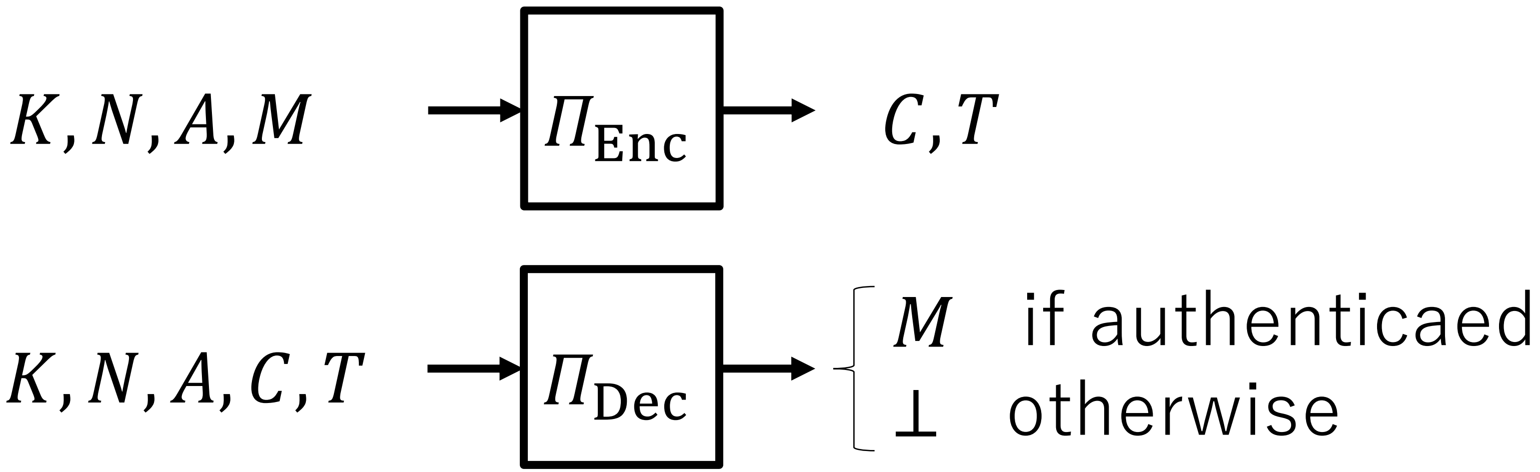
KIVR: Context-Committing Authenticated Encryption Using Plaintext Redundancy and Application to GCM and Variants

Yusuke Naito (Mitsubishi Electric Corporation)
Yu Sasaki (NTT Social Informatics Laboratories)
Takeshi Sugawara (The University of Electro-Communications)

Authenticated Encryption with Associated Data



- Security of AE is well studied.
- The security of AE schemes is usually proved with formal security notions.
- However, AE schemes are sometimes misused or abused beyond their promise.

Key Commitment

- Farshim et al. initiated the theoretical study in 2017, followed by the real-world attacks.
  - **Multi-recipient integrity attack** (delivering malicious content to a target user)
  - **Partitioning oracle attacks** (achieving faster password brute-force attack)

- Without key commitment, an adversary can efficiently find a ciphertext decrypted with multiple keys:

$$\Pi_{Enc}(K, N, A, M) = \Pi_{Enc}(K', N, A, M) \text{ with } K \neq K'$$

- Conventional AE security notions do not support the key commitment.

- $O(1)$ attacks are known for GCM, GCM-SIV, CCM, ChaCha20-Poly1305.

Generalization: Context Commitment

- In 2022, Bellare-Hoang introduced generalization of key commitment called context commitment.

- **Key commitment (CMT-1)**: K is different.

$$\Pi_{\mathrm{Enc}}(K, N, A, M) = \Pi_{\mathrm{Enc}}(K', N', A', M') \text{ with } K \neq K'$$

- **Context commitment (CMT-4)**: different values can be located in any of K, N, A, M.

$$\Pi_{\mathrm{Enc}}(K, N, A, M) = \Pi_{\mathrm{Enc}}(K', N', A', M') \text{ with } (K, N, A, M) \neq (K', N', A', M')$$

- CMT-4 guarantees more robust security than CMT-1.
- AE with CMT-4 security is an ongoing research challenge.

