The Exact Multi-User Security of (Tweakable) Key Alternating Ciphers with a Single Permutation

Yusuke Naito (Mitsubishi Electric Corporation)

Yu Sasaki (NTT Social Informatics Laboratories)

Takeshi Sugawara (The University of Electro-Communications)

EUROCRYPT2024

May 27, 2024

Summary

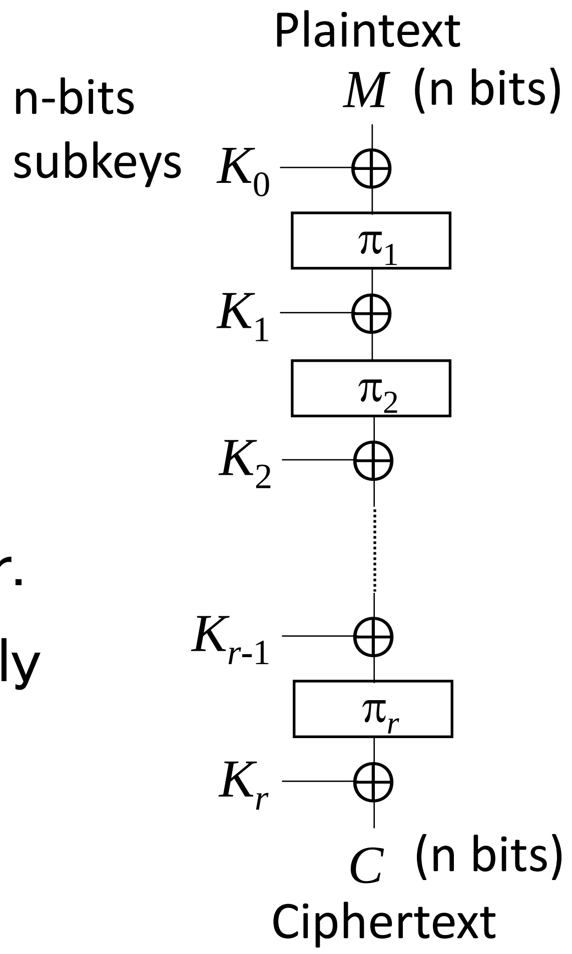
- Target scheme: r-round key alternating cipher (KAC) where r is any
  - Construct a block cipher,
  - Iterate a permutation and an XOR with a subkey,
  - Have r permutation calls and r+1 subkey XOR operations.

- Existing works for KAC:
  - Tight single-user security for KAC with a single random permutation.
  - Tight multi-user security for KAC with r random permutations and r+1 independent subkeys.

- We prove the tight multi-user security of the (tweakable) KAC
  - With a single permutation,
  - With r-wise independent subkeys (the number of independent values is r).

Key Alternating Cipher (KAC)

- r-round KAC
  - n-bit block cipher,
  - with r n-bit permutations $\pi_1, \ldots, \pi_r$,
  - with r + 1 n-bit subkeys $K_0, \ldots, K_r$.
- The single-round KAC is known as the Even-Mansour (EM) cipher, and the r-round KAC is referred to as the r-round iterated EM cipher.
- KAC describes the computational structure of block ciphers commonly used in the real world, such as AES and many other block ciphers.
- The provable security of KAC is their theoretical foundation.
- Proving a tight security of KAC has been an important challenge in symmetric key cryptography research.



Research Topics for KAC

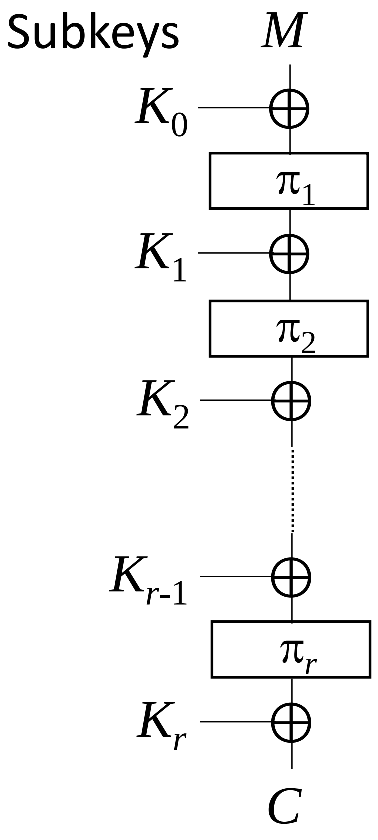
1. Proving the tight bound.
    - Attack bound: rn/(r+1) bits (the attack complexity is $2^{rn/(r+1)}$), i.e., r=3: 3n/4; r=: 4n/5; ….

2. Proving the security of any-round KAC, i.e., r is any.

3. Reducing the number of independent permutations (ideally, a single permutation, i.e., $\pi_1 = \ldots = \pi_r$).

4. Reducing the number of independent subkeys.

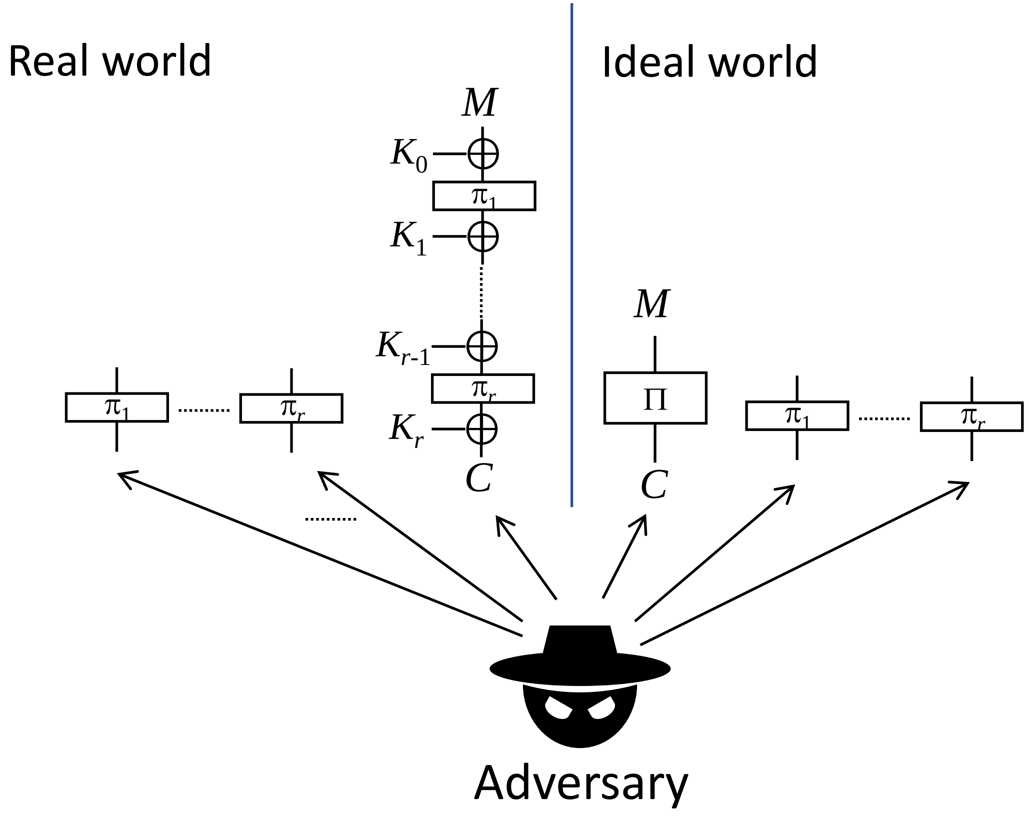
5. Proving the multi-user (mu) security.



