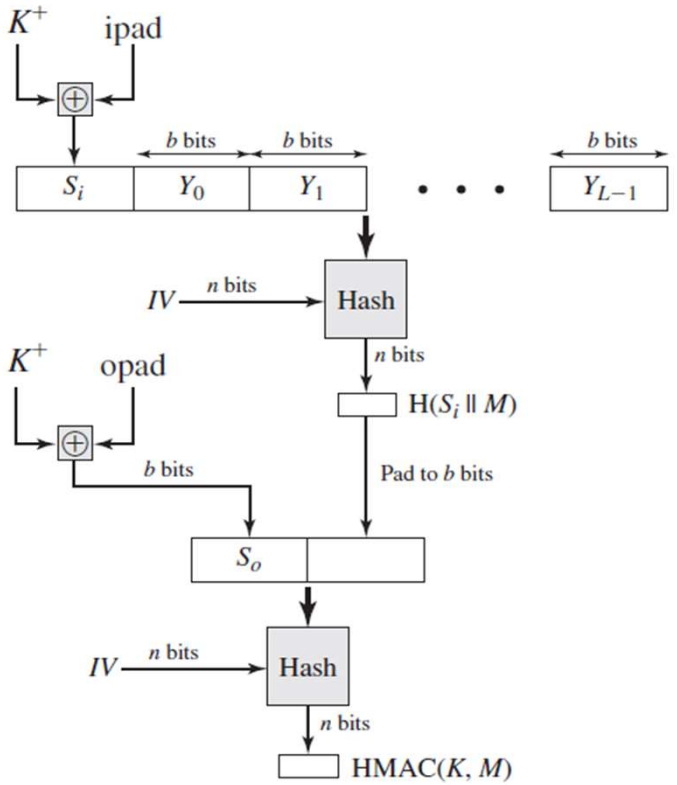
HMAC Algorithm:

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, $0 \le i \le (L - 1)$
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 $\ge n$; 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



We can describe the algorithm as follows.

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

ipad = 00110110 (36 in hexadecimal) repeated $b/8$ times
opad = 01011100 (5C in hexadecimal) repeated $b/8$ times

Then HMAC can be expressed as

$$\text{HMAC}(K, M) = \text{H}[(K^+ \oplus \text{opad}) \| \text{H}[(K^+ \oplus \text{ipad}) \| M]]$$

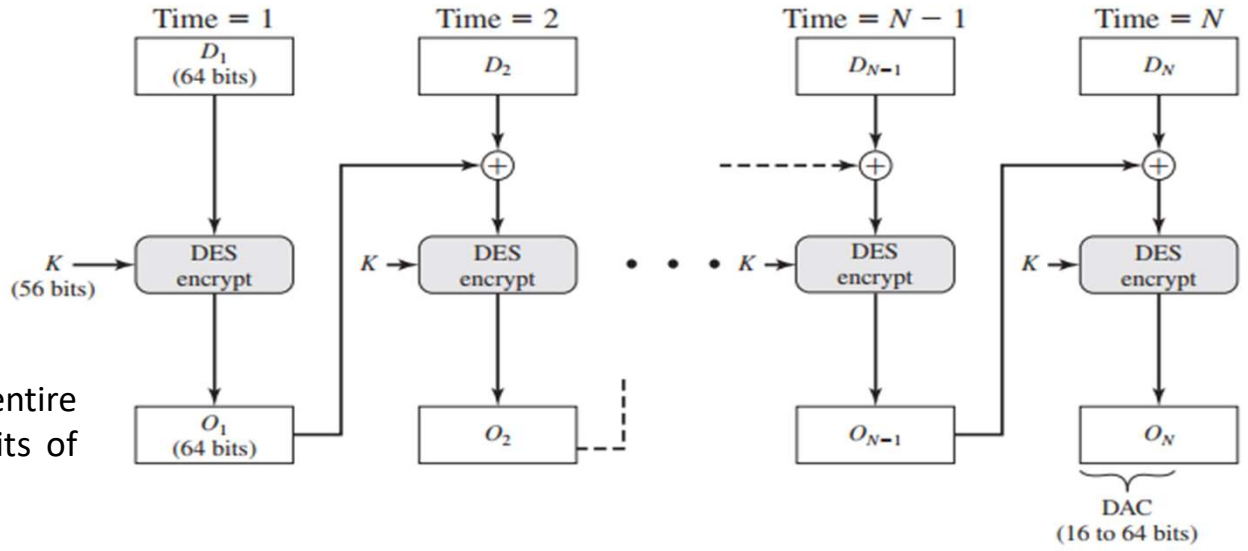
MACs based on block ciphers: DAA and CMAC:

There are two MACs that are based on the use of a block cipher mode of operation: The **Data Authentication Algorithm** (**DAA**), which is now obsolete, and the **Cipher-Based Message Authentication Code** (**CMAC**), which is designed to overcome the deficiencies of the DAA.

