Hash Functions

Pr. NoureddineChikouche

University of M'sila

https://sites.google.com/view/chikouchenoureddine

Course outline

- Concepts and definitions
- MD-5 Function
- Categories of hash functions
- Applications
- Recommendations

- How to ensure integrity of the message?
- How to ensure authentication of an entity?

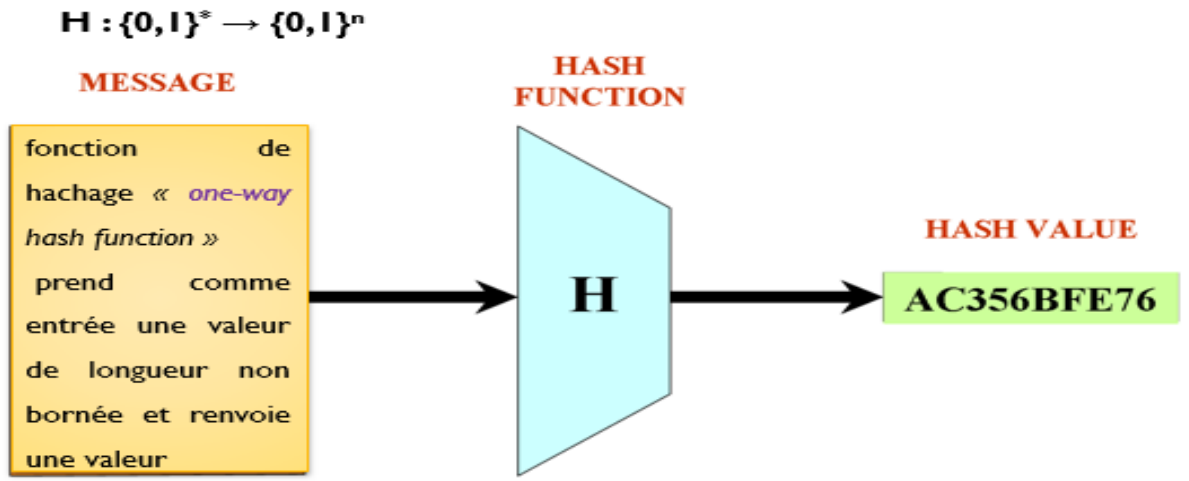
Concepts and definitions

Definition:

- Hash function is a cryptographic primitive that produces a fixed-size and condensed message from a non-fixed length message.

Concepts and definitions

- The output of the function is named: *digest*, *digest message*, fingerprint hash *or hash*.
- The length of the fingerprint hash depended on the type of the hash function.



Concepts and definitions

Original text: Computer Security
Hexadecimal:43 6f 6d 70 75 74 65 72 20 53 65 63 75 72 69 74 79

Hash F.	Fingerprint hash	n
MD5	F88c612e493e78ab649191e411da7e86	128
SHA-1	d678cf370ad77eb7f7d209d1311c160037423546	160
SHA-256	a3fa73779dd6d4c3fbe123d8ee1b18ed5702ab9a1ff4e5 6fc96393636062d3cf	256
RipeMD-128	c7a050c1dfdfb6bf83dca669a86a3429	128
SHA3-256	b8d60c6c40062dadbb6e551bbb8bc02afd5cac2fb24447 9979c330f2abcf817c	256

Concepts and definitions Applications

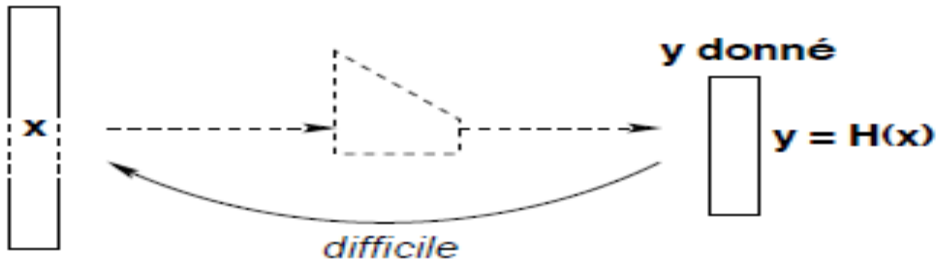
Hash functions are used, for example:

- Protection of passwords,
- Pseudo-random number generator,
- Message authentication code(MAC),
- Derive key ssession.
- Blockchain.
- Compress the messages in the generation and verification of the digital signature.