Research Directions

There are two possible research directions.

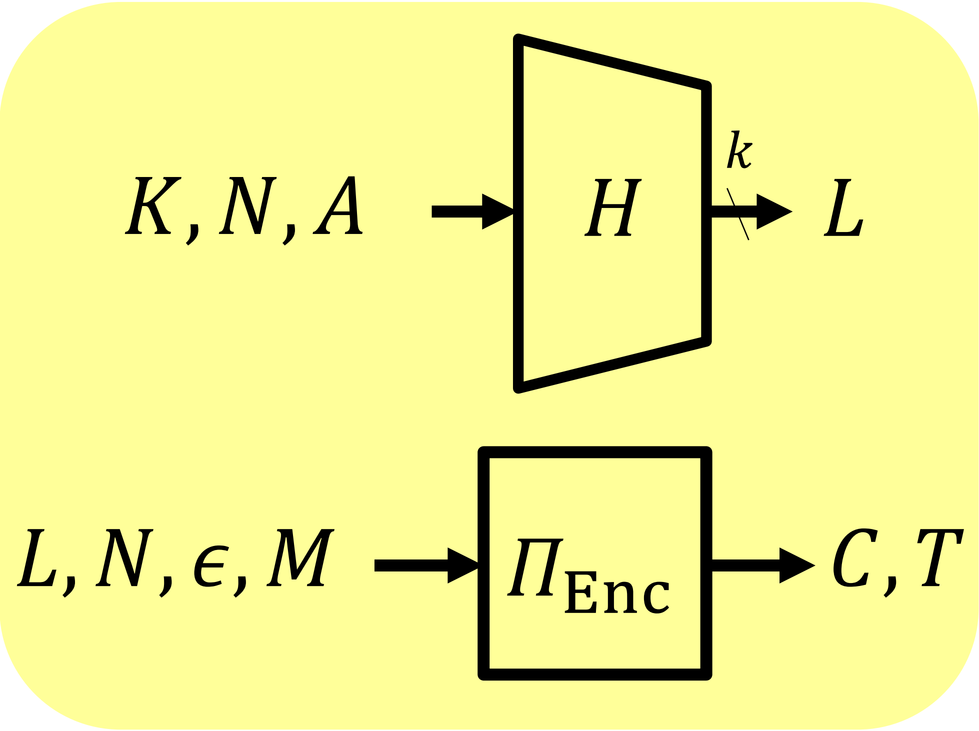
1. Designing a dedicated scheme with committing security.

2. Extending conventional AEs for committing security.

We are taking the second approach.
Particularly, we want to salvage GCM to provide CMT-4 security.

