SEGURIDAD EN SISTEMAS OPERATIVOS
4º Grado en Informática – Complementos de Ing. del Software
Curso 2017-18

Práctica [1]

Sesión [5]

Autor[1]: Iván Rodríguez Millán

Ejercicio 1.

a) Generamos las claves públicas y privadas con la siguiente orden:

```
[root@ivancito Documentos]# gpg --gen-key
gpg (GnuPG) 1.4.22; Copyright (C) 2015 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.

Seleccione el tipo de clave deseado:
   (1) RSA y RSA (por defecto)
   (2) DSA y ElGamal (por defecto)
   (3) DSA (sólo firmar)
   (4) RSA (sólo firmar)
¿Su elección?1
las claves RSA pueden tener entre 1024 y 4096 bits de longitud.
¿De qué tamaño quiere la clave? (2048)
El tamaño requerido es de 2048 bits
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)?
La clave nunca caduca
¿Es correcto? (s/n) s

Necesita un identificador de usuario para identificar su clave. El programa
construye el identificador de usuario a partir del Nombre Real, Comentario y Dirección
de Correo Electrónico de esta forma:
    "Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Nombre y apellidos: ivan rm
Dirección de correo electrónico: ivanrm@correo.ugr.es
```

1 Como autor declaro que los contenidos del presente documento son originales y elaborados por mi. De no cumplir con este compromiso, soy consciente de que, de acuerdo con la "Normativa de evaluación y de calificaciones de los estudiantes de la Universidad de Granada" esto "conllevará la calificación numérica de cero … independientemente del resto de calificaciones que el estudiante hubiera obtenido ..."

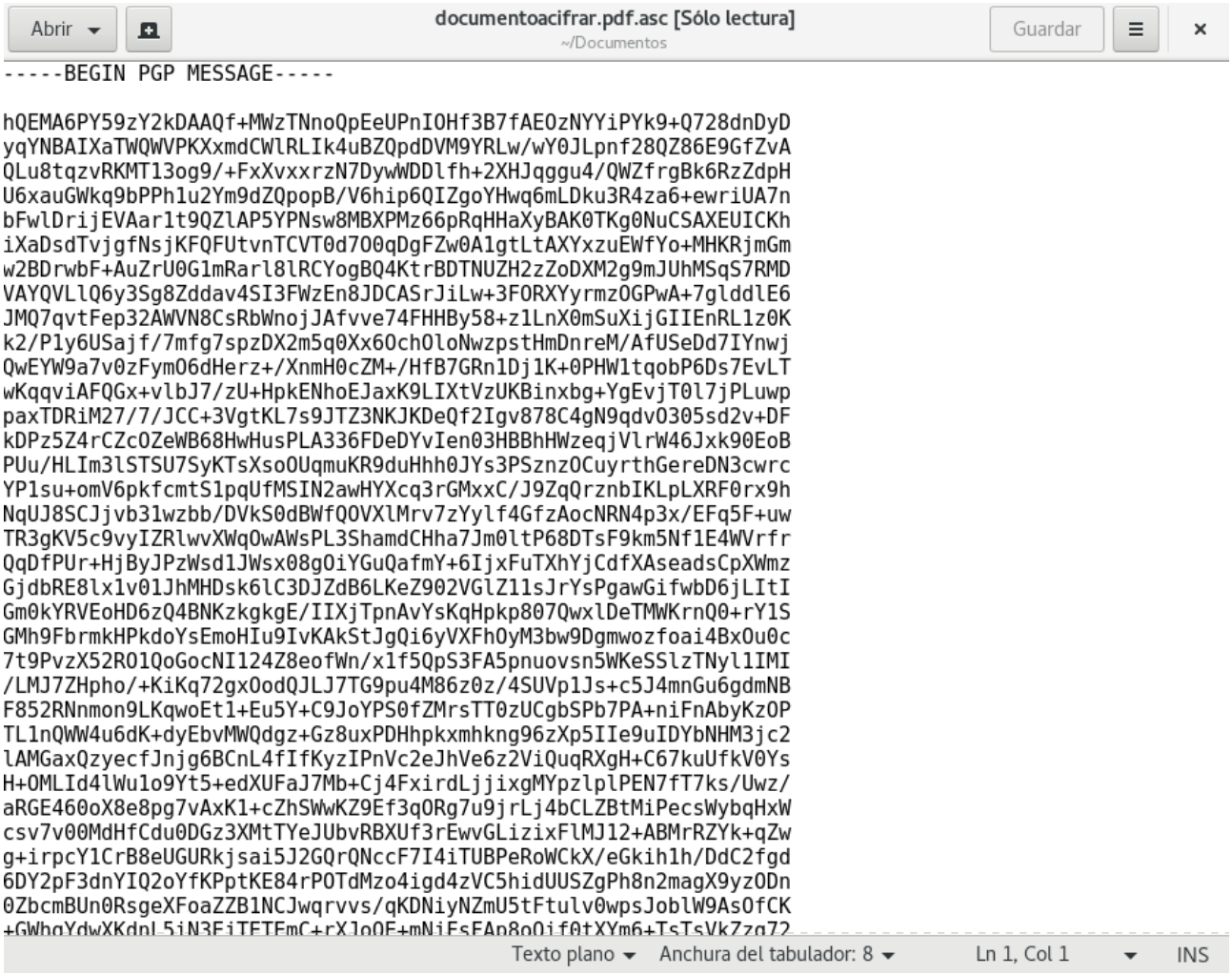
```
Comentario:
Ha seleccionado este identificador de usuario:
    "ivan rm <ivanrm@correo.ugr.es>"

¿Cambia (N)ombre, (C)omentario, (D)irección o (V)ale/(S)alir? V
Necesita una contraseña para proteger su clave secreta.

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: clave EE747C47 marcada como de confianza absoluta
claves pública y secreta creadas y firmadas.

gpg: comprobando base de datos de confianza
gpg: 3 dudosa(s) necesaria(s), 1 completa(s) necesaria(s),
modelo de confianza PGP
gpg: nivel: 0  validez:   1  firmada:   0  confianza: 0-, 0q, 0n, 0m, 0f, 1u
pub   2048R/EE747C47 2017-11-15
      Huella de clave = 3F76 FF84 0155 DC94 49DB  8ED8 F954 AC9B EE74 7C47
uid                  ivan rm <ivanrm@correo.ugr.es>
sub   2048R/D8DA40C0 2017-11-15
```

```
root@ivancito Documentos]# gpg --list-keys
/root/.gnupg/pubring.gpg
------------------------
pub   2048R/EE747C47 2017-11-15
uid                  ivan rm <ivanrm@correo.ugr.es>
sub   2048R/D8DA40C0 2017-11-15
```

