# One-Way Hash Function
# Computer Security IV1013

Axel Månson Lokrantz

2023-04-17

## Task 2.1: Generating Message Digest and MAC

### What differences do you see between the algorithms?

MD5 uses a 128-bit hash value. The number of hexvalues are equal to 32.
$32 * 4 = 128$ bits.
SHA1 uses 160-bit hash value. The number of hexvalues are equal to 40.
$40 * 4 = 160$ bits.
SHA256 uses 256-bit hash value. The number of hexvalues are equal to 64.
$64 * 4 = 256$ bits.

### Write down the digests generated using the three algorithms.

The following hash values were created with the the plaintext "axelml@kth.se".

MD5 = e0e817367b305c6c75eae3d9d2d7456f
SHA1 = a1f82c95d357d414f0d7ac2d3a2d9932f3a49b76SHA256
SHA256 = 4b21a9cd4eefea894d1442175c4c1b6299b498fc04034ee8a694d5fc00a4d622

## 2.2 Keyed Hash and HMAC

### Do we have to use a key with a fixed size in HMAC? If so, what is the key size? If not, why?

No, HMAC is designed to be used with keys of any length. If the key is longer than the hash function's block size, the key will be hashed first and then used as a key. If the key is shorter than the hash function's block size, the key will be padded with zeroes to match the block size of the hash function. It is recommended to use a key with a length that is at least equal to the underlying hash function's block size. Using a longer key does not necessarily make it more secure, and it might slow the algorithm down. On the other hand, using a shorter key could potentially make it susceptible to brute force attacks. The recommended key lengths for the three different hash functions are:

MD5 = 128 bits / 8 = 16 bytes.
SHA1 = 160 bits / 8 = 20 bytes.
SHA256 = 256 bits / 8 = 32 bytes.

### Now use the string IV1013-key as the secret key and write down the keyed hashes generated using the three algorithms.

The following hash values were created with the key ”IV1013-key” and the plaintext ”axelml@kth.se”.

HMAC-MD5 = 94172b6f066e9f4606f67a3c9c06ce48
HMAC-SHA1 = fded2b1e385f1269b2e4adcf9a177c60b43efc73
HMAC-SHA256 = 21f36f6a4d0b148b8529a74bf0b2ddb7dad54115c19e658fe2c314f592aee42e

### Count how many bits are the same between H1 and H2 for MD5 and SHA256 (writing a short program to count the same bits might help you). In the report, specfiy how many bits are the same.

To find the number of equal bits two hash values were generated using the two hash functions, md5 and sha256. H1 is the original file and H2 is the altered file where the first bit was swapped.

MD5_H1 = e0e817367b305c6c75eae3d9d2d7456f
MD5_H2 = 47c7f5fa806e409225db178f93d6a83f

The number of equal bits: 63.

SHA256_H1 = e0e817367b305c6c75eae3d9d2d7456f
SHA256_H2 = 47c7f5fa806e409225db178f93d6a83f

The number of equal bits: 133.

By changing a single bit in the plain text, about half of the the bits in the two hash values were altered.

**Investigate how many trials it will take to break the weak collision property using the bruteforce method. Below is a list of five messages. For each message, report how many trials it took before you could find a message with the same hash.**

Solution found after: 1812651 iterations.
Matching string: 000000000002A=o
Digest for the message "000000000002A=o", using SHA-256 is:
9c24cd22e186f0e6a405a9486dfaa7db7dcc38586e422a2fc93f06ab5a09e05d
Solution for "IV1013 security" is: 9c24cd

Solution found after: 7466868 iterations.
Matching string: 000000000008Xv
Digest for the message "000000000008Xv", using SHA-256 is:
ef9fa62121d05007945e032319e8553a78e0d0cb99fc9fac1c3052c136229afb
Solution for "Security is fun" is: ef9fa6

Solution found after: 10958446 iterations.
Matching string: 0000000C-M5
Digest for the message "0000000C-M5", using SHA-256 is:
0a16343ac345a8671d222a51a706271711379ffadcc0640be24de36136310793
Solution for "Yes, indeed" is: 0a1634