CS557: Cryptography

Message Authentication code (MAC)

S. Tripathy
IIT Patna

Limitation of Using Hash Functions for Authentication

- Require an authentic channel to transmit the hash of a message
  - anyone can compute the hash value of a message, as the hash function is public
- How to address this?
  - use a key to select which one to use
    - Keyed Hash or MAC

Requirements for MACs

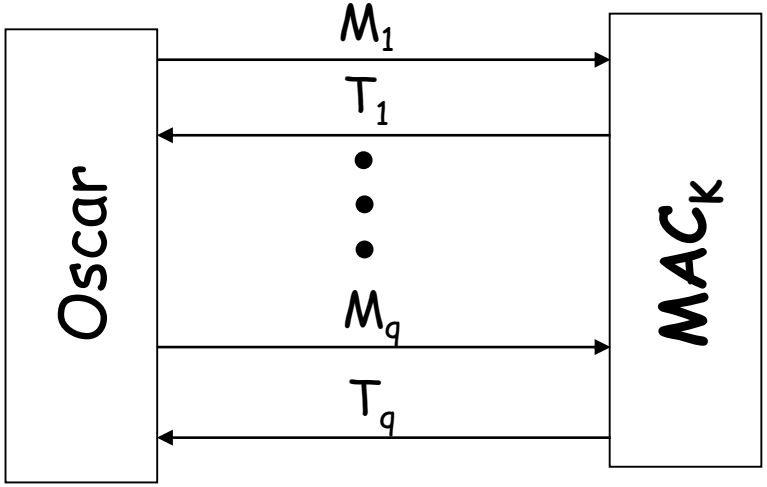
- taking into account the types of attacks MAC needs to satisfy the following:

1. knowing a message and MAC, it is infeasible to find another message with same MAC
2. MAC should depend equally on all bits of the message

3. MACs should be uniformly distributed

Distinguishing Attack

Stronger security notion than forging and Popular in the security analysis.

