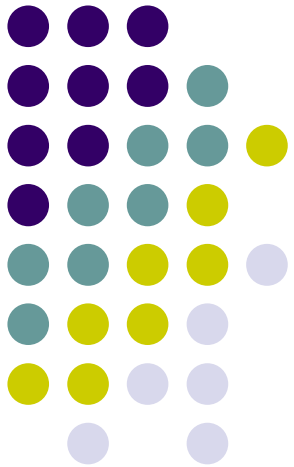
Universidade Federal de São Paulo – UNIFESP

Segurança Computacional

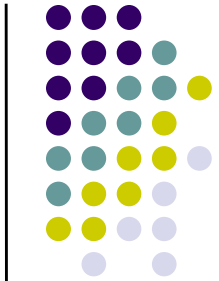
Aula 05: Autenticação



Prof.
Valério Rosset

Autenticação

Autenticação de Mensagens



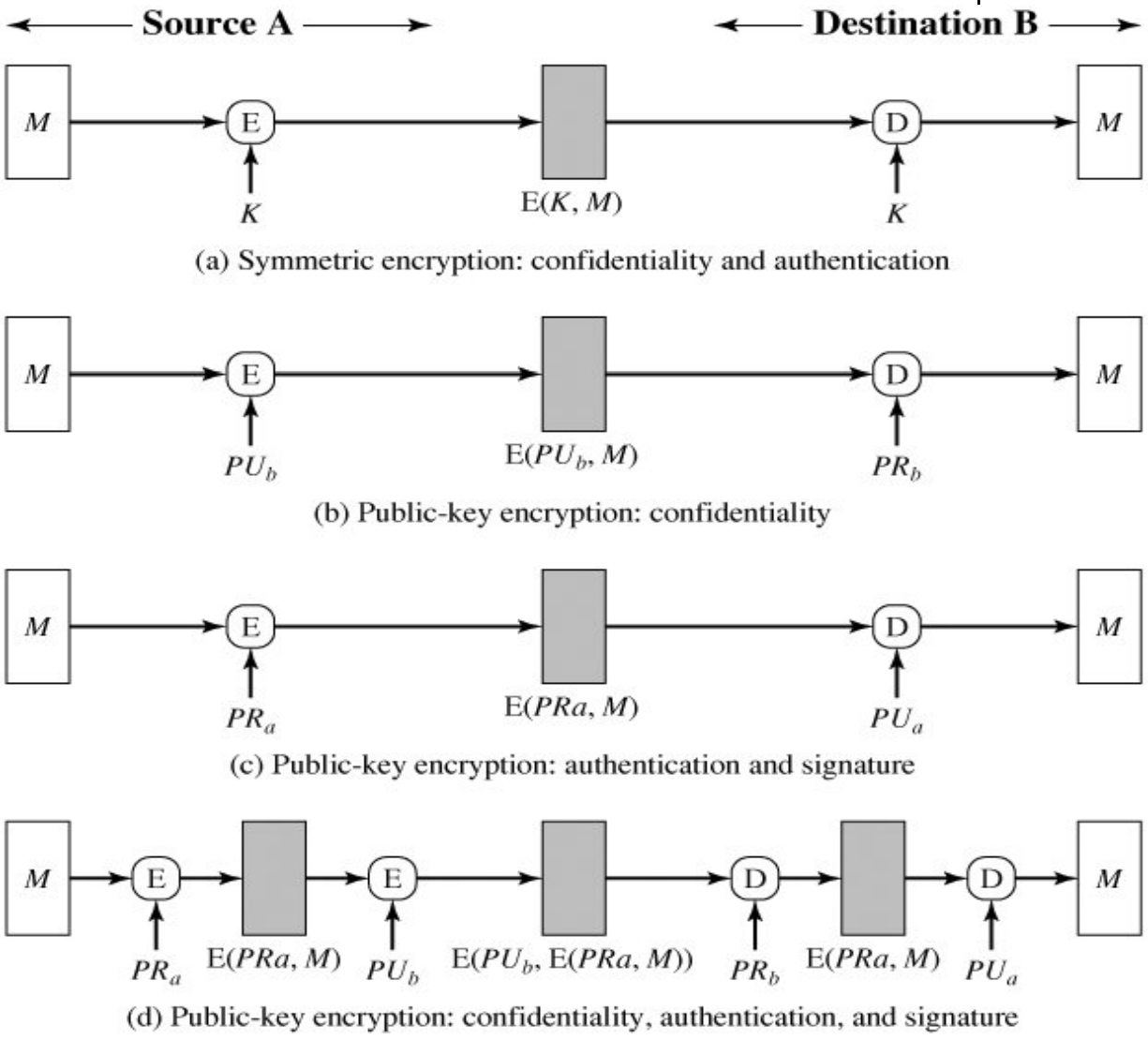
- Autenticação de Mensagens é utilizada para ***verificar a origem*** de mensagens e ***integridade*** do seu conteúdo.
- O processo de autenticação está baseado num segmento de informação calculado a partir da mensagem a ser enviada:
- Esse segmento pode ser obtido através das seguintes funções:
  - Criptografia da Mensagem
  - Código de Autenticação de Mensagens
  - Funções Hash unidirecionais

Autenticação

Autenticação de Mensagens

- Criptografia da Mensagem

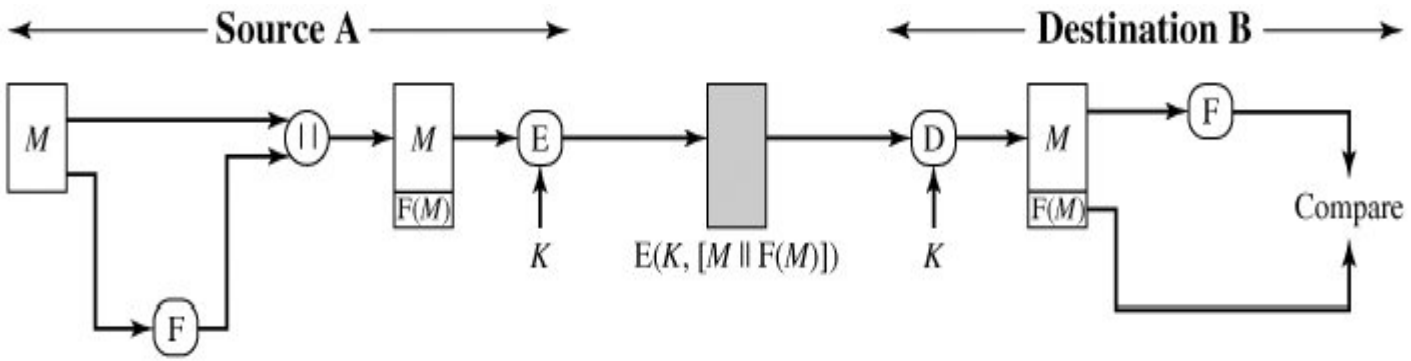
- Autenticação
- Confidencialidade
- Assinatura



