CSL759: Cryptography and Computer Security

Ragesh Jaiswal

CSE, IIT Delhi

Message Authentication

PRF as MAC

- Suppose we have a secure PRF $F: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ and suppose we only need to authenticate messages of size n, then consider the MAC associated with F:
  - $T_K(M) = F_K(M)$
  - $V_K(M, \sigma) = 1\ iff\ \sigma = F_K(M)$.
- Theorem: Consider the function family F above and the associated MAC MA. Let A be a UF-CMA adversary making q_s tag-generation queries and q_v tag-verification queries with $q_v \leq 2^{n-1}$ and having a running time t. There is a PRF adversary B such that:

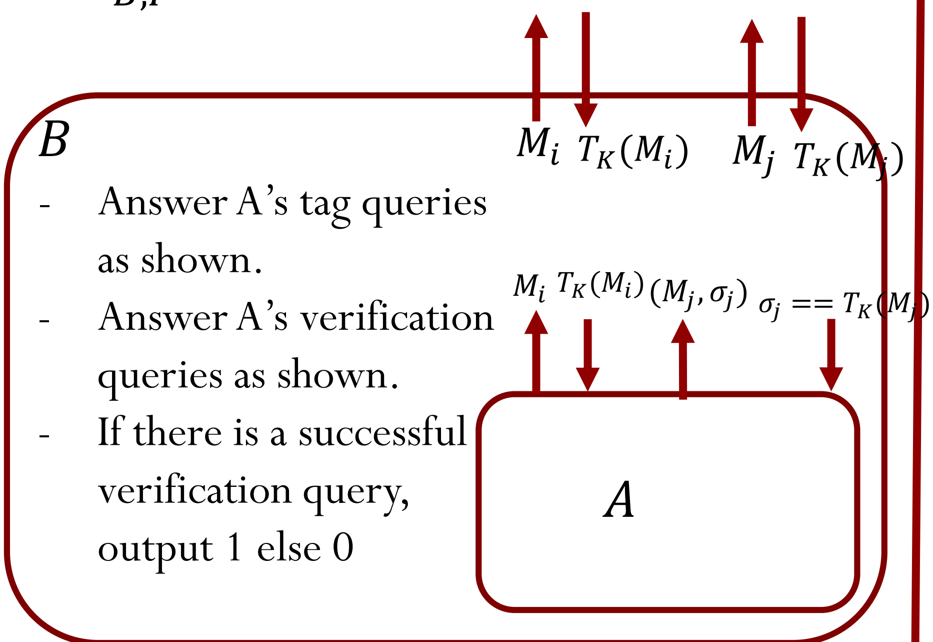
  $$Adv_{uf-cma}(A, MA) \leq Adv_{PRF}(B, F) + \frac{2q_v}{2^n}.$$

  Moreover, B makes $(q_s + q_v)$ queries and runs in time $t + \theta(n(q_s + q_v))$.

PRF as MAC

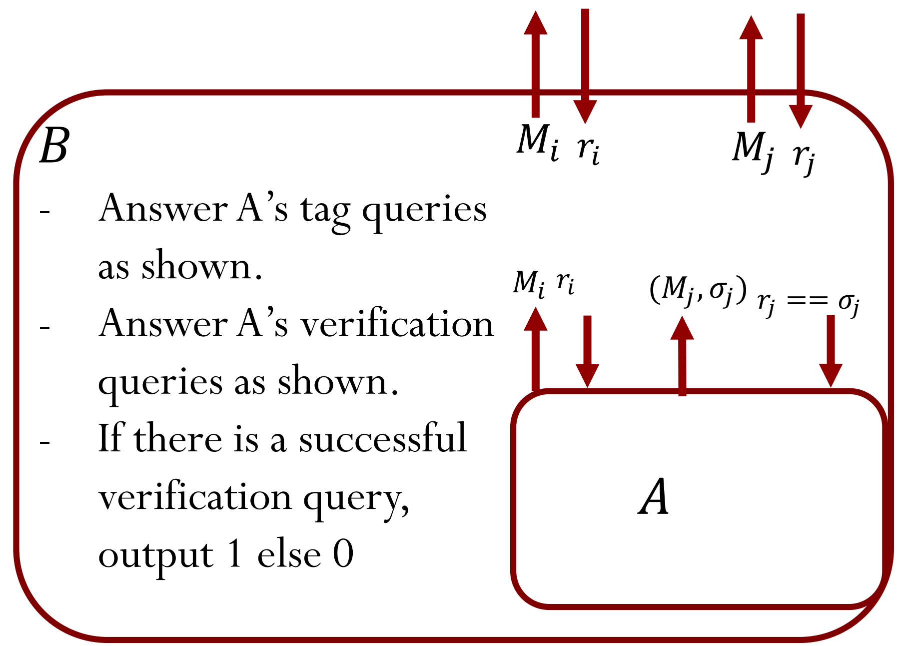
- Theorem: Consider the function family F above and the associated MAC MA. Let A be a UF-CMA adversary making q_s tag-generation queries and q_v tag-verification queries with $q_v \leq 2^{n-1}$ and having a running time t. There is a PRF adversary B such that:
$Adv_{uf-cma}(A, MA) \leq Adv_{PRF}(B, F) + \frac{2q_v}{2^n}$.
Moreover, B makes $(q_s + q_v)$ queries and runs in time $t + \theta(n(q_s + q_v))$.

