Experiment No. 02

Title

Calculate the message digest of a text using the MD5 algorithm in JAVA.

Objective

Learn working of MD5 algorithm.

Problem Definition

Message digest of a text using the MD5 algorithm in JAVA.

Outcome

After completion of this assignment students will be able to understand how to Calculate the message digest of a text using the MD5 algorithm.

Software and Hardware Requirements

Operating system: 64-bit Windows OS and Linux

RAM :2GB RAM (4GB preferable)

IDE: Java Development Kit (JDK), Eclipse, Netbeans, or any other Java editor etc

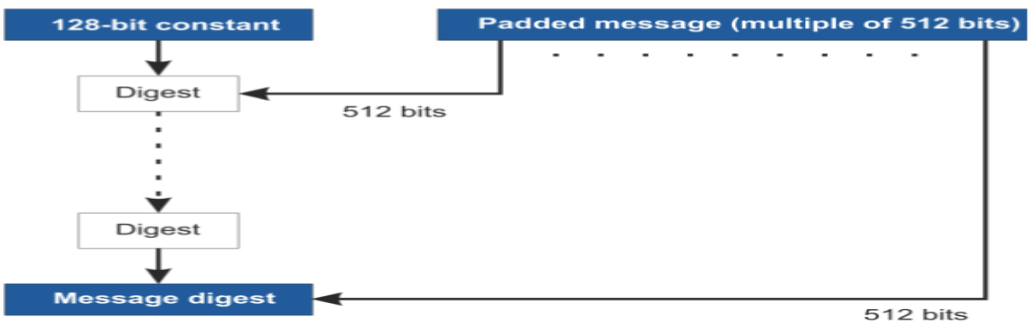
Theory

MD5 is a cryptographic algorithm that provides the hash functions to get a fixed length 128-bit (16 bytes) **hash value**. Using Java, we can implement the MD5 hash in an application by using the MessageDigest class which is defined in **java.security package**. The Java MessageDigest class provides the following types of hash algorithms:

1. MD5 2. SHA-1 3.SHA-256

The MD5 algorithm is proposed for digital signature app where a large file must be compressed in a secured manner before being encrypted with a private key under a public key cryptosystem like RSA.

Here, a point to note that two different inputs may produce the same value because it is not collision-resistant. It is advisable that do not to use the MD5 hash algorithm for many security-related cryptographic tasks.



MD5 Algorithm Structure

The MD5 algorithm is implemented in a predefined method named **getInstance().** Once we select the algorithm, it calculates the digest value and returns a byte array as a result.

```
MessageDigest md = MessageDigest.getInstance("MD5");
byte[] result = md.digest(input);
```

In order to convert the resultant byte array into its sign magnitude representations, use **BigInteger** class.

This representation converts into hex format to get the MessageDigest.

For example,

Input: hello world

Output: 5eb63bbbe01eeed093cb22bb8f5acdc3

In comparison to other digest algorithms, MD5 is simple, secure, efficient, and easy to implement. It provides a fingerprint or message digest arbitrary length. Its performance is fast on a 32-bit machines.

Advantages of MD5

- Easy to compare small hashes.
- Low resource consumption
- Storing password is convenient.
- Integrity check cannot be tampered with.

How MD5 algorithm works?

The MD5 algorithm includes the following four steps:

- Append Extra Bits (padding bits)
- Append Length
- Create and Initialize MD buffer
- Process Message in 16-word Block

Step1: Append Extra Bits (Padding Bits)

It is the initial step of the algorithm. In this step, we append padding bits (extra bits) to the given message or string. Because of this, the length of the original message or string corresponds to 418 modulo 512. The reason to append bits is that the length must be the multiple of 512 bits length.

Note that padding is also done if the original message is congruent to 448 modulo 512. In the padding bits, the first bit is 1 and the rest of the bits are 0.

Step 2: Append Length

After doing the padding, append length by adding 64-bits at the end. It records the length of the input given by the user. It gives the resulting message of the length that is multiple of 512 bit.

Step 3: Create and Initialize MD buffer

MD buffer is a four-word (A, B, C, D) buffer in which each word is 32-bit register. It is used to compute the value of the message digest. Each word is initialized in the following way:

A	01	23	45	67
B	89	Ab	Cd	Ef
C	Fe	Dc	Ba	98
D	76	54	32	10

Step 4: Process Message in 16-word Block

The MD5 algorithm uses the auxiliary functions that accept the inputs as three 32-bit numbers and produces 32-bits as result. The auxiliary function uses the three operators like OR, XOR, and NOR, as shown below.

F(X, Y, Z)	XY v not (X)Z
G(X, Y, Z)	XZ v Y not (Z)
H(X, Y, Z)	X xor Y xor Z
I(X, Y, Z)	Y xor (X v not (Z))

There are performed using the 16 basic operations in which the content of four buffers is mixed with the input using the auxiliary buffer.

Conclusion

Thus we learn that how to Calculate the message digest of a text using the MD5 algorithm in JAVA.