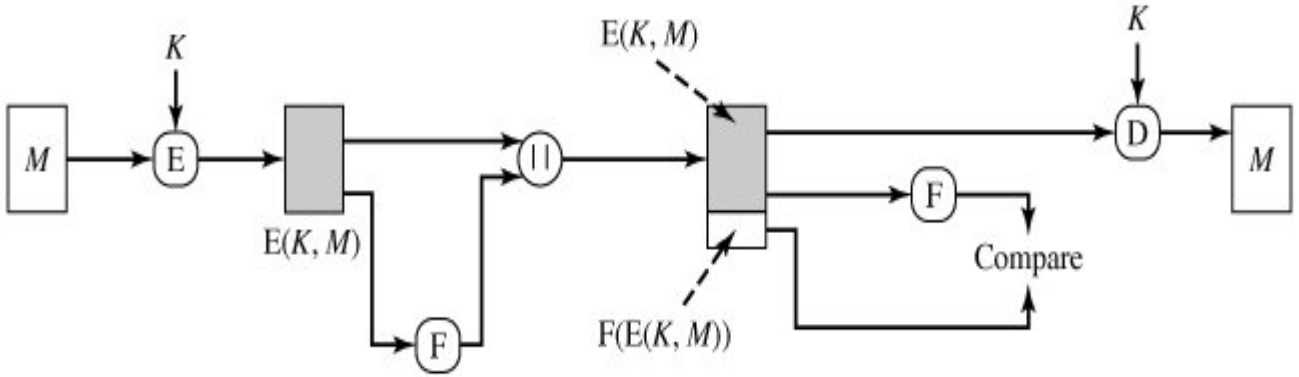
Autenticação

Autenticação de Mensagens

- **Integridade**
  - Controle de erros Internos
  - Controle de Erros Externos



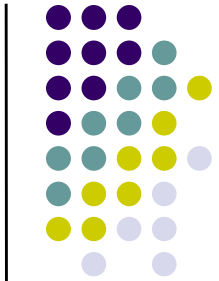
(a) Internal error control



(b) External error control

Autenticação

Métodos – Autenticação de Mensagens

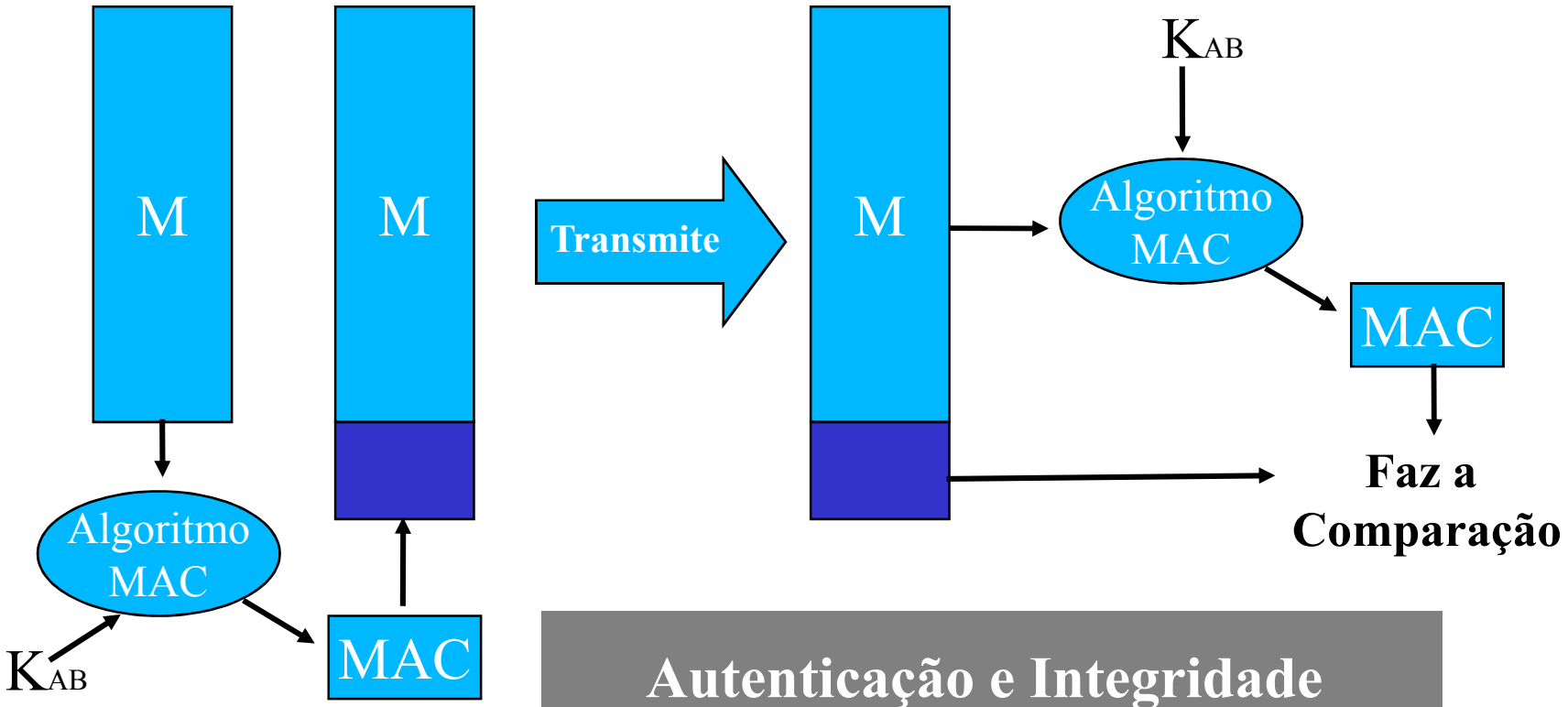


- **Código de Autenticação de Mensagens**
  - Conteúdo da mensagem não é cifrado. Porém é adicionado à mensagem um bloco contendo um hash.
    - Message Authentication Code (MAC)
    - Uso de uma chave secreta K_{AB} para gerar um pequeno bloco de dados conhecido como código de autenticação da mensagem, anexado a esta.
    - $MAC_M = F(K_{AB}, M)$
    - O receptor gera o mesmo código e compara

Autenticação

Mecanismos de Autenticação

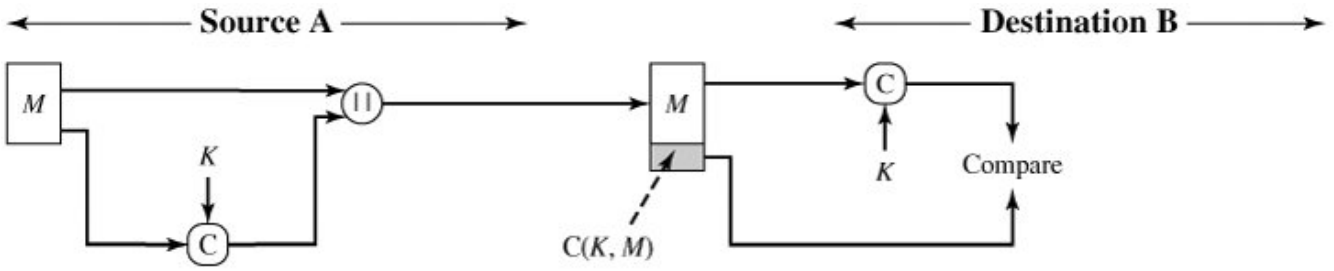
- ***Message Authentication Code (MAC)***



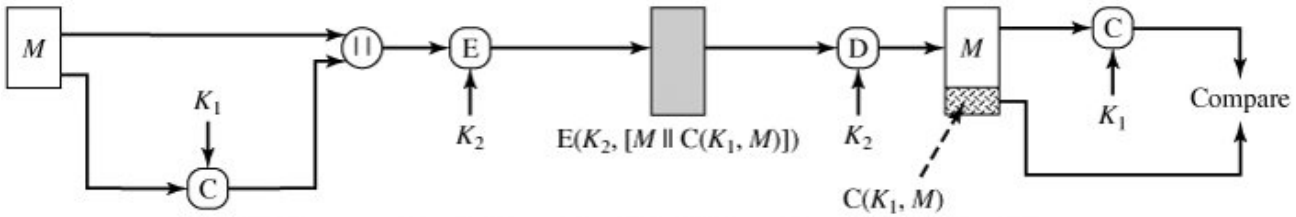
Autenticação

Mecanismos de Autenticação

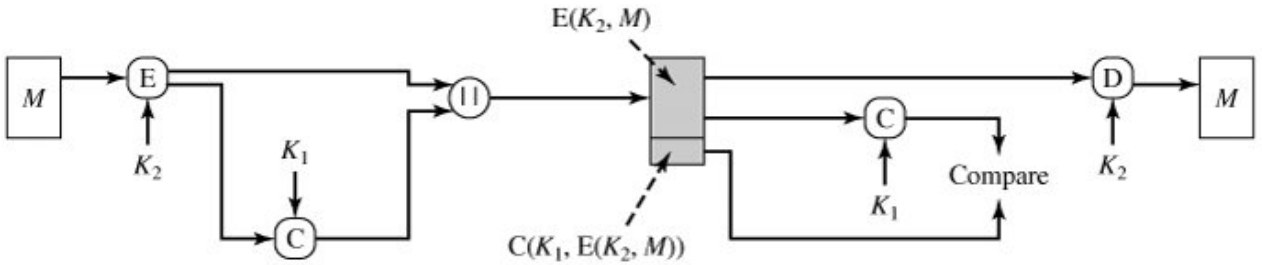
- ***Usos do Message Authentication Code (MAC)***