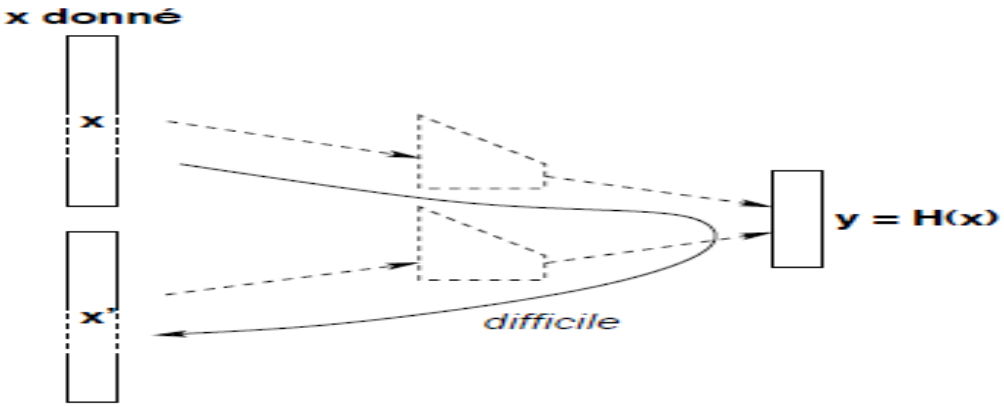
Concepts anddefinitions Properties

Properties:

1) Resistance to pre-image: Given $y=H(x)$, it is difficult to find x.

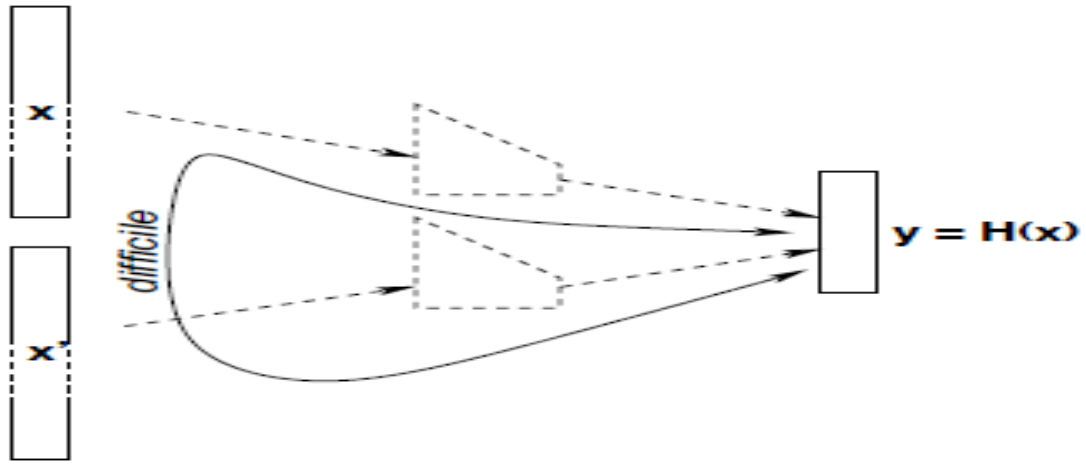


2) Resistanceto the second pre-image: Given x and H(x), it is hard to find $x \neq x'$ satisfying $H(x)=H(x')$.

