

Bahan kuliah II4031 Kriptografi dan Koding

# MAC

*(Message Authentication Code)*

**Oleh: Rinaldi Munir**

Program Studi Sistem dan Teknologi Informasi  
Sekolah Teknik Elektro dan Informatika  
ITB – 2024

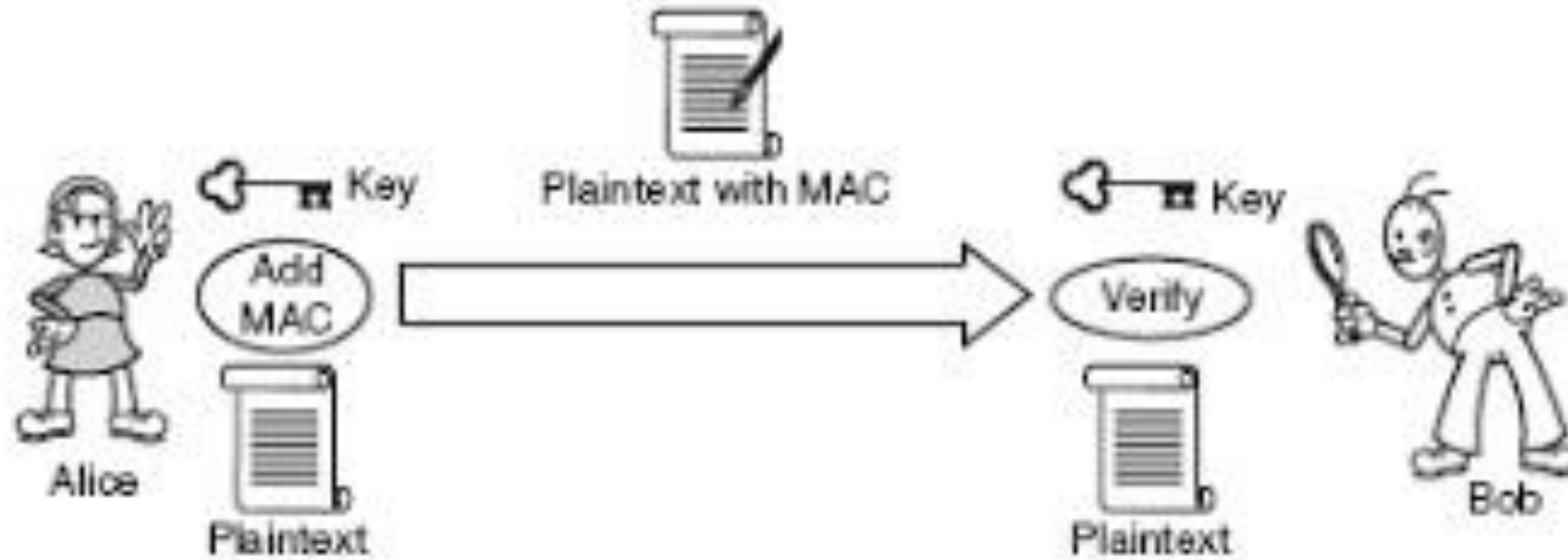


Figure 9.3 Message Authentication Code (MAC)

# Definisi

- MAC (*message authentication code*): kode kecil berukuran tetap (*fixed*) yang dihasilkan dari pesan dan kunci untuk mengotentikasi pengirim dan memeriksa integritas pesan.

$$MAC = C_k(M)$$

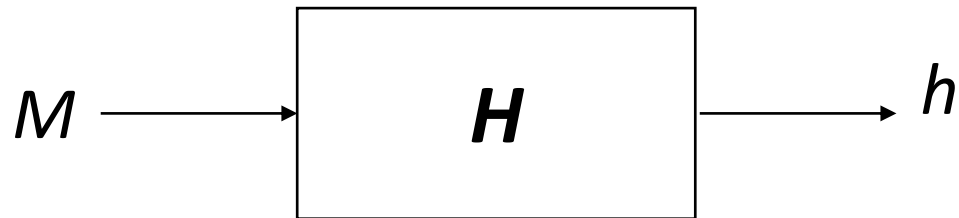
$MAC$  = kode otentikasi pesan

$C$  = algoritma  $MAC$

$K$  = kunci rahasia

- Nama lainnya: *Message Integrity Code* (MIC)