Strong Pseudo-Random Permutation (SPRP) Security

- The security of KAC was initially evaluated in the single-user (su) setting.

- Su-SPRP security (right figure):
  - Indistinguishability between a single instantiation of KAC and a random permutation Π.
  - An adversary has access to KAC and Π by construction queries.
  - An adversary has access to the underlying random permutations $\pi_1, \ldots, \pi_r$ by primitive queries (KAC with a single permutation: $\pi_1 = \ldots = \pi_r$).



Existing Works for Su-SPRP Security of KAC

- Since Even-Mansour's work, several works proved the tight su-security bounds of KACs.

Reference	Round w/ Tight Bound	Identical Permutation	Independent Subkeys†	Multi-User Security	Tweakable KAC
Even-Mansour [12]	1	N/A	All	—	—
Bogdanov et al. [3]	2	—	All	—	—
Steinberger [24]	3	—	All	—	—
Lampe et al. [16]	Asymptotic	—	All	—	—
Chen-Steinberger [5]	Any	—	All	—	—
Chen et al. [4]	2	✓	1	—	—
Wu et al. [27]	3	✓	All	—	—
Yu et al. [28]	Any	✓	All	—	—
Cogliati et al. [7]	2	—	2	—	✓
Cogliati et al. [7]	Asymptotic	—	r	—	✓
Cogliati-Seurin [8]	4	—	2	—	✓
Dutta [11]	4	—‡	2	—	✓

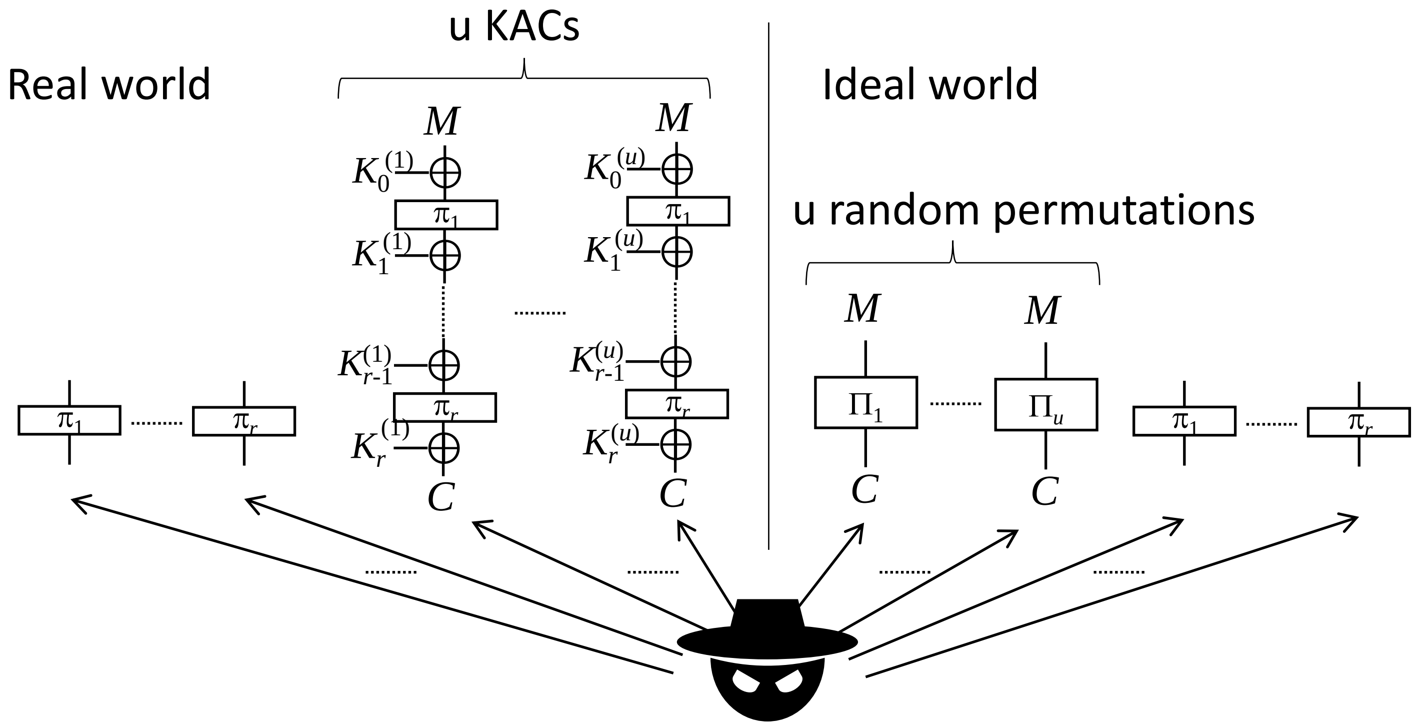
Increasing the number of rounds r.

Reducing the number of independent permutations,
i.e., considering KAC with a single permutation.

Reducing the number of independent subkeys.

