Committing Wide Encryption Mode with Minimum Ciphertext Expansion

Yusuke Naito (Mitsubishi Electric Corporation)

Yu Sasaki (NTT Social Informatics Laboratories)

Takeshi Sugawara (The University of Electro-Communications)

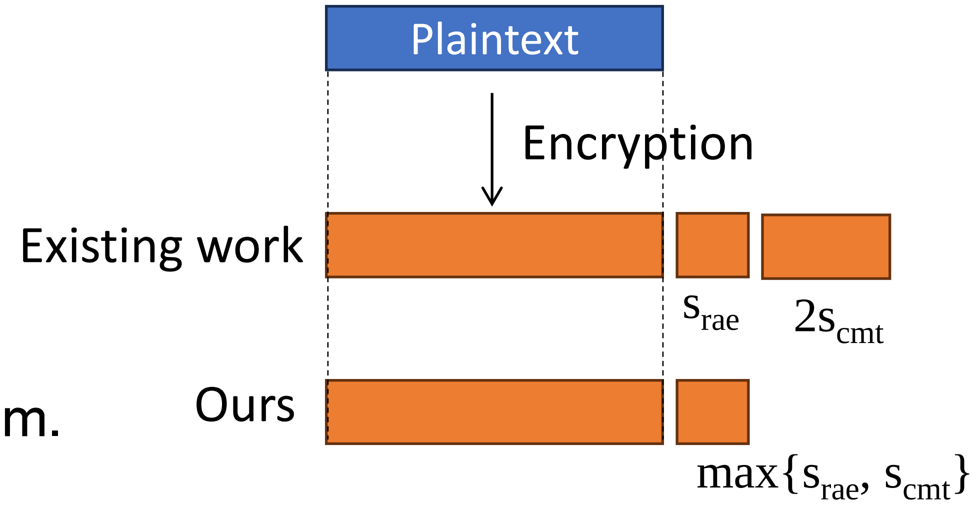
NIST Workshop on the Requirements for an Accordion Cipher Mode 2024
June 21, 2024

Summary

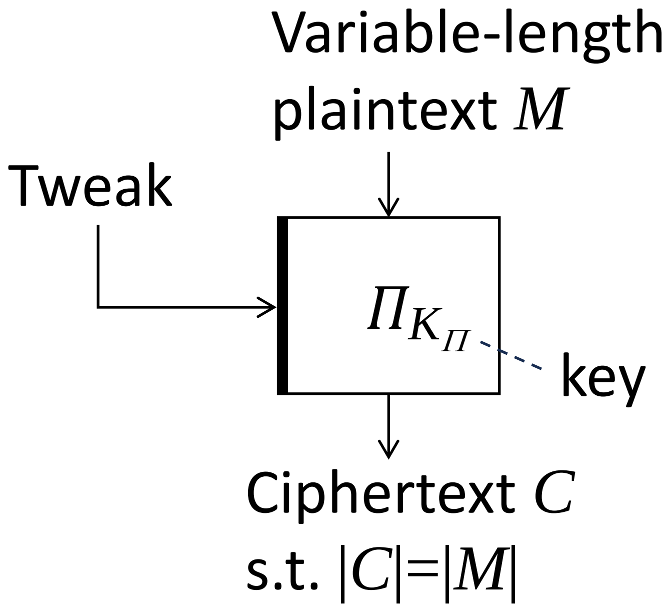
- We design a mode of operation based on a wide encryption (WE)
  - Provides a robust authenticated encryption (RAE), and
  - Ensures committing (CMT-4) security.

- State-of-the-art
  - Requires $s_{\text{rae}}+2s_{\text{cmt}}$-bit ciphertext expansion from an original plaintext to achieve s_{rae}-bit RAE security and s_{cmt}-bit CMT-4 security.

- Our new mode FFF
  - The expansion size is $\max\{s_{\text{rae}}, s_{\text{cmt}}\}$ bits and minimum.



