PSP - U6 Actividades

Descargar estos apuntes

Índice

Nombre del proyecto

En estas actividades volvemos a usar la nomenclatura de clases y proyectos explicada en la hoja de actividades de autoevaluación de la U2

1. Actividad U6A01_FileCheckSum

Vamos a realizar un programa que genere un resumen de uno o varios archivos de texto.

El programa recibirá como argumentos el nombre de los archivos a procesar y, como último argumento, el algoritmo a utilizar.

Cada archivo se leerá línea a línea y se irá generando el resumen por líneas. Una vez completado el resumen se procederá como se detalla a continuación:

- Si existe un archivo con el mismo nombre y acabado con el nombre del algoritmo indicado (por ejemplo, para el archivo prueba1.txt y el algoritmo SHA-1, existe el archivo prueba1.txt.sha-1), se leerá el valor almacenado en dicho archivo y se comparará con el resumen obtenido.
  - Si son iguales se indicará que el archivo no ha sido modificado
  - Si no coinciden se mostrará un mensaje indicando que el archivo ha sido modificado y se sobreescribirá el valor del archivo que contiene el resumen.
- Si no existe un archivo con el mismo nombre y acabado con el nombre del algoritmo indicado, se creará ese archivo y se guardará el resumen obtenido, en hexadecimal. Se mostrará un mensaje indicando que se ha creado el resumen para el archivo.

2. Actividad U6A02_CreditCardHack

Hemos conseguido una la base de datos con información de tarjetas de crédito.
Investigando hemos descubierto que este banco almacena el pin de las tarjetas con un resumen SHA1.

Nos hemos fijado en las siguientes entrada de la base de datos

Titular	PIN
Amancio Ortega	7110EDA4D09E062AA5E4A390B0A572AC0D2C0220
Luis Bárcenas	39DFA55283318D31AFE5A3FF4A0E3253E2045E43
Juan Carlos de Borbón	4170AC2A2782A1516FE9E13D7322AE482C1BD594

Sabiendo que el PIN de las tarjetas es un número de 4 dígitos, prepara un programa que reciba como parámetro n resúmenes correspondientes a PINs de tarjetas y, usando un algoritmo de fuerza bruta, busque a qué PIN corresponden esos resúmenes.

3. Actividad U6A03_EchoSeguro

Partiendo del cliente y servidor de ECHO que hicimos en el tema anterior, modifícalos para que la información que intercambian vaya cifrada.

La clave simétrica compartida la leerán de un archivo que estará en la carpeta claves del proyecto de cada uno.

Crea una clase auxiliar en el proyecto del server que genere una clave y guarde la clave en un archivo dentro de la carpeta claves. La generación de la clave debe ser aleatoria, indicando el tamaño de clave según el algoritmo de cifrado a utilizar.

Crea una clase de utilidades que tenga métodos para cifrar y descifrar. Esta clase puede ser compartida entre el cliente y el servidor.

Prueba qué pasa si la clave del cliente y del servidor no son la misma.

4. Actividad U6A04_EchoAsimétrico

Partiendo del cliente y servidor de ECHO que hicimos en el tema anterior, modifícalos para que la información que intercambian vaya cifrada y codificada en Base64.

En este caso la información estará cifrada con una clave privada en el servidor y la correspondiente clave pública en el cliente.

Crea una carpeta JavaSecurity en tu carpeta personal para guardar las claves que generes (publicEcho.der y privateEcho.pkcs8)
No uses rutas absolutas. Usa las System.Properties para llegar a la carpeta personal.

Cada parte debe mostrar tanto la información que le llega (en formato raw) y la información descifrada.

5. Actividad U6A05_ChatDuoSSL

Modifica la actividad de ChatDuo, la tuya o la que os he dejado como solución, y añade seguridad SSL en las comunicaciones.

Crea una carpeta **JavaSecurity** en tu carpeta personal para guardar los keyStores que crees, tanto para el cliente (**ChatDuoClientTrustServer**, password: duo) como para el servidor (**ChatDuoServerCertificate**, password: chat).

No uses rutas absolutas. Usa las System.Properties para llegar a la carpeta personal de cada usuario.

> ⚠ **Longitud de clave**
>
> Dependiendo de la versión del JDK, e incluso del tipo de JDK que empleemos, podemos encontrarnos con el siguiente error
>
> > “Keystore password must be at least 6 characters”
>
> En ese caso, añadid el año al password de los keyStores, teniendo por tanto duo2025 en el cliente y chat2025 en el servidor.