(a) Message authentication



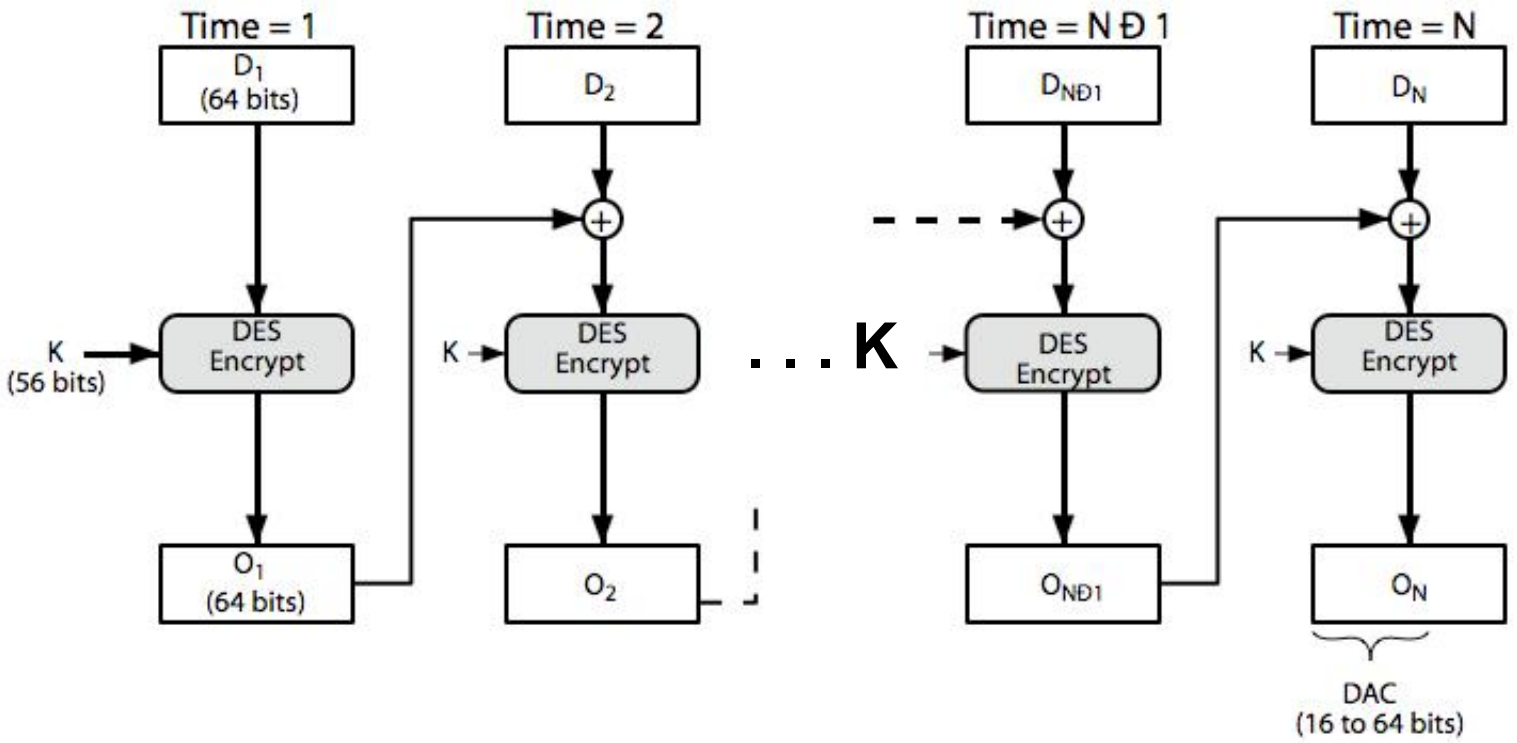
(b) Message authentication and confidentiality; authentication tied to plaintext



(c) Message authentication and confidentiality; authentication tied to ciphertext

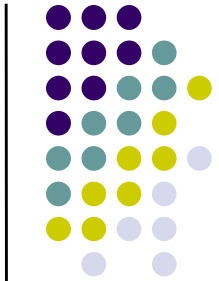
Data Authentication Algorithm (DAA)

- Baseado no DES



Autenticação

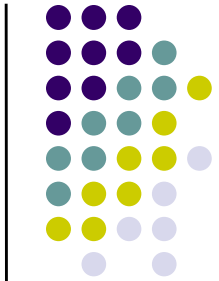
Mecanismos de Autenticação



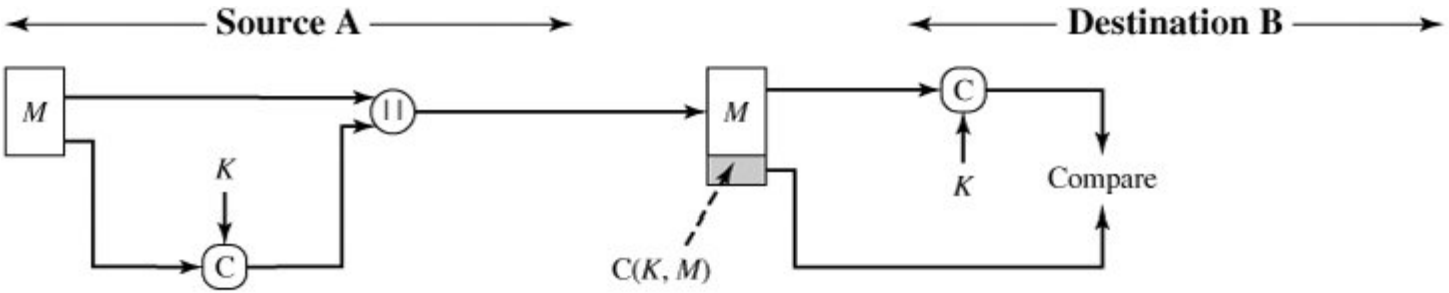
- Funções Hash Undirecionais
  - Tem como entrada uma mensagem M
  - Produz como saída um resumo de M
    - H(M)
  - Mudança de qualquer bit resulta uma mudança em H(M).
  - Hash é um função que independe de chaves criptográficas.

Autenticação

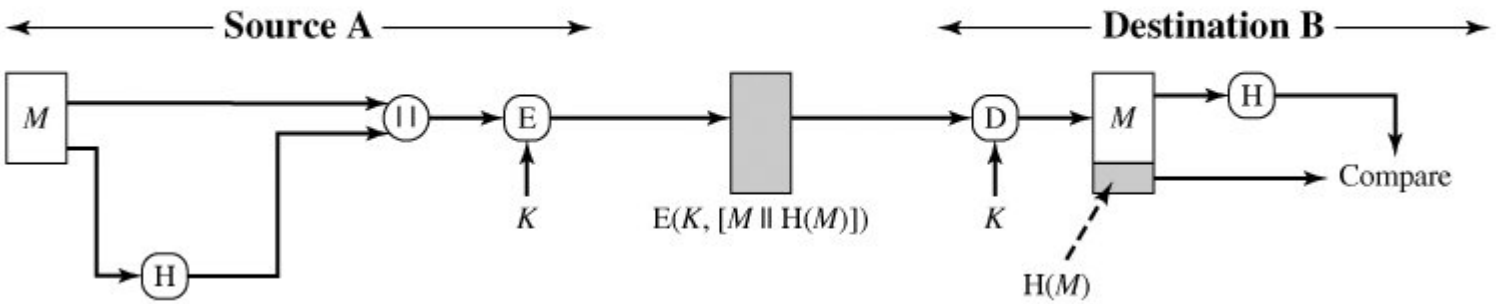
Mecanismos de Autenticação



- **Usos básicos de Hash**



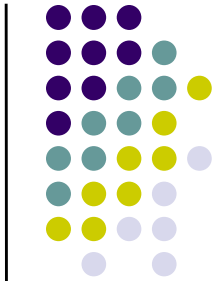
(a) Message authentication



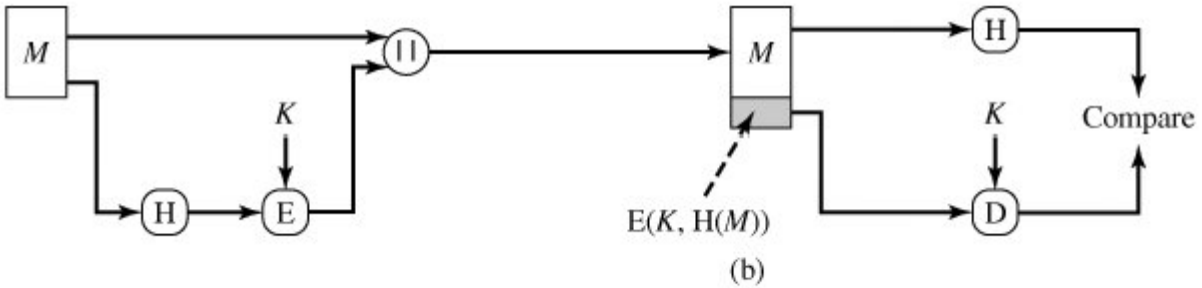
(a)