Multi-User (Mu) SPRP Security of KAC

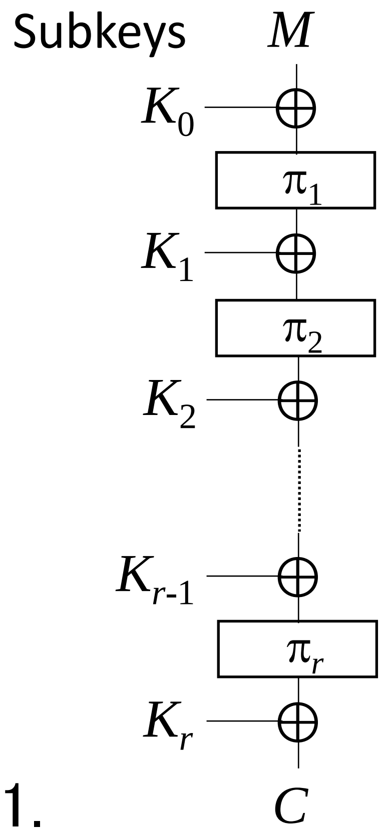
- Compared with the works for the su security, there are not so many results for the mu security of KAC.
- In the mu setting, an adversary wins by breaking any of the keys, which better represents the real-world attacks targeting a particular service rather than a particular user.
- The mu security considers u KAC's instantiations where the user's keys are independent.
- Mu-SPRP security: Indistinguishability between
  - u instantiations of KAC and
  - u random permutations $\Pi_1,\ldots,\Pi_u$.
- The mu adversary can obtain more information than the su adversary.
- The mu security proof is more complex than the su-security proof.



Existing Works for Mu Security of KAC

- There are two works for the tight mu security of KACs.
- Mouha and Luykx (CRYPTO 2015).
  - Tight mu-bound: n/2 bits.
  - Single-round KAC with a single subkey: $K_0 = K_1$.
- Hoang and Tessaro (CRYPTO 2016)
  - Tight mu-bound: rn/(r+1) bits for any r.
  - r independent permutations.
  - r+1 independent subkeys.
- Open problem:
  - Tight mu-bound: rn/(r+1).
  - any round KAC with a single permutation.
  - # of independent values in the subkeys is less than r+1.

Reference	Round w/ Tight Bound	Identical Permutation	Independent Subkeys[†]	Multi-User Security	Tweakable KAC
Mouha-Luykx [19]	1	N/A	1	✓	—
Hoang-Tessaro [14]	Any	—	All	✓	—



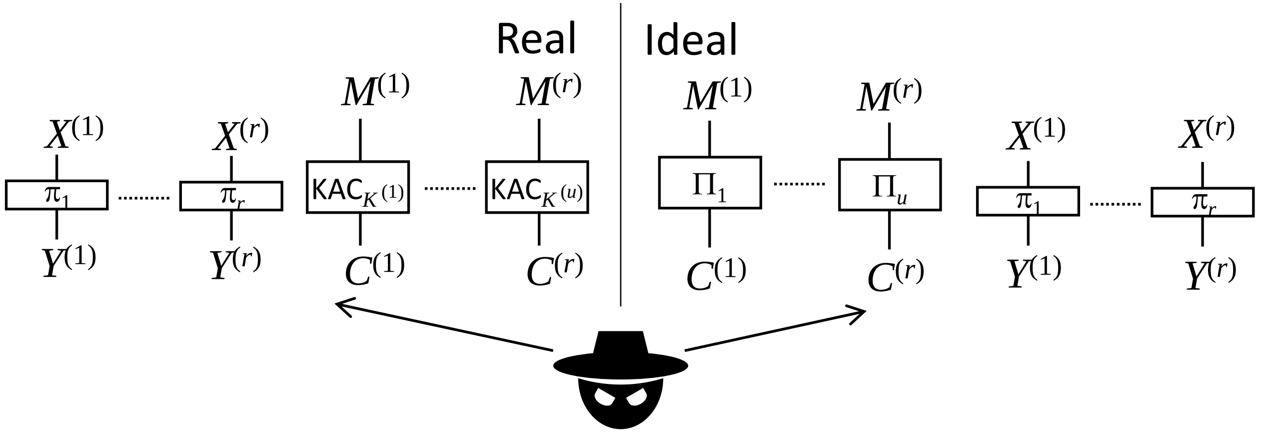
Our Result

- (Tweakable) KAC
  - any r,
  - a single random permutation,
  - a r–wise independent subkeys,
    (r+1 subkeys from r random values).
- Tight mu–bound: rn/(r+1) bits.
- Proof Methods
  - Patarin's coefficient–H technique.
  - New technique:
    Updated resampling method.

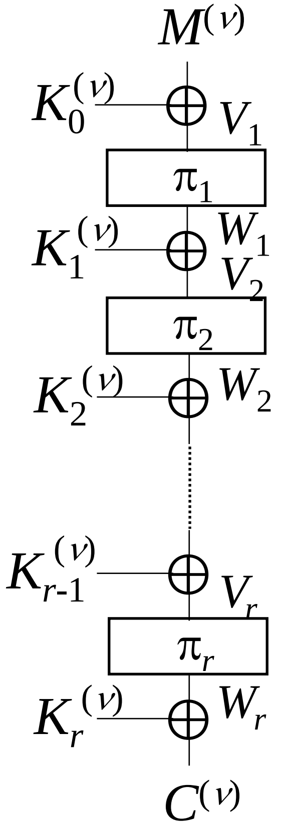
Reference	Round w/ Tight Bound	Identical Permutation	Independent Subkeys†	Multi-User Security	Tweakable KAC
Even-Mansour [12]	1	N/A	All	—	—
Bogdanov et al. [3]	2	—	All	—	—
Steinberger [24]	3	—	All	—	—
Lampe et al. [16]	Asymptotic	—	All	—	—
Chen-Steinberger [5]	Any	—	All	—	—
Chen et al. [4]	2	✓	1	—	—
Wu et al. [27]	3	✓	All	—	—
Yu et al. [28]	Any	✓	All	—	—
Dunkelman et al. [10]	1	N/A	1	—	—
Tessaro-Zhang [25]	Any	—	$r-1$	—	—
Mouha-Luykx [19]	1	N/A	1	✓	—
Hoang-Tessaro [14]	Any	—	All	✓	—
Cogliati et al. [7]	2	—	2	—	✓
Cogliati et al. [7]	Asymptotic	—	r	—	✓
Cogliati-Seurin [8]	4	—	2	—	✓
Dutta [11]	4	—‡	2	—	✓
This Work	Any	✓	r	✓	✓

Coefficient–H Technique

- Consider a transcript τ:
  information that an adversary obtains by queries such as $(M^{(\nu)}, C^{(\nu)})$, $(X^{(i)}, Y^{(i)})$, etc.