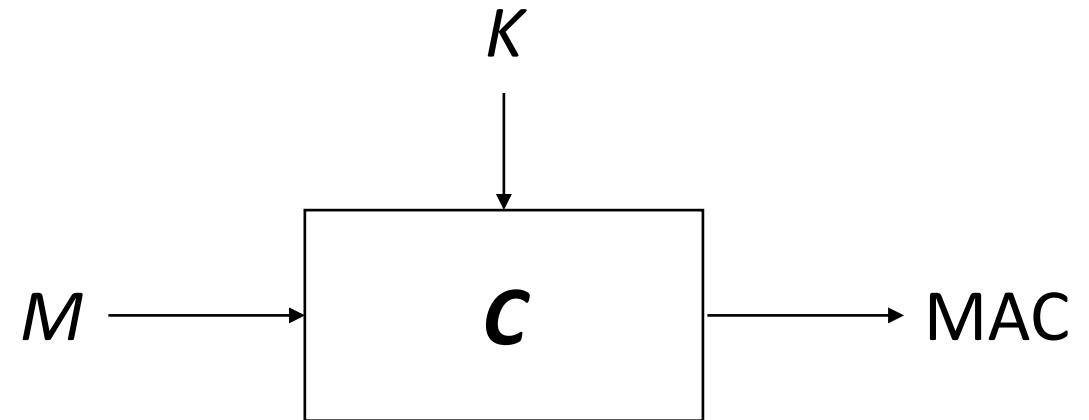
# Perbedaan Algoritma MAC dengan Fungsi Hash biasa



$$h = H(M)$$

*Message digest dengan fungsi hash*

(tidak membutuhkan kunci)



$$\text{MAC} = C_K(M)$$

*MAC dengan fungsi hash*

(membutuhkan kunci)

- *MAC* dilekatkan (*embed*) pada pesan sebagai “signature”
- *MAC* digunakan untuk memeriksa integritas (keaslian) pesan dan otentikasi pengirim.
- Jika *MAC* yang dikirim sama dengan *MAC* yang dihitung oleh penerima, maka pesan masih asli.