b)

```
[root@ivancito Documentos]# gpg --armor --recipient ivanrm@correo.ugr.es --encrypt
documentoacifrar.pdf
```

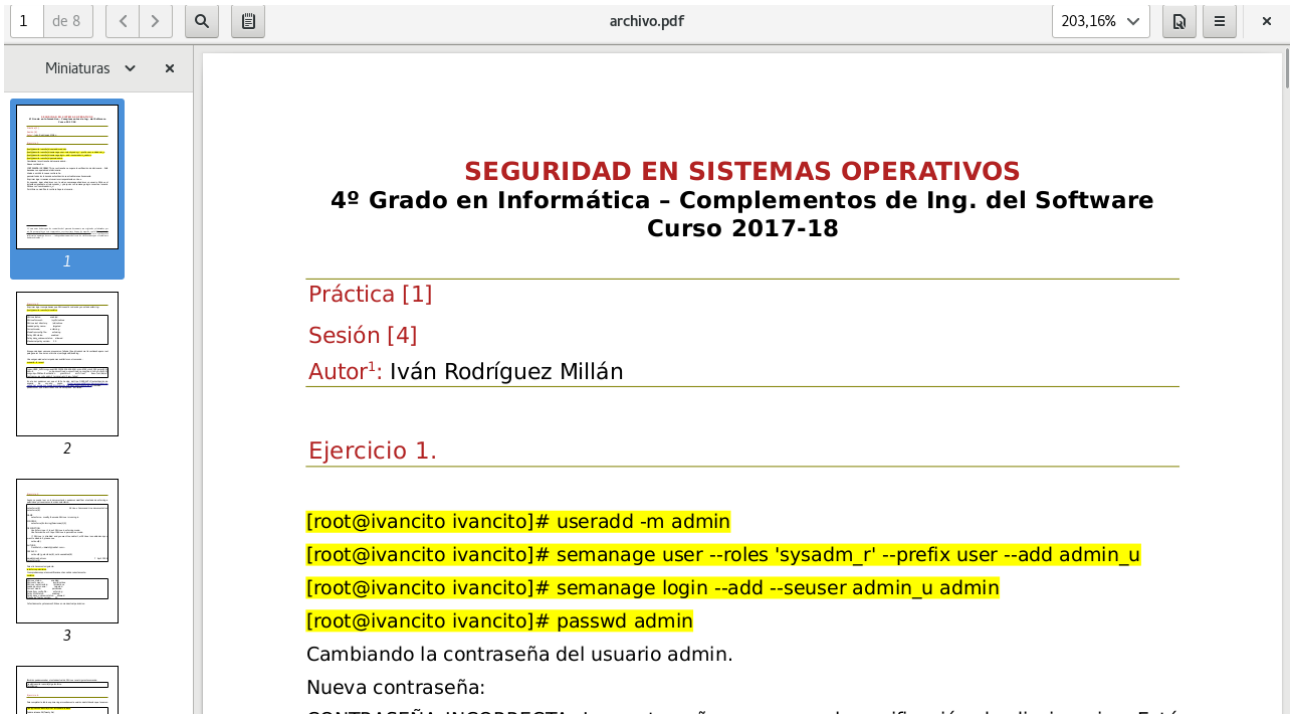
Y obtenemos un documento con extensión .asc que se puede comprobar que está cofrado por PGP:



