Security Analyses and Enhanced Variants of Standardized MAC Algorithms

Yaobin Shen

Xiamen University

Dec 17, 2024@ASK 2024

Content

1 **Background**

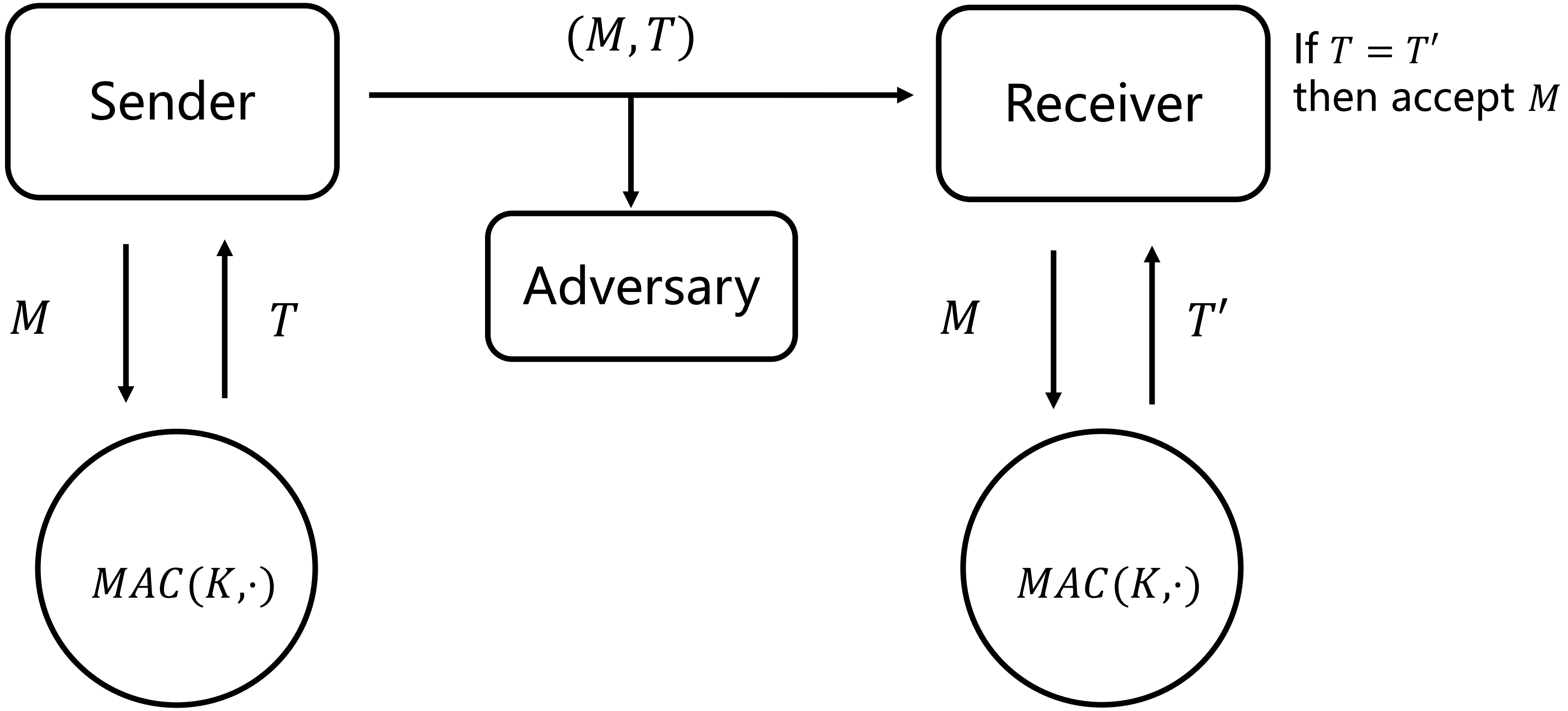
2 Analysis and Enhances

3 Discussion

Message authentication codes (MAC)

Definition

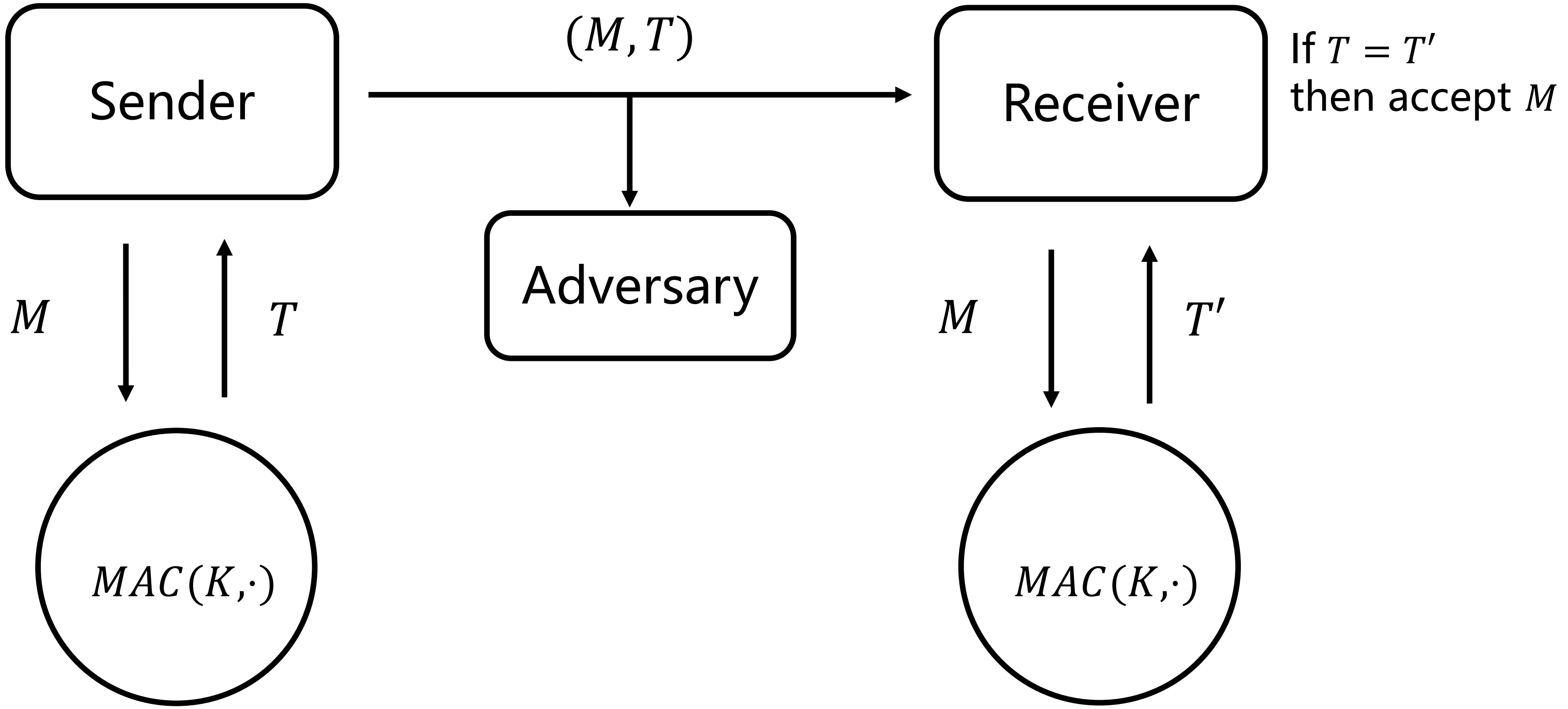
The secret key K is shared, tag generation $T \leftarrow MAC(K, M)$



