BCT 2314: CRYPTOGRAPHY AND COMPUTER SECURITY

Course Outline

1. Traditional and Modern cryptography and ciphers
2. Security in computing

- ➢ Encryption
- ➢ Decryption
- ➢ Public key cryptography

3. Security in computing environments

- ➢ Encryption
- ➢ Protocols
- ➢ Security programs

4. Operating systems, networks and communication

- Legal, Ethical and Human factors in computer security

Course Outline

1. **Introduction**
2. **Security in computing**

- Block Ciphers and Data Encryption Standards
- Block Cipher Standards
- The Data Encryption Standards
  - The DES Strength
- 3DES
- Differential and Linear Cryptanalysis
- Block Cipher Design Principles
- Advanced Encryption Standard
- Block Cipher Modes of operation
  - Electronic Code Block (ECB), Cipher Block Chaining (CBC), Cipher Feedback (CFB), Output Feedback (OFB), Counter (CTR)

3. **Public Key Cryptography**

- Elliptic Curve Arithmetic, Elliptic Curve Cryptography

4. **Message Authentication and Hash Functions**

- Message Authentication Codes (Covered)
- Hash Functions, Security of Hash Functions
- Secure Hash Algorithms i.e. SHA

5. **System and Network Security**

- Intruders, Malicious Software
- Port Scanning, Spoofs, Spam, DoS, Firewalls

6. **Legal, Ethical and Human factors in computer security**

Hash Functions

CRYPTOGRAPHIC HASH FUNCTIONS

Message Authentication

Goal: Having received a message one would like to make sure that the message has not been altered on the way

- Produce a short sequence of bits that depends on the message and on a secret key
- To authenticate the message, the partner will compute the same bit pattern, assuming he shares the same secret key

CRYPTOGRAPHIC HASH FUNCTIONS

Message Authentication

- This does not necessarily includes encrypting or signing the message
- The message can be sent in plain, with the authenticator appended
- This is not a digital signature: the receiver can produce the same MAC
- One may encrypt the authenticator with his private key to produce a digital signature
- One may encrypt both the message and the authenticator

CRYPTOGRAPHIC HASH FUNCTIONS

Message Authentication

Possible attacks on message authentication:

1. Content modification
2. Sequence modification: Modifications to a sequence of messages, including insertion, deletion, reordering
3. Timing modification : Delay or replay messages

CRYPTOGRAPHIC HASH FUNCTIONS

Message Authentication

- Three types of authentication exist

1. Message encryption – The ciphertext serves as authenticator
2. Message authentication code (MAC) – A public function of the message and a secret key producing a fixed-length value to serve as authenticator This does not provide a digital signature because A and B share the same key
3. Hash function – A public function mapping an arbitrary length message into a fixed-length hash value to serve as authenticator. This does not provide a digital signature because there is no key

CRYPTOGRAPHIC HASH FUNCTIONS

Message Authentication based on Hash Functions

- A fixed-length hash value h is generated by a function H that takes as input a message of arbitrary length: $h=H(M)$
  - A sends M and $H(M)$
  - B authenticates the message by computing $H(M)$ and checking the match

CRYPTOGRAPHIC HASH FUNCTIONS

Message Authentication based on Hash Functions

Requirements for a hash function

1. H can be applied to a message of any size
2. H produces fixed-length output
3. Computationally easy to compute H(M)
4. Computationally infeasible to find M such that H(M)=h, for a given h
5. Computationally infeasible to find M’ such that H(M’)=H(M), for a given M
6. Computationally infeasible to find M,M’ with H(M)=H(M’) (to resist birthday attacks)

CRYPTOGRAPHIC HASH FUNCTIONS

Message Authentication based on Hash Functions

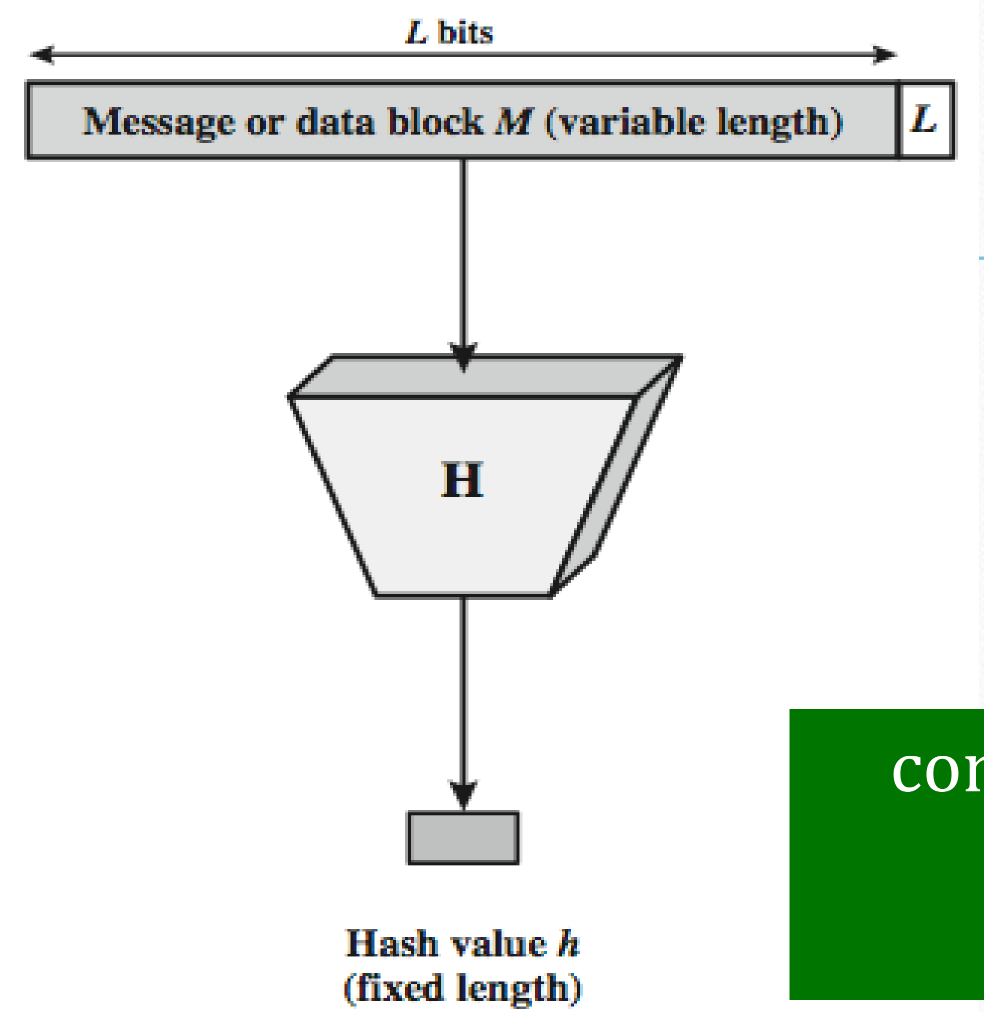
Points to note

- Note 1: The hash function is not considered secret – some other means are required to protect it
- Note 2: Hash function plus secrecy (key) gives a MAC – these are called HMACs

CRYPTOGRAPHIC HASH FUNCTIONS

Hashing function as “chewing” or “digest” function

Hash Function



- The hash value represents concisely the longer message
    - may called the *message diges*t

- A message digest is as a ``digital fingerprint'' of the original document

condenses arbitrary message to fixed size

$$h = H(M)$$

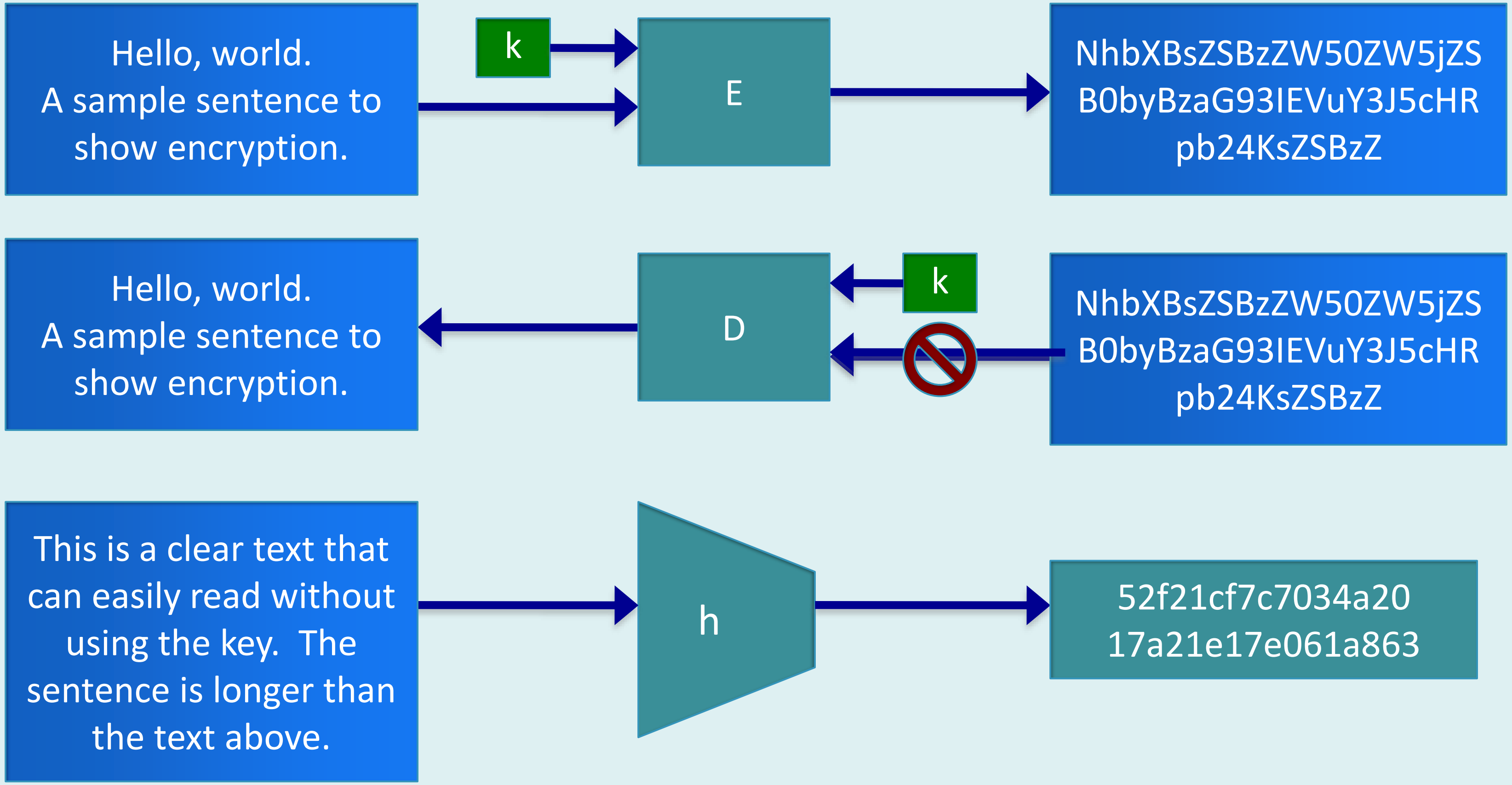
CRYPTOGRAPHIC HASH FUNCTIONS

Definition:- Hash Function

- Refers to a deterministic function which maps a bit string of arbitrary length to a hashed value which is a bit string of a fixed length.
- Let h denote a hash function whose fixed output length is denoted by $|h|$. Then h should have some specific properties:-