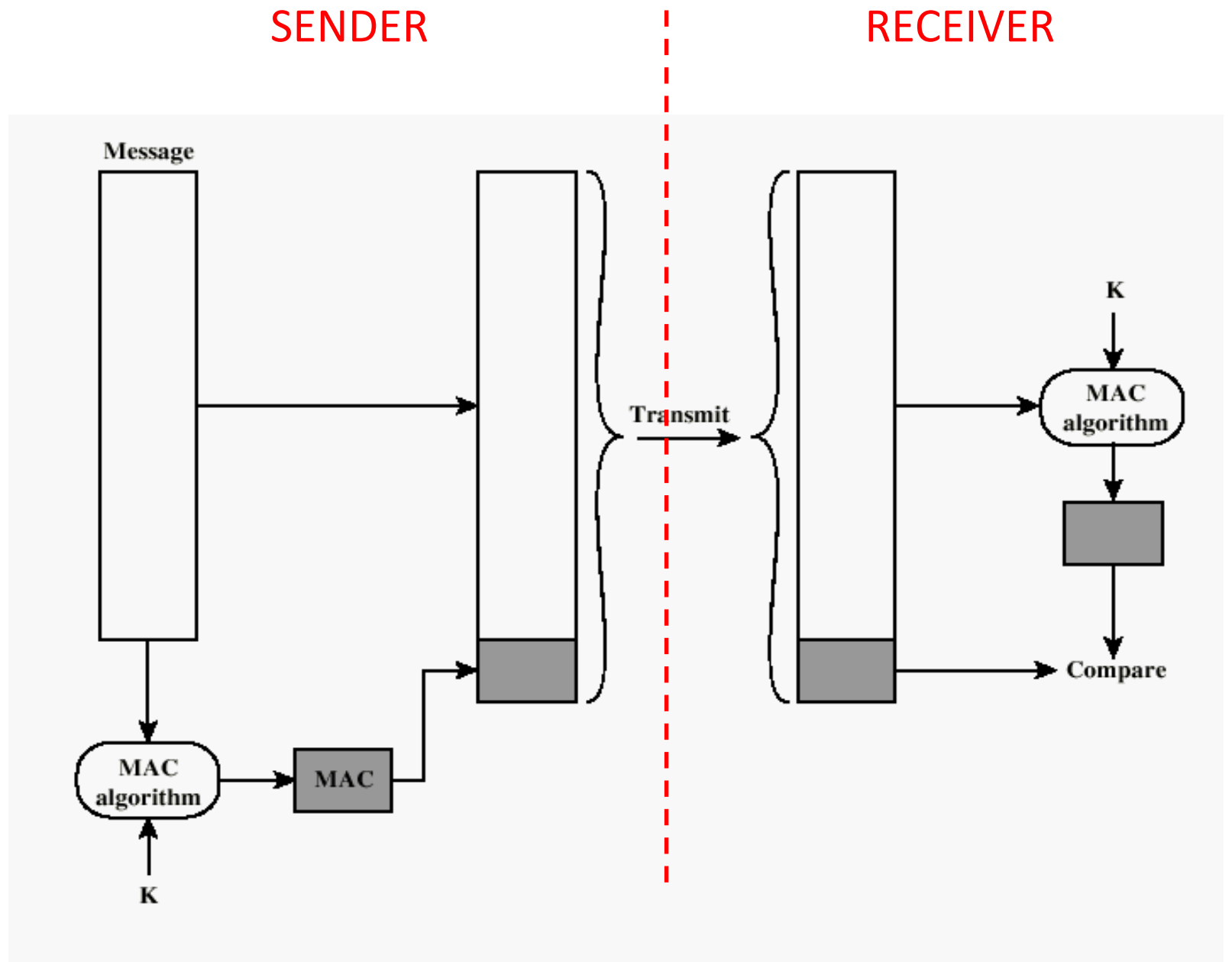
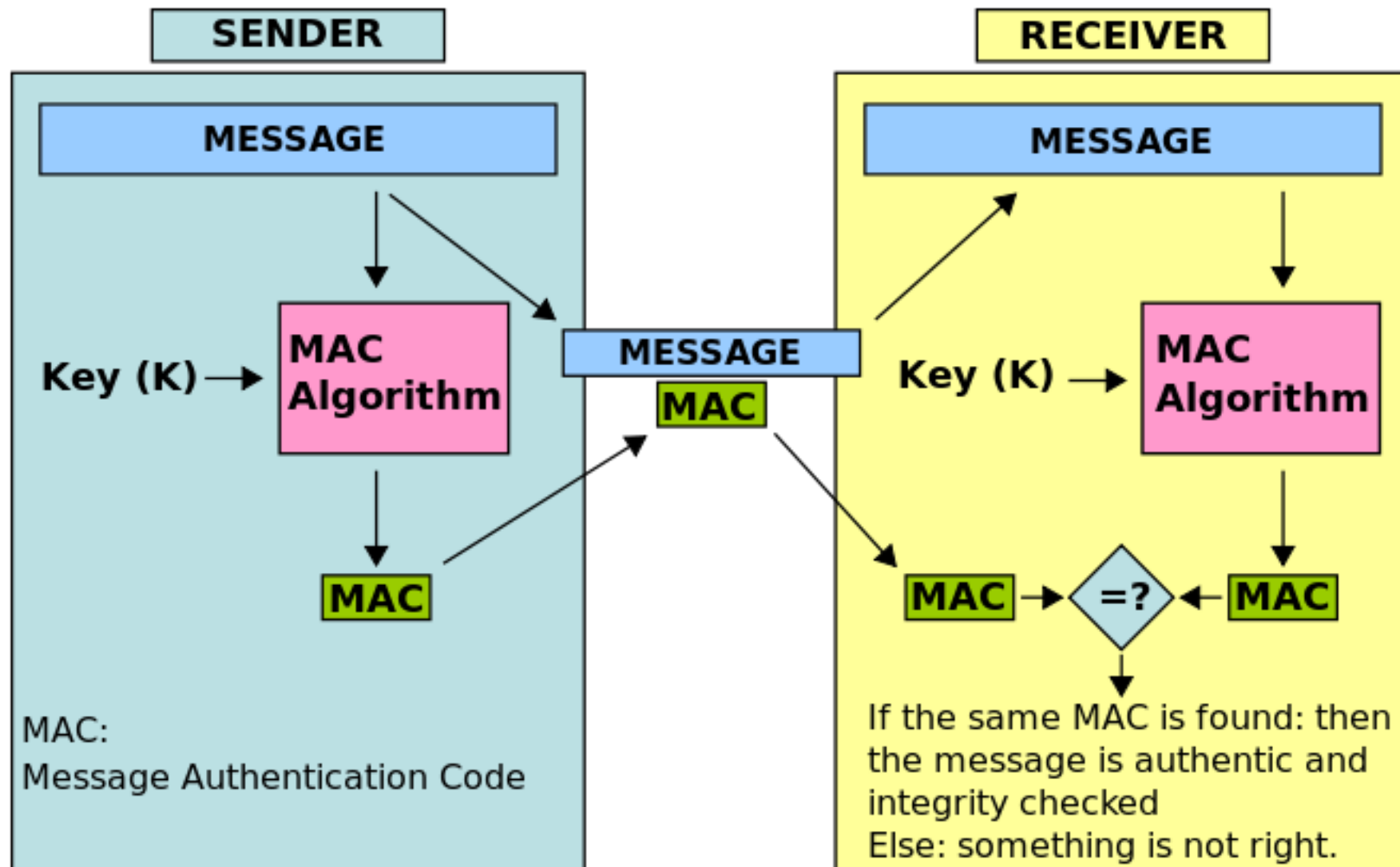