Ahora para volver a obtener el contenido desencriptado usamos el siguiente comando:

```
[root@ivancito Documentos]# gpg --output archivo.pdf --decrypt documentoacifrar.pdf.asc

Necesita una contraseña para desbloquear la clave secreta
del usuario: "ivan rm <ivanrm@correo.ugr.es>"
clave RSA de 2048 bits, ID D8DA40C0, creada el 2017-11-15(identificador de clave primaria EE747C47)

gpg: cifrado con clave RSA de 2048 bits, ID D8DA40C0, creada el 2017-11-15
      "ivan rm <ivanrm@correo.ugr.es>"
```

Y vemos el resultado:



c)

Para realizar este ejercicio voy a contar con la inestimable ayuda de Diego Iáñez Ávila.

En primer lugar exportamos la clave pública nuestra a un fichero para pasársela a Diego y que el pueda encriptar un fichero con ella, esto se hace con el siguiente comando:

```
gpg –armor –export –output ivanrm-pubkey.gpg ivanrm
```

Una vez exportada la clave pública y pasada a nuestro compañero, recivimos la clave pública de él.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1

mQENBFoPFZMBCAC3E/8UhByY4nVuWVSPJ7gmYEt4cIkm6+eAV0jbnjqETkQPcG5l
BXVbf2VTwzFV3eBRAKylyE9wuJ/4x5gTwlcliSH88+naMvxQi253r/fjuka+R0eE
9LYJr9Oyc4vVCoulX74yDLc6kF649WiyJmxL1/e6FjkXd9hhjAR7sMabY2ZAVnF3
Pvt5/e7mIXuHzNHlFsTeiGCqlsWkIOmfBBZLFIfNcdj9Bd9POlrf4x8zsUNCU62V
aE6dLxCaRLJxskvLt2S1Fw2qSVLMTEVAwO8CStF/FR3LbuGMSXU/8sYizeKvILQu
AA3N04JU159jmniw5v+7sOrYtApK3cn4We3hABEBAAG0NkRpZWdvIEnDocOxZXog
KFByw6FjdGljYXMgU1NPKSA8ZGlhbmF2aUBjb3JyZW8udWdyLmVzPokBOAQTAQIA
IgUCWg8VkwIbAwYLCQgHAwIGFQgCCQoLBBYCAwECHgECF4AACgkQFExj0LYabNtD
ygf8Cfs/OLdwfuhygYwhEy9u68GOR1B9Og/aIROO/plo9pEG60dBCX/T06z/XwAu
KdXFudWly6SNeEyF0jqS6yojfpg0NgGdFm+TRA38ZxpX3ZPk5a9E1olOdw51MWWR
csrhCQUmkqVqXqGA5Q//4tXgubSRAtSPIQ1MemHUBLd928cfhVhUDWUbCXjC3uAh
/PWqMLDPNcg1ixRG1SKVWpbkhxpcBZwWOJArYwKxRL4GLwhjFN5BtrZ9lb0d9MUH
q8x6hq2q88gWYtU8FsWbqBh84P50L9i0uadgpd42zpSm5ws6JL7qELxK+UXHOQOl
o9YlAeFVKpIDVtKEHZ8VocbT17kBDQRaDxWTAQgAnNJo9gpnx5eMkCBC3gZPOLpD
L2pXRIJ1MJrU4Ax4ERDaIPDbnX/itC/1vjrKi4Nwh5ty+VqHawIHp3MrK8Egw/QN
VSkDiGAjm5l8gyujBKHPTh2fMMz7jrVFYp3zIRdNHTRCaWvX+8d+aPXV4CJiOvFv
qr/X04yl6he2JMggS6322wyoCAzSYYFe6ZGCbYL7ZbIvQw3z8qPmxQr1i6/rPyQQ
```

