Kriptográfia

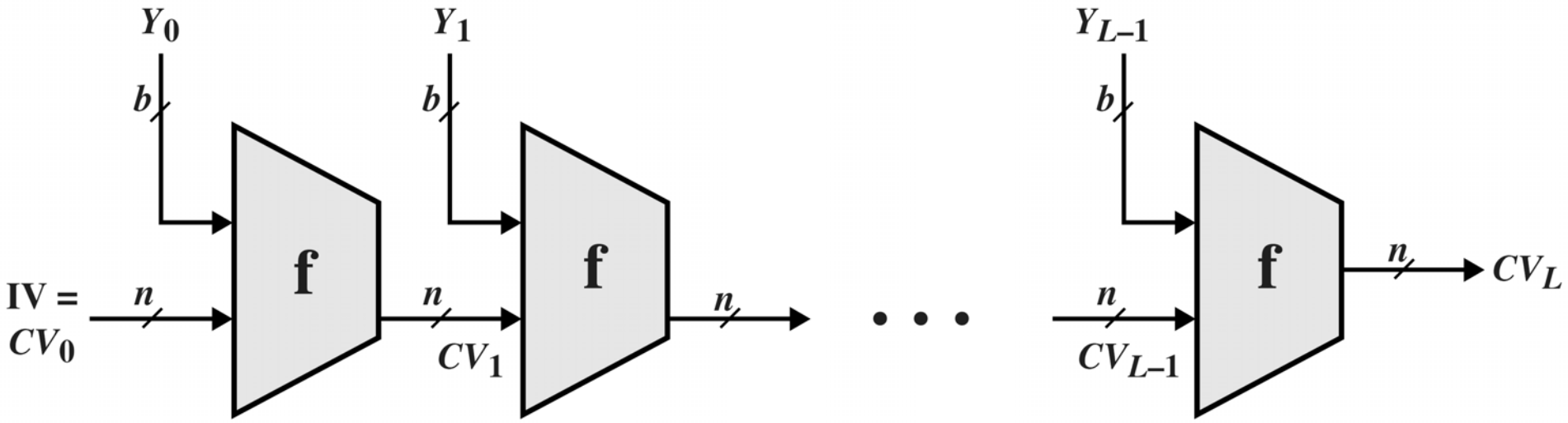
Tizedik előadás
SHA, Whirlpool, HMAC és CMAC

Németh L. Zoltán
SZTE, Számítástudomány Alapjai Tanszék
2008 ősz

Hash és MAC algoritmusok

- Hash Függvények
  - tetszőleges méretű adatot fix hosszúságúra sűrítenek
  - az üzenetet blokkonként dolgozzák fel
  - valamilyen **tömörítő fgv-en** (compression function) alapulnak, melyek
    - vagy kifejezetten erre a célra készültek (pl. SHA)
    - vagy blokktitkosítóra épülnek (Whirlpool /az AES-re/)
- Üzenethitelesítő kódok (MAC)
  - az üzenetből (kulccsal) fix hosszú hitelesítő kódot készít
  - ez a kód hitelesíti az üzenetet
    - vagy hash függvény alapúak (pl. HMAC)
    - vagy blokktitkosító alapúak (CBC módban, pl. CMAC)

Hash algoritmusok általános struktúrája



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

Az MD5

- az egyik legelterjedtebb hash függvény
- **128 bites** hash értéket ad
  - ez 32 db hexadecimális számjegy (16 pár)
- tervezője Ronald Rivest, 1991
- az MD4 helyettesítésére
- letöltéseknél ellenőrző összegként használt
  - persze ha a hash érték ugyanabból a forrásból származik (szintén letöltött), akkor csak véletlen hibák ellen véd, rosszindulatú hamisítások ellen **nem**
- jelszavaknak hash értékként való tárolására is használják
  - számos táblázat van, könnyű jelszavak MD5 hash értékeivel,
  - a táblázatok fejlettebb változata a szivárványtáblák (rainbow tables), melyek mérete jóval kisebb

Az MD5 biztonsága

- 128 bit -> születésnapi támadás -> 2^{64} **túl kicsi!**
- brute force módon spec. hardverrel ütközés található: (MD5CRK osztott projekt)
- 1996, 2004: kriptoanalízissel is ütközés található
- 2006: **ma már egy notebookon egy perc alatt két ellentétes tartalmú, de azonos MD5 hash értékű üzenet/program/tanúsítvány generálható**
- de egy adott üzenthez ugyanolyan hash értékű másikat keresni továbbra is nehéz
- ezért az őt alkalmazó hitelesítési protokollok és HMAC kódok nincsenek közvetlenül veszélyben
- peresze át kell róla térni újabb algoritmusokra

