ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

INSTITUTO DE POSTGRADO Y EDUCACIÓN CONTINUA

MAESTRÍA EN SEGURIDAD TELEMÁTICA

TAREA No. 04 FUNCIONES DE RESUMEN

MÓDULO: Criptografía y Encriptación

DOCENTE: Ing. Paúl Paguay

NOMBRE(s): Guillermo Valencia, Marcelo Núñez, Raúl Alarcón, Milton Escobar

FECHA: 10/05/2018

1. INTRODUCCIÓN

Son funciones matemáticas de transformación de valores de un conjunto grande en otro conjunto de valores más pequeño. Son funciones resistentes a colisiones. Las funciones resumen dependen de una clave criptográfica. Las funciones hash pueden tomar valores de un tamaño variable y siempre producen una salida de longitud fija.

Las funciones resumen conocidas como funciones hash o funciones digest, son funciones computables mediante la utilización de algoritmos que tiene como entrada un conjunto de elementos que por lo general son cadenas y se los convierte en un rango de salida finito, normalmente cadenas de longitud fija.

$H: U \rightarrow M$

$x \rightarrow h(x)$

Es decir, la función actúa como una proyección del conjunto U sobre el conjunto M.

Observar que M puede ser un conjunto definido de enteros. En este caso podemos considerar que la longitud es fija si el conjunto es un rango de números de enteros ya que podemos considerar que la longitud fija es la del número con mayor número de cifras. Todos los números se pueden convertir al número especificado de cifras simplemente anteponiendo ceros.

Normalmente el conjunto U tiene un número elevado de elementos y M es un conjunto de cadenas con un número más o menos pequeño de símbolos. La idea básica de un valor hash es que sirva como una representación compacta de la cadena de entrada.

Por esta razón se dice que estas funciones resumen datos del conjunto dominio.

2. OBJETIVOS

- **Usos de Funciones de resumen.**
- **Utilizar los algoritmos MD5,HMAC y SHA-3.**
- **Envío de mensajes entre dos computadoras con intercambio de mensajes de firma electrónica utilizando HMAC.**

