Outline

- Message Authentication Codes
- MAC requirements and security
- MACs based on Hash Functions
- MACs based on Block Ciphers

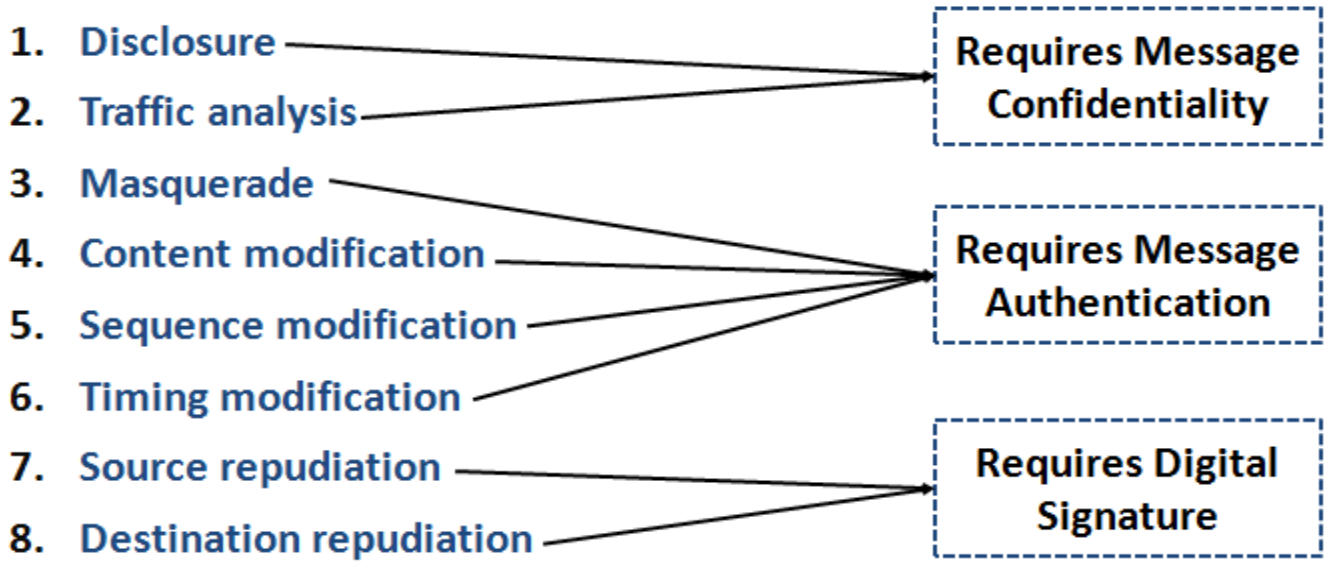
Message Authentication

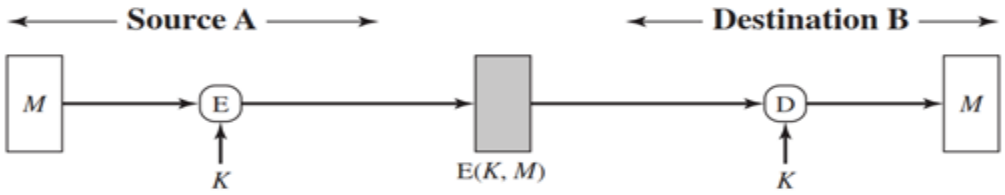
- **Message authentication** is a procedure to verify that received messages come from the genuine source and have not been altered.
- Message authentication may also verify sequencing and timeliness.
- Message authentication is a mechanism or service used to verify the **integrity of a message**.
- Message authentication assures that data received are exactly as sent (i.e., contain no modification, insertion, deletion, or replay).