(Tweakable) Wide Encryption (WE)

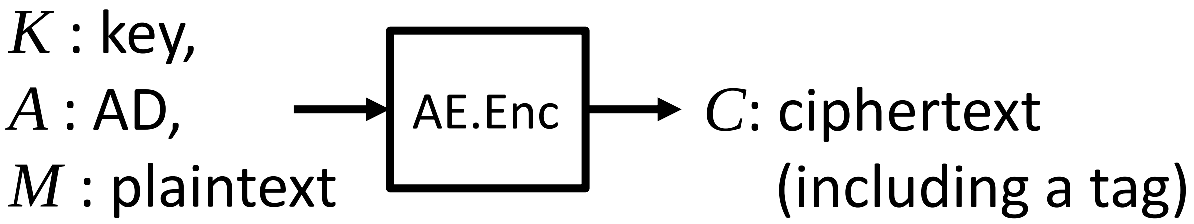


- Arbitrary-length tweakable block cipher.
  - A set of variable-length permutations indexed by a key and a tweak.
- WE has practical applications, including full-disk encryption.
- Several concrete schemes have been designed, e.g., AEZ, Shrimpton-Terashima, etc.
- There are several proposals from the industry, including Adiantum and HCTR2 by Google.
- Moreover, NIST has recently started a discussion on WE, which stimulated even more WE proposals in the last few years, including double-decker and docked-double-decker.
- Security goal: (tweakable) strong pseudorandom permutation (SPRP) security.
  - Indistinguishability between a WE and a tweakable random permutation in the CCA setting.
- With the SPRP assumption, WE is an efficient building block for authenticated encryption.

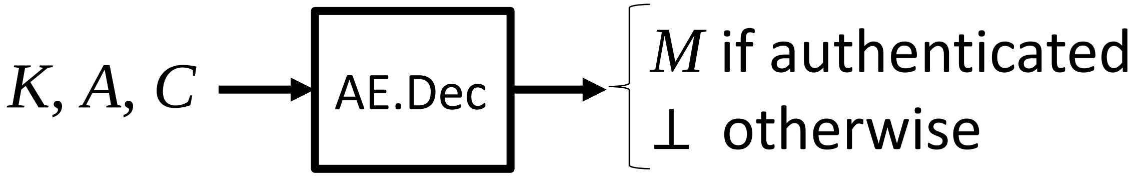
Authenticated Encryption (AE)

- AE provides confidentiality and authenticity.

- AE encryption AE.Enc:
  - Takes a key K,
    associated data A (including a nonce),
    and a plaintext M
  - Returns a ciphertext
    C = AE.Enc(K, A, M).



- AE decryption AE.Dec:
  - Takes K, A, and C,
  - Returns the valid plaintext M if authenticated;
    otherwise returns the invalid symbol $\perp$.

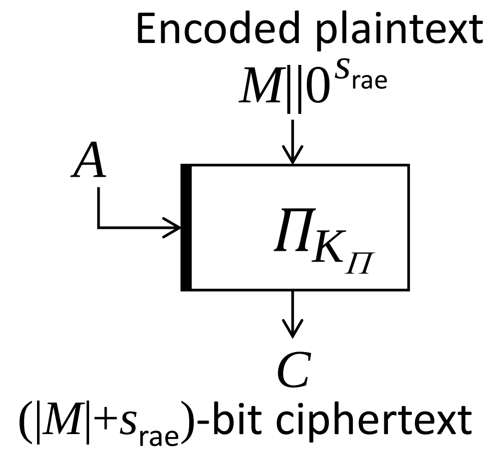


- Standard security goal: Indistinguishability between the target AE and an ideal AE (a random-bit oracle and a reject oracle).

