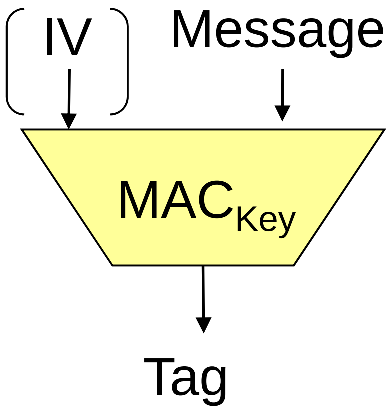
How to Thwart Birthday Attacks against MACs via Small Randomness

Kazuhiko Minematsu (NEC Corporation)

Fast Software Encryption 2010, Seoul, Korea

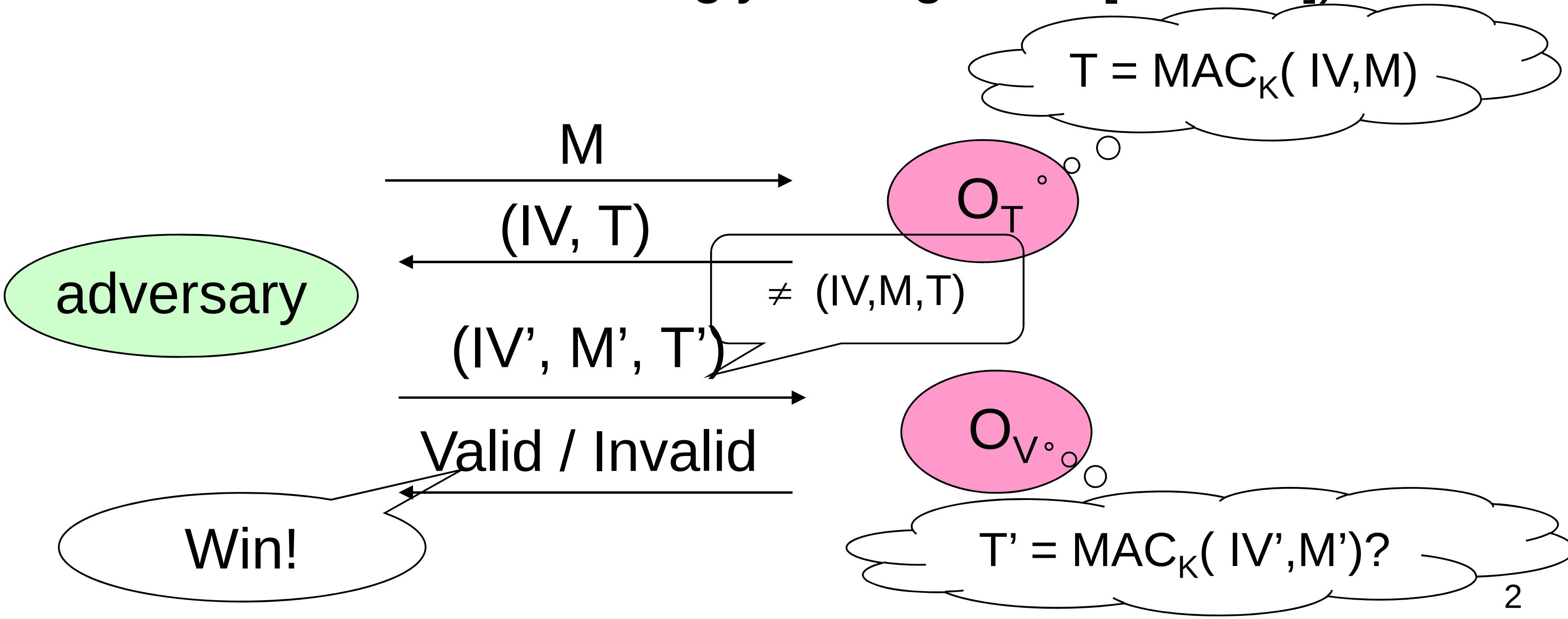
Introduction

- Message Authentication Code (MAC)
  - Use (Key, Message) to generate a fixed-length tag
  - An auxiliary input, initial vector (IV) may exist
- Three classes
  - No IV -> deterministic MAC
  - IV is random -> randomized MAC
  - IV is nonce -> stateful MAC



Goal of adversary

- Two oracles :
  - Tagging oracle (O_T) returns a tag (and IV) for a queried message
  - Verification oracle (O_V) returns a verification result for a queried transcript
- Goal is to produce a forgery (a valid transcript made w/o querying it to O_T)
- If this is hard, MAC is strongly unforgeable [BGK99])



Security measure

- Let adversary have q tagging queries and q_v verf. queries
  - with messages of length at most ℓ (in n-bit blocks)
- Forgery probability (FP) is the maximum prob. of receiving “Valid” from O_V , denoted as

$$\mathrm{FP}_{\mathrm{MAC}}(q, q_v, \ell)$$

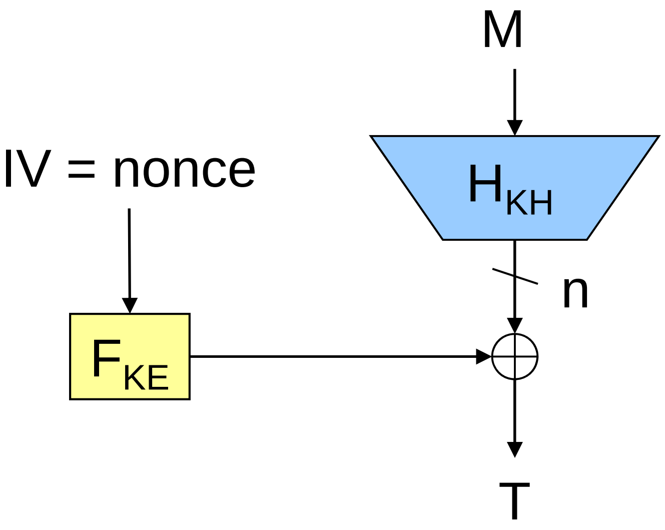
Typical IV-based MAC : Hash-then-Mask (HtM)