$Real_{B,F}$



$\Pr[Real_{B,F} = 1] = Adv_{uf-cma}(A, MA)$

$Random_B$

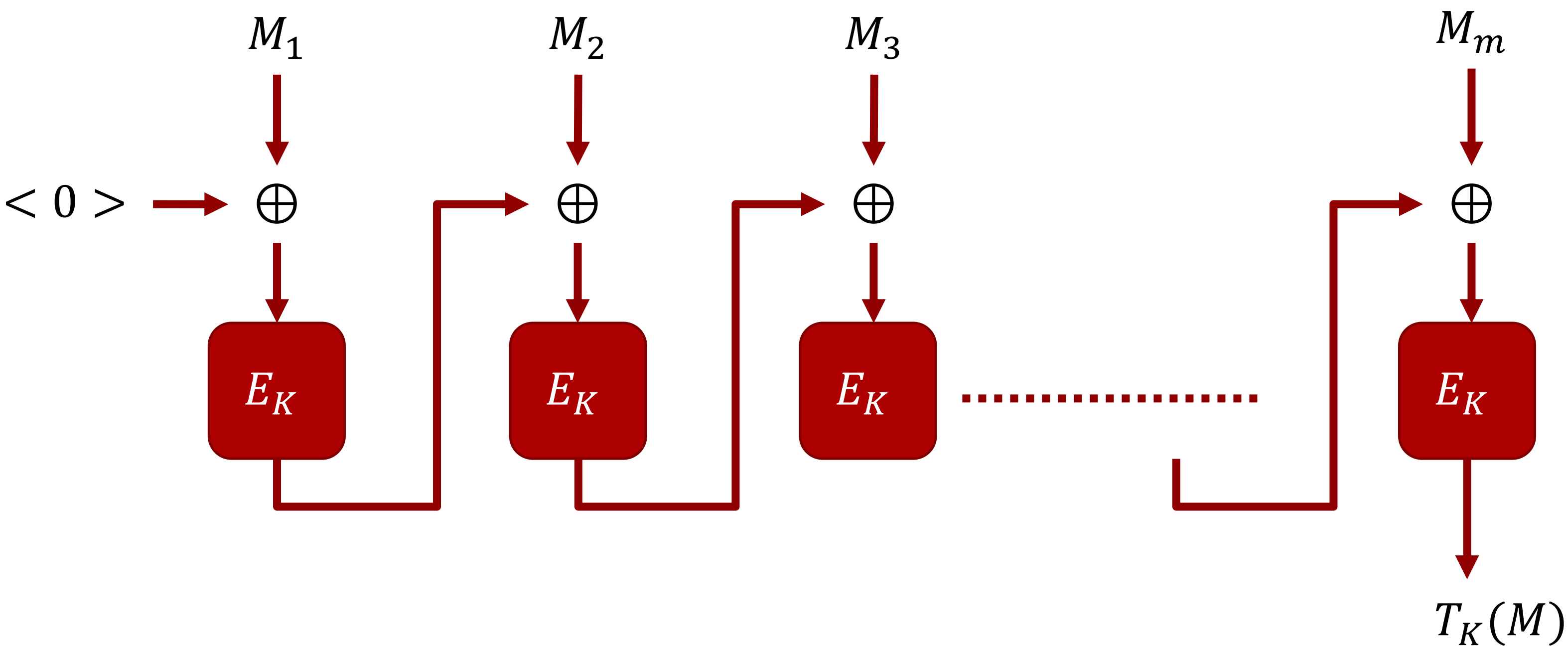


$\Pr[Random_B = 1] = \frac{2q_v}{2^n}$

MACs for arbitrary size messages

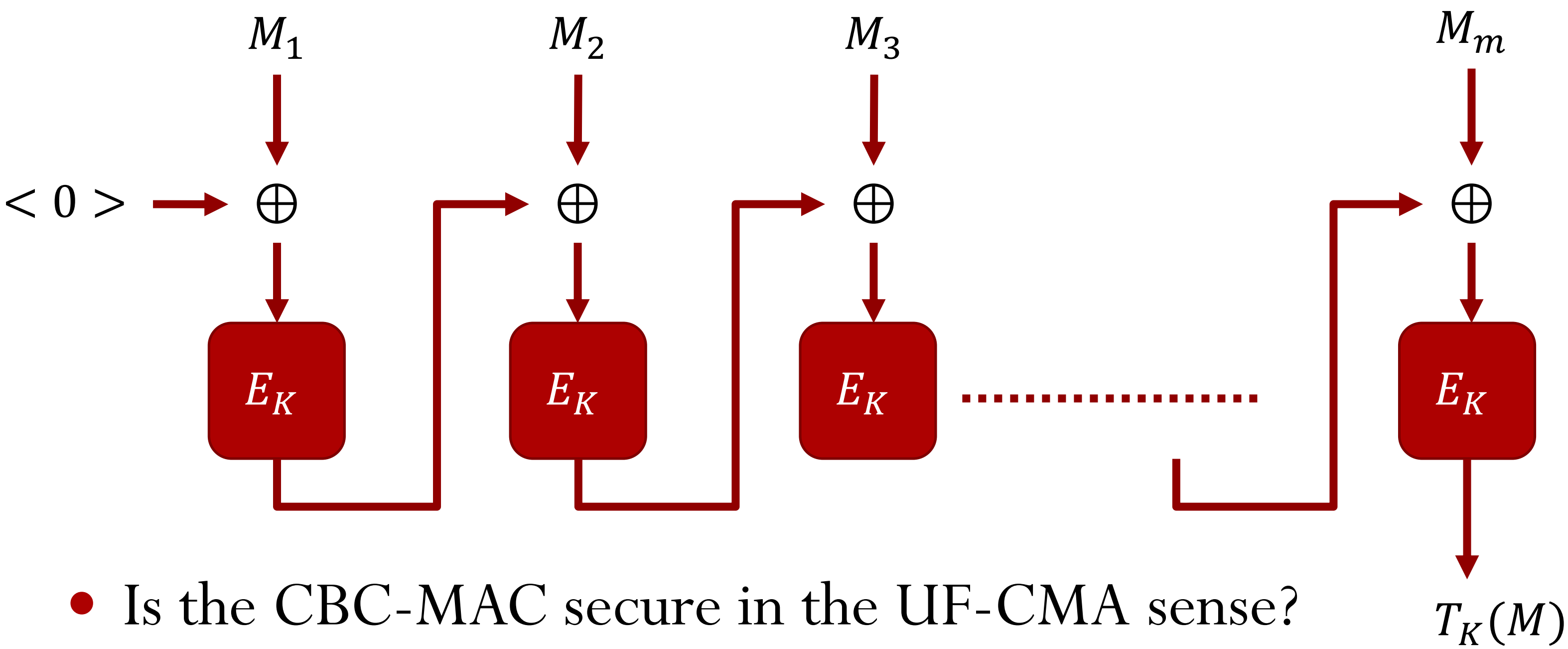
CBC MAC

- Suppose we have a secure block cipher $E\colon \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$. The tag generation algorithm is shown in the picture below:



CBC MAC

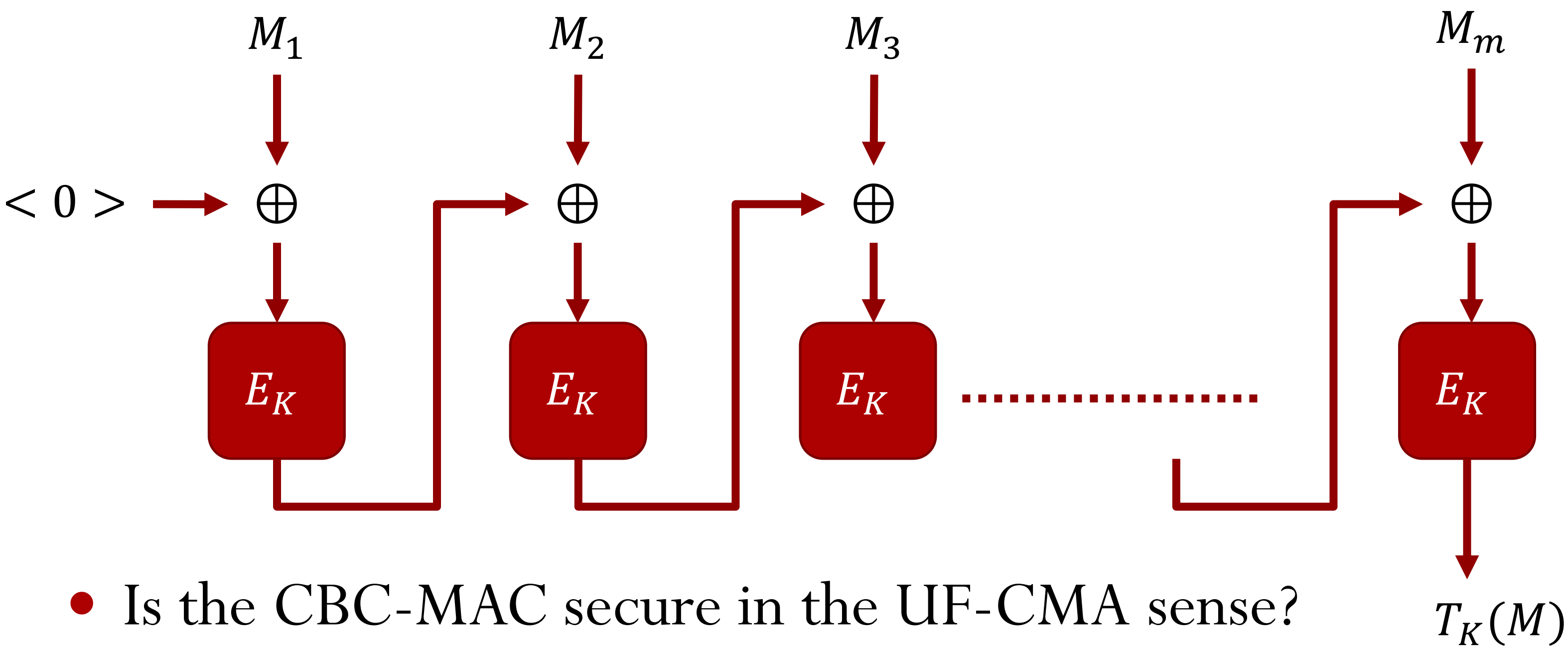
- Suppose we have a secure block cipher $E: \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$. The tag generation algorithm is shown in the picture below:



- Is the CBC-MAC secure in the UF-CMA sense?