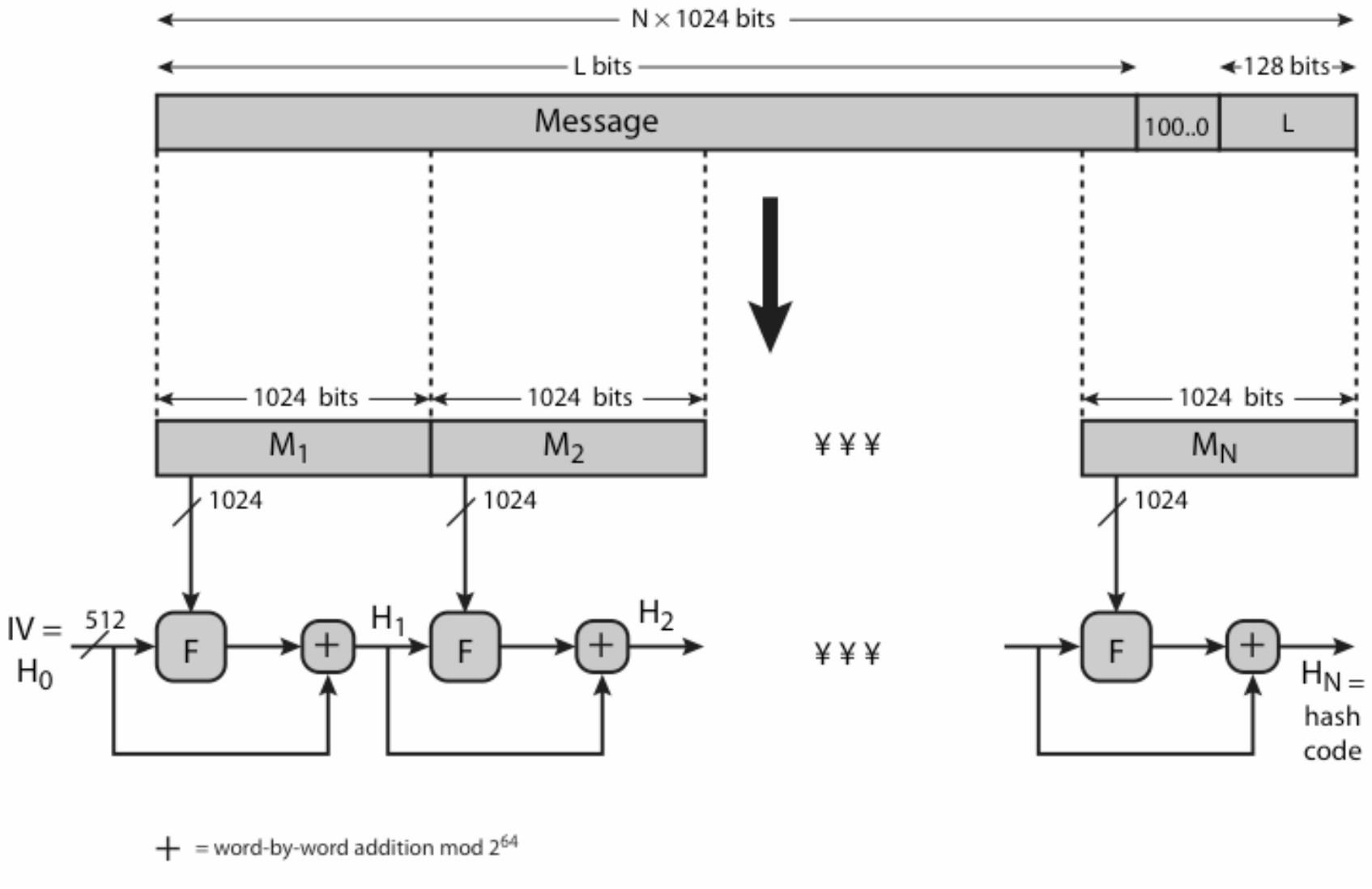
SHA, Secure Hash Algorithm

- SHA eredeti tervezői NIST & NSA 1993-ban
- felülvizsgálás1995-ben -> SHA-1 név
- USA szabvány FIPS 180, 180-1, 180-2, Internet RFC3174 is
- SHA – az algoritmus, SHS – a szabvány
- Alapja a **DSA** (Digital Signature Algorithm)-nak
- az MD4 hasfüggvény terevei alapján
- néhány fontos különbséggel
- 160-bites hash értéket ad
- 2005-ben: ütközés (azaz két egyforma SHA hash értékű fájl) található 2^{69} művelettel (a 2^{80} helyett), ami jövőbeli alkalmazását megkérdőjelezi
- helyette SHA-2 : több változat, hosszabb hash értékekkel

