Práctica 1: Cifrado

Descripción de la práctica

Objetivos

- Ser capaces de usar una herramienta como openSSL para cifrar/descifrar mensajes
- Observar las diferencias en el uso de distintos algoritmos de cifrado
- Experimentar con distintos métodos de intercambio de claves simétricas
- Ser capaz de utilizar la herramienta gnuPG para cifrar/descifrar mensajes

Entorno de realización

- Sistemas operativos Linux y Windows
- Openssl
- Gnupg

La práctica se realizará tanto en Windows como en Linux (equipos del laboratorio o servidor nogal).

En Linux la herramienta openSSL está instalada de manera predeterminada. Para instalar OpenSSL en Windows se pueden descargar el código binario de **http://www.openssl.org** o desde **Herramienta de cifrado** en Diaweb. Se puede obtener documentación online y el manual de la herramienta en **http://www.openssl.org/docs/apps/openssl.html**, **https://www.openssl.org/docs/man1.0.2/man1/openssl-enc.html** y **https://www.openssl.org/docs/man1.1.1/man1/**

openSSL es una herramienta muy potente que se maneja por línea de órdenes. El formato de una orden es:

openssl *command* [*command_opts*] [*command_args*]

donde:

command es el comando a ejecutar

command_opts representa las opciones del comando

command_args identifica los argumentos que necesite ese comando

gnuPG puede descargarse de **http://www.gnupg.org**

Todas las órdenes tienen la estructura '**gpg** --*opcion* *lista_parametros*'

donde:

opcion indica la acción a realizar

lista_parametros identifica los argumentos que necesita esta orden o acción.

Parte A: Cifrado simétrico con OpenSSL

Se usará la orden *enc* que sirve para cifrar/descifrar mensajes usando alguno de los algoritmos soportados. Para comprobar la colección de algoritmos soportados usaremos la orden *ciphers*.

A1.- Cifrar un mensaje de prueba

- Crear un archivo de texto con un editor (elegir el editor dependiendo del entorno de trabajo).
- Cifrar el fichero utilizando un algoritmo simétrico[1] [1]. Dejar el resultado en otro fichero.
- Realizar la transformación inversa (descifrar) a la realizada en el paso anterior.
- Comprobar que el mensaje obtenido en el proceso de descifrado es idéntico al original.
- Repetir con variantes para diferentes algoritmos, claves, longitudes de mensajes, modos de operacion y formas de introducir la clave de cifrado. contestar de forma razonada a las preguntas:
    - ¿El mensaje cifrado es de mayor o menor tamaño que el mensaje original?
    - ¿El tamaño depende del algoritmo de cifrado utilizado, del tamaño del mensaje original o de ambos?
- Probar y documentar las opciones de cifrado:
- `-in filename`
- `-out filename`
- `-pass arg`
- `-salt`
- `-nosalt`
- `-e`
- `-d`
- `-a`
- `-base64`
- `-k password`
  `-kfile filename`

- Probar con al menos 3 algoritmos de cifrado y con los 4 modos de operación básicos con cada uno de ellos.

A2.- Determinar qué algoritmo de cifrado se utiliza en las conexiones HTTPs

- Acceder a una página alojada en un servidor seguro (protocolo https) p.e. **https://www.usal.es**

    - Ver qué algoritmo simétrico se utiliza para conexión según el navegador usado (IE, Firefox, Chrome)
- Probar con al menos otra páginas seguras (www.google.com)

---

[1] [1] Usar la ayuda de openSSL para ver los algoritmos soportados, ordenes y opciones de cifrado y descifrado simétrico.

Parte B: Cifrado asimétrico

B1.- Cifrar un mensaje usando gnuPG

Con gnuPG, trabajando por parejas intercambiar ficheros cifrados usando un mecanismo de clave pública. Probar con mensajes de distintos tamaños, inferiores y superiores al tamaño de la clave que se esté usando.

- Usar el algoritmo (y por tanto claves) RSA para cifra y firma
- Usar el algoritmo El Gamal para cifrar y DSA para firmar

B2.- Cifrar un mensaje usando openSSL

Trabajando por parejas intercambiar ficheros cifrados usando un mecanismo de clave pública con algoritmo RSA. pero usando la suite openSSL. Hay que tener en cuenta que el manejo de las claves pública y privada se hace directamente a través de ficheros y no de anillos de claves.

Parte C: Generar firma de mensajes

C1.- Generar una firma (separada) de un mensaje usando gnuPG

Con las parejas de claves generadas en el apartado B1 se trata de **(a)** generar una firma digital (solo la firma) de un mensaje. La firma tiene que estar en ASCII. Asimismo, y una vez obtenida, **(b)** se validará dicha firma.

C2.- Generar una firma del compendio de un mensaje usando openSSL

Con las parejas de claves generadas en el apartado B2 se trata de **(a)** generar una firma digital (solo la firma) de un mensaje. Para crear la firma del mensaje, previamente se obtendrá una huella (compendio) SHA256 del mensaje. La firma tiene que estar en ASCII. Asimismo, y una vez obtenida, **(b)** se validará dicha firma.

Se entregará un informe en formato PDF que documente la actividad realizada para las tareas encomendadas. Los lotes de órdenes que se ejecuten desde consola se guardarán en scripts (.bat para windows y .sh para linux) que serán entregados junto con el informe. Se entregarán también los ficheros utilizados durante el desarrollo de esta práctica.