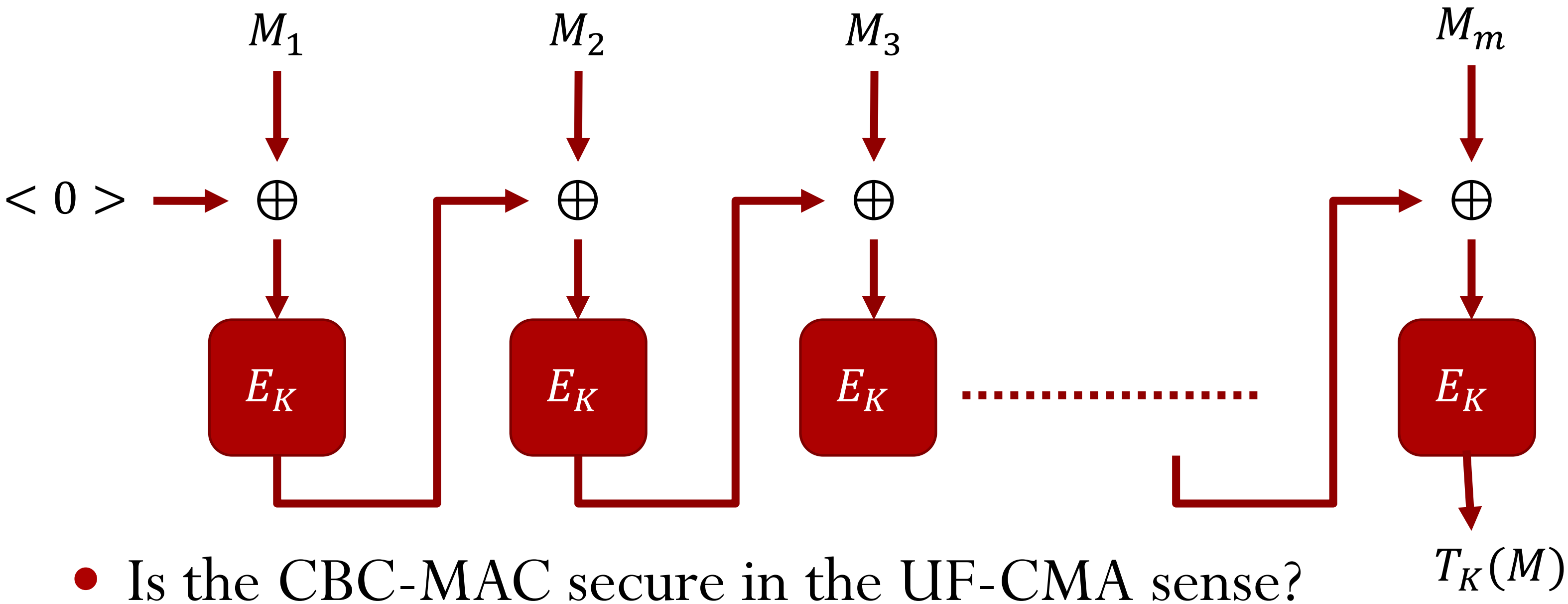
CBC MAC

- Suppose we have a secure block cipher $E: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$. The tag generation algorithm is shown in the picture below:



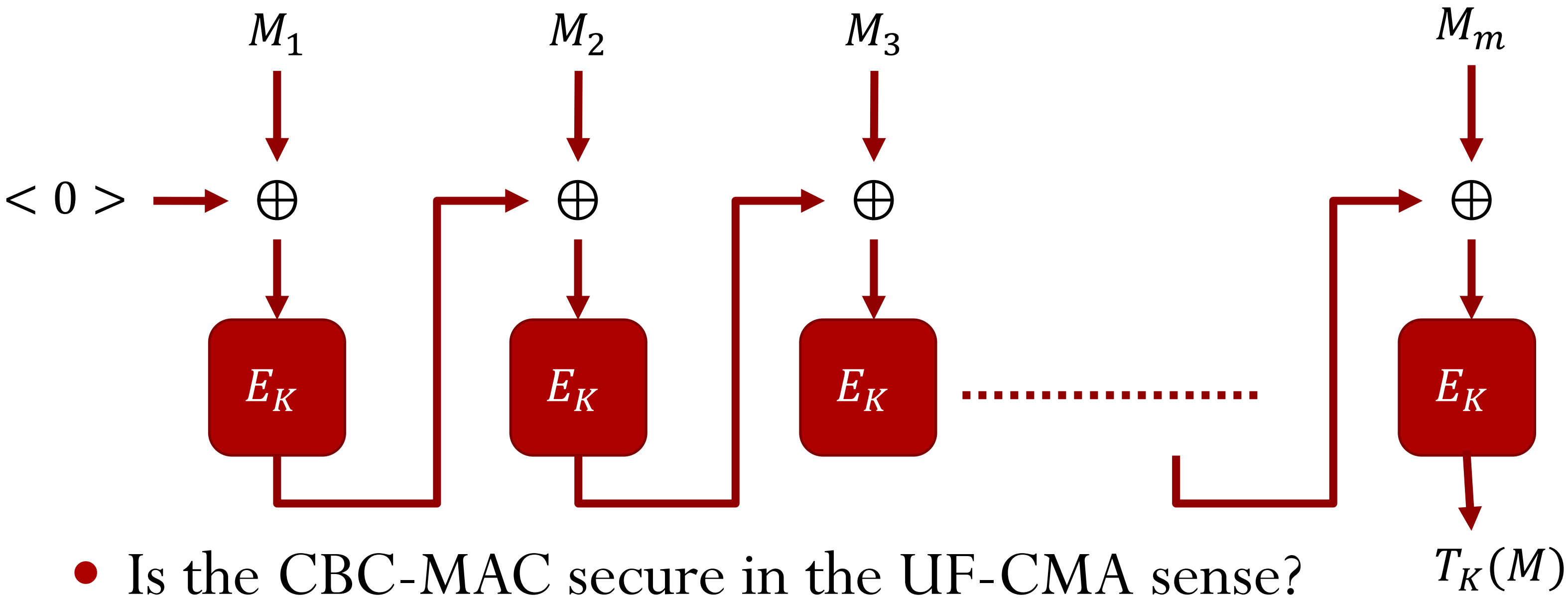
- Is the CBC-MAC secure in the UF-CMA sense?
  - No.
  - Can you give an attack?