A felülvizsgált SHA

- NIST bocsátotta ki: FIPS 180-2 (2002-ben)
- az SHA-2 három új változata, bithossz szerint
  - SHA-256, SHA-384, SHA-512
- CÉL: erőssége illeszkedjen az AES által nyújtott nagyobb biztonsághoz
- struktúrája és részletei az SHA-1-hez hasonlóak
- így analízisük is hasonló
- de jóval biztonságosabb
- 2005 NIST: 2010-ig át kell térni rájuk az SHA-1-ről

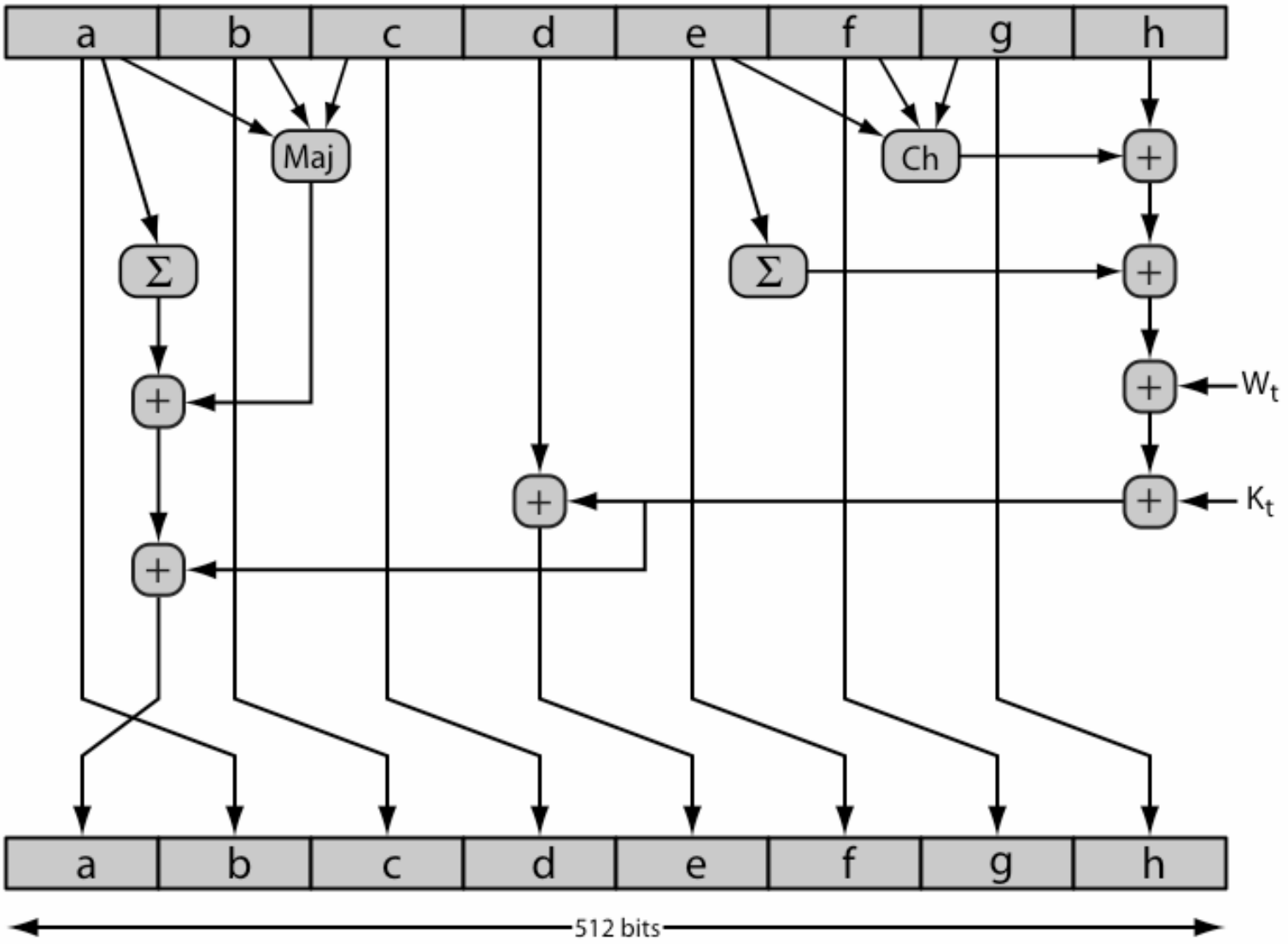
SHA-512 áttekintés



Az SHA-512 tömörítő függvénye