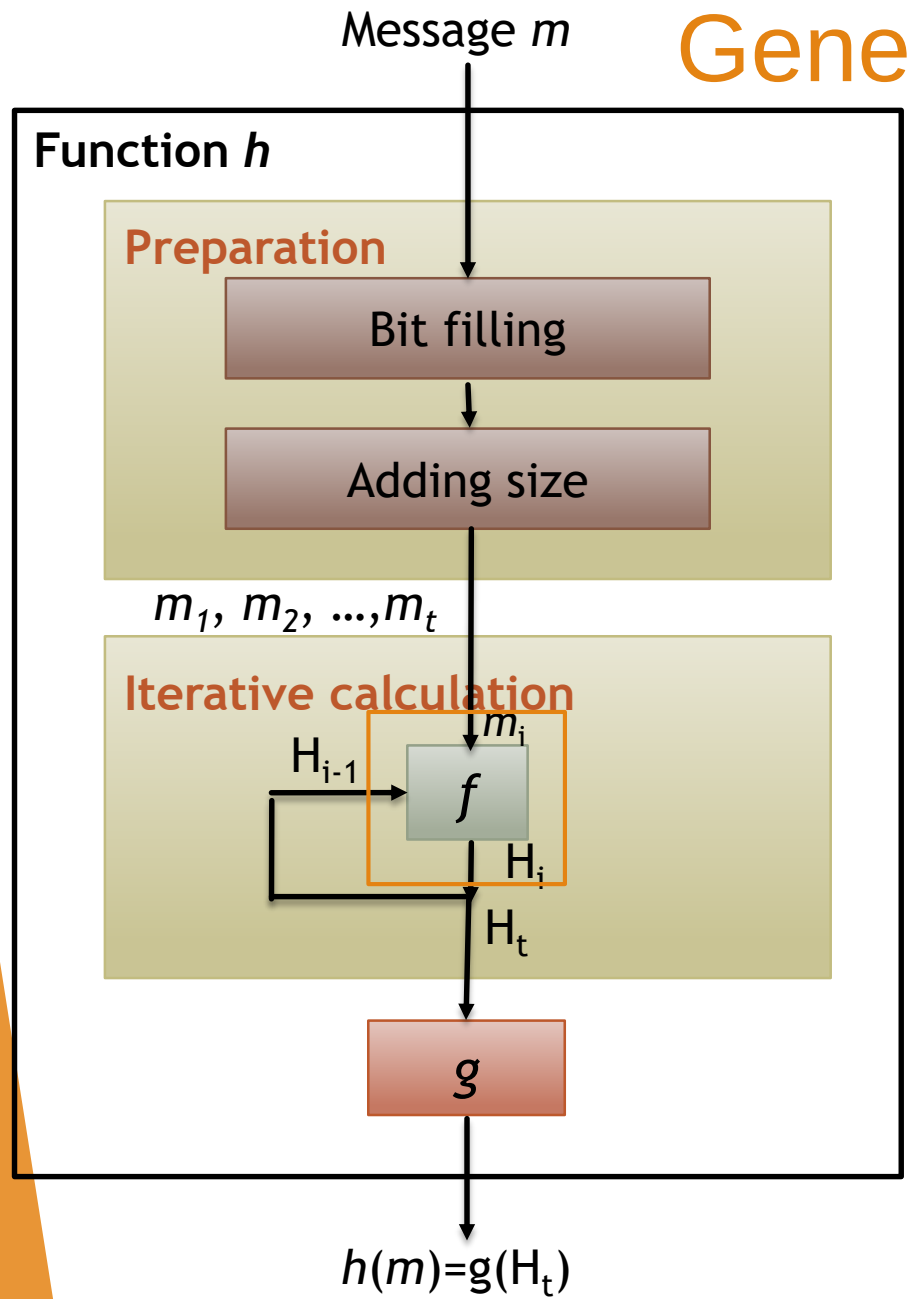


Concepts anddefinitions Properties

3) Collision resistance: it is difficult to find two different entries *x and x'* such as :*h (x) = h (x')*.



Concepts and definitions
General construction model



- Preparation:
  - Complete the message by bits. The extended message has a length multiple of block size.
  - Add the size of the input message
- Calculation iterative:
  - $H_i=f(H_{i-1}, m_i)$, $1 \le i \le t$
  - H_0=VI (initialization vector)
  - f: compression function
- Transformation result:
  - $H(m) = g(H_t)$
  - This is an optional transformation.

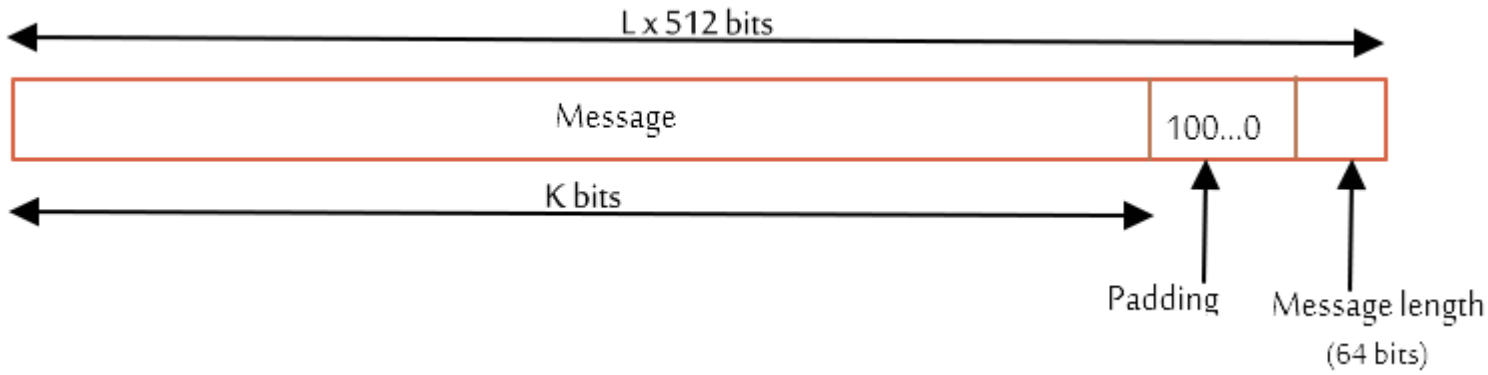
MD5 function

- MD5: Message Digest5
- Designed by Ronald Rivest in 1992
- MD4 Function Improvement.
- Hash size:128bits.
- Block size: 512 bits.
- IETF Standard: RFC1321.

MD5 function Algorithm

1. Preparation:

- Complete the message with an one (1) and sufficient of 0 (= 448 mod 512)
- The length of the extended message is a multiple of 512 bits.
- Divide each block into 16 sub-blocks Mi of 32bits.