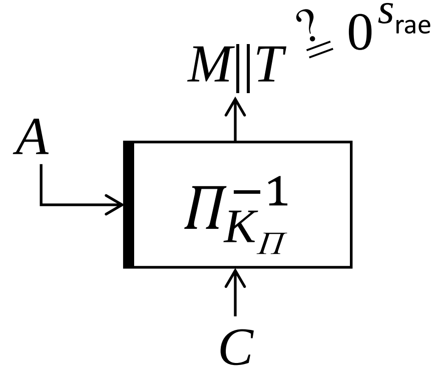
Encode-then-Encipher (EtE)

- EtE is a well-known way of constructing a robust AE (RAE) from a WE.
- Encryption of EtE:
  - Appends s_{rae} bits of zeros to the plaintext M, and
  - Encrypts the encoded plaintext by WE Π_{K_Π} to generate a ciphertext C.
- Decryption of EtE:
  - Decrypts a ciphertext C by $\Pi_{K_\Pi}^{-1}$, and check if $T=0^{s_{\text{rae}}}$.
  - Returns a valid plaintext M if $T=0^{s_{\text{rae}}}$.
- WE's tweak input is used for AD A.
- EtE is RAE secure with an SPRP-secure WE.
  - In the decryption, each unverified plaintext (and tag) $M \| T$ is randomly chosen.
  - Strong robustness against several misuses is ensured, e.g., nonce reuse and the release of unverified plaintexts.
- EtE achieves s_{rae}-bit RAE security which is equal to the size of ciphertext expansion, s_{rae} bits.

Encryption



Decryption



Committing Security

- An actively-studied new property for AE.
- Not covered by the standard AE security notion.
- Researches for committing security have been motivated by the real-world attacks.
  - The multi-recipient integrity attack that delivers malicious content to a targeted user.
  - The partitioning oracle attack that achieves efficient password brute-force attacks.
- The goal of the adversaries is to generate a ciphertext that is successfully decrypted with distinct decryption contexts, i.e., find the following collision.

$$\text{AE.Enc}(K, A, M) = \text{AE.Enc}(K', A', M') \text{ with } \begin{cases} K \neq K' \text{ (CMT-1)}, \\ (K, A, M) \neq (K', A', M') \text{ (CMT-4)}. \end{cases}$$

- Committing security is offline.
- The CMT-4 adversary can choose a key K as well as AD A and a plaintext M.
- Many AE schemes have been designed without considering committing security.
- There are efficient attacks on important AE schemes such as GCM, GCM-SIV, CCM, and ChaCha20-Poly1305.

Committing security of Existing EtE

- Chen et al. (ToSC2023-4, NIST block cipher modes workshop 2023)
  - showed the attacks and proofs on concrete schemes (AEZ, Adiantum-EtE, and HCTR2-EtE).
  - These schemes achieve s_{cmt}-bit CMT-1 security with $2s_{\mathrm{cmt}}$ bits of ciphertext expansion but the security is clipped at n/2 bits (n is a block size of a block cipher).

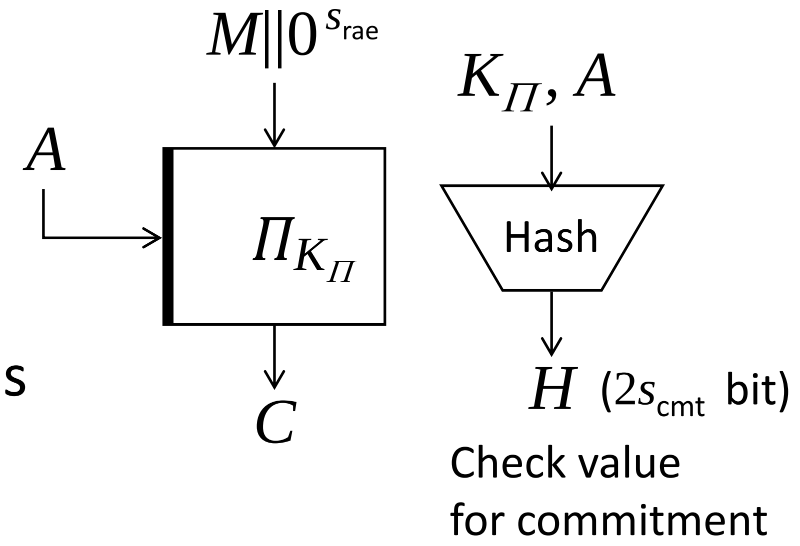
- With commonly used 128-bit block ciphers, i.e., n = 128, their CMT-1 security is at most 64 bits.