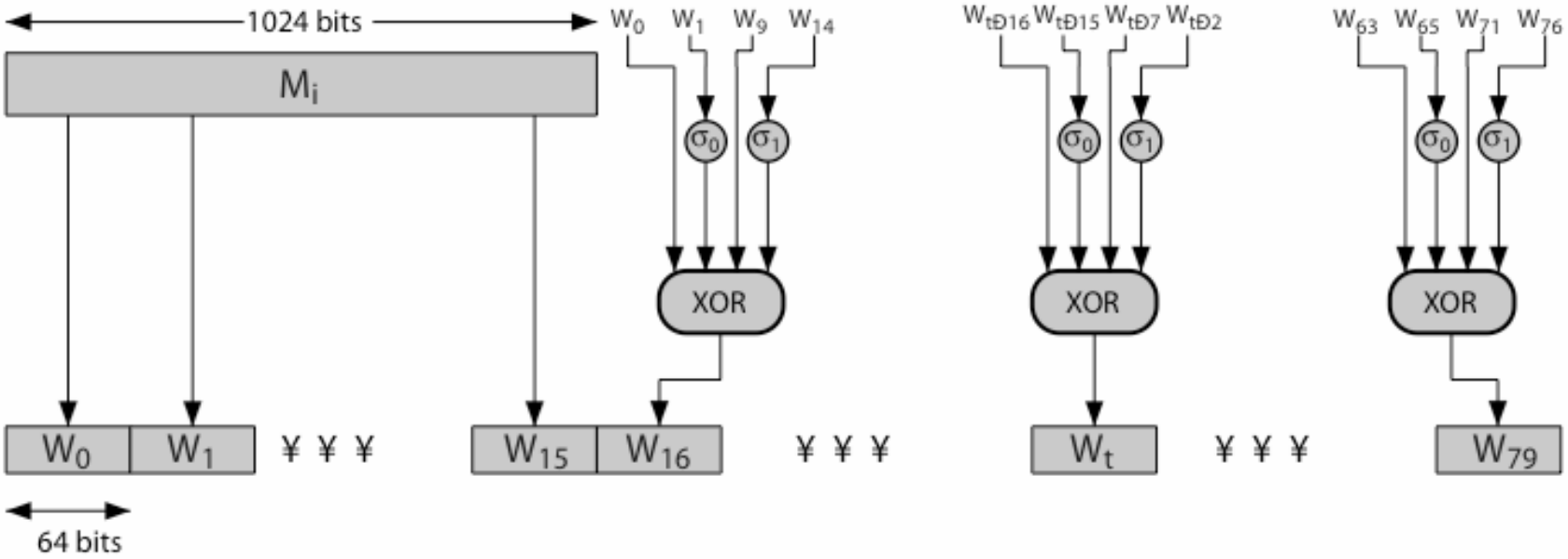
- ez az algoritmus szíve (**F**)
- az üzenetet 1024-bites blokkokra bontja
- 80 körön keresztül:
  - aktualizálja az 512-bites puffert
    - 8db x 64bites szó (regiszterek): a,b,c,d,e,f,g,h
    - egy 64-bites W_t értéket, ami az aktuális üzenetblokkból származik
    - és egy körkonstansot, ami az első 80 prímszám köbgyökéneinek a törtrészének az eleje (minden körben más)

Az SHA-512 körfüggvénye



Ezt ismétli körönként nyolcvanszor!