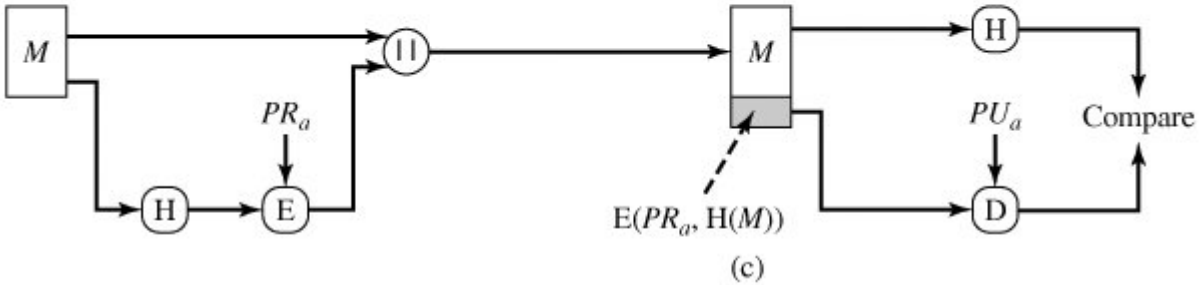
Autenticação

Mecanismos de Autenticação



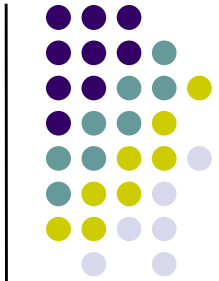
- **Usos básicos de Hash**



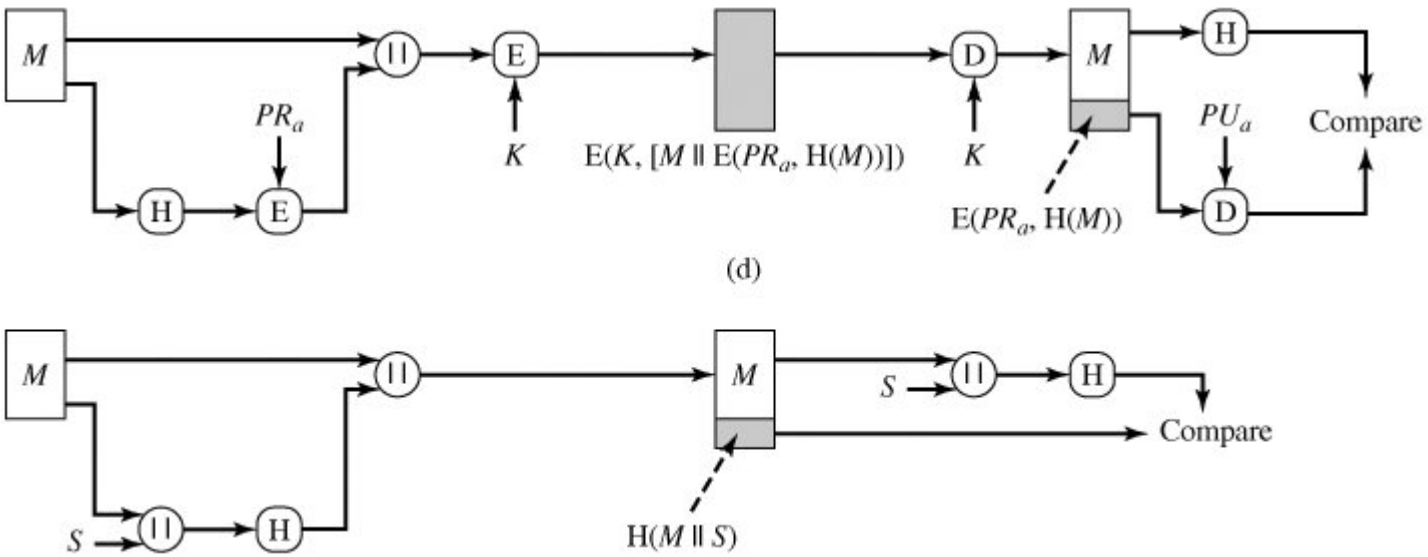


Autenticação

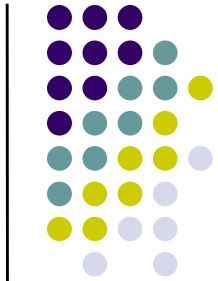
Mecanismos de Autenticação



- **Usos básicos de Hash**

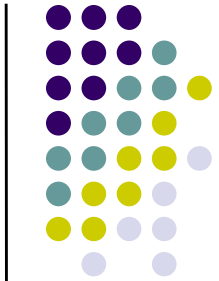


Algumas Funções Hash



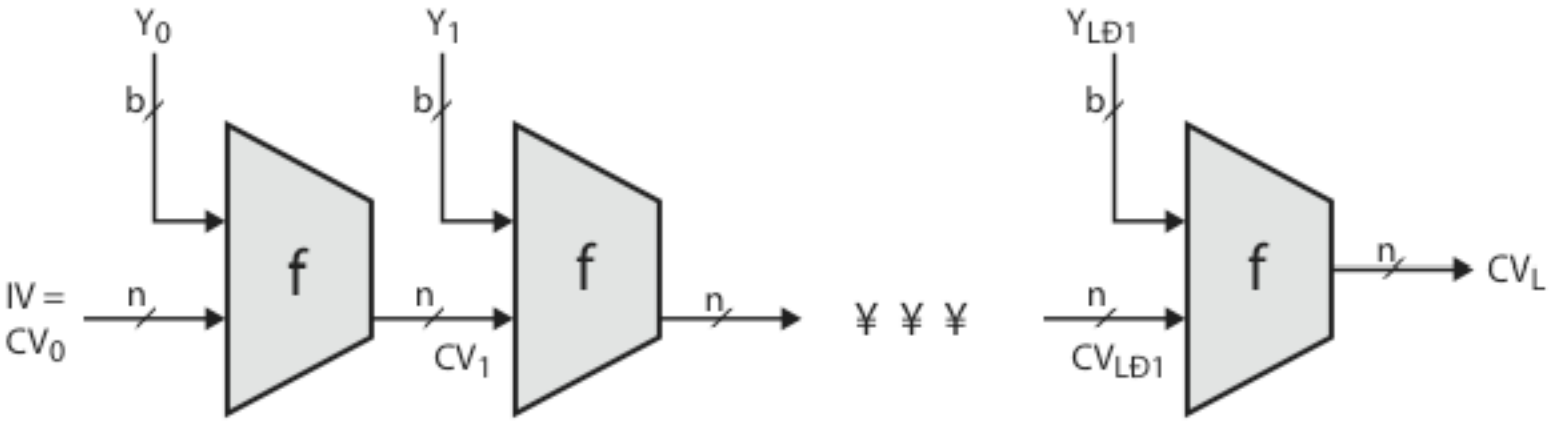
- SHA, SHA-1 e SHA2(512) SHA3
- Whirlpool
- Outros: MD2, MD4, MD5, MD6

Hash and MAC



- Funções Hash
  - Condensam uma mensagem de tamanho arbitrário para tamanho fixo processando mensagem em blocos por meio de alguma função de compactação personalizada ou baseada em codificação de bloco.
- Message Authentication Code (MAC)
  - Autenticador de tamanho fixo para alguma mensagem para fornecer autenticação para mensagem usando o modo de cifra de bloco ou a função hash.

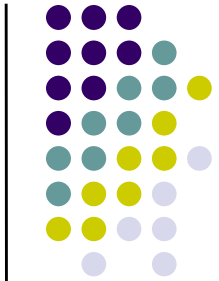
Algoritmo Hash: estrutura



IV = Initial value
CV_i = chaining variable
Y_i = ith input block
f = compression algorithm