Message Authentication Requirements

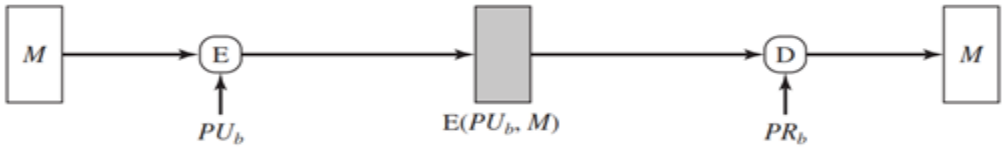
1. **Disclosure:** Release of message contents
2. **Traffic analysis:** Discovery of the pattern of traffic between parties
3. **Masquerade:** Insertion of messages into the network from a fraudulent source
4. **Content modification:** Changes to the contents of a message
5. **Sequence modification:** Any modification to a sequence of messages between parties
6. **Timing modification:** Delay or replay of messages
7. **Source repudiation:** Denial of transmission of message by source
8. **Destination repudiation:** Denial of receipt of message by destination

Message Authentication Requirements

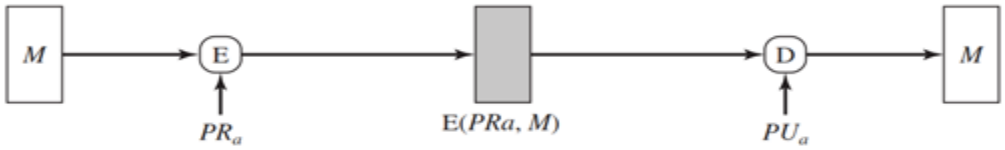




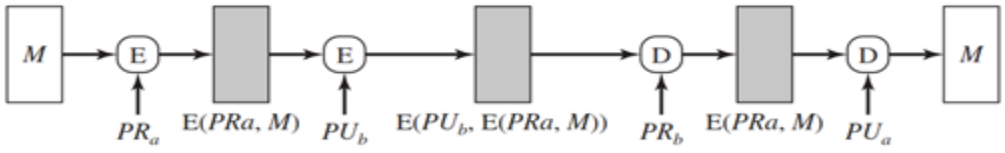
(a) Symmetric encryption: confidentiality and authentication



(b) Public-key encryption: confidentiality



(c) Public-key encryption: authentication and signature



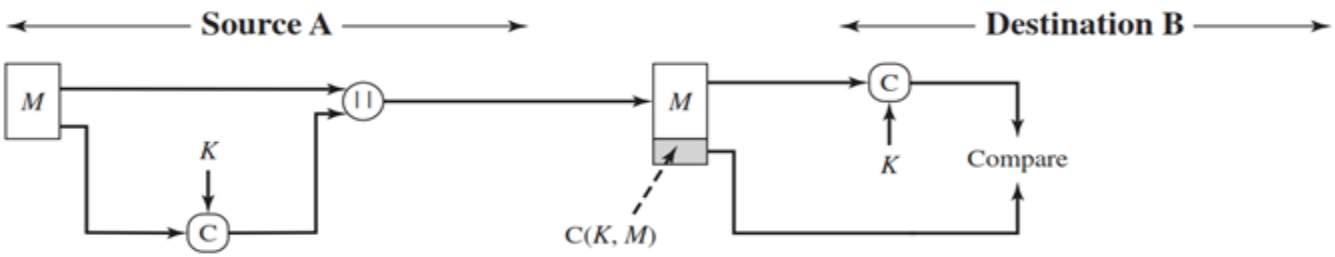
(d) Public-key encryption: confidentiality, authentication, and signature

Message Authentication Code

- An alternative authentication technique involves the use of a secret key to generate a small fixed-size block of data, known as a **cryptographic checksum** or **MAC**
- MAC is appended to the message. This technique assumes that two communicating parties, say A and B, share a common secret key K.
- When A has a message to send to B, it calculates the MAC as a function of the message and the key