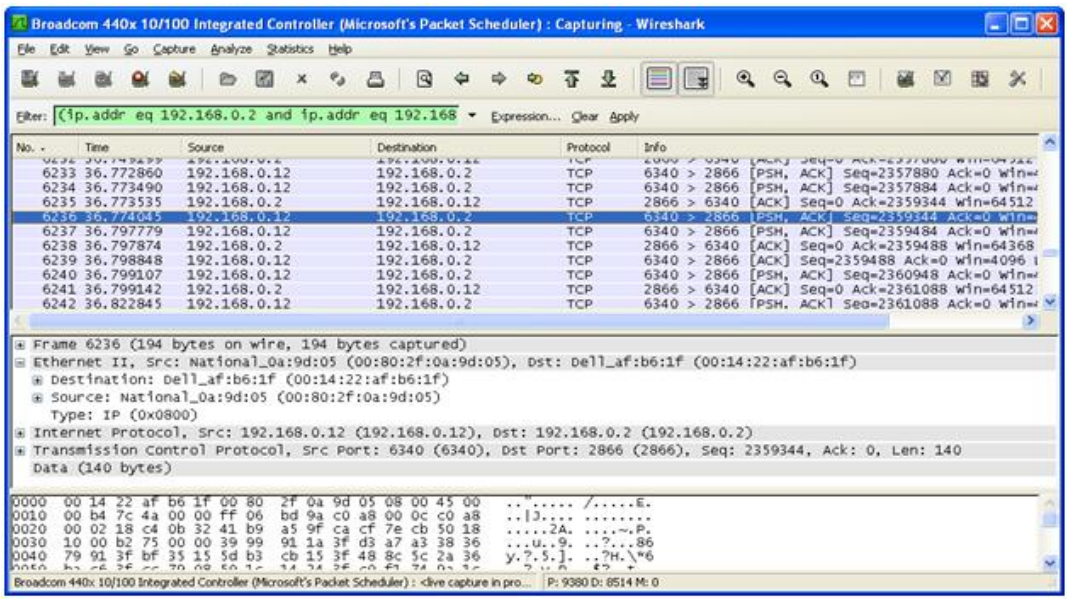
3. **DESARROLLO**

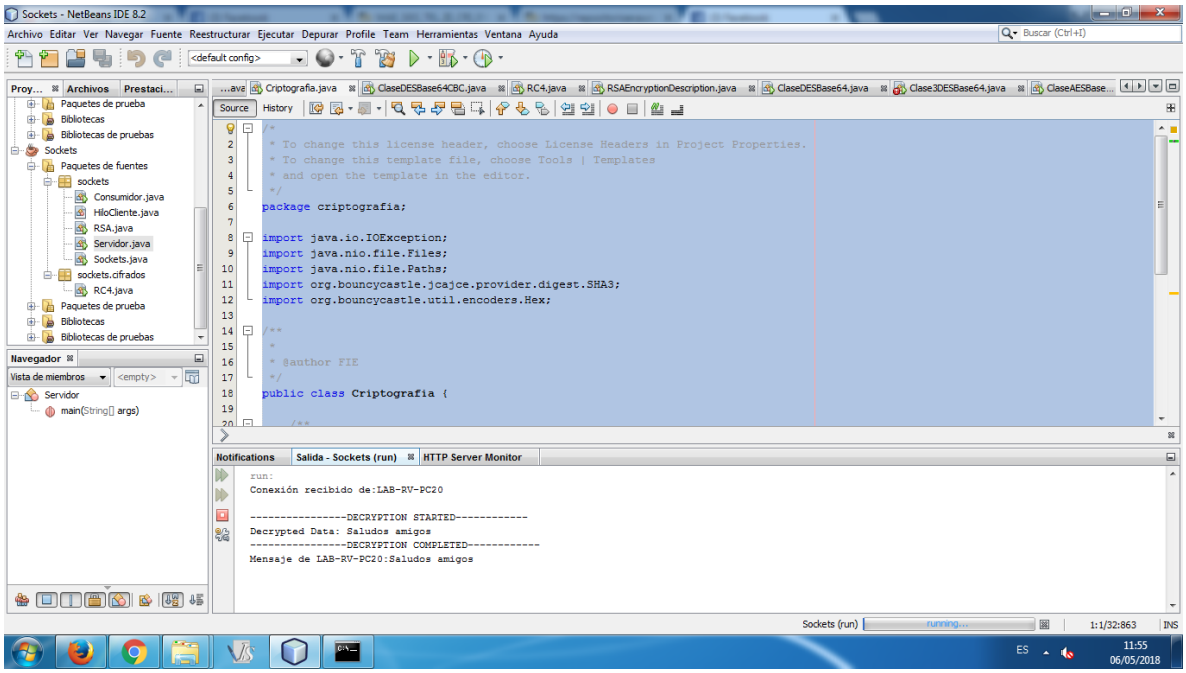
1. **Utilizando los algoritmos MD5, HMACy SHA-3. encuentre el resumen del siguiente mensaje:**
   MAESTRIA EN SEGURIDAD TELEMÁTICA