- $T = H_{KH}(M)+F_{KE}(IV)$
- H_{KH} is ε-almost XOR universal (ε-AXU)

$$\max_{M_1 \neq M_2} \Pr[H_{\mathrm{KH}}(M_1) \oplus H_{\mathrm{KH}}(M_2) = y] \leq \varepsilon$$

  - possibly defined w/ input-block length (ε (ℓ)-AXU)
- Stateful HtM is highly secure :

$$\mathrm{FP}_{\texttt{Stateful HtM}}(q, q_v, \ell) \leq \varepsilon(\ell) \cdot q_v$$



Problem of being stateful

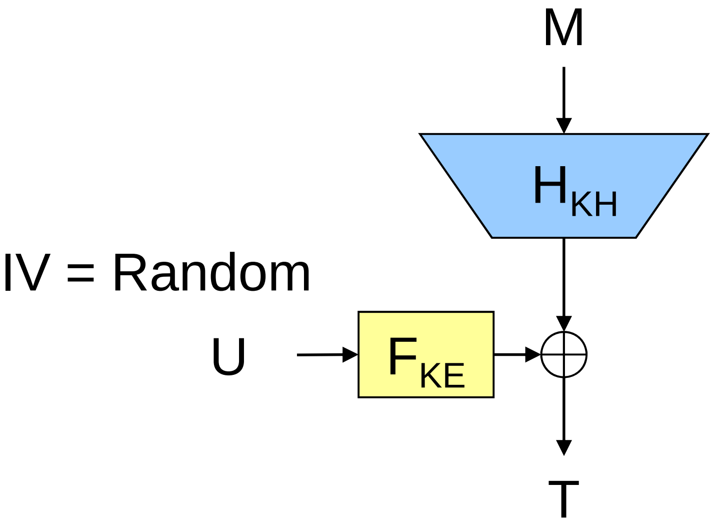
- Keeping state is difficult if (e.g.)
  - Same key is used by many distant devices
  - Key is in ROM and other non-volatile memory is not available

A natural substitute: use randomness

- What will happen if IV is an n-bit random value?
- Then, the security degrades to

$$\mathrm{FP}_{\text{randomized HtM}}(q, q_v, \ell) \leq \frac{q^2}{2^{n+1}} + \varepsilon(\ell) \cdot q_v$$

- as IVs may collide, which leaks the sum of hash values (total break in general)
- That is, we have a birthday attack w/ q = $2^{n/2}$



T = H(M)+F(U)
T’ = H(M’)+F(U’)
if U=U’ then T’+T’ = H(M’)+ H(M’)

Our goal

- Improve $O(q^2/2^n)$ term in the FP bound of n-bit-IV randomized HtM
  - so-called “beyond-birthday-bound-security”
- ...without expanding randomness! (longer IV is practically undesirable; comm. overhead, more random source, etc.)

Previous solutions

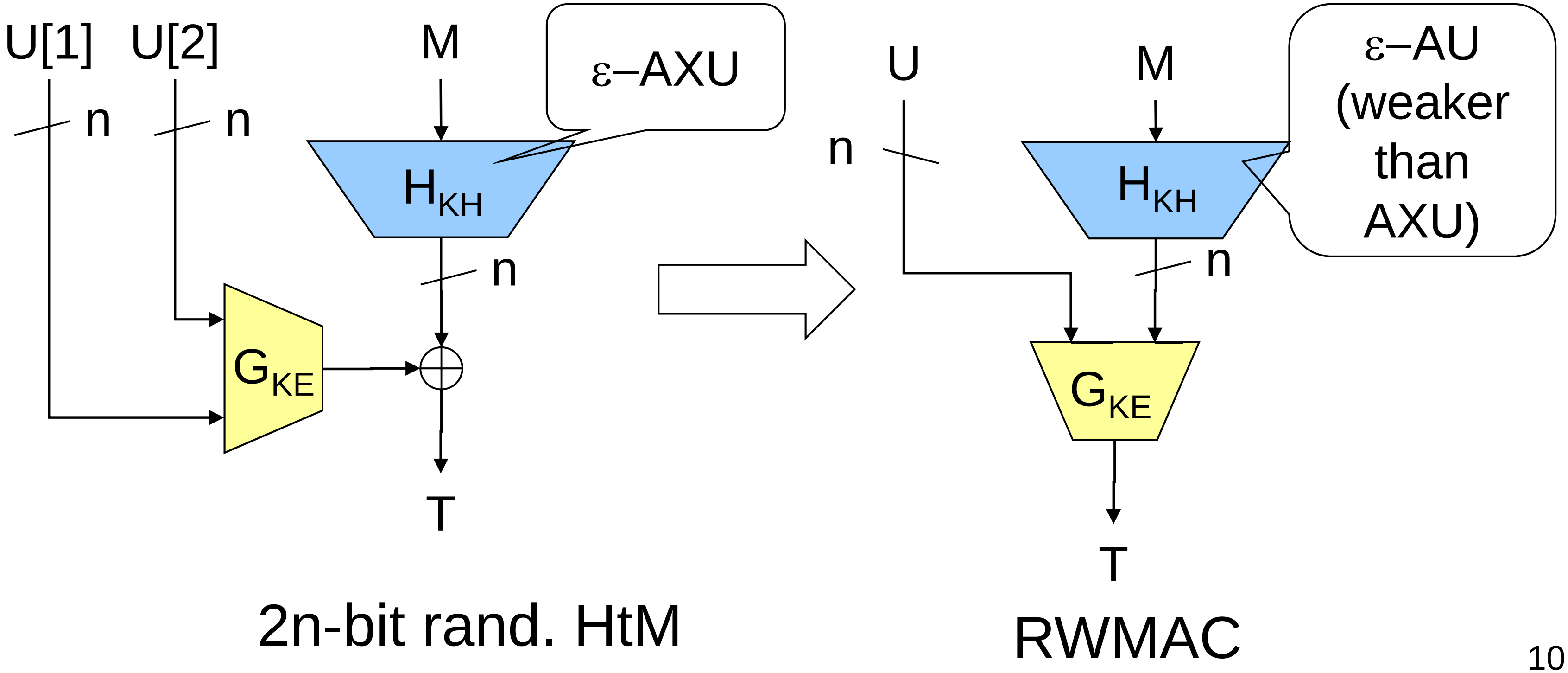
- Long-IV solutions (outside our scope)
  - Naïve 2n-bit rand. HtM
    - ✓ Use 2n-bit randomness, 2n-bit-input PRF
  - MACRX [BGK99]
    - ✓ Use 3n-bit randomness, n-bit-input PRF
- n-bit-IV solution (our scope)
  - RMAC/FRMAC [JJV02] [JL04]
    - ✓ Use n-bit randomness, n-bit blockcipher (nice)
    - ✓ BUT proof needs the ideal-cipher model (dangerous)