CRYPTOGRAPHIC HASH FUNCTIONS

Hashing Vs Encryption


Hello, world.
A sample sentence to show encryption.
k
E
NhbXBsZSBzZW50ZW5jZSB0byBzaG93IEVuY3J5cHRpb24KsZSBzZ
Hello, world.
A sample sentence to show encryption.
D
k
NhbXBsZSBzZW50ZW5jZSB0byBzaG93IEVuY3J5cHRpb24KsZSBzZ
This is a clear text that can easily read without using the key. The sentence is longer than the text above.
h
52f21cf7c7034a20
17a21e17e061a863

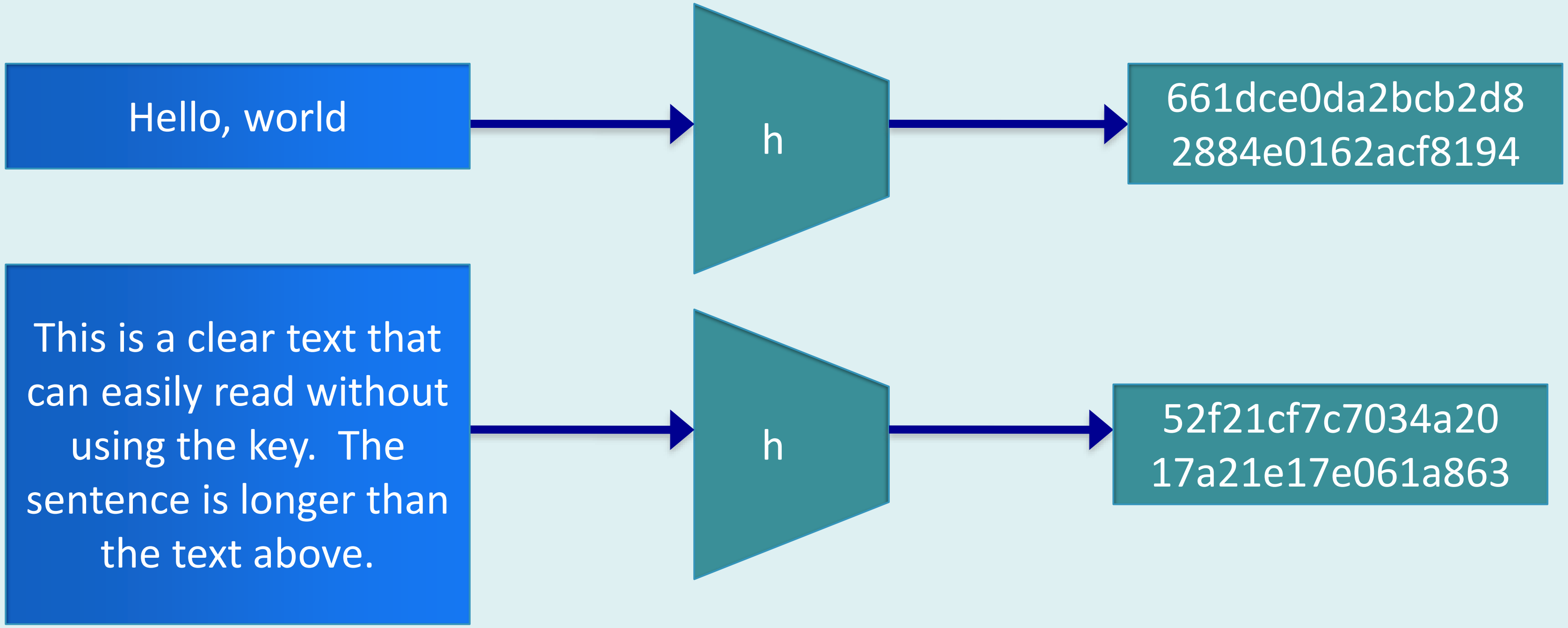
- Encryption is two way, and requires a key to encrypt/decrypt

- Hashing is one-way. There is no 'de-hashing’

CRYPTOGRAPHIC HASH FUNCTIONS

- Properties of a Hash Function

1. Fixed length output for arbitrary input


Hello, world
h
661dce0da2bcb2d8
2884e0162acf8194
This is a clear text that can easily read without using the key. The sentence is longer than the text above.
h
52f21cf7c7034a20
17a21e17e061a863

CRYPTOGRAPHIC HASH FUNCTIONS

- Properties of a Hash Function

1. Mixing transformation: On any input x, the output hashed value $h(x)$ should be computationally indistinguishable from a uniform binary string in the interval $[0,\ 2^{|h|}]$. (Property a*chieved using operations similar to those used in the design of block ciphers)*

- A “good” hash function has the property that the results of applying the function to a large set of inputs will produce outputs that are evenly distributed and apparently random.
- In general terms, the principal object of a hash function is data integrity. A change to any bit or bits in M results, with high probability, in a change to the hash code.

CRYPTOGRAPHIC HASH FUNCTIONS

Properties of a Hash Function

- The kind of hash function needed for security applications is referred to as a **cryptographic hash function.**

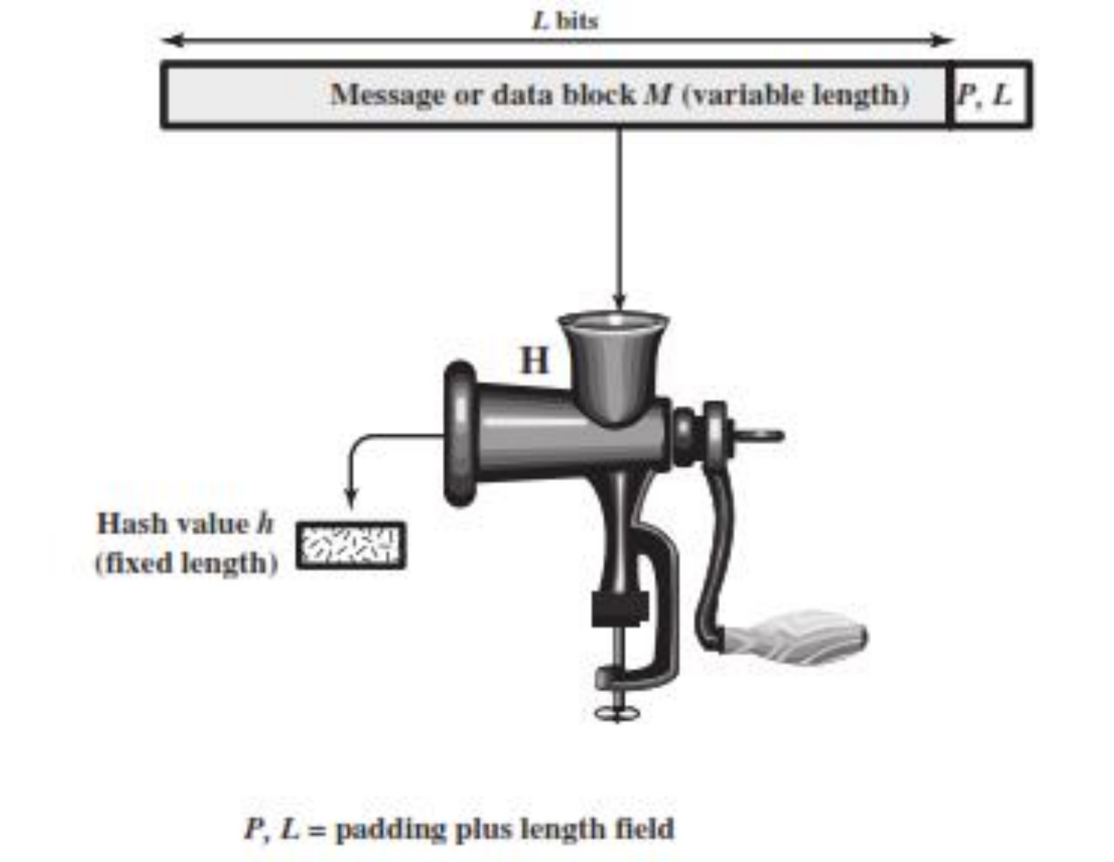
Definition: A cryptographic hash function is an algorithm for which it is computationally infeasible (because no attack is significantly more efficient than brute force) to find either

a) A data object that maps to a pre-specified hash result (the one-way property) or

b) Two data objects that map to the same hash result (the collision-free property). Because of these characteristics, hash functions are often used to determine whether or not data has changed.

CRYPTOGRAPHIC HASH FUNCTIONS

Cryptographic Hash Function h=H(M)



L bits
Message or data block M (variable length)
P, L
H
Hash value h
(fixed length)
P, L = padding plus length field

CRYPTOGRAPHIC HASH FUNCTIONS

- Properties of a Hash Function

2. Collision resistance: It should be computationally infeasible to find two inputs x, y with $\mathrm{x} \neq y$ such that $h(x) = h(y)$.

- For this to be reasonable, its necessary that the output space h of should be sufficiently large, its$(|h|)$ least value being 128 and typical value being 160. (*Property achieved using operations similar to those used in the design of block ciphers)*

CRYPTOGRAPHIC HASH FUNCTIONS

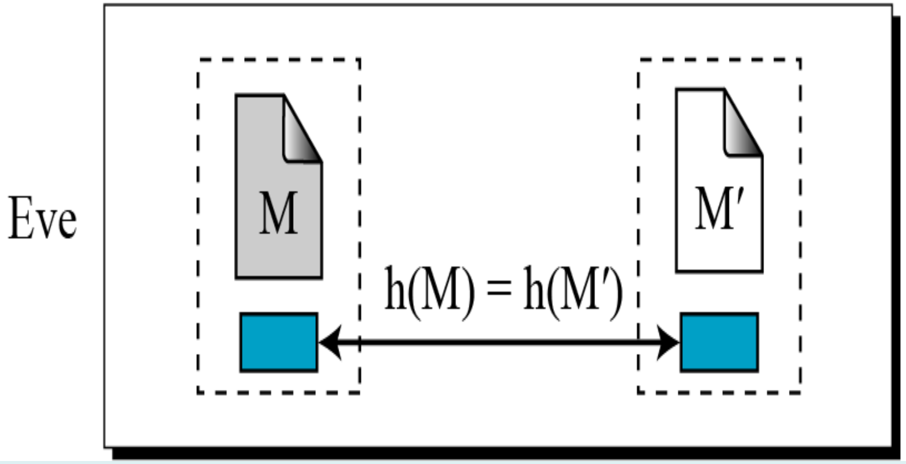
Collision resistance

Collision Attack

Given: none **Find: M′ ≠ M such that h(M) = h(M′)**

M: Message
Hash: Hash function
h(M): Digest

Find: M and M′ such that M ≠ M′, but h(M) = h(M′)