$$MAC = C(K, M)$$

Message Authentication Code



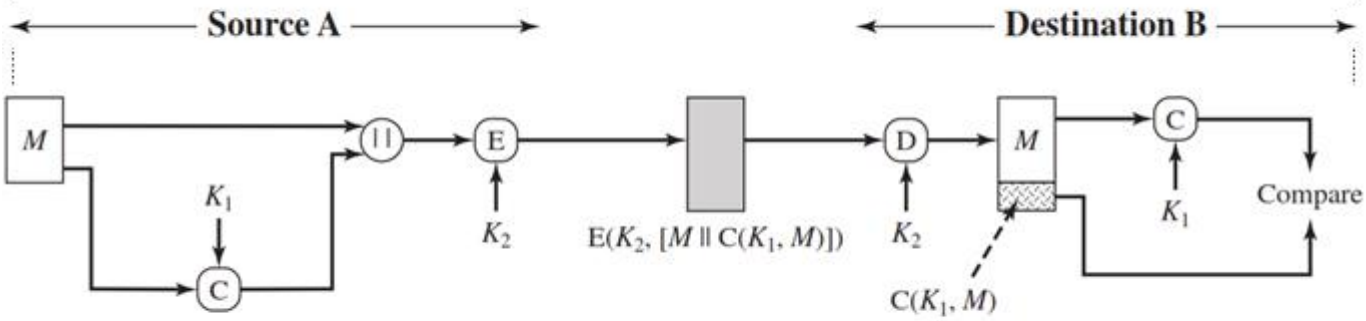
(a) Message authentication

- The receiver is assured that the message has not been altered. If an attacker alters the message but does not alter the MAC, then the receiver's calculation of the MAC will differ from the received MAC.
- Because the attacker is assumed not to Know the secret key, the attacker cannot alter the MAC to correspond to the alterations in the message.

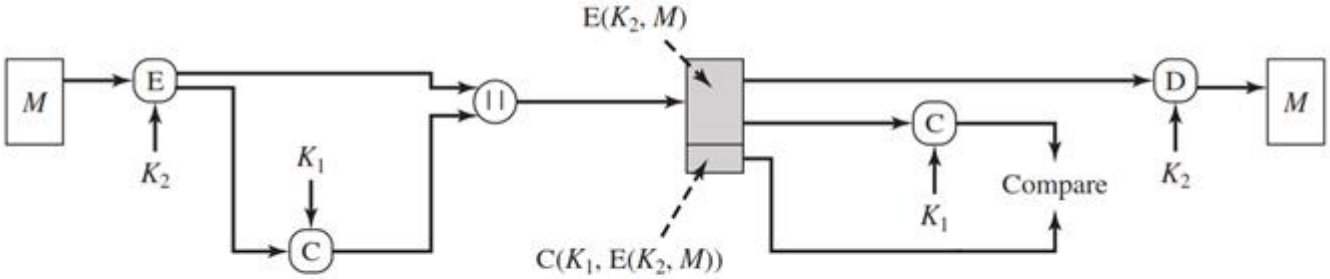
Message Authentication code - Cont...

- The receiver is assured that the message is from the alleged sender.
- Because no one else knows the secret key, no one else could prepare a message with a proper MAC.
- A MAC function is similar to encryption. One difference is that the MAC algorithm need not be reversible, as it must be for decryption.
- In general, the MAC function is a many-to-one function. The domain of the function consists of messages of some arbitrary length, whereas the range consists of all possible MACs and all possible keys.
- If an n-bit MAC is used, then there are 2^n possible MACs

Message Authentication code - Cont...



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

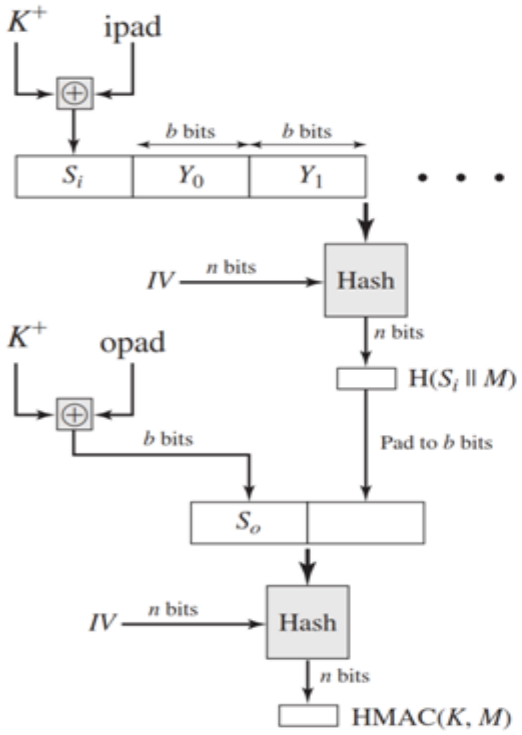


(c) Message authentication and confidentiality; authentication tied to ciphertext

MAC Based on Hash Functions - HMAC

- Cryptographic hash functions such as MD5 and SHA generally execute faster in software than symmetric block ciphers such as DES.
- Library code for cryptographic hash functions is widely available.