the shared key K provides authenticity
the tag T ensures integrity

Security property

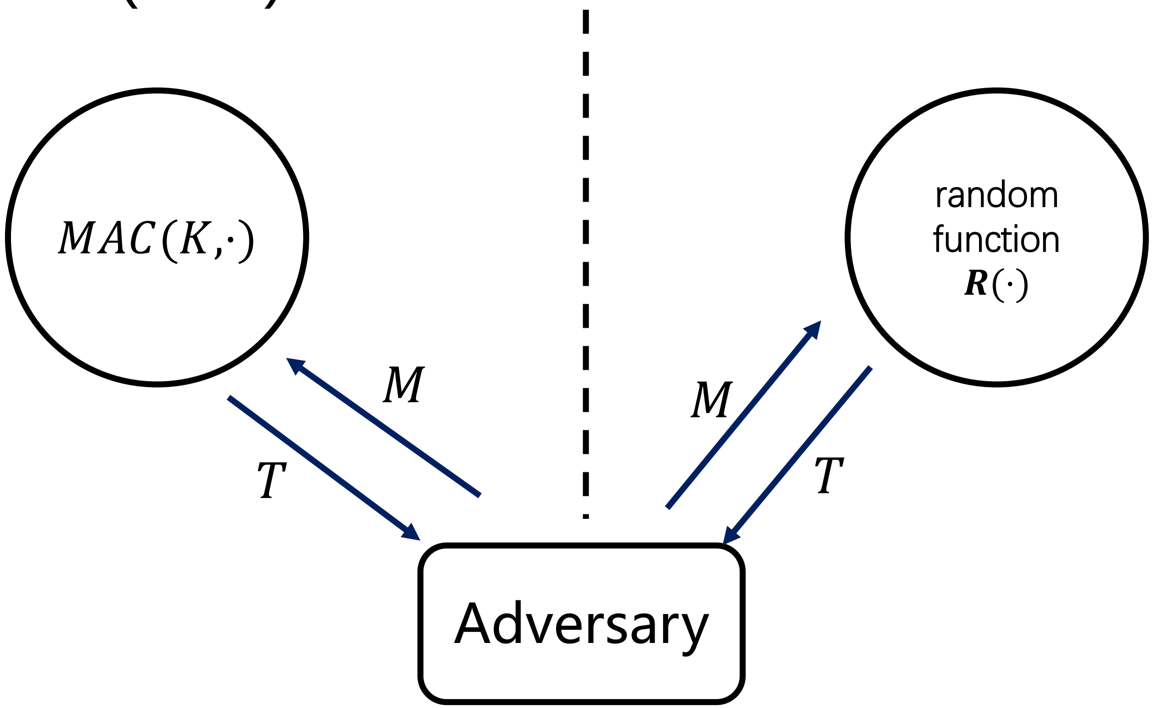
- Unforgeability



- The adversary obtains $\{(M_1, T_1), (M_2, T_2), \ldots (M_q, T_q)\}$, outputs (M', T')
- If $M' \notin \{M_1, M_2, \ldots, M_q\}$ and $T' = MAC(K, M')$, then forges successfully

Security property

- Pseudorandomness（PRF）



- The adversary cannot distinguish $MAC(K,\cdot)$ from a random function
- PRF implies unforgeability

Content

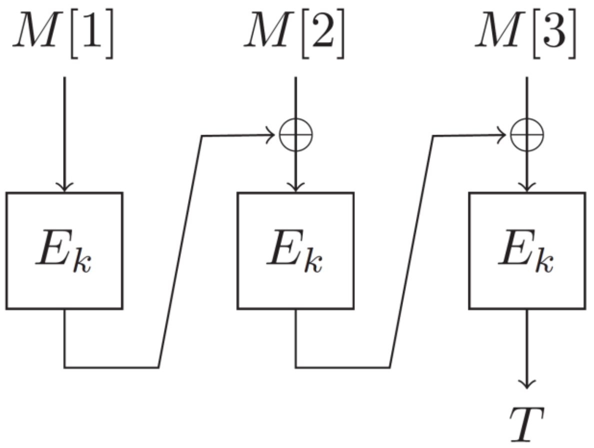
1 Background

2 **Analyses and Enhances**

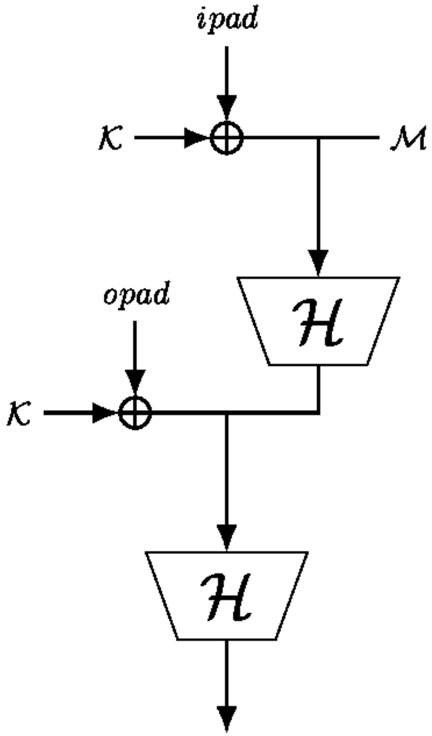
3 Discussion

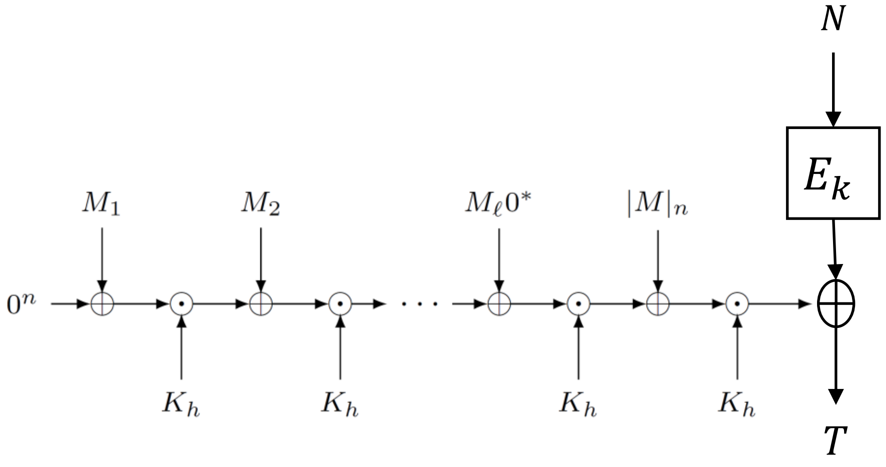
Categories

- Block cipher-based MACs
- Hash-based MACs
- Universal-hash-based MACs



CBC-MAC





GMAC

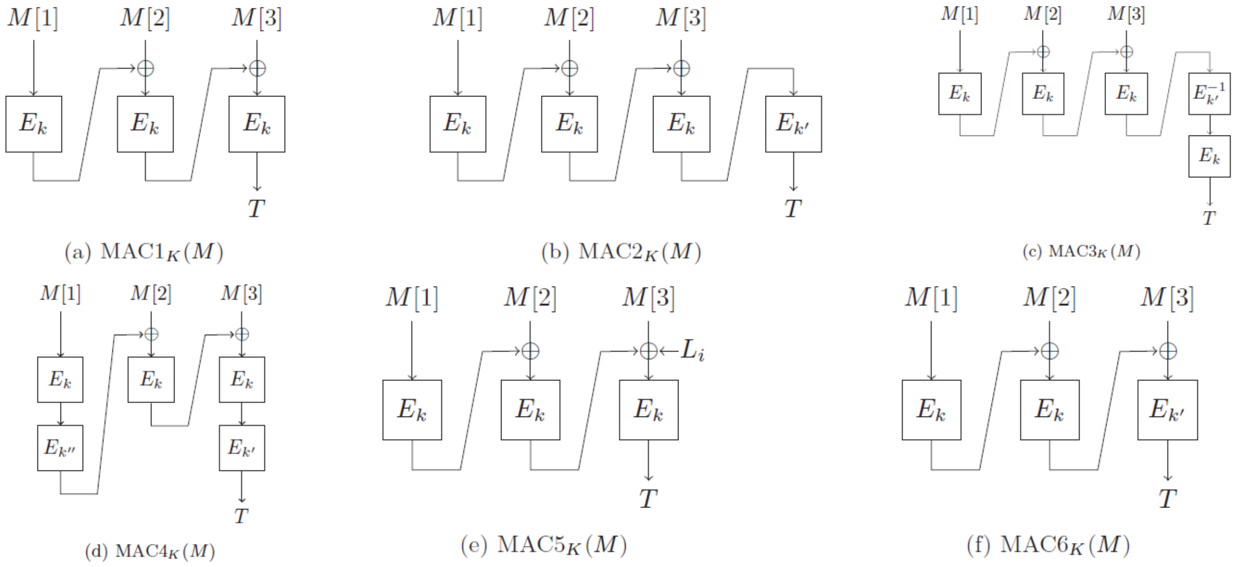
Standardized MACs

Xiamen University

Standards organization	Recommended MACs	Specification
ISO - International Organization for Standardization	CBC-MAC	ISO/IEC9797-1
	HMAC、MDx-MAC	ISO/IEC9797-2
	UMAC, Badger, Poly1305, GMAC	ISO/IEC9797-3
	LightMAC、 Tsudik's keymode、 Chaskey-12	ISO/IEC 29192-6
The 3rd Generation Partnership Project (3GPP)	UIA1 （f9)	2G/3G integrity algorithms
	UIA2 （SNOW + a variant of GMAC)	
	128EIA1 （SNOW + a variant of GMAC)	4G/5G integrity algorithms
	128EIA2 (AES + CMAC)	
	128EIA3 (ZUC + a variant of GMAC)	

ISO/IEC 9797-1:2011

- Specifies 6 different variants of CBC-MACs

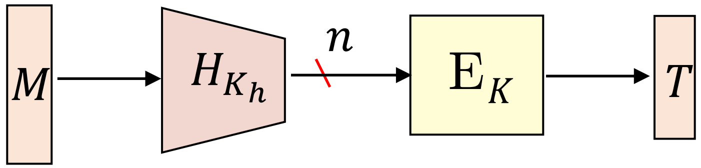


ISO/IEC 9797-1:2011 MACs.

