Security Analysis of NIST Key Derivation Using Pseudorandom Functions

Yaobin Shen

Joint work with Lei Wang and Dawu Gu

September 5, 2025@GAPS, Singapore

Xiamen University

Content

1 **Introduction**

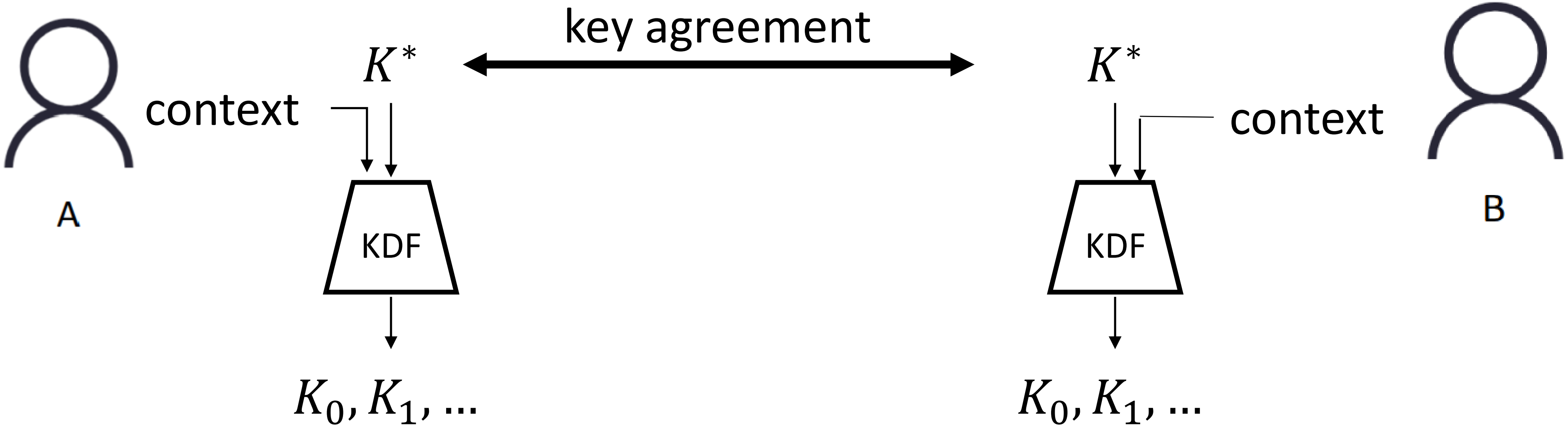
2 Our Contributions

3 Attacks and Proofs

4 Conclusion

Key derivation function

- Key derivation function (KDF): a KDF is a function that can be used to derive variable-length cryptographic keys from a short key



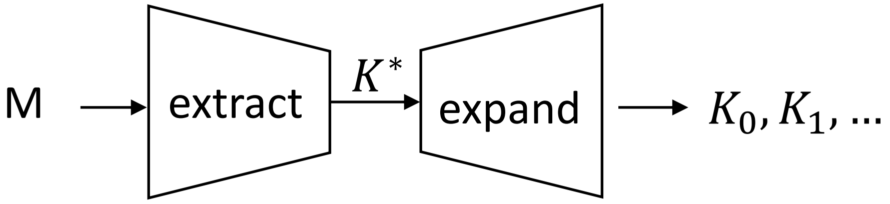
- KDFs play an important role in many cryptographic protocols

KDF and NIST SP 800-108

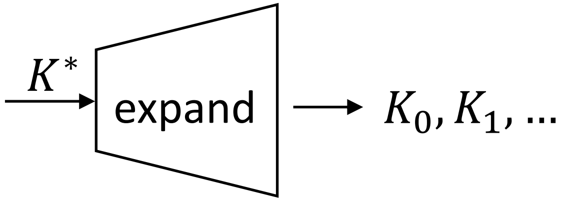
- There are two common methods to build a key derivation function

- Extract-then-Expand KDF (HKDF):
  - extracts a fixed-length pseudorandom key from a source then expands it to generate a key of variable length

- Only expand (NIST SP 800-108):
  - simply expands a fixed-length key to a variable-length key by pseudorandom functions like HMAC and CMAC



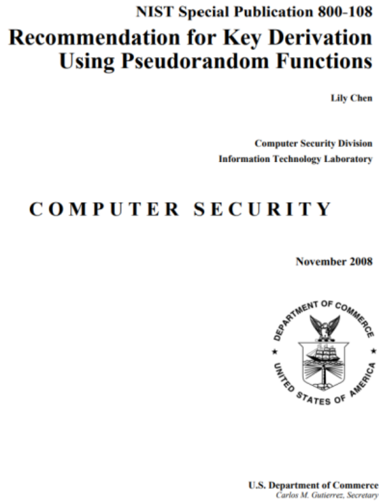
Extract-then-Expand KDF

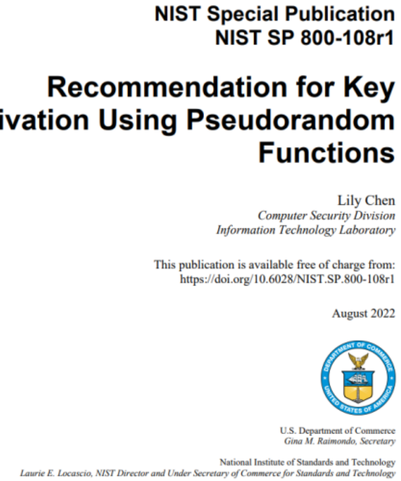


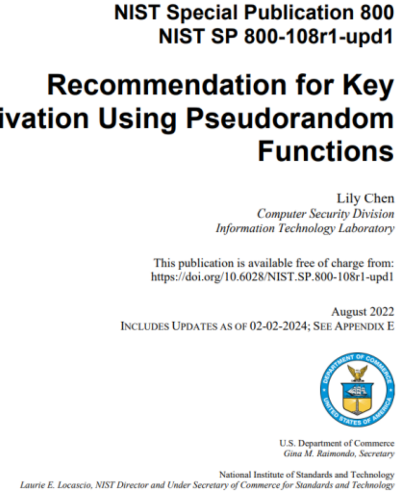
only expand KDF

History of NIST SP 800-108

- The first version of NIST SP 100-108 was published in 2008
  - Based on CMAC: KCTR-CMAC, KFB-CMAC, KDPL-CMAC
  - Based on HMAC: KCTR-HMAC, KFB-HMAC, KDPL-HMAC
- The second version that is named as NIST SP 800-108r1 and published in 2022 , a KDF using KMAC was included
- An updated version NIST SP 800-108r1-upd1 was published in 2024
  - Arciszewski et al. revealed a serious key control security issue regarding KDFs based on CMAC in these three modes