HMAC Structure



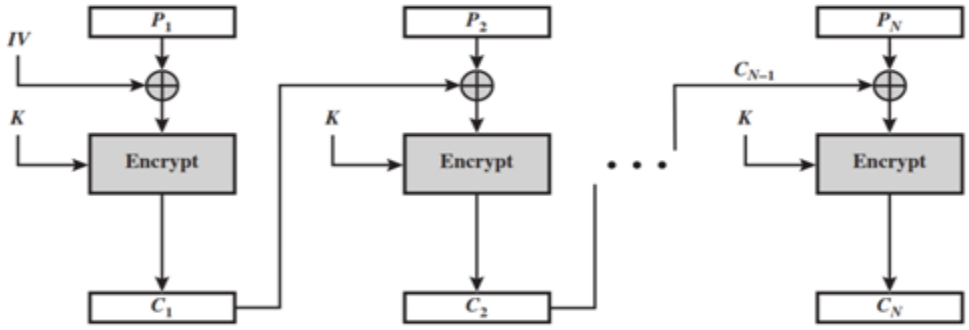
1. Append zeros to the left end of K to create a b-bit string K^+
2. XOR K^+ with ipad to produce the b-bit block Si.
3. Append M to Si.
4. Apply H to the stream generated in step 3.
5. XOR K^+ with opad to produce the b-bit block S_0.
6. Append the hash result from step 4 to S_0.
7. Apply H to the stream generated in step 6 and output the result.

HMAC Structure

- H = embedded hash function (e.g., MD5, SHA-1, RIPEMD-160)
- IV = initial value input to hash function
- M = message input to HMAC
- Yi = i th block of M
- L = number of blocks in M
- n = length of hash code produced by embedded hash function
- 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*

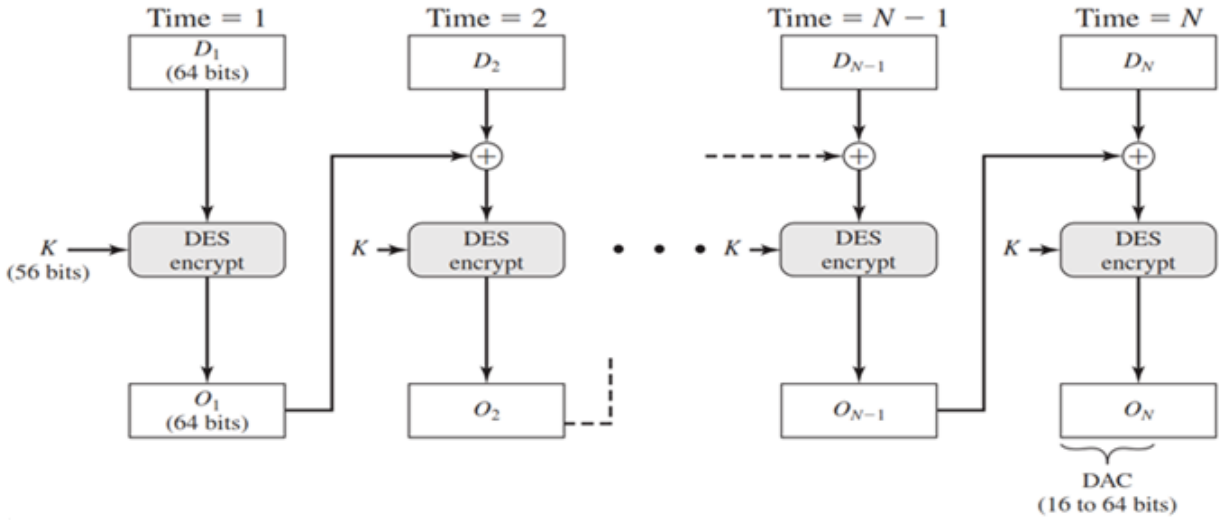
MAC based on Block Ciphers

- The **Data Authentication Algorithm** (DAA), based on DES, has been one of the most widely used MACs for a number of years.
- The algorithm can be defined as using the cipher block chaining (CBC) mode of operation of DES (Figure 6.4) with an initialization vector of zero.