CBC MAC



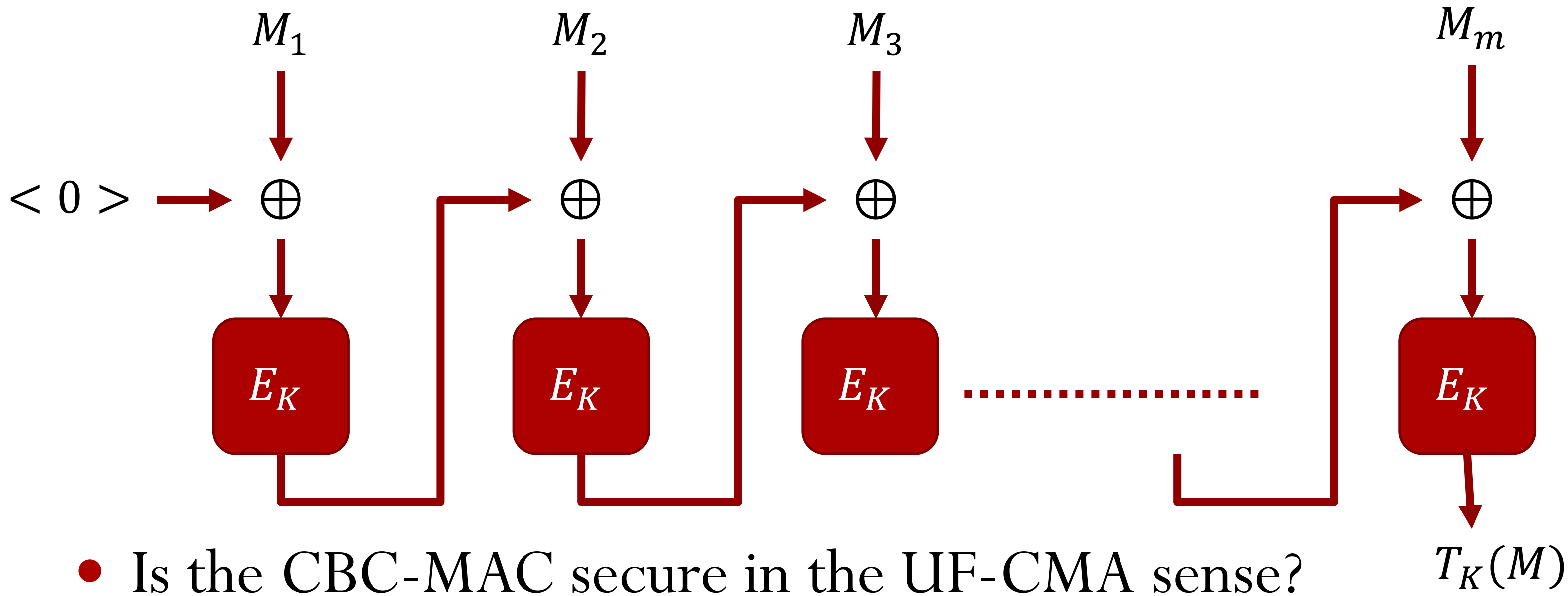
- Is the CBC-MAC secure in the UF-CMA sense?
  - No.
  - Can you give an attack?
- Adversary A
  - Make a tag-generation query x and receive the tag T.
  - Make a tag-verfification query $(x||x \oplus T, T)$.

CBC MAC



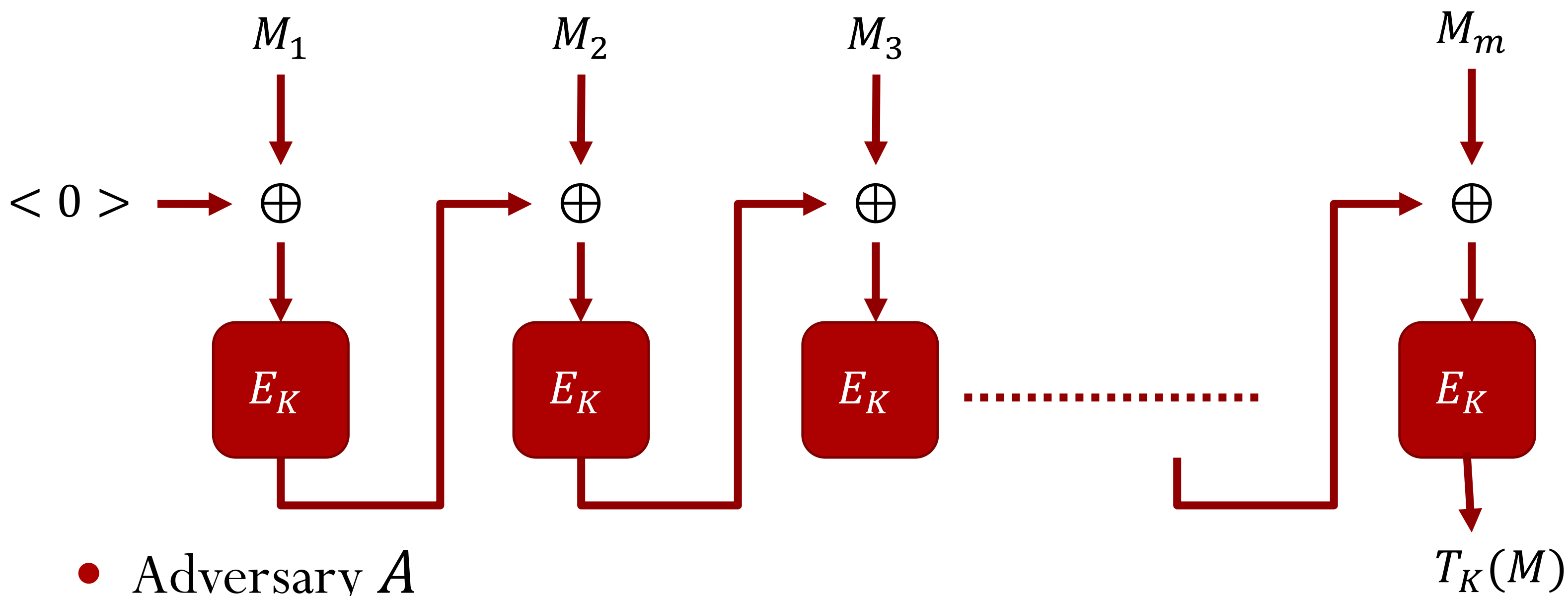
- Is the CBC-MAC secure in the UF-CMA sense?
  - No.
  - Can you give an attack?
- Adversary A
  - Make a tag-generation query x and receive the tag T.
  - Make a tag-verfification query $(x||x \oplus T, T)$.
- What is $Adv_{uf-cma}(A, MA)$?

CBC MAC



- Is the CBC-MAC secure in the UF-CMA sense?
  - No.
  - Can you give an attack?
- Adversary A
  - Make a tag-generation query x and receive the tag T.
  - Make a tag-verfification query $(x||x \oplus T, T)$.
- $Adv_{uf-cma}(A, MA) = 1.$