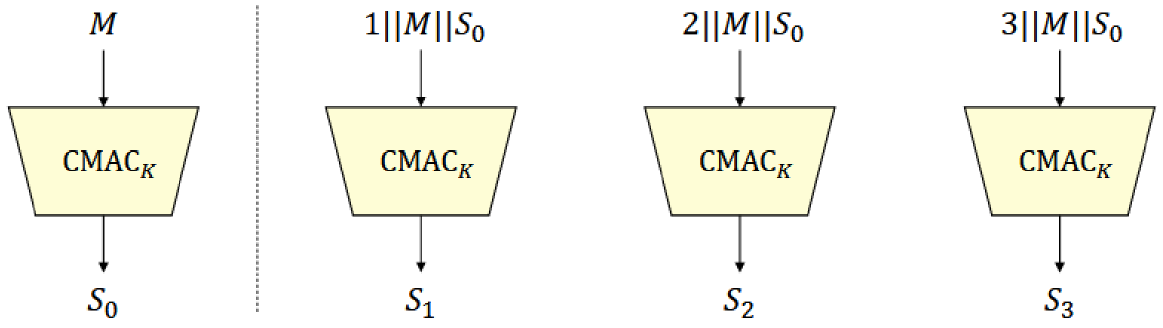

NIST Special Publication 800-108
Recommendation for Key Derivation Using Pseudorandom Functions
Lily Chen
Computer Security Division
Information Technology Laboratory
COMPUTER SECURITY
November 2008
U.S. Department of Commerce
Carlos M. Gutierrez, Secretary


NIST Special Publication
NIST SP 800-108r1
Recommendation for Key Derivation Using Pseudorandom Functions
Lily Chen
Computer Security Division
Information Technology Laboratory
This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-108r1
August 2022
U.S. Department of Commerce
Gina M. Raimondo, Secretary
National Institute of Standards and Technology
Laurie E. Locascio, NIST Director and Under Secretary of Commerce for Standards and Technology


NIST Special Publication 800
NIST SP 800-108r1-upd1
Recommendation for Key Derivation Using Pseudorandom Functions
Lily Chen
Computer Security Division
Information Technology Laboratory
This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-108r1-upd1
August 2022
INCLUDES UPDATES AS OF 02-02-2024; SEE APPENDIX E
U.S. Department of Commerce
Gina M. Raimondo, Secretary
National Institute of Standards and Technology
Laurie E. Locascio, NIST Director and Under Secretary of Commerce for Standards and Technology

KDFs from NIST SP 800-108 - Counter Mode

- Based on CMAC



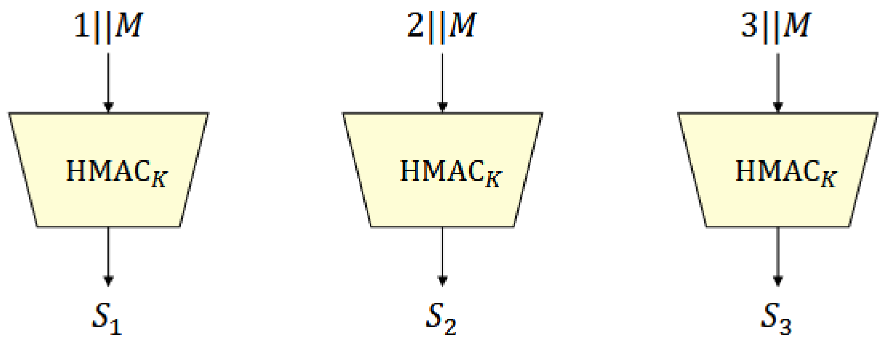
KCTR-CMAC

procedure KCTR-CMAC $(K, \text{Label}, \text{Context}, L)$
$b \leftarrow \lceil L/n \rceil$; $C \leftarrow \varepsilon$
if $b > 2^r - 1$ **then return** $\perp$
$S_0 \leftarrow \mathsf{CMAC}(K, \text{Label} \,\|\, 0x00 \,\|\, \text{Context} \,\|\, [L]_2)$
for $i \leftarrow 1$ **to** b **do**
  $S_i \leftarrow \mathsf{CMAC}(K, [i]_2 \,\|\, \text{Label} \,\|\, 0x00 \,\|\, \text{Context} \,\|\, [L]_2 \,\|\, S_0)$
  $C = C \,\|\, S_i$
return $C[1:L]$

- KCTR − CMAC (K, Label, Context, L)
  - K： Input Key Material
  - Lable： a bit string that identifies the purpose for the derived keying material
  - Context： a bit string that contains the information related to the dervied keying material
  - L： The desired bit length of the output key
  - $M = \text{Label} \,\|\, 0\text{x}00 \,\|\, \text{Context} \,\|\, [\text{L}]_2$

