Miguel Córdoba Aranda 2º ASIR SEGURIDAD

PRACTICA 1. CIFRADO ASIMÉTRICO CON GPG Y OPENSSL

TAREA1: GENERACIÓN DE CLAVES

1. Genera un par de claves (pública y privada). ¿En que directorio se guarda las claves de un usuario?

Para generar las claves hemos usado el comando **#gpg --gen-key** y para comprobar que se han creado las claves he usado el comando **#gpg -k** para la pública y **#gpg -K** para la privada.

```
root@MCA:/home/miguel# gpg --gen-key
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.

gpg: creado el directorio '/root/.gnupg'
gpg: caja de claves '/root/.gnupg/pubring.kbx' creada
Nota: Usa "gpg --full-generate-key" para el diálogo completo de generación de clave.

GnuPG debe construir un ID de usuario para identificar su clave.

Nombre y apellidos: Miguel Cordoba
Dirección de correo electrónico: miguelcor.rrss@gmail.com
Ha seleccionado este ID de usuario:
    "Miguel Cordoba <miguelcor.rrss@gmail.com>"

¿Cambia (N)ombre, (D)irección o (V)ale/(S)alir? v
Es necesario generar muchos bytes aleatorios. Es una buena idea realizar
alguna otra tarea (trabajar en otra ventana/consola, mover el ratón, usar
la red y los discos) durante la generación de números primos. Esto da al
generador de números aleatorios mayor oportunidad de recoger suficiente
entropía.
Es necesario generar muchos bytes aleatorios. Es una buena idea realizar
alguna otra tarea (trabajar en otra ventana/consola, mover el ratón, usar
la red y los discos) durante la generación de números primos. Esto da al
generador de números aleatorios mayor oportunidad de recoger suficiente
entropía.
gpg: /root/.gnupg/trustdb.gpg: se ha creado base de datos de confianza
gpg: clave 93E00F9A8C74FBC0 marcada como de confianza absoluta
gpg: creado el directorio '/root/.gnupg/openpgp-revocs.d'
gpg: certificado de revocación guardado como '/root/.gnupg/openpgp-revocs.d/0F99E175536
0586A9B7C1F9C93E00F9A8C74FBC0.rev'
claves pública y secreta creadas y firmadas.

pub   rsa3072 2021-11-11 [SC] [caduca: 2023-11-11]
      0F99E1755360586A9B7C1F9C93E00F9A8C74FBC0
uid                      Miguel Cordoba <miguelcor.rrss@gmail.com>
sub   rsa3072 2021-11-11 [E] [caduca: 2023-11-11]
```

El la ruta **#/root/.gnupg /pubring.kbx** se guardan tanto la pública como la privada.

2. Lista las claves públicas que tienes en tu almacén de claves. Explica los distintos datos que nos muestra. ¿Cómo deberías haber generado las claves para indicar, por ejemplo, que tenga un 1 mes de validez?

Para listar las claves de mi almacén he usado el comando **#gpg --list-keys**.

```
root@MCA:/home/miguel# gpg --list-keys
gpg: comprobando base de datos de confianza
gpg: marginals needed: 3  completes needed: 1  trust model: 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: 2021-12-11
/root/.gnupg/pubring.kbx
------------------------
pub   rsa3072 2021-11-11 [SC] [caduca: 2021-12-11]
      0F99E1755360586A9B7C1F9C93E00F9A8C74FBC0
uid        [  absoluta ] Miguel Cordoba <miguelcor.rrss@gmail.com>
sub   rsa3072 2021-11-11 [E] [caduca: 2023-11-11]
```

Marginales Necesarias →
Completas Necesarias →
Modelo de Confianza: pgp → Este campo define que firmas de clave seguir.
Nivel 0 → Indica que la clave no caduca.
Validez 1 →
Firmada → Es un contador que aumenta tantas veces como se haya firmado esa clave.
Confianza →
---
pub → Nos indica que es una clave pública,
rsa3072 → Tipo de clave pública e indica los bits de longitud, que por defecto son **3072** bits.
2021-11-11 → Fecha de Creación de la clave.
[caduca: 2021-12-11] → Como el campo indica, es la fecha de caducidad de la clave.
UID → Es el identificador único de la clave.