(a) Encryption

Data Authentication Algorithm (DAA)



Data Authentication Algorithm (DAA)

- The data (e.g., message, record, file, or program) to be authenticated are grouped into contiguous 64-bit blocks:
- D1, D2, …. Dn. If necessary, the final block is padded on the right with zeroes to form a full 64-bit block. Using the DES encryption algorithm E and a secret key K, **a data authentication code (DAC)** is calculated as follows

$$
\begin{aligned}
O_1 &= \mathrm{E}(K, D) \\
O_2 &= \mathrm{E}(K, [D_2 \oplus O_1]) \\
O_3 &= \mathrm{E}(K, [D_3 \oplus O_2]) \\
&\cdot \\
&\cdot \\
&\cdot \\
O_N &= \mathrm{E}(K, [D_N \oplus O_{N-1}])
\end{aligned}
$$

Cipher-Based Message Authentication Code (CMAC)

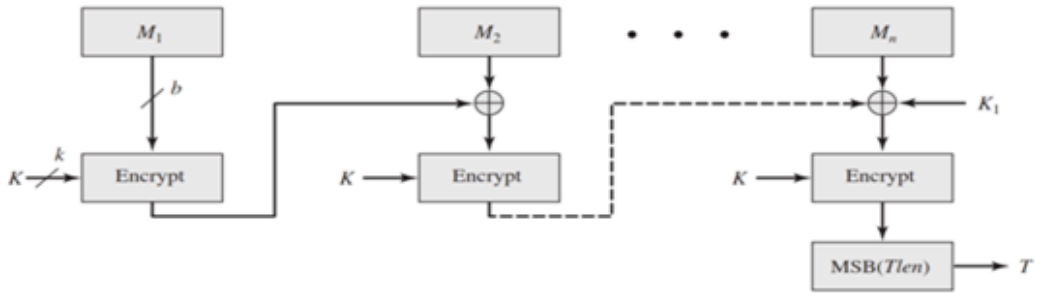
- **Cipher-based Message Authentication Code (CMAC)** mode of operation for use with AES and triple DES.
- First, let us define the operation of CMAC when the message is an integer multiple n of the cipher block length b. For AES, b = 128, and for triple DES, b = 64. The message is divided into n blocks (M1, M2,… Mn)

Cipher-Based Message Authentication Code (CMAC)

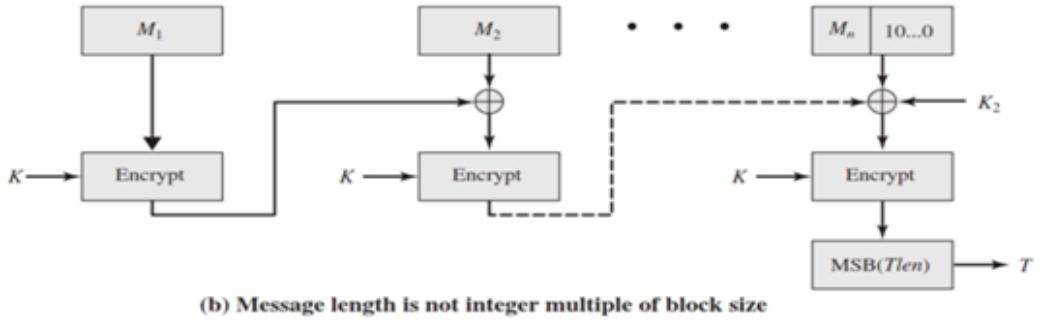
- The algorithm makes use of a k-bit encryption key K and a b-bit constant, K1.
- For AES, the key size k is 128, 192, or 256 bits; for triple DES, the key size is 112 or 168 bits.
- CMAC is calculated as follows

$$
\begin{aligned}
C_1 &= \mathrm{E}(K, M_1) \\
C_2 &= \mathrm{E}(K, [M_2 \oplus C_1]) \\
C_3 &= \mathrm{E}(K, [M_3 \oplus C_2]) \\
&\cdot \\
&\cdot \\
&\cdot \\
C_n &= \mathrm{E}(K, [M_n \oplus C_{n-1} \oplus K_1]) \\
T &= \mathrm{MSB}_{Tlen}(C_n)
\end{aligned}
$$

Cipher-Based Message Authentication Code (CMAC)



(a) Message length is integer multiple of block size



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