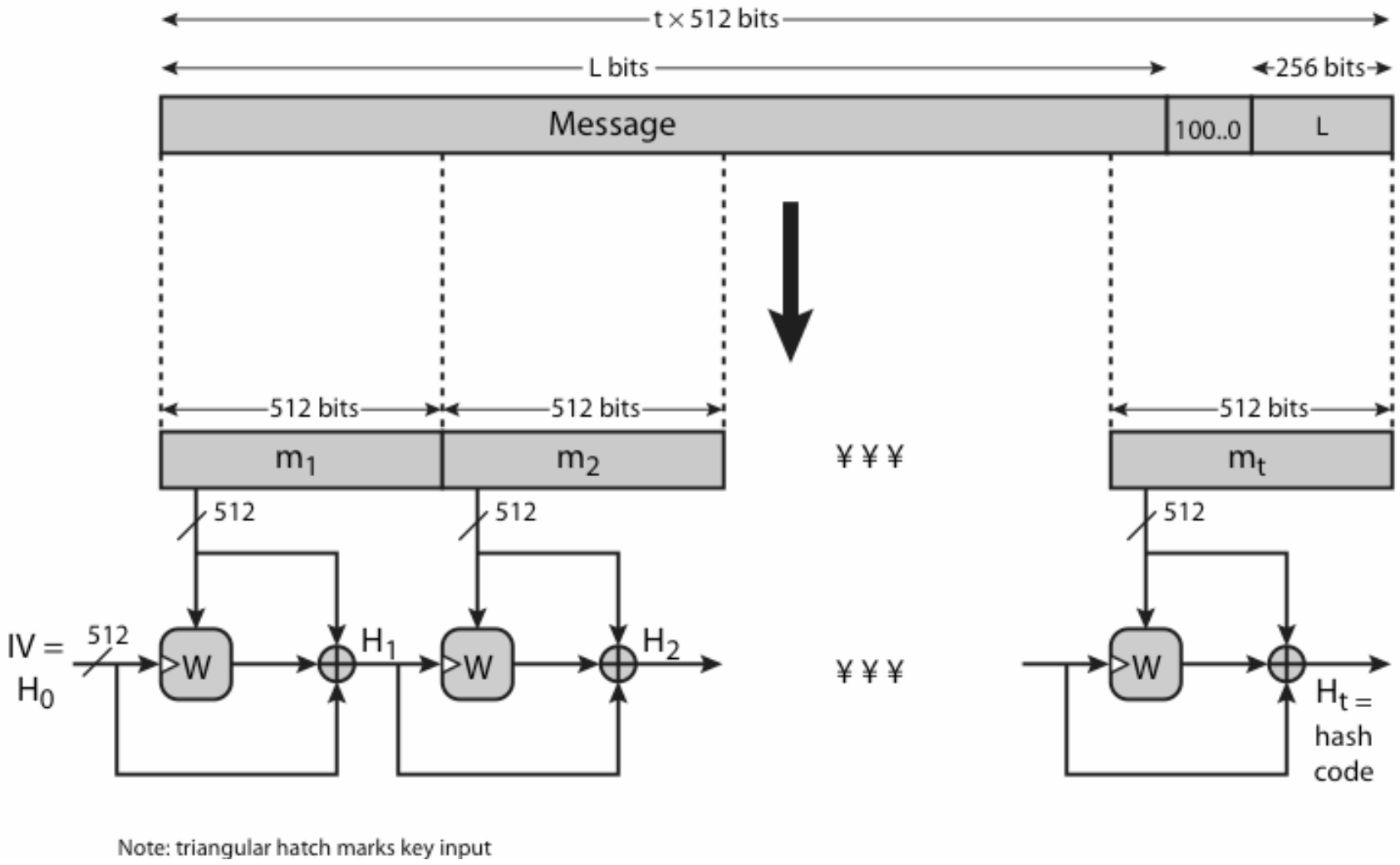
A W_t értékek számolása az üzenetblokkból



Whirlpool

- az európai NESSIE) project által támogatott másik **hash függvény**
- NESSIE = New European Schemes for Signatures, Integrity, and Encryption
- tömörítő függvényként az AES módosított változatát használja
- teljesítménye és biztonsága összevethető, (ha nem jobb), mint az MD és SHA hasfüggvény családnak