Figure 3.1 Message Authentication Using a Message Authentication Code (MAC)



# Kegunaan MAC

1. Seperti *message digest*, MAC digunakan untuk memeriksa keaslian pesan, dokumen, dsb. (*cryptographic checksum*)

→ Menjaga integritas (keaslian) pesan terhadap perubahan oleh pihak lawan, misalnya akibat serangan *hacker*, virus, dsb.

- Jika MAC yang dihitung oleh penerima pesan = MAC yang melekat pada pesan, berarti pesan masih asli.
- Jika pemilik pesan menggunakan fungsi *hash* satu-arah biasa (seperti MD5 atau SHA), maka pihak lawan dapat menghitung *message digest* yang baru dari dokumen yang sudah diubah, lalu menggantinya.

Tetapi, jika digunakan *MAC*, pihak lawan tidak dapat melakukan hal ini karena ia tidak mengetahui kunci yang asli untuk menghitung *MAC*.



Home

Windows

Mac

Linux

Freeware



e.g. Spyware Removal

Windows

Go

## Choose Download Location

### Norton AntiVirus 2010

You have chosen to download **Norton AntiVirus 2010**. Check the file details to make sure this is the correct program and version, and that your operating system is supported.

#### Download Details

OPERATING SYSTEMS 7 / XP / VISTA

FILE NAME NAV60TMD.exe

MD5 HASH CE0F5F1BF0F165465BE97BAEB4BD940C

FILE SIZE 85.03 MB

In order to make the download process as fast for you as possible, this file exists on several Tucows Downloads servers around the world. Please choose the location closest to you from which to download the file.

Hacker bisa mengganti file dengan file lain, mengganti nilai MD5 semula dengan nilai MD5 yang baru. Pengunduh file tidak dapat menyadarinya.



## 2. Mengotentikasi pengirim pesan

- Selain memeriksa keaslian pesan, MAC dapat digunakan untuk mengotentikasi pengirim pesan, bahwa pesan memang berasal dari pengirim yang sesungguhnya (asli).
- Hal ini karena hanya pengirim dan penerima pesan yang mengetahui kunci  $K$ .
- Jika  $K$  pengirim pesan tidak sama dengan  $K$  penerima pesan, maka MAC yang dihitung oleh penerima pesan pasti tidak sama dengan MAC yang melekat pada pesan yang diterimanya.

# Kapan menggunakan MAC?

- Bila hanya otentikasi pesan saja yang diperlukan, sedangkan kerahasiaan pesan tidak ditekankan.
- Namun, jika dikombinasikan dengan enkripsi pesan, maka enkripsi pesan dan perhitungan MAC menggunakan kunci yang terpisah.
- MAC dapat dihitung setelah pesan dienkripsi atau sebelum pesan dienkripsi. Umumnya lebih baik MAC dihitung sebelum pesan dienkripsi.
- Perlu dicatat bahwa MAC tidak sama dengan “digital signature” karena ia tidak menyediakan layanan nir-penyangkalan (*non-repudiation*).
- Digital signature akan dibahas pada pokok bahasan berikutnya

