RSAConference2020

San Francisco | February 24 – 28 | Moscone Center

HUMAN ELEMENT

SESSION ID: **CRYP-T08**

Universal Forgery Attack against GCM-RUP

Yanbin Li[1], Gaëtan Leurent[2], Meiqin Wang[1], Wei Wang[1], Guoyan Zhang[1], Yu Liu[1]

[1] Shandong University, China
[2] Inria, France

Presented by Ferdinand Sibleyras

Ph.D. Student
Inria, France

RSAConference2020

Universal Forgery Attack against GCM-RUP

Yanbin Li, Gaëtan Leurent, Meiqin Wang, Wei Wang, Guoyan Zhang, Yu Liu

Outline

- About GCM-RUP
- Motivation and Contributions
- Brief Description of GCM-RUP
- Partial Authentication Key Recovery for GCM-RUP
- Universal Forgery Attack of GCM-RUP
- Variant of GCM-RUP

About GCM-RUP

About GCM-RUP

- GCM (Galois/Counter Mode)
  - Authenticated Encryption scheme following the Encrypt-then-MAC paradigm, proposed by Dworkin
  - Not robust against implementation errors or misuse
  - Lose its security if a device releases the plaintext corresponding to invalid ciphertext before verifying the tag
- GCM-RUP
  - Instantiation of the variant construction of GCM, proposed by Ashur *et al.*
  - Secure even in the releasing unverified plaintext (RUP) setting
  - Designers prove that GCM-RUP is secure up to the birthday bound in the nonce-respecting model

Motivation and Contributions

Motivation and Contributions

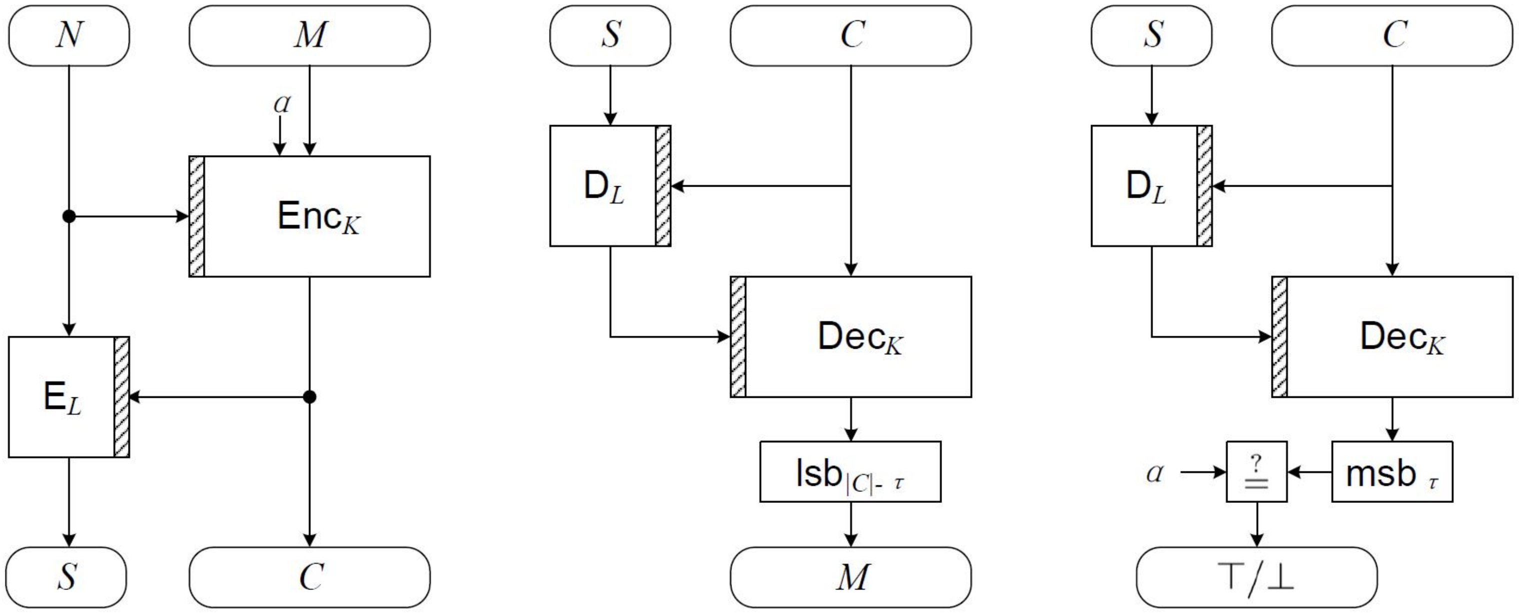
- Motivation
  - No attacks are known so far against the authentication part of GCM-RUP
  - Is the security proof of GCM-RUP tight?
  - What kind of security degradation to expect after the birthday bound
- Contributions
  - Partial key recovery by utilizing collision on inner states, leading to universal forgeries
  - Birthday-bound universal forgery attack against GCM-RUP, matching the security proof
  - Minor modification to GCM-RUP to avoid our attack

