Actividad 6: Entrega del informe sobre criptografía de clave pública

Seguridad Informática

30/10/2014

Brais López Yáñez

ÍNDICE

1. GENERACIÓN DE CLAVES

EJERCICIO 1.1

Cread una clave privada RSA de 1024 bits, y almacenadla en el fichero privada1.pem.

Realizada con el comando, mostrado en la siguiente captura:

```
computer@Computer:~/Seguridad6$ openssl genrsa -out clave.pem 1024
Generating RSA private key, 1024 bit long modulus
..++++++
.....++++++
e is 65537 (0x10001)
computer@Computer:~/Seguridad6$
```

EJERCICIO 1.2

Mostrad la información de la clave privada obtenida anteriormente. ¿Está la clave almacenada de forma segura en el ficheroprivada1.pem?

Realizada con el comando, mostrado en la siguiente captura:

```
computer@Computer:~/Seguridad6$ openssl genrsa -des3 -out clave_privada.pem 1024
Generating RSA private key, 1024 bit long modulus
...++++++
....++++++
e is 65537 (0x10001)
Enter pass phrase for clave_privada.pem:
Verifying - Enter pass phrase for clave_privada.pem:
3074169032:error:28069065:lib(40):UI_set_result:result too small:ui_lib.c:869:Yo
u must type in 4 to 8191 characters
Enter pass phrase for clave_privada.pem:
Verifying - Enter pass phrase for clave_privada.pem:
computer@Computer:~/Seguridad6$
```

La clave privada generada, no se almacena de forma segura en un archivo de texto plano, además la extensión .pem no usa ningún algoritmo de cifrado simétrico.

Este tipo de archivos, no los podemos ver desde el escritorio, accediendo a él con el ratón. Pero si podremos ver la contraseña haciendo un cat en consola.

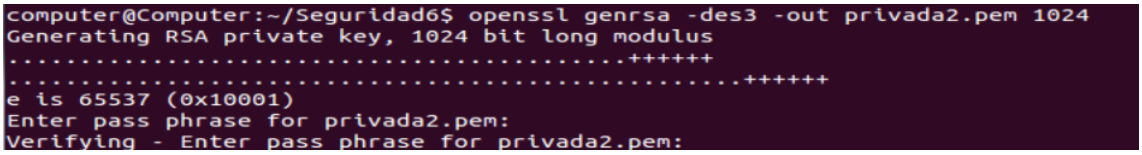
****Acceso a privada1.pem con el ratón*

****Captura de la terminal mostrando el comando cat*

EJERCICIO 1.3

Cread una clave privada privada2.pem de 1024 bits, pero ahora con cifrado del fichero. Recuperad la información de ambas claves:privada1.pem y privada2.pem.

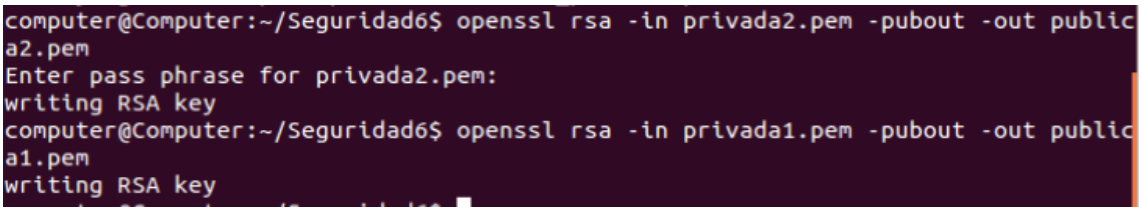
Realizada con el comando, mostrado en la siguiente captura:



EJERCICIO 1.4

Cread las claves públicas correspondientes a las claves privadas privada1.pem y privada2.pem. ¿Es seguro almacenar las claves públicas como texto plano, sin cifrado?

Realizada con el comando, mostrado en la siguiente captura:



Sería el mismo caso que el apartado 1.2, con un algoritmo de cifrado simétrico conseguiríamos proteger la clave pública, el tipo de archivo no sería lo importante, todo dependería de emplear un cifrado simétrico. El tipo de archivo podría guardarse en texto plano, al ser claves públicas, el contenido puede ser visualizado por cualquier persona.

EJERCICIO 1.5

Obtened la información de las dos claves públicas.