- Derive a security bound by the following steps.
  1. Bad events on transcripts τ.
  2. Split all possible transcripts τ into bad transcripts τ_{bad} and good transcripts τ_{good} from the bad events.
  3. Security bound = sum of the following bounds.
     - Upper–bound of Pr[one of the bad events occur in the ideal world].
     - Lower–bound of the ratio for good transcripts:
       Pr[Real–world sampling = τ_{good}]/Pr[Ideal–world sampling = τ_{good}] for any τ_{good}.



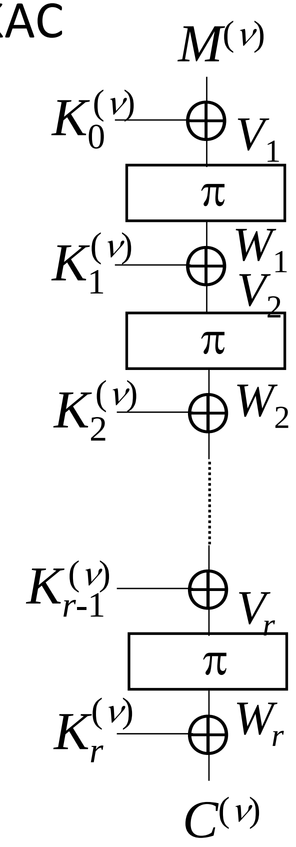
- Difficult step: evaluating the real–world probability for good transcript tightly.
  - Count the number of solutions of the internal pairs $(V_1,W_1),\ldots,(V_r,W_r)$ for each $(M^{(\nu)}, C^{(\nu)})$.
  - The number of the solutions drastically increases according to r.
  - The evaluation is quite complex for large r.

- Following the approach for good transcript is not reasonable.

Our Approach

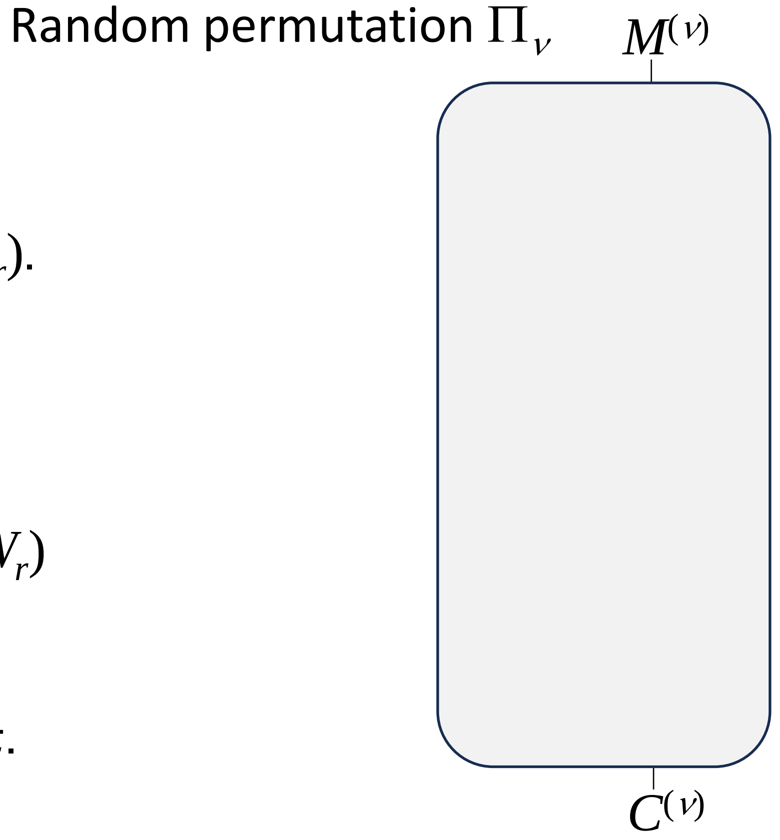
- We fix the game so that the internal pairs are introduced in τ.
- The internal pairs for each $(M^{(\nu)}, C^{(\nu)})$ are uniquely fixed.
- We don't need to count the number of solutions of $(V_1,W_1),\ldots,(V_r,W_r)$.
- The evaluation for good transcripts becomes simpler.

KAC



Our Approach

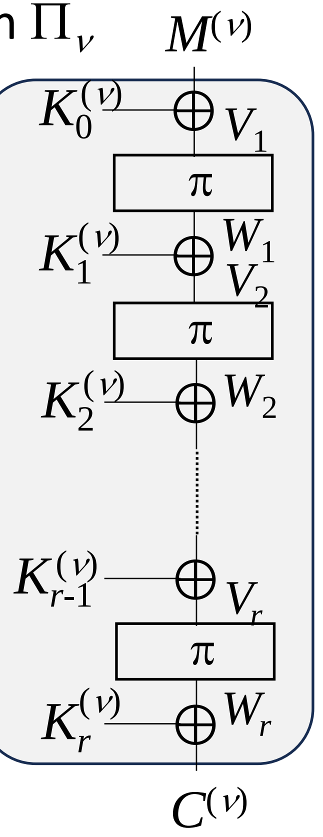
- We fix the game so that the internal pairs are introduced in τ.
- The internal pairs for each $(M^{(\nu)}, C^{(\nu)})$ are uniquely fixed.
- We don't need to count the number of solutions of $(V_1,W_1),\ldots,(V_r,W_r)$.
- The evaluation for good transcripts becomes simpler.
- Since random permutations Π_ν are monolithic in the ideal world, in order to introduce the internal pairs in the transcript τ,
    - Define dummy keys $K_0^{(\nu)},\ldots, K_r^{(\nu)}$ and internal pairs $(V_1,W_1),\ldots,(V_r,W_r)$ according to the structure of KAC with a single permutation π,.
    - Reveal the (dummy) keys and internal pairs to an adversary, i.e., the keys and internal pairs are introduced in the transcript τ.



