Universidad Politécnica de Cartagena

Escuela Técnica Superior de Ingeniería de Telecomunicación

SEGURIDAD EN REDES

Práctica 3: Usos cotidianos de la criptografía (I):

Transferencia segura de archivos
Sistemas criptográficos simétricos
Sistemas criptográficos asimétricos
Conexión remota segura

Autores: Josemaría Malgosa Sanahuja
María Dolores Cano Baños

SEGURIDAD EN REDES

Usos cotidianos de la criptografía PRÁCTICA 3 (primera parte)

En esta práctica se experimentará con distintas técnicas criptográficas que se utilizan de forma cotidiana. En concreto se estudiará:

- La transferencia segura de archivos entre máquinas remotas (*scp*)
- Cifrar y descifrar con sistemas simétricos
- Cifrar y descifrar con sistemas asimétricos
- La conexión remota segura (*ssh*)
- Técnicas esteganográficas
- La verificación de un hash (*openssl*)
- La generación de contraseñas seguras
- El acceso a portales a través de certificados digitales
- El escaneo de puertos TCP/UDP abiertos
- La firma electrónica
- El envío de correos electrónicos firmados y encriptados mediante el estándar openPGP

IMPORTANTE: En algunos apartados se necesita trabajar de forma remota con otro ordenador. Para facilitar la realización de la práctica, dicho ordenador será el servidor del laboratorio (labit201.upct.es). El *login* del usuario remoto es serXX (siendo XX un número que el profesor deberá asignaros) y la contraseña es *la seguridad no es importante*.

TRANSFERENCIA SEGURA DE ARCHIVOS ENTRE MÁQUINAS REMOTAS

El comando *scp* permite transferir un archivo entre dos terminales remotos de forma segura. El comando *scp* utiliza los servicios de *ssh* para realizar su propósito, por lo que -en términos generales- todo lo que se diga (más adelante) acerca de *ssh* puede aplicarse también en *scp*.

La primera vez que te conectas a una máquina remota mediante *scp* o *ssh* aparecerá un mensaje parecido a éste advirtiéndonos de un posible ataque de suplantación de identidad (*spoofing*):

```
The authenticity of host '192.168.56.10 (192.168.56.10)' can't be established.
RSA key fingerprint is SHA256:qCflpywsqSxQYVIkID10feDBzO18ZCnyq/lkQKZT5Ps.
Are you sure you want to continue connecting (yes/no)?
```

En principio, debe responderse que sí, salvo en aquellas situaciones en que se tenga realmente duda de que el servidor no sea realmente quien dice ser.

1. Copiar el fichero *logname* ubicado en el directorio */usr/bin* del puesto de trabajo al directorio *~/pr3* del usuario *serXX* en la máquina remota (la contraseña es la seguridad no es importante)
   scp /usr/bin/logname ser70@labit201.upct.es:~/pr3
2. ¿Cómo debería escribirse el comando anterior si el nombre de los usuarios en ambas máquinas fueran idénticos?

3. ¿El archivo se transmite en claro o cifrado?
   Cifrado

CONEXIÓN REMOTA SEGURA (parte I)

El comando *ssh* permite tanto ejecutar comandos en una máquina remota como conectarse directamente a ella. *ssh* se encarga de cifrar toda la comunicación mediante un sistema simétrico. En la versión 1 del protocolo *ssh*, la clave simétrica se genera al azar y se transmite utilizando cifrado asimétrico RSA. En la versión 2 la clave simétrica se intercambia por el método DH.

4. Ejecutar el comando *ssh serXX@labit201.upct.es ls –l*. El comando *ls* ¿se ha ejecutado en local o en remoto? remoto

5. Ejecutar el comando *ssh serXX@labit201.upct.es* y autenticarse introduciendo la contraseña correcta ¿En qué ordenador se está en estos momentos trabajando? ¿Se pueden utilizar los comandos *ls*, *cp*, *mkdir*? ¿y el comando *xclock*? Salir del ordenador remoto (*exit* o bien ctrl-D) y volverse a conectar con *ssh* –X *serXX@labit201.upct.es* ¿se puede ejecutar ahora el comando *xclock*? Solo se puede ejecutar xclock con ssh -X

CIFRAR Y DESCIFRAR CON SISTEMAS SIMÉTRICOS

Crear el directorio *~/pr3* y ubicarse en él. En el portal de la asignatura encontrará el archivo secreto *TheDesignOfRijndael.pdf.des* cifrado con el método DES.