RSAConference2020

Brief Description of GCM-RUP

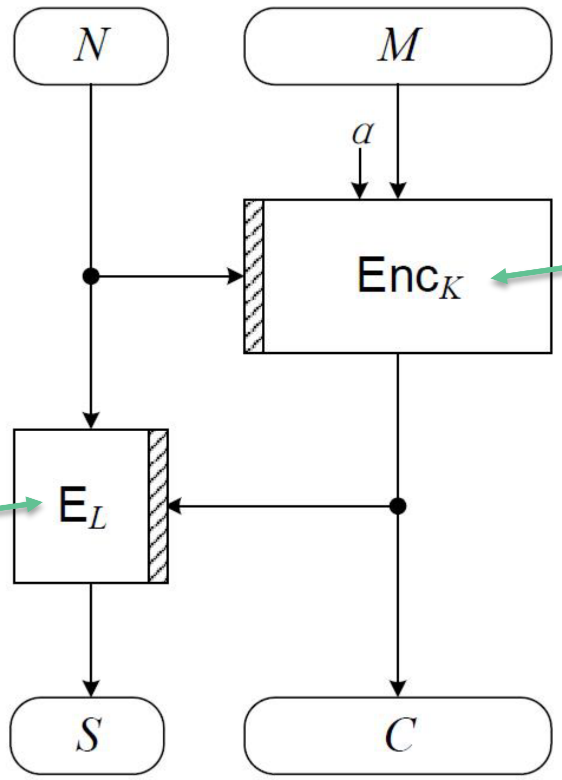
Generic Construction with RUP Security

- (Enc, Dec): encryption scheme (without authentication)
  - $\mathcal{K}$: key space; $\mathcal{N}$: nonce space; $\mathcal{M}$: message space; $\mathcal{C}$: ciphertext space.
- (E, D): TBC
  - key space $\mathcal{L}$, tweak space $\mathcal{T} = \mathcal{C}$, domain $\mathcal{X} = \mathcal{N}$.