- The 64-bit security is insufficient (Chan and Rogaway, ESORICS 2022).
  - At least 80-bit security level is necessary because committing security is offline, where an adversary can efficiently make and verify guesses without any online query, in the same way as brute-force key recovery attack.

- The situation is even worse with CMT-4 security:
  - AEZ, Adiantum-EtE, and HCTR2-EtE are all broken in a constant time.

WE Mode with Collision-Resistant Hash Function

- The committing security of AE is equal to its collision resistance of the encryption of AE.
- By using a collision-resistant hash function, AE can be converted to have CMT-4 security.
- Existing hash-based committing modes.
  - Farshim et al. designed a mode for CMT-1 security.
  - Chan and Rogaway designed CTX for CMT-4 security.
- By the birthday attack, the hash size must be at least $2s_{\mathrm{cmt}}$ bits to achieve s_{cmt}-bit CMT-4 security.
- A naïve combination of EtE and a hash function (right Fig.) comes with $s_{\mathrm{rae}}+2s_{\mathrm{cmt}}$ -bit ciphertext expansion.
- The expansion size
  - Longer than the security level $\max\{s_{\mathrm{rae}}, s_{\mathrm{cmt}}\}$, and
  - Has room for improvement.



Contribution

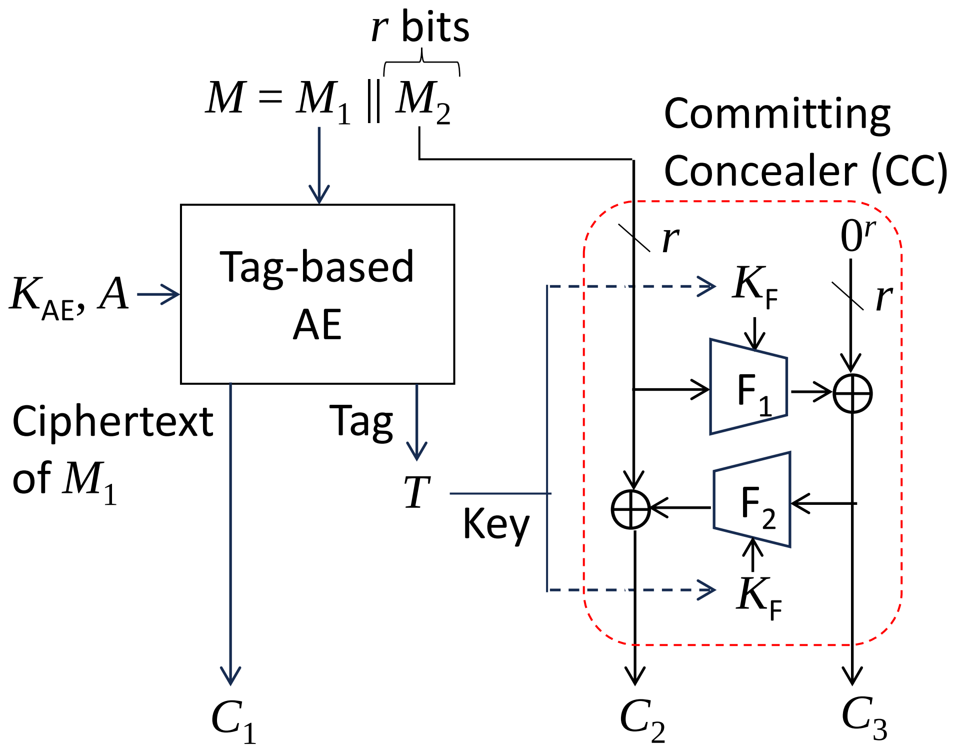
- We present FFF, a new mode that builds an AE from a WE with the following criteria.
  - s_{rae}-bit RAE Security.
  - s_{cmt}-bit CMT-4 Security.
  - Minimum ciphertext expansion, i.e., $\max\{s_{\text{rae}}, s_{\text{cmt}}\}$ bits.