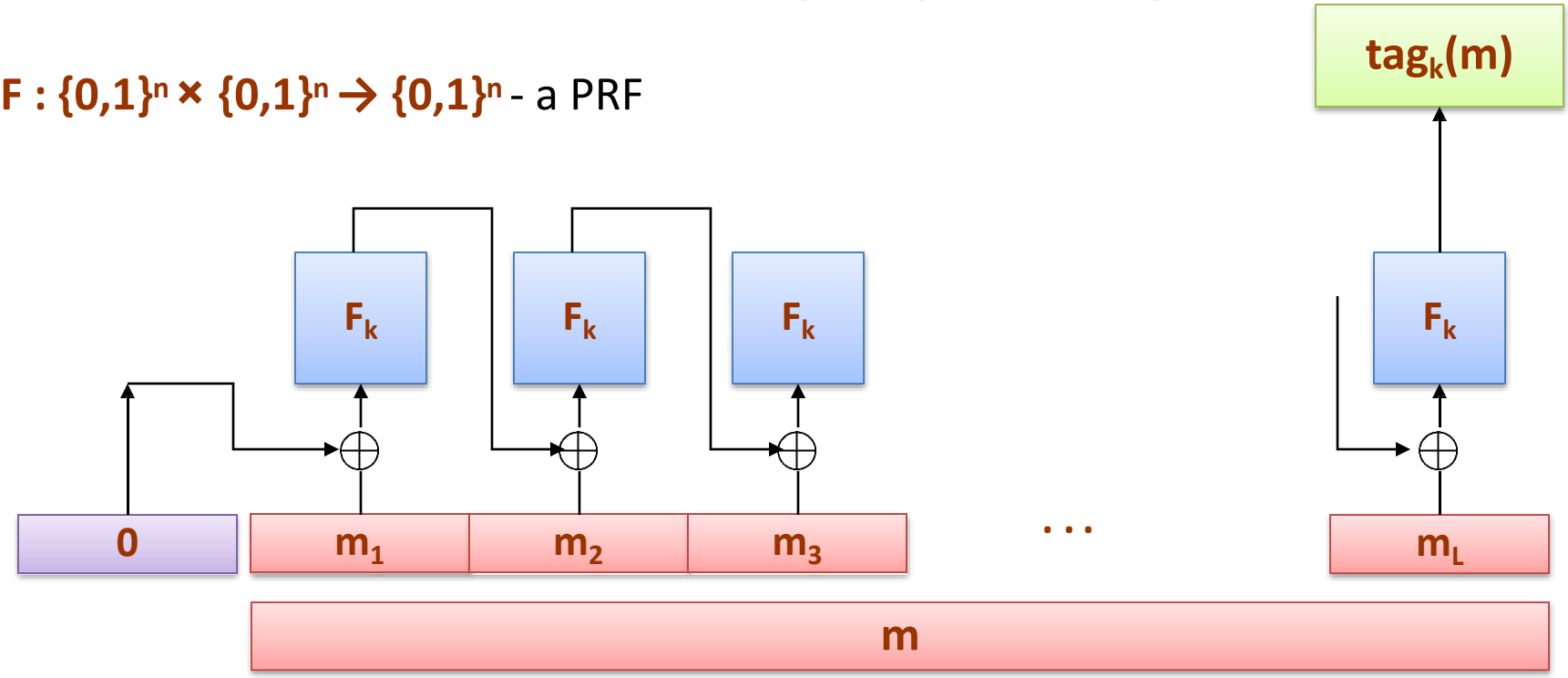


Finally, Oscar has to distinguish $\mathbf{T} = (T_1, \ldots, T_q)$ from a q-tuple of random strings.

- Forgery: adversary creates a valid message and *MAC* pair that was not created by the legitimate signer.
  - Existential forgery: The adversary creates a valid message and signature pair where the message can be anything, including gibberish.
  - Selective Forgery: The adversary creates a valid message and signature pair where the message was chosen by the challenger before the attack.

CBC-MAC

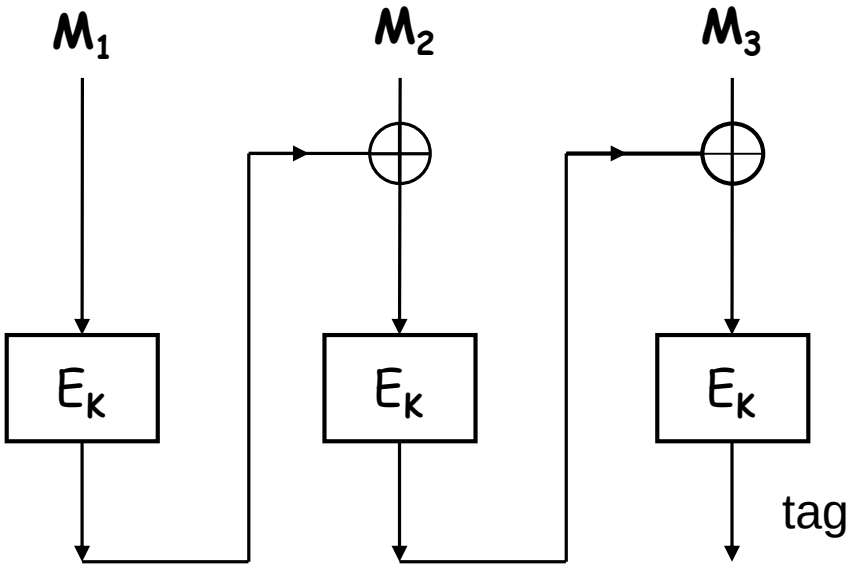
$F : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$ - a PRF



- CBC-MAC uses first block fixed at 0
- CBC-MAC with random IV is insecure!