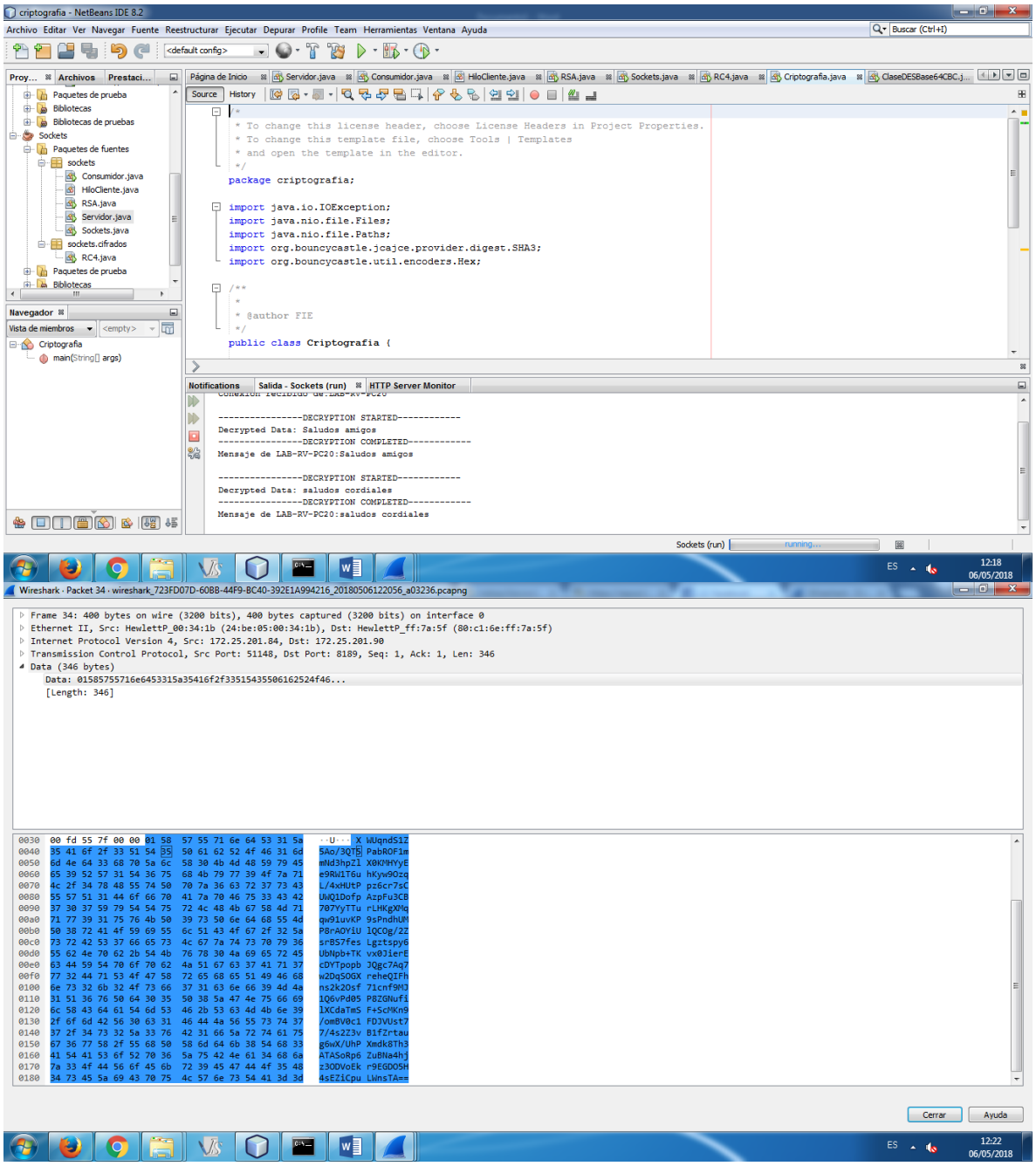
```
import md5
>>> m = md5.new()
>>> m.update("MAESTRIA EN SEGURIDAD TELEMÁTICA")
>>> m
<md5 HASH object @ 0x7fe61f431648>

>>> import hashlib
>>> m = hashlib.sha256()
>>> m.update("MAESTRIA EN SEGURIDAD TELEMÁTICA")
>>> m
<sha256 HASH object @ 0x7fe61f431788>
>>>m=hashlib.sha256("MAESTRIA.EN.SEGURIDAD.TELEMÁTICA").hexdige
st()
>>> m
'ccdf84cf8d7bfb24b267ad5dbcb62b969e578e8162cd1cb0c2f0f0b00ea4795b'

>>> import hmac
>>> import base64
>>> import hashlib
>>> key = 'password'
>>> digest = hmac.new(key, 'MAESTRIA EN SEGURIDAD TELEMÁTICA',
hashlib.sha1).digest()
>>> print base64.encodestring(digest)
KzaHASbfSKGaRp0kZF6TlQNd03E=
```

2. **En grupos de 3 o 4 personas, codificar un cliente (Consumidor)y un servidor que intercambie mensajes con firma electrónica utilizando HMAC:**