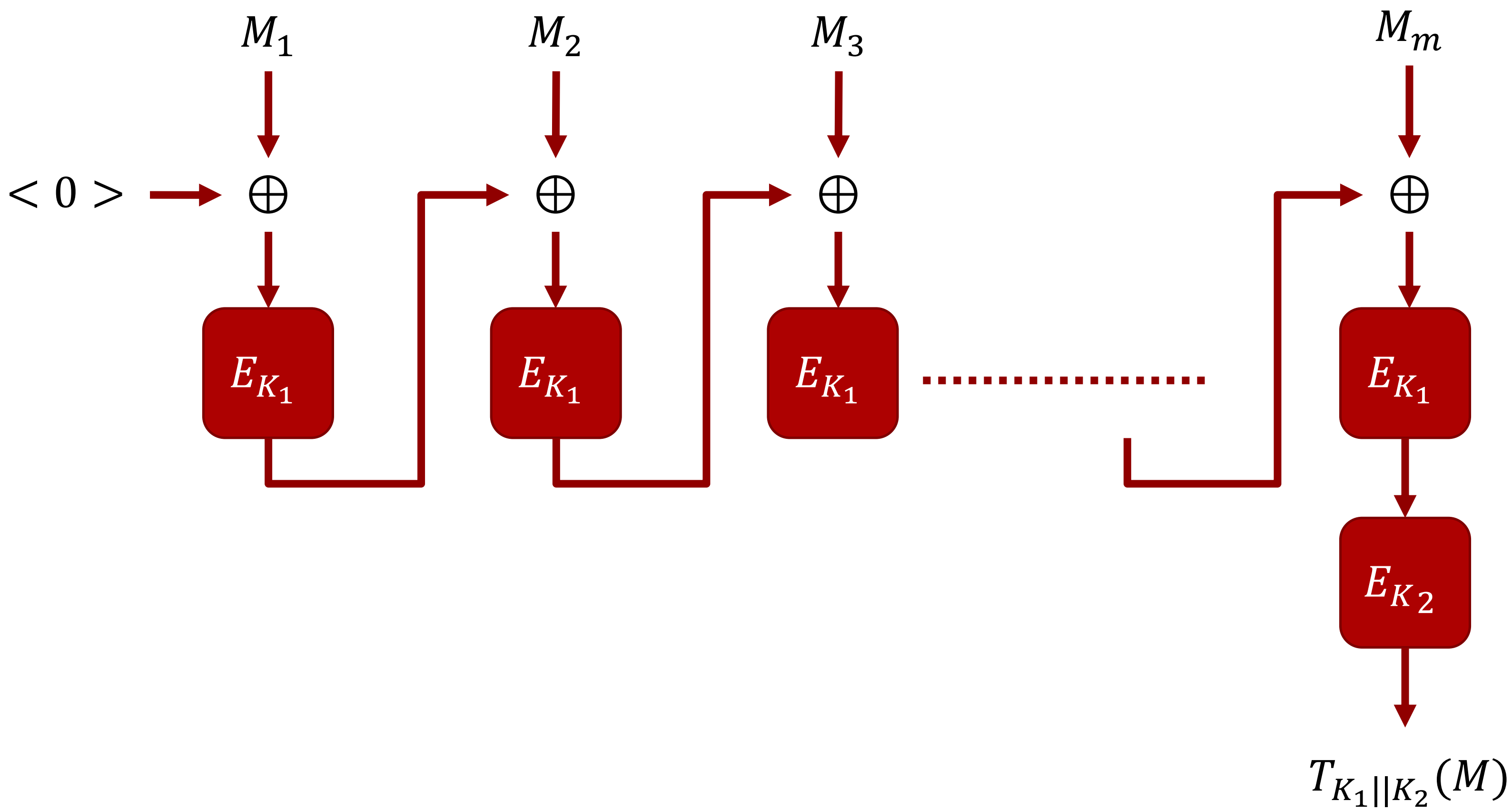
CBC MAC



- Adversary A
  - Make a tag-generation query x and receive the tag T.
  - Make a tag-verfification query $(x||x \oplus T, T)$.
- This attack is known as the *slicing attack*. The main reason it works is due to the fact that we used this MAC for message of arbitrary size.
- What if we use the authentication scheme for message of fixed size?

ECBC(Encrypted CBC) MAC

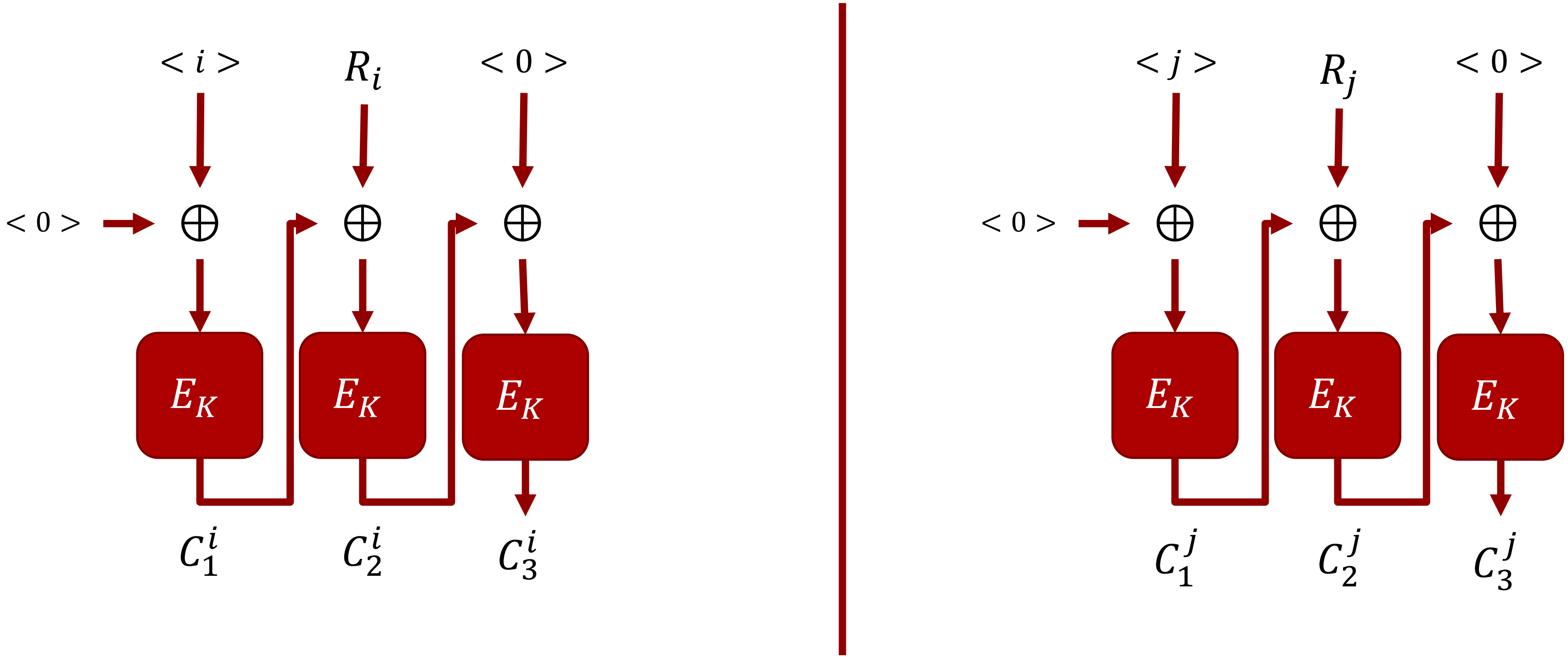
- Suppose we have a secure block cipher $E: \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$. The tag generation algorithm $T: \{0,1\}^{2k} \times \{0,1\}^L \to \{0,1\}^n$ is shown in the picture below:



Birthday attack on Chaining based MACs

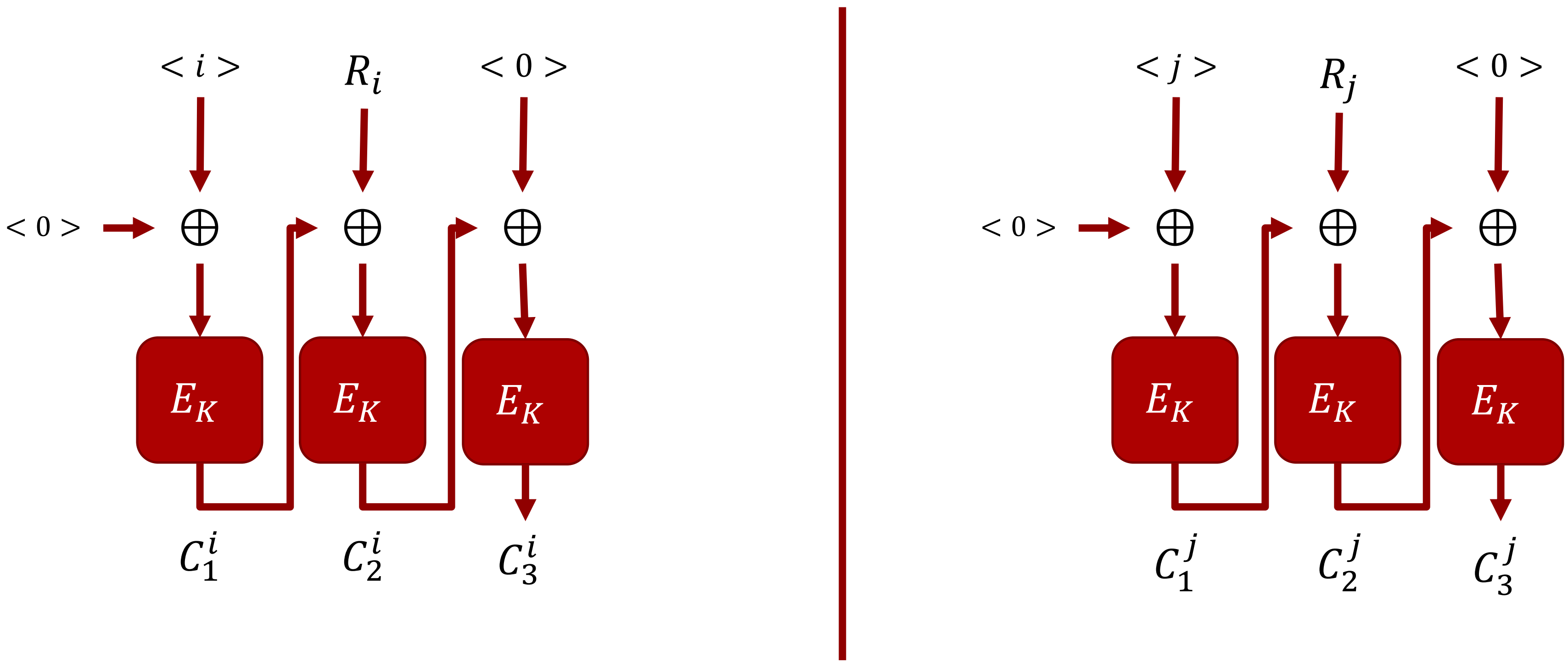
Birthday Attack on CBC MAC

- <u>Main Idea</u>: *Internal collision*. Consider message spanning 3 blocks.



Birthday Attack on CBC MAC

- Main Idea: *Internal collision*. Consider message spanning 3 blocks.



- If $C_2^i = C_2^j$, then
  $\forall x, T_K(< i >, R_i, x) = T_K(< j >, R_j, x)$.

Birthday Attack on CBC MAC

- Adversary A
  - For $i = 1$ to q
    - Randomly pick $R_i \in \{0,1\}^n$
    - Make a tag-generation query $(< i > || R_i || < 0 >)$ and receive the tag T_i.
  - If there exists indices $i \neq j$ such that $T_i = T_j$
    - Make a tag-generation query $(< i > || R_i || < 1 >)$ and receive the tag T.
    - Make a tag-verification query $(< j > || R_j || < 1 >, T)$.
- What is $Adv_{uf-cma}(A, MA)$?

Birthday Attack on CBC MAC

- Adversary A
  - For $i = 1$ to q
    - Randomly pick $R_i \in \{0,1\}^n$
    - Make a tag-generation query $(< i > ||R_i|| < 0 >)$ and receive the tag T_i.
  - If there exists indices $i \neq j$ such that $T_i = T_j$
    - Make a tag-generation query $(< i > ||R_i|| < 1 >)$ and receive the tag T.
    - Make a tag-verification query $(< j > ||R_j|| < 1 >, T)$.

- $Adv_{uf-cma}(A, MA) = C(q, 2^n)$.

Birthday Attack on CBC MAC

- Adversary A
  - For $i = 1$ to q
    - Randomly pick $R_i \in \{0,1\}^n$
    - Make a tag-generation query $(< i > ||R_i|| < 0 >)$ and receive the tag T_i.
  - If there exists indices $i \neq j$ such that $T_i = T_j$
    - Make a tag-generation query $(< i > ||R_i|| < 1 >)$ and receive the tag T.
    - Make a tag-verification query $(< j > ||R_j|| < 1 >, T)$.

- $Adv_{uf-cma}(A, MA) = C(q, 2^n)$.
- Does there exist an adversary that does much better that A?

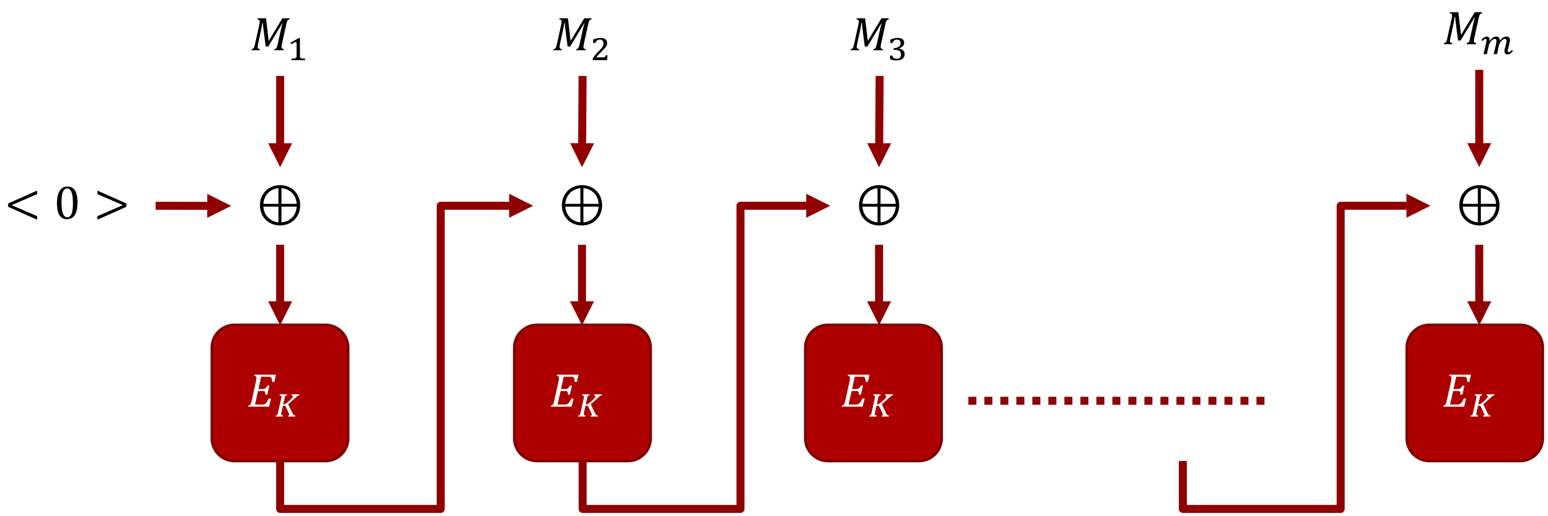
Birthday Attack on CBC MAC

- Adversary A
  - For $i = 1$ to q
    - Randomly pick $R_i \in \{0,1\}^n$
    - Make a tag-generation query $(< i > || R_i || < 0 >)$ and receive the tag T_i.
  - If there exists indices $i \neq j$ such that $T_i = T_j$
    - Make a tag-generation query $(< i > || R_i || < 1 >)$ and receive the tag T.
    - Make a tag-verification query $(< j > || R_j || < 1 >, T)$.

- $Adv_{uf-cma}(A, MA) = C(q, 2^n)$.
- Does there exist an adversary that does much better that A?
  - No.