```
38/t7332Xs05ePHa1hYnwreBRXn0iu16fFlSrG28t6gj2yVcCD0uuMJY3vuX90wd
BXYgajiLkWG9OWjfmXufLHomvNmoQtU9r9UUDDj/guNr0kEh2Psv7qrLm5EWGwAR
AQABiQEfBBgBAgAJBQJaDxWTAhsMAAoJEBRMY9C2Gmzber4H/0LSKbdd5m5rkwBZ
QuBTfhwGeHtEIT7bLe+AJMurfm2OjF9ixc/PJwWUpYuWPj4IP0J4V9aSESDIr+Eb
4WPm8JNXAaeV5zOxh8xp4qCZWKt3pAm3/M69XFJFpBehNT5u3v+Mn8XXMm1OX5Oy
5jHwHT/uQf3dwjmFPqYjRkwp+6O4+uEdnvyNlC0bMcFN1kYUQaQOXMGAAh8TpMHK
hRpamtkMJvbk78gxo2VwZGM7QdEztobNpO2xN/whBAoGMMDWPspZvEM0VdiRUIv/
DN6e93Canni73Hllpdlx4dr98qTntVAcEiU4mfK3RG8G/Hy6+CfowqYjwoJsdMBK
7SG39oM=
=80O8
-----END PGP PUBLIC KEY BLOCK-----
```

Y procedemos a importarla:

```
[root@ivancito Documentos]# gpg --import diego-pubkey.gpg
gpg: clave B61A6CDB: clave pública "Diego Iáñez (Prácticas SSO) <dianavi@correo.ugr.es>" importada
gpg: Cantidad total procesada: 1
gpg:               importadas: 1  (RSA: 1)
[root@ivancito Documentos]# gpg --list-keys
/root/.gnupg/pubring.gpg
------------------------
pub   2048R/EE747C47 2017-11-15
uid                  ivan rm <ivanrm@correo.ugr.es>
sub   2048R/D8DA40C0 2017-11-15

pub   2048R/B61A6CDB 2017-11-17
uid                  Diego Iáñez (Prácticas SSO) <dianavi@correo.ugr.es>
sub   2048R/F26BED94 2017-11-17
```

A la vez que se importa el me envía un documento .txt cifrado llamado secreto.txt.asc y yo le envío el siguiente fichero con el contenido siguiente:

```
FicheroIvan.txt
Esto es un ejercicio de SSO



Proceso de encriptación:
[root@ivancito Documentos]# gpg --armor --recipient dianavi@correo.ugr.es --encrypt ficheroIvan.txt
gpg: F26BED94: No hay seguridad de que esta clave pertenezca realmente
al usuario que se nombra

pub          2048R/F26BED94 2017-11-17 Diego Iáñez (Prácticas SSO) <dianavi@correo.ugr.es>
 Huella de clave primaria: 5FE8 AAD8 83D6 AB2E 4EAB  A15D 144C 63D0 B61A 6CDB
      Huella de subclave: 6C6E AE3A 5CC2 F8DD 8878  4D5A 6D02 BFF4 F26B ED94

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
```

Una vez que ambos tenemos los ficheros encriptamos procedemos a desencriptarlos, en mi caso a desencriptar el archivo que Diego encriptó con mi clave pública:

```
[root@ivancito Documentos]# gpg --output ficheroDiego.txt --decrypt secreto.txt.asc