- Can’t find any two different messages with the same message digest
  - Collision resistance implies second preimage resistance
  - Collisions, if we could find them, would give signatories a way to repudiate their signatures

CRYPTOGRAPHIC HASH FUNCTIONS

- Properties of a Hash Function

3. Pre-image resistance: Is the one-way property: It is easy to generate a code given a message, but virtually impossible to generate a message given a code. Given a hashed value h it should be computationally infeasible to find an input string x such that $h = h(x)$. The space for h should be large. (*This property is achieved using data compression techniques.)*

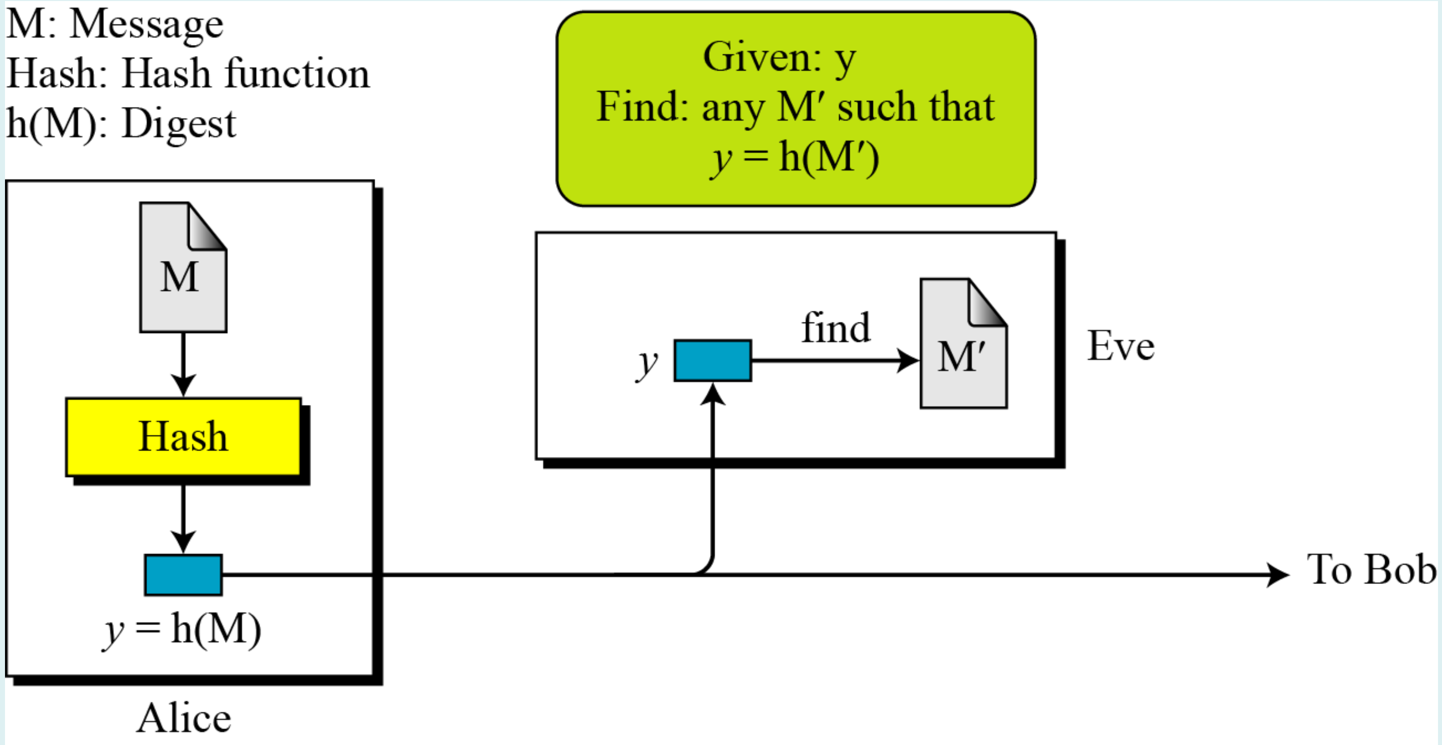
CRYPTOGRAPHIC HASH FUNCTIONS

3. Pre-image resistance:

- This measures how difficult to devise a message which hashes to the known digest . Roughly speaking, the hash function must be one-way.

Preimage Attack

Given: y = h(M) **Find: M′ such that y = h(M′)**



M: Message
Hash: Hash function
h(M): Digest

Given: y
Find: any M′ such that
$y = h(M')$

Given only a message digest, can’t find any message (or *preimage*) that generates that digest.

CRYPTOGRAPHIC HASH FUNCTIONS

Questions

- Can we use a conventional lossless compression method such as zip as a cryptographic hash function?

Answer : No, a lossless compression method creates a compressed message that is reversible.

- Can we use a checksum function as a cryptographic hash function?

Answer : No, a checksum function is not preimage resistant, Eve may find several messages whose checksum matches the given one.

CRYPTOGRAPHIC HASH FUNCTIONS

- Properties of a Hash Function

4. Second Pre-image resistance: Guarantees that it is impossible to find an alternative message with the same hash value as a given message. This prevents forgery when an encrypted hash code is used. If this property were not true, an attacker would be capable of the following sequence:

- First, observe or intercept a message plus its encrypted hash code; second, generate an unencrypted hash code from the message; third, generate an alternate message with the same hash

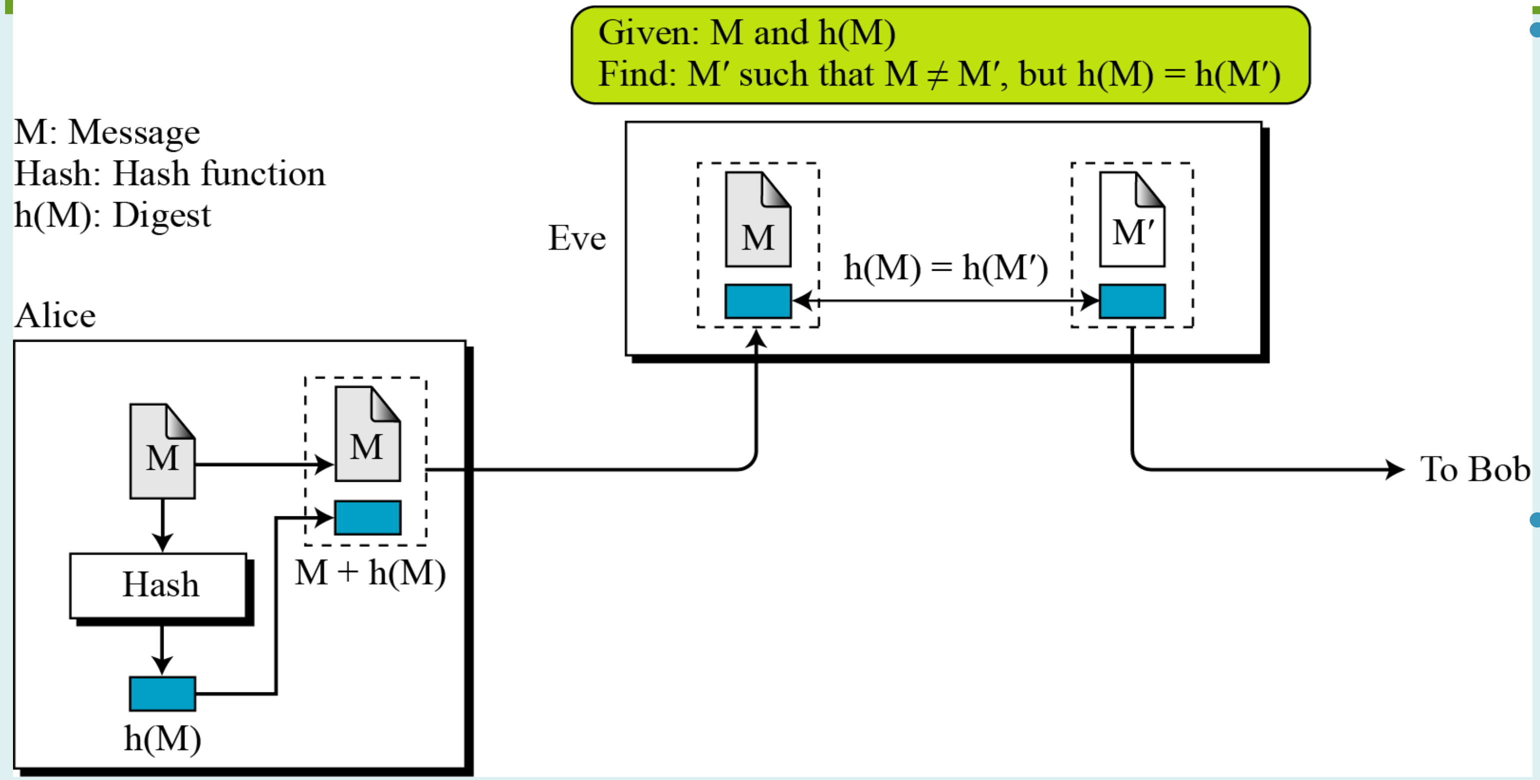
CRYPTOGRAPHIC HASH FUNCTIONS

Second Pre-image resistance

- This measures how difficult to devise a message which hashes to the known digest and its message

Second Preimage Attack

Given: M and h(M) **Find: M′ ≠ M such that h(M) = h(M′)**



Given: M and h(M)
Find: M′ such that M ≠ M′, but h(M) = h(M′)