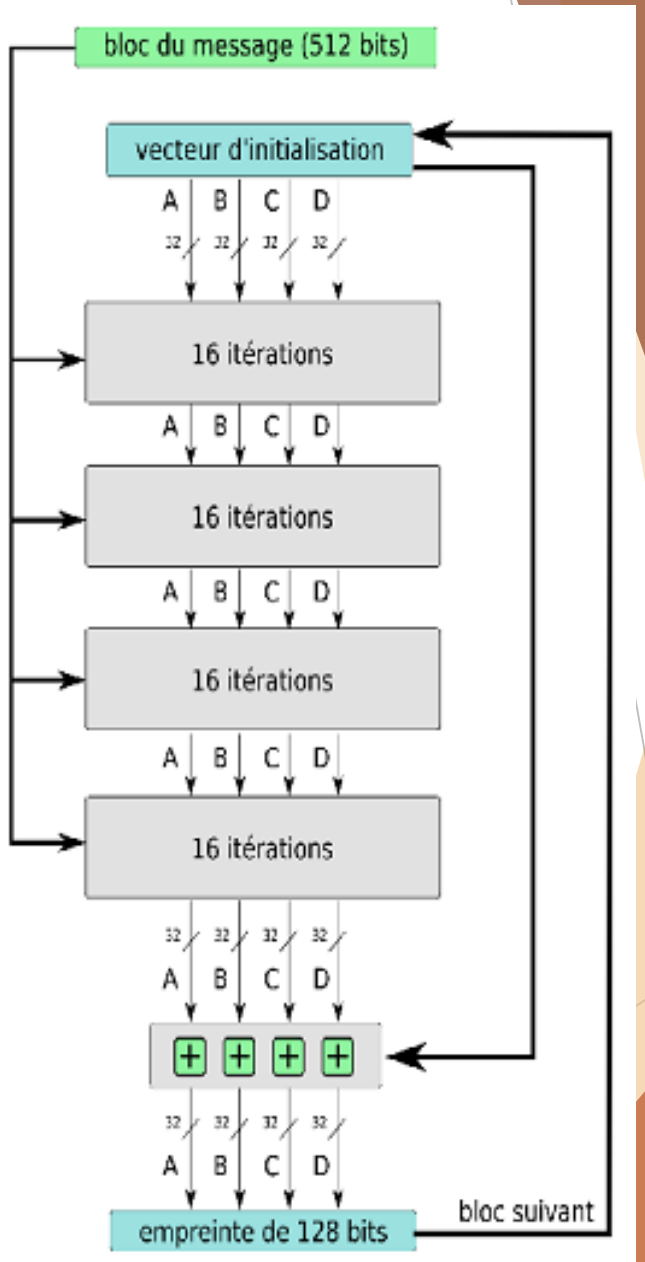
MD5 function Algorithm

2. Iterative calculation:

- Process the message in 512-bit blocks.
- 4 rounds of 16 operations (64 iterations) for a 512-bit block.
- The 128-bit initialization vector (VI)
- VI divided into 4 words (A, B, C, D) of size 32 bits.
- The result is used to initialize the following words.

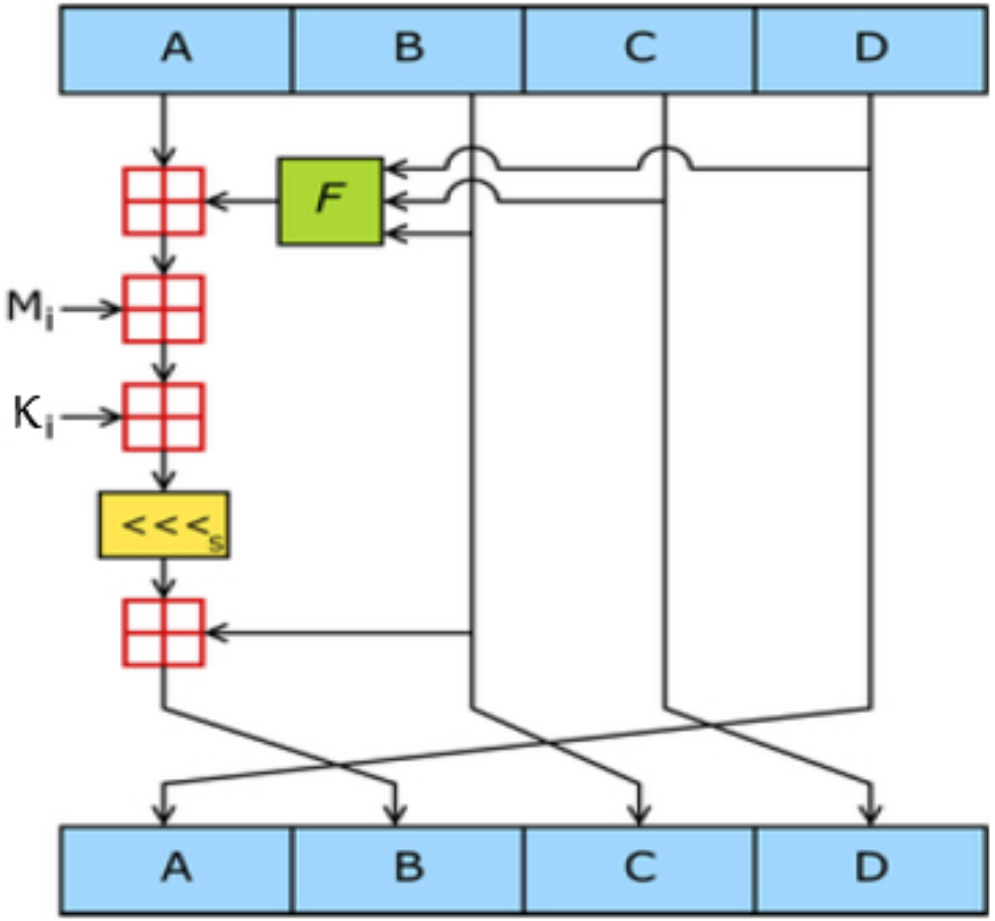
3. Result:

- The final result is obtained by concatenating A, B, C, and D.