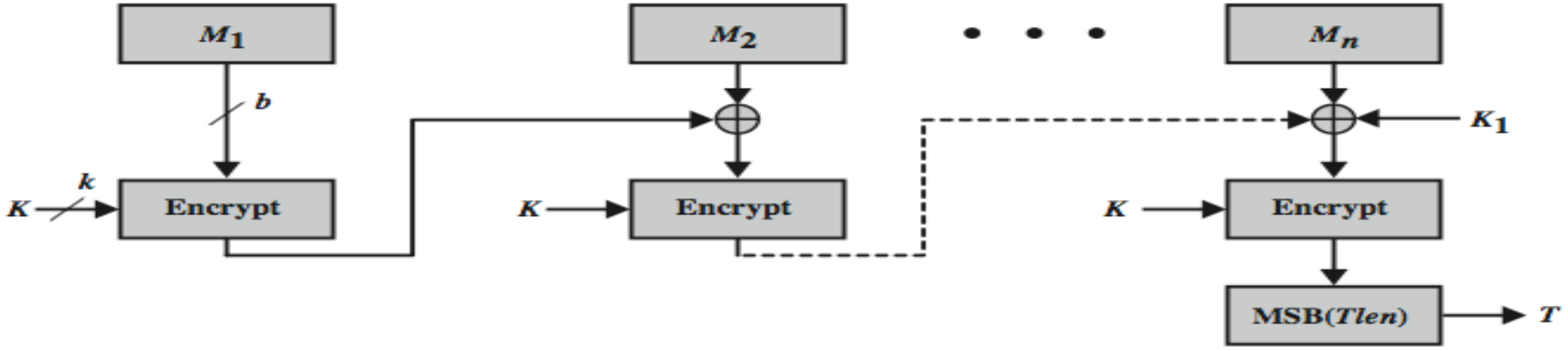
CBC-MAC: Block Cipher based MAC

- CBC MAC secure for prefix-free message space only.
- Secure for fixed length
- Length extension attack is valid for arbitrary domain

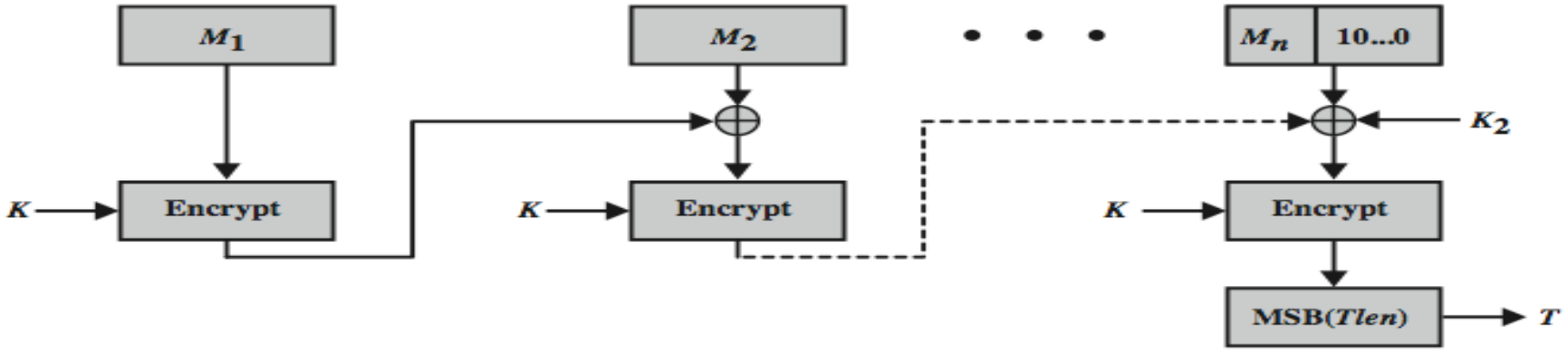


MAC using Block cipher (CMAC: Cipher based message authentication code)

- Avoids that flaw in last slide



(a) Message length is integer multiple of block size



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

Keyed Hash Functions as MACs

- have desire to create a MAC using a hash function rather than a block cipher
  - because hash functions are generally faster
  - not limited by export controls unlike block ciphers
  - hash includes a key along with the message
- Preliminary proposal:

  KeyedHash = Hash(Key||Message)

  - Has some weaknesses ?
  - H_k (x||x') = compress(H_k(x) || x')

HMAC

- specified as Internet standard RFC2104
- uses hash function on the message:

$$HMAC_K = Hash[(K^+ \text{ XOR opad}) \,||\, Hash[(K^+ \text{ XOR ipad})||M)]]$$

- where K^+ is the key padded out to size
- and opad, ipad are specified padding constants
- overhead is just 3 more hash calculations than the message needs alone
- any of MD5, SHA-1, RIPEMD-160 can be used