M: Message
Hash: Hash function
h(M): Digest

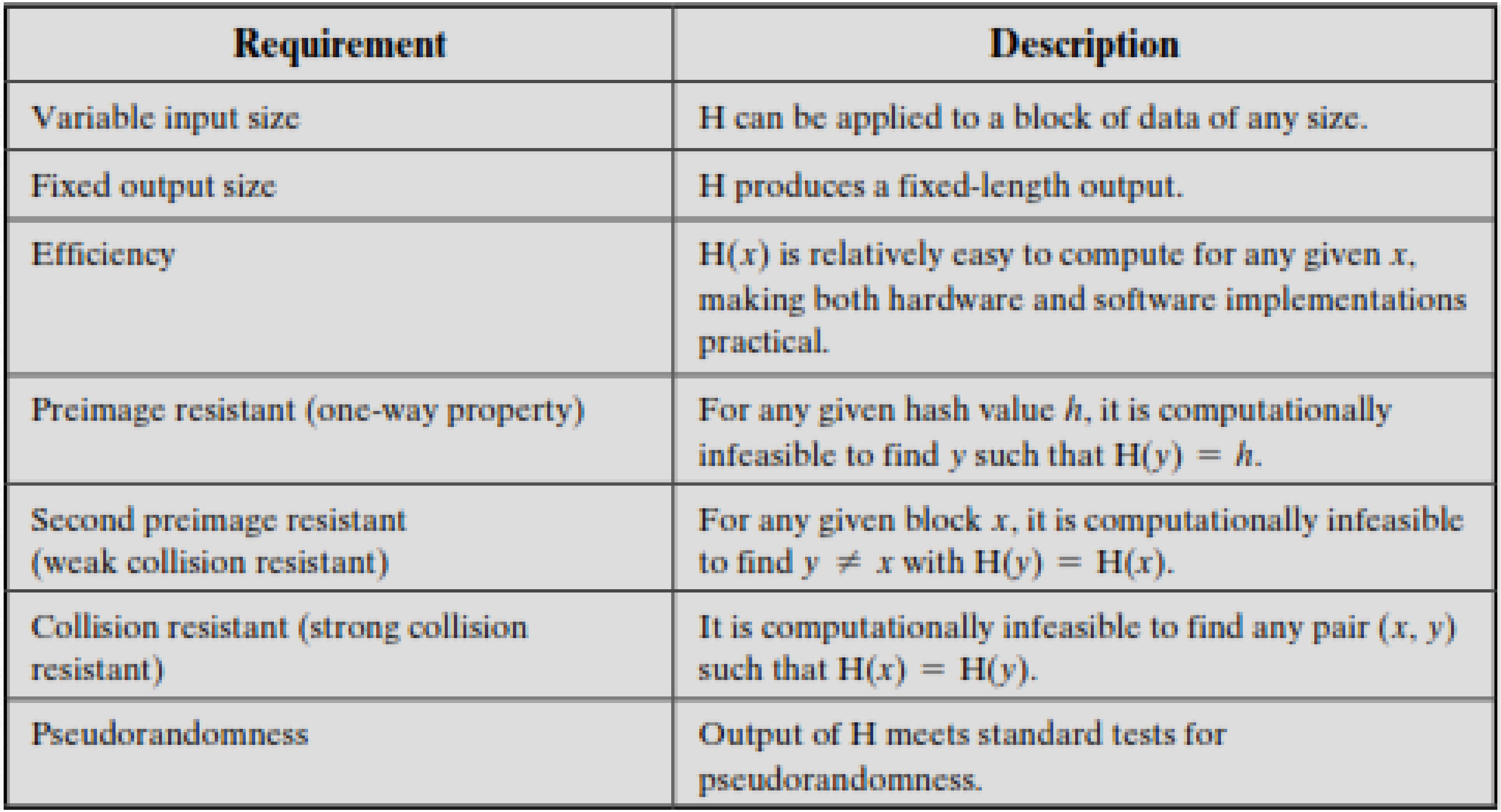
- Given one message, can’t find another message that has the same message digest. An attack that finds a second message with the same message digest is a *second pre-image* attack.
- It would be easy to forge new digital signatures from old signatures if the hash function used weren’t second preimage resistant

CRYPTOGRAPHIC HASH FUNCTIONS

- Properties of a Hash Function

5. Practical efficiency: Given input string x, the computation of $h(x)$ can be done in time bounded by a small degree polynomial (linear) in the size of x.

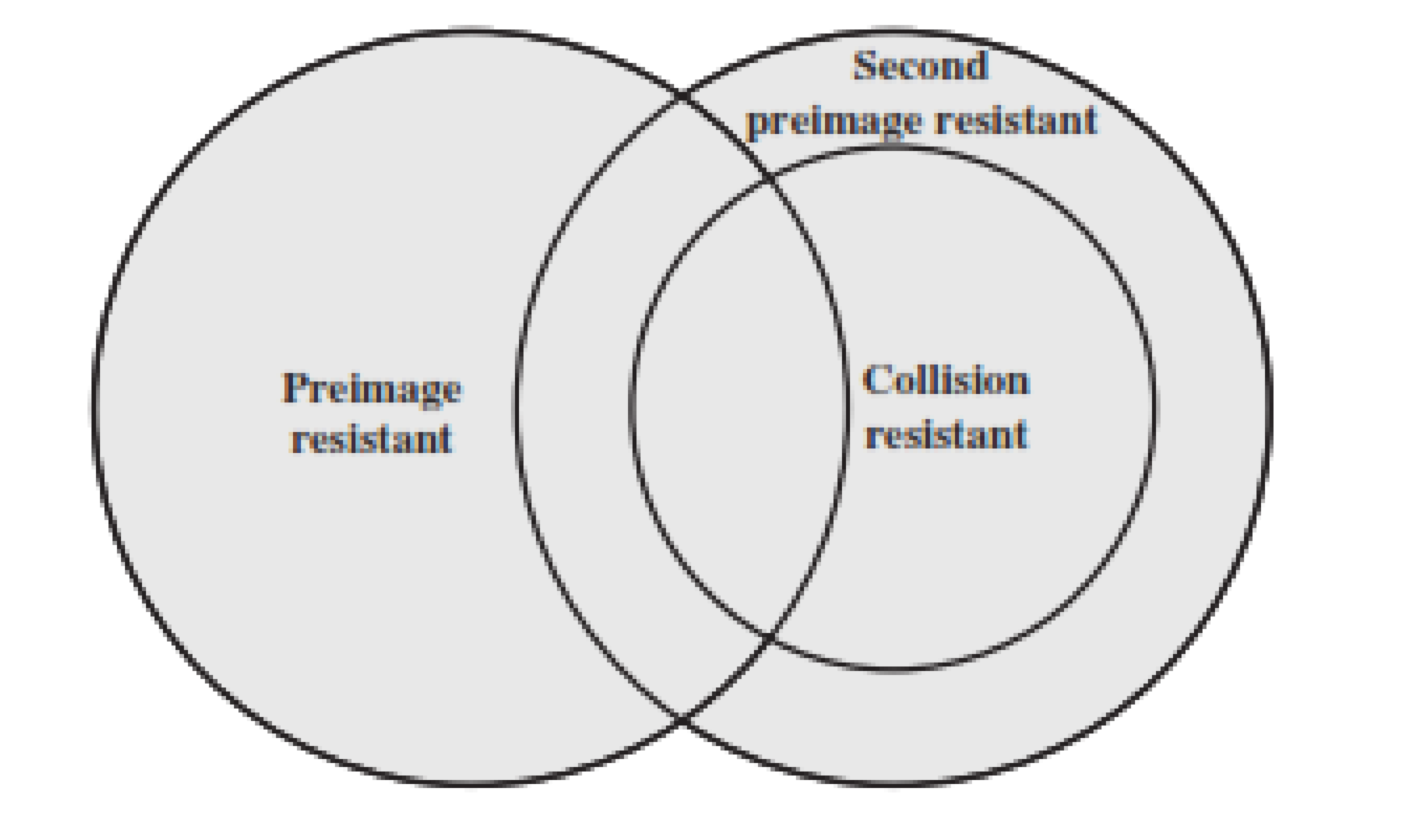
CRYPTOGRAPHIC HASH FUNCTIONS



Requirement	Description
Variable input size	H can be applied to a block of data of any size.
Fixed output size	H produces a fixed-length output.
Efficiency	H(x) is relatively easy to compute for any given x, making both hardware and software implementations practical.
Preimage resistant (one-way property)	For any given hash value h, it is computationally infeasible to find y such that H(y) = h.
Second preimage resistant (weak collision resistant)	For any given block x, it is computationally infeasible to find $y \neq x$ with H(y) = H(x).
Collision resistant (strong collision resistant)	It is computationally infeasible to find any pair (x, y) such that H(x) = H(y).
Pseudorandomness	Output of H meets standard tests for pseudorandomness.

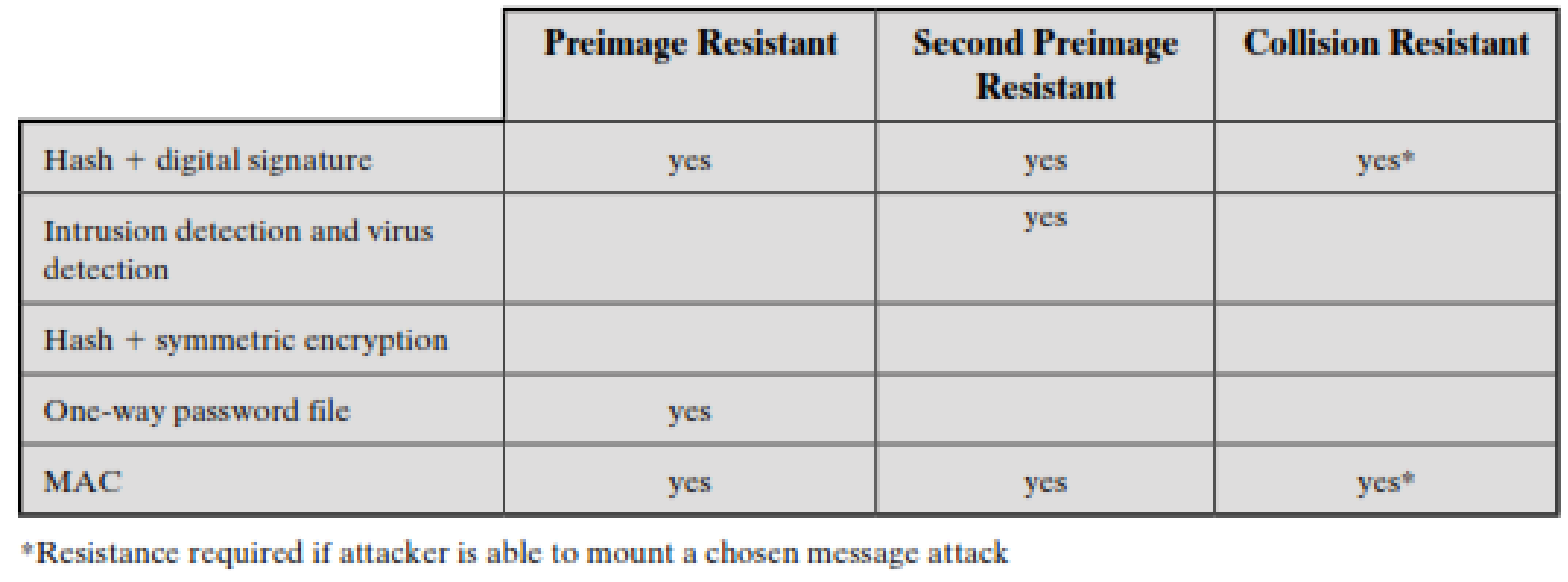
CRYPTOGRAPHIC HASH FUNCTIONS

- Relationship among Hash Function Properties



CRYPTOGRAPHIC HASH FUNCTIONS

- Hash Function Resistance Properties Required for Various Data Integrity Applications



	Preimage Resistant	Second Preimage Resistant	Collision Resistant
Hash + digital signature	yes	yes	yes*
Intrusion detection and virus detection		yes	
Hash + symmetric encryption			
One-way password file	yes		
MAC	yes	yes	yes*