Realizada con el comando, mostrado en la siguiente captura:

```
computer@Computer:~/Seguridad6$ openssl rsa -in publica1.pem -pubin
writing RSA key
-----BEGIN PUBLIC KEY-----
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDOzCFom8mK5N+qes/xyRMD/syz
BfYalv9AhglXoRxlxwvLfXJw3w8FYGr9KJzDD1LDXUx3mG22YY9IuRgHguVDqRvP
v3lA0GfkPBur3o1x25hRzwlx7b7qrCU+GseU11jFZpurp0asFMWHKhqp4YxheY8G
Dp44oedXPPmmMhP6iwIDAQAB
-----END PUBLIC KEY-----
computer@Computer:~/Seguridad6$ openssl rsa -in publica2.pem -pubin
writing RSA key
-----BEGIN PUBLIC KEY-----
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCbivcKW7FEZ/xE4cNByoLtjUps
uZ5c20eo15dy1tTke/Av2s87ATDCXapLGRq9idd2R+VG9YT7KfgiAl8VtOLM6s7p
Omgm9lwglzNM6Qlsjt+9EVSyzqNwO3xfSIHE1CSTXh9gCog9mb7mCS+VdNxbowuJ
eknZGtkN2UIwNGx2mQIDAQAB
-----END PUBLIC KEY-----
computer@Computer:~/Seguridad6$
```

2. CIFRADO Y DESCIFRADO

EJERCICIO 2.1

Cifrad y descifrad un texto pequeño.

Realizada con el comando, mostrado en la siguiente captura:

```
computer@Computer:~/Seguridad6$ openssl rsautl -encrypt -in textoplano -inkey pu
blica2.pem -pubin -out textocifrado
computer@Computer:~/Seguridad6$
```

```
computer@Computer:~/Seguridad6$ openssl rsautl -decrypt -in textocifrado -inkey
privada2.pem -out textodescifrado
Enter pass phrase for privada2.pem:
computer@Computer:~/Seguridad6$
```

EJERCICIO 2.2

Tratad de cifrar un fichero más grande. ¿Qué ocurre? ¿Por qué?

Realizada con el comando, mostrado en la siguiente captura:

```
computer@Computer:~/Seguridad6$ openssl rsautl -encrypt -in textoplanoLargo -ink
ey publica2.pem -pubin -out textoplanoLargoCIF
RSA operation error
3074025672:error:0406D06E:rsa routines:RSA_padding_add_PKCS1_type_2:data too lar
ge for key size:rsa_pk1.c:151:
```

Lo que ocurre un error, debido a que la clave pública ha sido obtenida a través de una clave privada de 1024 bits. Por lo tanto, el archivo es demasiado grande para esa clave pública. La solución sería ampliar la clave o disminuir la longitud del texto.

EJERCICIO 2.3

Pedid a un compañero su clave pública. Cifrad un mensaje corto con su clave pública. Pasadle el mensaje cifrado, y pedidle que lo descifre.

Esta práctica, al igual que la anterior, se hará con el compañero Manuel Cascallar Autrán.

Le envíamos la clave pública a Manuel y procedemos a cifrad el mensaje con su clave:

```
computer@Computer:~/Seguridad6/Manu$ openssl rsautl -encrypt -in textoplano -ink
ey publica2 -pubin -out textocifrado
computer@Computer:~/Seguridad6/Manu$
```

Tras cifrar el mensaje, se lo enviamos para que lo descifre, el nos envía otro. Procedemos a descifrarlo con el siguiente comando:

```
computer@Computer:~/Seguridad6/Manu$ openssl rsautl -decrypt -in textocifrado -i
nkey privada2.pem -out textodescifradoManu
Enter pass phrase for privada2.pem:
computer@Computer:~/Seguridad6/Manu$
```

```
computer@Computer:~/Seguridad6/Manu$ cat textodescifradoManu
Para descifrar, se cambia la opción a_ -decrypt.
```

Ejercicio 2.4: Cifrad un mensaje largo con un algoritmo de cifrado simétrico (por ejemplo AES), usando una contraseña aleatoria. Cifrad la contraseña con la clave pública de un compañero. Enviadle al compañero la contraseña cifrada, y el mensaje cifrado. Pedidle que descifre el mensaje.

