**Experiment 5**

**Web and Mobile Security(CSP-338)**

**Aim:** Write a program to generate message digest for the given message using the SHA/MD5 algorithm and verify the integrity of message.

**Software/Hardware Requirements:**

window 7 and above version

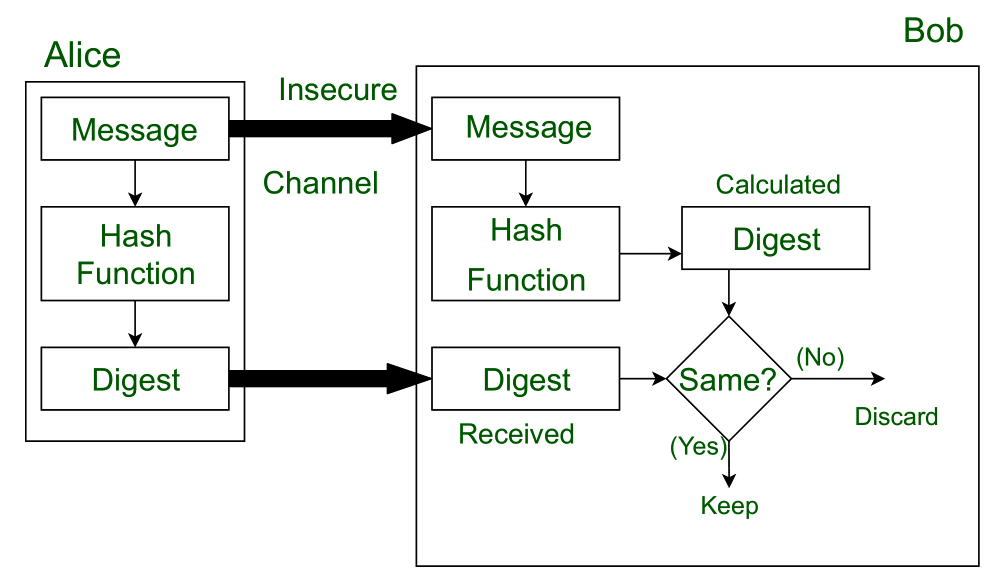
**Tools to be used:**

1. Eclipse IDE
2. JDK (Java Development kit)
3. IntelliJ IDEA

INTRODUCTION

**Message Digest** is used to ensure the integrity of a message transmitted over an insecure channel (where the content of the message can be changed). The message is passed through a [Cryptographic hash function](https://www.geeksforgeeks.org/passwords-and-cryptographic-hash-function/). This function creates a compressed image of the message called **Digest**.

**Message Digest** is used to ensure the integrity of a message transmitted over an insecure channel (where the content of the message can be changed). The message is passed through a [Cryptographic hash function](https://www.geeksforgeeks.org/passwords-and-cryptographic-hash-function/). This function creates a compressed image of the message called **Digest**.

Lets assume, Alice sent a message and digest pair to Bob. To check the integrity of the message Bob runs the cryptographic hash function on the received message and gets a new digest. Now, Bob will compare the new digest and the digest sent by Alice. If, both are same then Bob is sure that the original message is not changed

This message and digest pair is equivalent to a physical document and fingerprint of a person on that document. Unlike the physical document and the fingerprint, the message and the digest can be sent separately.

* Most importantly, the digest should be unchanged during the transmission.
* The cryptographic hash function is a one way function, that is, a function which is practically infeasible to invert. This cryptographic hash function takes a message of variable length as input and creates a **digest / hash / fingerprint** of fixed length, which is used to verify the integrity of the message.
* Message digest ensures the integrity of the document. To provide authenticity of the message, digest is encrypted with sender’s private key. Now this digest is called digital signature, which can be only decrypted by the receiver who has sender’s public key. Now the receiver can authenticate the sender and also verify the integrity of the sent message.

**Example:**  
The hash algorithm MD5 is widely used to check the integrity of messages. MD5 divides the message into blocks of 512 bits and creates a 128 bit digest(typically, 32 Hexadecimal digits). It is no longer considered reliable for use as researchers have demonstrated techniques capable of easily generating MD5 collisions on commercial computers.