Wikipedia

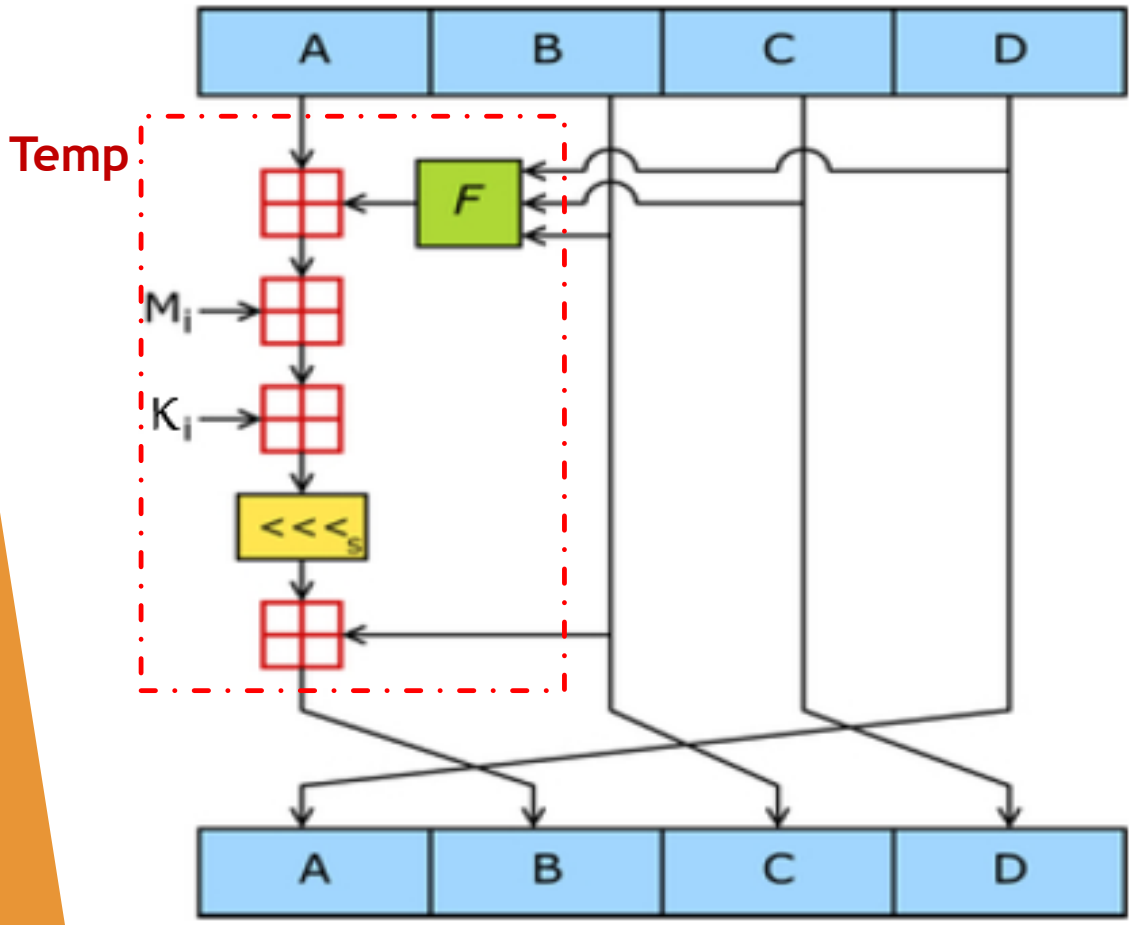
MD5 function
Algorithm: compression function



Wikipedia

- F: non-linear function, varies according to the round.
- Mi: sub-block of 32 bits.
- K_i : constant 32-bit,
- $\lll_s(.)$: rotation of *s* bits to the left.
- K_i *And* $<<<_s$ are different for each operation.