*Resistance required if attacker is able to mount a chosen message attack

CRYPTOGRAPHIC HASH FUNCTIONS

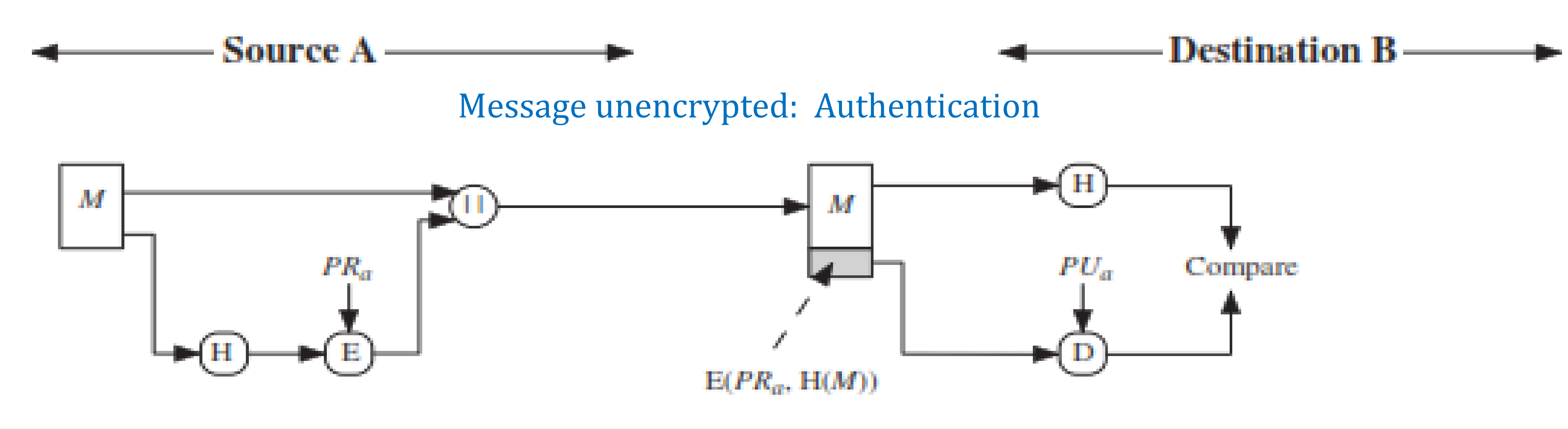
- Application of Hash Functions in Cryptography

1. Digital signatures: Modern, collision resistant hash functions were designed to create small, fixed size message digests so that a digest could act as a proxy for a possibly very large variable length message in a **digital signature algorithm**, such as RSA or DSA.

- These hash functions have since been widely used for many other "ancillary" applications, including hash-based **message authentication codes, pseudo random number generators**, and **key derivation functions**

CRYPTOGRAPHIC HASH FUNCTIONS

- Application of Hash Functions in Cryptography
- Digital signatures: The hash code is encrypted, using public-key encryption with the sender's private key providing authentication. It also provides a digital signature, because only the sender could have produced the encrypted hash code. In fact, this is the essence of the digital signature

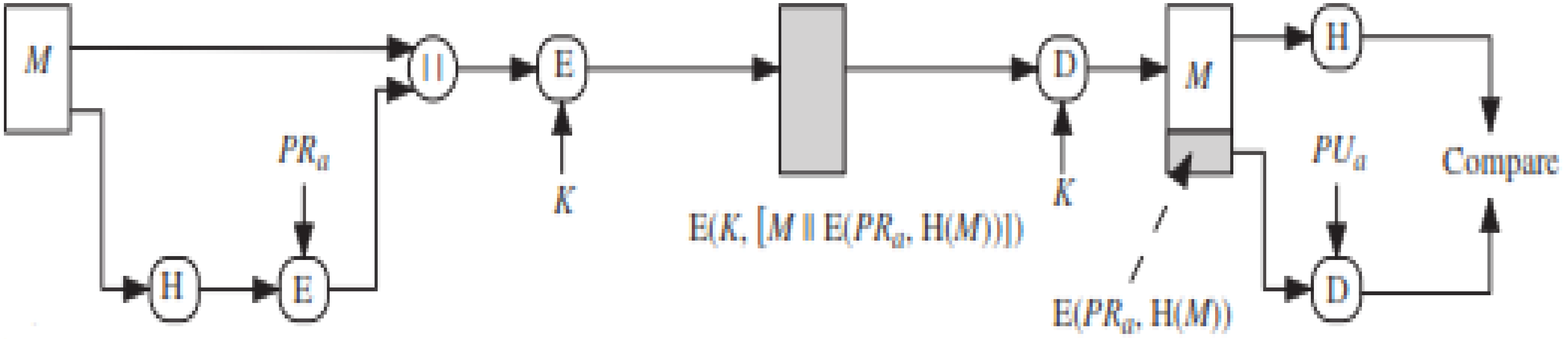


Message unencrypted: Authentication

CRYPTOGRAPHIC HASH FUNCTIONS

- Application of Hash Functions in Cryptography
- Digital signatures: If confidentiality as well as a digital signature is desired, then the message plus the private-key-encrypted hash code can be encrypted using a symmetric secret key.

Message encrypted : Confidentiality and authentication



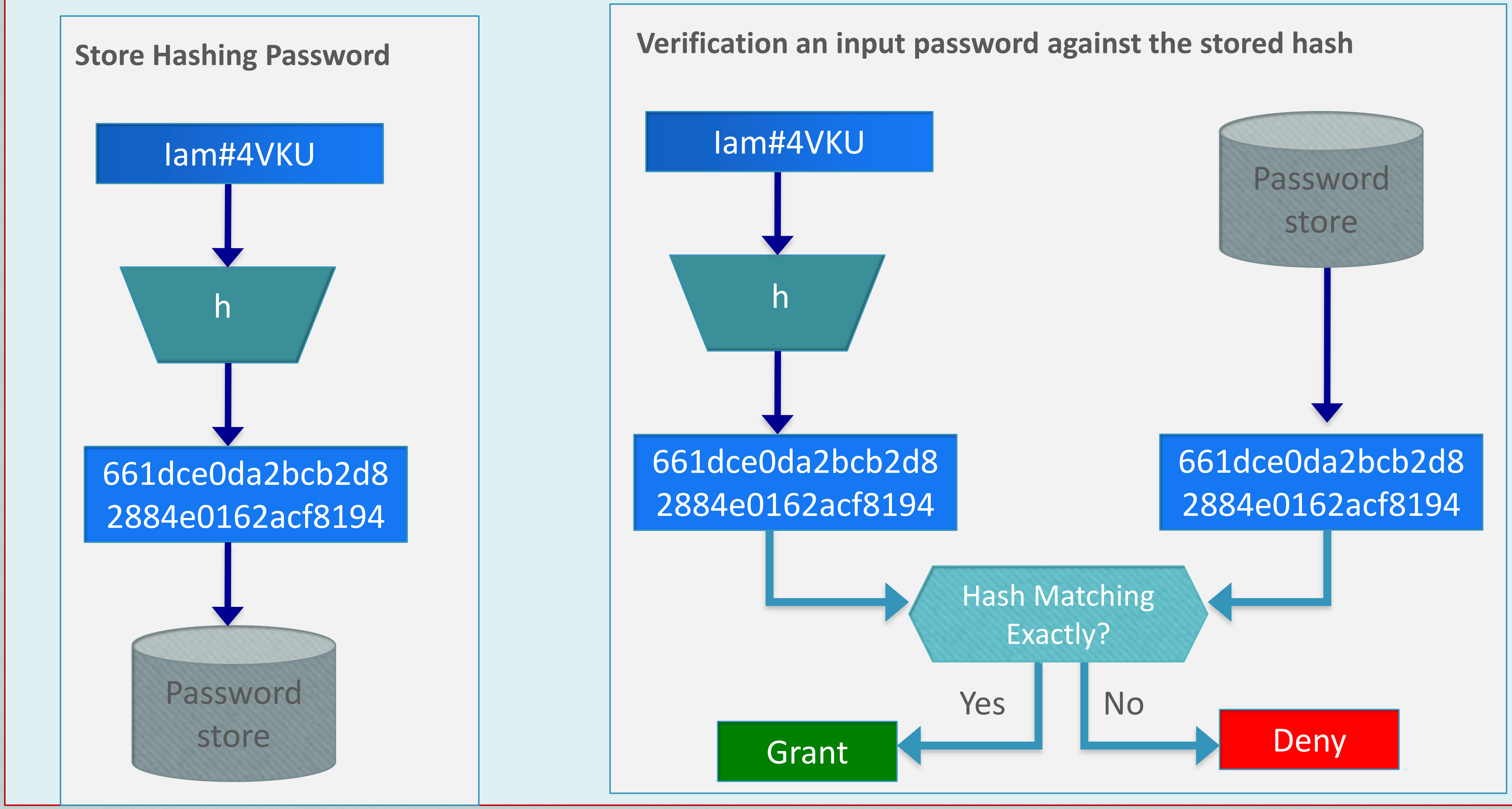
CRYPTOGRAPHIC HASH FUNCTIONS

- Application of Hash Functions in Cryptography

2. Public key cryptosystems with fit-for-application security
3. Hash functions are commonly used to create a **one-way password file.** A hash of a password is stored by an operating system rather than the password itself.
   - Thus, the actual password is not retrievable by a hacker who gains access to the password file. h
   - In simple terms, when a user enters a password, the hash of that password is compared to the stored hash value for verification.
   - This approach to password protection is used by most operating systems.

CRYPTOGRAPHIC HASH FUNCTIONS

- Password Verification: - One way Password File



CRYPTOGRAPHIC HASH FUNCTIONS

- Application of Hash Functions in Cryptography

4. Hash functions can be used for **intrusion detection and virus detection.**
   - Store H(F) for each file on a system and secure the hash values (e.g., on a CD-R that is kept secure).
   - One can later determine if a file has been modified by re-computing H(F).
   - An intruder would need to change F without changing H(F).
5. A cryptographic hash function can be used to construct a **pseudorandom function (PRF) or a pseudorandom number generator (PRNG).**
   - A common application for a hash-based PRF is for the generation of symmetric keys.

CRYPTOGRAPHIC HASH FUNCTIONS

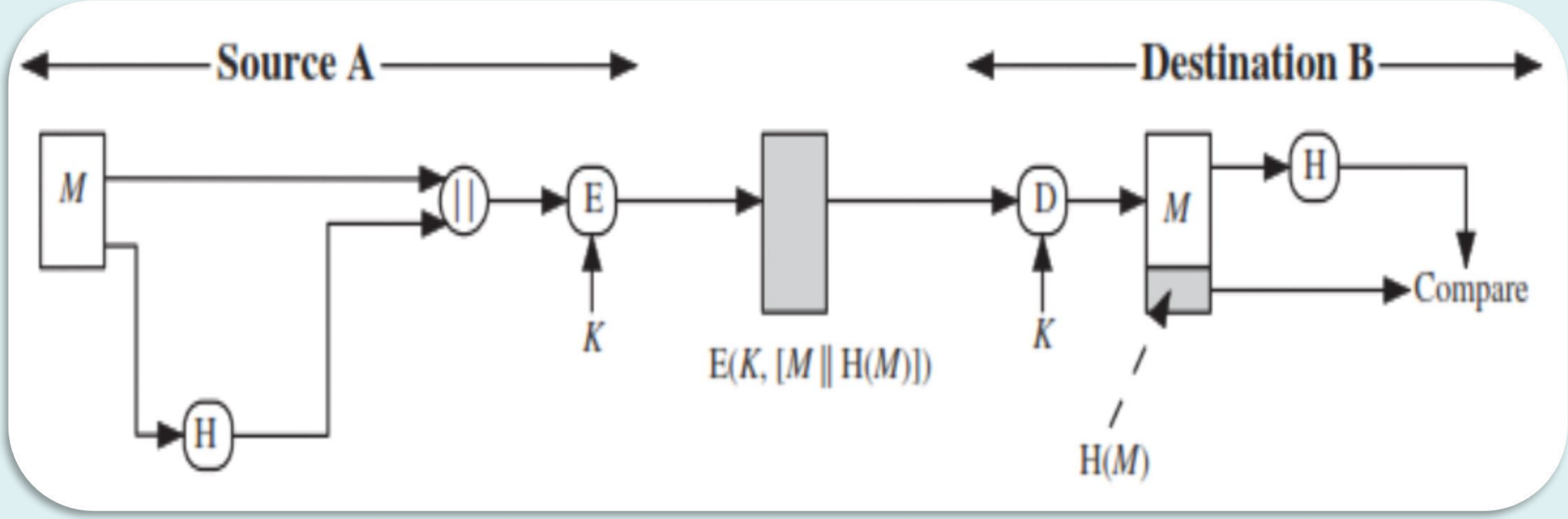
- Detailed basic uses of Hash Functions in Cryptography

1. The message plus concatenated hash code is encrypted using symmetric encryption.

- Because only A and B share the secret key, the message must have come from A and has not been altered.
- The hash code provides the structure or redundancy required to achieve authentication. Because encryption is applied to the entire message plus hash code, confidentiality is also provided.