6. Descargar el archivo y descifrarlo utilizando el comando *openssl* (la contraseña es la de la cuenta Linux) El archivo cifrado, ¿ocupa aproximadamente el mismo tamaño que el archivo original? openssl des -d -in TheDesign.pdf.des -out TheDesign.pdf Contraseña: IceldlcL

7. Cifrar con el método AES-128 el archivo *TheDesignOfRijndael.pdf* ¿ocupa aproximadamente el mismo tamaño que la versión DES? ¿El archivo cifrado es binario o de texto?
   openssl aes-128-cbc -e -in TheDesign.pdf -out TheDesign.pdf.aes128
8. Volver a cifrar con el método AES-128 el archivo *TheDesignOfRijndael.pdf* pero forzando que el formato del archivo cifrado sea de texto (base64) ¿Qué tamaño tiene el fichero cifrado? ¿Qué ventaja tiene utilizar el formato de texto frente al binario? openssl aes-128-cbc -e -base64 .....

9. ¿Aproximadamente, cuántos métodos de cifrado simétrico admite *openssl*? 100

CIFRAR Y DESCIFRAR CON SISTEMAS ASIMÉTRICOS

Antes de utilizar un sistema asimétrico, es necesario que el usuario disponga de un par de claves: la pública y la privada. La clave pública debe publicitarse en algún repositorio electrónico para que cualquier persona pueda disponer a ella. Por el contrario, la clave privada debe permanecer siempre bajo el más estricto secreto.

10. Utilizando *openssl*, generar una clave privada RSA (*RSApriv.pem*) ¿Está codificada en binario (DER) o en ASCII (PEM)? ¿Cómo generaría la clave en formato binario? Está codificado en ASCII
    openssl genrsa -out RSApriv.pem 2048
11. ¿Cómo podemos averiguar es el valor de los exponentes, números primos y el módulo que definen la clave privada RSA? openssl pkey -in RSApriv.pem -text -noout

12. Utilizando *openssl*, generar la clave pública RSA (*RSApub.pem*) asociada con la clave privada anterior. ¿Qué información contiene la clave pública RSA? openssl rsa -in RSApriv.pem -pubout > RSApub.pem

13. Descargar del portal de la asignatura el archivo *rfc4251.txt*. Utilizando *openssl*, cifrar el archivo *rfc4251.txt* por el método RSA (*rfc4251.txt.rsa*) de tal forma que pueda ser descifrado por el usuario remoto (su clave pública SER_pubRSA.pem está disponible en el portal) ¿Qué mensaje de error recibe de *openssl*?

```
alumno@localhost:~/pr3> openssl pkeyutl -encrypt -in rfc4251.txt -out rfc4251.txt.rsa -inkey SER_pubRSA.pem -pubin
RSA operation error
139984833171904:error:0406706C:rsa routines:rsa_ossl_public_decrypt:data greater than mod len:crypto/rsa/rsa_ossl.c:619:
```

14. Reducir el tamaño del fichero *rfc4251.txt* hasta que finalmente pueda ser cifrado por el método RSA. Relacionar el tamaño del fichero (en bits) con el tamaño de la clave RSA
    El archivo no puede ser más grande que la clave En este caso podemos cifrar 117 bytes con los 272 de la clave
15. Transferir el fichero *rfc4251.txt.rsa* al usuario remoto y verificar que éste puede descifrar el archivo utilizando su clave privada. Verificar también que el fichero no puede ser descifrado por ninguna otra persona

```
ser70@labit201:~/pr3> openssl pkeyutl -decrypt -in rfc4251.txt.rsa -out rfc4251.txt -inkey SER_privRSA.pem
```

Atendiendo al resultado anterior, se deduce que:

