técnicas son más ou menos laboriosas e algunas outras más custosas de comprender, estos mecanismos cada vez intentáse axustar máis a demanda de usuarios finais, os cales sin moitos coñecementos de informática poidan usar estas técnicas para poder cifrar e manter segura a súa información xa sexa en local ou a través da rede de Internet.

Es os cales tentan proporcionar unha maior seguridade para todos.