Our contributions

- Two simple proposals
- RWMAC
  - Use n-bit randomness and 2n-bit-input PRF
- Enhanced Hash-then-Mask (Main contribution)
  - Use n-bit randomness and n-bit-input PRF
  - Very efficient : one additional PRF call to n-bit rand. HtM
- Blockcipher modes based on EHtM
  - Provably secure if blockcipher is a PRP (standard assumption)
  - Good alternatives to RMAC

First step : modify 2n-bit rand. HtM

- Encrypt $H_{KH}(M)$ and U together with 2n-bit-input PRF, G_{KE}
  - using ε-AU hash (coll. prob. is at most ε)
- Result is RWMAC, a rand. version of stateful MAC called WMAC [BC09]



Why beyond birthday bound ?

- Unless U and S=H_{KH}(M) collide together, tags are perfectly random (secure)
  - (U,S)-collision prob. for two distinct messages is ε /2^n
    - ✓ Note: for the same messages U-collision does not help
- Hence we obtain the security bound:

$$\mathrm{FP}_{\mathrm{RWMAC}[H,G]}(q, q_v, \ell) = q^2 \frac{\varepsilon(\ell)}{2^{n+1}} + q_v \left(2(n-1)\varepsilon(\ell) + \frac{1}{2^{\pi}} \right) .$$

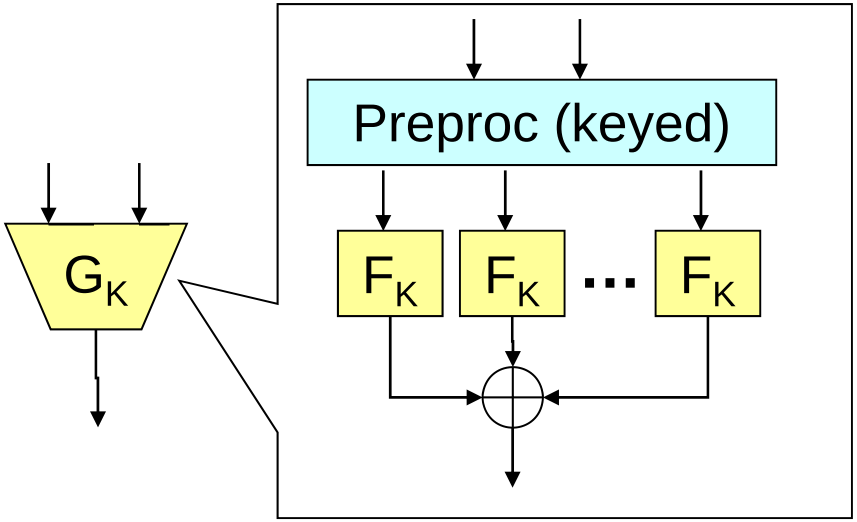
(w/ final tag truncation to π bits)

- If π = n and ε 7 2^{-n}, it is about $q^2/2^{2n}$ + q_v /2^n

(similar observation given by WMAC paper)