L = number of input blocks
n = length of hash code
b = length of input block

Secure Hash Algorithm



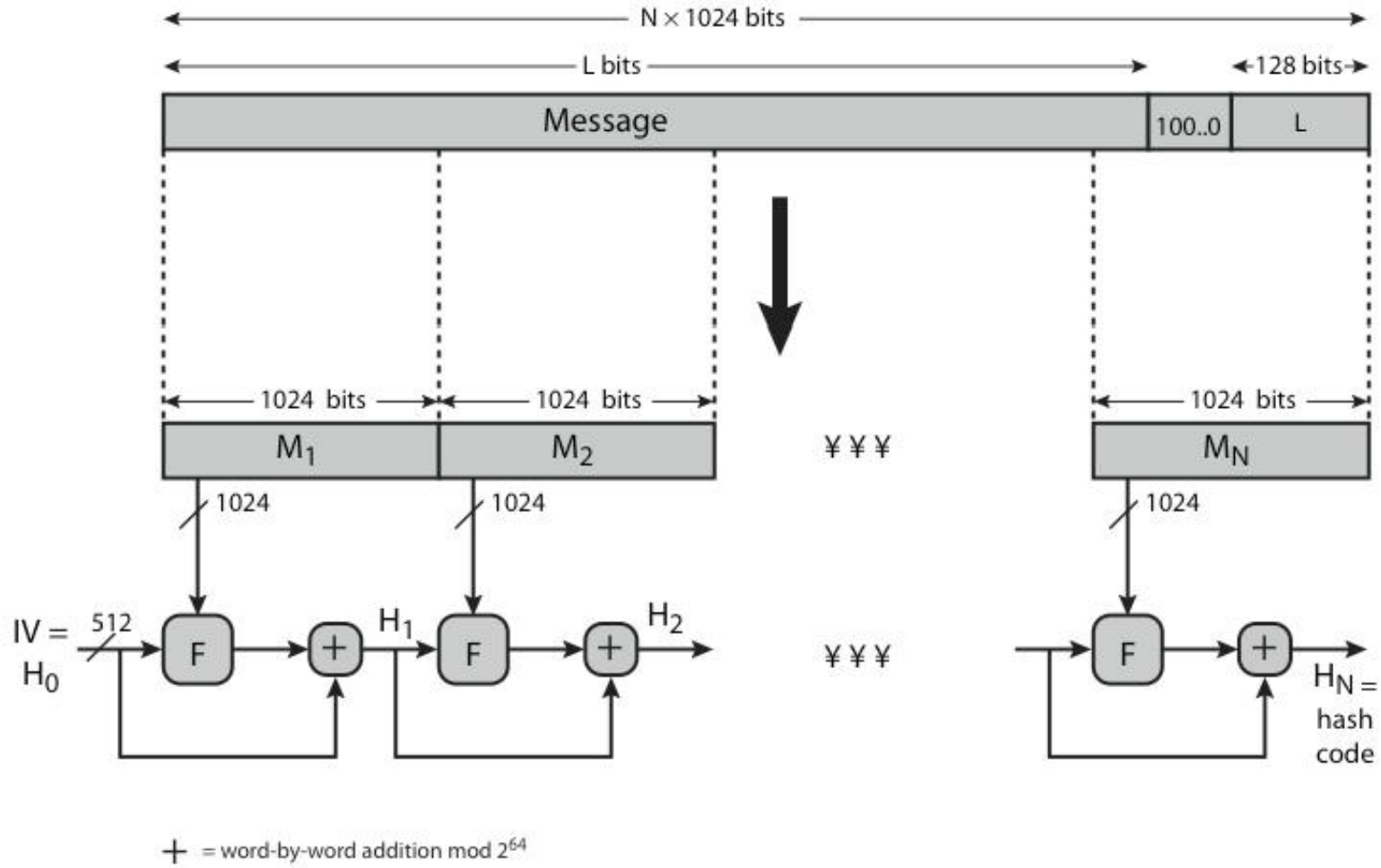
- SHA originalmente projetado pelo NIST & NSA em 1993
- Foi revisado em 1995 como SHA-1
- Padrão dos EUA para uso com o esquema de assinatura do DSA
  - Padrão é FIPS 180-1 1995, também Internet RFC3174
  - O algoritmo é SHA, o padrão é SHS
- Baseado no design do MD4 com diferenças
- Originalmente com hash de 160 bits
- Resultados de 2005 sobre a segurança do SHA-1 levantaram preocupações sobre seu uso em algumas aplicações .
- Sucessores : SHA-2 (2001) e SHA-3 (2015)

SHA - Variações

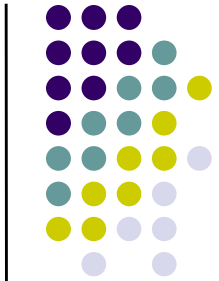
Alg.	Variação	Tamanho do Digest	Tamanho do Bloco	Tamanho da Mensagem	Rounds
	SHA-0	160	512	$2^{64} - 1$	80
	SHA-1				
SHA-2	***SHA-224***	224	512	$2^{64} - 1$	64
	SHA-256	256			
	SHA-384	384	1024	$2^{128} - 1$	80
	SHA-512	512			
	SHA-512/224	224			
	SHA-512/256	256			
SHA-3	***SHA3-224***	224	1152	Sem limites	24
	SHA3-256	256	1088		
	SHA3-384	384	832		
	SHA3-512	512	576		