CRYPTOGRAPHIC HASH FUNCTIONS

- Detailed basic uses of Hash Functions in Cryptography



CRYPTOGRAPHIC HASH FUNCTIONS

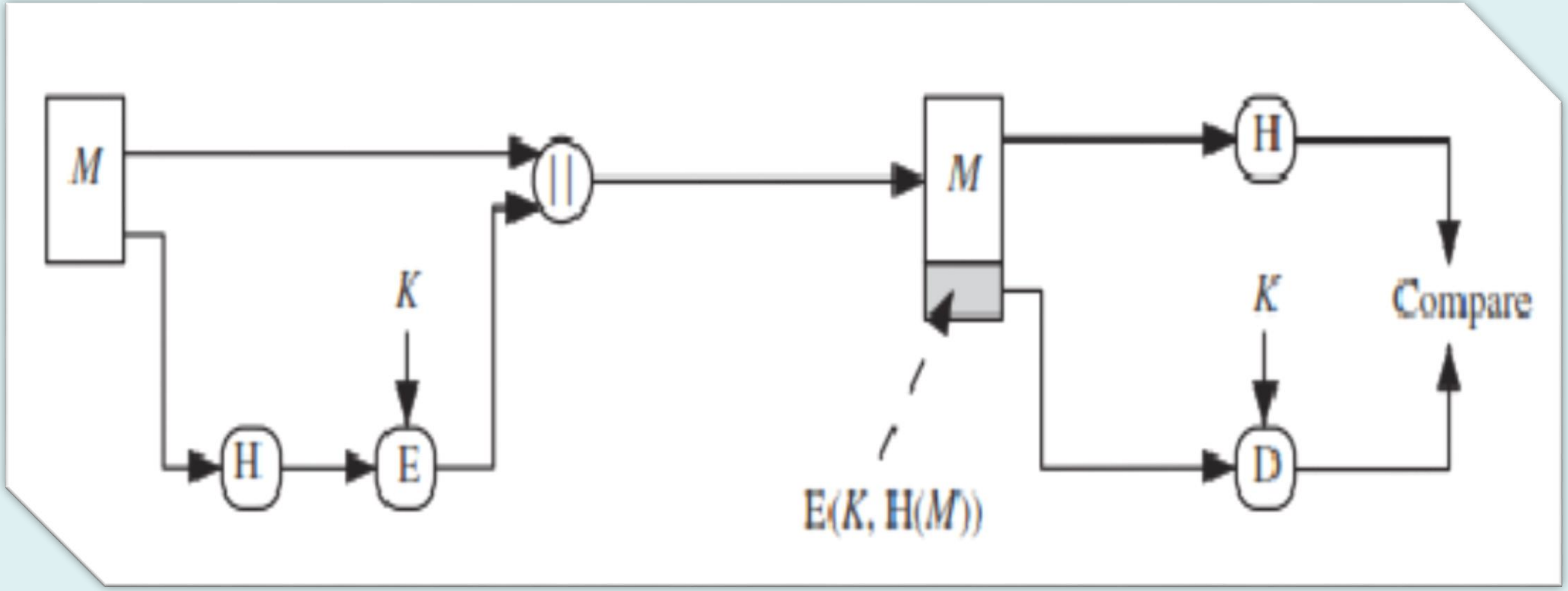
- Detailed basic uses of Hash Functions in Cryptography

2. Only the hash code is encrypted, using symmetric encryption.

- This reduces the processing burden for those applications that do not require confidentiality

CRYPTOGRAPHIC HASH FUNCTIONS

- Detailed basic uses of Hash Functions in Cryptography



CRYPTOGRAPHIC HASH FUNCTIONS

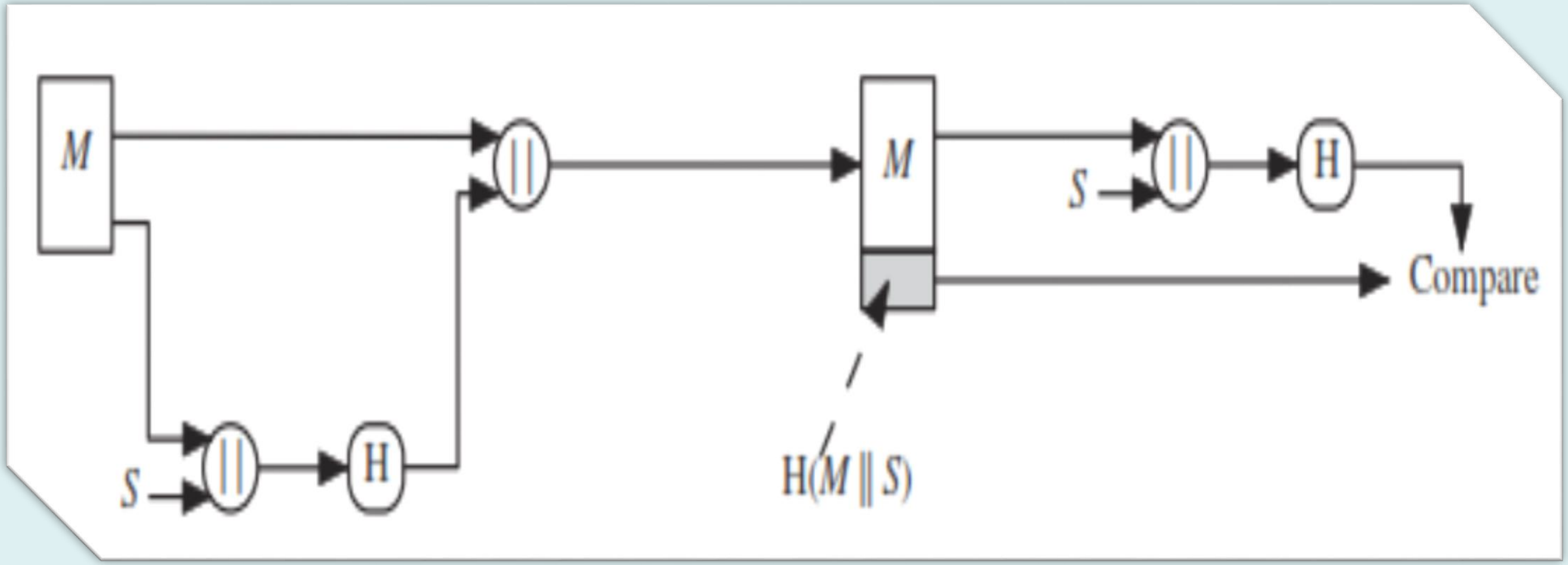
- Detailed basic uses of Hash Functions in Cryptography

3. It is possible to use a hash function but no encryption for message authentication.

- The technique assumes that the two communicating parties share a common secret value .
- A computes the hash value over the concatenation of and appends the resulting hash value to .
- Because B possesses , it can recomputed the hash value to verify. Because the secret value itself is not sent, an opponent cannot modify an intercepted message and cannot generate a false message.