- Con algoritmos simétricos cualquier persona que sepa la clave puede descifrar el archivo mientras que con el asimétrico sólo lo puede descifrar el poseedor la clave privada asociada con la clave pública utilizada en le proceso de cifrado
- Para cifrar grandes cantidades de información es necesario utilizar un algoritmo simétrico (DES, 3DES, AES, etc.). Los algoritmos simétricos son extremadamente robustos frente a cualquier tipo de ataque y además, el coste computacional asociado a las operaciones de cifrado y descifrado es muy bajo. El único problema que presentan es la necesidad de intercambiar la clave

16. Describir sucintamente- pero con detalle- los mecanismos para el intercambio de claves Diffie-Hellman (DH) y Rivest-Shamir-Adleman (RSA)

17. Cambiar el nombre del archivo *TheDesignOfRijndael.pdf* por ser_*Rijndael.pdf*. Utilizando *openssl*, cifrar el archivo ser_*Rijndael.pdf* por el método triple-des con una contraseña inventada (llamarlo vm_*Rijndael.pdf.des3*). Guardar la contraseña seleccionada en el archivo *clave_des3.txt*. Utilizando *openssl*, cifrar el archivo *clave_des3.txt* por el método RSA de tal forma que pueda ser descifrado solo por el usuario remoto (llamarlo *clave_des3.txt.rsa*). Transferir ambos archivos al ordenador del usuario remoto y que éste se encargue de restaurar el archivo original

```
alumno@localhost:~/pr3> openssl enc -des3 -e -md md5 -in ser_Rijndael.pdf -out ser_Rijndael.pdf.des3
```

18. Si quisiera que el archivo pudiera ser descifrado por otra persona ¿qué es lo único que debería modificar del proceso anterior? Solo tendríamos que cifrar el archivo de la clave no el archivo que hemos cifrado con la clave simetrica

19. A partir de estos resultados ¿cree que es posible intercambiar grandes cantidades de información de forma segura entre cualquier par de personas/máquinas?

    Con una clave simétrica sí

CONEXIÓN REMOTA SEGURA (parte II)

20. Estudiar el proceso de intercambio de información utilizado por *ssh* (anexo I y II). ¿Qué algoritmos simétricos admite *ssh*? ¿Cuál es el algoritmo simétrico utilizado por defecto?

    DES, AES, IDEA, Blowfish.

21. Partiendo de la base de que RSA y DH son ambos igualmente robustos frente a un posible ataque criptográfico, explicar razonadamente porqué es preferible utilizar el método DH frente al RSA para el intercambio de la clave simétrica

Además de transmitir de forma segura, otra de las funciones importantes de *ssh* es la autenticación tanto del servidor y como del cliente (usuario). La autenticación del servidor se realiza mediante *fingerprint* de la clave pública del servidor. La autenticación del usuario puede hacerse por contraseña o por clave pública RSA.

22. En los sistemas de clave pública/privada (asimétricos) ¿cuál de las dos claves se utiliza para identificar (autenticar) a un usuario? La clave pública

23. ¿En qué consiste un *fingerprint*? Verificar el *fingerprint* del ordenador remoto copiando primero en el directorio *~/pr3* su clave pública (scp labit201.upct.es*:/etc/ssh/ssh_host_ecdsa_key.pub ~/pr3*) y ejecutando después el comando `ssh-keygen -lf ~/pr3/ssh_host_ecdsa_key.pub`

```
alumno@localhost:~/pr3> ssh-keygen -lf ssh_host_ecdsa_key.pub
256 SHA256:s9ikruF8Lom3NrEKwAzY6C5MjRIk0yl2EONDr7t1eYs root@linux-45do (ECDSA)
```

24. ¿Por qué se muestra un *fingerprint* y no directamente la clave pública del servidor?