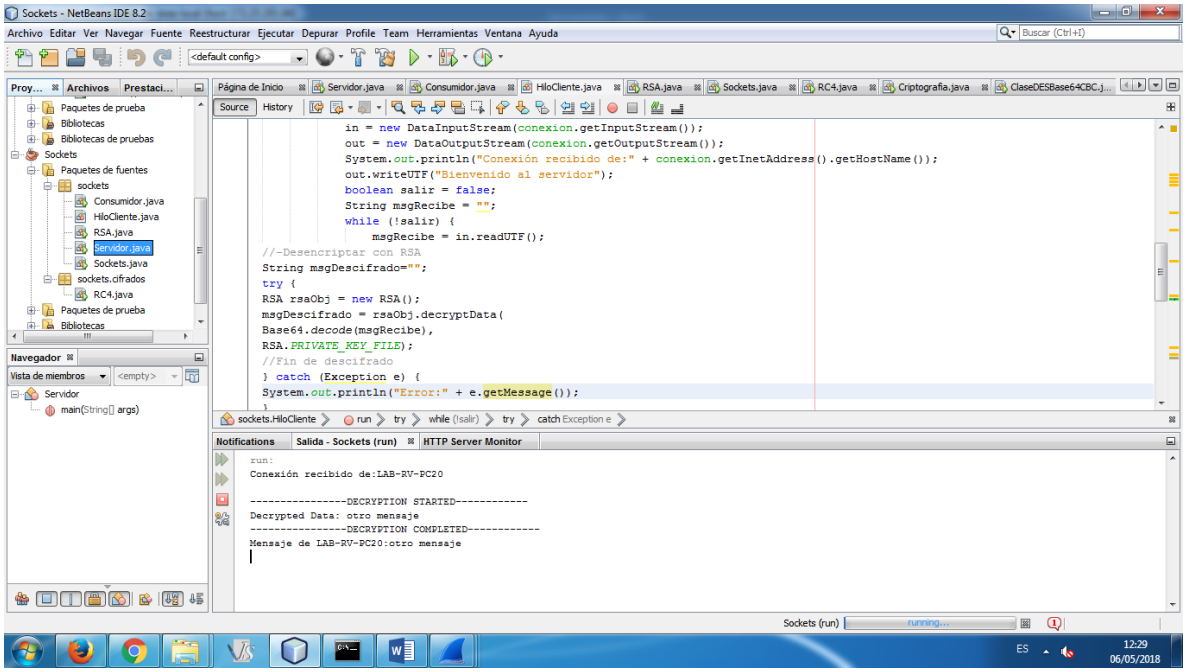
   a. Incluir el código fuente del cliente y servidor, capturar pantallas de la ejecución.