Security of CBC MAC

- Theorem: Let $E: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ be a family of functions. For any integer $m \geq 1$, conside the function family $E^m: \{0,1\}^k \times \{0,1\}^{nm} \rightarrow \{0,1\}^n$ defined as below:



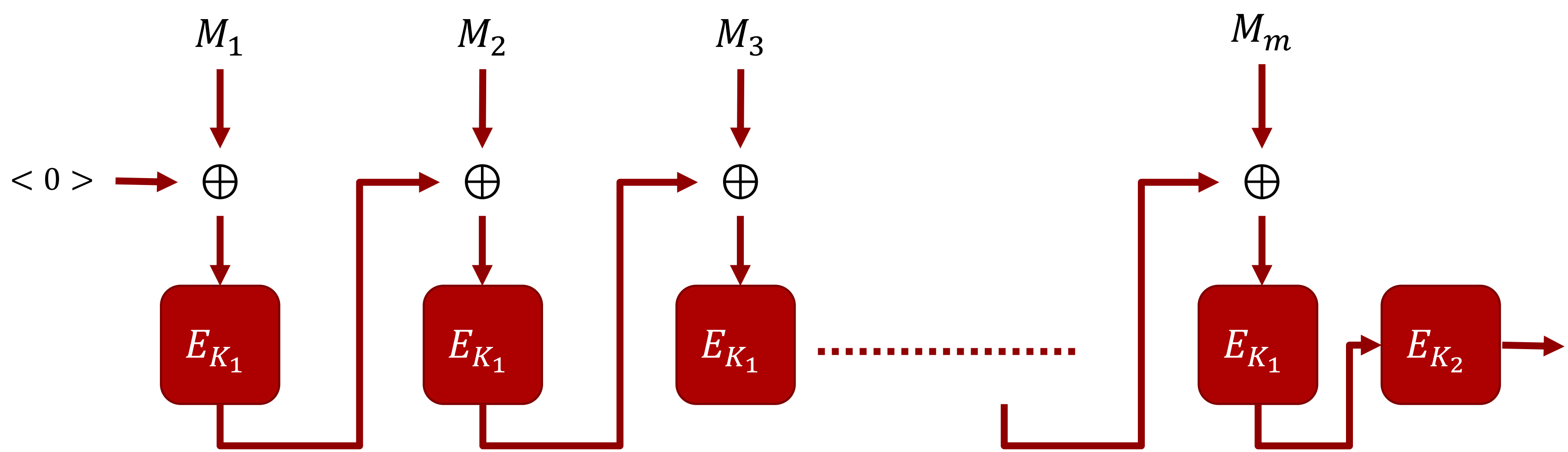
Let A be a PRF adversary against E^m that makes q oracle queries and has a running time of t. Then there is a PRF adversary B against E such that

$$Adv_{PRF}(A, E^m) \leq Adv_{PRF}(B, E) + \frac{q^2 m^2}{2^n}$$

and B makes at most qm oracle queries and runs in time t.

Security of ECBC MAC

- <u>Theorem</u>: Let $E: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ be a family of functions. Conside the function family $\mathrm{F}: \{0,1\}^{2k} \times \{0,1\}^{\leq L} \rightarrow \{0,1\}^n$ defined as below:



Let A be a PRF adversary against F that makes q oracle queries totalling σ blocks and has a running time of t. Then there is a PRF adversary B against E such that

$$Adv_{PRF}(A, F) \leq Adv_{PRF}(B, E) + \frac{\sigma^2}{2^n}$$

and B makes at most σ oracle queries and runs in time t.

Case Study: Block Cipher based MACs

CMAC

Case Study: CMAC

CMAC Components and Setup

- $E : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ is a block cipher, in practice AES.
- $\mathrm{CBC}_K(M)$ is the basic CBC MAC of a full message M under key $K \in \{0,1\}^n$ and using E.
- $J \in \{0,1\}^n$ is a particular fixed constant.

CMAC uses its key $K \in \{0,1\}^n$ to derive subkeys K_1, K_2 via

- $K_0 \leftarrow E_K(0)$
- if $\mathrm{msb}(K_0) = 0$ then $K_1 \leftarrow (K_0 \ll 1)$ else $K_1 \leftarrow (K_0 \ll 1) \oplus J$
- if $\mathrm{msb}(K_1) = 0$ then $K_2 \leftarrow (K_1 \ll 1)$ else $K_2 \leftarrow (K_1 \ll 1) \oplus J$

where $x \ll 1$ means x left shifted by 1 bit, so that the msb vanishes and the lsb becomes 0. These bit operations reflect simple finite-field operations.

Case Study: CMAC

CMAC Algorithm

Alg $\mathrm{CMAC}_K(M)$

$M[1] \ldots M[m-1]M[m] \leftarrow M \quad // \ |M[m]| \leq n$

$\ell \leftarrow |M[m]| \quad // \ \ell \leq n$

if $\ell = n$ then $M[m] \leftarrow K_1 \oplus M[m]$

else $M[m] \leftarrow K_2 \oplus (M[m] \| 10^{n-\ell-1})$

$M \leftarrow M[1] \ldots M[m-1]M[m]$

$T \leftarrow \mathrm{CBC}_K(M)$

return T

- Splicing attack does not work.
- There is a security proof showing that no attack is significantly better than the Birthday attack.
- NIST Standard for Message Authentication.

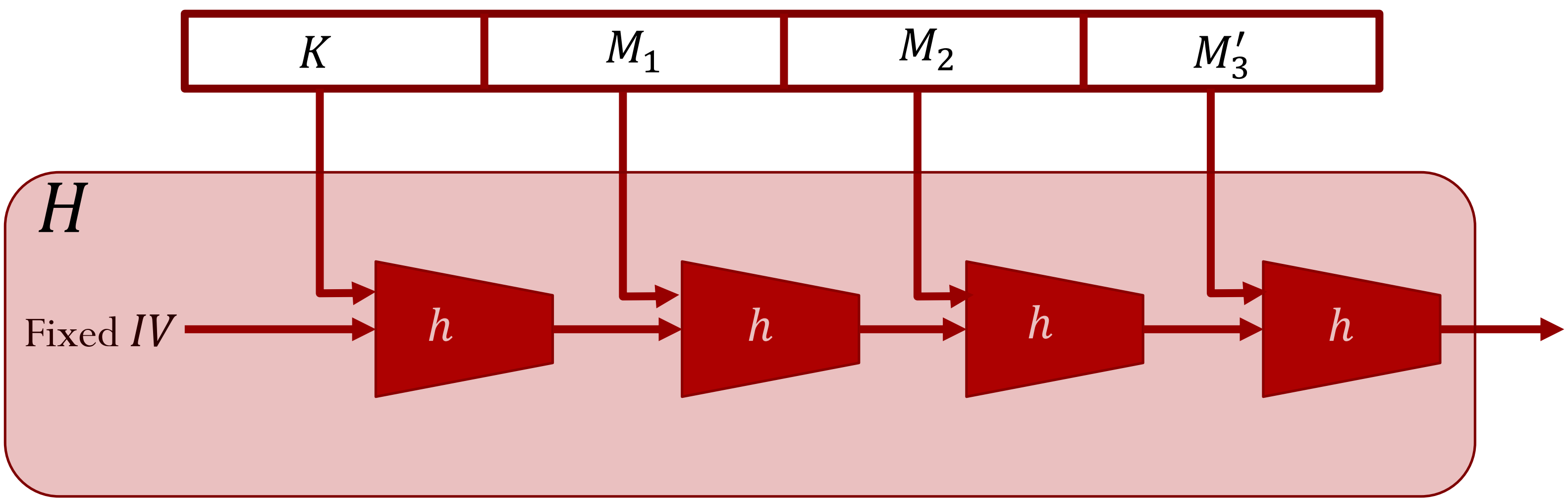
Hash Function based MACs

Hash Function based MACs

- Can we construct a secure MAC using collision-resistant hash functions?
  - Issue: Hash functions are *keyless*.
- What if we use $T_K(M) = H(K||M)$? Is this secure?