Next step: remove 2n-bit-input PRF

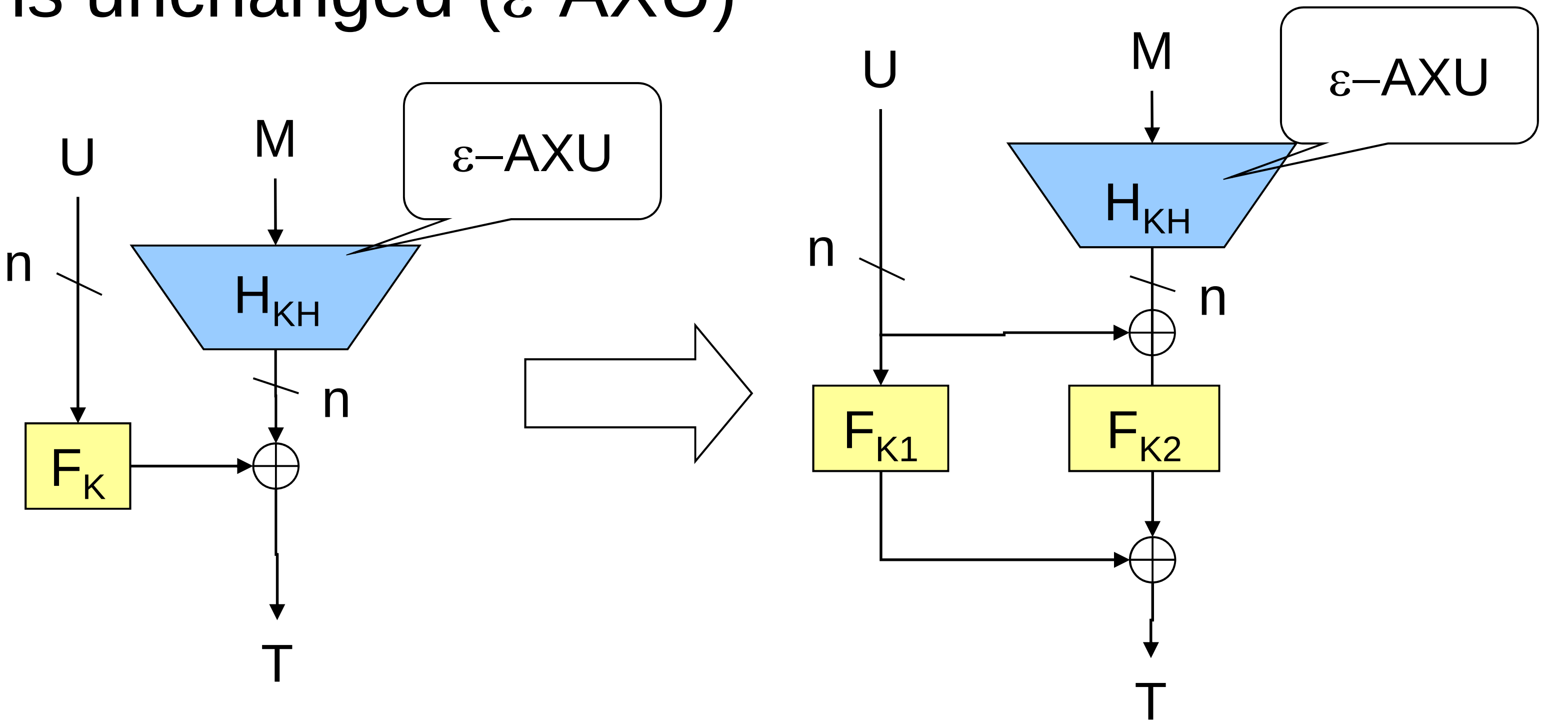
- Naïve approach : RWMAC + some PRF domain extension w/ *beyond-birthday-bound-security*
  - known scheme of Maurer [M02] is not that efficient
- Idea : G's inputs of RWMAC are not arbitrarily chosen, thus full-fledged PRF might not be needed
- … but how?



Maurer's domain extension scheme

Enhanced Hash-then-Mask (EHtM)

- We insert one additional (independently-keyed) n-bit PRF before masking w/ a simple preproc. (x,y)->(x,x+y)
- H is unchanged (ε-AXU)



rand. HtM

EHtM

Security bound of EHtM

- The bound is :

$$\mathrm{FP}_{\mathrm{EHtM}[H,F_1,F_2]}(q,q_v,\ell) \le \frac{q^3}{6}\left(\frac{\epsilon(\ell)}{2^n}+\frac{1}{2^{3n}}\right)+q_v\left(4\epsilon(\ell)+\frac{1}{2^{\pi}}\right)$$

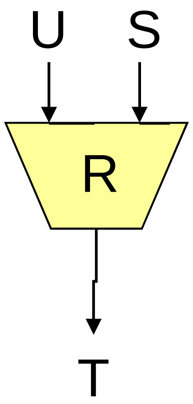
(w/ final tag truncation to π bits)

- If π = n and ε 7 2^{-n}, the bound is about $q^3/2^{2n} + q_v/2^n$
  - not as good as RWMAC bound, but still an improvement over HtM's bound $q^2/2^n + q_v/2^n$

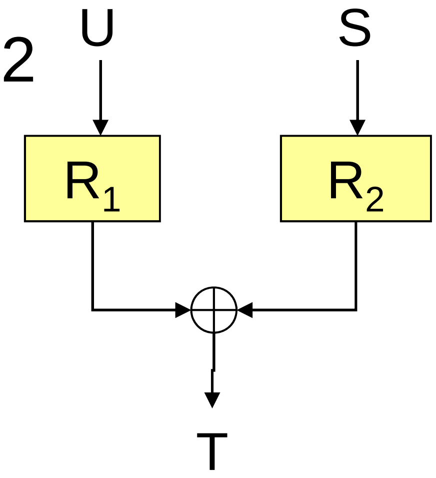
Proof idea

- Compare the finalizations of RWMAC and EHtM
  - If BAD = [$U_i=U_j \neq U_k$, $S_i \neq S_j = S_k$] for some distinct (i,j,k) occurs, the difference between two cases is detectable,
  - as output of Case2 for input (U_k,S_i) is predictable $(T_i+T_j+T_k)$, while Case1's output for (U_k,S_i) is random

