**Cryptographic Hash Function**

A cryptographic hash function is a mathematical algorithm that transforms an input (message) into a fixed-size string of bytes, typically a digest that is unique for every unique input. These functions are widely used in security applications like digital signatures, data integrity verification, and password storage.

**Properties of a Cryptographic Hash Function**

* Deterministic: For the same input, the function always produces the same output.
* Fixed Length Output: Regardless of the size of the input, the output (digest) is always of a fixed size.
* Efficiency: The hash should be quick to compute for any input.
* Pre-image Resistance: It should be computationally infeasible to reverse the hash function to retrieve the original input.
* Weak Collision Resistance: It should be infeasible to find another input that produces the same hash as a given input.
* Strong Collision Resistance: It should be infeasible to find two different inputs that produce the same hash value.
* Avalanche Effect: A slight change in the input drastically changes the output hash.

**Avalanche Effect**

The avalanche effect is a critical property of cryptographic hash functions. It ensures that:

* Even a small change in the input (e.g., flipping one bit) produces a completely different hash output.
* This property enhances security by making it practically impossible to infer anything about the input from the output.

*Example:*

*Input 1: hello*

*MD5 Hash: 5d41402abc4b2a76b9719d911017c592*

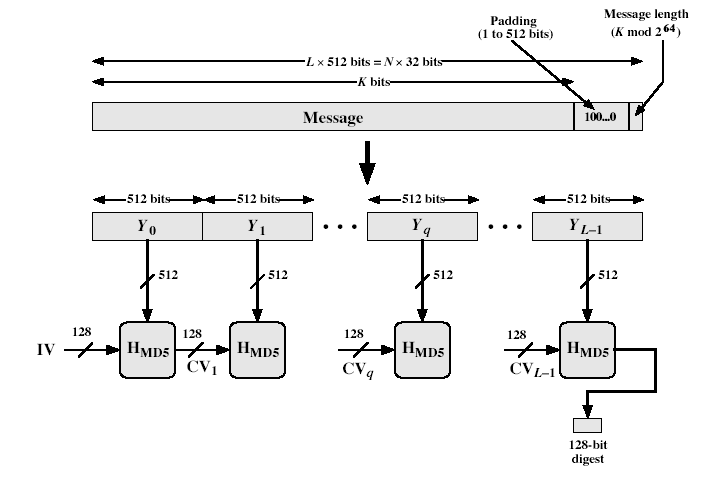
*Input 2: Hello (note the capitalization of 'H')*

*MD5 Hash: 8b1a9953c4611296a827abf8c47804d7*

The two outputs are vastly different due to the avalanche effect.

**Message Digest Algorithms**

**MD5 (Message Digest 5)**

****

Designed by: Ronald Rivest (1991).

Output Size: 128 bits.

Purpose: Initially developed for integrity checks but later found vulnerable.

Algorithm Steps:

* Padding: Extend the message so that its length is congruent to 448 mod 512.
* Appending Length: Append a 64-bit representation of the original message length.
* Initialize Buffers: Four 32-bit buffers (A, B, C, D) are initialized to fixed constants.
* Processing in Blocks: Process the message in 512-bit blocks over 64 iterations (four rounds of 16 operations).
* Output: The final buffer values form the 128-bit hash.

Weaknesses of MD5:

* Vulnerable to collision attacks.
* Not recommended for security-sensitive applications like SSL certificates.

**SHA-1 (Secure Hash Algorithm 1)**

Designed by: NIST & NSA (1993).

Output Size: 160 bits.

Purpose: Developed for secure cryptographic applications, including digital signatures.

Algorithm Steps:

* Padding: Similar to MD5, the message is padded to be 448 mod 512 in length.
* Appending Length: Append a 64-bit length representation.
* Initialize Buffers: Five 32-bit buffers (A, B, C, D, E) are initialized with specific constants.
* Processing: Operates on 512-bit blocks using 80 iterations grouped into four rounds.
* Output: The final buffer state forms the 160-bit hash.

Advantages of SHA-1 over MD5:

* Produces a longer hash (160 bits vs. 128 bits), offering more collision resistance.
* Designed to be more robust against cryptanalysis than MD5.

Weaknesses of SHA-1:

* SHA-1 has been deprecated due to vulnerabilities to collision attacks (e.g., in 2017, researchers demonstrated a practical collision attack).

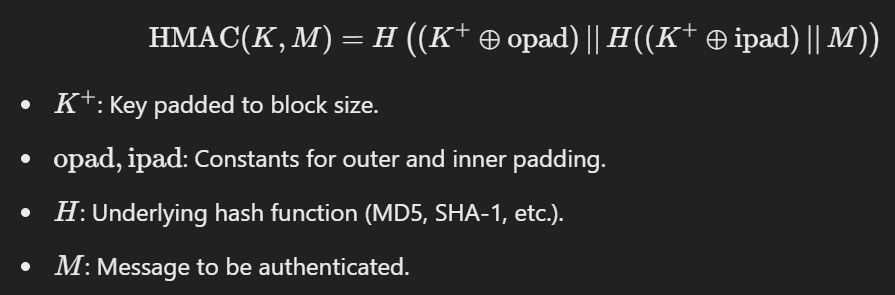
**Keyed Hash Functions as Message Authentication Code (MAC)**

Concept:

A MAC ensures both data integrity and authenticity using a secret key and a hash function.

**HMAC (Hash-based MAC):**

Formula:



Advantages of HMAC:

* Combines the security of hash functions with the secrecy of a key.
* Resistant to birthday attacks because of the keyed approach.

**Difference between MD5 and SHA-1**