CRYPTOGRAPHIC HASH FUNCTIONS

- Detailed basic uses of Hash Functions in Cryptography



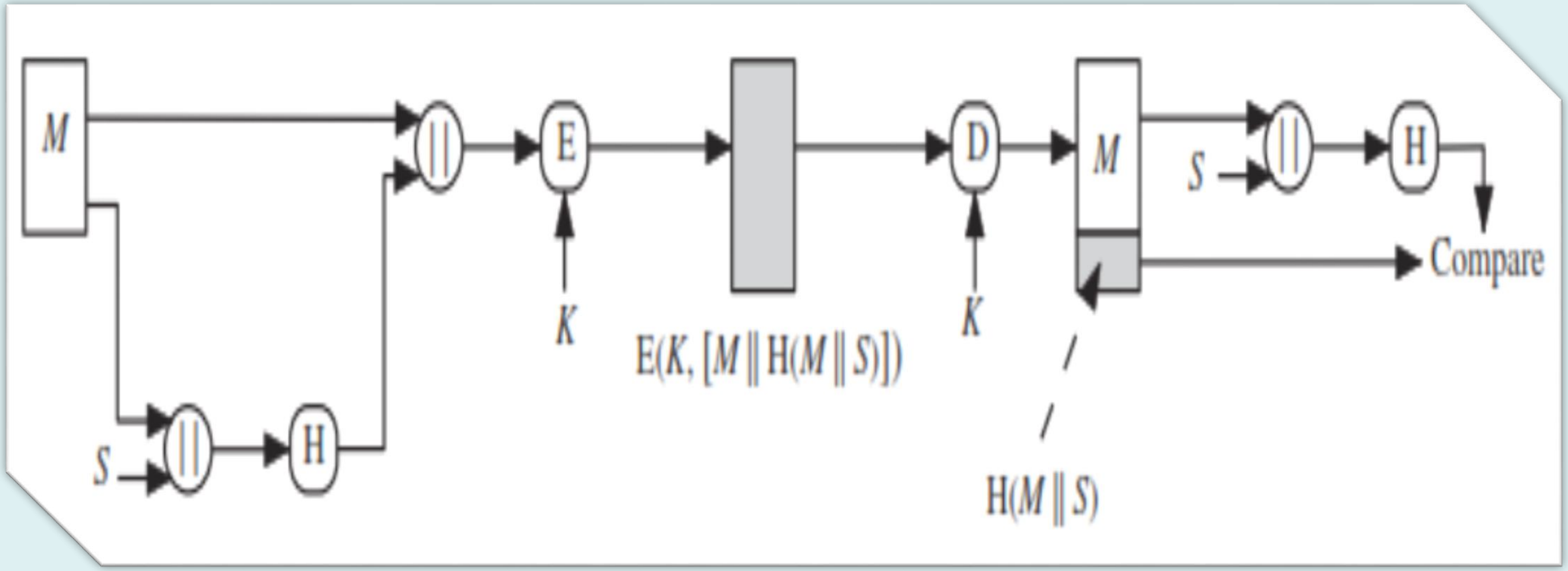
CRYPTOGRAPHIC HASH FUNCTIONS

- Detailed basic uses of Hash Functions in Cryptography

4. Confidentiality can be added to the approach of method (c) by encrypting the entire message plus the hash code.

CRYPTOGRAPHIC HASH FUNCTIONS

- Detailed basic uses of Hash Functions in Cryptography



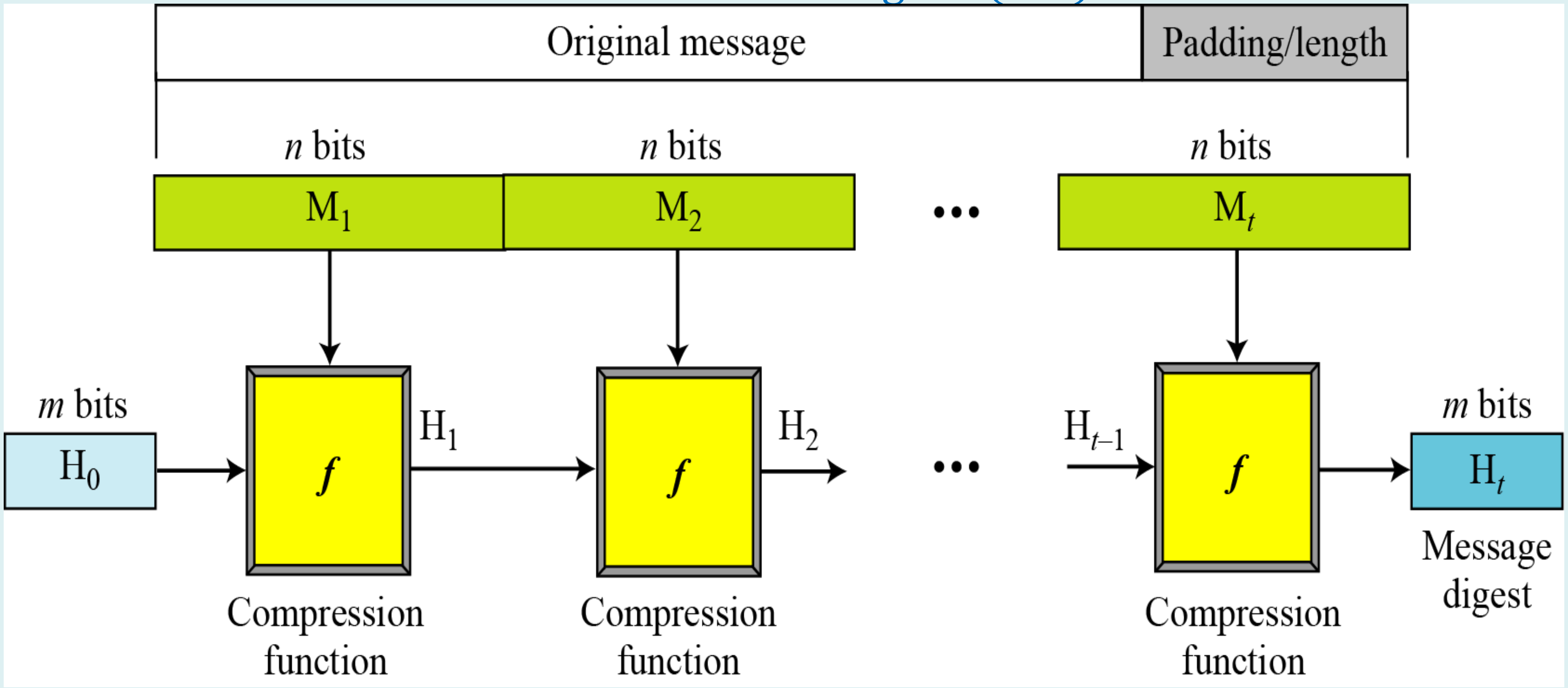
CRYPTOGRAPHIC HASH FUNCTIONS

Structure of Hash Functions: Merkle-Damgard (MD)Scheme

- Well-known method to build cryptographic hash function
- A message of arbitrary length is broken into blocks
  - Length depends on the compression function *f*
  - Pad the size of the message into a multiple of the block size.
  - Sequentially process blocks , taking as input the result of the hash so far and the current message block, to produce the final fixed length output

CRYPTOGRAPHIC HASH FUNCTIONS

- Structure of Hash Functions: Merkle-Damgard (MD)Scheme



CRYPTOGRAPHIC HASH FUNCTIONS

Hash Function Families:

1. MD (Message Digest)

- Family of one-way hash functions by Ronald Rivest
  - All produces 128 bits hash value
- **MD2: 1989**
  - Optimized for 8 bit computer
  - Collision found in 1995
- **MD4: 1990**
  - Full round collision attack found in 1995
- **MD5: 1992**
  - Specified as Internet standard in RFC 1321
  - Since 1997 it was theoretically not so hard to create a collision
  - Practical Collision MD5 has been broken since 2004
  - Collision attack published in 2007

CRYPTOGRAPHIC HASH FUNCTIONS

Popular Hash Algorithms

1. MD5

- Most popular hash algorithm until very recently – concerns for its security were raised and was proposed to be replaced by SHA-1, SHA-2
- Developed by Rivest at MIT
- For a message of arbitrary length, it produces an output of 128 bits

CRYPTOGRAPHIC HASH FUNCTIONS

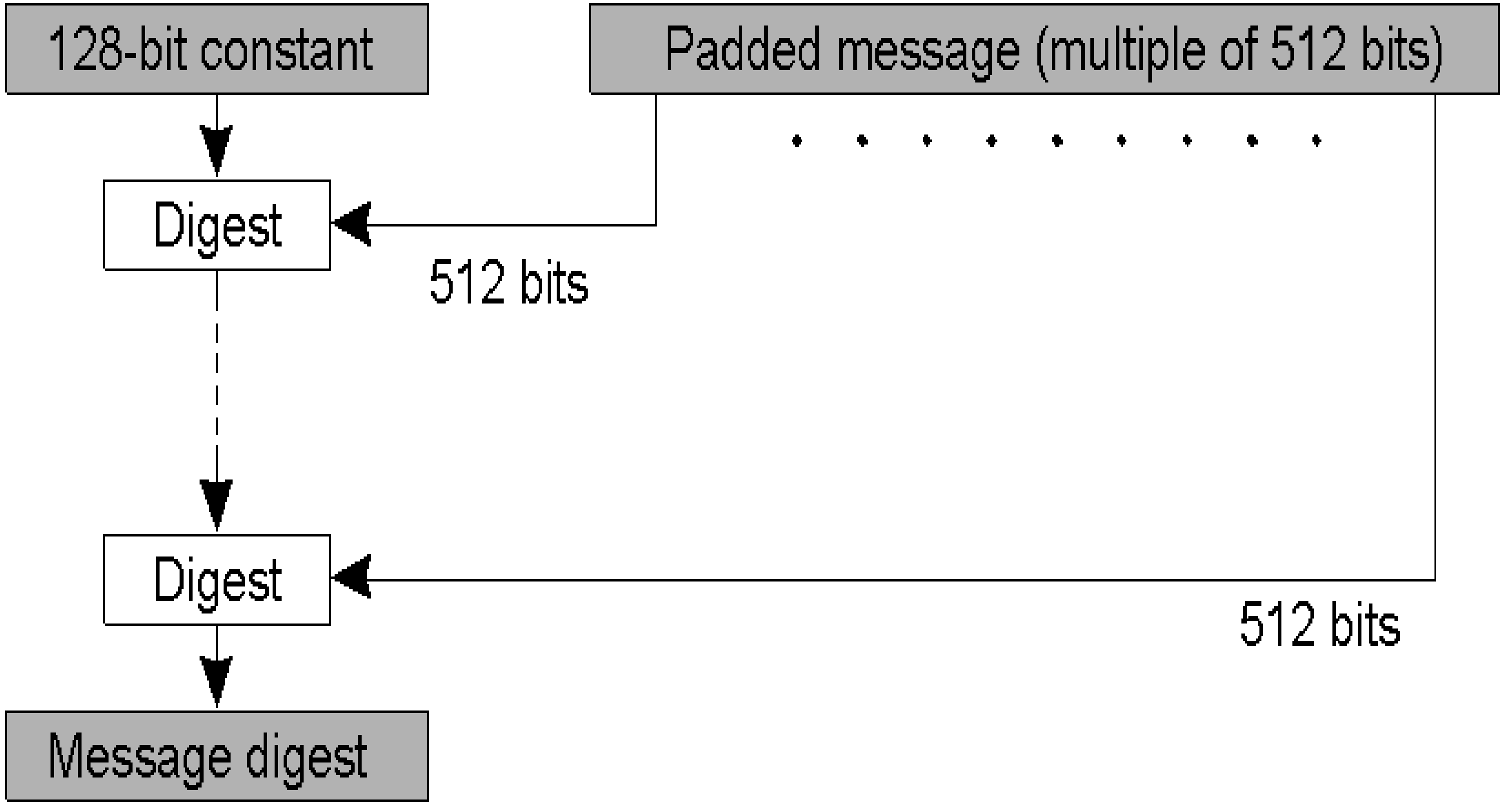
Popular Hash Algorithms

1. MD5

- MD5 message-digest algorithm takes as input a message of arbitrary length and produces as output a 128-bit "fingerprint" or "message digest" of the input
- The MD5 algorithm is intended for digital signature applications, where a large file must be "compressed" in a secure manner before being encrypted with a private (secret) key under a public-key cryptosystem such as RSA.