Case1



Case2

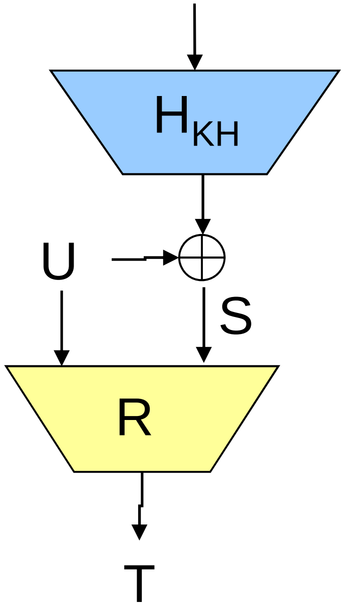


Note: similar observation was seen in MACRX and Maurer's PRF domain extension

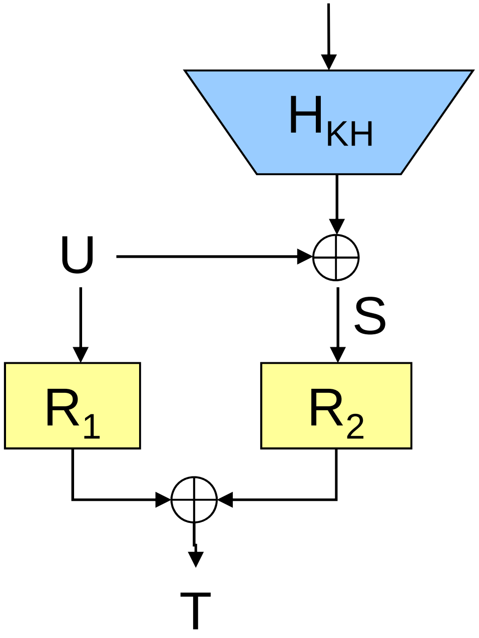
Proof idea (contd.)

- Add ε-AXU hash function to both cases
  - Now BAD occurs at most prob. ε /2^n for any (i,j,k), (both under EHtM and RWMAC) thus the difference is detectable w/ probability O(q^3 ε/ 2^n)
  - If BAD does not occur FP of EHtM is the same as that of mod. RWMAC, which is easy to derive (the same as RWMAC)
- Details are more complicated ...

modified RWMAC

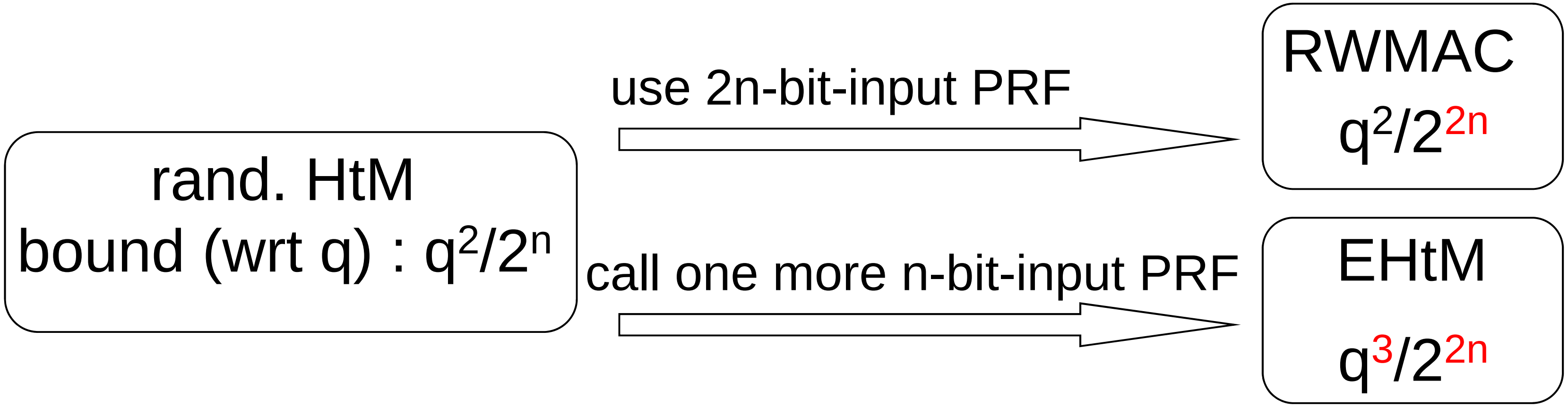


EHtM



Quick summary

- Roughly, the result can be summarized as;