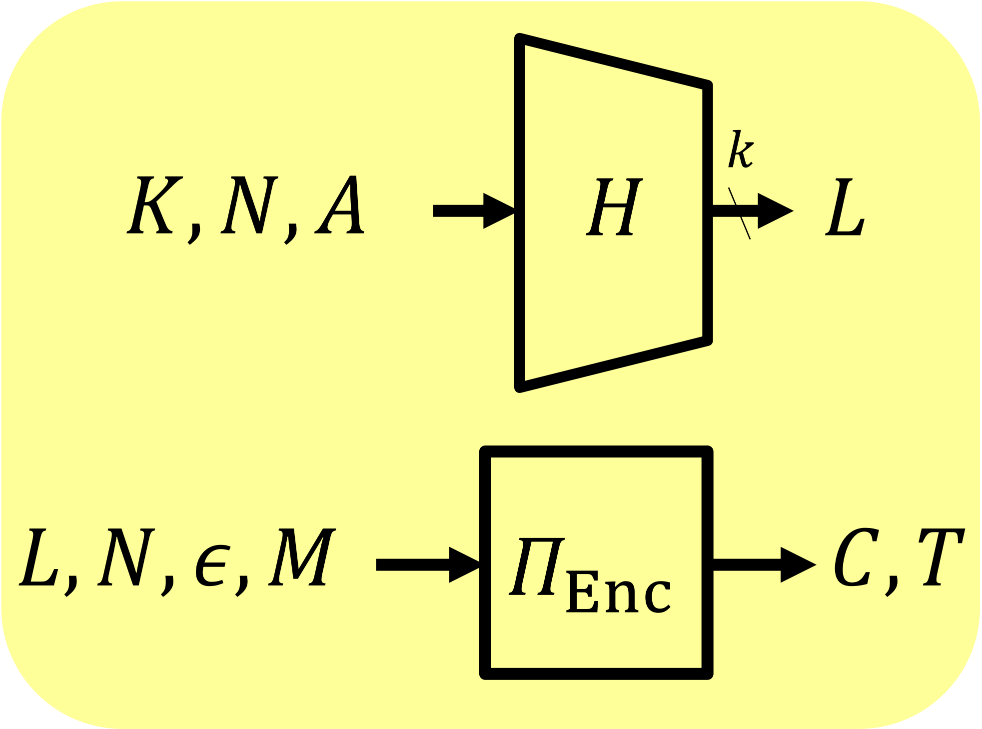
Previous Work: Hash-then-Enc (HtE) [BH22]

- HtE generates a temporary key L by a collision-resistant hash H, then compute AE by using L as a key.
- HtE converts CMT-1 secure AE to CMT-4 secure AE.
- Generic conversions: UtC and RtC
  - from any AE to CMT-1 secure AE
  - with ciphertext expansion (ciphertext size is increased).
- By using both, any AE can be converted to CMT-4 secure AE with ciphertext expansion.



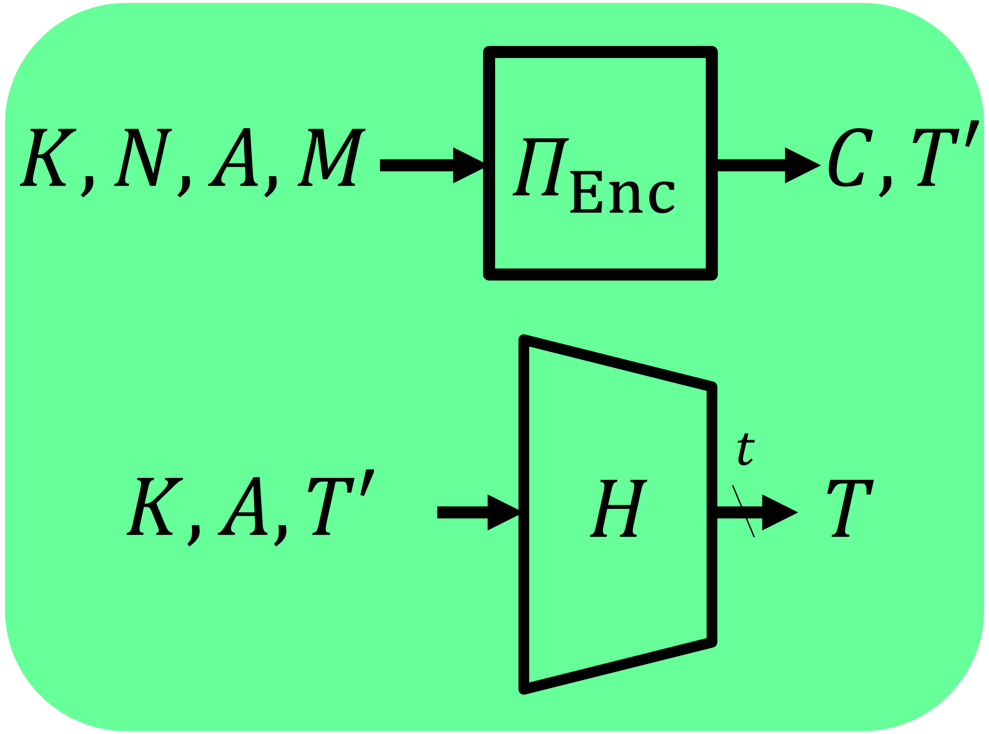
Previous Work: Hash-then-Enc (HtE) [BH22]