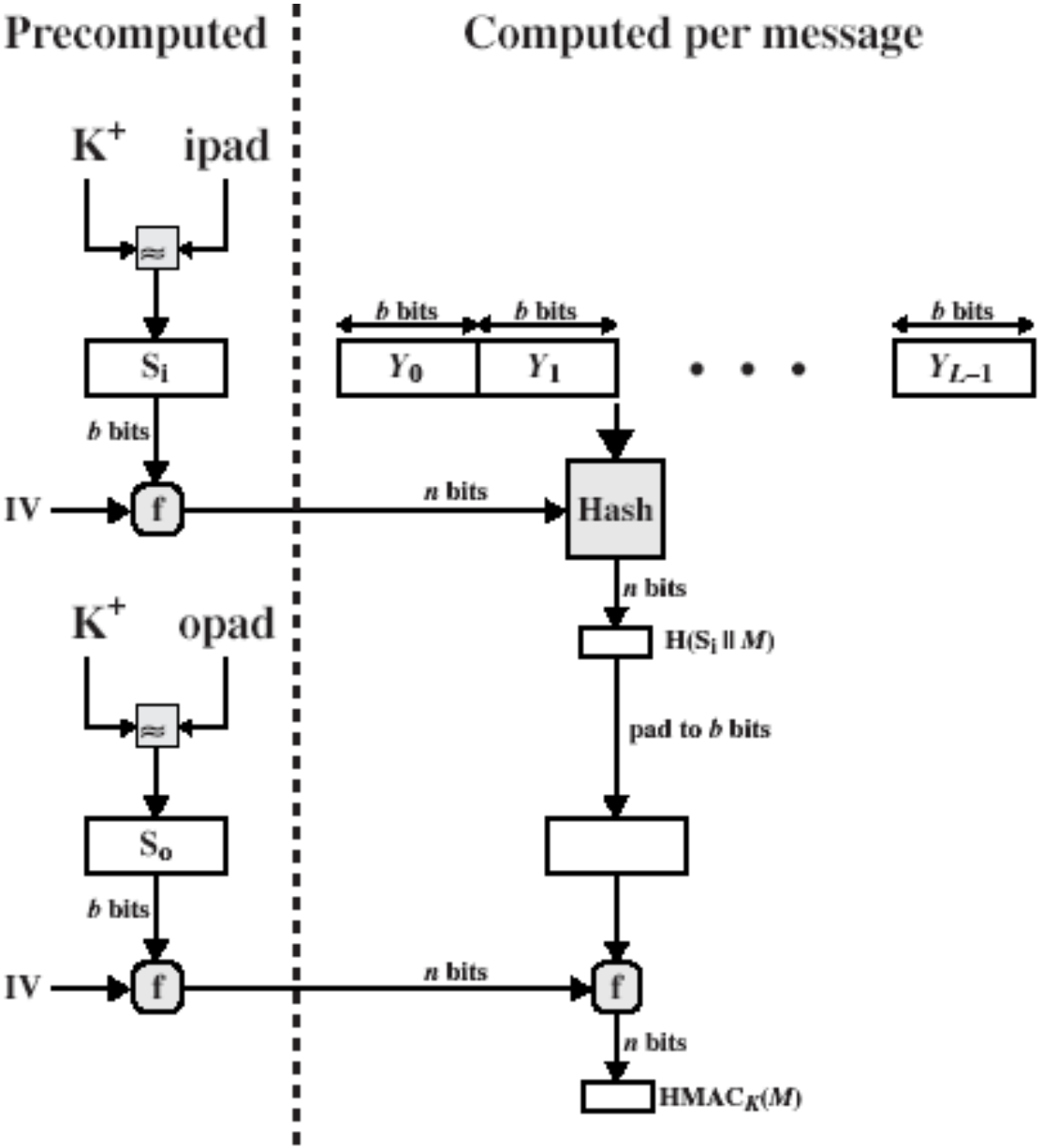
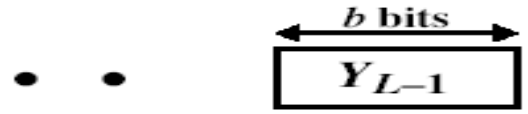


Figure 12.11 Efficient Implementation of HMAC

ew

ɬash[(K+ XOR opad)
(K+ XOR ipad) || M)]]