Para indicar la fecha de caducidad desde el primer momento hay que usar el comando **#gpg --full-generate-key**. Con este comando nos preguntará el tipo de **clave y su longitud**,

```
miguel@MCA:~$ gpg --full-generate-key
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.

Por favor seleccione tipo de clave deseado:
   (1) RSA y RSA (por defecto)
   (2) DSA y ElGamal
   (3) DSA (sólo firmar)
   (4) RSA (sólo firmar)
  (14) Existing key from card
Su elección: 1
las claves RSA pueden tener entre 1024 y 4096 bits de longitud.
¿De qué tamaño quiere la clave? (3072)
El tamaño requerido es de 3072 bits
```

el **tiempo de caducidad**; cuando confirmemos que está correcto nos pasará una frase de paso para proteger las claves;

```
Por favor, especifique el período de validez de la clave.
         0 = la clave nunca caduca
      <n>  = la clave caduca en n días
      <n>w = la clave caduca en n semanas
      <n>m = la clave caduca en n meses
      <n>y = la clave caduca en n años
¿Validez de la clave (0)? 0
La clave nunca caduca
¿Es correcto? (s/n) s
```

y por últimos nos pedirá unas credenciales tales como nuestro **nombre y apellido**, **correo** y algún posible **comentario (no obligatorio)**.

```
GnuPG debe construir un ID de usuario para identificar su clave.

Nombre y apellidos: Miguel Córdoba
Dirección de correo electrónico: miguelcor.rrss@gmail.com
Comentario: prueba
Está usando el juego de caracteres 'utf-8'.
Ha seleccionado este ID de usuario:
    "Miguel Córdoba (prueba) <miguelcor.rrss@gmail.com>"
```

3. Lista las claves privadas de tu almacén de claves.

Para listar las claves privadas he usado el comando **`#gpg -K`**.

```
root@MCA:/home/miguel# gpg -K
/root/.gnupg/pubring.kbx
------------------------
sec   rsa3072 2021-11-11 [SC] [caduca: 2021-12-11]
      0F99E1755360586A9B7C1F9C93E00F9A8C74FBC0
uid           [  absoluta ] Miguel Cordoba <miguelcor.rrss@gmail.com>
ssb   rsa3072 2021-11-11 [E] [caduca: 2023-11-11]
```

TAREA 2: IMPORTAR/EXPORTAR CLAVE PÚBLICA

1. Exporta tu clave pública en formato ASCII y guárdalo en un archivo `nombre_apellido.asc` y envíalo al compañero con el que vas a hacer esta práctica.

Para exportar mi clave pública he usado el comando:
`#gpg --export -a “Miguel Cordoba” > Miguel_cordoba.asc`.

```
miguel@MCA:~$ ls | egrep Miguel_cordoba.asc
Miguel_cordoba.asc
```

El compañero con el que voy a hacer esta práctica es **Omar Elhani** y le he enviado mi clave pública a través de **scp**.

2. Importa las claves públicas recibidas de vuestro compañero y Comprueba que las claves se han incluido correctamente en vuestro keyring.

Para importar la clave de mi compañero he usado el comando:
#gpg --import omar_elhani.asc y luego comprobaremos que se ha importado con éxito usando **#gpg -k** .

```
root@MCA:/home/miguel# gpg --import omar_elhani.asc
gpg: clave 17EC279EAC46DAA8: clave pública "omar elhani <omar.elhani1@gmail.com>" impor
tada
gpg: Cantidad total procesada: 1
gpg:               importadas: 1
root@MCA:/home/miguel# gpg -k
/root/.gnupg/pubring.kbx
------------------------
pub   rsa3072 2021-11-11 [SC] [caduca: 2021-12-11]
      0F99E1755360586A9B7C1F9C93E00F9A8C74FBC0
uid        [  absoluta ] Miguel Cordoba <miguelcor.rrss@gmail.com>
sub   rsa3072 2021-11-11 [E] [caduca: 2023-11-11]

pub   rsa3072 2021-11-11 [SC] [caduca: 2023-11-11]
      519FF0F6205352C7646C3CF417EC279EAC46DAA8
uid        [desconocida] omar elhani <omar.elhani1@gmail.com>
sub   rsa3072 2021-11-11 [E] [caduca: 2023-11-11]
```

NOTA: He unido el Ejercicio 2 y 3.