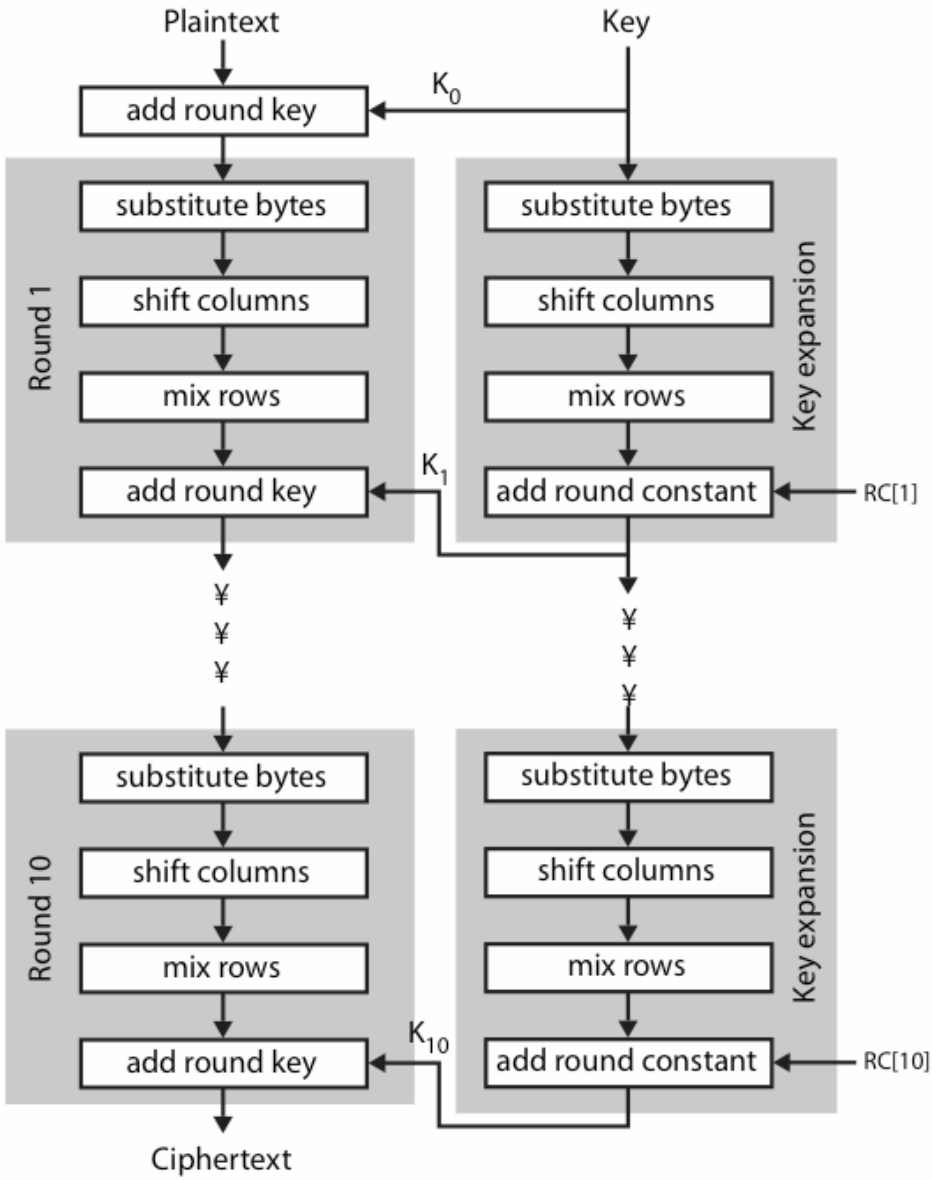
Whirlpool áttekintés



A W Whirlpool blokktitkosító

- kifejezetten hash függvény készítéséhez használt titkosító
- az AES biztonságával és hatékonyságával
- de a blokkméret 512-bit, és így a hash érték is
- struktúrája és függvényei hasonlóak az AES-hez de
  - a bemenetet soronként kezeli
  - 10 körrel rendelkezik
  - a GF(2^8) testet más polinommal adja meg
  - más S-box, más tervezési módszerrel

A W Whirlpool blokktitkosító



A Whirlpool biztonsága és teljesítménye

- Whirlpool elég új javaslat
- ezért kevés tapasztalat áll rendelkezésre vele kapcsolatban
- de sokminden, ami az AES-re igaz, igaz rá is
- vizsgálatok szerint : nagyobb a hardverigénye, mint az SHA-nak
- de az általa nyújtott teljesítmény is jobb

Üzenethitelesítő algoritmusok

- a hash függvényekkel szemben **kulcs segítségével** képezik az üzenetből a hitelesítő MAC kódot
- eredeti javaslat: használjunk hash függvényeket