Necesita una contraseña para desbloquear la clave secreta
del usuario: "ivan rm <ivanrm@correo.ugr.es>"
clave RSA de 2048 bits, ID D8DA40C0, creada el 2017-11-15(identificador de clave primaria EE747C47)

gpg: cifrado con clave RSA de 2048 bits, ID D8DA40C0, creada el 2017-11-15
      "ivan rm <ivanrm@correo.ugr.es>"
```

Resultado del fichero una vez desencriptado:

```
Esto es un mensaje supersecreto.
```

Ejercicio 2.

En primer lugar creamos el siguiente fichero:

```
[root@ivancito Documentos]# cat ficheroencriptaropenssl.txt
Esto es una prueba para OpenSSL.
```

Para cifrar un archivo con OpenSSL usamos la siguiente sentencia con el algoritmo **DES** (Data Encryption Standard) utilizando el modo **CBC** (C*pher-Block Chaining*) en donde antes de ser cifrado a cada bloque de texto se le aplica una operación XOR. La clave secreta se calcula a partir de la clave dada, en nuestro caso “123”.

```
openssl enc -des-cbc -in ficheroencriptaropenssl.txt -out ficherocifradosso1.bin

enter des-cbc encryption password:

Verifying - enter des-cbc encryption password:
```

Para descifrar usamos la siguiente sentencia:

```
[root@ivancito Documentos]# openssl enc -des-cbc -d -in ficherocifradosso1.bin
enter des-cbc decryption password:
Esto es una prueba para OpenSSL.
```

Ejercicio 3.

a) Heartbleed es un bug que se aprovecha de una vulnerabilidad de la librería OpenSSL. De tal forma que permite robar información en donde si todo funciona correctamente debería estar protegida por SSL/TLS, por ejemplo aplicaciones como email, mensajería instantanea, VPN, etc.

En el enlace que se deja en la documentación y al que se hace referencia en este documento en la sección de bibliografía, se habla de la realización de pruebas para verificar de lo que es capaz el bug, en donde se habla de que cuando ellos realizaron las pruebas desde el punto de vista del atacante pudieron robar datos tales como la clave secreta para certificados X.509, nombres de usuarios y contraseñas, etc.

b)

Se sabe que las versiones OpenSSL siguientes tienen estos distintos comportamientos:

OpenSSL 1.0.1 through 1.0.1f (inclusive) are vulnerable

OpenSSL 1.0.1g is NOT vulnerable

OpenSSL 1.0.0 branch is NOT vulnerable

OpenSSL 0.9.8 branch is NOT vulnerable

Aunque a la pregunta de si mi sistema lo sufre en este sitio se habla de que los sitios afectados van desde redes sociales, sitios de comercios, sitios de compañías, etc hasta incluso entidades gubernamentales.

Los sistemas vulnerables son:

Debian Wheezy (stable), OpenSSL 1.0.1e-2+deb7u4

Ubuntu 12.04.4 LTS, OpenSSL 1.0.1-4ubuntu5.11

CentOS 6.5, OpenSSL 1.0.1e-15

Fedora 18, OpenSSL 1.0.1e-4

OpenBSD 5.3 (OpenSSL 1.0.1c 10 May 2012) and 5.4 (OpenSSL 1.0.1c 10 May 2012)

FreeBSD 10.0 - OpenSSL 1.0.1e 11 Feb 2013

NetBSD 5.0.2 (OpenSSL 1.0.1e)

OpenSUSE 12.2 (OpenSSL 1.0.1c)

Los sistemas no vulnerables son:

Debian Squeeze (oldstable), OpenSSL 0.9.8o-4squeeze14

SUSE Linux Enterprise Server

FreeBSD 8.4 - OpenSSL 0.9.8y 5 Feb 2013

FreeBSD 9.2 - OpenSSL 0.9.8y 5 Feb 2013

FreeBSD 10.0p1 - OpenSSL 1.0.1g (At 8 Apr 18:27:46 2014 UTC)

FreeBSD Ports - OpenSSL 1.0.1g (At 7 Apr 21:46:40 2014 UTC)

c) Para subsanarlo basta con usar la versión OpenSSL 1.0.1g o superior. En caso de que esto no sea posible se puede recompilar OpenSSL con la opción de compilación -DOPENSSL_NO_HEARTBEATS

Bibliografía

https://www.gnupg.org/gph/es/manual/x129.html

https://es.wikipedia.org/wiki/Data_Encryption_Standard

https://es.wikipedia.org/wiki/Modos_de_operación_de_una_unidad_de_cifrado_por_bloques

http://heartbleed.com/