25. Dirigirse al directorio *~/.ssh* ¿Qué información contiene el archivo *known_hosts*? Explicar brevemente- pero con detalle- la función de dicho archivo

    Se ven los host a los que nos hemos conectado

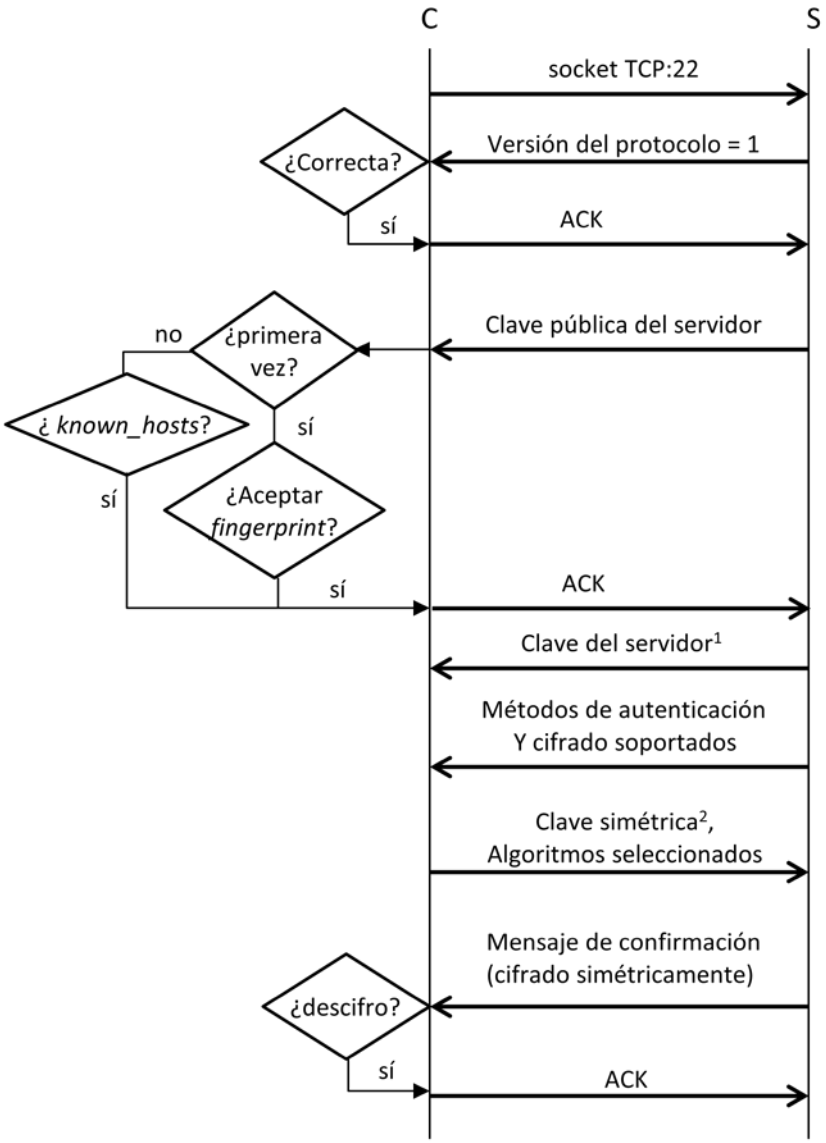
26. ¿Qué tipo de autenticación de usuario utiliza *ssh* por defecto? ¿La contraseña se transmitirá en claro o cifrada? ¿Con un algoritmo simétrico o asimétrico? Cifrada con un algoritmo asimetrico

23. Una huella digital de clave pública, en la criptografía de clave pública, es una secuencia corta de bytes utilizada para identificar una clave pública más larga. Las huellas digitales se crean al aplicar una función de cifrado hash a una clave pública.