Our Approach

- We fix the game so that the internal pairs are introduced in τ.
- The internal pairs for each $(M^{(\nu)}, C^{(\nu)})$ are uniquely fixed.
- We don't need to count the number of solutions of $(V_1,W_1),\ldots,(V_r,W_r)$.
- The evaluation for good transcripts becomes simpler.
- Since random permutations Π_ν are monolithic in the ideal world, in order to introduce the internal pairs in the transcript τ,
  - Define dummy keys $K_0^{(\nu)},\ldots, K_r^{(\nu)}$ and internal pairs $(V_1,W_1),\ldots,(V_r,W_r)$ according to the structure of KAC with a single permutation π,.
  - Reveal the (dummy) keys and internal pairs to an adversary, i.e., the keys and internal pairs are introduced in the transcript τ.

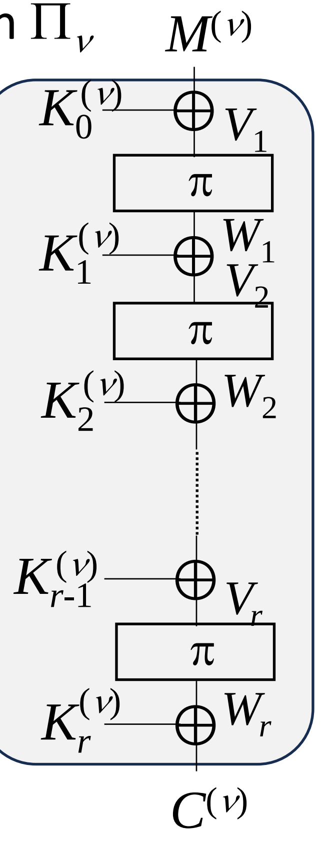
Random permutation Π_ν



Our Approach

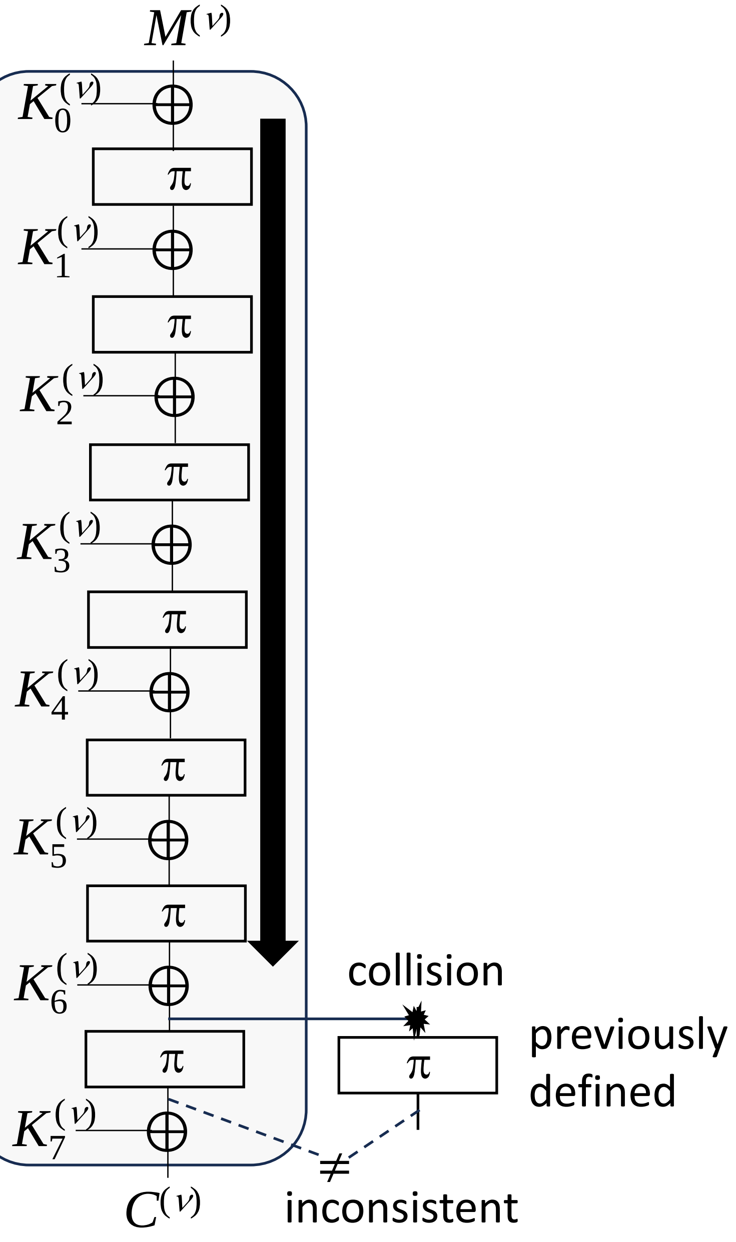
- We fix the game so that the internal pairs are introduced in τ.
- The internal pairs for each $(M^{(\nu)}, C^{(\nu)})$ are uniquely fixed.
- We don't need to count the number of solutions of $(V_1,W_1),\ldots,(V_r,W_r)$..
- The evaluation for good transcripts becomes simpler.
- Since random permutations Π_ν are monolithic in the ideal world, in order to introduce the internal pairs in the transcript τ,
  - Define dummy keys $K_0^{(\nu)},\ldots, K_r^{(\nu)}$ and internal pairs $(V_1,W_1),\ldots,(V_r,W_r)$ according to the structure of KAC with a single permutation π,.
  - Reveal the (dummy) keys and internal pairs to an adversary, i.e., the keys and internal pairs are introduced in the transcript τ.
- The remaining step is defining a sampling method of the dummy keys and dummy internal pairs in the ideal world.