- Provides with four padding schemes
  - pad1: $X||0^*$
  - pad2: $X||10^*$
  - pad3: $bin_n(|X|)||X||0^*$
  - pad4: X if $|X|\ mod\ n = 0$, otherwise $X||10^*$ (only for MAC5)

Birthday-bound security

- Collision on the n-bit internal value



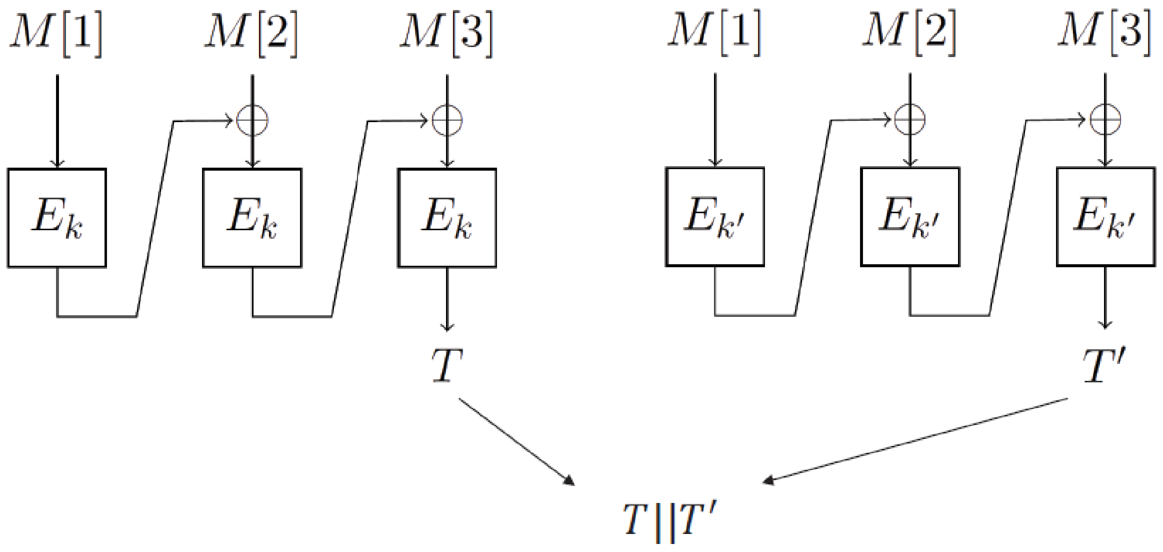
  - $H_{K_h}(M_1) = H_{K_h}(M_2) \Rightarrow T_1 = T_2$
  - at most birthday bound security $O(2^{\frac{n}{2}})$
- Limitations on the birthday bound
  - lightweight blockciphers or TDES typically have 64-bit block, 2^{32} is vulnerable
  - Even for 128-bit block cipher like AES, worse bound will decrease the key lifetime

[BL16] Sweet32

ISO/IEC 9797-1:2011’ s Recommendation

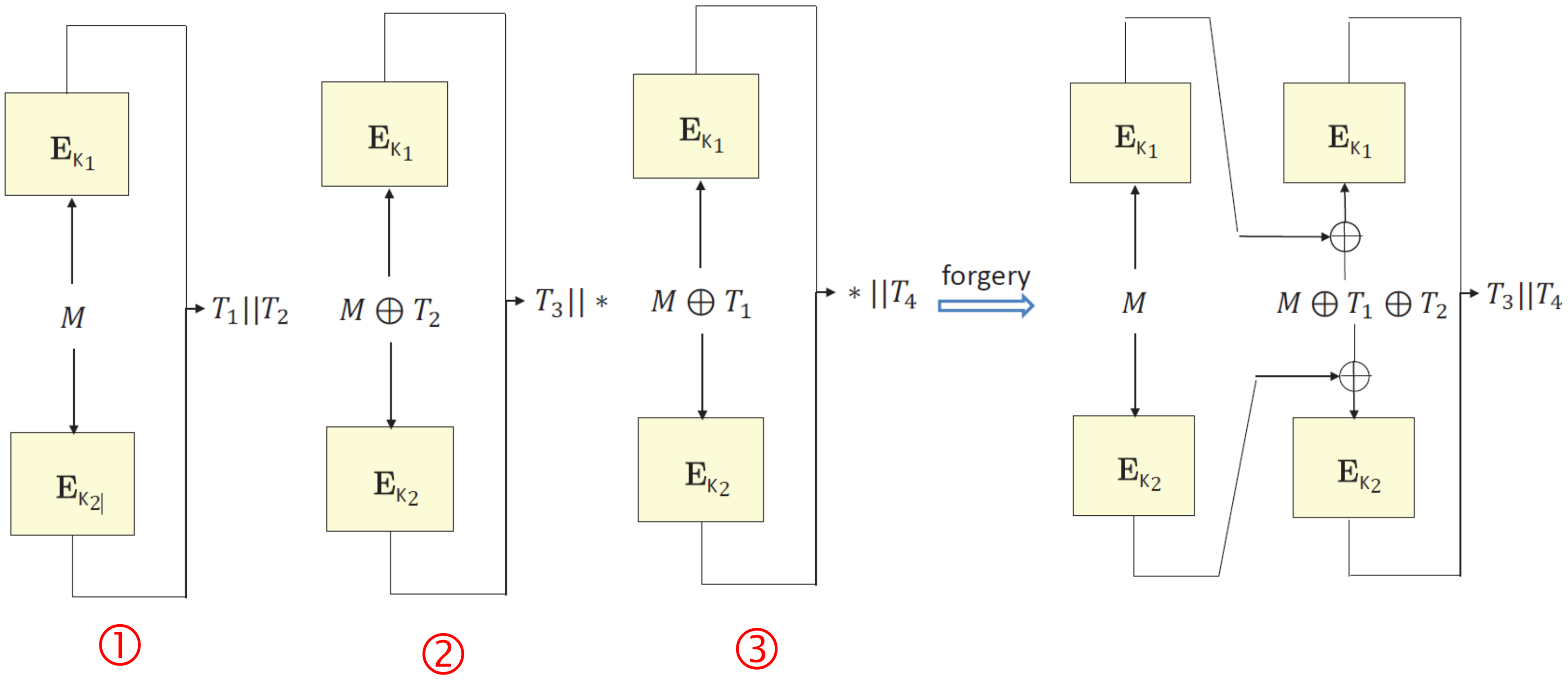
- Concatenating to lift the security level: ISO/IEC 9797-1:2011 Annex C

if a MAC algorithm with a higher security level is needed, it is recommended to perform two MAC calculations with independent keys and concatenate the results (rather than XORing them).



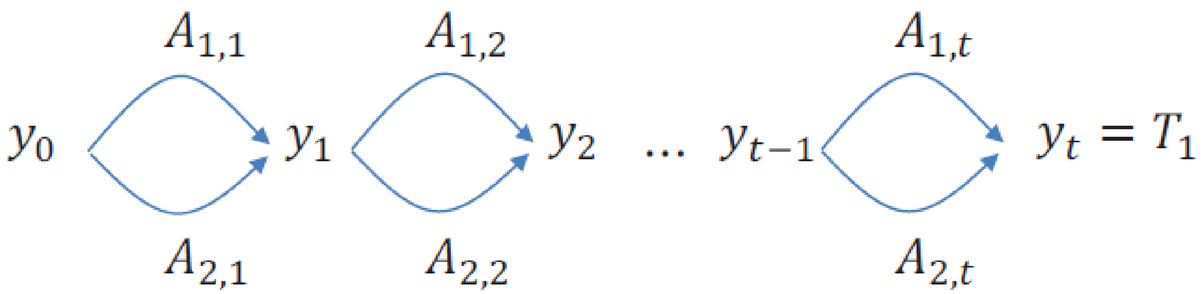
Attack on the concatenation of two MAC1

- $\mathrm{MAC1}_{K_1}(M) \parallel \mathrm{MAC1}_{K_2}(M)$ with pad2 ($M||10^*$)
- 3 queries, succeeds with probability 1