- We design FFF by following the design of the committing concealer (CC)
  - Designed by Bellare and Hoang (NIST block cipher modes workshop 2023, CRYPTO2024).
  - Transforms any tag-based AE to a CMT-4-secure AE with minimum ciphertext expansion.
  - Not support WE.

- We extend the CC's design for WE.

Committing Concealer (CC)

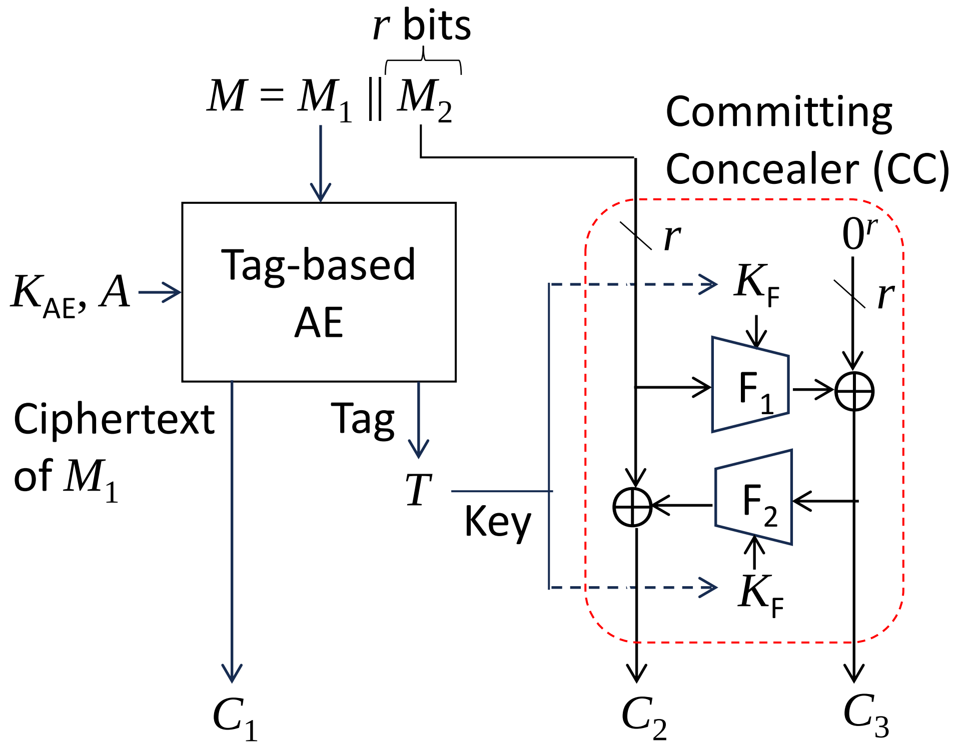
- CC has a 2-round Feistel structure with 0^r, where F_1 and F_2 are keyed hash functions.
- Encryption of tag-based AE with CC:
  - Encrypts M_1 with the underlying tag-based AE to obtain a ciphertext C_1 and a tag T.
  - Encrypts M_2 by CC that uses the tag as its key and generates a $2r$-bit ciphertext $C_2 \| C_3$.
  - $C_1 \| C_2 \| C_3$ is the ciphertext (does not include T).