TAREA 3. CIFRADO ASIMÉTRICO CON CLAVES PÚBLICAS

1. Cifraremos un archivo cualquiera y lo remitiremos por email a uno de nuestros compañeros que nos proporcionó su clave pública.

Primero crearemos el fichero **#nano miralosipuedes.txt** y luego lo cifraremos con el comando **#gpg -e u "Miguel Cordoba" -r "Omar Elhani" miralosipuedes.txt** .

```
miguel@MCA:~$ ls | egrep miralosipuedes
miralosipuedes.txt
miralosipuedes.txt.gpg
```

2. Nuestro compañero, a su vez, nos remitirá un archivo cifrado para que nosotros lo descifremos.

Omar me ha enviado el fichero **apuntes.txt.gpg** por email y lo he descifrado usando el siguiente comando **#gpg -d apuntes.txt.gpg** .

3. Tanto nosotros como nuestro compañero comprobaremos que hemos podido descifrar los mensajes recibidos respectivamente.

```
miguel@MCA:~/Descargas$ sudo su
root@MCA:/home/miguel/Descargas# gpg -d apuntes.txt.gpg
gpg: cifrado con clave de 3072 bits RSA, ID 264D8A7CC114656B, creada el 2021-11-11
      "Miguel Cordoba <miguelcor.rrss@gmail.com>"
Hola
```

4. Por último, enviaremos el documento cifrado a alguien que no estaba en la lista de destinatarios y comprobaremos que este usuario no podrá descifrar este archivo.

En mi caso le he enviado el documento a Antonio Castro por email.

5. Para terminar, indica los comandos necesarios para borrar las claves públicas y privadas que posees.

Primero tienes que borrar la clave privada con **#gpg --delete-secret-key "Miguel Cordoba"** y te pedirá confirmación y luego borra la clave pública con **#gpg --delete-key "Miguel Cordoba"** y se borrará completamente la clave.

TAREA 4. EXPORTAR CLAVE A UN SERVIDOR PÚBLICO DE CLAVES PGP

1. Genera la clave de revocación de tu clave pública para utilizarla en caso de que haya problemas.

Para generar dicha clave he usado el comando #gpg --gen-revoke <UID>
Me ha pedido rellenar una serie de opciones como, el motivo para la revocación, y la confirmación final para crear la clave de revocación.

```
root@MCA:/home/miguel# gpg --gen-revoke BF0B32BB10F98F2E861AF9B7EEDED9FCCE3CFC9A

sec  rsa3072/EEDED9FCCE3CFC9A 2021-11-15 Miguel Cordoba

¿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; acábela con una línea vacía:
> Ejercicio Seguridad
>
Razón para la revocación: No se dio ninguna razón
Ejercicio Seguridad
¿Es correcto? (s/N) s
se fuerza salida con armadura ASCII.
```

2. Exporta tu clave pública al servidor `pgp.rediris.es`.

Para exportar la clave primero hemos mirado cual es su UID y luego con el comando **#gpg--keyserver pgp.rediris.es --send-key <UID>**.

```
root@MCA:/home/miguel# gpg --keyserver pgp.rediris.es --send-key BF0B32BB10F98F2E861AF9
B7EEDED9FCCE3CFC9A
gpg: enviando clave EEDED9FCCE3CFC9A a hkp://pgp.rediris.es
root@MCA:/home/miguel# gpg --delete-key "omar elhani"
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.
```

Search results for 'miguel cordoba'