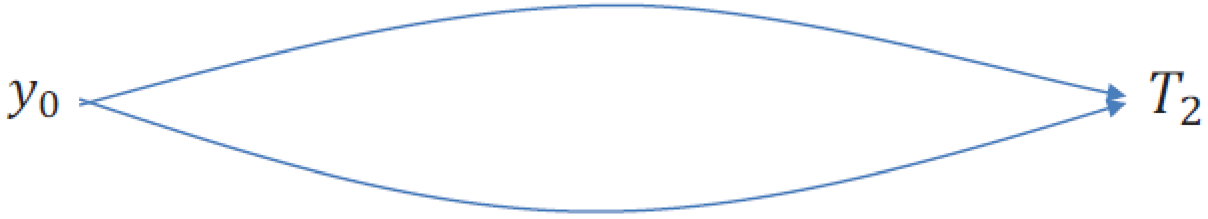


Attack on the concatenation of any two MACs

- $\mathrm{MAC}i_{K_1}(M) \,\|\, \mathrm{MAC}j_{K_2}(M)$ with pad2, pad3, and pad4
- Forgery attack
  - left collision



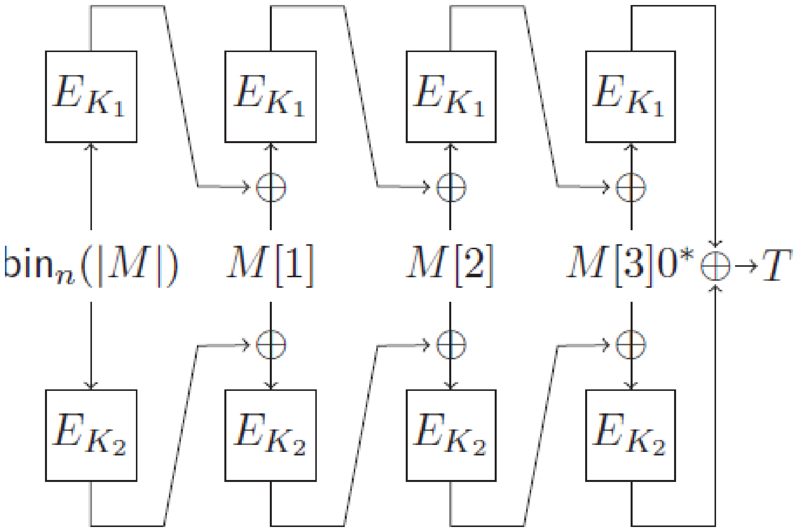
  - right collision



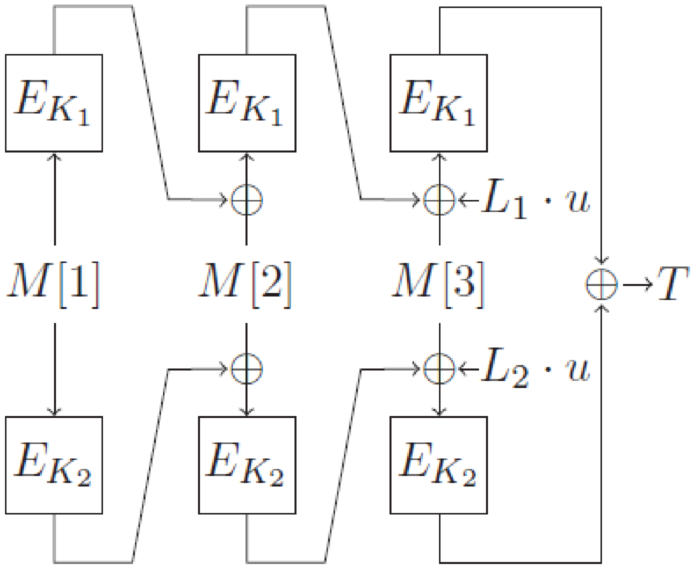
  - full collision: $O(n2^{\frac{n}{2}})$

Patches

- XOR of two MACs can improve the security



XOR of two MAC1



XOR of two MAC5

- Beyond-birthday bound $O(2^{\frac{2n}{3}})$

ISO’s reaction

- ISO/IEC 9797-1:2011/Amd 1:2023
- Remove the concatenation suggestion

INTERNATIONAL STANDARD

ISO/IEC 9797-1

Second edition
2011-03-01

AMENDMENT 1
2023-08

Information technology — Security techniques — Message Authentication Codes (MACs) —

Part 1:
Mechanisms using a block cipher

AMENDMENT 1

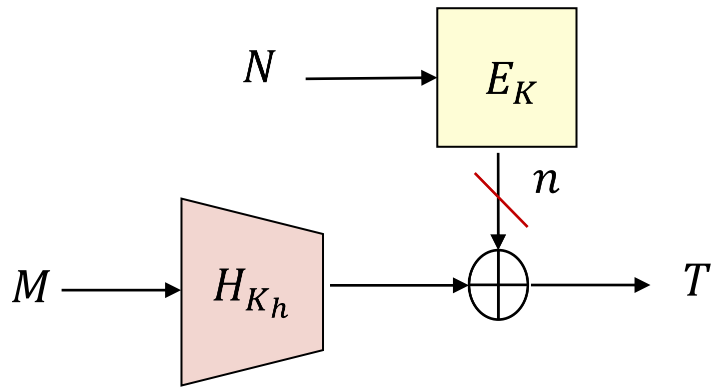
Technologies de l'information — Techniques de sécurité — Codes d'authentification de message (MAC) —

Partie 1: Mécanismes utilisant un chiffrement par blocs

AMENDEMENT 1

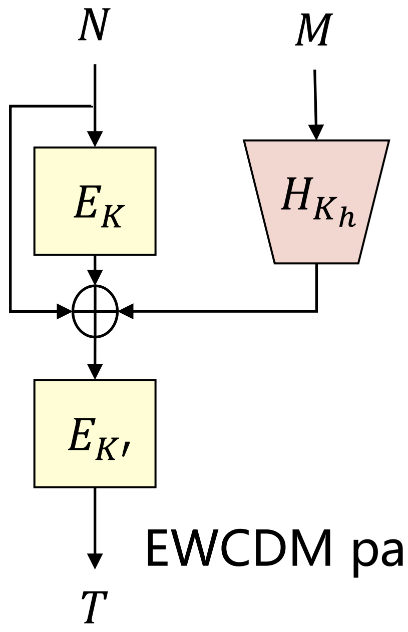
Universal-hash-based MAC

- Wegman-Carter paradigm
  - GMAC/GCM
  - Poly1305-AES



- Security capped at the birthday bound $O(2^{\frac{n}{2}})$

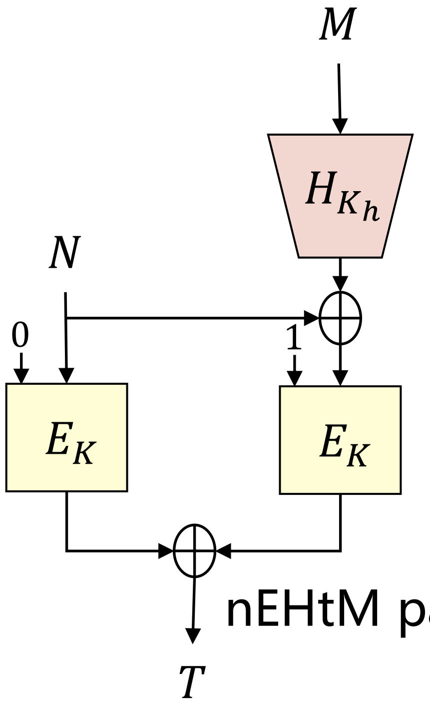
Recent trend: beyond-birthday-bound (BBB) MACs



EWCDM paradigm [CS16]

- nPolyMAC [DDNY18]
- PDM*MAC [CNTY20]
- 1k-PDM*MAC [CNTY20]

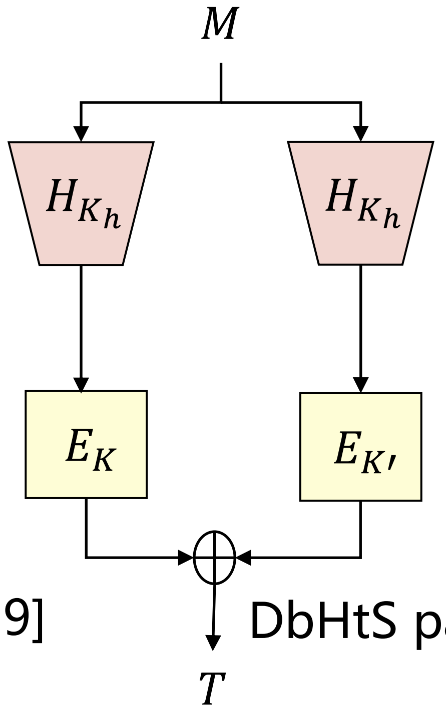
security: $\geq 2^{\frac{2n}{3}}$ queries



nEHtM paradigm [DNT19]