Comando de cifrado con contraseña aleatoria (1234)

```
computer@Computer:~/Seguridad6/Manu/AES$ openssl aes-128-cbc  -p -in textoplanoL
argo -out textocifradoLargo
enter aes-128-cbc encryption password:
Verifying - enter aes-128-cbc encryption password:
salt=566AEBCA2F9F2B65
key=D9998C8C23FD2149B7B08660EA3AA77F
iv =0B7969A8C3A4C70B2FC6EF6E5280B421
computer@Computer:~/Seguridad6/Manu/AES$
```

Cifrado de la contraseña con la pública de Manuel

```
computer@Computer:~/Seguridad6/Manu/AES$ openssl rsautl -encrypt -in pass -inkey
 publica2 -pubin -out passCifrada
computer@Computer:~/Seguridad6/Manu/AES$
```

Descifrado del mensaje enviado por Manuel

```
computer@Computer:~/Seguridad6/Manu/AES$ openssl aes-128-cbc -d -in cifrad -out
descifrado
enter aes-128-cbc decryption password:
computer@Computer:~/Seguridad6/Manu/AES$
```

Texto descifrado

```
computer@Computer:~/Seguridad6/Manu/AES$ cat descifrado
Para descifrar, se cambia la opción a  -decrypt.

Ejercicio 2.1: Cifrad y descifrad un texto pequeño.

Ejercicio 2.2: Tratad de cifrar un fichero más grande. ¿Qué ocurre? ¿Por qué?

Ejercicio 2.3: Pedid a un compañero su clave pública. Cifrad un mensaje corto co
n su clave pública. Pasadle el mensaje cifrado, y pedidle que lo descifre.

Ejercicio 2.4: Cifrad un mensaje largo con un algoritmo de cifrado simétrico (po
r ejemplo AES),  usando una contraseña aleatoria. Cifrad la contraseña con la cl
ave pública de un compañero. Enviadle al compañero la contraseña cifrada, y el m
ensaje cifrado. Pedidle que descifre el mensaje.
```

3. FIRMAS

EJERCICIO 3.1

¿Debemos usar la clave pública o la clave privada para la firma?

Realizada con los comandos, mostrados en las siguientes capturas:

```
computer@Computer:~/Seguridad6/Manu/Firmas$ openssl dgst -md5 -out resumen texto
plano
computer@Computer:~/Seguridad6/Manu/Firmas$
```

```
computer@Computer:~/Seguridad6/Manu/Firmas$ openssl rsautl -sign -in resumen -in
key privada2.pem -out firma
Enter pass phrase for privada2.pem:
computer@Computer:~/Seguridad6/Manu/Firmas$
```

La clave privada se usa para cifrar, al contrario que la privada, la clave pública se utiliza para descifrar.

EJERCICIO 3.2

Firmad un mensaje. A continuación, verificad la firma comparando el resumen original y el resumen descifrado. Podéis usar la herramienta diff.

Continuando lo realizado en el ejercicio 3.1, procedemos a verificar la firma y a usar el comando diff.

Realizada con los comandos, mostrados en la siguientes capturas:

Verificación de la firma:

```
computer@Computer:~/Seguridad6/Manu/Firmas$ openssl rsautl -verify -in firma -pu
bin -inkey publica2.pem -out resumen2
computer@Computer:~/Seguridad6/Manu/Firmas$
```

Comparación de los archivos, usando diff:

```
computer@Computer:~/Seguridad6/Manu/Firmas$ diff -s resumen resumen2
Los archivos resumen y resumen2 son idénticos
computer@Computer:~/Seguridad6/Manu/Firmas$
```

EJERCICIO 3.3

Cread tres mensajes. Firmad los tres. Modificad ligeramente uno o dos de los mensajes, después de haberlos firmado, y enviádselos a un compañero, junto con las firmas obtenidas anteriormente. Pedidle que averigüe qué mensajes fueron alterados.

