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:

1. Autenticación de usuarios
2. Cifrado/descifrado simétrico (o asimétrico)
3. Generación/verificación de etiquetas de autenticación de mensajes (e.g., con funciones hash y HMAC)
4. Generación/verificación de firma digital
5. Autenticación de las claves públicas mediante certificados (despliegue de PKI)

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.

1. Autenticación de usuarios

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

- Basado en algo que sabemos: contraseñas; se deben almacenar convenientemente y a ser posible que sean robustas. Esta será la opción que se requerirá que se implemente en todos los casos como mínimo.
- 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.

2. 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 (o cualquiera de las operaciones criptográficas siguientes) se aplica, por ejemplo, en comunicaciones o es 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.

3. Generación/verificación de etiquetas de autenticación de mensajes

La información de valor que se intercambie o almacene además de cifrada debe estar autenticada. Los algoritmos de códigos (o etiquetas) de autenticación de mensajes proporcionan este servicio.

4. Generación/verificación de firma digital

Se generarán y verificarán firmas digitales emitidas sobre algún tipo de información. En el caso mínimo será el propio sistema el que genere las firmas, pero dependiendo de las necesidades del sistema podría ser adecuado que las generasen los propios usuarios. La verificación como mínimo se realizará de forma automática (reflejando el resultado en un log o mensaje de depuración) o permitiendo a los usuarios solicitar dicha verificación.

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

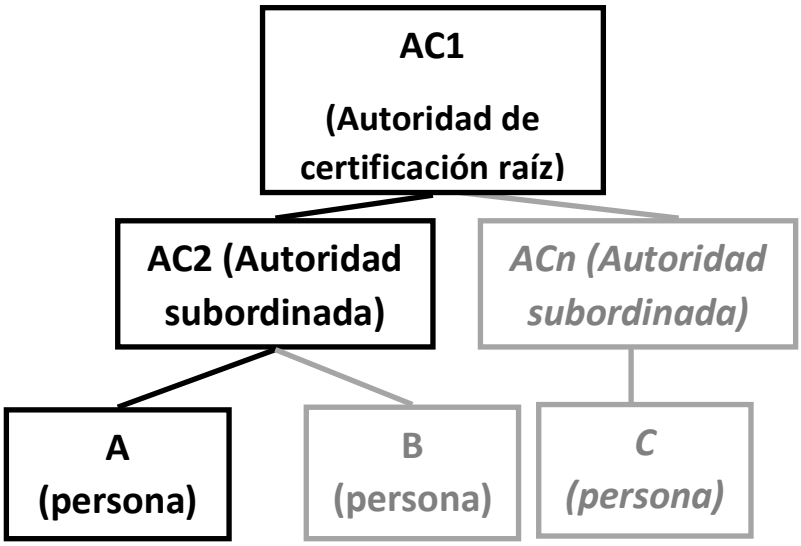
- Las claves asimétricas han de tener una longitud apropiada y en relación con el algoritmo que se esté utilizando.

5. Autenticación de las claves públicas mediante certificados (despliegue de PKI)

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) contará con un certificado autofirmado.

Por cuestiones organizativas (por ejemplo, para tener una delegación en cada comunidad autónoma) podría ser conveniente contar con varias AUTORIDADES DE CERTIFICACIÓN SUBORDINADAS (AC2,…, ACn), las cuales se dedicarían a emitir certificados de clave pública a las personas (A, B, C), como se muestra en la siguiente imagen.

Así, las infraestructuras de clave pública (PKIs) desplegadas estarán compuestas como mínimo por una AC raíz (AC1) y, de forma altamente recomendable, por una AC subordinada (AC2). *Nótese que se puede crear esa PKI u otra de más niveles o con mayor número de autoridades subordinadas.*



Se expedirán certificados de clave pública para todos los usuarios (personas) que quieran o necesiten hacer uso de las claves privadas (es decir, al menos para las entidades generadoras de firmas digitales, y en caso de que se haya incluido cifrado asimétrico, para aquellas entidades que lo utilicen).

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, cifrada con una contraseña introducida por el usuario).
  - También es posible no almacenar la clave:
    - Si el usuario la memoriza el usuario y la proporciona en los momentos necesarios.
    - Si ésta se genera a partir de la contraseña del usuario.
- Etiquetas de autenticación de mensajes: en la generación de MACs se utiliza una única clave, de modo que las consideraciones son las establecidas para el cifrado simétrico.
- Cifrado asimétrico/firma digital: las claves de cifrado y descifrado (generación y verificación en el caso de la firma) 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. Para autenticar las claves públicas se hará uso de la PKI como la anteriormente descrita.

Hay que remarcar la importancia de que las claves tengan la longitud y entropía apropiada, así como que se almacenen adecuadamente.

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 en base de datos.
- Utilización de distintos modos de operación (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.
- Rotación de claves.
- Otras mejoras establecidas por el alumno y que estén convenientemente justificadas (deben aceptarse como tales por los profesores).

Evaluación

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

Criterio	**Puntuación máxima**	**Evaluación**
Desarrollo:		
Autenticación de usuarios	1	Eval. 1
Cifrado simétrico/asimétrico	0,75	Eval. 1
Etiquetas de autenticación de mensajes	0,75	Eval. 1
Firma digital	1	Eval. 2
Certificados y PKI	1	Eval. 2
Complejidad y diseño de la aplicación desarrollada	0,5	Eval. 2
Mejoras (opcional)	1 (adicional)	Eval. 2
Total desarrollo	**5 (potencialmente +1)**	
Documentación y defensa:		
Memorias	1 + 1	Eval. 1/Eval. 2
Defensas	1,5 + 1,5	Eval. 1/Eval. 2
Total documentación y defensa	**5**	
Total práctica	**10**	

Entregables

Se realizarán 2 entregables, uno de ellos asociado a los aspectos etiquetados como Eval. 1 en la tabla y otro asociado a los aspectos etiquetados como Eval. 2 en la tabla.

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:

Correo de todos los alumnos:

Repositorio de código (enlace) si lo hubiera:

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

- ¿Cuál es el propósito de su aplicación?
- ¿Cómo se realiza la autenticación de usuarios? ¿Qué algoritmos ha utilizado y por qué? Detalle cómo se gestionan las contraseñas de los usuarios y si se generan claves a partir de éstas.
- ¿Para qué utiliza el cifrado simétrico? ¿Qué algoritmos ha utilizado y por qué? ¿Cómo gestiona las claves? Explique los mismos aspectos si se utiliza cifrado asimétrico para este tipo de cifrado.
- ¿Para qué utiliza las funciones de códigos de autenticación de mensajes (MAC)? ¿Qué algoritmos ha utilizado y por qué? ¿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:

Correo de todos los alumnos:

Repositorio de código (enlace) si lo hubiera:

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 la firma digital? ¿Qué algoritmos ha utilizado y por qué? ¿Cómo se gestionan y almacenan las claves y las firmas?
- ¿Cómo se generan los certificados de clave pública? ¿Qué jerarquía de autoridades de certificación se ha desplegado? ¿Por qué ha escogido esta configuración y no otra? ¿Cómo se ha implementado? ¿En qué momento se utilizan los certificados y para qué?
- Discuta la complejidad y diseño del código de su aplicación.
- 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 10, excluyendo la portada y los anexos. El tamaño de letra debe ser 11, con interlineado simple, de tipo Calibri/ Arial o Times New Roman.