*Tamanhos em bits

SHA-512 Overview

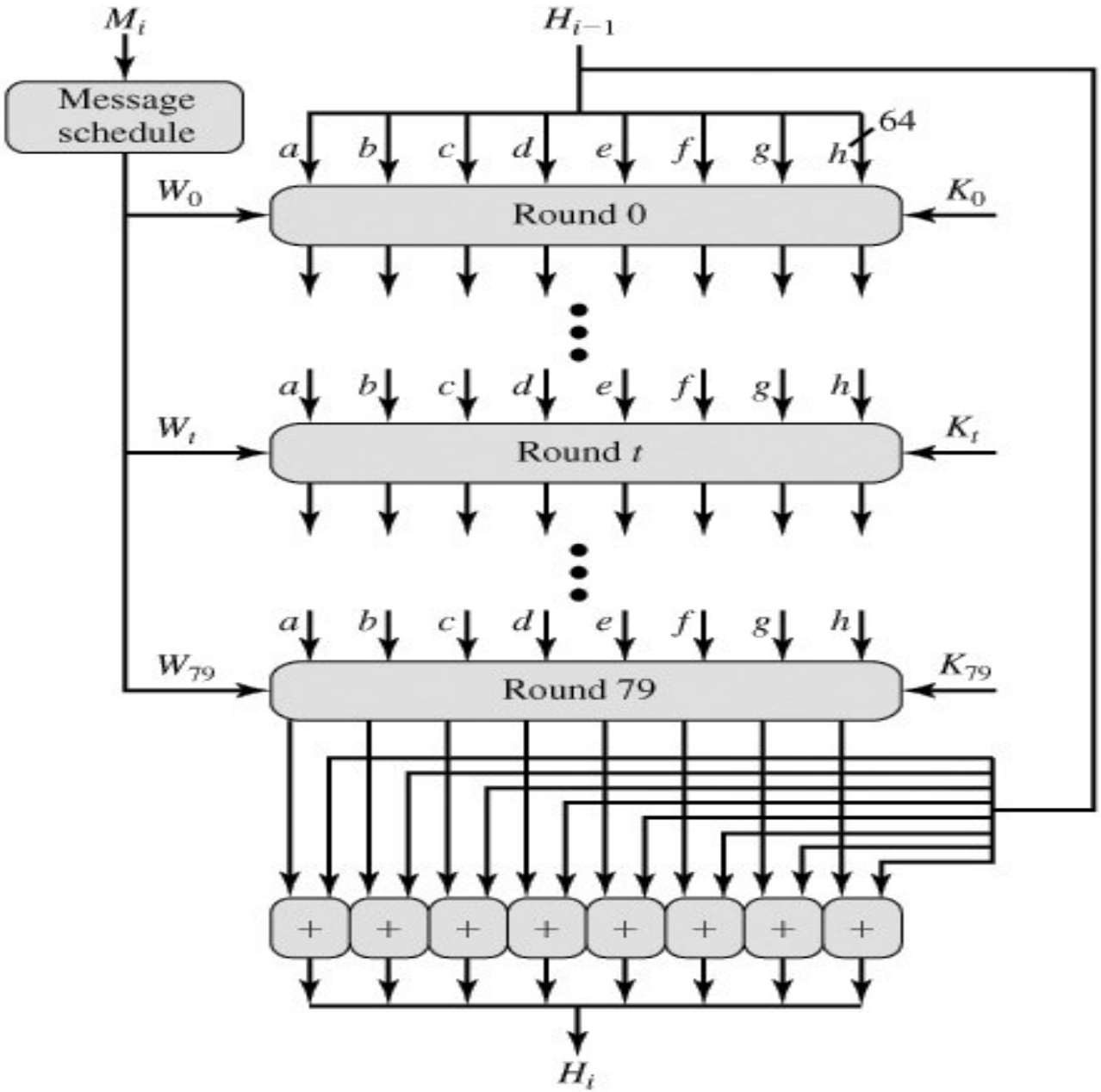


SHA-512 Função de Compressão

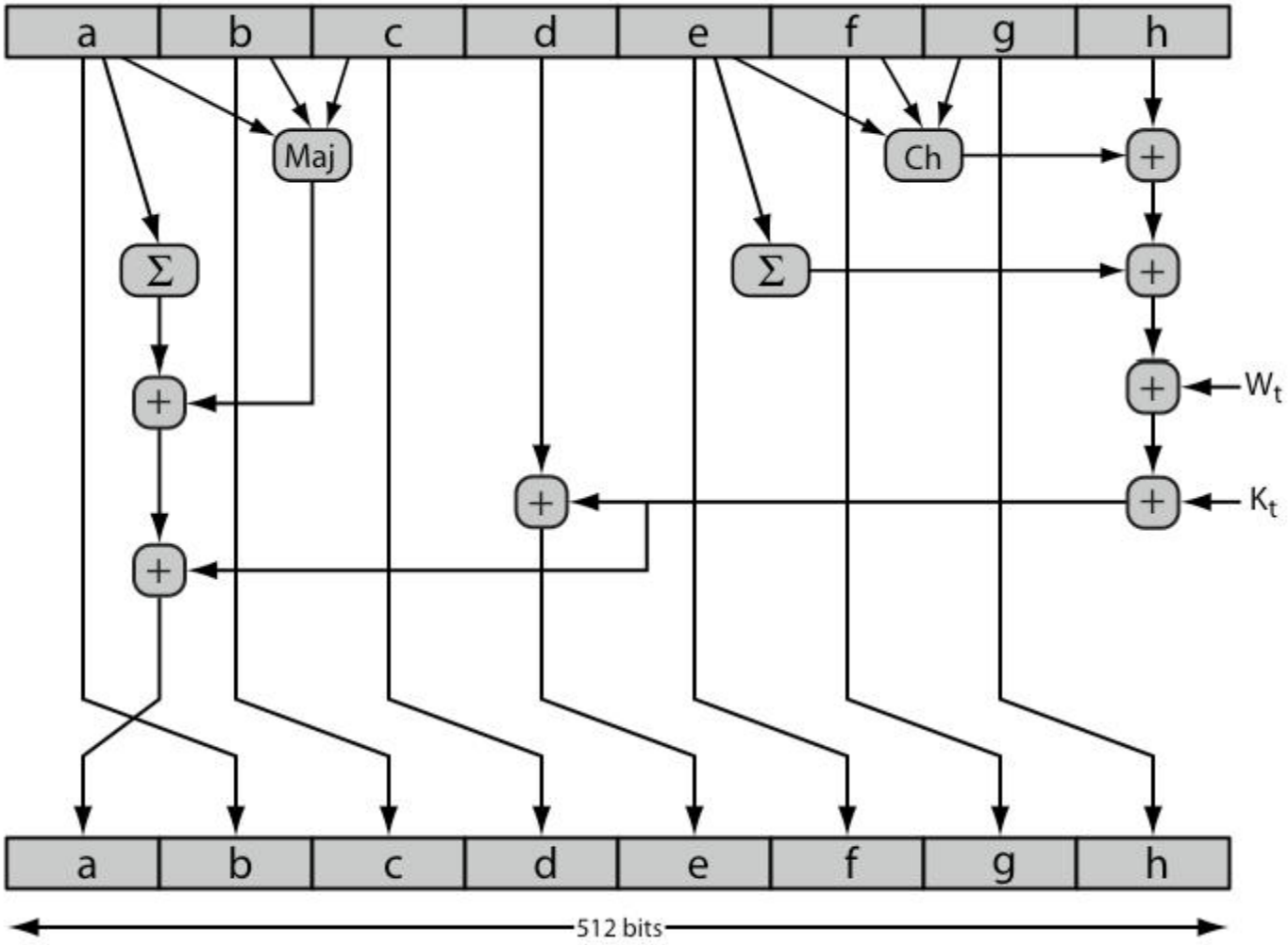


- Coração do algoritmo
- Processa mensagens em blocos de 1024-bits
- Consiste em 80 rounds
  - update um buffer de 512-bits
  - usa um valor de 64-bits, Wt derivado do bloco atual
  - e uma constante baseada na raiz cúbica dos primeiros 80 números primos

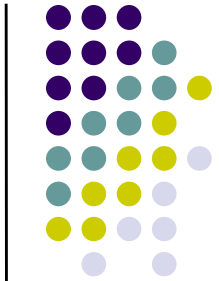
SHA-512 Função de Compressão



SHA-512: um Round



SHA-512 Um Round



$$T_1 = h + \mathrm{Ch}(e,f,g) + \left(\sum_1^{512} e\right) + W_t + K_t$$

$$T_2 = \left(\sum_0^{512} a\right) + \mathrm{Maj}(a,b,c)$$

$$a = T_1 + T_2$$

$$b = a$$

$$c = b$$

$$d = c$$

$$e = d + T_1$$

$$f = e$$

$$g = f$$

$$h = g$$

where

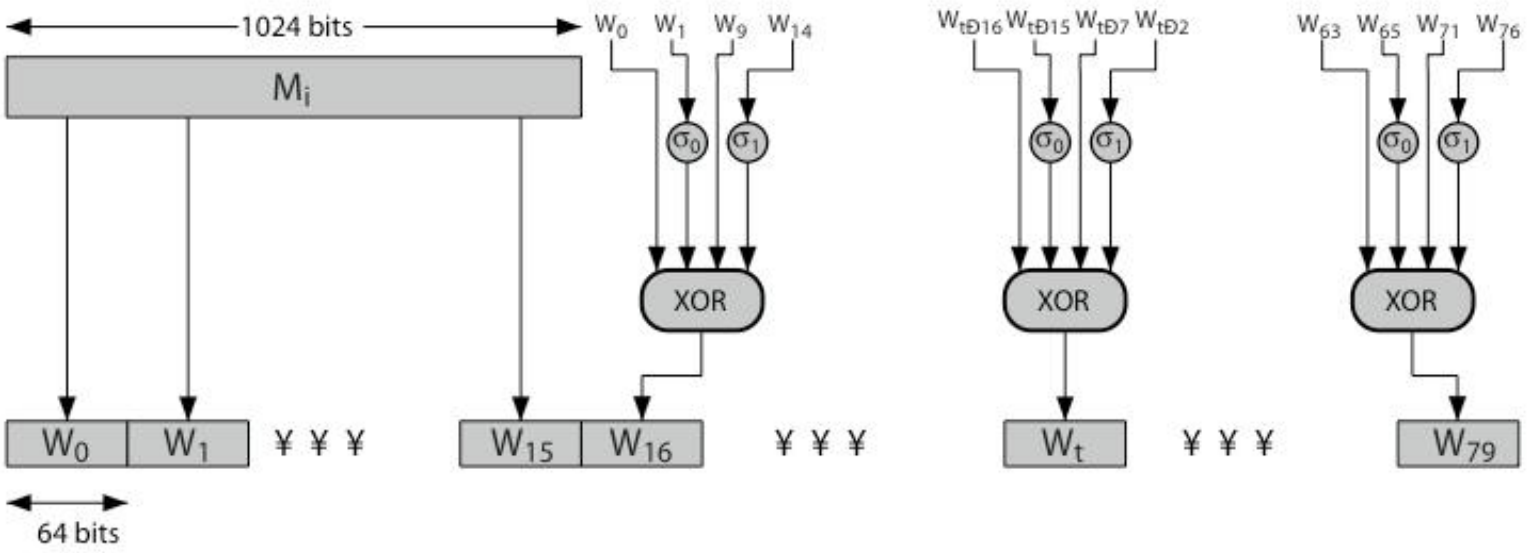
t	=step number; $0 \leq t \leq 79$
Ch(e, f, g)	= (e AND f) $\oplus$ (NOT e AND g) *the conditional function: If e then f else g*
Maj(a, b, c)	= (a AND b) $\oplus$ (a AND c) $\oplus$ (b AND c) *the function is true only of the majority (two or three) of the arguments are true.*

$$\left(\sum_0^{512} a\right) = \mathrm{ROTR}^{28}(a) \oplus \mathrm{ROTR}^{34}(a) \oplus \mathrm{ROTR}^{39}(a)$$

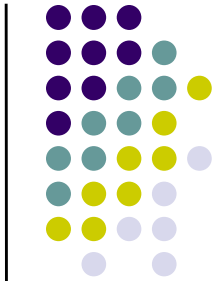
$$\left(\sum_1^{512} e\right) = \mathrm{ROTR}^{14}(e) \oplus \mathrm{ROTR}^{18}(e) \oplus \mathrm{ROTR}^{41}(e)$$

$\mathrm{ROTR}^n(x)$	= circular right shift (rotation) of the 64-bit argument x by n bits
W_t	= a 64-bit word derived from the current 512-bit input block
K_t	= a 64-bit additive constant
+	= addition modulo 2^{64}