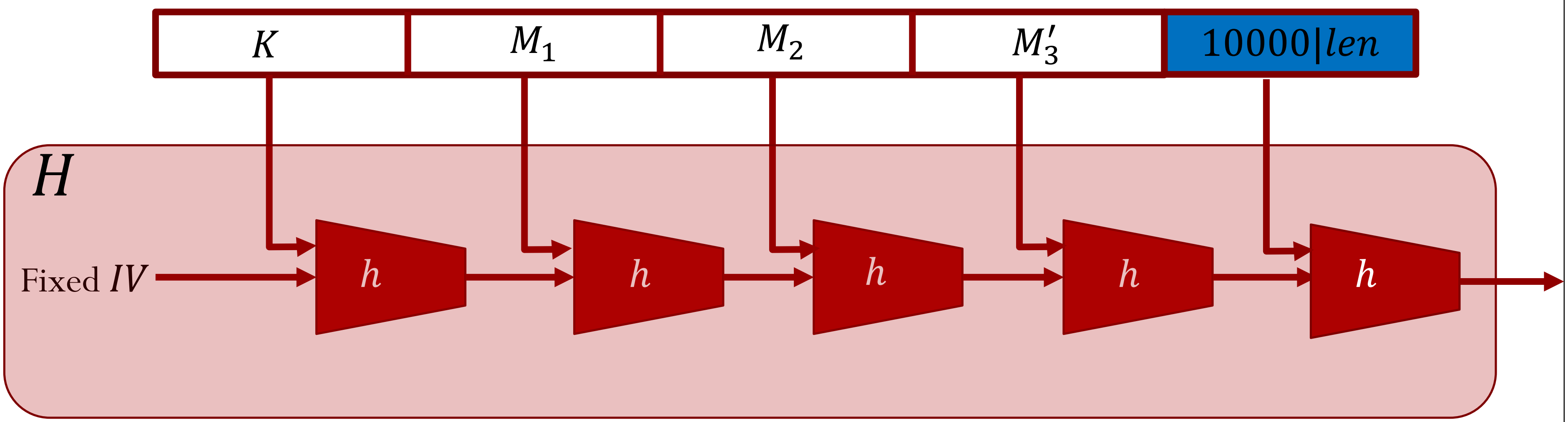
Hash Function based MACs

- Can we construct a secure MAC using collision-resistant hash functions?
  - Issue: Hash functions are *keyless*.
- What if we use $T_K(M) = H(K||M)$? Is this secure?
  - No. There can be an *extension attack*.



Hash Function based MACs

- Can we construct a secure MAC using collision-resistant hash functions?
  - Issue: Hash functions are *keyless*.
- What if we use $T_K(M) = H(K||M)$? Is this secure?
  - No. There can be an *extension attack*.



- The tag for $M = M_1||M_2||M_3$ gives the correct tag for $M_1||M_2||M_3'$.

Hash Function based MAC: HMAC

HMAC [BCK96]

Suppose $H : D \rightarrow \{0,1\}^{160}$ is the hash function. HMAC has a 160-bit key K. Let

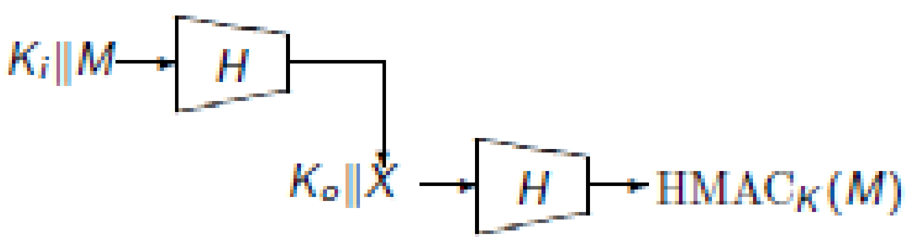
$$K_o = \text{opad} \oplus K||0^{352} \text{ and } K_i = \text{ipad} \oplus K||0^{352}$$

where

$$\text{opad} = 5D \text{ and ipad} = 36$$

in HEX. Then

$$\text{HMAC}_K(M) = H(K_o||H(K_i||M))$$



Hash Function based MAC: HMAC

HMAC

Features:

- Blackbox use of the hash function, easy to implement
- Fast in software

Usage:

- As a MAC for message authentication
- As a PRF for key derivation

Security:

- Subject to a birthday attack
- Security proof shows there is no better attack [BCK96,Be06]

Adoption and Deployment: HMAC is one of the most widely standardized and used cryptographic constructs: SSL/TLS, SSH, IPSec, FIPS 198, IEEE 802.11, IEEE 802.11b, ...

Hash Function based MAC: HMAC

HMAC Security

Theorem: [BCK96] HMAC is a secure PRF assuming

- The compression function is a PRF
- The hash function is collision-resistant (CR)

But recent attacks show MD5 is not CR and SHA1 may not be either.

So are HMAC-MD5 and HMAC-SHA1 secure?

- No attacks so far, but
- Proof becomes vacuous!

Theorem: [Be06] HMAC is a secure PRF assuming only

- The compression function is a PRF

Current attacks do not contradict this assumption. This new result may explain why HMAC-MD5 is standing even though MD5 is broken with regard to collision resistance.

Hash Function based MAC: HMAC

HMAC Recommendations

- Don't use HMAC-MD5
- No immediate need to remove HMAC-SHA1
- Use HMAC-SHA256 for new applications

MACs using Universal Hash Function Families

Carter-Wegman

Carter-Wegman MACs

- Chain based constructions like ECBC, HMAC are expensive as it involves repeated executions of a block cipher.
- Definition (δ-almost universal hash function family): A function family $H: keys(H) \times D \rightarrow \{0,1\}^n$ is called δ-almost-universal hash function if for all $M_1, \neq M_2 \in D$:

$$\Pr[H_K(M_1) = H_K(M_2)] \leq \delta$$

Carter-Wegman MACs

- Chain based constructions like ECBC, HMAC are expensive as it involves repeated executions of a block cipher.
- <u>Definition (δ-almost universal hash function family)</u>: A function family $H: keys(H) \times D \to \{0,1\}^n$ is called δ-almost-universal hash function if for all $M_1, \neq M_2 \in D$:
$$\Pr[H_K(M_1) = H_K(M_2)] \leq \delta$$
- Example of almost universal hash function family.
  - Let p be a large prime (say $\geq 2^{128}$)
  - $K = (a, b) \in \{1 \ldots q\} \times \{1 \ldots q\}$
  - $H_K(M) = (a^{m+1} + M_m \cdot a^m + \cdots + M_1 \cdot a + b) \ (mod\ p)$

Carter-Wegman MACs

- Carter-Wegman MAC
  - Suppose we have a δ-almost-universal hash function family $H: keys(H) \times D \to \{0,1\}^n$ and a secure PRF $E: \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$, consider the following many-time MAC for messages in the domain D:
    - $T_K(M) = (r, E_{K_1}(r) \oplus H_{K_2}(M))$, where $K \in keys(H) \times \{0,1\}^k$.
  - Theorem(informal): The above MAC is UF-CMA secure assuming that E is a secure PRF and H is almost-universal.
  - Examples:
    - UMAC: (NH + HMAC-SHA1)
    - Poly127-AES: (Poly127 + AES)
    - Poly1305-AES: (Poly1305 + AES)

End

The following slides have been borrowed from Mihir Bellare's Course on Cryptography: 24, 25, 30, 31, 32, 33.