KDFs from NIST SP 800-108 - Counter Mode

- Based on HMAC



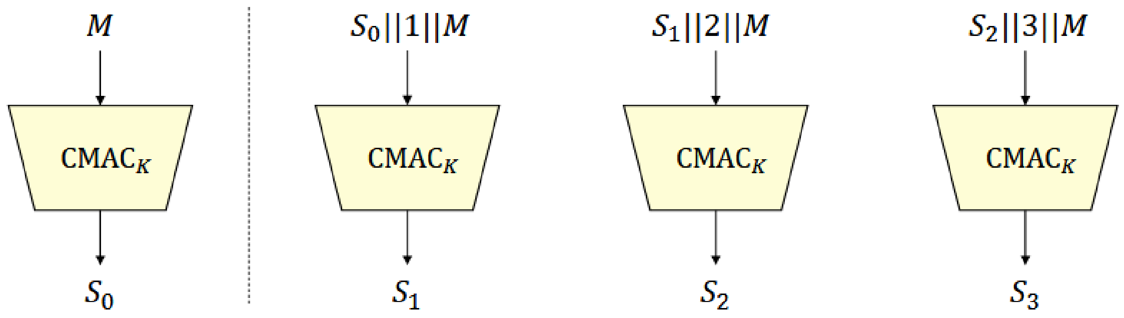
KCTR-HMAC

procedure KCTR-HMAC $(K, \text{Label}, \text{Context}, L)$
$b \leftarrow \lceil L/n \rceil$; $C \leftarrow \varepsilon$
if $b > 2^r - 1$ **then return** $\perp$
for $i \leftarrow 1$ **to** b **do**
    $S_i \leftarrow \text{HMAC}(K, [i]_2 \parallel \text{Label} \parallel 0x00 \parallel \text{Context} \parallel [L]_2)$
    $C = C \parallel S_i$
return $C[1:L]$

- $\text{KCTR} - \text{HMAC}\ (\text{K}, \text{Label}, \text{Context}, \text{L})$
  - K： Input Key Material
  - Lable： a bit string that identifies the purpose for the derived keying material
  - Context： a bit string that contains the information related to the dervied keying material
  - L： The desired bit length of the output key
  - $\text{M} = \text{Label} \parallel 0\text{x}00 \parallel \text{Context} \parallel [\text{L}]_2$

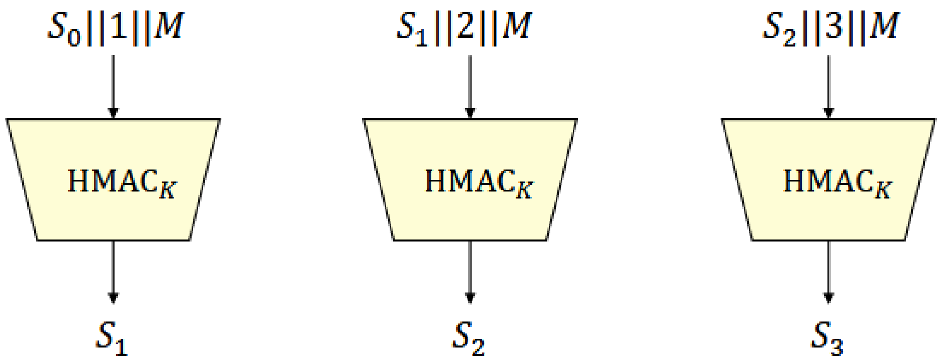
KDFs from NIST SP 800-108 - Feedback Mode