(a) Encryption (b) Decryption (c) Verification

GCM-RUP



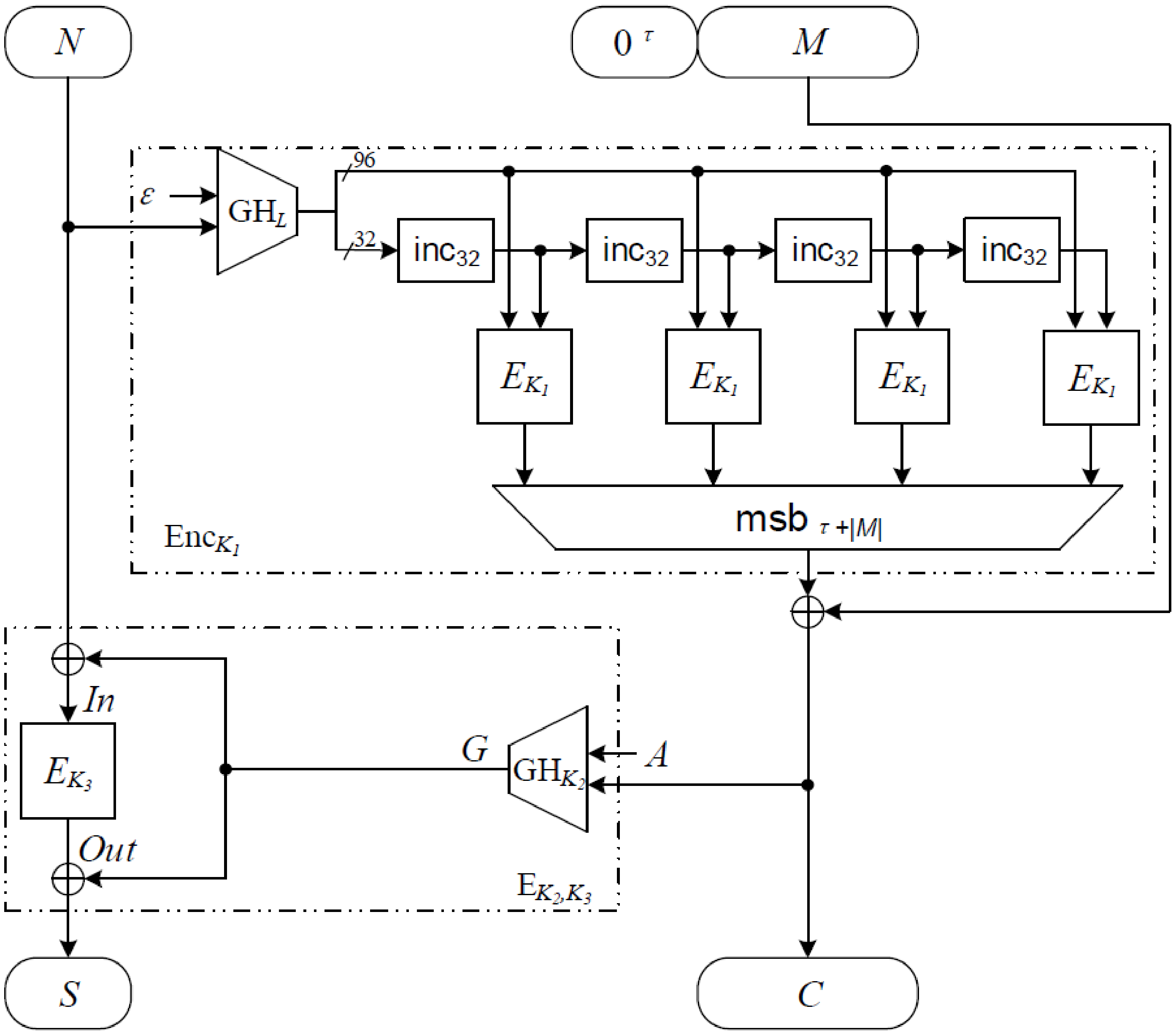
(a) Encryption

CTR for encryption

XTX construction with GHASH

GCM-RUP

- CTR for encryption
- XTX construction with GHASH for TBC



Universal Hash Function GHASH

- $GHASH_{K_2}(A, C)$ is defined by
$$GHASHcore_{K_2}(A \parallel C \parallel |A| \parallel |C|)$$
- Key K_2 and inputs A and C.
- Polynomial evaluation:
$$GHASHcore_{K_2}(x) = \bigoplus_{i=0}^{|x|_n - 1} x[i] \cdot K_2^{|x|_n - 1}$$

where x is a full-block string and the symbol "$\cdot$" represents multiplication in $GF(2^n)$.

RSAConference2020

Partial Authentication Key Recovery for GCM-RUP

Properties of GHASH

- Focus on the component $GHASH_{K_2}$ with inputs the associated data A and the ciphertext C.

$$
\begin{aligned}
G &= GHASH_{K_2}(A, C) \\
&= GHASHcore_{K_2}(A \parallel C \parallel |A| \parallel |C|)
\end{aligned}
$$

- G is linearly independent on the A and C for a fixed K_2.
- Hence, we consider the difference ΔG in the output of $GHASH_{K_2}$ for a pair of inputs.