Realizada con los comandos, mostrados en la siguientes capturas:

Creación de los resúmenes de los 3 mensajes:

```
computer@Computer:~/Seguridad6/Manu/Firmas$ openssl dgst -md5 -out resumenM1 m1
computer@Computer:~/Seguridad6/Manu/Firmas$ openssl dgst -md5 -out resumenM2 m2
computer@Computer:~/Seguridad6/Manu/Firmas$ openssl dgst -md5 -out resumenM3 m3
```

Creación de las 3 firmas:

```
computer@Computer:~/Seguridad6/Manu/Firmas$ openssl rsautl -sign -in resumenM1 -
inkey privada2.pem -out firmaM1
Enter pass phrase for privada2.pem:
computer@Computer:~/Seguridad6/Manu/Firmas$ openssl rsautl -sign -in resumenM2 -
inkey privada2.pem -out firmaM2
Enter pass phrase for privada2.pem:
computer@Computer:~/Seguridad6/Manu/Firmas$ openssl rsautl -sign -in resumenM3 -
inkey privada2.pem -out firmaM3
Enter pass phrase for privada2.pem:
```

Tras la ejecución de los comandos indicados y la modificación de 2 de los archivos, le enviamos los documentos necesarios (firmas y mensajes) a Manuel.

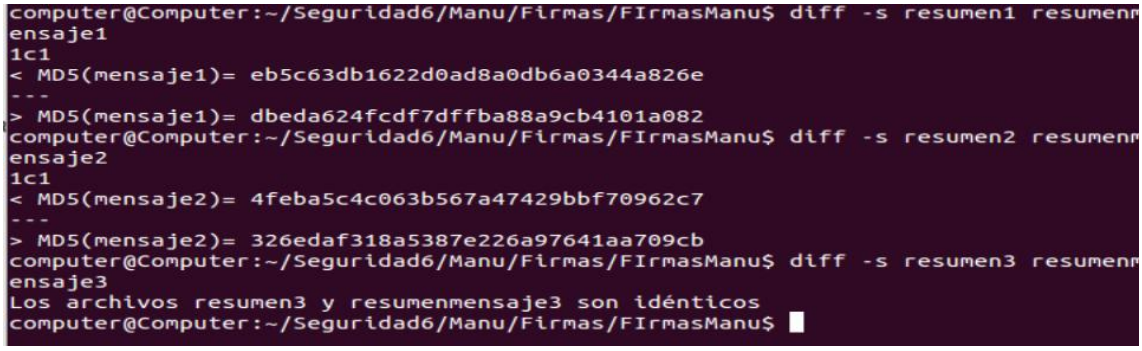
Creación de los resúmenes, con los mensajes de Manuel:

```
computer@Computer:~/Seguridad6/Manu/Firmas/FIrmasManu$ openssl dgst -md5 -out re
sumenmensaje1 mensaje1
computer@Computer:~/Seguridad6/Manu/Firmas/FIrmasManu$ openssl dgst -md5 -out re
sumenmensaje2 mensaje2
computer@Computer:~/Seguridad6/Manu/Firmas/FIrmasManu$ openssl dgst -md5 -out re
sumenmensaje3 mensaje3
computer@Computer:~/Seguridad6/Manu/Firmas/FIrmasManu$
```

Obtención de los nuevos resúmenes a través de las firmas, descifrado.

```
computer@Computer:~/Seguridad6/Manu/Firmas/FIrmasManu$ openssl rsautl -verify -i
n firma1 -pubin -inkey publica2 -out resumen1
computer@Computer:~/Seguridad6/Manu/Firmas/FIrmasManu$ openssl rsautl -verify -i
n firma2 -pubin -inkey publica2 -out resumen2
computer@Computer:~/Seguridad6/Manu/Firmas/FIrmasManu$ openssl rsautl -verify -i
n firma3 -pubin -inkey publica2 -out resumen3
computer@Computer:~/Seguridad6/Manu/Firmas/FIrmasManu$
```

Comparación de los mensajes, con diff:



Como podemos observar en esta última captura, los mensajes que han sido modificados han sido el 1 y el 2, siendo el último el que permanece sin alterar.

EJERCICIO 3.4

¿Cómo mejoraríais el proceso en el ejercicio 2.4, para incluir la firma del emisor en un mensaje cifrado, de forma que el receptor pueda verificar la autenticidad del mensaje?

Para mejorar el proceso del ejercicio 2.4, para ser más seguro se podría incluir la utilización de una firma. Esto sería un punto favorable para la verificación del mensaje y la comprobación de si ha sido alterado.

Todo esto, se haría siguiendo los pasos del apartado de firmas, en concreto el ejercicio 3.3.