MD5 function Algorithm:functioncompression



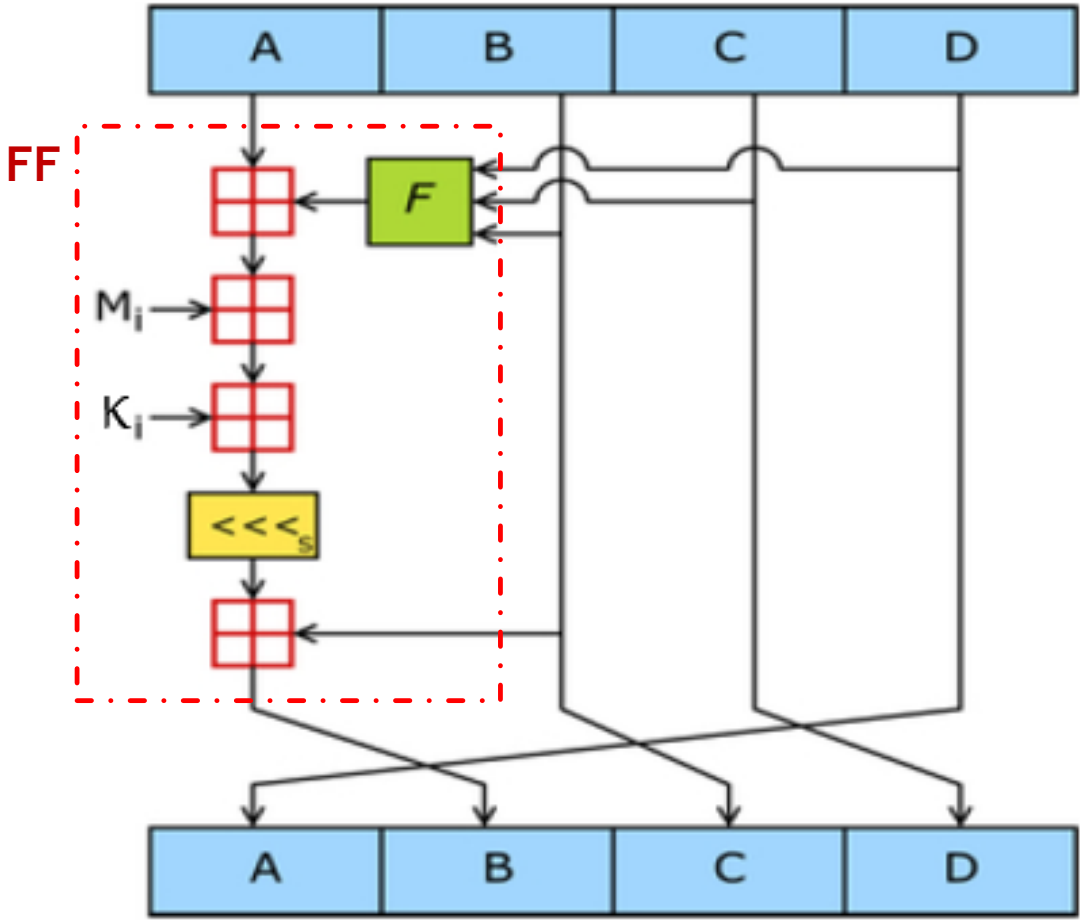
Function F according to the round:

- $F_1(B,C,D) = (B \wedge C) \vee (-B \wedge D)$
- $F_2(B,C,D) = (B \wedge D) \vee (C \wedge -D)$
- $F_3(B,C,D) = B \oplus C \oplus D$
- $F_4(B,C,D) = C \oplus (B \vee -D)$

The new values of VI:

- Temp= $\lll_s(M_i+F(B,\ C,\ D)+HAS+K_i)\ +B$
- A=D
- B=Temp
- C=B
- D=C

MD5 function
Example: execution of the first round



(A,B,C,D)=(D,FF, B, C)

FF (a, b, c, d, M(0),7, D76AA478)
FF(d, a, b, c, M(1),12, E8C7B756)
FF(c, d, a, b, M(2),17, 242070DB)
FF(b, c, d, a, M(3),22, C1BDCEEE)
FF(a, b, c, d, M(4),7, F757C0FAF)
FF(d, a, b, c, M(5),12, 4787C62A)
FF(c, d, a, b, M(6),17, A8304613)
FF(b, c, d, a, M(7),22, FD469501)
FF(a, b, c, d, M(8),7, 698098D8)
FF(d, a, b, c, M(9),12, 8B44F7AF)
FF(c, d, a, b, M(10),17, FFFF5BB1)
FF(b, c, d, a, M(11),22, 895CD7BE)
FF(a, b, c, d, M(12),7, 6B901122)
FF(d, a, b, c, M(13),12, FD987193)
FF(c, d, a, b, M(14),17, A679438E)
FF(b, c, d, a, M(15),22, 49B40821)

MD5 function Security

- In 2004, X. Wang et al. Have permits to find collisions on MD5. They detected an attack that was completely successful.
- MD5 is not recommended for applicationshigh security:
- MD5 is used by some applications that do not require a high level of security.

Categories of hash functions

1) Function secure hashing without key

- MDC: *Message Digest Code*.
- A hash function *H* which can be calculated without knowledge of a secret.
- Ensures integrity.
- The function is defined as follows: H(m)

Categories of hash functions

2) Secure hashing function with key

- HMAC: *Keyed-Hash Message Authentication Code*
- Ensures integrity and authentication
- is a type of message authentication code (MAC), calculated using a *hash function*in combination with a *secret key*.
- The security of HMAC is dependent on used hash function.
- Typical examples: HMAC-SHA-1 or HMAC-SHA-256

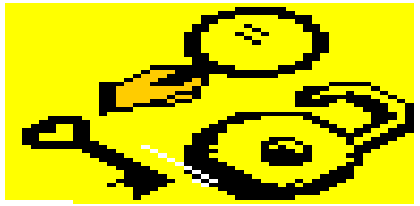
Categories hash functions

2) Secure hashing function with key

- The HMAC function is defined as follows::

$$HMAC_K(m) = H((K \oplus opad) \parallel H(K \oplus ipad) \parallel m))$$

  - H: hash function without key
  - k: private key completed with zeros
  - iPad: predefined by 0x363636...3636 (number of repetitions) of the bytes depending on the block size).
  - opad: predefinedby 0x5c5c5c...5c5c (number of byte repetitions depending on the block size).