- GCM cannot be salvaged with HtE without ciphertext expansion (UtC and RtC).
- Modify GCM to be CMT-1 secure.
- Two CMT-1 variants
  - CAU-C1 (a variant of GCM)
  - CAU-SIV-C1 (a variant of GCM-SIV)
- GMAC is modified e.g.
  - by adding the feed-forward or
  - by changing the position of XOR of hash value.



Previous Work: CTX [CR22]

- A hash function is applied to the tag T' of AE, and the hash value is a tag of the CTX-based AE.
- Verification is done with T.
- CTX converts any AE to CMT-4 secure AE.



Our Goals

Design a CMT-4 conversion with the following goals

- **Construct CMT-4 secure AE for the following classes of CTR mode-based AEs**
  - CTRAE: Enc-then-MAC scheme (including GCM and CAU-C1)
  - CTRSIV: SIV paradigm (including GCM-SIV and CAU-SIV-C1)

- **Avoid ciphertext expansion**
  - The ciphertext size should be preserved to maintain compatibility with the hardware, database, or communication protocol, already deployed.

- **Beyond-the-Birthday-Bound (BBB) Security for Key Size**
  - Commitment is an offline security, i.e., there is no secret and adversaries choose key values.
  - Offline complexity of standard AE security is k bits.
  - Hence, we aim at least greater than $k/2$-bit security for committing security.

Limitation for Committing Security (Generic Attack)

- Consider a class of AEs s.t. AD A affects the tag generation but does not affect the message/plaintext conversion, such as GCM.
  - $C = Enc(K, N, M)$
  - $T = Tag\ (K, N, A, C)$

- The birthday attack with distinct AD A breaks the CMT-4 security
  - Changing AD A and fixing the other inputs (K, N, M)
  - $\Pi_{Enc}(K, N, A, M) \ = \ \Pi_{Enc}(K', N, A, M)$ with $A \neq A'$
  - Complexity: $2^{\frac{t}{2}}$, where t is the tag size, usually smaller than or equal to the key size.

- Without some special features, our goals cannot be achieved.

Our Approach

- We make use of the plaintext contains redundancy to salvage GCM.

Ex. A plaintext in HTTP starts with "**HTTP/1.1**" which can be used as 8-byte redundancy.