rand. HtM
bound (wrt q) : $q^2/2^n$

use 2n-bit-input PRF

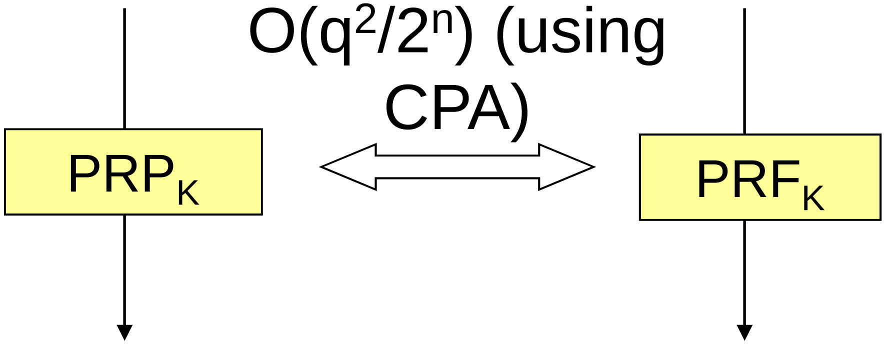
RWMAC
$q^2/2^{2n}$

call one more n-bit-input PRF

EHtM
$q^3/2^{2n}$

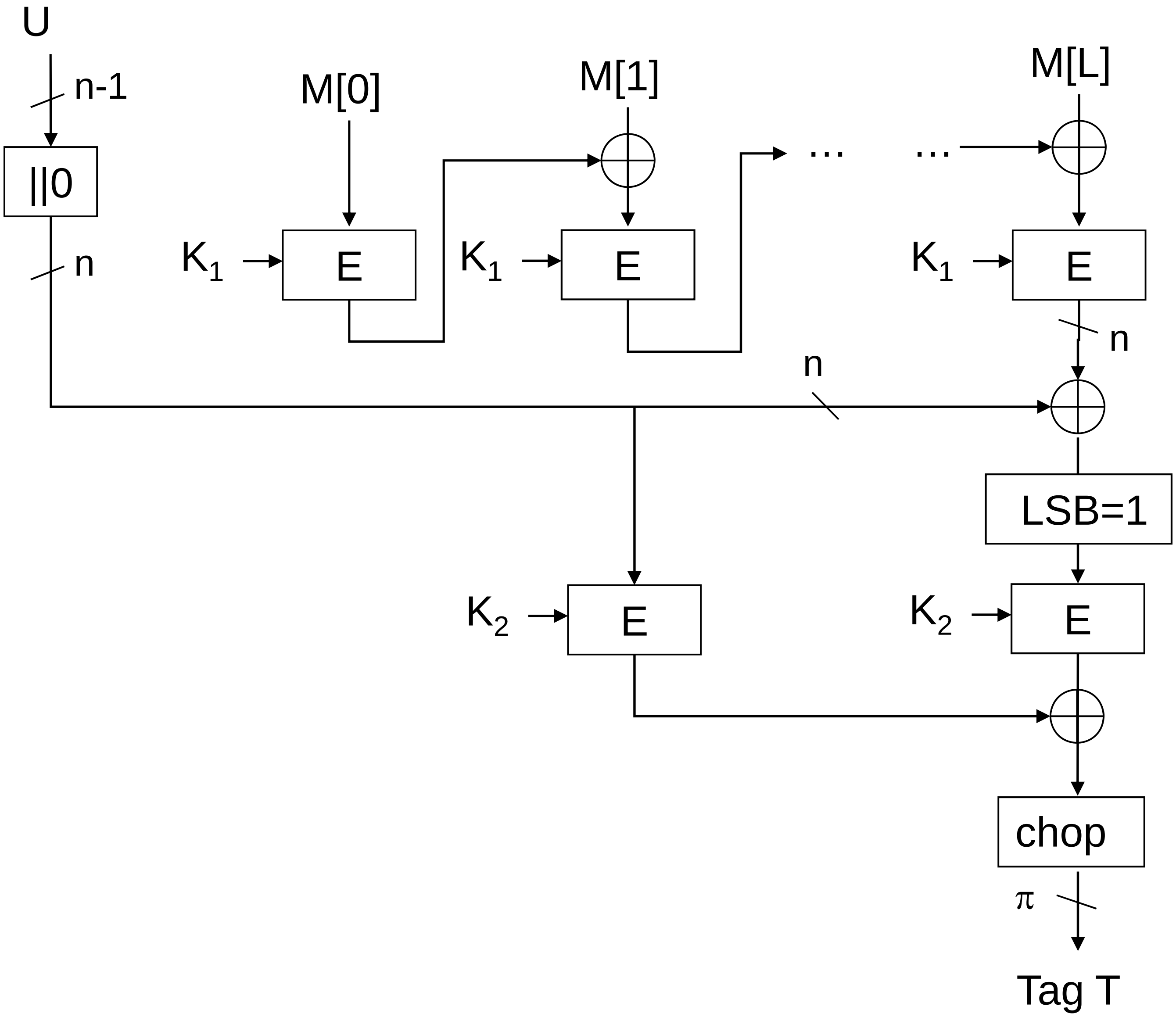
Blockcipher modes

- Next, we try to instantiate EHtM w/ a blockcipher (which is assumed to be a PRP)
- PRP-based finalizations needed
- Main obstacle: PRP-PRF switching lemma will bring $O(q^2/2^n)$-security degradation