Data Authentication Algorithm

The **Data Authentication Algorithm** (**DAA**), based on **DES**, has been one of the most widely used MACs for a number of years. The algorithm is both a **FIPS** publication (**FIPS PUB 113**) and an **ANSI** standard (**X9.17**). The algorithm can be defined as using the cipher block chaining (**CBC**) mode of operation of **DES** with an initialization vector of zero. The data (e.g., message, record, file, or program) to be authenticated are grouped into contiguous 64-bit blocks: $D_1, D_2, \ldots, D_N$. 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

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



The **DAC** consists of either the entire block O_N or the leftmost M bits of the block, with $16 \leq M \leq 64$.

Cipher-Based Message Authentication Code (CMAC):

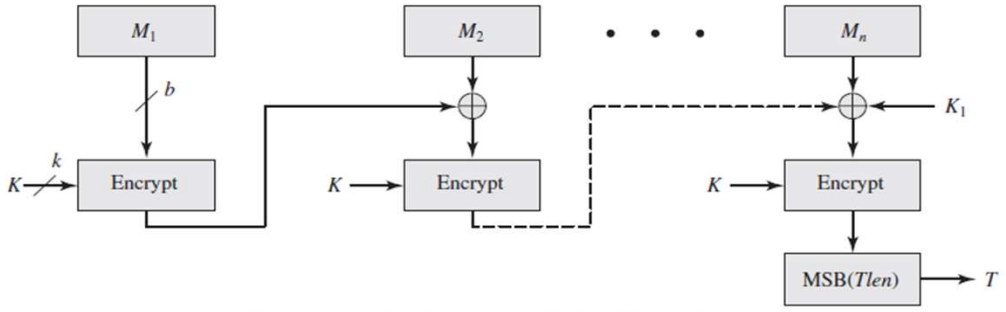
- It is a MAC that is based on the use of a block cipher mode of operations for use with AES and triple DES. It is also adopted by NIST.
- The **CMAC** overcomes the limitations of the Data Authentication Algorithm (DAA) which is based on DES.
- The **operation of the CMAC** can be defined as follows:

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 (M_1, M_2,…, M_n). The algorithm makes use of a **k**-bit encryption key K and a b-bit constant, $\mathbf{K_1}$. For AES, the key size **k** is 128, 192, or 256 bits; for triple DES, the key size is 112 or 168 bits.

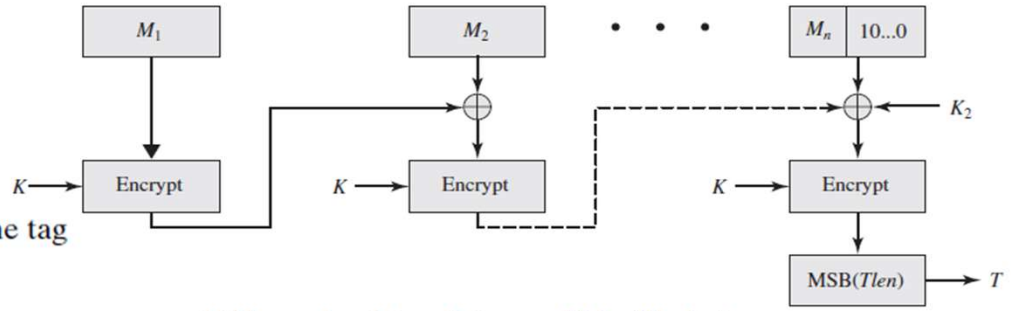
- **CMAC** is calculated as follows:

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

T = message authentication code, also referred to as the tag
$Tlen$ = bit length of T
$\mathrm{MSB}_s(X)$ = the s leftmost bits of the bit string X



(a) Message length is integer multiple of block size



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

Authenticated Encryption (AE):

Authenticated encryption (**AE**) is a term used to describe encryption systems that simultaneously protect confidentiality and authenticity (integrity) of communications. Many applications and protocols require both forms of security.

❑ There are four common approaches to providing both confidentiality and encryption for a message M.

• **Hashing followed by encryption (H → E):** First compute the cryptographic hash function over M as h = H(M). Then encrypt the message plus hash function: E(K, (M || h)).
• **Authentication followed by encryption (A → E):** Use two keys. First authenticate the plaintext by computing the MAC value as T = MAC(K1, M). Then encrypt the message plus tag: E(K2, [M||T]). This approach is taken by the SSL/TLS protocol.
• **Encryption followed by authentication (E → A):** Use two keys. First encrypt the message to yield the ciphertext C = E(K2, M). Then authenticate the ciphertext with T = MAC(K1, C) to yield the pair (C, T). This approach is used in the IPSec protocol.
• **Independently encrypt and authenticate (E → A).** Use two keys. Encrypt the message to yield the ciphertext C = E(K2, M). Authenticate the plaintext with T = MAC(K1, M) to yield the pair (C, T). These operations can be performed in either order. This approach is used by the SSH protocol.