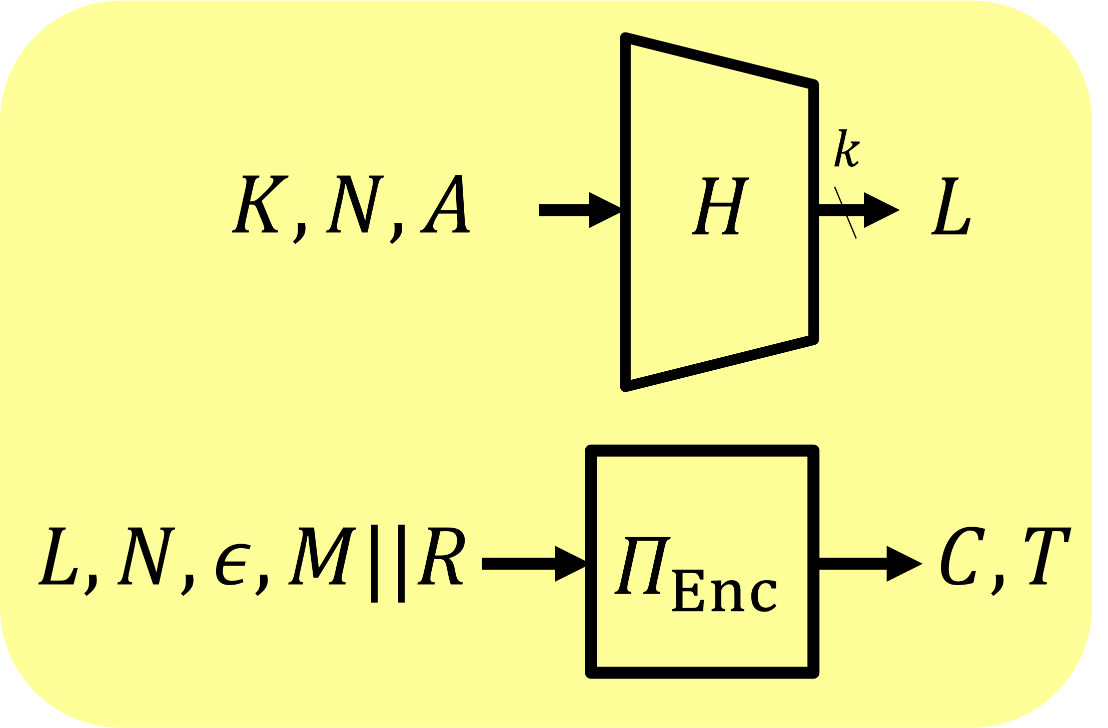
- The redundant part is known to recipients who decrypts the ciphertext, thus can be used as another source of integrity check.
- This is a natural extension of [ADG+22] that design a conversion to CMT-1 security by using the zero padding $M||0\ldots0$.
  - The zero padding expands the ciphertext length.
- For protocols with redundancy in plaintexts, we can enhance the security without ciphertext expansions by adding redundancy.

Existing Conversion + Plaintext redundancy

HtE + Plaintext Redundancy

- Change AD A and fix the other inputs
- We can find a collision of L with 2^{k/2} complexity
- The collision on L yields the collision $\mathrm{HtE}(\Pi_{Enc})(K, N, A, M) = \mathrm{HtE}(\Pi_{Enc})(K', N, A, M)$ with $A \neq A'$.
- Committing security is not enhanced from $k/2$ bits.