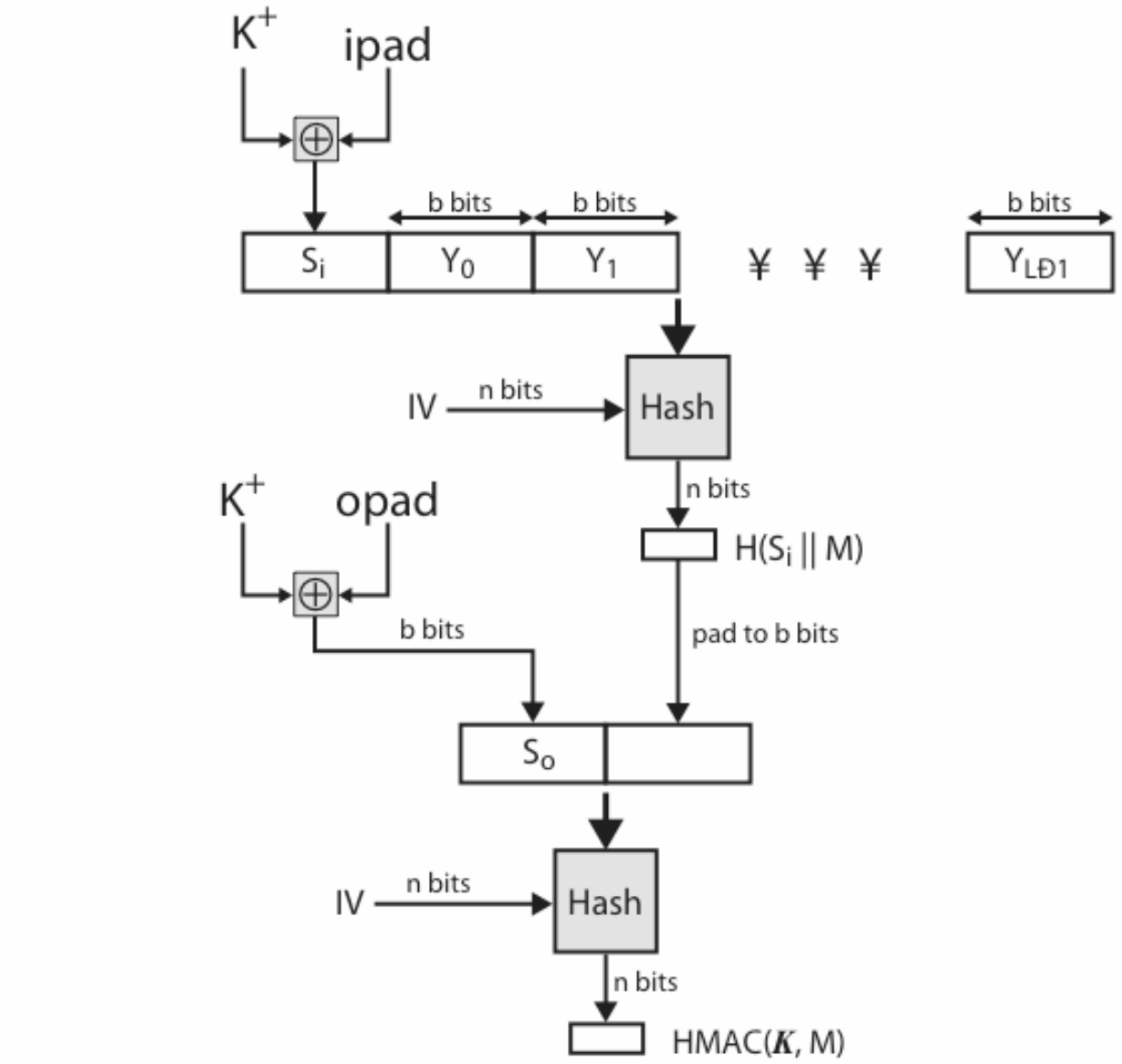
  `MAC(kucs, üzenet) := Hash(kulcs||üzenet)`

  - de ennek több gyenge pontja van,
  - nem biztonságos
- két általános tervezési elv:
  - **HMAC**: hash függvény alapú MAC
  - **CMAC**: titkosító alapú MAC
- Így minden konkrét hash függvénnyel (MD5, SHA-512, Whirlpool) illetve blokktitkosítóval (3DES, AES) lehet üzenethitelesítő algoritmust készíteni

HMAC

- Az RFC2104 Internet szabvány specifikálja
- $\mathbf{HMAC_K = Hash[(K^+ \text{ XOR opad}) \,||\, Hash[(K^+ \text{ XOR ipad}) \,||\, M)]]}$
- ahol K^+ a kulcs feltöltve
- opad, ipad speciális feltöltő konstansok
- a választott hash függvény, csak 3-mal több blokkra kell alkalmazni, mint, ha csak az üzenet hash értéket számolnánk
- bármely hash függvényre alkalmazható
  - pl. MD5, SHA-1, RIPEMD-160, Whirlpool
  - ez a sebesség/biztonság függvényében választható meg