- Decryption of tag-based AE with CC :
  - Calls the decryption of the tag-based AE to recover M_1 and T.
  - Calls the inverse of CC and checks the authenticity by confirming if the right r-bit part of CC is equal to 0^r.
  - $M_1 \| M_2$ is a valid plaintext if the equation holds.

Committing Concealer (CC)

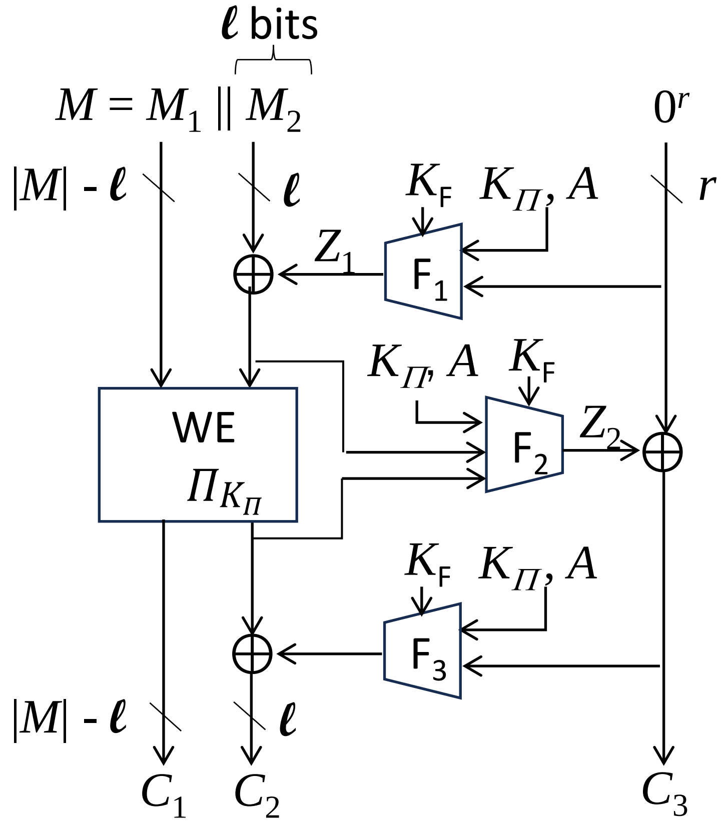
- In order to break the CMT-4 security,
    1. a collision on the $2r$-bit part $C_2||C_3$ must occur (by the forward operation of CC), or
    2. the inverse of CC must hit 0^r.
- For the attack 1, by the birthday analysis, the security level is r bits.
- For the attack 2, the right r-bit part is a random value, and the security level is r bits.



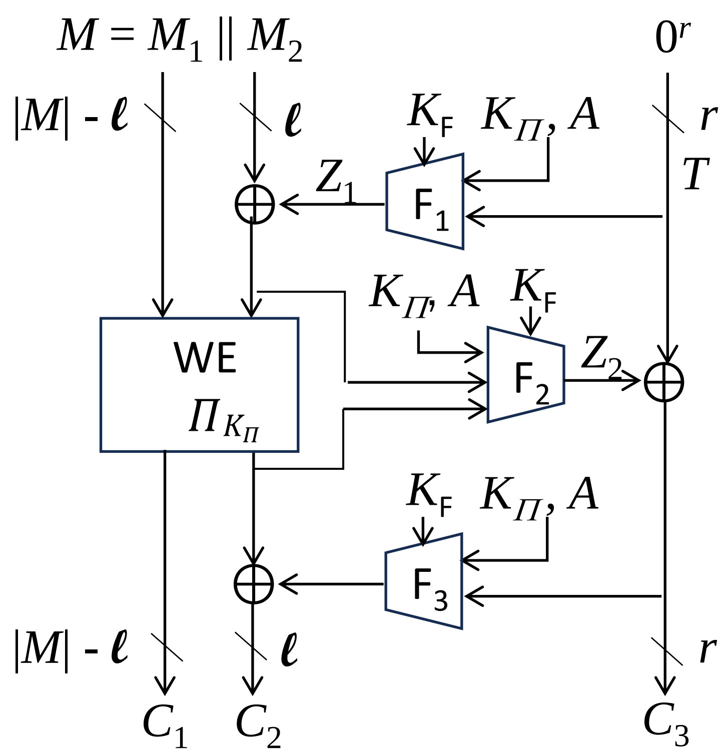
- The tag-based AE with CC achieves r-bit CMT-4 security, which is equal to the size of ciphertext expansion r bits.
- Its extension to WE is not straightforward because the scheme relies on the tag-based AE for recovering T not included in the ciphertext.
- When replacing the tag-based AE with a WE, the inverse of the WE can't be performed since a fraction of its ciphertext is lost.