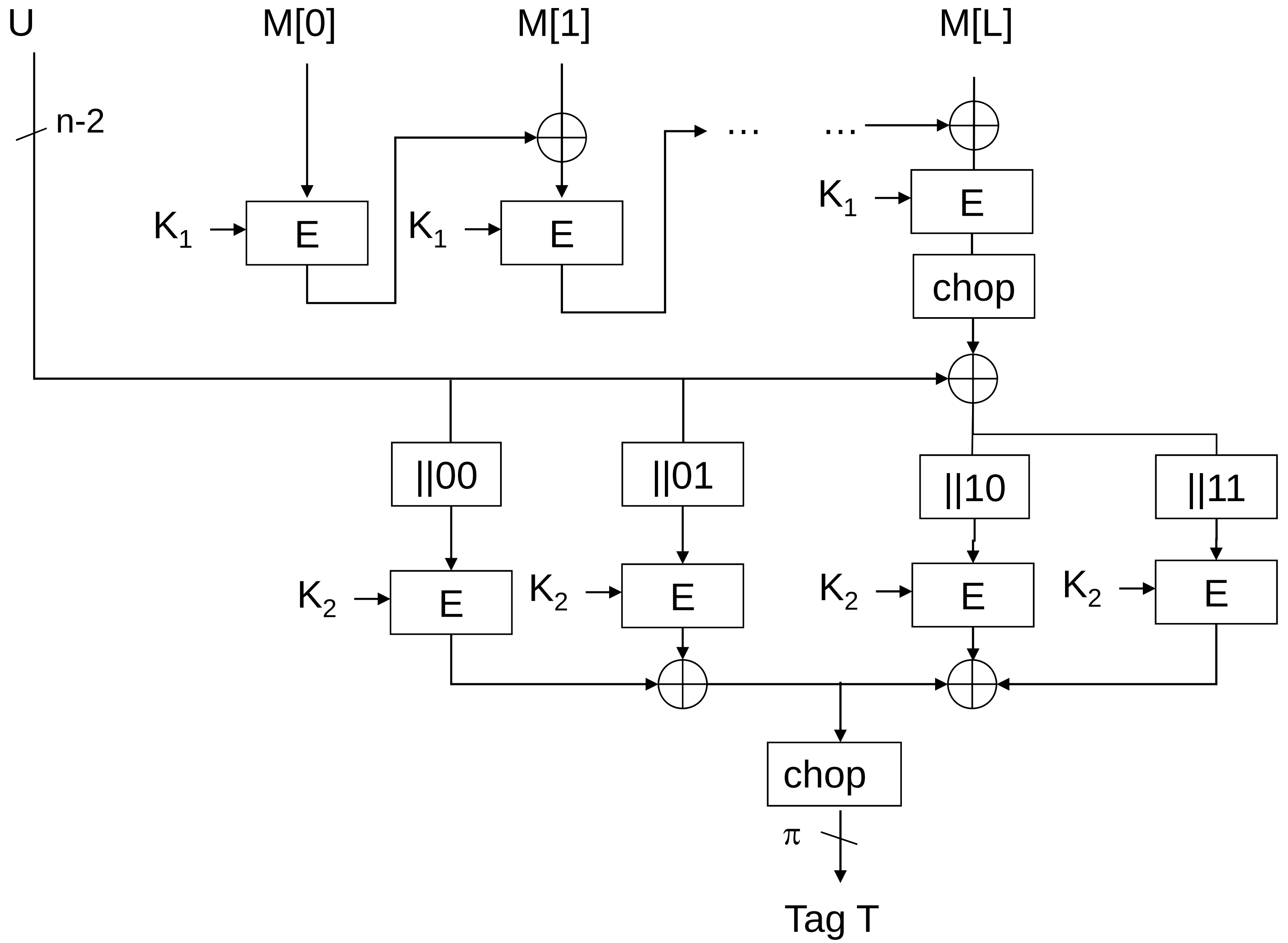


distinguishable w/ advantage $O(q^2/2^n)$ (using CPA)

A CBC-based Mode: MAC-R1



An Alternative Mode: MAC-R2

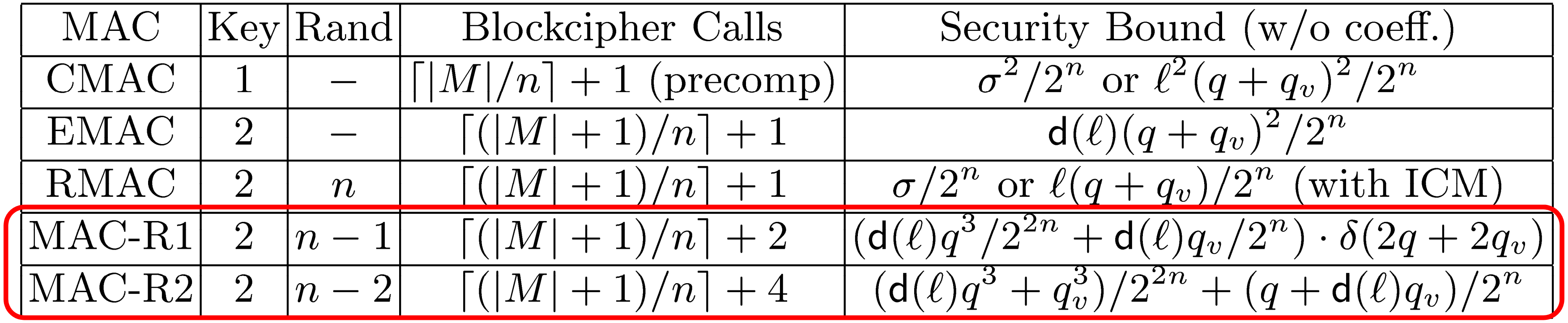


Proofs of MAC-R1 and R2

- Just a combination of previous results
  - CBC-MAC collision prob. [BPR05] and differential prob. [MM07]
  - For R1, Bernstein's lemma [B05] instead of switching lemma
    - ✓ gives an improved *unpredictability* (but not indistinguishability) ; only applicable to FP evaluation
  - For R2, Lucks's TWIN construction [L00]
    - ✓ taking the sum of two PRP distinct inputs yield a PRF w/ beyond-birthday-bound-security

Comparison of MAC modes

- VERY roughly, MAC-R2 bound is $(q+q_v)^3/2^{2n}$
- MAC-R1 bound is something worse (difficult to see from the table)



MAC	Key	Rand	Blockcipher Calls	Security Bound (w/o coeff.)
CMAC	1	–	$\lceil \lvert M\rvert/n \rceil + 1$ (precomp)	$\sigma^2/2^n$ or $\ell^2(q+q_v)^2/2^n$
EMAC	2	–	$\lceil (\lvert M\rvert+1)/n \rceil + 1$	$\mathsf{d}(\ell)(q+q_v)^2/2^n$
RMAC	2	n	$\lceil (\lvert M\rvert+1)/n \rceil + 1$	$\sigma/2^n$ or $\ell(q+q_v)/2^n$ (with ICM)
MAC-R1	2	$n-1$	$\lceil (\lvert M\rvert+1)/n \rceil + 2$	$(\mathsf{d}(\ell)q^3/2^{2n} + \mathsf{d}(\ell)q_v/2^n)\cdot\delta(2q+2q_v)$
MAC-R2	2	$n-2$	$\lceil (\lvert M\rvert+1)/n \rceil + 4$	$(\mathsf{d}(\ell)q^3 + q_v^3)/2^{2n} + (q+\mathsf{d}(\ell)q_v)/2^n$