- Based on CMAC



KFB-CMAC

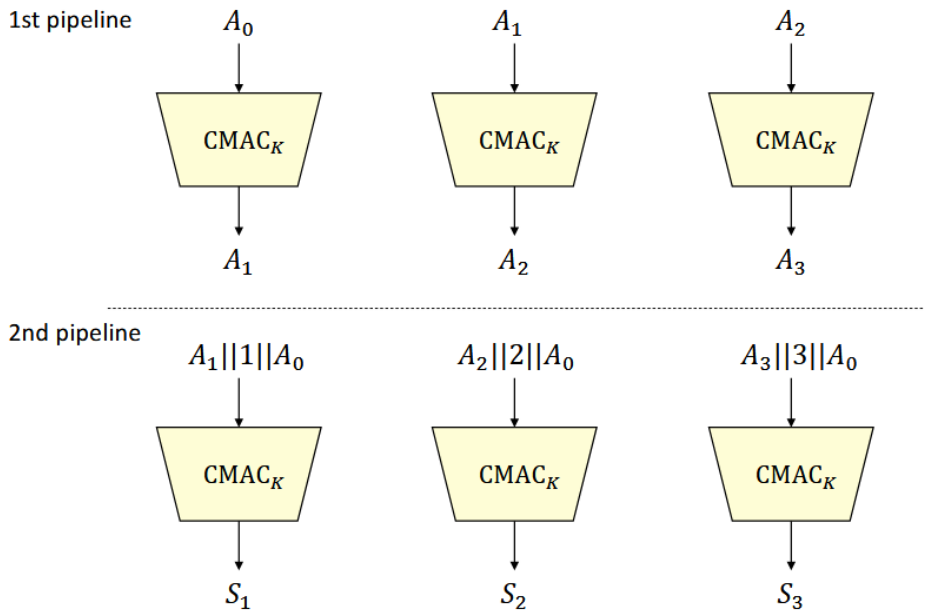
- Based on HMAC



KFB-HMAC

KDFs from NIST SP 800-108 - Double-pipeline Mode

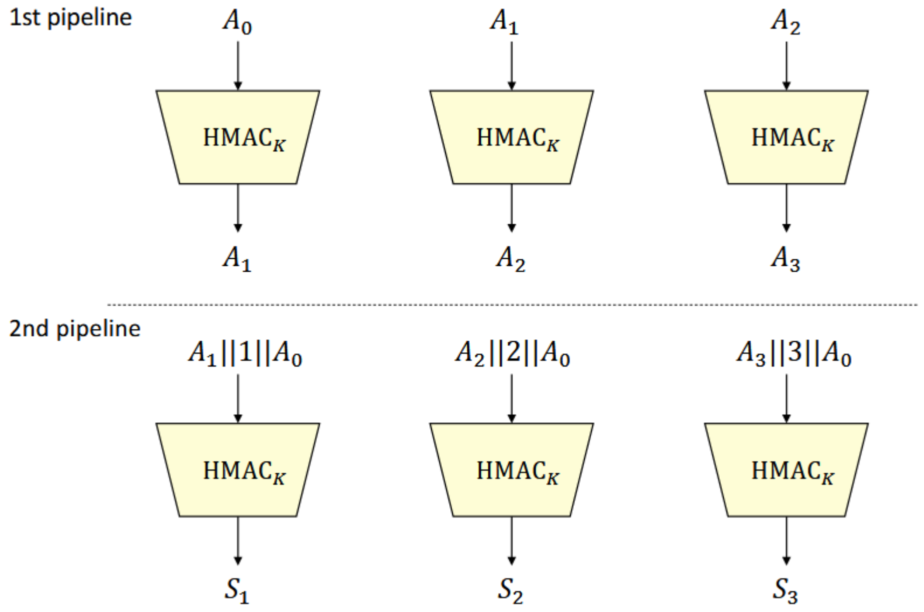
- Based on CMAC



KDPL-CMAC

- The counter i is mandatory

- Based on HMAC

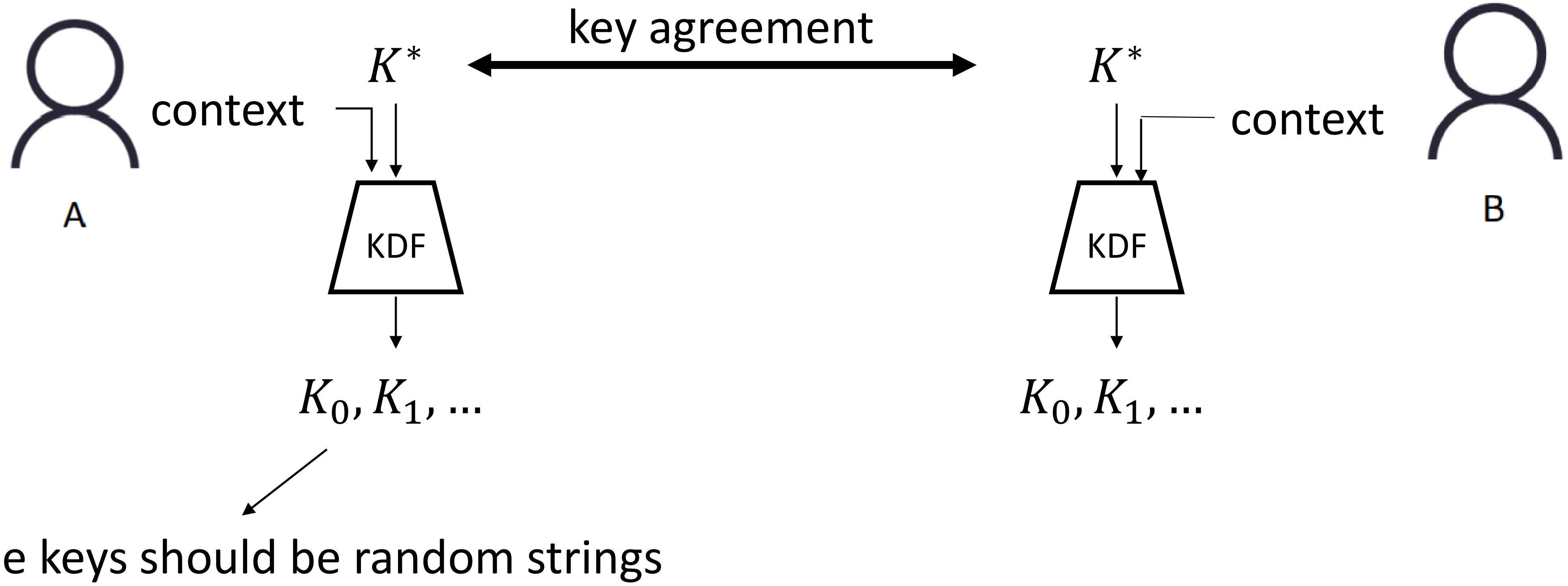


KDPL-HMAC

- The counter i is optional

Requested security property: volPRF

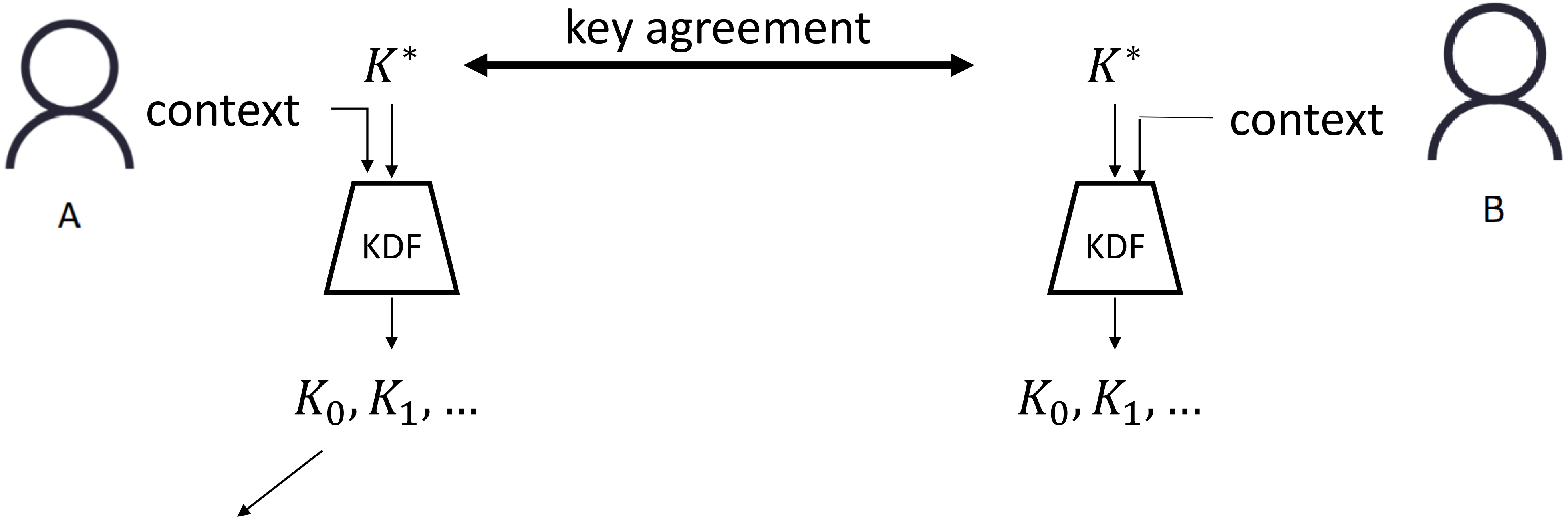
- volPRF (variable output length PRF) Security
  - the basic security property of a KDF to output many keys