SHA-512 – 80 Wt inputs

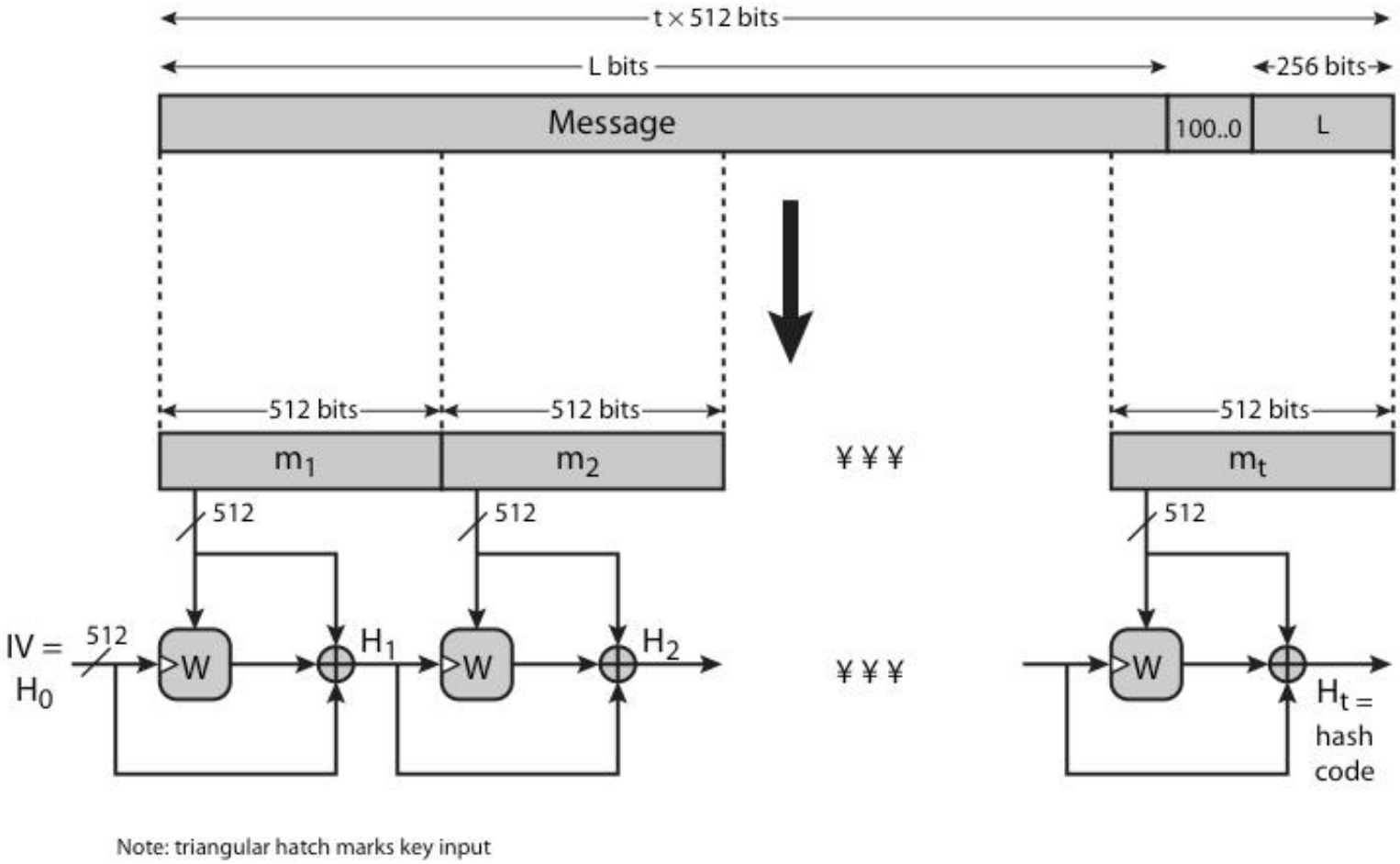


Whirlpool



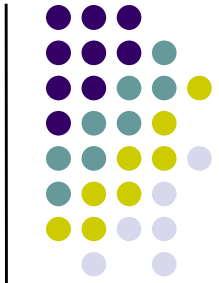
- Aprovado pelo projeto europeu NESSIE
- Usa uma modificação do AES como função de compressão
- Aborda preocupações sobre o uso de cifras de bloco como hash (Reversibilidade, tamanho do hash)
- Desempenho comparável a algoritmos dedicados hash, como o SHA.

Whirlpool Overview



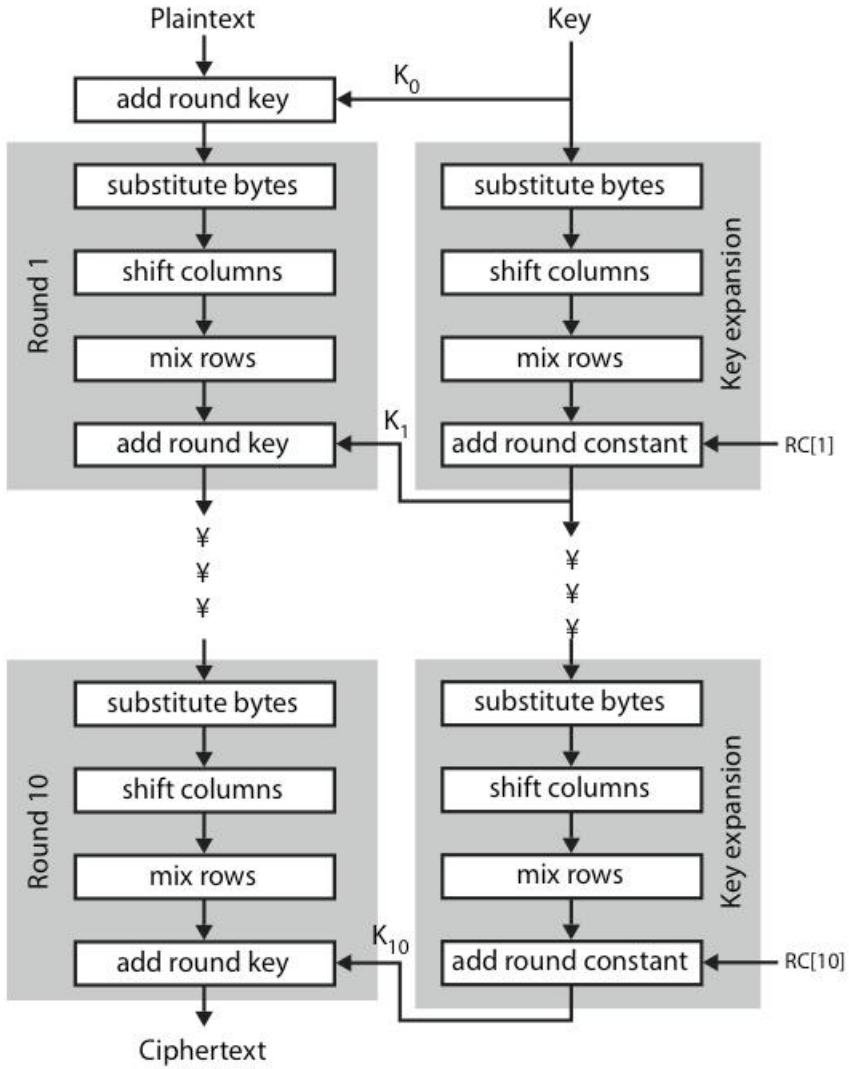
Note: triangular hatch marks key input

Whirlpool Block Cipher W

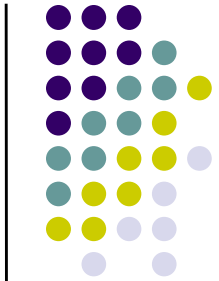


- Projetado especificamente para uso de função hash com segurança e eficiência de AES
- mas com tamanho de bloco de e chave de 512 bits e, portanto, estrutura semelhante a hash e funções do AES, mas
  - tem10 rounds
  - diferentes S-box & valores

Whirlpool Block Cipher W



Hash com Chaves como MACs



- Queremos um MAC baseado em funões hash

- Funções hash são geralmente mais rápidas
- Códigos de hash amplamente conhecidos

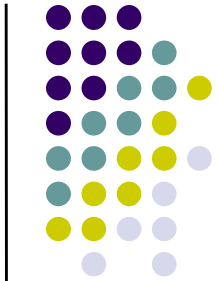
- Incluí-se uma chave ao processo de hash

-Proposta original:

- `KeyedHash = Hash(Key|Message)`

- Fraqueas foram encontradas nesse modelo levando ao desenvolvimento de outras propostas como o HMAC e o CMAC.

