Desarrollo de una aplicación que utilice criptografía

Objetivo

El objetivo de esta práctica es que los alumnos conozcan y aprendan a utilizar librerías criptográficas para así afianzar los conceptos criptográficos estudiados en teoría. Así, se proporciona un enunciado para crear un programa/ aplicación, cuya funcionalidad ha de ser escogida por los alumnos, pero que debe realizar una serie de operaciones criptográficas.

Descripción

El programa que se debe implementar en esta práctica debe realizarse en Python o Java y ha de implementar de forma obligatoria las siguientes funciones criptográficas:

- Cifrado/ descifrado simétrico y asimétrico
- Funciones hash y HMAC
- Firma digital
- Método de autenticación

En todo momento es necesario utilizar algoritmos que se utilicen en la actualidad y que no hayan sido comprometidos. Así, por ejemplo, DES no se debe utilizar, debiéndose utilizar AES en su lugar.

Cifrado y descifrado simétrico o asimétrico

En algún momento, en el sistema a desarrollar se tiene que producir un cifrado y descifrado de información, pudiéndose ver el resultado de dichas operaciones. Nótese que, si el cifrado se aplica, por ejemplo en comunicaciones, siendo transparente para el usuario, se ha de mostrar el resultado en un log o en un mensaje de depuración, junto con el tipo de algoritmo y la longitud de clave utilizada.

En lo referente a la generación de claves hay que considerar lo siguiente:

- Las claves han de tener una longitud apropiada y en relación con el algoritmo que se esté utilizando.

Generación y almacenamiento de claves

Las claves son elementos que han de estar protegidos frente a posibles ataques, aunque hay que considerar las diferencias entre cifrado simétrico, asimétrico, firma digital y HMAC:

- Simétrico: dado que la clave de cifrado es la misma que la de descifrado, ésta podría estar:
    - Almacenada en un fichero/base de datos. Dado que esta clave es secreta, podría ocurrir que se almacenase con algún tipo de protección (por ejemplo, cifrado con una contraseña introducida por el usuario)
    - Recordada por el usuario y utilizada en los momentos adecuados.

- Asimétrico: la clave de cifrado y descifrado son distintas y lo más habitual es que, dada su longitud, no sean introducidas por los usuarios, sino que se creen y posteriormente, el usuario podría utilizarlas porque estén:
    - Almacenadas en un fichero/base de datos y se seleccione la pública o la privada, según corresponda. Es posible que el acceso a la clave privada esté protegido y que ésta sólo sea accesible a través de una contraseña.
- Firma digital: dado que se utiliza cifrado asimétrico para realizar las firmas, las cuestiones a considerar son las anteriormente expuestas.
    - En el caso de realizar firmas se utilizarán claves asimétricas, que también podrían ser utilizadas para el cifrado asimétrico, pero se hará uso de una PKI como la posteriormente descrita para la creación de dichas claves.
- HMAC: en la generación de HMAC se utiliza una única clave, de modo que las consideraciones son las establecidas para el cifrado simétrico.

Hay que remarcar la importancia de que las claves tengan la longitud apropiada, tal y como se indicaba anteriormente.

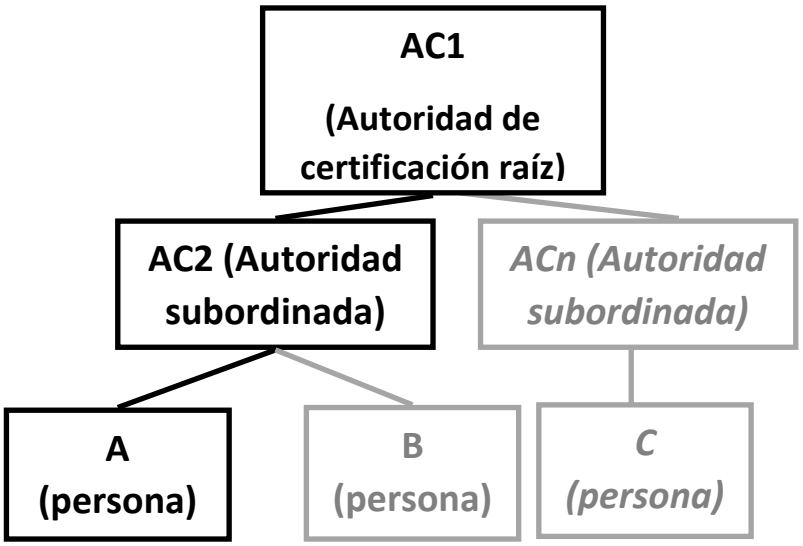
Generación de resúmenes (hashes)

Los hashes se pueden utilizar para multitud de funciones, tan sólo hay que tener en cuenta que:

- Han de tener una longitud adecuada para que no sean fáciles de romper. Esto significa que se han de utilizar algoritmos que no se consideren obsoletos.
- El tiempo en el cómputo de los hashes puede ser un requisito relevante en algunas aplicaciones/ programas, de modo que se han de escoger atendiendo tanto a la seguridad como el propósito establecido.

