Estrategias de seguridad (2023-2024)

Trabajo práctico

Se realizará un trabajo en grupo (3-4 alumnos) relacionado con la seguridad de la información que deberá implementarse en un lenguaje de programación que permita la comprobación del resultado de la práctica en Windows.

Se entregará (*antes de las 23:59 h. del día 12/05/2023*) como trabajo definitivo en el classroom los archivos con la versión ejecutable y el código fuente, además de la correspondiente memoria que no excederá de 25 folios por una cara y debe contener al menos los siguientes apartados:

- Índice
- Descripción general de los archivos entregados (breve descripción del contenido de cada fichero o carpeta)
- Manual de usuario (instrucciones de uso, ejemplos, etc.)
- Documentación sobre la implementación:
    - **Diagramas** de los principales protocolos implementados
    - **Explicación** de las principales funciones

La práctica consistirá en la realización de un programa de gestión documental que permitirá como mínimo:

- Gestionar *n* documentos, cada documento tendrá que contener como mínimo un título, una descripción, *n* archivos asociados. La persistencia de los datos se debe realizar en archivos de textos que serialicen la información de cada proyecto (xml, json, etc.). La información global de configuración que pueda necesitar el cliente y el servidor, tambien debe ser almacenada en archivos. Todos estos archivos se deben almacenar de forma segura. No se debe utilizar un gestor de BBDD.
- Registrarse en un servidor para poder subir los documentos y almacenarlos de forma segura.
- Compartir de forma segura en modo lectura los documentos con otros usuarios.

La aplicación constará de un modo que permita usarla de forma insegura. También se programará un cliente malicioso de la aplicación que, en el caso de usarla en modo no seguro, sea capaz de:

- Descifrar los archivos almacenados cifrados con AES: El modo inseguro debe usar una contraseña elegida entre las 10 contraseñas más usadas. El cliente malicioso intentará descifrar el archivo, probando una a una las 10 contraseñas más usadas y detectará cuando ha tenido éxito.
- Descargase archivos de cualquier usuario sin iniciar sesión: El modo inseguro no protegerá el endpoint de descargar archivo y listar archivos de un usuario con ninguna técnica de control de acceso o sesión (usuario/contraseña, token, etc.) El cliente malicioso probará urls del endpoint de listar datos subidos hasta encontrar nombres de usuario que devuelvan datos, después usará el endpoint de recuperar archivo para descargárselo.

Se realizará la propuesta en las siguientes fases:

1. En una **primera fase** se debe diseñar e implementar un sistema que permita al usuario, **almacenar paquetes de datos con el conjunto de archivos y los datos asociado**s en un archivo de datos (clases serializadas, xml, json, csv, etc…) que se deberán **comprimir** en un archivo y **cifrar** con AES usando una contraseña aleatoria para cada archivo comprimido. También se implementará el modo

inseguro y el cliente malicioso para descifrar archivos cifrados con contraseñas débiles.

2. En una **segunda fase** se debe diseñar e implementar un sistema para **registrarse, autenticarse e interactuar** con un **servidor** por un **canal seguro**. Cualquier cliente podrá registrarse en el servidor, se debe autenticar en todas las interacciones, como mínimo, con un usuario y contraseña que proporcionará al servidor en el momento del registro. Esta **contraseña de login** debe ser segura por longitud y complejidad y **distinta de la contraseña de cifrado** de los datos del usuario. Debe ser almacenada de forma segura por el servidor. Las interacciones mínimas a implementar son: **subir archivo de datos cifrado, listar datos subidos a servidor** y **recuperar un archivo de datos cifrado**. También se implementará el modo inseguro y el cliente malicioso para descargase archivos de cualquier usuario sin iniciar sesión.
3. En una **tercera fase** se generarán un par de claves, pública y privada, con RSA en cada cliente antes de registrarse en el servidor. Se subirá, en el registro del usuario, la clave publica al servidor para que la use y distribuya y la privada estará almacenada en el archivo de datos principal cifrado con la contraseña de acceso a la aplicación. Se añadirá en el cliente y servidor la función de **compartirlos con el resto de los usuarios** usando un esquema de **clave publica** con un subconjunto de usuarios del servidor.

Retos adicionales (pueden añadirse o mejorarlos en cualquiera de las fases):

1. Añadir nuevos campos al documento.
2. Generador automático de contraseñas seguras de registro.
3. Interfaz gráfica de usuario avanzada.
4. Mejorar el sistema de autenticación: usar tokens en las interacciones, autenticación en dos pasos, etc.
5. Conocimiento cero en del servidor.
6. Revocar permisos de compartición.
7. Edición de los documentos de datos después de subidos al servidor.
8. Sistema de logs avanzados.
9. Añadir un registro de eventos de cambios en el proyecto.
10. Añadir firma digital de los cambios del registro de eventos.
11. Añadir nuevas vulnerabilidades al programa y forma de explotarlas al cliente malicioso.
12. Otros retos propuestos por el alumnado.

Se hará un seguimiento de los trabajos en las clases de prácticas (orientación, sugerencias, etc.) a fin de mejorar aquellos aspectos que lo necesiten.

Los trabajos serán entregados a través del Classroom de la asignatura.

- Se entregará la primera fase **antes de las 23:59 h. del día 10 de marzo de 2023**. Se entregará el código fuente en su estado actual, un borrador de la memoria completada hasta la primera fase y un **video** de un máximo **5 minutos** en el que se mostraran los **hitos** de la primera fase completados, así como **una traza del código para cifrar**.
- Se entregará la segunda fase **antes de las 23:59 h. del día 14 de abril de 2023**. Se entregará el código fuente en su estado actual, un borrador de la memoria completada hasta la segunda fase y un **video** de un máximo **5 minutos** en el que se mostraran los **hitos** de la segunda fase completados, así como una **traza del código para registrarse y subir un archivo al servidor**.
- **Los trabajos definitivos (memoria y código fuente) serán entregados antes de las 23:59 h. del día 12 de mayo de 2023 y serán presentados** en las clases de prácticas de los días 13 y 20 de mayo; para lo cual se hará la correspondiente asignación de turno para la presentación.

PUNTUACIÓN:

10% Objetivos de primera fase.
15% Objetivos de segunda fase.
20% Objetivos de tercera fase.
10% Memoria.
35% Retos adicionales.
5% Consecución de los objetivos mínimos de las fases primera y segunda en fecha de entrega.
5% Defensa del trabajo final.