HtE

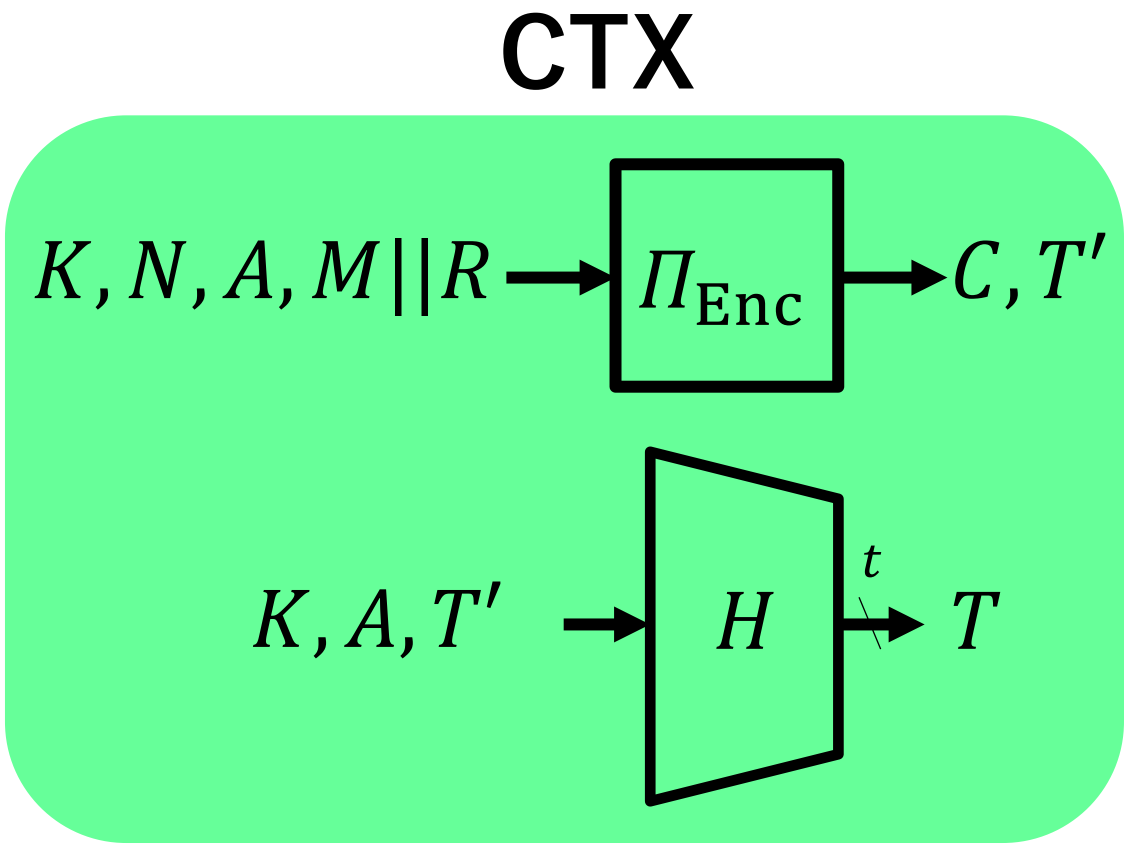


Existing Conversion + Plaintext Redundancy

CTX + Plaintext redundancy

- We consider CTRAE including GCM.
  - $C = CTR(K, N, M)$
  - $T = Tag\ (K, N, A, C)$
- Change A and fix the other inputs.
- A collision on T with 2^{t/2} complexity

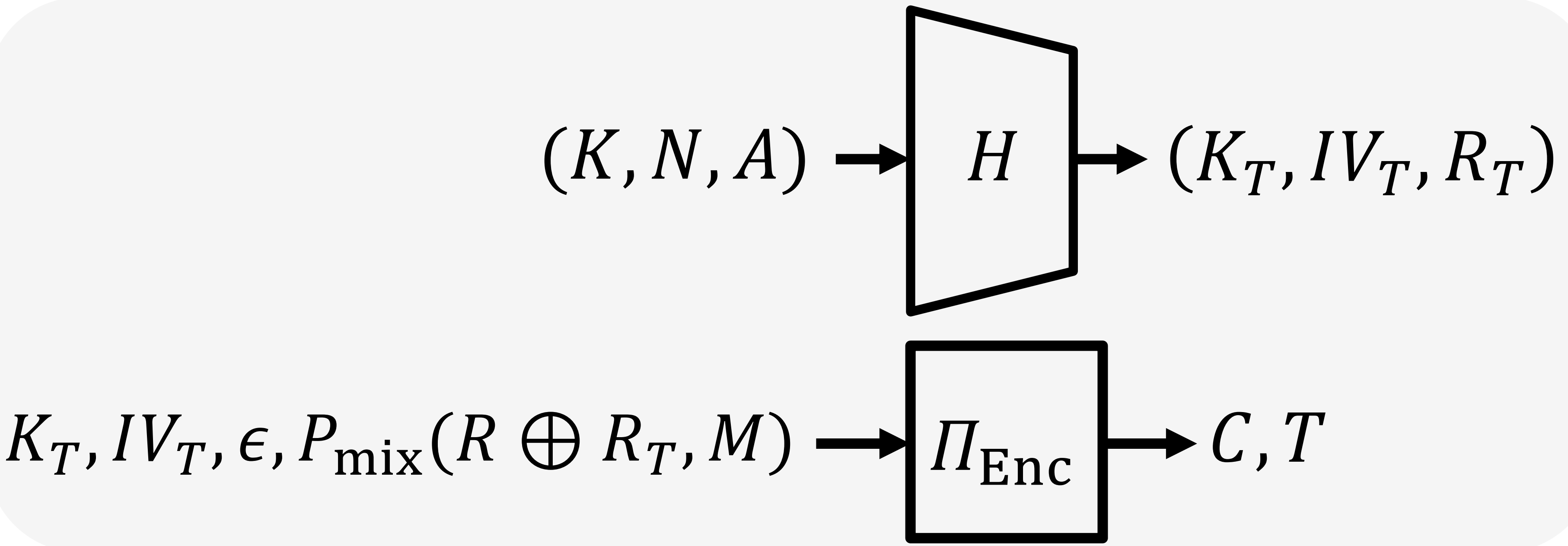
  ⇨
  $\mathrm{HtE}(\Pi_{Enc})(K, N, A, M) = \mathrm{HtE}(\Pi_{Enc})(K', N, A, M)$ with $A \neq A'$.
- The committing security cannot be enhanced from $t/2$ bits, and usually $t \leq k$.



