Secure Hash Algorithm (SHA)

Bahan Kuliah

IF4020 Kriptografi

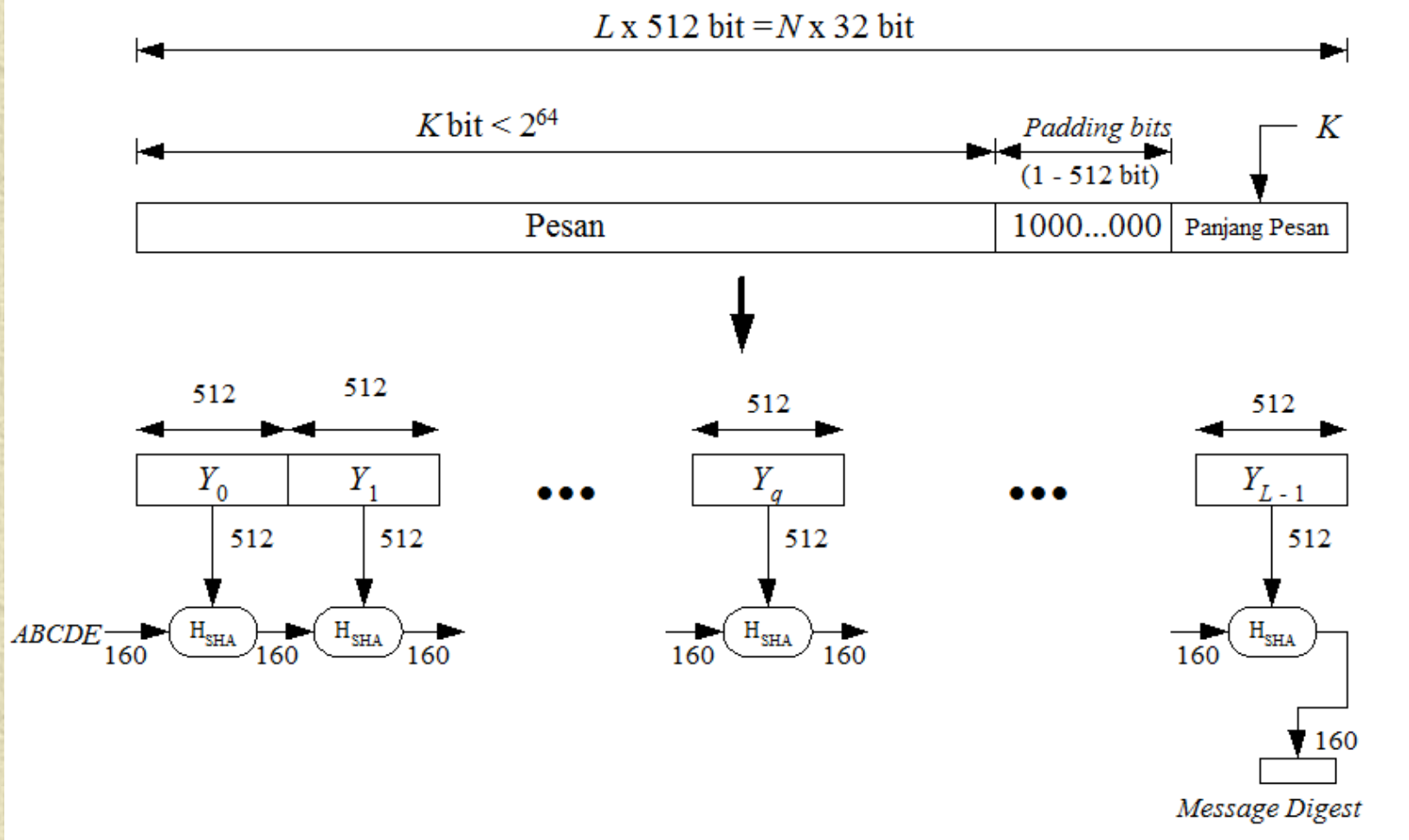
Secure Hash Algorithm (SHA)