I M)

bits

D)

Ipad: 00110110
repeated b/8 times
Opad: 01011010
repeated b/8 times

HMAC Security

- the security of HMAC relates to that of the underlying hash algorithm
- attacking HMAC requires either:
  - brute force attack on key used
  - birthday attack (but since keyed would need to observe a very large number of messages)
- choose hash function used based on speed verses security constraints

Authenticated encryption

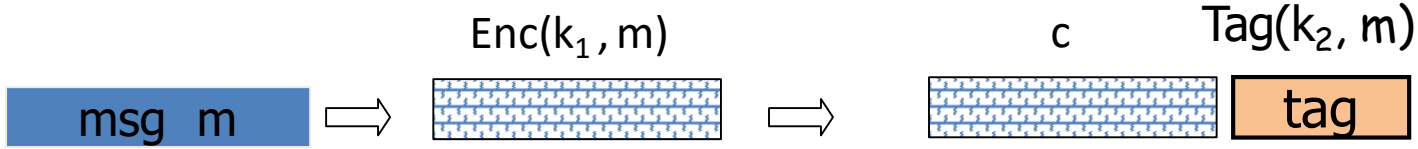
- Simultaneously protect confidentiality and authenticity of communications
- Security properties
  - *Confidentiality*: CCA security
  - *Integrity*: attacker cannot create new ciphertexts that decrypt properly
- Decryption returns either
  - Valid messages
  - Or invalid symbol (when ciphertext is not valid)

Combining MAC and ENC

Encryption key k_1. MAC key = k_2

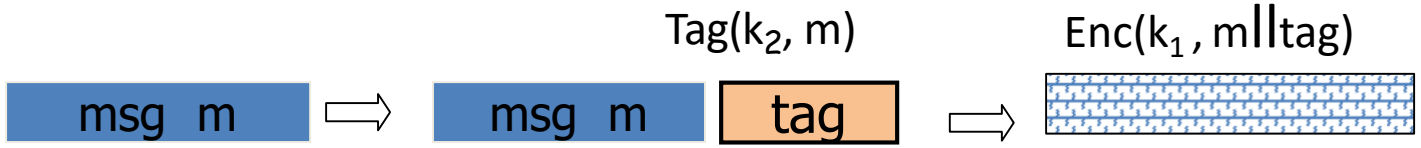
Enc-and-MAC

Option 1:(SSH)



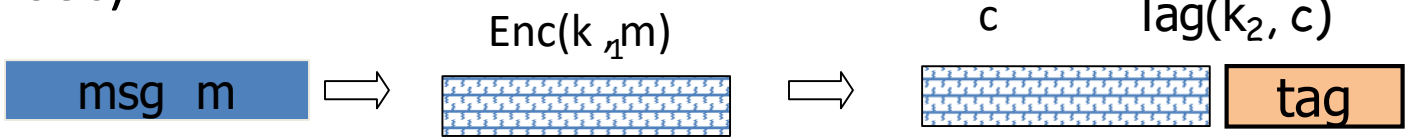
Option 2: (SSL)

MAC-then-enc



Enc-then-MAC

Option 3: (IPsec)



CCM: Counter with Cipher Block Chaining-MAC

Algorithmic ingredients:

1. AES encryption algorithm
2. CTR mode of operation
3. CMAC authentication algorithm

Single key for both encryption & MAC (symmetric key, block size is 128 bits, AES)

It is not designed to support partial processing or stream processing.

CCM Contd..

The input to CCM includes three elements:

1) Data that will be both authenticated and encrypted, called the payload;

2) Associated data: a header, that will be authenticated but not encrypted;

3) A unique value, called a **nonce**, that is assigned to the payload and the associated data