Our Design: New Conversion KIVR

Generalization of **HtE** + **redundancy**.

1. Generate temporal data: $(K_T, IV_T, R_T) \leftarrow H(K, N, A)$
2. Extract redundant data: R
3. The redundant data R is masked as $R \oplus R_T$,
4. Perform the original AE with a key K_T, a nonce IV_T, and the masked redundancy $R \oplus R_T$ as a plaintext.



CMT-4 Security for KIVR

- We prove the CMT-4 security of KIVR with CTRAE and CTRSIV and wtih plaintext redundancy.
- Let $tagcol$ be security for tag-collision attacks by changing K, N, A.
- Let $r = |R|$ be the length of redundancy.
- CMT-4 security of CTRAE: $\max\{\frac{r}{2}, tagcol\}$
- CMT-4 security of CTRSIV: $\frac{r}{2} + tagcol$
- $tagcol$:
  - GCM and GCM-SIV: $tagcol = 0$
  - CAU-C1 and CAU-SIV-C1: $tagcol = \frac{t}{2}$

Comparison with Parameter of GCM

Table 2. CMT-4 security of the instantiations with r-bit redundancy.

Conversion	AE	CMT-4 Security w/ $k = 128$	Ref.
CTX [6] + Redundancy	GCM, CAU-C1	64	Prop. 3†
HtE [4]+ Redundancy	GCM	$\min\{\frac{r}{2}, 64\}$	Prop. 5†
HtE [4]+ Redundancy	CAU-C1	64	Prop. 6†
HtE [4]+ Redundancy	GCM-SIV	$\min\{\frac{r}{2}, 64\}$	Prop. 7†
HtE [4]+ Redundancy	CAU-SIV-C1	64	Prop. 8†
KIVR + Redundancy	GCM	$\frac{r}{2}$	Cor. 2‡
KIVR + Redundancy	CAU-C1	$\max\{\frac{r}{2}, 64\}$	Cor. 3‡
KIVR + Redundancy	GCM-SIV	$\frac{r}{2}$	Cor. 5‡
KIVR + Redundancy	CAU-SIV-C1	$\frac{r}{2} + 64$	Cor. 6‡

- If sufficiently large redundancy is available, KIVR-based schemes achieve BBB-security for the key size.
- For XML and HTTP2 with $r = 192$, KIVR with GCM achieves 96-bit CMT-4 security.
- For PNG and HTTP with $r = 64$, KIVR with CAU-SIV-C1 achieves 96-bit CMT-4 security.

Conclusion

- We propose a new mode KIVR
  - transforms existing AEs to have CMT-4 security
  - without increasing the ciphertext size
  - by exploiting plaintext redundancy found in practical use cases.

- KIVR uses a collision-resistant hash to convert a tuple of (K, N, A) into (K_T, IV_T, R_T), and use them as a key and IV of an underlying AE and the mask value for the redundant data.

- Security of KIVR linearly increases with the number of redundant bits r and can achieve the BBB security for key size with a sufficiently large r.

Thank you for your attention.