Random permutation Π_ν



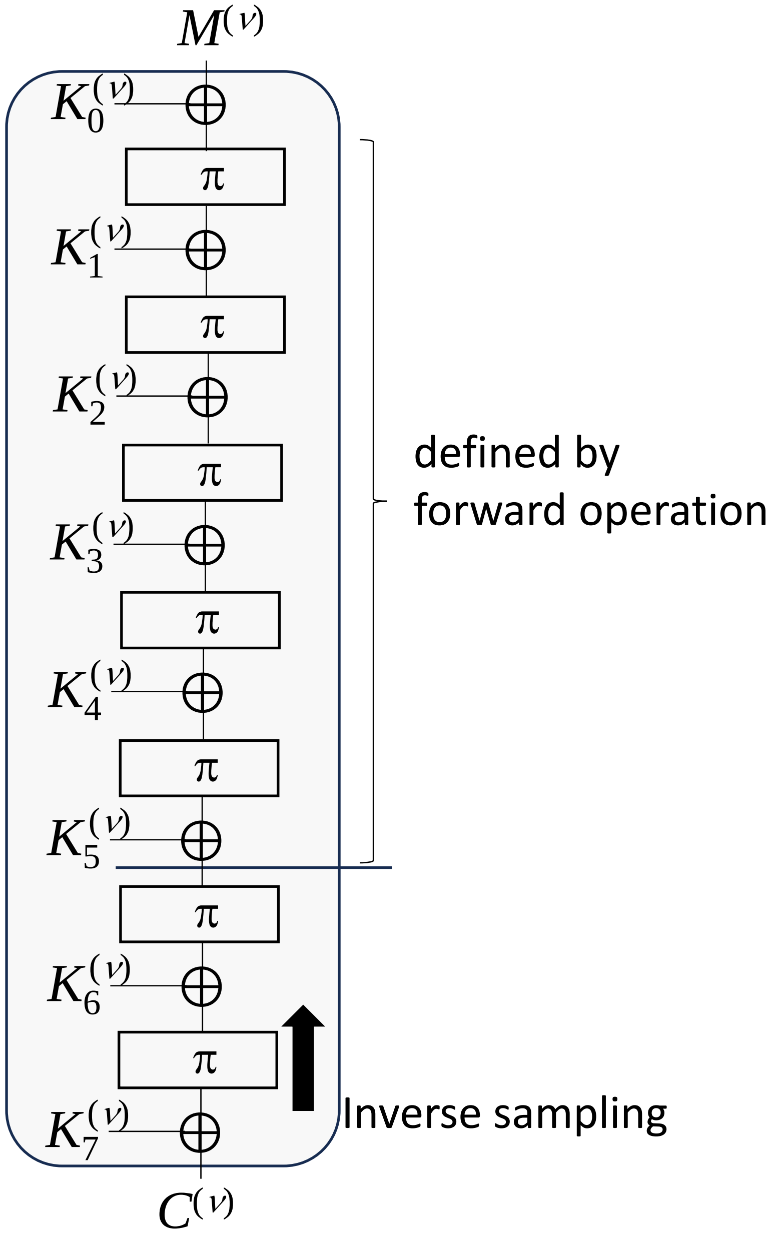
Sampling Method

- Naive sampling method (forward sampling).
  - Define the internal pairs from the first round to the last round.
  - The sampling successes if all the pairs are consistent with respect to random permutation, i.e., for each input (resp. output), there is no distinct outputs (resp. inputs).
  - The failure event is the inconsistent event: a collision occurs at the last round, i.e., the last-round input collides with the other pair.
  - The failure probability is the birthday bound n/2 bits, i.e., security up to n/2 bits (not tight).



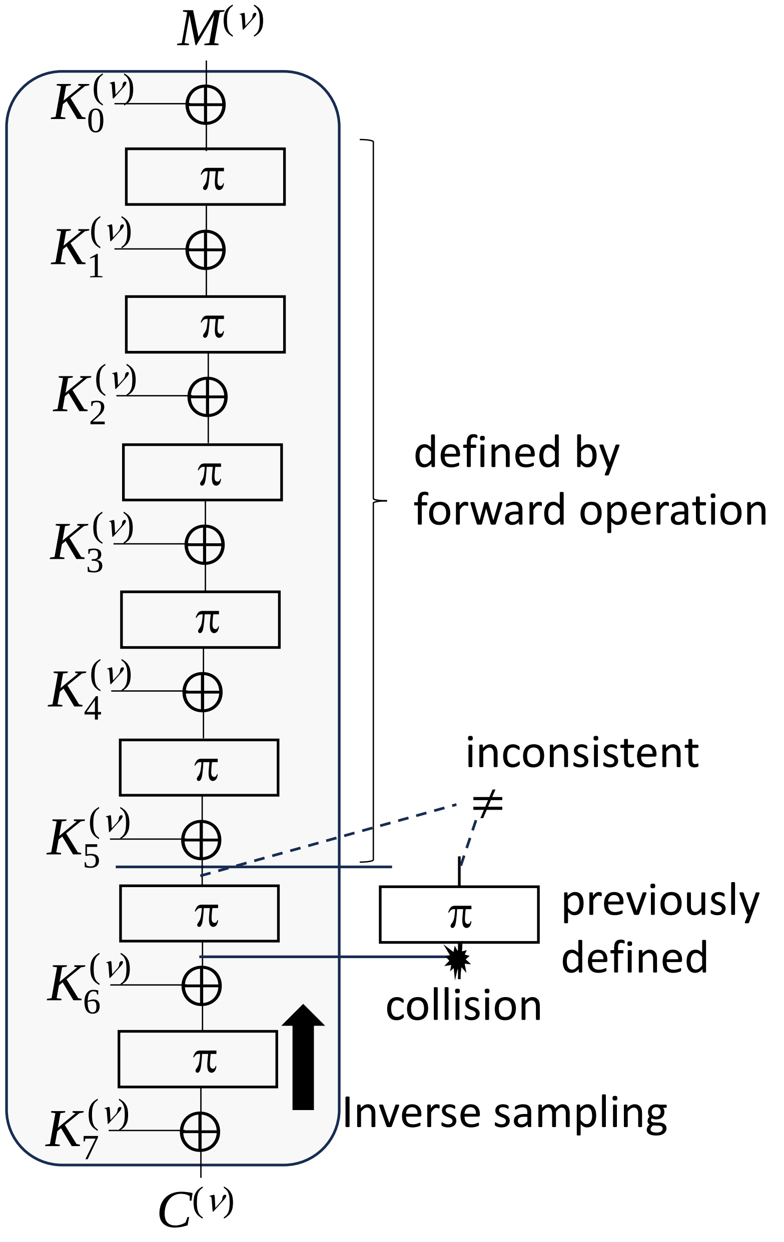
Sampling Method

- Resampling method (Naito et al. CCS 2022).
  - Sampling method for Triple encryption.
  - Inverse sampling is introduced:
    If the forward sampling fails (the pairs up to r−2 round are defined), the last-round pair is re-defined by the inverse sampling.