Our Mode FFF

- We design FFF so that the inverse of WE can be performed.
- FFF has a 3-round Feistel structure with 0^r, and the expansion size is r bits.
- Each round performs a (keyed) hash function F_i with a hash key K_F, a WE's key K_Π, and AD A.
- Since the ciphertext includes all bits of the output of WE, the decryption with FFF can perform the inverse of WE.
- Hash functions F_1, F_2, and F_3
  - Serve as three random oracles for CMT-4 security and pseudorandom functions for RAE security.
  - Can be Instantiated with a single hash function by using domain separations.
  - By using an iterated hash function such as SHA-3 and SHA-2, the internal state after processing the input tuple (K_F, K_Π, A) can be reused for efficiency.



Security of FFF



<u>CMT-4 Security of FFF: $\min\{r, \ell\}$ bits</u>

- To break the CMT-4 security, for distinct tuples (K_F, K_Π, A),
    1. a collision on $C_2 \| C_3$ must occur (forward) or
    2. the right r-bit parts must be 0^r (inverse).
- For the attack 1, the collision security is $\min\{r, \ell\}$ bits by using the multi-collision-based evaluation.
- For the attack 2, the security for the collision with 0^r is r bits.

<u>RAE security of FFF: $\min\{r, \ell/2\}$ bits</u>

- In the decryption, any change in C_1, C_2, C_3, A must change all of M_1, M_2, T randomly.
- A change in C_1 changes the input to Π^{-1}, thus changes M_1, M_2, randomly, and changes the input to F_2 as long as no collision occurs on the ℓ-bit part. Namely, the change randomly changes all of M_1, M_2, T.
- Similarly, any change in C_2, C_3, A will change all of M_1, M_2, T, through Π^{-1} and 3-round Feistel.

<u>Expansion Size of FFF</u>

- FFF achieves r-bit security with $r \geq \ell/2$, i.e., the expansion size is $\max\{s_\mathsf{rae}, s_\mathsf{cmt}\}$ and minimum.

Conclusion

- We studied a WE-based AE mode with s_{rae}-bit RAE security and s_{cmt}-bit CMT-4 security.
- State-of-the-art hash-based mode required $s_{\mathrm{rae}}+2s_{\mathrm{cmt}}$-bit ciphertext expansion, which is not minimum.
- The new WE-based AE mode FFF.
  - The size of ciphertext expansion is $\max\{s_{\mathrm{rae}}, s_{\mathrm{cmt}}\}$ bits, which is minimum.
  - 3-round Feistel Structure, where each hash function call takes a hash key K_{F}, WE's key K_{Π}, and AD A.
  - By using an iterated hash function, such as SHA-3 and SHA-2, the internal state can be efficiently reused between the three hash function calls within FFF.
- Minimizing communication costs is an important criteria for small bandwidth, and our mode is effective for the applications.
- Since not all WE applications require committing security, we suggest that a WE algorithm and a committing mode are designed separately.
- Our research offers a candidate for the committing mode.