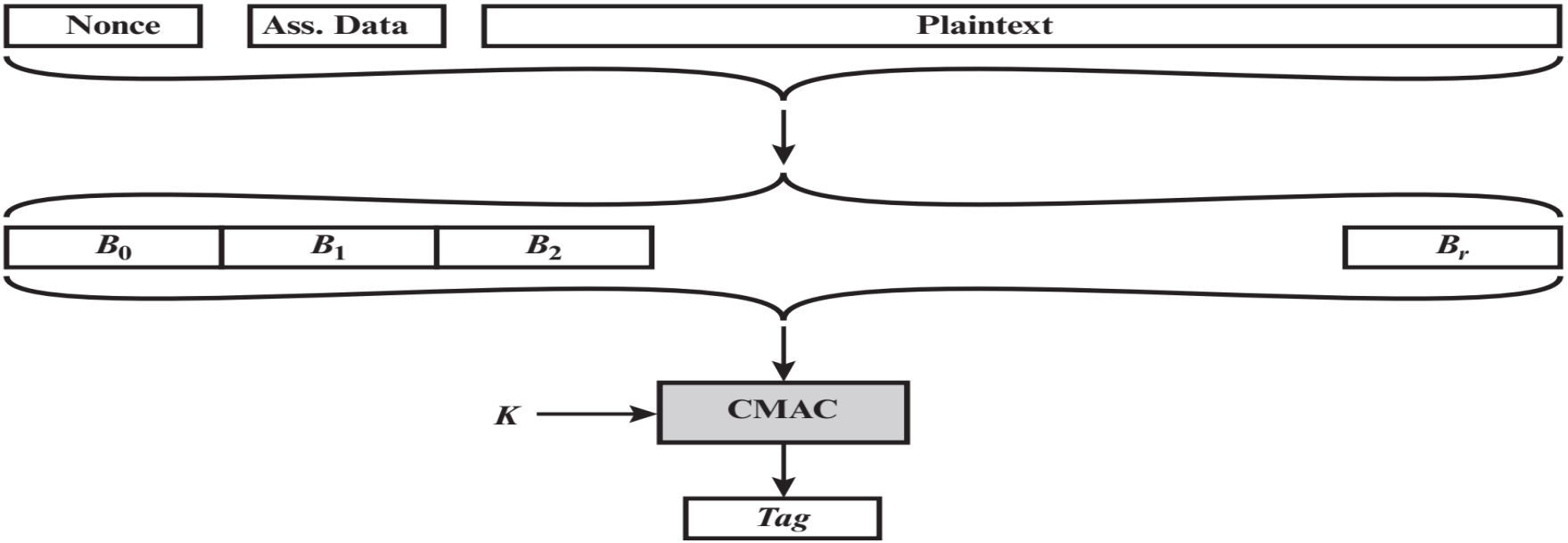
It consists of two related processes:

1. Generation-Encryption
2. Decryption-Verification

which combine two cryptographic primitives: counter mode encryption and cipher block chaining-based authentication.

