Cifrado de un fichero con clave privada DES

Preparar fichero a encriptar

- Creamos un archivo de nombre fichero, con algún texto
- Lo situamos en el directorio c:\cripto

Paquetes, Clases y métodos

- Los paquetes necesarios son:

```
import java.security.*;
import javax.crypto.*;
import java.io.*;
```

- Para cifrar y descifrar el fichero básicamente utilizaremos:
  - KeyGenerator.getInstance("DES")
    - .init(56)
    - .generateKey
  - Cipher.getInstance("DES")
    - .init(Cipher.ENCRYPT_MODE, key)
    - .doFinal(textoPlano)
    - .init(Cipher.DECRYPT_MODE, key)
    - .doFinal(textoCifrado)

Método main()

- El método main():
  - Declara un objeto tipo SecretKey
  - Invoca a un método para cifrar fichero cifrarFichero()
  - Invoca a un método para descifrar fichero descifrarFichero()

```
public static void main(String[] Args) {
  //declara e incializa objeto tipo clave secreta
  SecretKey clave = null;

  //llama a los métodos que encripta/desencripta un fichero
  try {
    //Llama al método que encripta el fichero que se pasa como parámetro
    clave = cifrarFichero("c:\\cripto\\fichero");
    //Llama la método que desencripta el fichero pasado como primer parámetro
    descifrarFichero("c:\\cripto\\fichero.cifrado", clave,
            "c:\\cripto\\fichero.descifrado");
  } catch (Exception e) {
    e.printStackTrace();
  }
}
```

Método cifrarFichero() I

Se le pasa como parámetro el fichero a cifrar y retorna la clave utilizada para cifrar

La primera tarea de este método será crear la clave secreta para el algoritmo DES con un tamaño por ejemplo de 56 bits.

```
private static SecretKey cifrarFichero(String file) throws NoSuchAlgorithm      ion,
    FileInputStream fe = null; //fichero de entrada
    FileOutputStream fs = null; //fichero de salida
    int bytesLeidos;

    //1. Crear e inicializar clave
    System.out.println("1.-Genera clave DES");
    //crea un objeto para generar la clave usando algoritmo DES
    KeyGenerator keyGen = KeyGenerator.getInstance("DES");
    keyGen.init(56); //se indica el tamaño de la clave
    SecretKey clave = keyGen.generateKey(); //genera la clave privada

    System.out.println("Clave");
    mostrarBytes(clave.getEncoded()); //muestra la clave
    System.out.println();
```

Método cifrarFichero() II

```
    //Se Crea el objeto Cipher para cifrar, utilizando el algoritmo DES
    Cipher cifrador = Cipher.getInstance("DES");
    //Se inicializa el cifrador en modo CIFRADO o ENCRIPTACIÓN
    cifrador.init(Cipher.ENCRYPT_MODE, clave);
    System.out.println("2.- Cifrar con DES el fichero: " + file
            + ", y dejar resultado en " + file + ".cifrado");
//declaración  de objetos
    byte[] buffer = new byte[1000]; //array de bytes
    byte[] bufferCifrado;
    fe = new FileInputStream(file); //objeto fichero de entrada
    fs = new FileOutputStream(file + ".cifrado"); //fichero de salida
    //lee el fichero de 1k en 1k y pasa los fragmentos leidos al cifrador
    bytesLeidos = fe.read(buffer, 0, 1000);
    while (bytesLeidos != -1) {//mientras no se llegue al final del fichero
      //pasa texto claro al cifrador y lo cifra, asignándolo a bufferCifrado
      bufferCifrado = cifrador.update(buffer, 0, bytesLeidos);
      fs.write(bufferCifrado); //Graba el texto cifrado en fichero
      bytesLeidos = fe.read(buffer, 0, 1000);
    }
    bufferCifrado = cifrador.doFinal(); //Completa el cifrado
    fs.write(bufferCifrado); //Graba el final del texto cifrado, si lo hay
    //Cierra ficheros
    fe.close();
    fs.close();
    return clave;
```

Crea objeto cifrador y lo pone en modo Encriptación.

Va leyendo el fichero y pasando al cifrador para que lo cifre

Método descifrarFichero()

```
private static void descifrarFichero(String file1, SecretKey key, String file2)
  FileInputStream fe = null; //fichero de entrada
  FileOutputStream fs = null; //fichero de salida
  int bytesLeidos;
  Cipher cifrador = Cipher.getInstance("DES");
  //3.- Poner cifrador en modo DESCIFRADO o DESENCRIPTACIÓN
  cifrador.init(Cipher.DECRYPT_MODE, key);
  System.out.println("3.- Descifrar con DES el fichero: " + file1
          + ", y dejar en  " + file2);
  fe = new FileInputStream(file1);
  fs = new FileOutputStream(file2);
  byte[] bufferClaro;
  byte[] buffer = new byte[1000]; //array de bytes
  //lee el fichero de 1k en 1k y pasa los fragmentos leidos al cifrador
  bytesLeidos = fe.read(buffer, 0, 1000);
  while (bytesLeidos != -1) {//mientras no se llegue al final del fichero
    //pasa texto cifrado al cifrador y lo descifra, asignándolo a bufferClaro
    bufferClaro = cifrador.update(buffer, 0, bytesLeidos);
    fs.write(bufferClaro); //Graba el texto claro en fichero
    bytesLeidos = fe.read(buffer, 0, 1000);
  }
  bufferClaro = cifrador.doFinal(); //Completa el descifrado
  fs.write(bufferClaro); //Graba el final del texto claro, si lo hay
  //cierra archivos
  fe.close();
  fs.close();
}
```

Parámetros:
file1: fichero cifrado
key: Clave privada
file2: fichero desencriptado

Crea objeto cifrador y lo pone en modo Desencriptación.

Va leyendo el fichero y pasando al cifrador para que lo descifre .

Resultado de ejecución

C:\cripto\fichero

Este texto será encriptado y desencriptado mediante clave privada y algoritmo DES

Fichero original

C:\cripto\fichero.cifrado

©_oQÀ]÷§,PÔ«Ž
ì2CÖÁ¤Ìvˆsó‰tn+¢2CÖÁ¤ÌvCðÆnF
çËZ°~KYÖ4R‡(5ˆˆ-å›⊙,)XŠ“—

Fichero obtenido en la encriptación

C:\cripto\fichero.descifrado

Este texto será encriptado y desencriptado mediante clave privada y algoritmo DES

Fichero obtenido en la desencriptación

Credenciales

Imagen	Datos de licencia
Todas las capturas de pantalla de este presentación, salvo la diapositiva 2 tienen como datos de licencia: Autoría: Isabel M. Cruz Granados Licencia: Uso educativo-no comercial. Procedencia: Captura de pantalla del programa y editor de código NetBeans, propiedad Sun Microsystems, bajo licencia GNU GPL v2.	
	Autoría: Isabel M. Cruz Granados Licencia: Uso educativo-no comercial. Procedencia: Dibujo realizado por la autora.