- *SHA* adalah fungsi *hash* satu-arah yang dibuat oleh *NIST* dan digunakan bersama *DSS* (*Digital Signature Standard*).
- Oleh *NSA*, *SHA* dinyatakan sebagai standard fungsi *hash* satu-arah.
- *SHA* didasarkan pada *MD4* yang dibuat oleh Ronald L. Rivest dari *MIT*.
- Algoritma *SHA* menerima masukan berupa pesan dengan ukuran maksimum 2^{64} bit (2.147.483.648 *gigabyte*) dan menghasilkan *message digest* yang panjangnya 160 bit, lebih panjang dari *message digest* yang dihasilkan oleh *MD5*.

- *SHA* mengacu pada keluarga fungsi *hash* satu-arah.
- Enam varian *SHA*: SHA-0, SHA-1, SHA-224, SHA-256, SHA-384, SHA-512.
- SHA-0 sering diacu sebagai *SHA* saja
- Yang akan dibahas: SHA-1

		Output size (bits)	Internal state size (bits)	Block size (bits)	Max message size (bits)	Word size (bits)	Rounds	Operations	Collisions found?
SHA-0									Yes
SHA-1		160	160	512	$2^{64} - 1$	32	80	+,and,or, xor,rot	Theoretical attack (2^{51})[5]
SHA-2	*SHA-256/224*	256/224	256	512	$2^{64} - 1$	32	64	+,and,or, xor,shr,rot	No
	SHA-512/384	512/384	512	1024	$2^{128} - 1$	64	80		

Skema pembuatan *message digest* dengan *SHA-1*



Langkah-langkah pemuatan *message digest* dengan *SHA*-1

1. Penambahan bit-bit pengganjal (*padding bits*).
2. Penambahan nilai panjang pesan semula.
3. Inisialisasi penyangga (*buffer*) MD.
4. Pengolahan pesan dalam blok berukuran 512 bit.

- *SHA* membutuhkan 5 buah penyangga (*buffer*) yang masing-masing panjangnya 32 bit.
- Total panjang penyangga adalah $5 \times 32 = 160$ bit.
- Kelima penyangga *MD* ini diberi nama *A*, *B*, *C*, *D*, dan *E*. Setiap penyangga diinisialisasi dengan nilai-nilai (dalam notasi HEX) sebagai berikut:

  A = 67452301

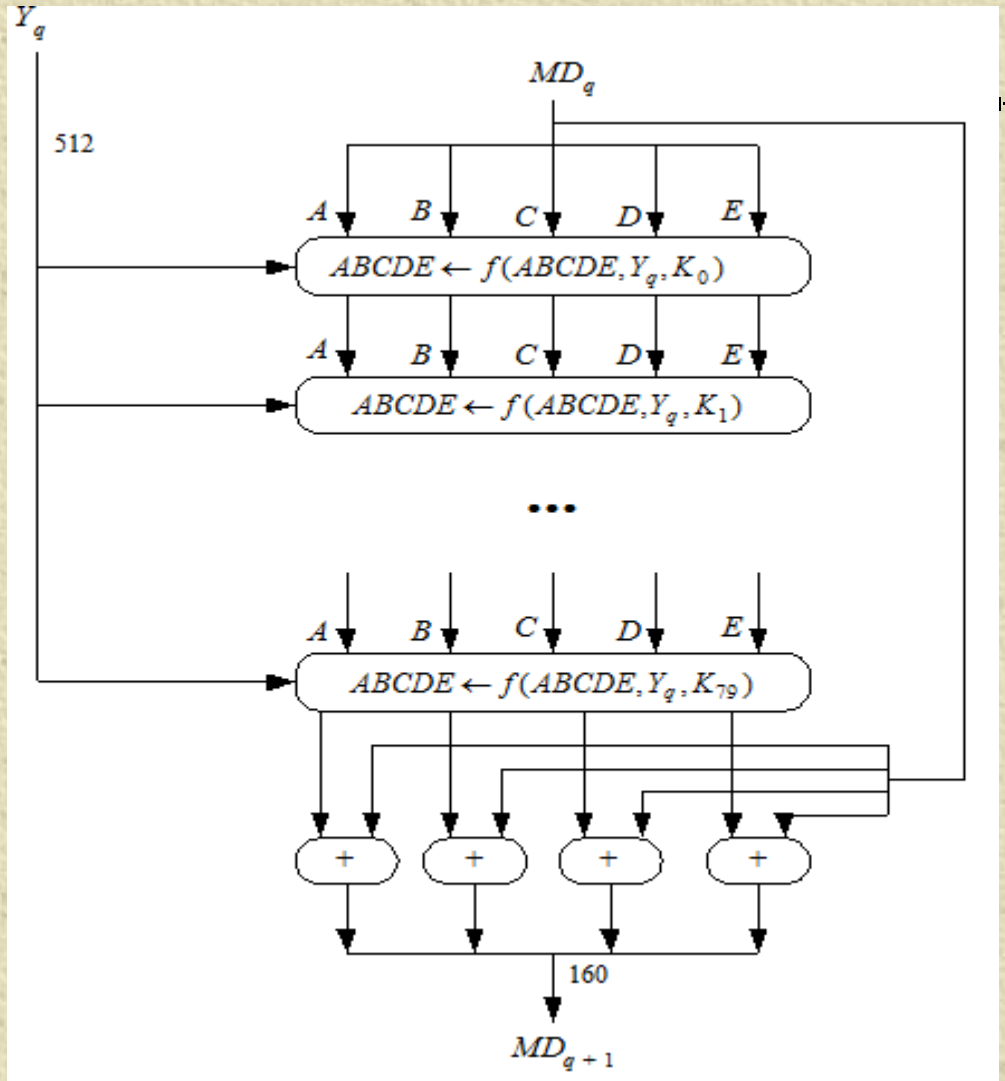
  B = EFCDAB89

  C = 98BADCFE

  D = 10325476

  E = C3D2E1F0

Pengolahan blok 512-bit (Proses H_{SHA})



- Proses H_{SHA} terdiri dari 80 buah putaran (*MD5* hanya 4 putaran)
- Masing-masing putaran menggunakan bilangan penambah K_t, yaitu:

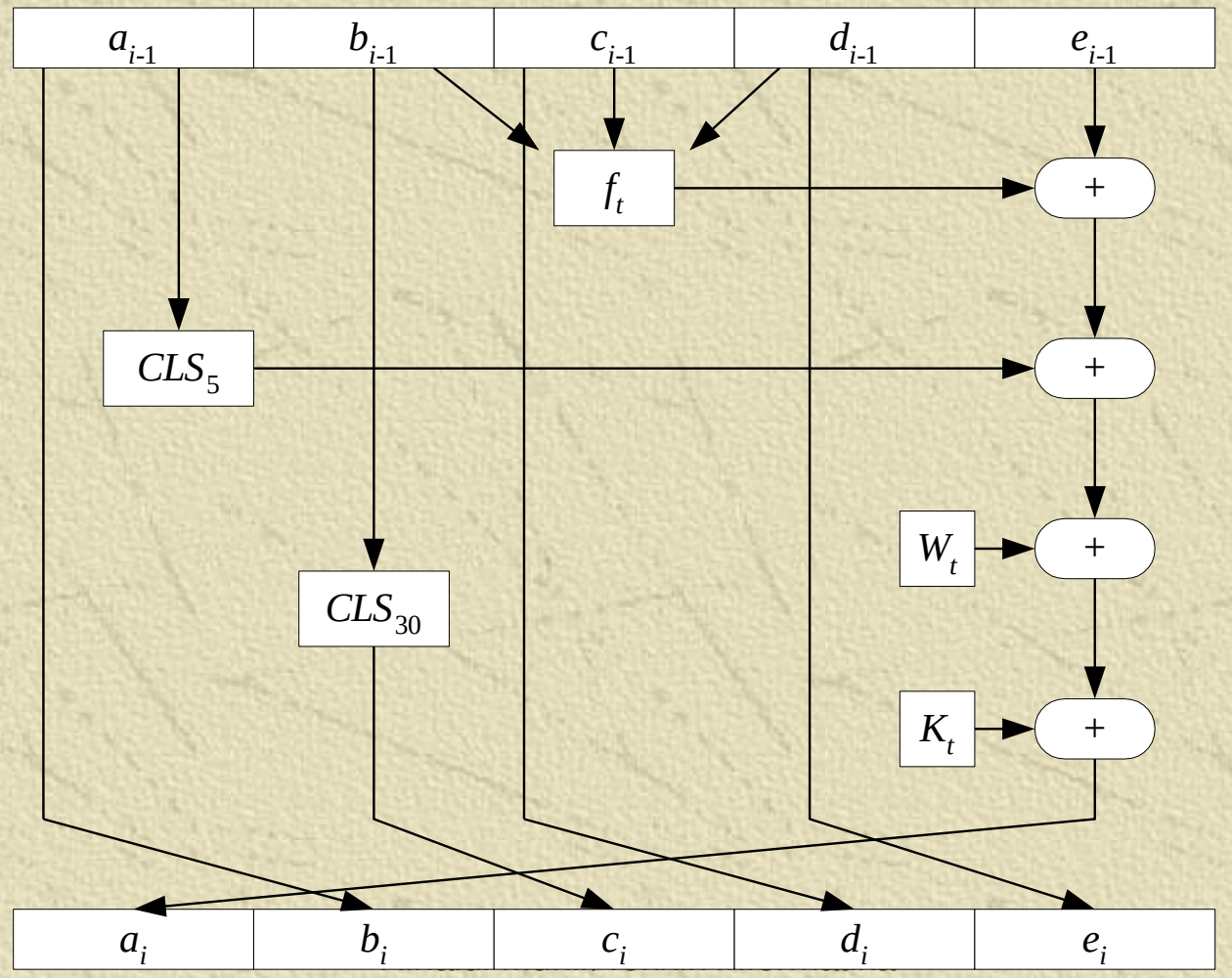
Putaran $0 \leq t \leq 19$ K_t = 5A827999

Putaran $20 \leq t \leq 39 K_t$ = 6ED9EBA1

Putaran $40 \leq t \leq 59 K_t$ = 8F1BBCDC

Putaran $60 \leq t \leq 79 K_t$ = CA62C1D6

Operasi dasar pada setiap putaran:



Tabel 1. Fungsi logika f_t pada setiap putaran

Putaran	$f_t(b, c, d)$
0 .. 19	$(b \wedge c) \vee (\sim b \wedge d)$
20 .. 39	$b \oplus c \oplus d$
40 .. 59	$(b \wedge c) \vee (b \wedge d) \vee (c \wedge d)$
60 .. 79	$b \oplus c \oplus d$

- Nilai W_1 sampai W_{16} berasal dari 16 *word* pada blok yang sedang diproses, sedangkan nilai W_t berikutnya didapatkan dari persamaan

$$W_t = W_{t-16} \oplus W_{t-14} \oplus W_{t-8} \oplus W_{t-3}$$

Kriptanalisis *SHA*-1

- Pada tahun 2005, Rijmen dan Oswald mempubliksikan serangan pada versi *SHA*-1 yang direduksi (hanya menggunakan 53 putaran dari 80 putaran) dan menemukan kolisi dengan kompleksitas sekitar 2^{80} operasi (lihat di http://eprint.iacr.org/2005/010) [WIK06]

- Pada bulan Februari 2005, Xiayoun Wang, Yiqun Lisa Yin, dan Hongbo Yo mempublikasikan serangan yang dapat menemukan kolisi pada versi penuh SHA-1, yang membutuhkan sekitar 2^{69} operasi (lihat beritanya di http://www.schneier.com/blog/archives/2005/02/sha_1broken.html) [WIK06].