CCM Generation

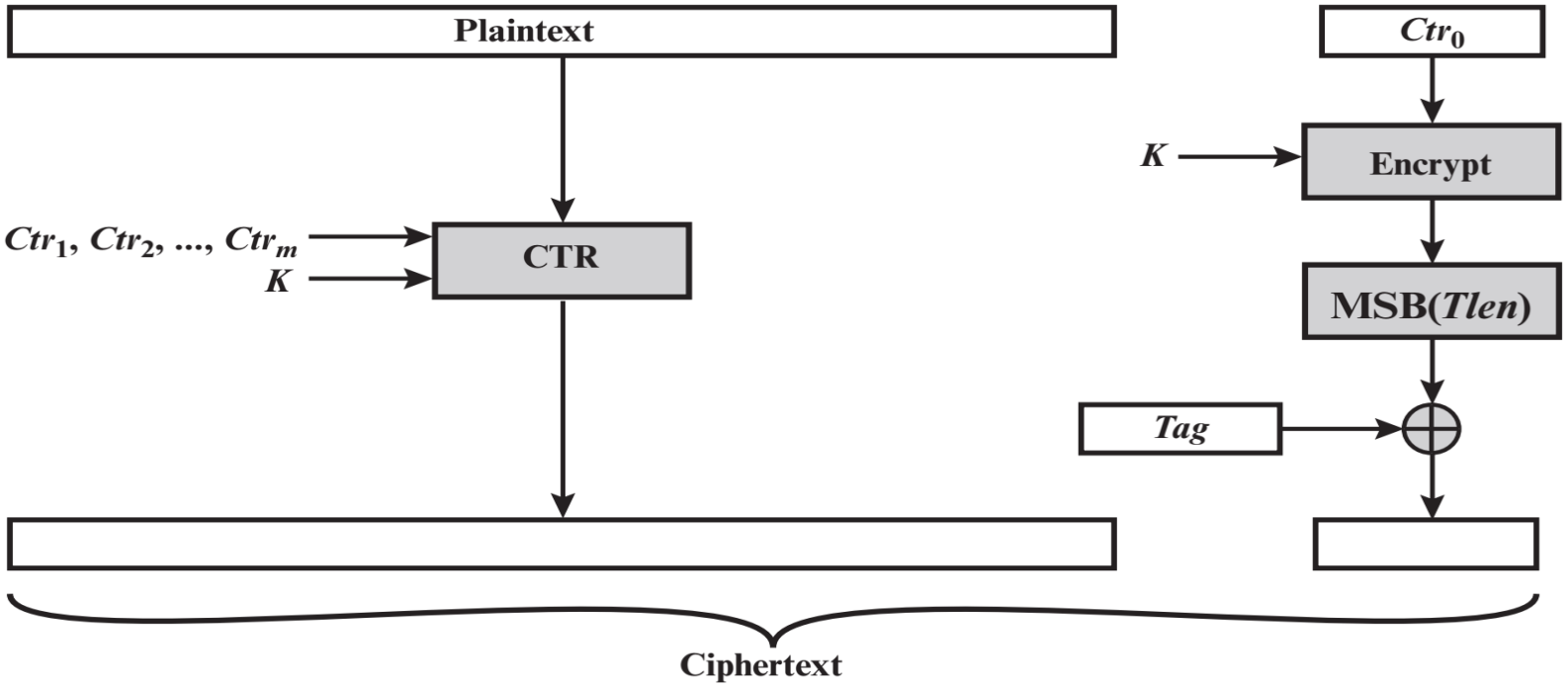
1. Apply the formatting function to (N, A, P) to produce the blocks $B_0, B_1, \ldots, B_r$.
2. Set $Y_0 = \mathrm{E}(K, B_0)$.
3. For $i = 1$ to r, do $Y_i = \mathrm{E}(K, (B_i \oplus Y_{i-1}))$.
4. Set $T = \mathrm{MSB}_{Tlen}(Y_r)$.



(a) Authentication

CCM Encryption

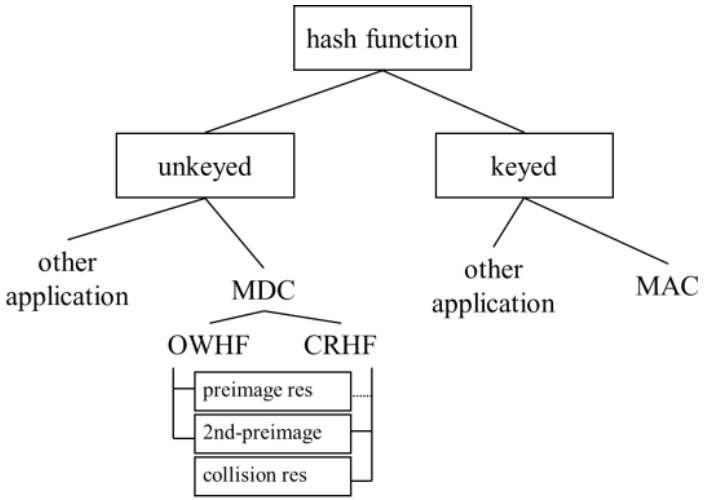
5. Apply the counter generation function to generate the counter blocks $Ctr_0, Ctr_1, \ldots, Ctr_m$, where $m = \lceil Plen/128 \rceil$.
6. For $j = 0$ to m, do $S_j = E(K, Ctr_j)$.
7. Set $S = S_1 \| S_2 \| \cdots \| S_m$.
8. Return $C = (P \oplus \text{MSB}_{Plen}(S)) \| (T \oplus \text{MSB}_{Tlen}(S_0))$.



(b) Encryption

Summary

- Hash
  - Block cipher based
  - MD5, SHA1
- MAC
  - CMAC
  - HMAC
- Authenticated Encryption
  - CCM



- Thanks