σ = total message blocks

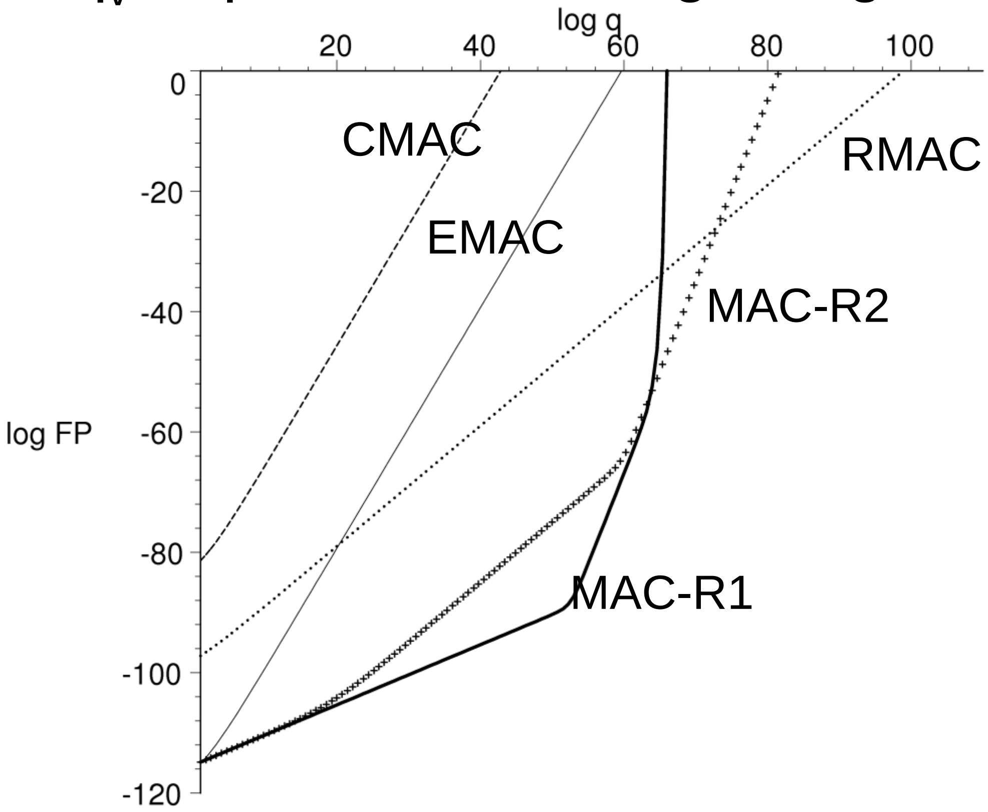
tag length is n bits

$$\delta(a) = \left(1 - \frac{a-1}{2^n}\right)^{-\frac{a}{2}}, \mathsf{d}(\ell) \approx \log \ell$$

note: CMAC bound was improved to O(σ q/2^n) by Nandi

A graphical bound comparison

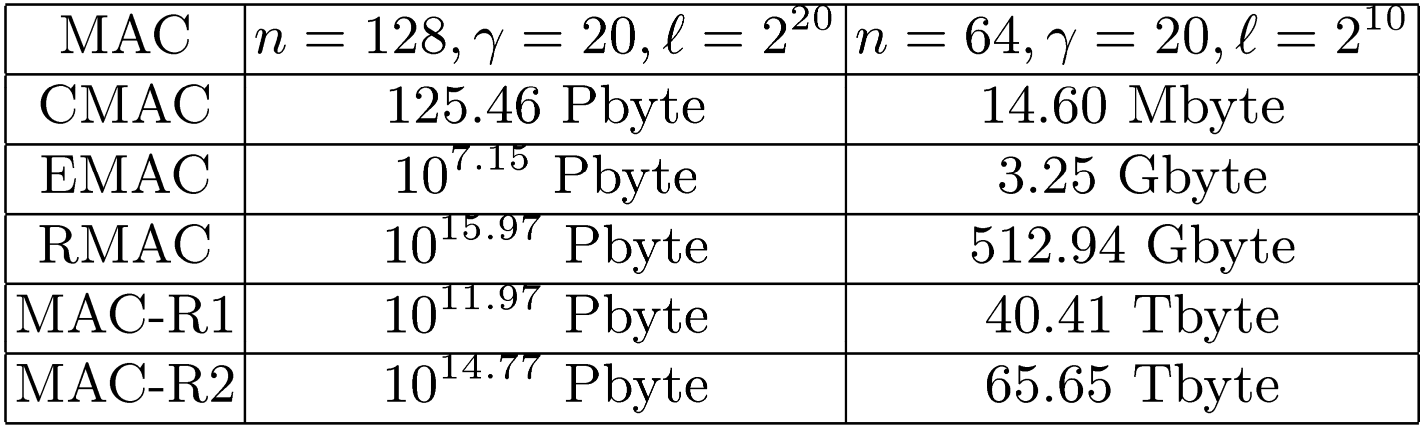
n=128, $q_v = q^{1/2}$, fixed message length $\ell = 2^{20}$



- MAC-R1 bound quickly reaches 1 after 2^{64}
- R1, R2 are even better than RMAC for a certain range
  - due to the difference in the shapes of $q/2^n$ (RMAC) and $q^3/2^{2n}$ (ours)

A numerical comparison

- Let $2^{-\gamma}$ be the maximum acceptable FP
- We compute the maximum amount of data processed by one key
  - When n=64, R1 and R2 can process order of terabytes



MAC	$n = 128, \gamma = 20, \ell = 2^{20}$	$n = 64, \gamma = 20, \ell = 2^{10}$
CMAC	125.46 Pbyte	14.60 Mbyte
EMAC	$10^{7.15}$ Pbyte	3.25 Gbyte
RMAC	$10^{15.97}$ Pbyte	512.94 Gbyte
MAC-R1	$10^{11.97}$ Pbyte	40.41 Tbyte
MAC-R2	$10^{14.77}$ Pbyte	65.65 Tbyte

Conclusion

- Two randomized MAC schemes w/ beyond-birthday-bound-security wrt IV length
  - RWMAC : n-bit randomness, 2n-bit-input PRF
  - EHtM : n-bit randomness, n-bit-input PRF, very efficient (only one add. PRF call from HtM)
- Blockcipher modes based on EHtM
  - Secure, efficient MACs using 64-bit blockciphers

Thank you!