Properties of GHASH

- Property 1.

If GCM-RUP is used to process a fixed associated data A and message M under two distinct nonces N_1 and N_2, the output difference of function $GHASH_{K_2}$ is only dependent on the nonces N_1 and N_2, but independent on A and M. This also holds for the input difference of E_{K_3}.

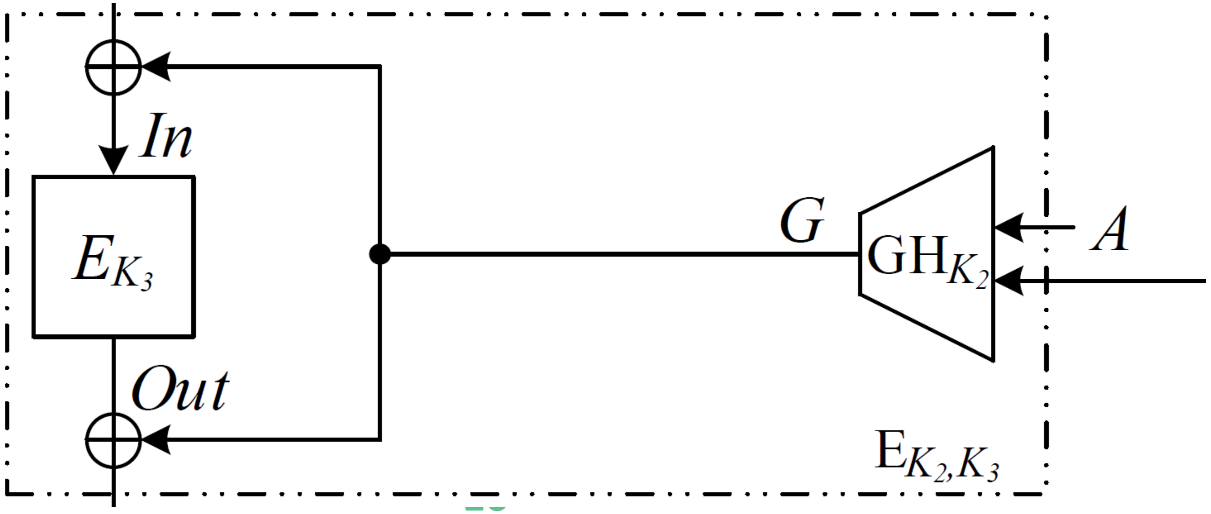
- So let $C_1 = M \oplus E_{K_1}(N_1)$ and $C_2 = M \oplus E_{K_1}(N_2)$:

$$\Delta G = GHASH_{K_2}(A, C_1) \oplus GHASH_{K_2}(A, C_2)$$
$$= GHASH_{K_2}(0, C_1 \oplus C_2)$$
$$= GHASH_{K_2}\left(0, E_{K_1}(N_1) \oplus E_{K_1}(N_2)\right)$$