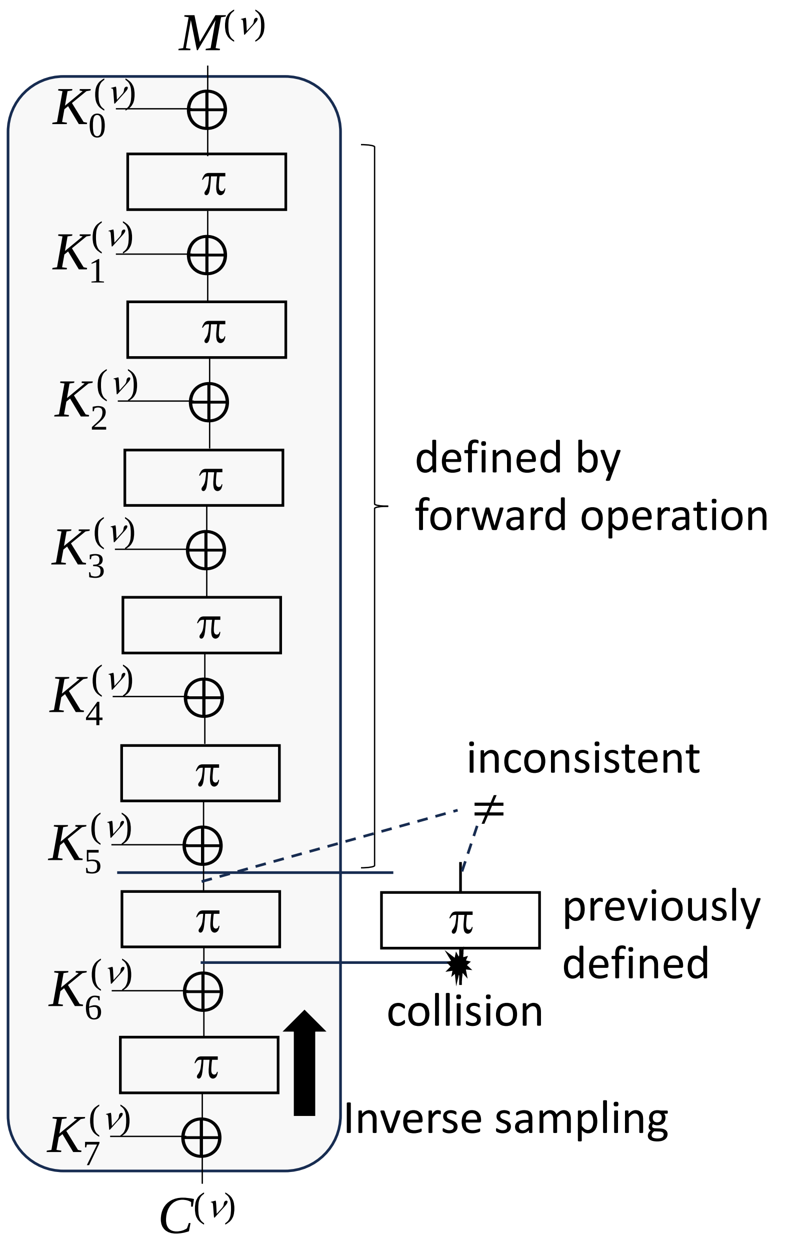
Sampling Method

- Resampling method (Naito et al. CCS 2022).
  - Sampling method for Triple encryption.
  - Inverse sampling is introduced:
    If the forward sampling fails (the pairs up to r−2 round are defined), the last-round pair is re-defined by the inverse sampling.
  - If no collision occurs in the inverse sampling, then all the internal pairs can be consistently defined.



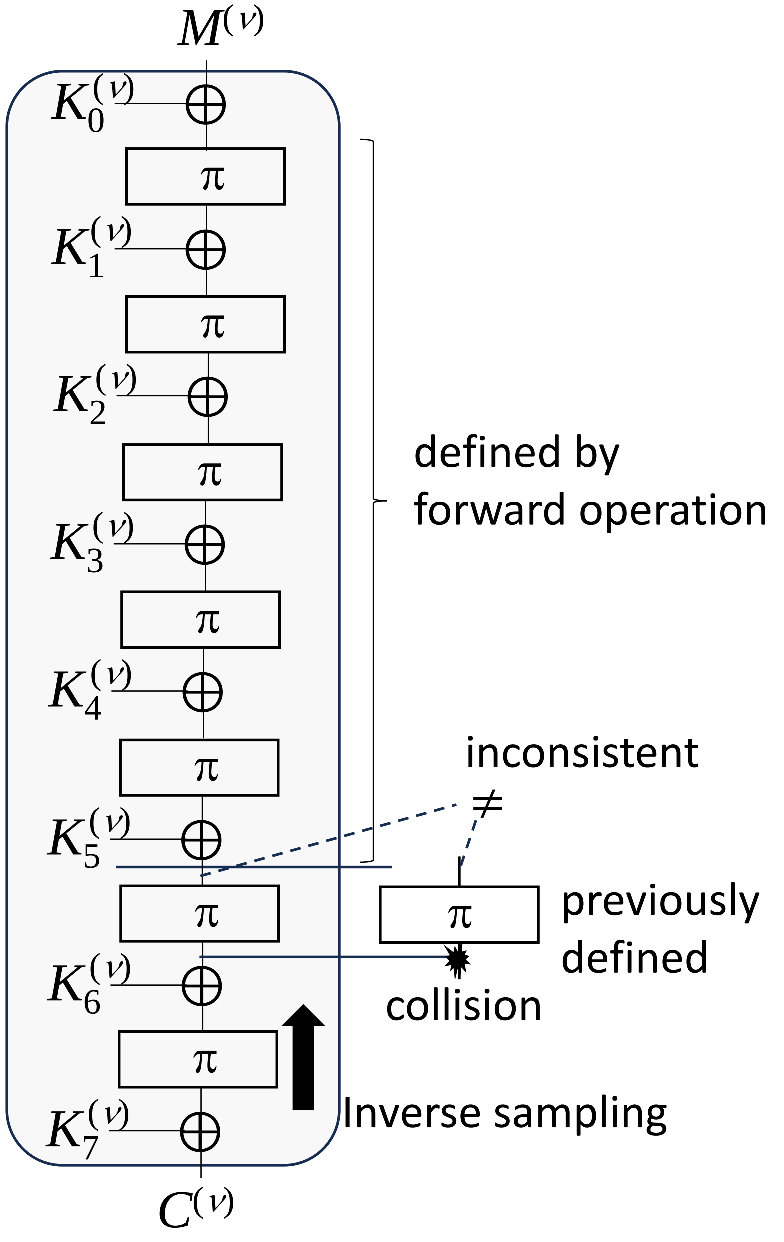
Sampling Method

- Resampling method (Naito et al. CCS 2022).
  - Sampling method for Triple encryption.
  - Inverse sampling is introduced:
    If the forward sampling fails (the pairs up to r−2 round are defined),
    the last-round pair is re-defined by the inverse sampling.
  - If no collision occurs in the inverse sampling, then
    all the internal pairs can be consistently defined.
  - The failure event of the resampling method is that collisions occur in both the forward and inverse samplings.
  - The security from the two collisions is 2n/3 bits (not tight).