Requested security property: collision resistance

- Context binding security -> Collision Resistance

"assurance that all parties who (correctly) derive the keying material share the same understanding of who will access it and in which session it will be used. If those parties have different understandings, then they will derive different keying material"
[NIST, section 6.6]

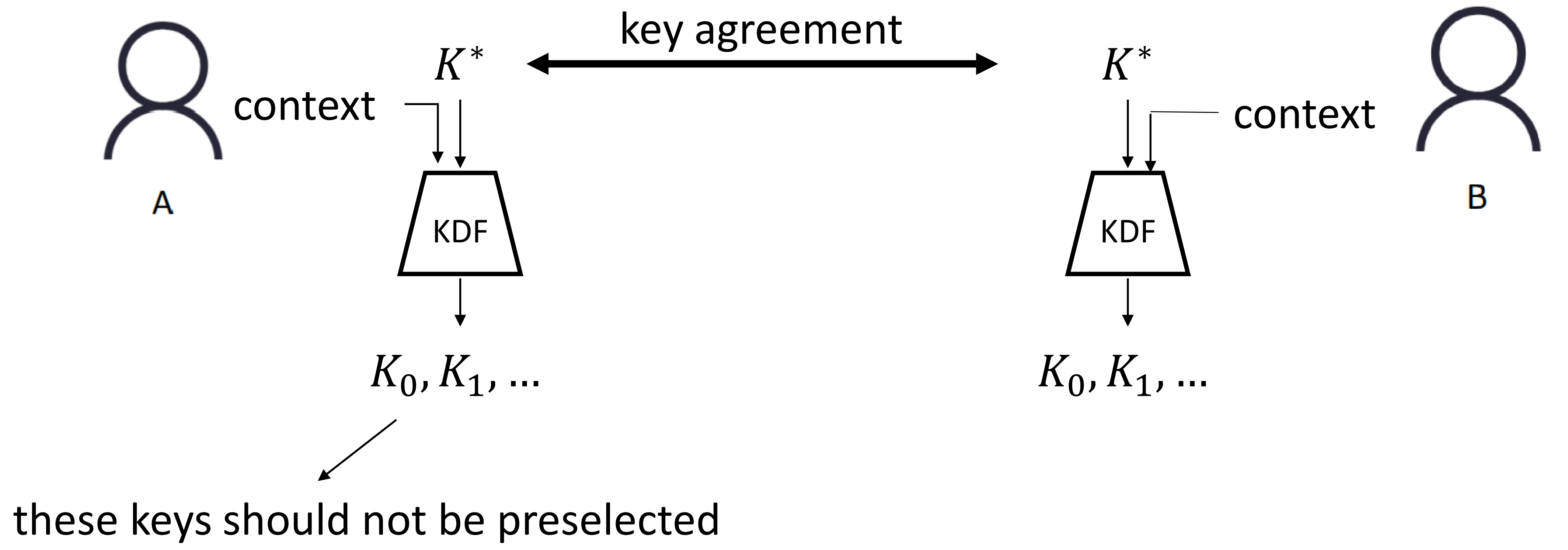


these keys should not collide for different context

Requested security property: Preimage resistance

- Key control security -> Preimage resistance

"(even with knowledge of the input key K) no single party can manipulate the process in such a way as to force output keying material to a preselected value" [NIST, section 6.7]



Motivation

- Despite its standardization in 2008 and widespread use, until now, NIST SP 800-108 has lacked a formal security analysis to validate these security properties, including
  - volPRF
  - collision resistance (context binding)
  - preimage resistance (key control security)

Content

1 Introduction

2 **Our Contributions**

3 Attacks and Proofs

4 Conclusion

Concurrent work

- Ritam Bhaumik, Avijit Dutta, Akiko Inoue, Tetsu Iwata, Ashwin Jha, Kazuhiko Minematsu, Mridul Nandi, Yu Sasaki, Meltem Sönmez Turan, Stefano Tessaro: Cryptographic Treatment of Key Control Security - In Light of NIST SP 800-108, CRYPTO 2025, ePrint 2025/1123
- They focus on key control security (preimage resistance)
  - they provide a generalized security definition of key control security
  - they give birthday-bound proofs of key control security of KDFs based on KMAC, HMAC
  - they show birthday-bound key control attacks of KDFs based on CMAC
  - proofs of key control security of KDFs based on CMAC remain open

Our results

- We give formal security analysis of NIST SP 800-108r1-upd1, including {KCTR, KFB, KDPL}-CMAC, and {KCTR, KFB, KDPL}-HMAC
- Three security properties are covered
  - volPRF
  - collision resistance
  - preimage resistance

Scheme	volPRF	Collision	Preimage
KCTR-CMAC	$O(\frac{q^2b^2}{2^n} + \frac{qb\ell^2}{2^n})$	no	$O(\frac{p}{2^n} + \frac{p^2\ell}{2^n})$
KCTR-HMAC	$O(\frac{q^2b^2}{2^n})$	$O(\frac{p^2}{2^n})$	$O(\frac{p^2}{2^n})$
KFB-CMAC	$O(\frac{q^2b^2}{2^n} + \frac{qb\ell^2}{2^n})$	$O(\frac{p^2}{2^n} + \frac{p^2\ell}{2^n})$	$O(\frac{p}{2^n} + \frac{p^2\ell}{2^n})$
KFB-HMAC	$O(\frac{q^2b^2}{2^n} + \frac{qb^2}{2^n})$	$O(\frac{p^2}{2^n})$	$O(\frac{p^2}{2^n})$
KDPL-CMAC	$O(\frac{q^2b^2}{2^n} + \frac{qb\ell^2}{2^n})$	$O(\frac{p^2}{2^n} + \frac{p^2\ell}{2^n})$	$O(\frac{p}{2^n} + \frac{p^2\ell}{2^n})$
KDPL-HMAC	$O(\frac{q^2b^2}{2^n} + \frac{qb^2}{2^n})$	$O(\frac{p^2}{2^n})$	$O(\frac{p^2}{2^n})$