# Algoritma MAC

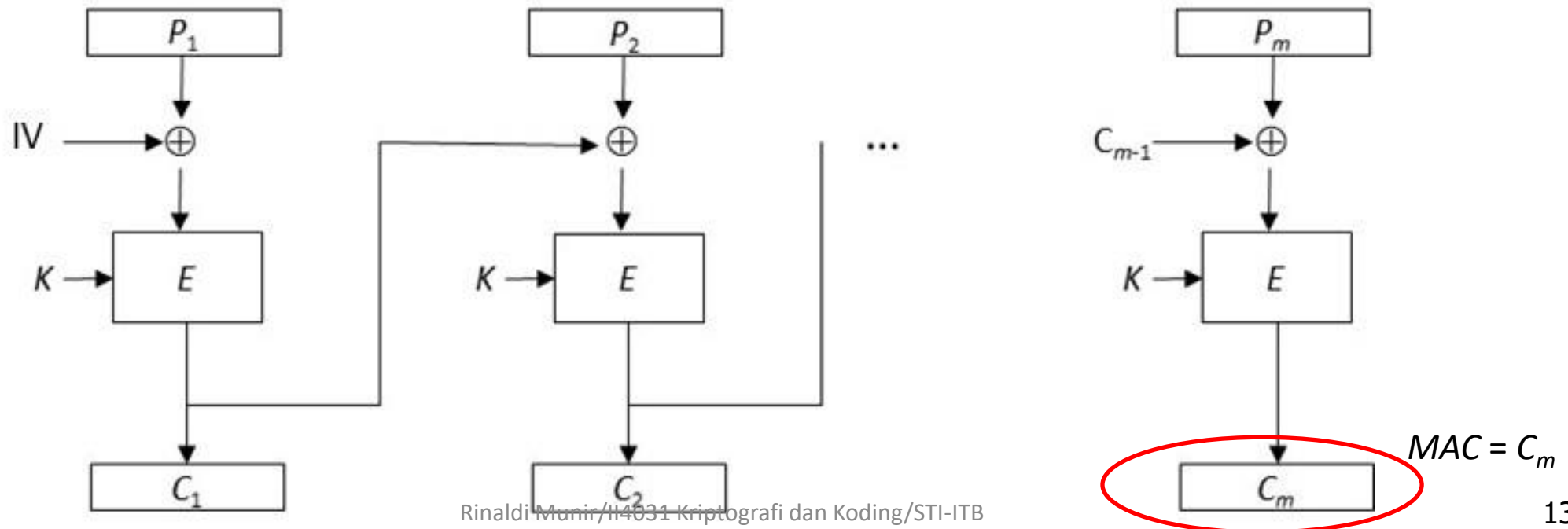
- Sebagaimana fungsi hash, MAC dihasilkan dengan oleh algoritma yang melakukan kompresi pesan yang bersifat *irreversible*.
- Algoritma MAC dalah fungsi banyak-ke-satu (many-to-one function), artinya potensial banyak pesan memiliki MAC yang sama
- Namun menemukan dua atau lebih pesan yang memiliki MAC yang sama adalah sangat sulit.
- Persyaratan algoritma MAC:
  1. Diberikan sebuah pesan dan MAC, maka sangat sukar menemukan pesan lain yang memiliki MAC yang sama.
  2. MAC dari berbagai pesan seharusnya terdistribusi secara *uniform*
  3. MAC seharusnya bergantung rata pada semua bit-bit pesan

- MAC dapat dihasilkan dengan menggunakan:
  1. Algoritma MAC berbasis *block cipher*
  2. Algoritma MAC berbasis fungsi *hash* satu arah yang sudah ada