- CWC+ [DNT19]
- nEHtM_p^+ [CDN22]

security: $2^{\frac{2n}{3}}$ queries

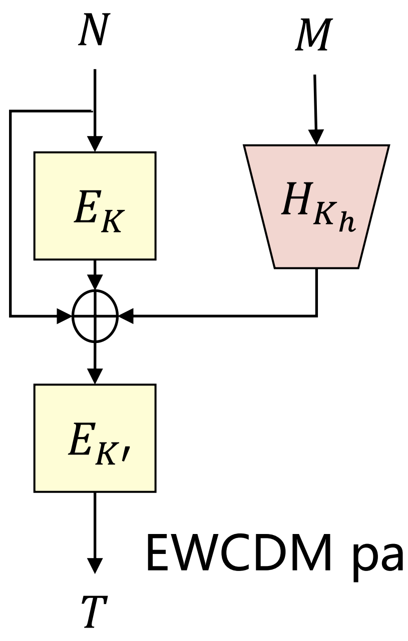


DbHtS paradigm [DDNP18]

- PolyMAC [DDNP18,KLL20]
- 2k-PolyMAC [DDNP18, DDNT23]

security: $2^{\frac{3n}{4}}$ queries

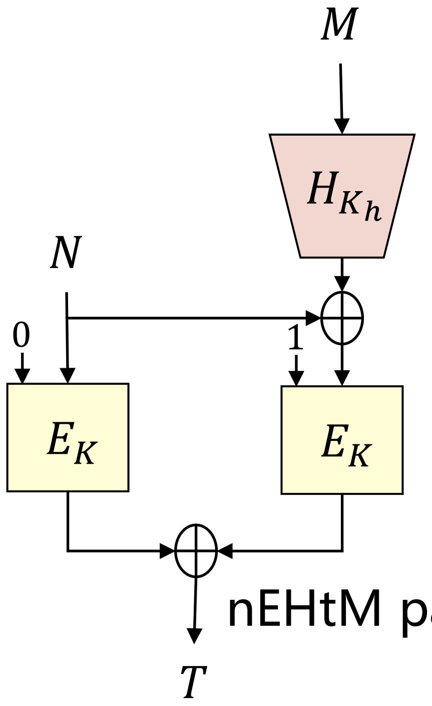
Recent trend: beyond-birthday-bound (BBB) MACs



EWCDM paradigm [CS16]

- nPolyMAC [DDNY18]
- PDM*MAC [CNTY20]
- 1k-PDM*MAC [CNTY20]

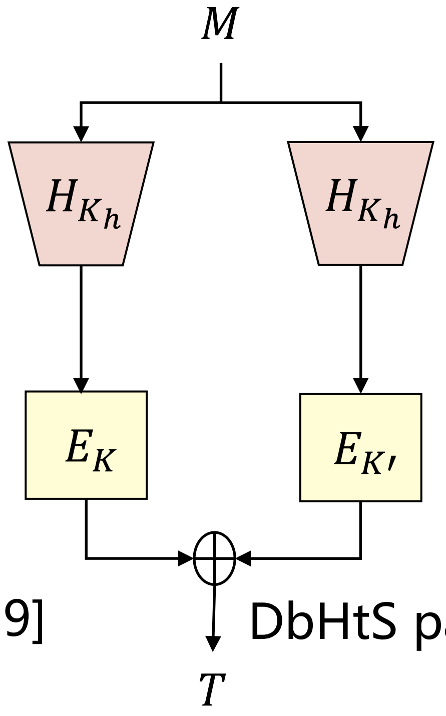
security: $\geq 2^{\frac{2n}{3}}$ queries



nEHtM paradigm [DNT19]

- CWC+ [DNT19]
- nEHtM_p^+ [CDN22]

security: $2^{\frac{2n}{3}}$ queries



DbHtS paradigm [DDNP18]

- PolyMAC [DDNP18,KLL20]
- 2k-PolyMAC [DDNP18, DDNT23]

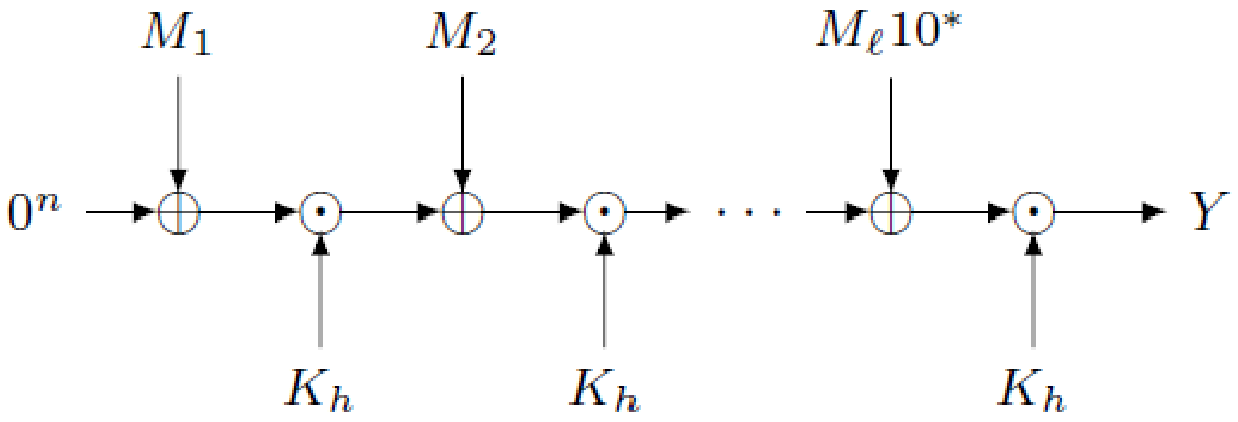
security: $2^{\frac{3n}{4}}$ queries

Using two queries can forge a tag successfully against these MACs
This attack is due to the vulnerability in the polynomial hash Poly

The vulnerability in the polynomial hash Poly

Xiamen University

- These MACs use a polynomial hash Poly as the underlying hash function
  - $\mathrm{Poly}_{K_h}(M) = M_1 \cdot K_h^{\ell} \oplus M_2 \cdot K_h^{\ell-1} \oplus \cdots \oplus M_\ell 10^* \cdot K_h$

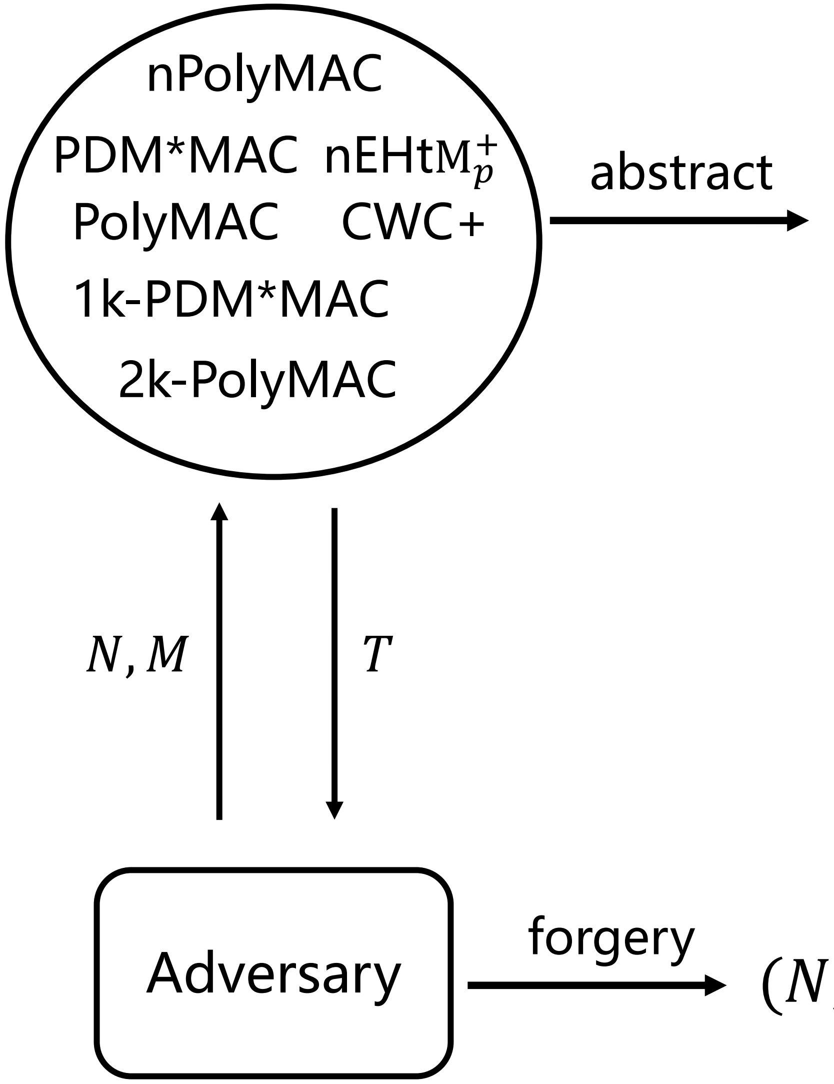


- It was proved that Poly was almost-xor-universal (AXU), yet it is not:

$$\mathrm{Poly}_{K_h}(M) = \mathrm{Poly}_{K_h}((0^n)_a || M)$$

  - 0^n is a fixed point for finite field multiplication
  - the length-dependent term $M_i \cdot K_h^{\ell+1-i}$ will be cancelled out if $M_i = 0^n$