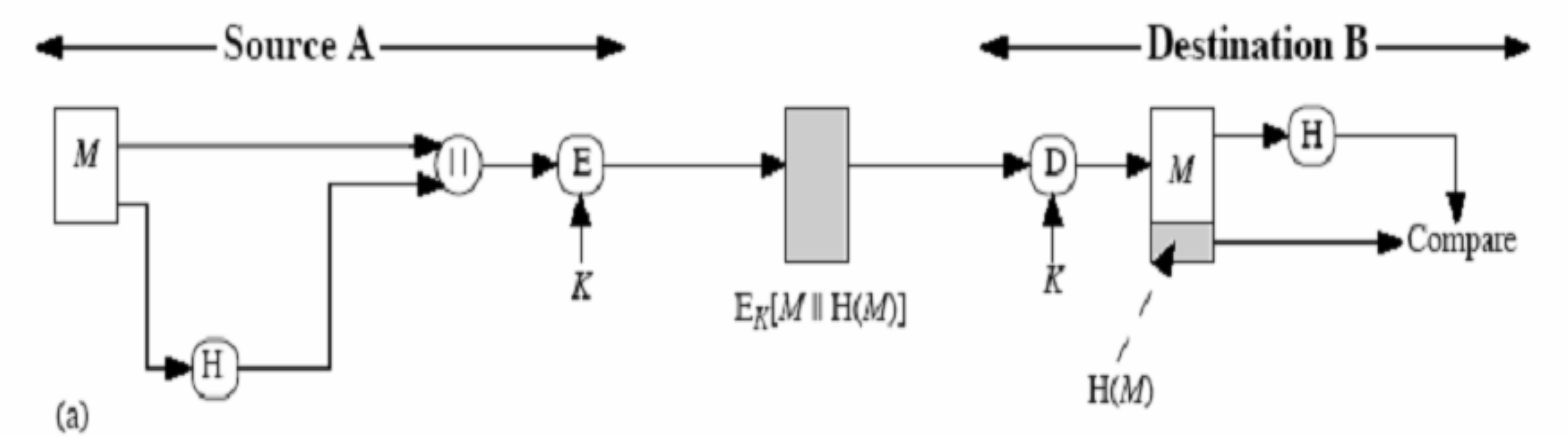
CrypTool

Individual.Procedure → Hash

-Hash Demonstration

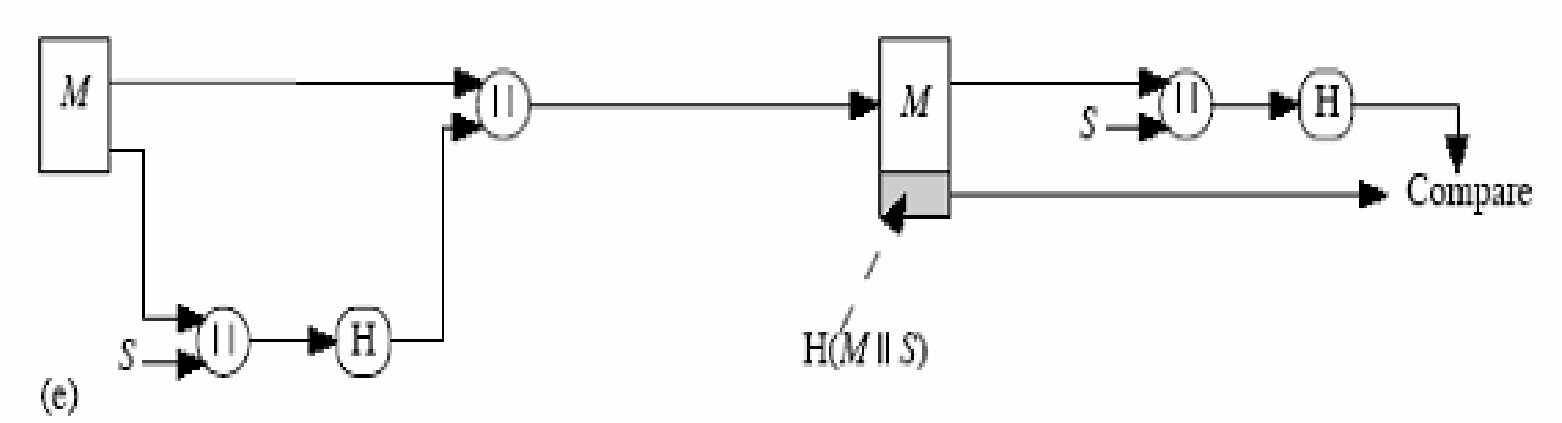
- Generation of HMAC

Example of uses (1)