Security analysis of CMAC-Based KDFs

- volPRF Security
  - KCTR-CMAC, KFB-CMAC, and KDPL-CMAC are a secure volPRF with the bound $O(\frac{q^2b^2}{2^n} + \frac{qbl^2}{2^n})$
- Collision Resistance
  - KFB-CMAC and KDPL-CMAC are collision resistant with the bound $O(\frac{p^2}{2^n} + \frac{p^2l}{2^n})$
  - KCTR-CMAC is not collision resistant
- Preimage Resistance
  - these three KDFs based on CMAC are preimage resistant with the bound $O(\frac{p}{2^n} + \frac{p^2l}{2^n})$

Security analysis of HMAC-Based KDFs

Xiamen University

- volPRF Security
  - KCTR-HMAC, KFB-HMAC, and KDPL-HMAC are a secure volPRF with the bound around $O(\frac{q^2b^2}{2^n}+\frac{qb^2}{2^n})$

- Collision Resistance
  - negative results： if key is of variable length, there are collision attacks against these KDFs
  - positive results： if key is of fixed length and less than d − 1 bits, these KDFs are collision resistant with the bound $O(\frac{p^2}{2^n})$

- Preimage Resistance
  - these KDFs are preimage resistant with the bound $O(\frac{p^2}{2^n})$

Content

1 Introduction

2 Our Contributions

3 **Attacks and Proofs**

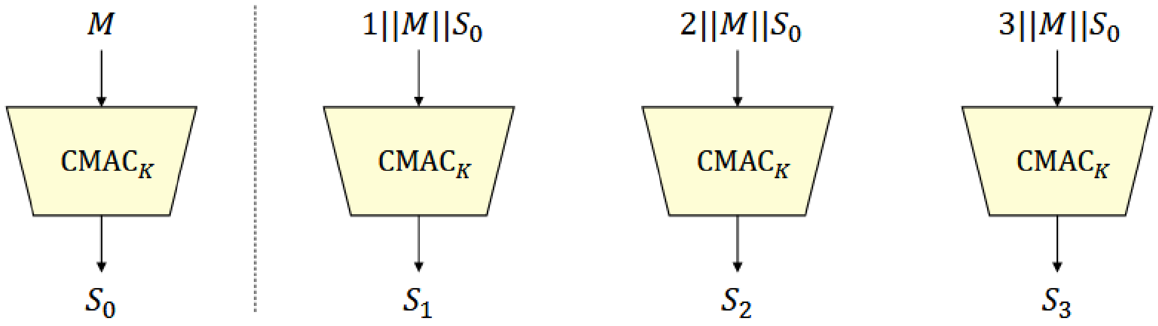
4 Conclusion

KCTR-CMAC: volPRF security

Theorem 1. *For any adversary $\mathcal{A}$ against the volPRF security of* KCTR-CMAC *that runs in time at most t, makes at most q queries, with each query being of block length at most ℓ and being of output block length at most b, we have*

$$\mathsf{Adv}^{\mathrm{volprf}}_{\mathsf{KCTR\text{-}CMAC}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathrm{prp}}_{E}(\mathcal{B}) + \frac{20q^2(b+1)^2}{2^n} + \frac{23q(b+1)(\ell+2)^2}{2^n} ,$$

by assuming $q(b+1) \leq 2^{n/2-1}$ and $\ell + 2 \leq 2^{n/4-0.5}$, where $\mathcal{B}$ is an adversary against the PRP security of the block cipher E that runs in time at most $t' = t + q(b+1)(\ell+3)t_E$ and makes at most $q(b+1)(\ell+3)$ block cipher queries where t_E denotes the running time for one computation of E.