Principle of this attack



$$\mathrm{MAC}(N, M) = \mathrm{F}(N, \mathrm{Poly}(M))$$

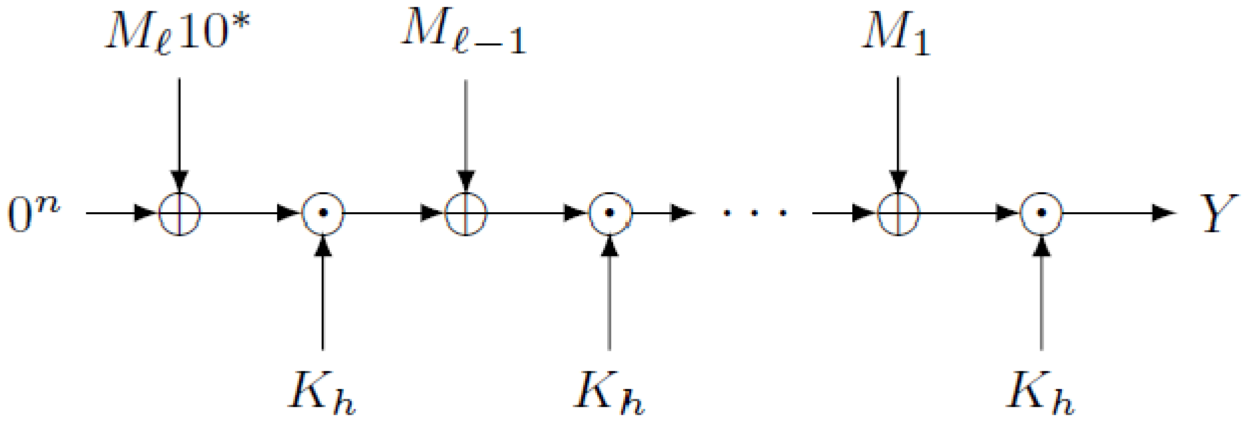
$$(N, (0^n)_a || M, T) \quad T = \mathrm{F}(N, \mathrm{Poly}((0^n)_a || M)) = \mathrm{F}\big(N, \mathrm{Poly}(M)\big)$$

Attack complexity

- requires 2 queries
- nonce-respecting
- succeed probability 1

Patches：two new polynomial hash PolyX and GHASHX

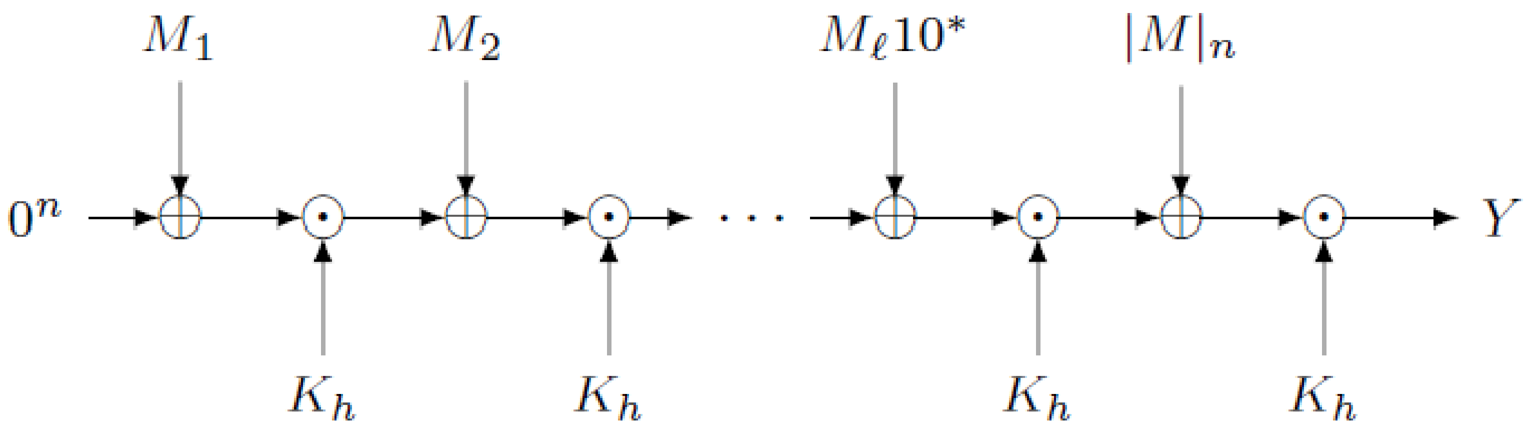
- PolyX: reversing the order of a message in the polynomial of Poly
  - $\mathrm{PolyX}_{K_h}(M) = M_1 \cdot K_h \oplus M_2 \cdot K_h^2 \oplus \cdots \oplus M_\ell 10^* \cdot K_h^\ell$



- The length-dependent term $M_\ell 10^* \cdot K_h^\ell$ will never be zeroed out

Patches: two new polynomial hash PolyX and GHASHX

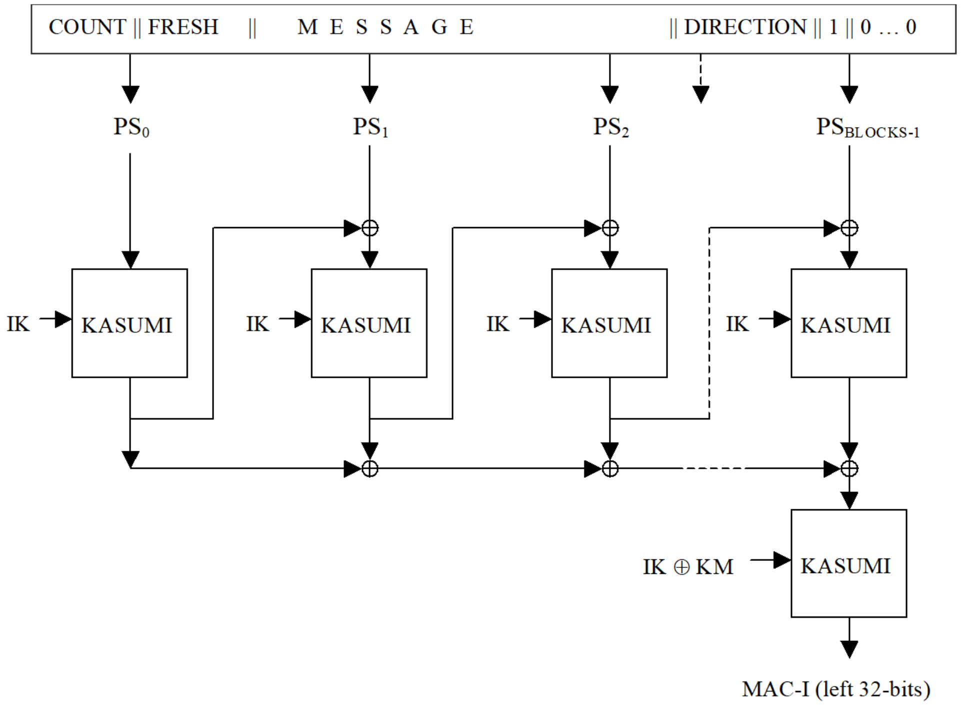
- GHASHX: replacing the 0^* padding with 10^* in GHASH
  - GHASH is not regular as $\mathrm{GHASH}(\varepsilon) = 0^n$, causing a forgery attack against nPolyMAC
  - $\mathrm{GHASHX}_{K_h}(M) = M_1 \cdot K_h^{\ell+1} \oplus M_2 \cdot K_h^{\ell} \oplus \cdots \oplus M_\ell 10^* \cdot K_h^2 \oplus |M|_n \cdot K_h$