HMAC áttekintés



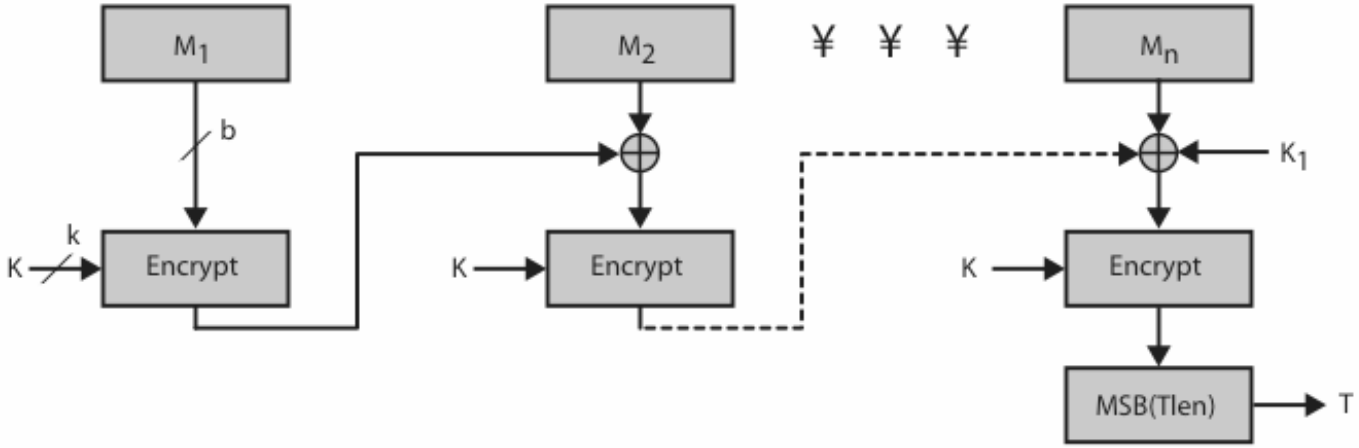
A HMAC biztonsága

- bizonyítható, hogy a biztonsága arányos a használt hash algoritmus biztonságával
- biztonságát egy üzenet HMAC kódjának sikeres hamisításához szükséges üzenet—HMAC érték párok számában mérik
- támadási típusok:
  - brute force a használt kulcs ellen (2^n)
  - születésnapi támadások ($2^{n/2}$ de ehhez nagyon sok (2^n) üzenet—HMAC érték pár kell ugyanazzal a kulccsal.)
  - **így még a 128-bites MD5-tel is biztonságos**

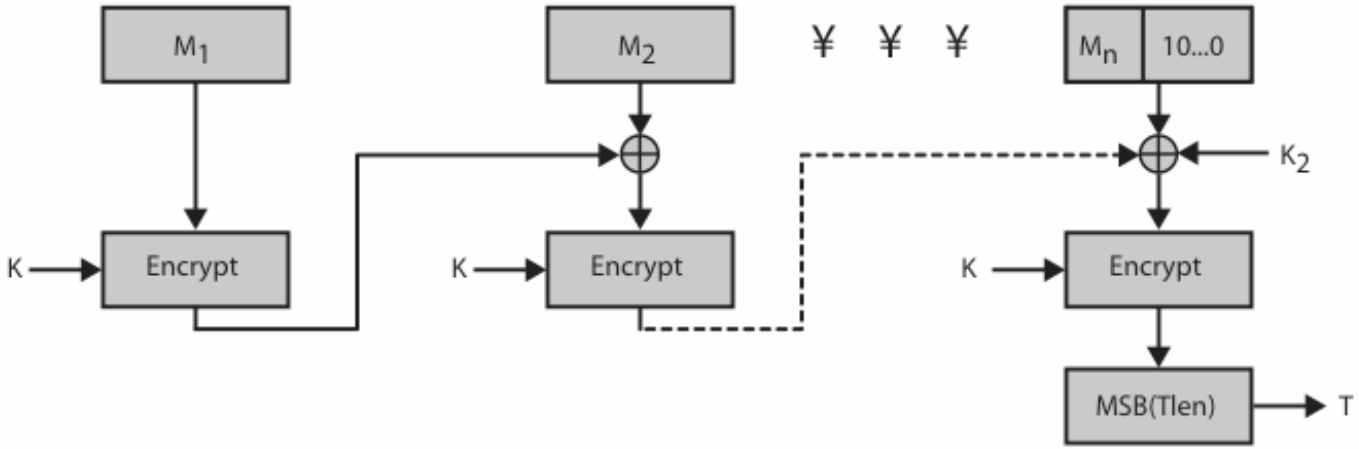
CMAC

- **CMAC** = titkosító alapú üzenethitelesítő kód
  Cipher-based Message Authentication Code
- már láttuk a **DAA**-t (CBC-MAC (DES-re alapul))
- széles körben használt (USA)
- de az üzenet mérete korlátozott
- ez kiváltható 2 kulcs használatával és feltöltéssel
- NIST szabvány: SP800-38B

CMAC áttekintés



(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)

Felhasznált irodalom

- Virrasztó Tamás: Titkosítás és adatrejtés: Biztonságos kommunikáció és algoritmikus adatvédelem, NetAcademia Kft., Budapest, 2004. Online elérhető: http://www.netacademia.net/book.aspx?id=1#

- William Stallings: Cryptography and Network Security, 4th Edition, Prentice Hall, 2006. (Chapter 12)
- Lawrie Brown előadás fóliái (Chapter 12)