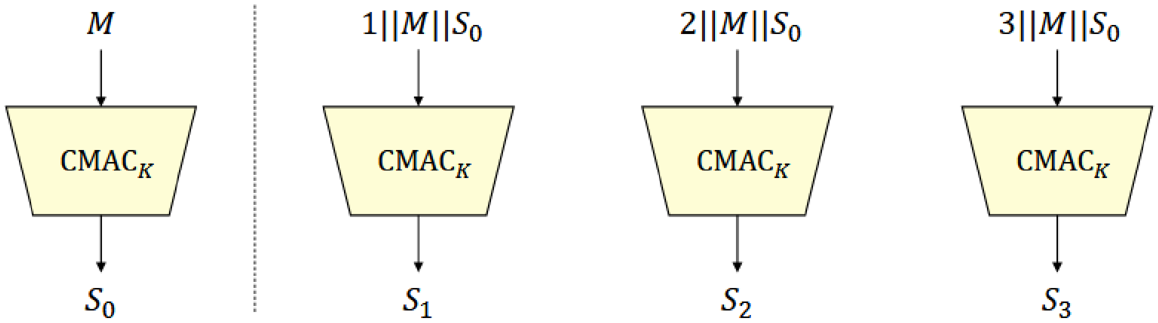
KCTR-CMAC

- Proof idea： reduction to the PRF security of CMAC

KCTR-CMAC: collision attack

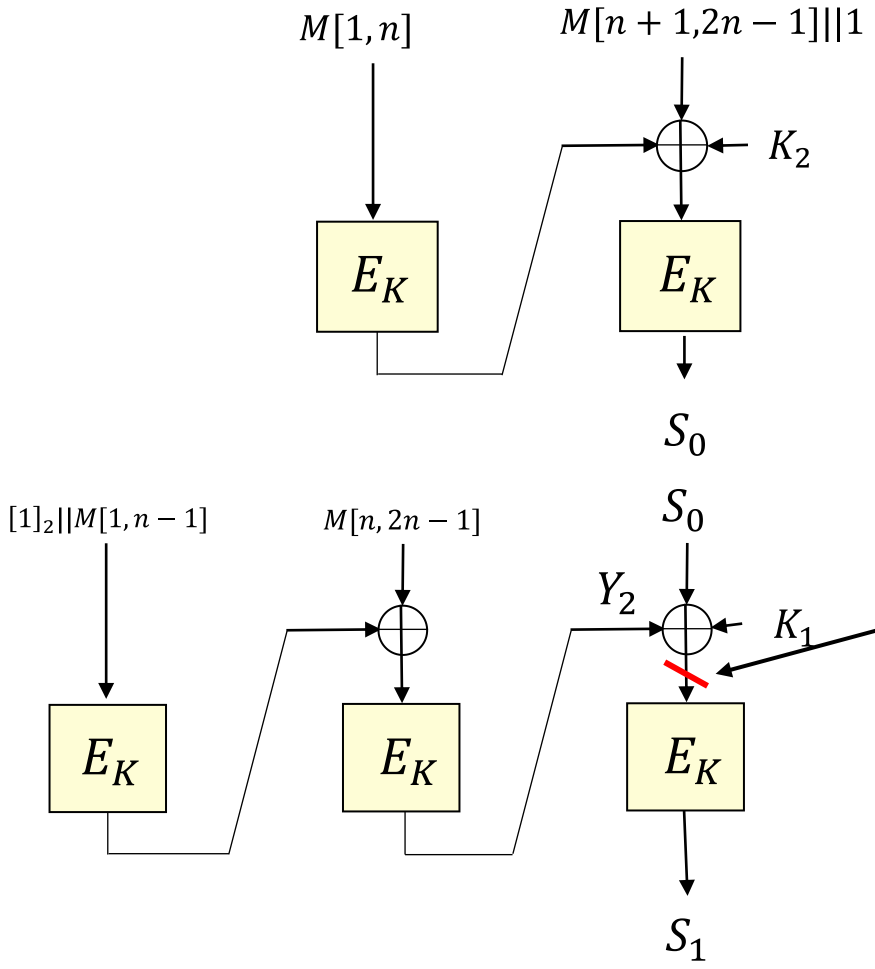
- Goal: find a pair of (K, Label, Context, L) an (K', Label', Context', L') such that

$$\text{KCTR-CMAC}(K, \mathrm{Label}, \mathrm{Context}, L) = \text{KCTR-CMAC}(K', \mathrm{Label}', \mathrm{Context}', L')$$



KCTR-CMAC

KCTR-CMAC: for two block message



a collision here leads to the collision on the output

Find a collision on the input to the last block cipher:

$$Y_2 \oplus S_0 = Y_2' \oplus S_0'$$

KCTR-CMAC: collision attack procedures

- It requires for two messages M and M'

$$E_K(S_0 \oplus K_1 \oplus E_K(M[n:2n-1] \oplus E_K([1]_2 \parallel M[1:n-1])))$$
$$= E_K(S_0' \oplus K_1 \oplus E_K(M'[n:2n-1] \oplus E_K([1]_2 \parallel M'[1:n-1])))$$

- Removing the outer block cipher call:

$$S_0 \oplus E_K(M[n:2n-1] \oplus E_K([1]_2 \parallel M[1:n-1]))$$
$$= S_0' \oplus E_K(M'[n:2n-1] \oplus E_K([1]_2 \parallel M'[1:n-1]))$$

- If $S_0 = E_K(M[n:2n-1] \oplus E_K([1]_2 \parallel M[1:n-1]))$

  and $S_0' = E_K(M'[n:2n-1] \oplus E_K([1]_2 \parallel M'[1:n-1]))$

  then the above equation holds (both equal to 0^n)

KCTR-CMAC: collision attack procedures

- The condition $S_0 = E_K(M[n:2n-1] \oplus E_K([1]_2 \parallel M[1:n-1]))$ is the same as

$$E_K((M[n+1:2n-1] \parallel 1) \oplus K_2 \oplus E_K(M[1:n]))$$
$$= E_K(M[n:2n-1] \oplus E_K([1]_2 \parallel M[1:n-1]))$$

- Removing the outer block cipher call:

$$(M[n+1:2n-1] \| 1) \oplus K_2 \oplus E_K(M[1:n]) = M[n:2n-1] \oplus E_K([1]_2 \| M[1:n-1])$$

- Let $(M[n+1:2n-1] \parallel 1) \oplus M[n:2n-1] = \mathsf{cst}$
- Then the above equation is the same as

$$\begin{aligned} M[n+1] \oplus M[n] &= \mathsf{cst}[1] \\ M[n+2] \oplus M[n+1] &= \mathsf{cst}[2] \\ &\vdots \\ M[2n-1] \oplus M[2n-2] &= \mathsf{cst}[n-1] \\ 1 \oplus M[2n-1] &= \mathsf{cst}[n] \ . \end{aligned}$$

- Solve these equations, we can obtain M. Similarly, obtain M'

Xiamen University

KCTR-CMAC: collision attack probability

- The success probability of finding this pair is about $2^{-2\lceil \log_2(n) \rceil - 2}$ as the last $\lceil \log_2(n) \rceil$ bits of M and M' should be the length encoding

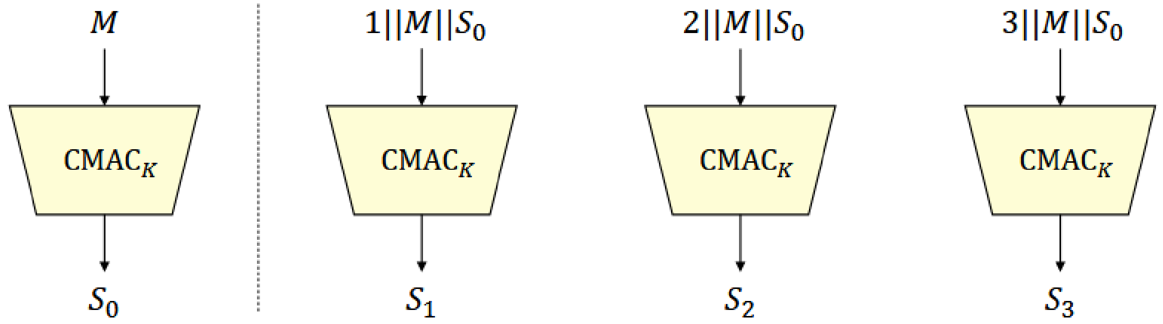
- However, if the input data is defined in the order of Label||0x00||L||Context||S_0 as permitted by NIST standard, **the collision probability becomes 1/4 and requires only 6 block cipher queries**

KCTR-CMAC: preimage resistance

Theorem 2. *For any adversary $\mathcal{A}$ that makes at most p ideal-cipher queries to E and E^{-1}, we have*

$$\mathsf{Adv}^{\mathsf{epre}}_{\mathsf{KCTR\text{-}CMAC}}(\mathcal{A}) \leq \frac{4p}{2^n} + \frac{2p^2\ell}{2^n} \ ,$$

by assuming $p \leq 2^{n-1}$ where ℓ is the maximum block length of a query to the key derivation function.