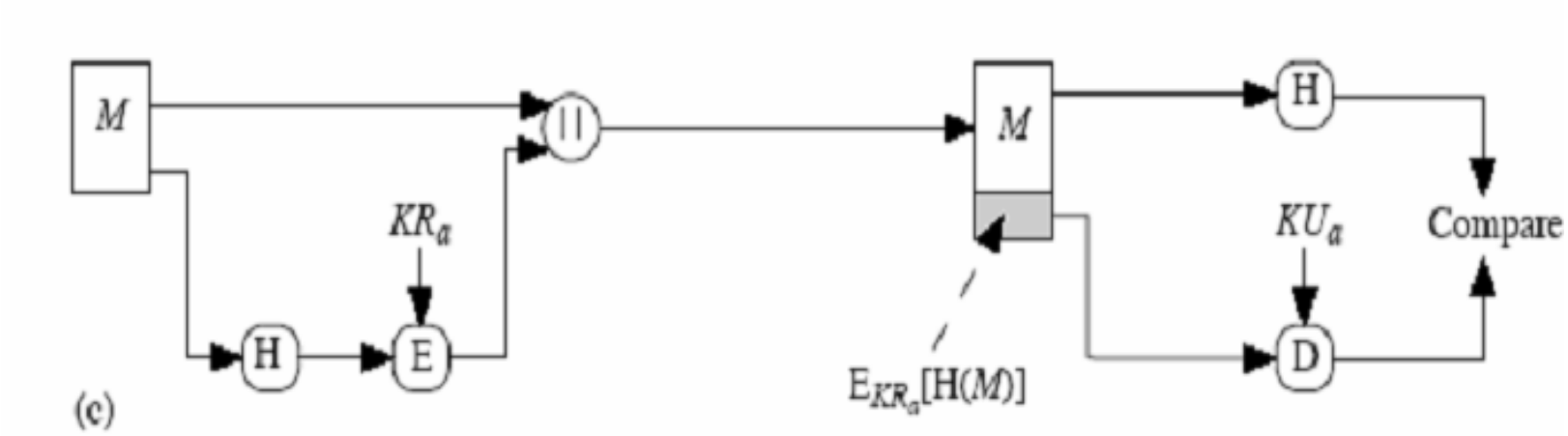
The message concatenated with a hash code is encrypted using symmetric encryption. Since only A and B share the secret key, the message must originate from A and have not been altered. The hash provides the structure required for authentication. Because encryption is applied to both the entire message and the hash code, confidentiality is also ensured.

Example uses (2)



- he technique assumes that the two communicating parties share a common secret value S.
- A calculates the hash value by concatenating M and S and adds the resulting hash value to M.
- Since B possesses S, it can recalculate the hash value to be checked. Because the secret value itself is not sent, an adversary cannot modify a stopped message and cannot produce a false message.

Example uses (3)



Only the hash code is encrypted, using public-key encryption and the sender's private key. This provides authentication. It also provides a digital signature, since only the sender could have produced the encrypted hash code.

Recommendations
Examples of hash functions

Hash Function	Date	Block	n	Author	Security
MD5	1992	512	128	RSA	-(2004)
RIPEMD	1992	512	128, 160,…	H.Dobbertin et al.	-(2004)
SHA-1	1995	512	160	NIST	-(2017)
SHA-512 (SHA-2)	2001	1024	SHA-512	NIST	+
SHA-3	2015	1152, 1088, 832, or 576	224, 256, 384 and 512	NIST	+

Recommendations

- NIST (2019)

Date	Niveau de Sécurité	Algorithme symétrique	Factorisation Module	Logarithme discret Clef	Logarithme discret Groupe	Courbe elliptique	Hash (A)	Hash (B)
Legacy [1]	80	2TDEA	1024	160	1024	160	SHA-1 [2]	
2019 - 2030	112	(3TDEA) [3] AES-128	2048	224	2048	224	SHA-224 SHA-512/224 SHA3-224	
2019 - 2030 et au-delà	128	AES-128	3072	256	3072	256	SHA-256 SHA-512/256 SHA3-256	SHA-1 KMAC128
2019 - 2030 et au-delà	192	AES-192	7680	384	7680	384	SHA-384 SHA3-384	SHA-224 SHA-512/224 SHA3-224
2019 - 2030 et au-delà	256	AES-256	15360	512	15360	512	SHA-512 SHA3-512	SHA-256 SHA-512/256 SHA-384 SHA-512 SHA3-256 SHA3-384 SHA3-512 KMAC256

Hash (A):Digital signatures and other uses requiring collision resistance.
Hash (B):HMAC, KMAC, key derivation functions and random number generation.

https://www.keylength.com/fr/4/

Quiz

- which hash functions are secured:
  - MD-5
  - SHA-3
  - SHA-1
  - SHA-512
- Hash functions are used to ensure confidentiality:
  - Yes
  - No