**In response to the insecurities of MD5 hash algorithms, the**[**Secure Hash Algorithm (SHA)**](https://www.geeksforgeeks.org/sha-1-hash-in-java/)**was invented.**

**Steps/Method/Coding:**

To calculate cryptographic hashing value in Java, **MessageDigest** Class is used, under the package java.security.

MessageDigest Class provides following cryptographic hash function to find hash value of a text as follows:

* MD2
* MD5
* SHA-1
* SHA-224
* SHA-256
* SHA-384
* SHA-512

1.This Algorithms are initialize in static method called **getInstance()**.

2. After selecting the algorithm it calculate the **digest** value and return the results in byte array.

3. BigInteger class is used, which converts the resultant byte array into its **sign- magnitude representation**.

4.This representation is then converted into a hexadecimal format to get the expected MessageDigest.

Examples:

**Input :** hello world

**Output :** 5eb63bbbe01eeed093cb22bb8f5acdc3

**Input :** GeeksForGeeks

**Output :** e39b9c178b2c9be4e99b141d956c6ff6

**Coding (MD5 algorithm)**

import java.math.BigInteger;

import java.security.MessageDigest;

import java.security.NoSuchAlgorithmException;

// Java program to calculate MD5 hash value

public class MD5 {

public static String getMd5(String input)

{

try {

// Static getInstance method is called with hashing MD5

MessageDigest md = MessageDigest.getInstance("MD5");

// digest() method is called to calculate message digest

// of an input digest() return array of byte

byte[] messageDigest = md.digest(input.getBytes());

// Convert byte array into signum representation

BigInteger no = new BigInteger(1, messageDigest);

// Convert message digest into hex value

String hashtext = no.toString(16);

while (hashtext.length() < 32) {

hashtext = "0" + hashtext;

}

return hashtext;

}

// For specifying wrong message digest algorithms

catch (NoSuchAlgorithmException e) {

throw new RuntimeException(e);

}

}

// Driver code

public static void main(String args[]) throws NoSuchAlgorithmException

{

String s = "GeeksForGeeks";

System.out.println("Your HashCode Generated by MD5 is: " + getMd5(s));

}

}

References:

1. [Java Docs for MessageDigest](https://docs.oracle.com/javase/7/docs/api/java/security/MessageDigest.html)

**2.** [**MD5 hash in Java**](https://www.geeksforgeeks.org/md5-hash-in-java/)

**Coding (SHA algorithm)**

// Java program to calculate SHA-1 hash value

import java.math.BigInteger;

import java.security.MessageDigest;

import java.security.NoSuchAlgorithmException;

public class GFG {

public static String encryptThisString(String input)

{

try {

// getInstance() method is called with algorithm SHA-1

MessageDigest md = MessageDigest.getInstance("SHA-1");

// digest() method is called

// to calculate message digest of the input string

// returned as array of byte

byte[] messageDigest = md.digest(input.getBytes());

// Convert byte array into signum representation

BigInteger no = new BigInteger(1, messageDigest);

// Convert message digest into hex value

String hashtext = no.toString(16);

// Add preceding 0s to make it 32 bit

while (hashtext.length() < 32) {

hashtext = "0" + hashtext;

}

// return the HashText

return hashtext;

}

// For specifying wrong message digest algorithms

catch (NoSuchAlgorithmException e) {

throw new RuntimeException(e);

}

}

// Driver code

public static void main(String args[]) throws

NoSuchAlgorithmException

{

System.out.println("HashCode Generated by SHA-1 for: ");

String s1 = "GeeksForGeeks";

System.out.println("\n" + s1 + " : " + encryptThisString(s1));

String s2 = "hello world";

System.out.println("\n" + s2 + " : " + encryptThisString(s2));

}

}

**Output screenshot:**

**OUTPUT using MD5**

****

****

**Learning Outcomes:**

Output is often known as hash values, hash codes, message digest. The length of output hashes is generally less than its corresponding input message length.