KCTR-CMAC

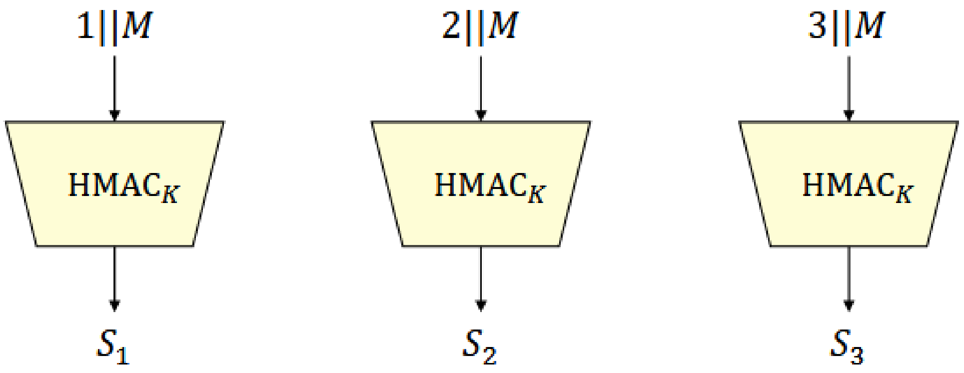
- Proof intuition : it requires handling a message twice to produce a block

KCTR-HMAC: volPRF security

Theorem 3. *For any adversary $\mathcal{A}$ against the volPRF security of* KCTR-HMAC *that runs in time at most t, makes at most q queries, with each query being of block length at most ℓ and being of output block length at most b, we have*

$$\mathsf{Adv}^{\mathrm{volprf}}_{\mathsf{KCTR\text{-}HMAC}}(\mathcal{A}) \leq (\ell+4)\,\mathsf{Adv}^{\mathrm{prf}}_{h}(\mathcal{A}_1)+\mathsf{Adv}^{\mathrm{prf}}_{\overline{h}}(\mathcal{A}_2)+\mathsf{Adv}^{\mathrm{rkaprf}}_{\Phi_{\mathrm{zio},e},\overline{h}}(\mathcal{B})+\frac{qb(qb-1)}{2^{n+1}} \;.$$

Adversaries $\mathcal{A}_1$ and $\mathcal{A}_2$ are against the PRF security of h and $\overline{h}$ respectively, where $\mathcal{A}_1$ makes at most qb queries and $\mathcal{A}_2$ makes one query. Adversary $\mathcal{B}$ makes two queries. The running times of adversaries $\mathcal{A}_1, \mathcal{A}_2$ and $\mathcal{B}$ are about the same as that of $\mathcal{A}$.



KCTR-HMAC

- Proof idea: reduce to the PRF security of HMAC

KCTR-HMAC: collision resistance and preimage resistance

Xiamen University

Theorem 4. *Suppose that the key length of* KCTR-HMAC *is fixed and less than $d-1$ bits. For any adversary $\mathcal{A}$ that makes at most p queries to the compression function h, we have*

$$\mathsf{Adv}^{\mathsf{coll}}_{\mathsf{KCTR\text{-}HMAC}}(\mathcal{A}) \leq \frac{13p^2}{2^n} \ .$$

Theorem 5. *Suppose that the key length of* KCTR-HMAC *is fixed and less than $d-1$ bits. For any adversary $\mathcal{A}$ that makes at most p queries to the compression function h, we have*

$$\mathsf{Adv}^{\mathsf{epre}}_{\mathsf{KCTR\text{-}HMAC}}(\mathcal{A}) \leq \frac{13p^2}{2^n} \ .$$

- Proof idea: rely on the indifferentiability of the underling HMAC

KCTR-HMAC with variable-length key : collision attack

Xiamen University

- Definition of HMAC:

$$\mathrm{HMAC}(K, M) = H(K' \oplus \mathrm{opad} \parallel H(K' \oplus \mathrm{ipad} \parallel M))$$

$K' = K||0$ if $|K| < d$, $K' = H(K)$ if $|K| > d$

- If the key is of variable length (which is allowed in this standard), there is a collision attack against HMAC
  - the pair of $(\mathrm{K}_1, \mathrm{M}_1)$ and $(\mathrm{K}_2, \mathrm{M}_1)$ can result in a collision where

if $|K_1| < d$ set $K_2 = K_1 \parallel 0^*$

if $|K_1| > d$ set $K_2 = H(K_1)$

- This collision attack applies to KCTR-HMAC and other HMAC-based KDFs

Content

1 Introduction

2 Our Contributions

3 Attacks and Proofs

4 **Conclusion**

Conclusion

- KCTR-CMAC may not be a good choice in general as a KDF as it is vulnerable to collision attack
- For other KDFs, they are basically good as a KDF, as they are volPRF, collision resistant, and preimage resistant
- KDFs based on HMAC should use a key of fixed length that is less than d − 1 bits (otherwise collision attacks exist)
- NIST 800-108 may be revised for a stronger security, especially KCTR-CMAC as it is not collision resistant (or not context binding) as required by this standard
- More details can be found in ePrint: 2025/815

Responsible disclosure

- We have shared both of our attacks and proofs with NIST (Lily Chen)

Questions or comments?

Thanks!

yaobin.shen@xmu.edu.cn