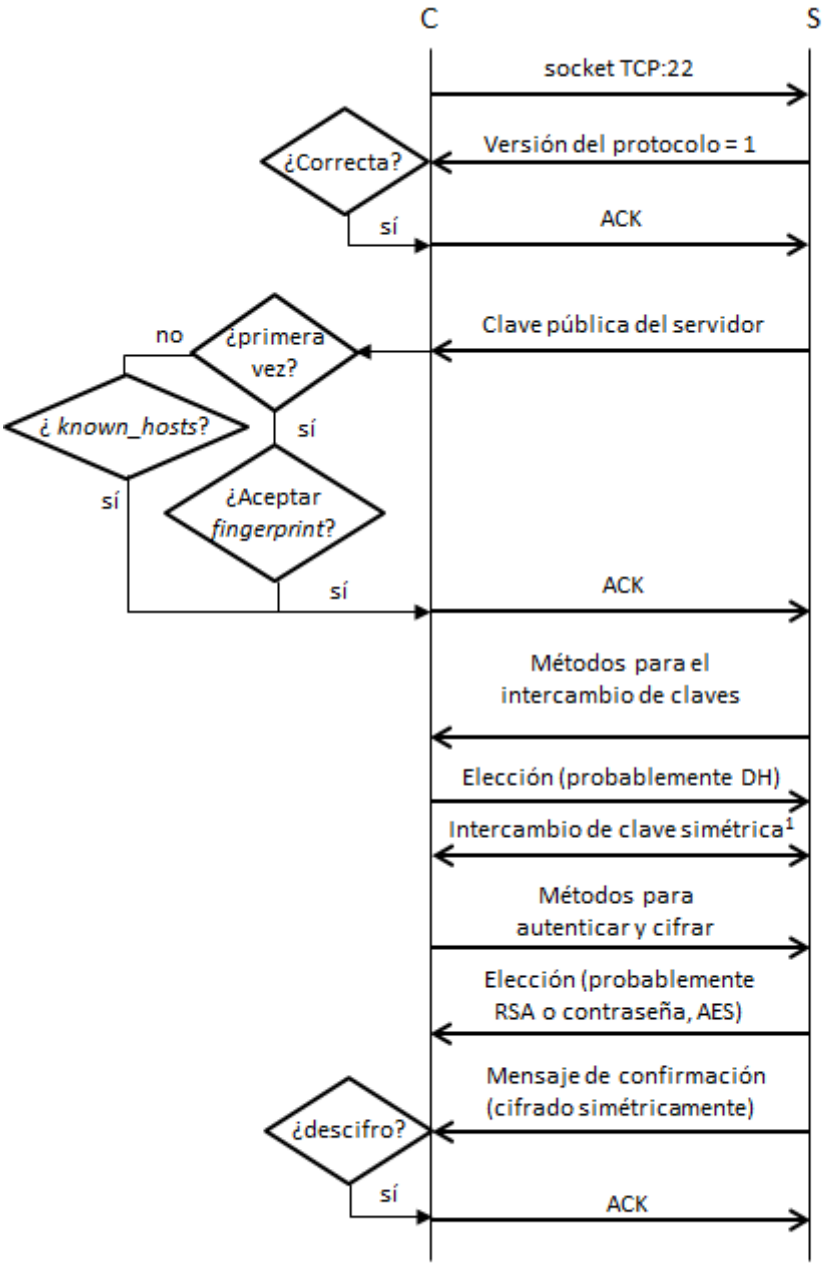
27. Existe otro método más cómodo para autenticar usuarios de una sesión *ssh*. Diríjase al directorio *~/.ssh* y ejecute el comando *ssh-keygen* (dejar sin respuesta todas las preguntas que hace el programa). ¿Qué ficheros ha generado?

28. Copiar la clave pública *ssh* al directorio *~/.ssh/* del usuario remoto con el nombre *authorized_keys*. Conéctese de nuevo al ordenador remoto (*ssh* serXX@labit201.upct.es) ¿Pide contraseña? ¿Cómo ha autenticado en este caso *ssh* al usuario?

29. Borrar el fichero *authorized_keys* del ordenador remoto. En el puesto de trabajo ejecutar el comando `ssh-copy-id serXX@labit201.upct.es` (pedirá la contraseña de la máquina remota). Ahora conéctese al ordenador remoto mediante *ssh* ¿pide contraseña? ¿qué ha hecho el comando *ssh-copy-id*?



[1]Cifrada con la clave privada del servidor. Se cambia cada hora.
[2]Cifrada primero con la clave pública del servidor y después con la clave del servidor.



[1]Cambia cada hora o después de haber transferido 1Gbyte