criptografia - NetBeans IDE 8.2
Archivo Editar Ver Navegar Fuente Reestructurar Ejecutar Depurar Profile Team Herramientas Ventana Ayuda
Buscar (Ctrl+I)
Proy... Archivos Prestaci...
Paquetes de prueba
Bibliotecas
Bibliotecas de pruebas
Sockets
Paquetes de fuentes
sockets
Consumidor.java
HiloCliente.java
RSA.java
Servidor.java
Sockets.java
sockets.cifrados
RC4.java
Paquetes de prueba
Bibliotecas
Navegador
Vista de miembros
Criptografia
main(String[] args)
Página de Inicio
Servidor.java
Consumidor.java
HiloCliente.java
RSA.java
Sockets.java
RC4.java
Criptografia.java
ClaseDESBase64CBC.j...
Source
History
/*
* To change this license header, choose License Headers in Project Properties.
* To change this template file, choose Tools | Templates
* and open the template in the editor.
*/
package criptografia;
import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Paths;
import org.bouncycastle.jcajce.provider.digest.SHA3;
import org.bouncycastle.util.encoders.Hex;
/**
*
* @author FIE
*/
public class Criptografia {
Notifications
Salida - Sockets (run)
HTTP Server Monitor
----------DECRYPTION STARTED----------
Decrypted Data: Saludos amigos
----------DECRYPTION COMPLETED----------
Mensaje de LAB-RV-PC20:Saludos amigos
----------DECRYPTION STARTED----------
Decrypted Data: saludos cordiales
----------DECRYPTION COMPLETED----------
Mensaje de LAB-RV-PC20:saludos cordiales
Sockets (run)
running...
ES
12:18
06/05/2018
Wireshark · Packet 34 · wireshark_723FD07D-60BB-44F9-BC40-392E1A994216_20180506122056_a03236.pcapng
Frame 34: 400 bytes on wire (3200 bits), 400 bytes captured (3200 bits) on interface 0
Ethernet II, Src: HewlettP_00:34:1b (24:be:05:00:34:1b), Dst: HewlettP_ff:7a:5f (80:c1:6e:ff:7a:5f)
Internet Protocol Version 4, Src: 172.25.201.84, Dst: 172.25.201.90
Transmission Control Protocol, Src Port: 51148, Dst Port: 8189, Seq: 1, Ack: 1, Len: 346
Data (346 bytes)
Data: 01585755716e6453315a35416f2f33515435506162524f46...
[Length: 346]
0030 00 fd 55 7f 00 00 01 58 57 55 71 6e 64 53 31 5a ··U····X WUqndS1Z
0040 35 41 6f 2f 33 51 54 35 50 61 62 52 4f 46 31 6d 5Ao/3QT5 PabROF1m
0050 6d 4e 64 33 68 70 5a 6c 58 30 4b 4d 48 59 79 45 mNd3hpZl X0KMHYyE
0060 65 39 52 57 31 54 36 75 68 4b 79 77 39 4f 7a 71 e9RW1T6u hKyw9Ozq
0070 4c 2f 34 78 48 55 74 50 70 7a 36 63 72 37 73 43 L/4xHUtP pz6cr7sC
0080 55 57 51 31 44 6f 66 70 41 7a 70 46 75 33 43 42 UWQ1Dofp AzpFu3CB
0090 37 30 37 59 79 54 54 75 72 4c 48 4b 67 58 4d 71 707YyTTu rLHKgX^q
00a0 71 77 39 31 75 76 4b 50 39 73 50 6e 64 68 55 4d qw91uvKP 9sPndhUM
00b0 50 38 72 41 4f 59 69 55 6c 51 43 4f 67 2f 32 5a P8rAOYiU lQCOg/2Z
00c0 73 72 42 53 37 66 65 73 4c 67 7a 74 73 70 79 36 srBS7fes Lgztspy6
00d0 55 62 4e 70 62 2b 54 4b 76 78 30 4a 69 65 72 45 UbNpb+TK vx0JierE
00e0 63 44 59 54 70 6f 70 62 4a 51 67 63 37 41 71 37 cDYTpopb JQgc7Aq7
00f0 77 32 44 71 53 4f 47 58 72 65 68 65 51 49 46 68 w2DqSOGX reheQIFh
0100 6e 73 32 6b 32 4f 73 66 37 31 63 6e 66 39 4d 4a ns2k2Osf 71cnf9MJ
0110 31 51 36 76 50 64 30 35 50 38 5a 47 4e 75 66 69 1Q6vPd05 P8ZGNufi
0120 6c 58 43 64 61 54 6d 53 46 2b 53 63 4d 4b 6e 39 lXCdaTmS F+ScMKn9
0130 2f 6f 6d 42 56 30 63 31 46 44 4a 56 55 73 74 37 /omBV0c1 FDJVUst7
0140 37 2f 34 73 32 5a 33 76 42 31 66 5a 72 74 61 75 7/4s2Z3v B1fZrtau
0150 67 36 77 58 2f 55 68 50 58 6d 64 6b 38 54 68 33 g6wX/UhP Xmdk8Th3
0160 41 54 41 53 6f 52 70 36 5a 75 42 4e 61 34 68 6a ATASoRp6 ZuBNa4hj
0170 7a 33 4f 44 56 6f 45 6b 72 39 45 47 44 4f 35 48 z3ODVoEk r9EGDO5H
0180 34 73 45 5a 69 43 70 75 4c 57 6e 73 54 41 3d 3d 4sEZiCpu LWnsTA==
Cerrar
Ayuda
ES
12:22
06/05/2018

Descifrando con la clave publica confidencialidad



4. Conclusiones y Recomendaciones

Las funciones hash son muy usadas, una de las utilidades que tiene es proteger la confidencialidad de una contraseña, ya que podría estar en texto plano y ser accesible por cualquiera y aún así no poder ser capaces de deducirla.

Es recomendable saber si una contraseña que está guardada, por ejemplo, en una base de datos es igual a la que hemos introducido no se descifra el hash ya que debería de ser imposible hacerlo, sino que se aplicará la misma función de resumen a la contraseña que especificamos y se comparará el resultado con el que tenemos guardado

5. Bibliografía

https://repositorioeva.espoch.edu.ec/pluginfile.php/379691/mod_resource/content/1/unidad6.pdf