Universidad de los Andes
Ingeniería de Sistemas y Computación
ISIS 2203 Infraestructura Computacional

Taller Librería de Seguridad

El propósito es estudiar algunos métodos de cifrado de información implementados en librerías Java (java.crypto.*) .

Cifrado Asimétrico

Escriba una clase para ejecutar un algoritmo de cifrado asimétrico. A continuación encuentra parte del código necesario para cifrar y descifrar información:

1. Definición de atributos de la clase:

```
private final static String ALGORITMO="RSA";
private KeyPair keyPair;
```

2. Método para cifrar

```
public byte[] cifrar() {
      try {
            KeyPairGenerator generator =
                  KeyPairGenerator.getInstance(ALGORITMO);
            generator.initialize(1024);
            keyPair = generator.generateKeyPair();
            Cipher cipher = Cipher.getInstance(ALGORITMO);
            BufferedReader stdIn =
                  new BufferedReader(new InputStreamReader(System.in));
            String pwd = stdIn.readLine();
            byte [] clearText = pwd.getBytes();
            String s1 = new String (clearText);
            System.out.println("clave original: " + s1);

            cipher.init(Cipher.ENCRYPT_MODE, keyPair.getPublic());
            long startTime = System.nanoTime();
            byte [] cipheredText = cipher.doFinal(clearText);
            long endTime = System.nanoTime();
            System.out.println("clave cifrada: " + cipheredText);
            System.out.println("Tiempo asimetrico: " +
                                    (endTime - startTime));
            return cipheredText;
      }
      catch (Exception e) {
            System.out.println("Excepcion: " + e.getMessage());
            return null;
      }
}
```

Método para descifrar

```
public void descifrar(byte[] cipheredText) {
	try {
		Cipher cipher = Cipher.getInstance(ALGORITMO);
		cipher.init(Cipher.DECRYPT_MODE, keyPair.getPrivate());
		byte [] clearText = cipher.doFinal(cipheredText);
		String s3 = new String(clearText);
		System.out.println("clave original: " + s3);
	}
	catch (Exception e) {
		System.out.println("Excepcion: " + e.getMessage());
	}

}
```

Escriba una clase para crear una instancia de la clase que cifra y descifra información de forma asimétrica.

1. En el main
    a. Cree una instancia de la clase que cifra y descifra información de forma asimétrica
    b. Ejecute un llamado al método cifrar (con los parámetros apropiados)
    c. Ejecute un llamado al método descifrar (con los parámetros apropiados)
    d. Ejecute el programa y observe los resultados (el texto original y el texto descifrado deberían coincidir)
2. Responda las siguientes preguntas:
    a. ¿Cuál es el propósito del llamado al método initialize con el parámetro 1024?
    b. Describa brevemente el funcionamiento del algoritmo RSA.
    c. ¿Cuál llave se usa para cifrar la información en el código presentado?
    d. ¿Cuál llave se usa para descifrar la información?

Cálculo de Digest (o código de resumen):

Escriba una clase para ejecutar un algoritmo de digest. A continuación encuentra parte del código necesario:

1. Definición de atributos de la clase

```
private final static String ALGORTIMO = "DSA";
private KeyPair keyPair;
```

2. Método para calcular

```
public byte[] calcular() {
	try {
		KeyPairGenerator generator =
			KeyPairGenerator.getInstance(ALGORTIMO);
		generator.initialize(1024);
		keyPair = generator.generateKeyPair();
		PrivateKey priv = keyPair.getPrivate();
		PublicKey pub = keyPair.getPublic();
		System.out.println(pub);

		Signature firma = Signature.getInstance(priv.getAlgorithm());
```

```
            firma.initSign(priv);

            FileInputStream arch = new FileInputStream(ARCHIVO);
            BufferedInputStream bufin = new BufferedInputStream(arch);
            byte [] buffer = new byte[1024];
            int len;
            while (bufin.available() != 0) {
                  len = bufin.read(buffer);
                  firma.update(buffer,0,len);
            }
            bufin.close();

            byte [] signature = firma.sign();
            String s1 = new String(signature);
            System.out.println("Firma: " + s1);

            return signature;
      }
      catch (Exception e) {
            System.out.println("Excepcion: " + e.getMessage());
            return null;
      }
}
```

3. Método para verificar

```
public void verificar(byte[] firma) {
      try {
            PublicKey pub = keyPair.getPublic();
            Signature sig = Signature.getInstance(pub.getAlgorithm());
            sig.initVerify(pub);

            FileInputStream arch = new FileInputStream(ARCHIVO);
            BufferedInputStream bufin = new BufferedInputStream(arch);
            byte [] buffer = new byte[1024];
            int len;
            while (bufin.available() != 0) {
                  len = bufin.read(buffer);
                  sig.update(buffer,0,len);
            }
            bufin.close();
            boolean verifies = sig.verify(firma);
            System.out.println("Verificacion: " + verifies);

      }
      catch (Exception e) {
            System.out.println("Excepcion: " + e.getMessage());
      }
}
```

Escriba una clase para crear una instancia de la clase que calcula un digest y lo verifica.

1. En el main
    a. Cree una instancia de la clase que calcula un código digest y lo verifica
    b. Ejecute un llamado al método calcular (con los parámetros apropiados)

c. Ejecute un llamado al método verificar (con los parámetros apropiados)
d. Ejecute el programa y observe los resultados (la verificación debería ser exitosa)
e. Modifique el programa, es decir los métodos calcular y verificar, para generar dos códigos que correspondan a datos diferentes y ejecute la verificación del primero contra el código digest del segundo (la verificación debería fallar).

2. Responda las siguientes preguntas:
   a. En el método calcular, ¿cuál es el propósito del ciclo que ejecuta el llamado al método update para una firma?
   b. Describa brevemente el propósito del algoritmo DSA.
   c. ¿Cuál llave se usa para generar el código?
   d. ¿Cuál llave se usa para verificar el código?