Firma digital e infraestructura de clave pública

Cada grupo se convierte en una AUTORIDAD DE CERTIFICACIÓN RAÍZ (como puede ser en el mundo real, la Fábrica Nacional de Moneda y Timbre). Dicha Autoridad (AC1) y podrá, bien crear un certificado autofirmado o bien utilizar una PKI como la indicada en la siguiente figura. Por cuestiones organizativas (por ejemplo, para tener una delegación en cada comunidad autónoma) tiene varias AUTORIDADES DE CERTIFICACIÓN SUBORDINADAS (AC2,…, ACn), las cuales se dedican a emitir certificados de clave pública a las personas (A, B, C), como se muestra en la siguiente imagen. Así, hay que crear una PKI compuesta por una AC raíz (AC1) y una AC subordinada (AC2). *Nótese que se puede crear esa PKI u otra de más niveles.*



Para limitar la carga de trabajo, sólo vamos a suponer que existe la autoridad raíz (AC1), una única autoridad subordinada (AC2), de modo que se expedirán certificados para todos los usuarios (personas) que quieran o necesiten hacer uso del programa/ aplicación. y una persona (A).

Métodos de autenticación

Cada grupo puede escoger el método de autenticación que considere más adecuado en cada caso:

- Basado en algo que sabemos: contraseñas, aun teniendo en cuenta que sean robustas y se almacenen convenientemente.
- Basado en algo que tenemos: token, pudiendo ser un mensaje al móvil (aunque sms no son la mejor alternativa), un tipo de tarjeta, etc. Hay múltiples alternativas.
- Basado en algo que somos: rasgo biométrico, desde la huella dactilar, hasta la imagen facial o el iris, entre otros.

Mejoras

En la realización de un programa o aplicación hay gran cantidad de mejoras que pueden realizarse considerando la necesidad de criptografía y seguridad. Unas de las posibles mejoras podrían ser las siguientes:

- Almacenamiento de claves en base de datos – esto supone que las claves se almacenen convenientemente, especialmente en al caso de las que sean privadas o secretas.
- Utilización de distintos modos de operación – siempre utilizando aquellos que se consideren seguros.
- Validación de los datos que introduce un usuario – muchos ataques comienzan por no validar las entradas de los usuarios. Por tanto, no suponer que se introducen los datos correctamente y validarlos, es una buena práctica en materia de seguridad.
- Otras mejoras establecidas por el alumno y que estén convenientemente justificadas.

Evaluación

La siguiente tabla resume la calificación que es posible obtener en esta práctica.

Criterio	Puntuación máxima
Desarrollo:	
Cifrado simétrico/ asimétrico	0,75
Uso de funciones hash/ HMAC	0,75
Firma digital	1,5
Autenticación	1,5
Complejidad y diseño de la aplicación desarrollada	0,5
Mejoras	1 (adicional)
Total desarrollo	**5 (potencialmente +1)**
Memorias	2
Defensas	3
Total práctica	10

Entregables

Se realizarán 2 entregables, uno de ellos asociado a la implementación del cifrado simétrico y la utilización de funciones hash o HMAC y otro asociado a la implementación del cifrado asimétrico, la firma digital y las mejoras, si las hubiese.

Cada uno de los entregables deberá responder a las preguntas aquí planteados y no excederse de la longitud indicada. Además, se debe presentar el **código asociado, el cual debe estar debidamente documentado y utilizando buenas prácticas de programación y diseño de software**. Si no se siguen estos criterios habrá una penalización en la puntuación.

Entregable 1

Grupo:

ID de grupo de prácticas:

Nombre de todos los alumnos:

Responda a las siguientes preguntas. Incluya capturas de pantalla que soporten sus argumentos.

- ¿Cuál es el propósito de su aplicación?
- ¿Para qué utiliza el cifrado simétrico? ¿Qué algoritmos ha utilizado y por qué? ¿Cómo gestiona las claves?
- ¿Para qué utiliza las funciones hash o HMAC? ¿Qué algoritmos ha utilizado y por qué? En caso de HMAC, ¿cómo gestiona la clave/s?

El límite de páginas es de 5, excluyendo la portada y los anexos. El tamaño de letra debe ser 11, con interlineado simple, de tipo Calibri/ Arial o Times New Roman.

Entregable 2

Grupo:

ID de grupo de prácticas:

Nombre de todos los alumnos:

Responda a las siguientes preguntas. Incluya capturas de pantalla que soporten sus argumentos.

- ¿Cuál es el propósito de su aplicación? (repetir lo indicado en informe 1)
- ¿Para qué utiliza el cifrado asimétrico? ¿Qué algoritmos ha utilizado y por qué? ¿Cómo gestiona las claves?
- ¿Para qué utiliza la firma digital? ¿Qué algoritmos ha utilizado y por qué? ¿Cómo gestiona las claves? ¿Cuál es la PKI que ha desarrollado?
- ¿Qué tipo de autenticación ha implementado? ¿Por qué ha escogido este tipo y no otro? ¿Cómo lo ha implementado?
- Si ha realizado mejoras, explique cuáles y las implicaciones de seguridad de cada una de ellas en su programa/ aplicación.

El límite de páginas es de 8, excluyendo la portada y los anexos. El tamaño de letra debe ser 11, con interlineado simple, de tipo Calibri/ Arial o Times New Roman.