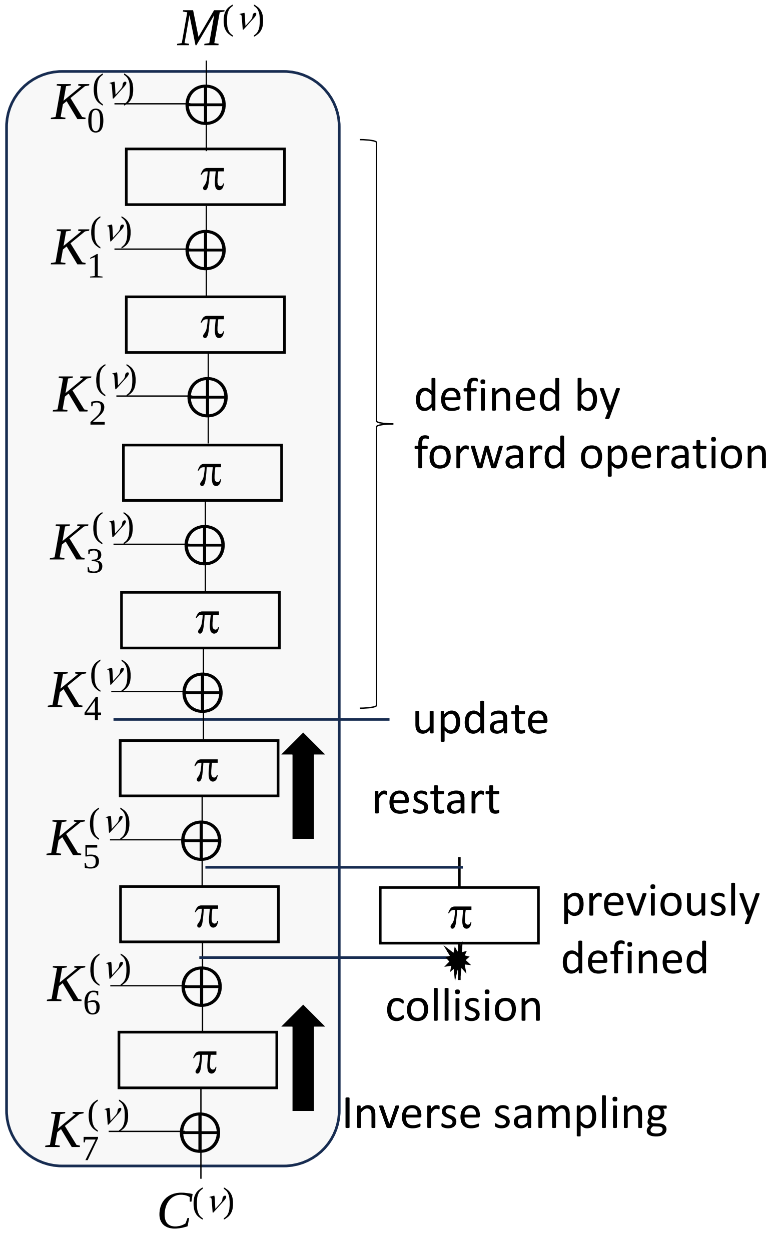
Our Resampling Method

- Update the resampling method to achieve the tight mu-bound rn/(r+1) bits.
- We update the inverse sampling as follows.
  - If a collision occurs in the inverse sampling, then the rounds defined by the forward sampling are updated.
  - The inverse sampling is restarted from the updated round.
  - The updates are allowed up to the first round.
- The updated resampling method tolerates the collisions multiple times.
- The probability of the multiple collisions is rn/(r+1) bits.
- The updated resampling method can consistently define the internal pairs up to the tight mu-bound rn/(r+1) bits.



Our Resampling Method

- Update the resampling method to achieve the tight mu-bound rn/(r+1) bits.
- We update the inverse sampling as follows.
  - If a collision occurs in the inverse sampling, then the rounds defined by the forward sampling are updated.
  - The inverse sampling is restarted from the updated round.
  - The updates are allowed up to the first round.
- The updated resampling method allows the multiple collisions.
- The probability of the multiple collisions is rn/(r+1) bits.
- The updated resampling method can consistently define the internal pairs up to the tight mu-bound rn/(r+1) bits.



Conclusion

- We consider the security of r-round key alternating cipher (KAC).
- Existing works for r-round KAC
  - Tight single-user security for KAC with a single random permutation.
  - Tight multi-user security for KAC
    - with r random permutations,
    - with r+1 independent subkeys.
- We prove the tight muti-user security of any round KAC
  - with a single random permutation,
  - with r-wise independent subkeys.
- We present the updated resampling method.
- Our result offers the tight multi-user security of tweakable KACs.

Reference	Round w/ Tight Bound	Identical Permutation	Independent Subkeys†	Multi-User Security	Tweakable KAC
Even-Mansour [12]	1	N/A	All	—	—
Bogdanov et al. [3]	2	—	All	—	—
Steinberger [24]	3	—	All	—	—
Lampe et al. [16]	Asymptotic	—	All	—	—
Chen-Steinberger [5]	Any	—	All	—	—
Chen et al. [4]	2	✓	1	—	—
Wu et al. [27]	3	✓	All	—	—
Yu et al. [28]	Any	✓	All	—	—
Dunkelman et al. [10]	1	N/A	1	—	—
Tessaro-Zhang [25]	Any	—	$r-1$	—	—
Mouha-Luykx [19]	1	N/A	1	✓	—
Hoang-Tessaro [14]	Any	—	All	✓	—
Cogliati et al. [7]	2	—	2	—	✓
Cogliati et al. [7]	Asymptotic	—	r	—	✓
Cogliati-Seurin [8]	4	—	2	—	✓
Dutta [11]	4	—‡	2	—	✓
This Work	Any	✓	r	✓	✓