Type	bits/keyID	Date	User ID
pub	3072R/CE3CFC9A	2021-11-15	Miguel Cordoba
pub	3072R/8C74FBC0	2021-11-11	Miguel Cordoba <miguelcor.rrss@gmail.com>
pub	1024D/3C6326CB	2005-05-18	Miguel Angel Cordoba (Feina) <cordoba@grahi.upc.edu>
pub	1024D/BBB21A20	2005-04-06	Miguel Angel Cordoba (Feina) <cordoba@grahi.upc.edu>
pub	2048R/D0373F8B	2002-03-14	Miguel Araya Cordoba <marayac@bncr.fi.cr>
pub	1024D/20417CC9	2000-07-12	Miguel A. Cordoba <macordoba@usa.net>

3. Borra la clave pública de alguno de tus compañeros de clase e impórtala ahora del servidor público de rediris.

Para borrar la clave de mi compañero usando: **#gpg --delete-key “omar elhani”**, luego he comprobado que se ha borrado usando el comando **#gpg –list-keys**

```
root@MCA:/home/miguel# gpg --delete-key "omar elhani"
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  rsa3072/17EC279EAC46DAA8 2021-11-11 omar elhani <omar.elhani1@gmail.com>

¿Eliminar esta clave del anillo? (s/N) s
root@MCA:/home/miguel# gpg --list-keys
/root/.gnupg/pubring.kbx
------------------------
pub   rsa3072 2021-11-15 [SC] [caduca: 2023-11-15]
      BF0B32BB10F98F2E861AF9B7EEDED9FCCE3CFC9A
uid           [  absoluta ] Miguel Cordoba
sub   rsa3072 2021-11-15 [E] [caduca: 2023-11-15]
```

y por último he importado la clave usando: **#gpg --keyserver pgp.rediris.es --recv-keys <UID de la pagina rediris>**.

```
root@MCA:/home/miguel# gpg --keyserver pgp.rediris.es --recv-keys 644AC899
gpg: clave CA261D60644AC899: clave pública "omar elhani <omar.elhani1@gmail.com>" impor
tada
gpg: Cantidad total procesada: 1
gpg:               importadas: 1
```

NOTA: Los UID de la página rediris.es son los últimos 8 digitos del UID de la clave pública.

TAREA 5. CIFRADO ASIMÉTRICO CON OPENSSL

1. Genera un par de claves (pública y privada).

Para generar la clave privada he usado el comando **#openssl genrsa -aes128 -out miguelito.pem 2084** y para crear la pública he usado **#openssl rsa -in miguelito.pem -pubout miguelito.publico.pem**

```
root@MCA:/home/miguel# openssl rsa -in miguelito.pem -pubout -out miguelito.publico.pem
Enter pass phrase for miguelito.pem:
writing RSA key
```

```
miguel@MCA:~$ ls | egrep miguelito
miguelito.pem
miguelito.publico.pem
```

2. Envía tu clave pública a un compañero.

La clave pública se la he enviado a Omar por email.

3. Utilizando la clave pública cifra un fichero de texto y envíalo a tu compañero.

Primero he creado el fichero **#echo "fichero desencriptado" > fichero.txt**.

y luego lo he cifrado usando **#openssl rsault -encrypt -in fichero.txt -out fichero.enc -inkey miguelito.publico.pem -pubin**.

```
root@MCA:/home/miguel# echo "fichero desencriptado" > fichero.txt
root@MCA:/home/miguel#  openssl rsautl -encrypt -in fichero.txt -out fichero.enc -inkey
 miguelito.publico.pem -pubin
```

4. Tu compañero te ha mandado un fichero cifrado, muestra el proceso para el descifrado.

1. Me he descargado el fichero del correo.
2. Para desencriptar el fichero mandado por mi compañero usaré el comando:
#openssl rsault -decrypt -in secreto.enc -out secreto.txt -inkey key.pem.

```
root@MCA:/home/miguel# openssl rsautl -decrypt -in secreto.enc -out secreto.txt -inkey
key.pem
root@MCA:/home/miguel# cat secreto.txt
Prueba de fichero encriptado
```