- Replacing Poly with either PolyX or GAHSHX can restore the beyond-birthday-bound security of these MACs

f9 algorithm

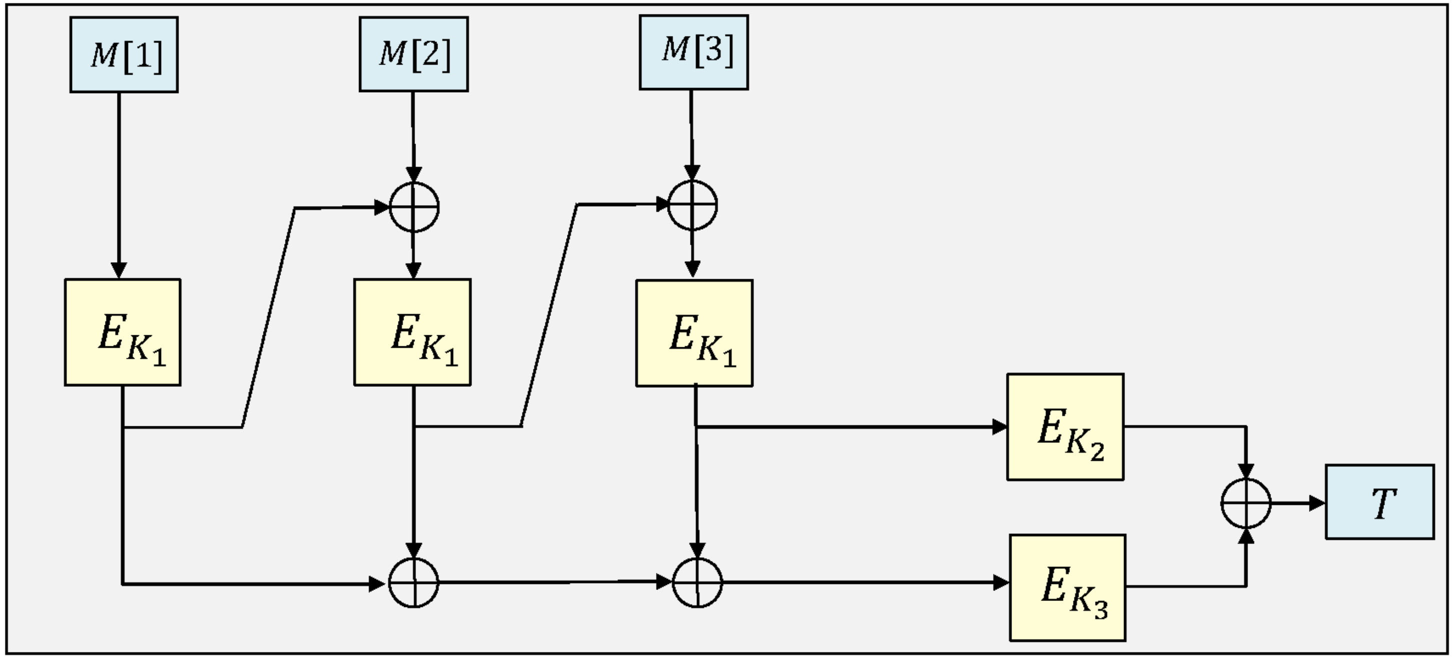
- Used as 2G/3G integrity algorithm



- Capped at the birthday-bound security $O(2^{\frac{n}{2}})$

Enhanced variant of f9：3kf9

- 3kf9 [ZWSW12]

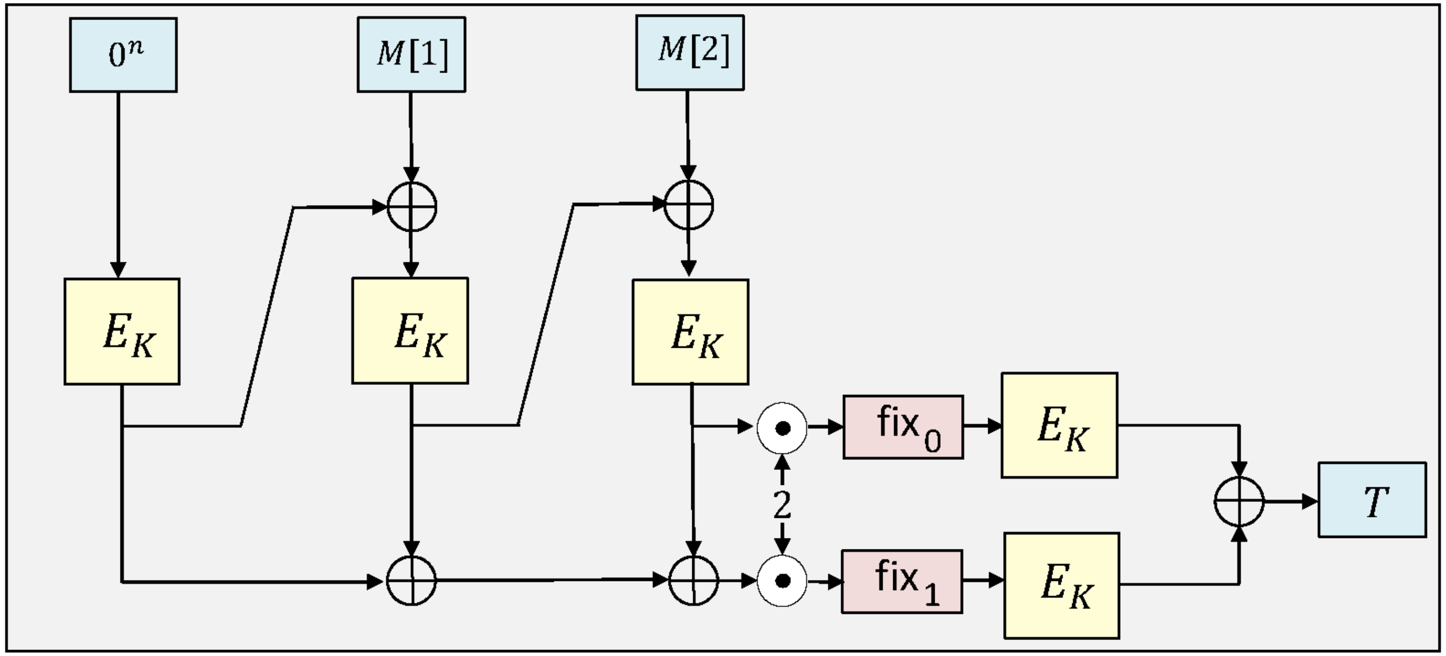


- can achieve beyond-birthday-bound security $O(2^{\frac{2n}{3}})$
- requires 3 keys

Key-reduced variants of 3kf9: 1kf9

Xiamen University

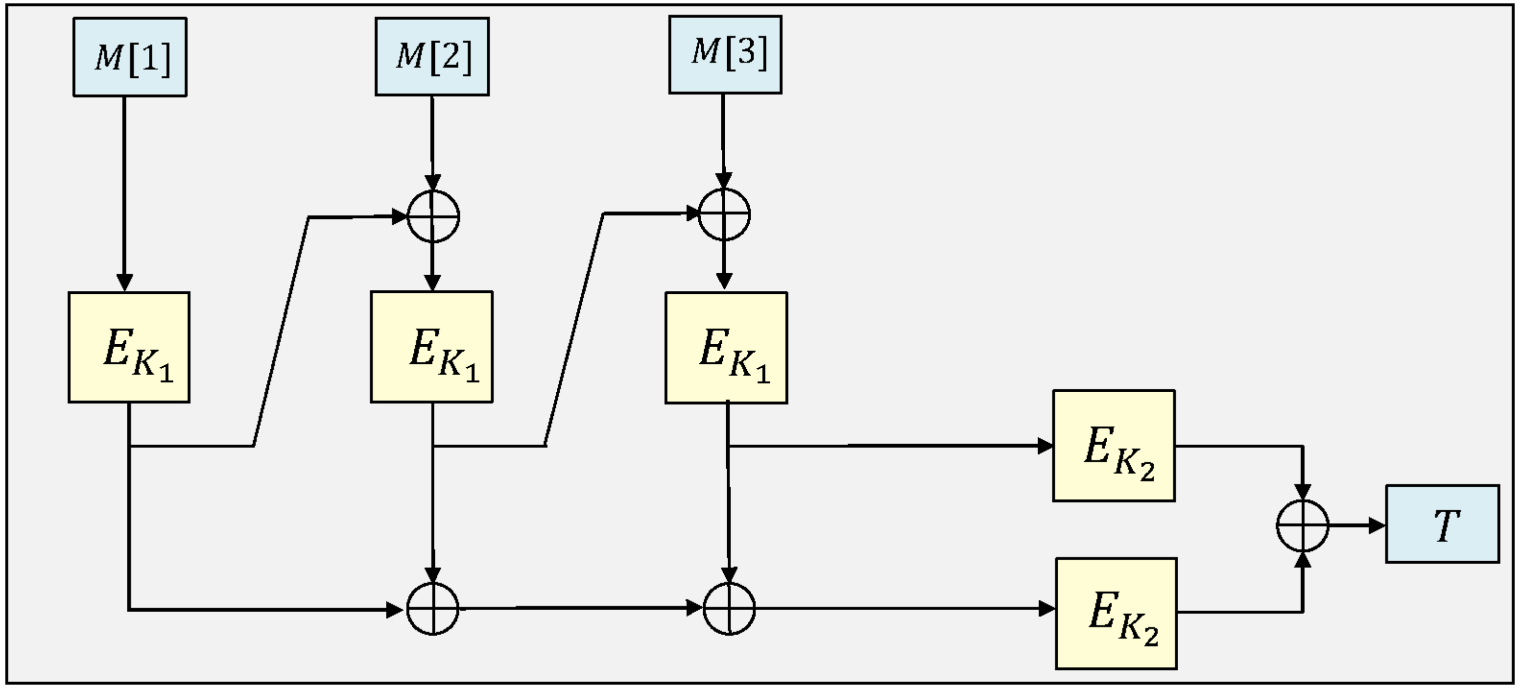
- 1kf9 [DDN+15]:



- Birthday-bound attack by exploiting fix functions [LNS18]:
  - find $x||0^n$ and $y||d$ such that $E_K(E_K(0^n) \oplus x) \oplus E_K(E_K(0^n) \oplus y) = d$ where d is the inverse of 2

Key-reduced variants of 3kf9: 2kf9

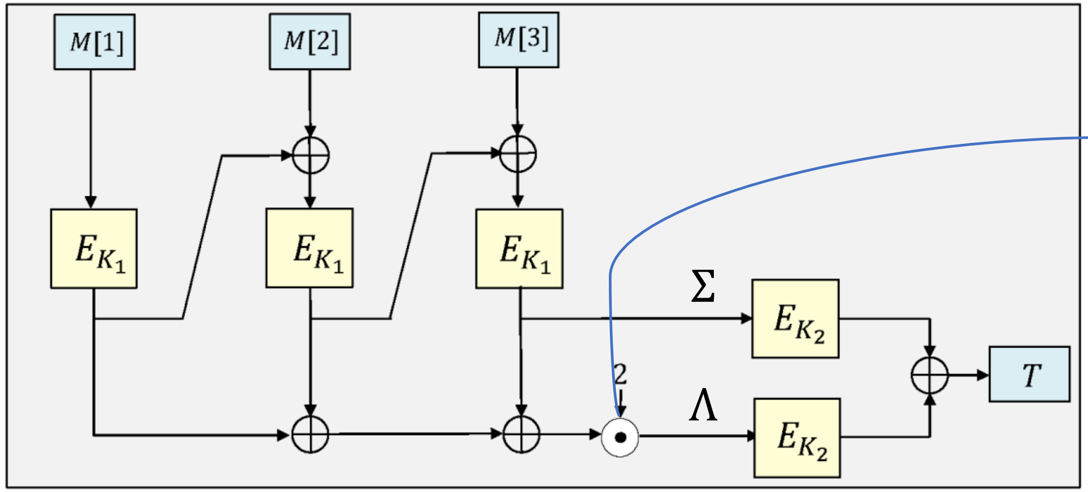
- 2kf9 [DDN19]:



- One-query attack: $(M, 0^n)$ is a valid forgery for any $|M|=n$ [SWGW21]

Key-reduced variants of 3kf9: n2kf9 and n1kf9

- A simple doubling near the end: n2kf9



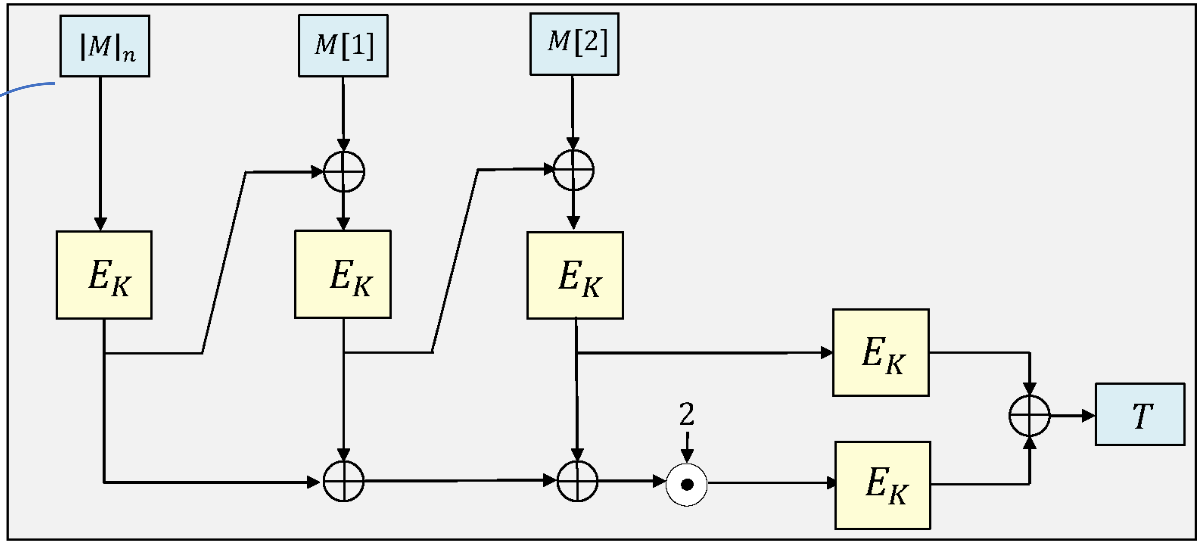
≈ one-bit shift and one conditional XOR with some constant

- Beyond-birthday bound security $O(2^{\frac{2n}{3}})$

Key-reduced variants of 3kf9: n2kf9 and n1kf9

- one-key MAC: n1kf9

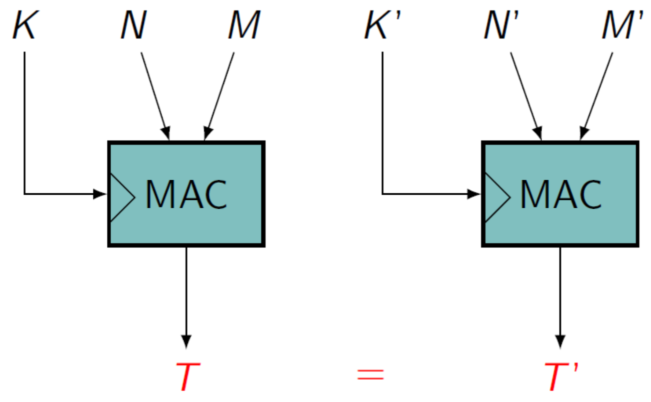
to be prefix-free



- Beyond-birthday bound security $O(2^{\frac{2n}{3}})$

Committing secure MACs

- Definition



Notion	Requirement
CMT_k	$K \neq K'$
CMT	$(K, N, M) \neq (K', N', M')$

- Summary of standard MACs

Scheme	CMT_k	CMT	CDY
CBC-type MACs	no	no	no
HMAC with variable-length keys	no	no	?
Badger	no	no	no
Poly1305-AES	no	no	no
GMAC	no	no	no
LightMAC	no	no	no
Chaskey	no	no	no
CBC-MAC-C1 [this work]	yes	no	yes
CMAC-C1 [this work]	yes	no	yes
HMAC with fixed-length keys	yes	yes	yes

Content

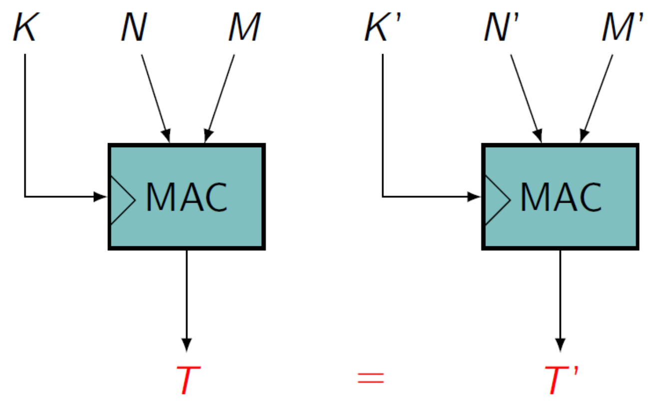
1 Background

2 Analysis and Enhances

3 **Discussion**

Discussion and future work

- Continual security analyses of standardized algorithms
- The usage of MAC is far beyond merely ensuring authenticity
  - used as key derivation functions (NIST SP 800-108r1)
  - used in PAKE recommended by CFRG
  - used in timed efficient stream loss-tolerant authentication (RFC 4082)
- Design context-committing secure MACs



notion	requirement
CMT	$(K, N, M) \neq (K', N', M')$

Thanks for your attention