HMAC



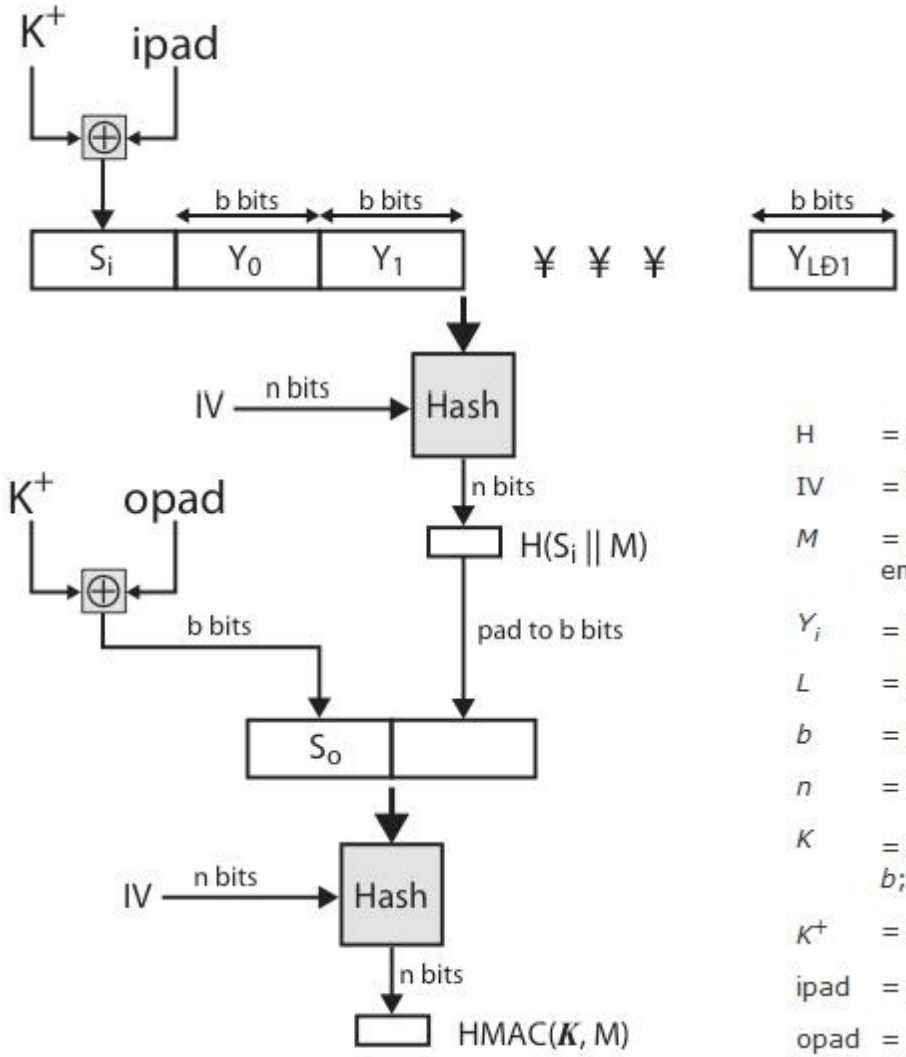
-Especificado pelo RFC2104 usa a função hash na mensagem:

$$HMAC_K = Hash[(K^+ \text{ XOR } opad) \,||\, Hash[(K^+ \text{ XOR } ipad)||M)]]$$

em que K^+ é a chave key ajustada ao tamanho desejado (padding), e opad e ipad sendo valores constantes utilizados nesse ajuste.

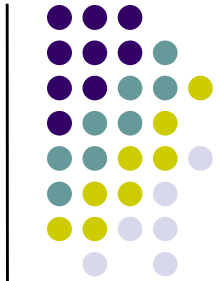
- Qualquer função hash pode ser utilizada: eg. MD5, SHA-1, RIPEMD-160, Whirlpool

HMAC Overview



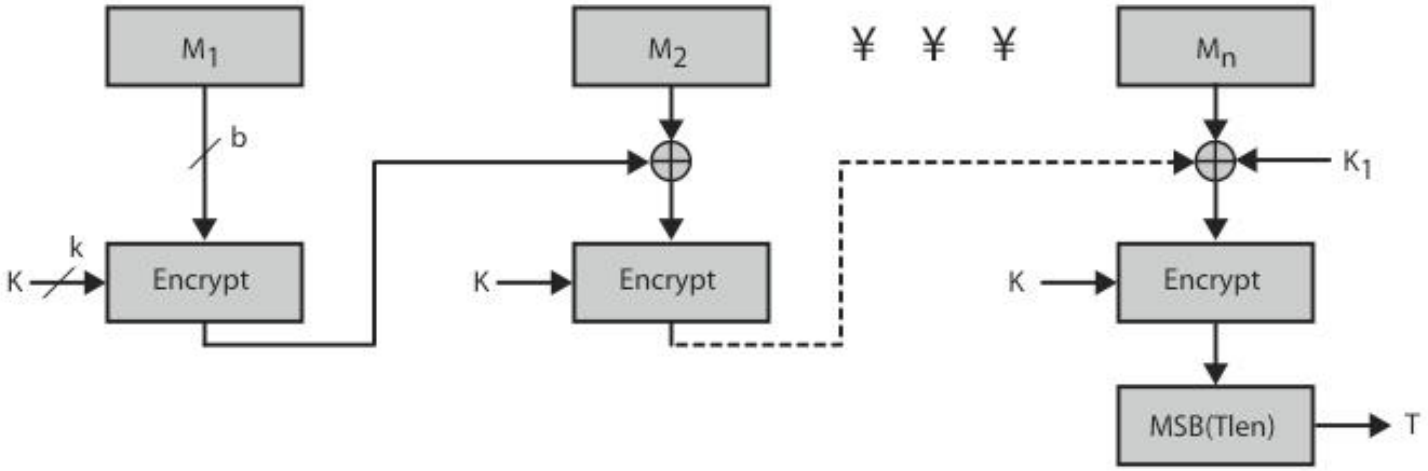
H	= embedded hash function (e.g., MD5, SHA-1, RIPEMD-160)
IV	= initial value input to hash function
M	= message input to HMAC(including the padding specified in the embedded hash function)
Y_i	= *i*th block of *M*, $0 \leq i \leq (L - 1)$
L	= number of blocks in *M*
b	= number of bits in a block
n	= length of hash code produced by embedded hash function
K	= secret key recommended length is $\geq n$; if key length is greater than *b*; the key is input to the hash function to produce an *n*-bit key
K^+	= *K* padded with zeros on the left so that the result is *b* bits in length
ipad	= 00110110 (36 in hexadecimal) repeated *b*/8 times
opad	= 01011100 (5C in hexadecimal) repeated *b*/8 times

CMAC

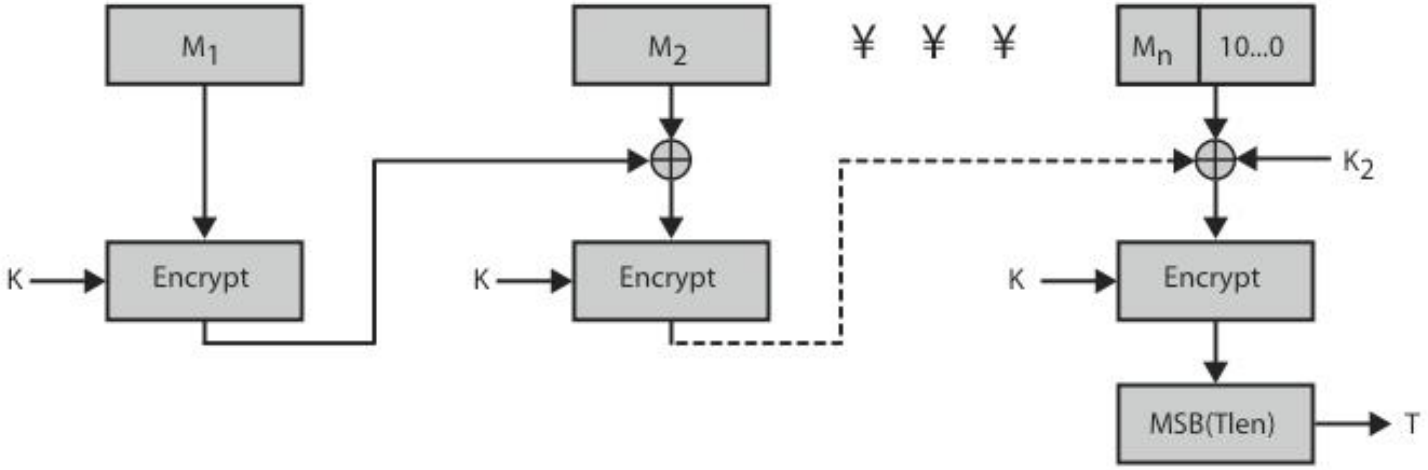


- Também chamado de DAA (CBC-MAC)
- AES A 3-DES
- Cipher-based Message Authentication Code (CMAC)
- NIST SP800-38B

CMAC – Visão Geral



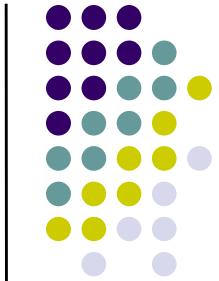
(a) Message length is integer multiple of block size



(b) Message length is not integer multiple of block size

Figure 12.12 Cipher-Based Message Authentication Code (CMAC)

Atividade



- Visite o site: http://pt.wikipedia.org/wiki/HMAC
- Implemente:
  - Um método ou função que realize o cálculo do HMAC para os exemplos de texto apresentados no site.
  - Confira se os resultados batem com os apresentados no site.