CRYPTOGRAPHIC HASH FUNCTIONS

MD5



CRYPTOGRAPHIC HASH FUNCTIONS

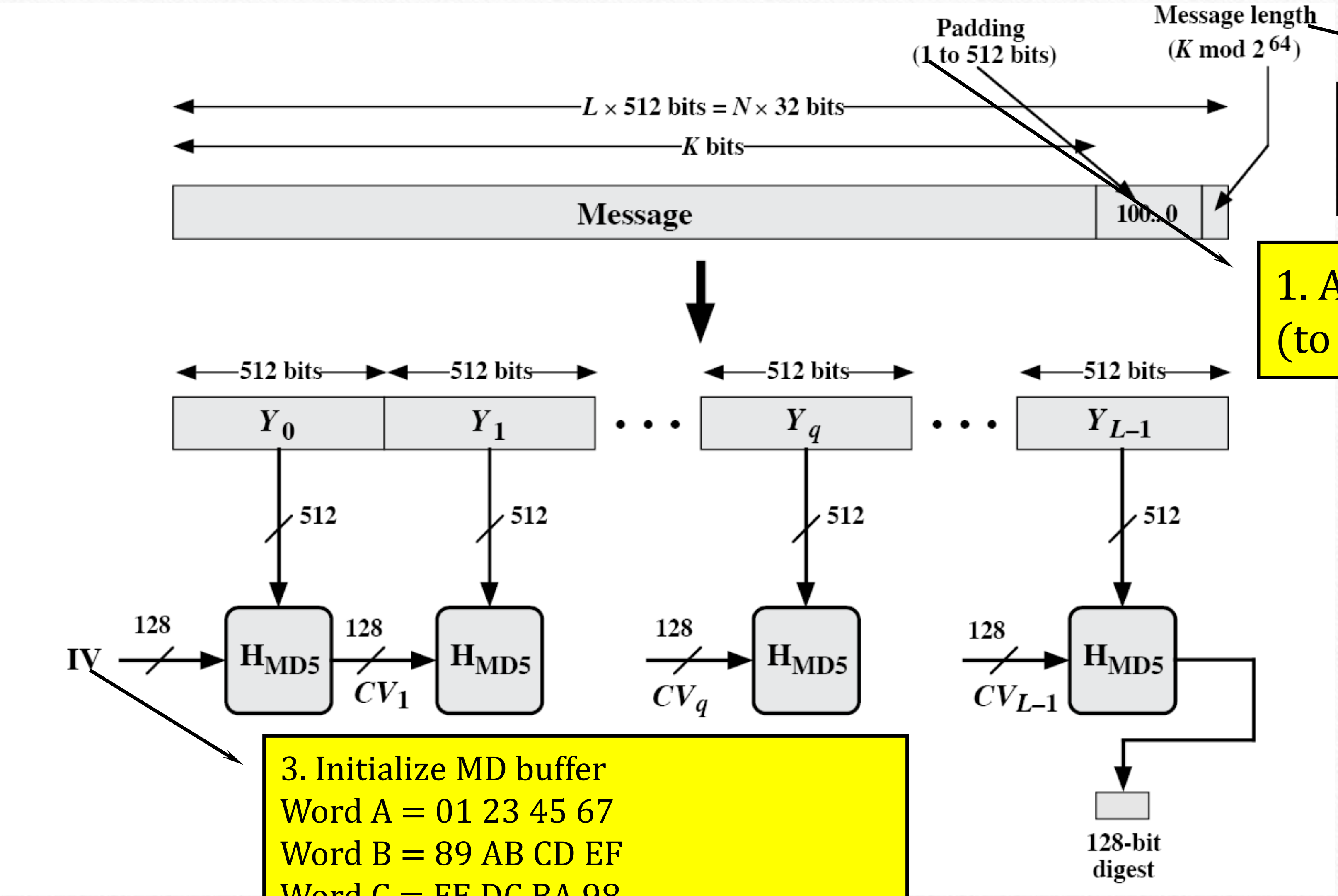
1. MD5

Processes the input in blocks of 512 bits

Idea:

1. Start by padding the message to a length of 448 bits modulo 512 – padding is always added even if the message is of required length; the length of the message is added on 64 bits so that altogether the length is a multiple of 512 bits
2. Several rounds (4 rounds with 16 steps each), each round takes a block of 512 bits from the message and mixes it thoroughly with a 128 bit buffer that was the result of the previous round
3. The last content of the buffer is the hash value

CRYPTOGRAPHIC HASH FUNCTIONS



2. Append length (64bits)

1. Append padding bits (to 448 mod 512)

3. Initialize MD buffer
Word A = 01 23 45 67
Word B = 89 AB CD EF
Word C = FE DC BA 98
Word D = 76 54 32 10

CRYPTOGRAPHIC HASH FUNCTIONS

MD5 – the algorithm

1. Padding

- The input message is "padded" (extended) so that its length (in bits) equals to 448 mod 512. Padding is always performed, even if the length of the message is already 448 mod 512.
- Padding is performed as follows: a single "1" bit is appended to the message, and then "0" bits are appended so that the length in bits of the padded message becomes congruent to 448 mod 512. At least one bit and at most 512 bits are appended.

CRYPTOGRAPHIC HASH FUNCTIONS

MD5 – the algorithm

2. Append length

- A 64-bit representation of the length of the message is appended to the result of step1. If the length of the message is greater than 2^{64}, only the low-order 64 bits will be used.
- The resulting message (after padding with bits) has a length that is an exact multiple of 512 bits. The input message will have a length that is an exact multiple of 16 (32-bit) words. The rules of appending length are:

i. The length of the original message in bytes is converted to its binary format of 64 bits. If overflow happens, only the low-order 64 bits are used.

ii. Break the 64-bit length into 2 words (32 bits each).

iii. The low-order word is appended first and followed by the high-order word.

CRYPTOGRAPHIC HASH FUNCTIONS

MD5 – the algorithm

3. Initialize MD buffer

- A four-word buffer (A, B, C, D) is used to compute the message digest. Each of A, B, C, D is a 32-bit register. These registers are initialized to the following values in hexadecimal, low-order bytes first: The rules of initializing buffer are:
- The buffer is divided into 4 words (32 bits each), named as A, B, C, and D.
  - i. Word A is initialized to: 0x67452301
  - ii. Word B is initialized to: 0xEFCDAB89
  - iii. Word C is initialized to: 0x98BADCFE
  - iv. Word D is initialized to: 0x10325476

CRYPTOGRAPHIC HASH FUNCTIONS

MD5 – the algorithm

4. Process message in 16-word blocks

- Four functions will be defined such that each function takes an input of three 32-bit words and produces a 32-bit word output.

F (X, Y, Z) = XY or not (X) Z

G (X, Y, Z) = XZ or Y not (Z)

H (X, Y, Z) = X xor Y xor Z

I (X, Y, Z) = Y xor (X or not (Z))

CRYPTOGRAPHIC HASH FUNCTIONS

MD5 – the algorithm (Summary)

1. Padding: add a bit 1 followed by the necessary number of bits 0
2. Append length – the length is represented by 64 bits. If the length is larger than 2^{64} , take the 64 least representative bits
3. Initialize MD buffer with the following 4 values, all on 32 bits:

A=0x67452301.

B=0xEFCDAB89.

C=0x98BADCFE.,

D=0x10325476.

CRYPTOGRAPHIC HASH FUNCTIONS

MD5 – the algorithm (Summary)

4. Process each message block of 512 bits in 4 rounds. Each round takes as input the 512 bits in the input and the content of the buffer ABCD and updates the buffer ABCD. The four words A,B,C,D in the output of the 4^{th} round are added modulo 2^{32} to the corresponding words A,B,C,D of the input to the first round
5. Output: the 128 bits in the buffer ABCD after the last round

CRYPTOGRAPHIC HASH FUNCTIONS

Strength of MD5

- Every bit of the output is a function of all bits of the input

Rivest’s conjecture: As strong as it can be for a 128-bit hash: birthday attack on the order of $2^{64}$and finding a message with a given digest is on the order of 2^{128}

- Vulnerabilities found in 1996, then after 10 years a number of other weaknesses reported, most serious in 2008
  - 2008: fake certification of SSL was demonstrated based on MD5
  - Currently classified as cryptographically weak
- Used in HMAC

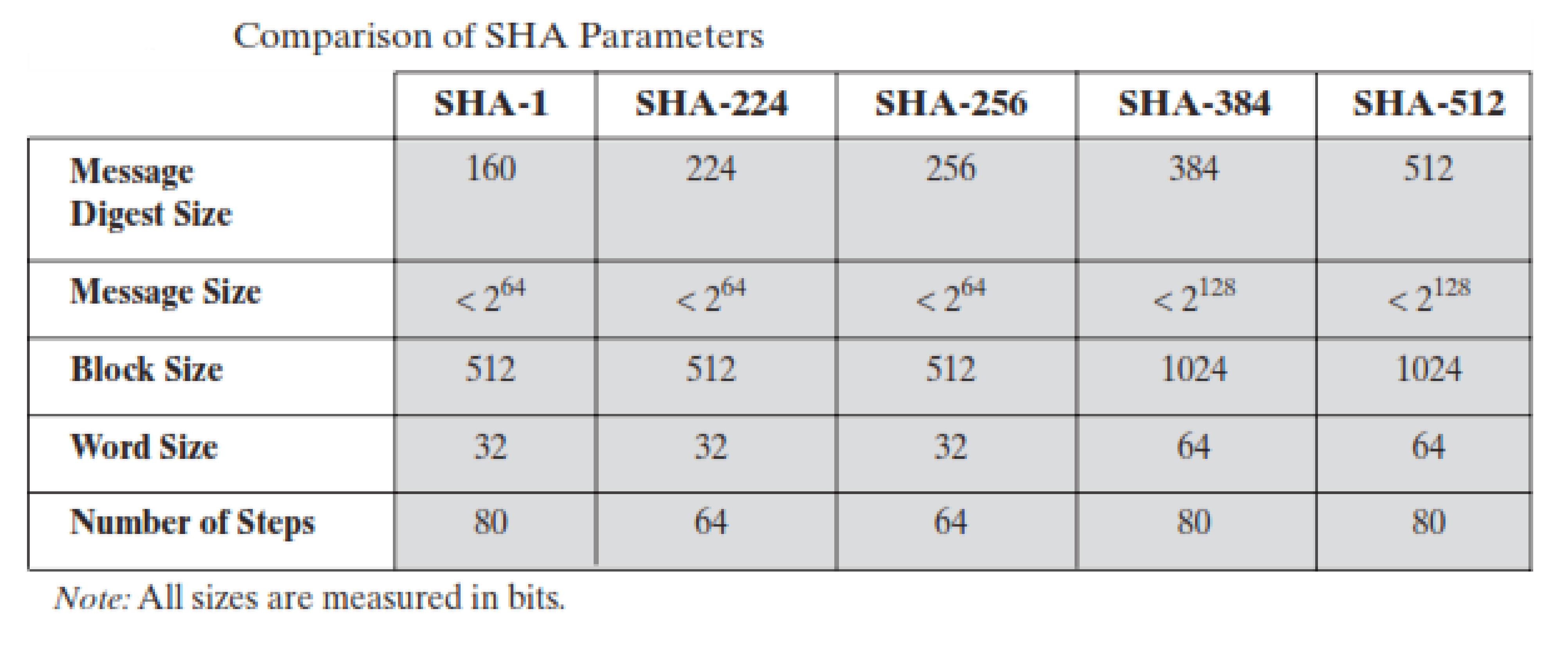
CRYPTOGRAPHIC HASH FUNCTIONS

Hash Function Families

- **SHA (Secure Hash Algorithm)**
  - Designed by NIST
  - Family: SHA-0, SHA-1, and SHA-2
    - SHA-2: SHA-224, SHA-256, SHA-384, SHA-512
    - SHA-3: New standard
- **RIPEMD (Race Integrity Primitive Evaluation Message Digest)**
  - Developed by Katholieke University Leuven Team
  - Family : RIPEMD-128, RIPEMD-160, RIPEMD-256, RIPEMD-320,

CRYPTOGRAPHIC HASH FUNCTIONS

SHA 2



Comparison of SHA Parameters

	SHA-1	SHA-224	SHA-256	SHA-384	SHA-512
Message Digest Size	160	224	256	384	512
Message Size	$< 2^{64}$	$< 2^{64}$	$< 2^{64}$	$< 2^{128}$	$< 2^{128}$
Block Size	512	512	512	1024	1024
Word Size	32	32	32	64	64
Number of Steps	80	64	64	80	80

Note: All sizes are measured in bits.

CRYPTOGRAPHIC HASH FUNCTIONS

Assignment

Discuss and compare the operations of SHA family of hash functions (20 marks)