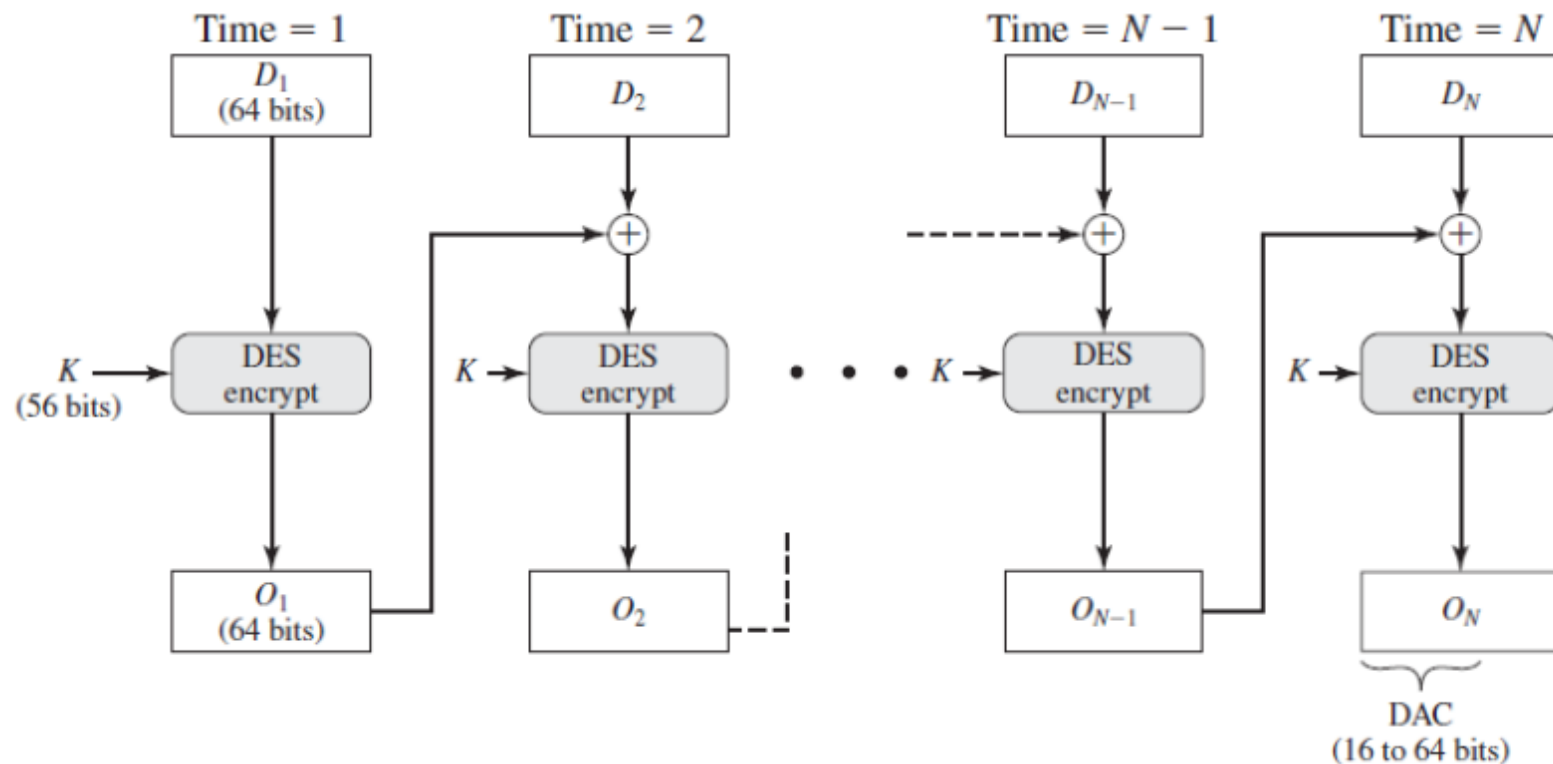
# Algoritma MAC

## (a) Algoritma MAC berbasis *block cipher*

- *MAC* dibangkitkan dari *block cipher* dengan mode *CBC* atau *CFB*.
- Nilai *hash*-nya (yang menjadi *MAC*) adalah hasil enkripsi blok terakhir.



- Misalkan *DES* digunakan sebagai *cipher* blok, maka MAC = ukuran blok = 64 bit, dan kunci rahasia *MAC* adalah kunci DES yang panjangnya 56 bit.
- *Data Authentication Algorithm (DAA)* adalah algoritma MAC berbasis *DES-CBC* yang digunakan secara luas:



## (b) Algoritma MAC berbasis fungsi *hash* satu-arah (HMAC)

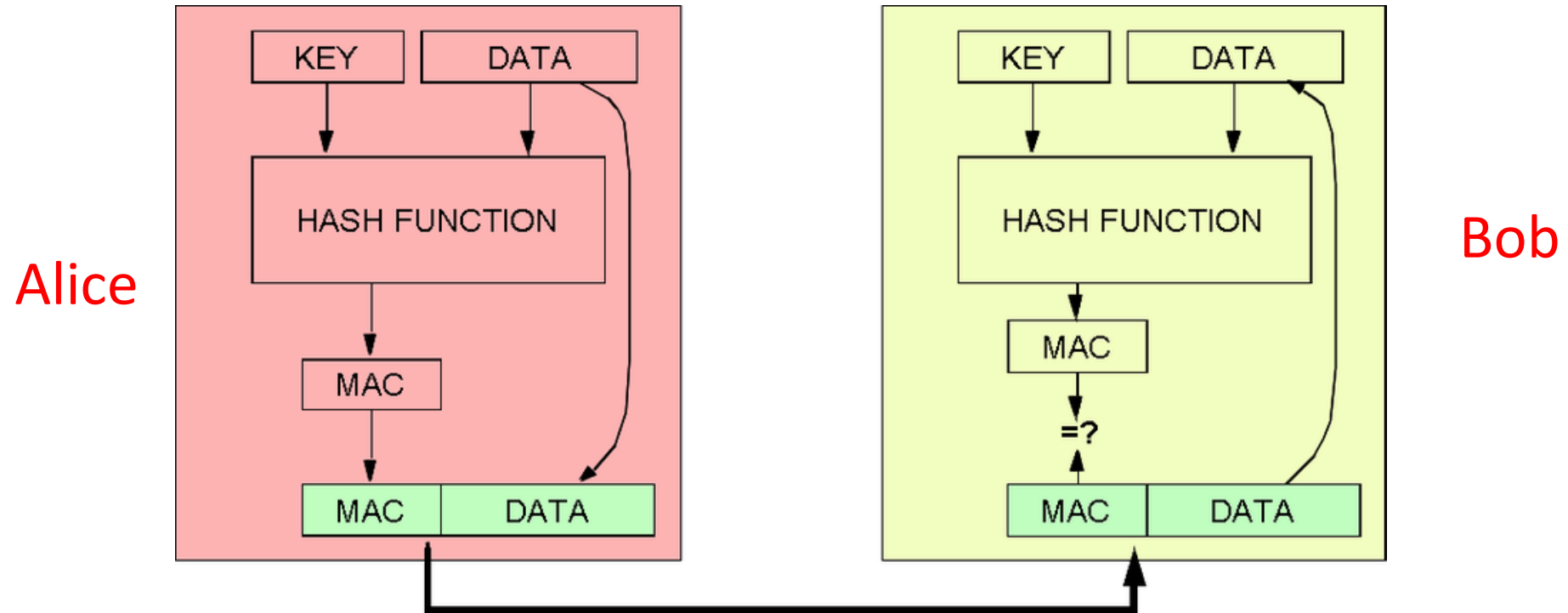
- Fungsi *hash* seperti *MD5* dan *SHA* dapat digunakan sebagai *MAC*
- Pesan *M* disambung (*concat*) dengan kunci *K*, lalu dihitung nilai hash dari hasil penggabungan tersebut dengan dengan fungsi hash *H* seperti *MD5* atau *SHA*

$$\text{MAC} = H(M|K)$$

ket: '|' adalah simbol *concatenation*

- Panjang *MAC* tergantung dari fungsi hash yang digunakan. Jika menggunakan fungsi *SHA-1*, maka *MAC* yang dihasilkan adalah 160 bit

Misalkan Alice dan Bob akan saling bertukar DATA. Alice dan Bob telah berbagi sebuah kunci rahasia *KEY*.





## HMAC Generator / Tester Tool

[Encoders - Cryptography](#) / [HMAC Generator](#)

Computes a Hash-based message authentication code (HMAC) using a secret key. A HMAC is a small set of data that helps authenticate the nature of message; it protects the integrity and the authenticity of the message.

The secret key is a unique piece of information that is used to compute the HMAC and is known both by the sender and the receiver of the message. This key will vary in length depending on the algorithm that you use.

I use [Bouncy Castle](#) for the implementation.

Copy-paste the string here

Kita bikin romantis  
Bikin paling romantis

Secret key

MALIQ

Digest algorithm

SHA256

Compute HMAC

erator ...  
r  
Tester  
tor

## -Computed HMAC-

4f1e8af74ed79da8220d7e60febaa375483d38021da884774c10e994a356515c

Copy

Save

Contoh:

$M = \textit{Halo, Bob!}$

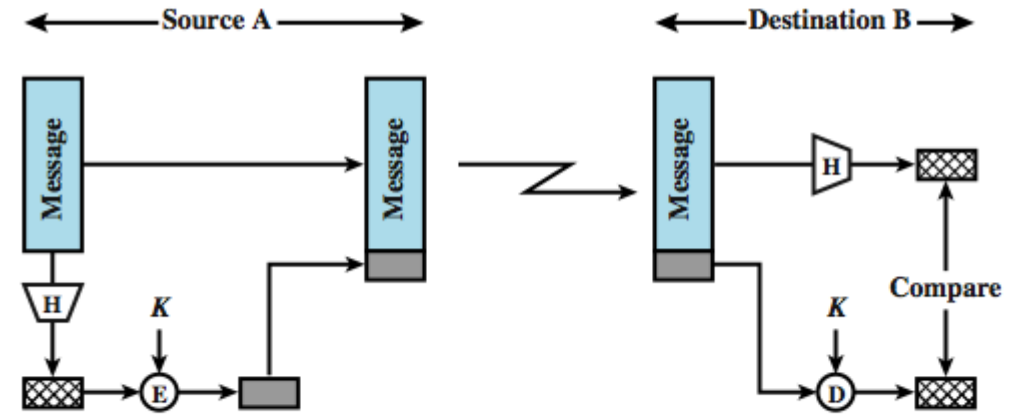
$K = 12345678$

Fungsi Hash: SHA-1

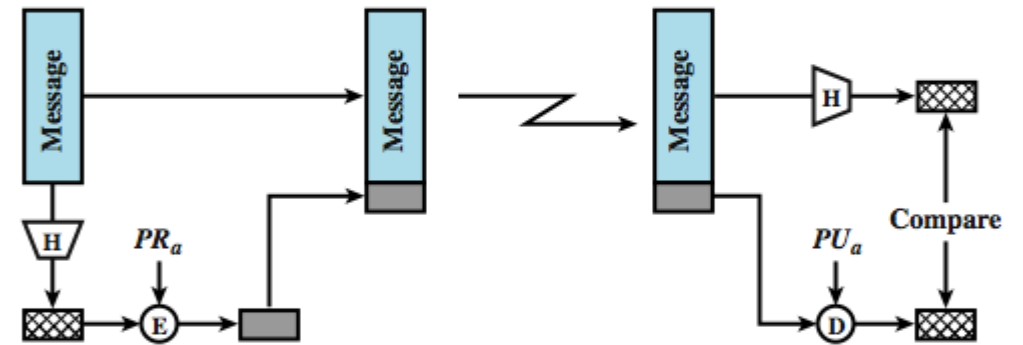
$MAC = 6f8605c7c3a649a40abfb87b44aa21f356e931a0$

Sumber: MAC online <https://www.freeformatter.com/hmac-generator.html>

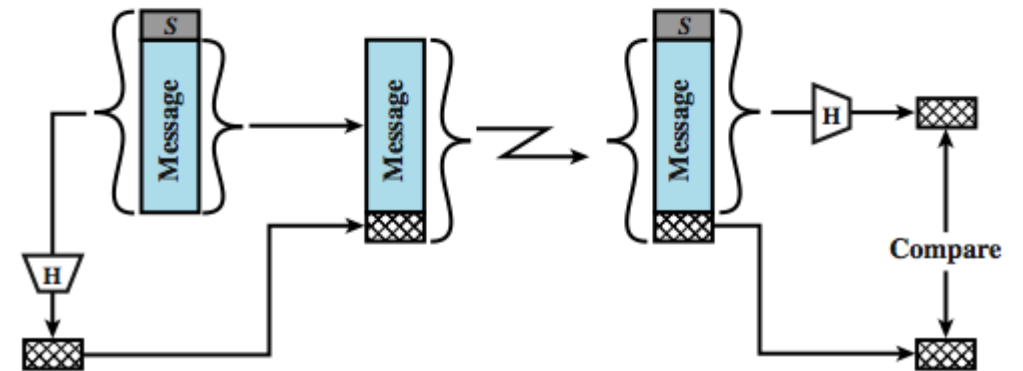
- Selain menggunakan MAC, terdapat teknik lain untuk mengotentikasi pesan menggunakan kunci
- (a) Menggunakan fungsi hash dan enkripsi simetri
- (b) Menggunakan enkripsi dengan kriptografi kunci publik
- (c) Menggunakan nilai rahasia  $S$  yang disepakati oleh pengirim dan penerima pesan



(a) Using conventional encryption



(b) Using public-key encryption



(c) Using secret value

**SELAMAT BELAJAR**