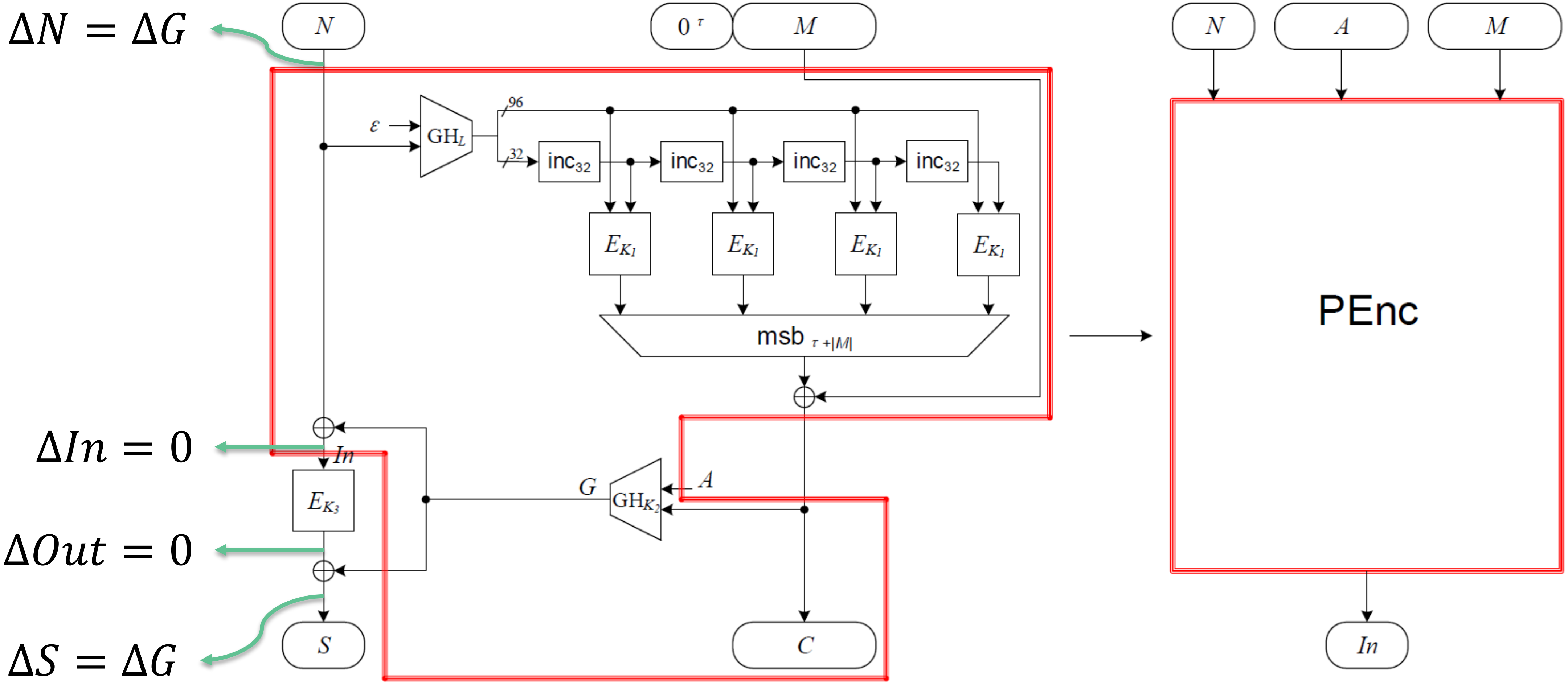
Recovering K_2 from Inner Collisions

Based on Property 1, we can retrieve the authentication key K_2 with the following two steps.

- For a fixed associated data A and M, search for a pair of nonces (N_1, N_2) which produce a collision for the input of E_{K_3} (*i.e.* inner collision) using a birthday attack.
- With a known $\Delta G = N_1 \oplus N_2$, a polynomial equation in K_2 is derived from the $GHASH_{K_2}$ definition. Then K_2 can be retrieved by solving this equation.



Find Inner Collisions



$$\Pr[\Delta In = 0 \mid N_1 \oplus N_2 = S_1 \oplus S_2] = 1/2$$

Find Inner Collisions

Number of nonces needed q is related to the probability of success p.

$$q \approx \sqrt{2 \times 2^{128} \times \ln(\frac{1}{1-p})}$$

Number of nonces to identify inner collision	Probability of finding inner collision
2^{63}	11%
2^{64}	39%
2^{65}	86%
2^{66}	99.9%

Universal Forgery Attack of GCM-RUP

Almost Universal Forgery Attack

Let $G = GHASH_{K_2}(A, C)$ and the key-stream used to XOR message is $E_{K_1}(N) = C \oplus M$.

- Query (N, A, M) and receive the ciphertext (S, C)
- Compute $C^* = M^* \oplus E_{K_1}(N)$
- Construct A' such that $GHASH_{K_2}(A', C^*) = GHASH_{K_2}(A, C)$, where A, C, C^* and K_2 are known

Universal Forgery Attack

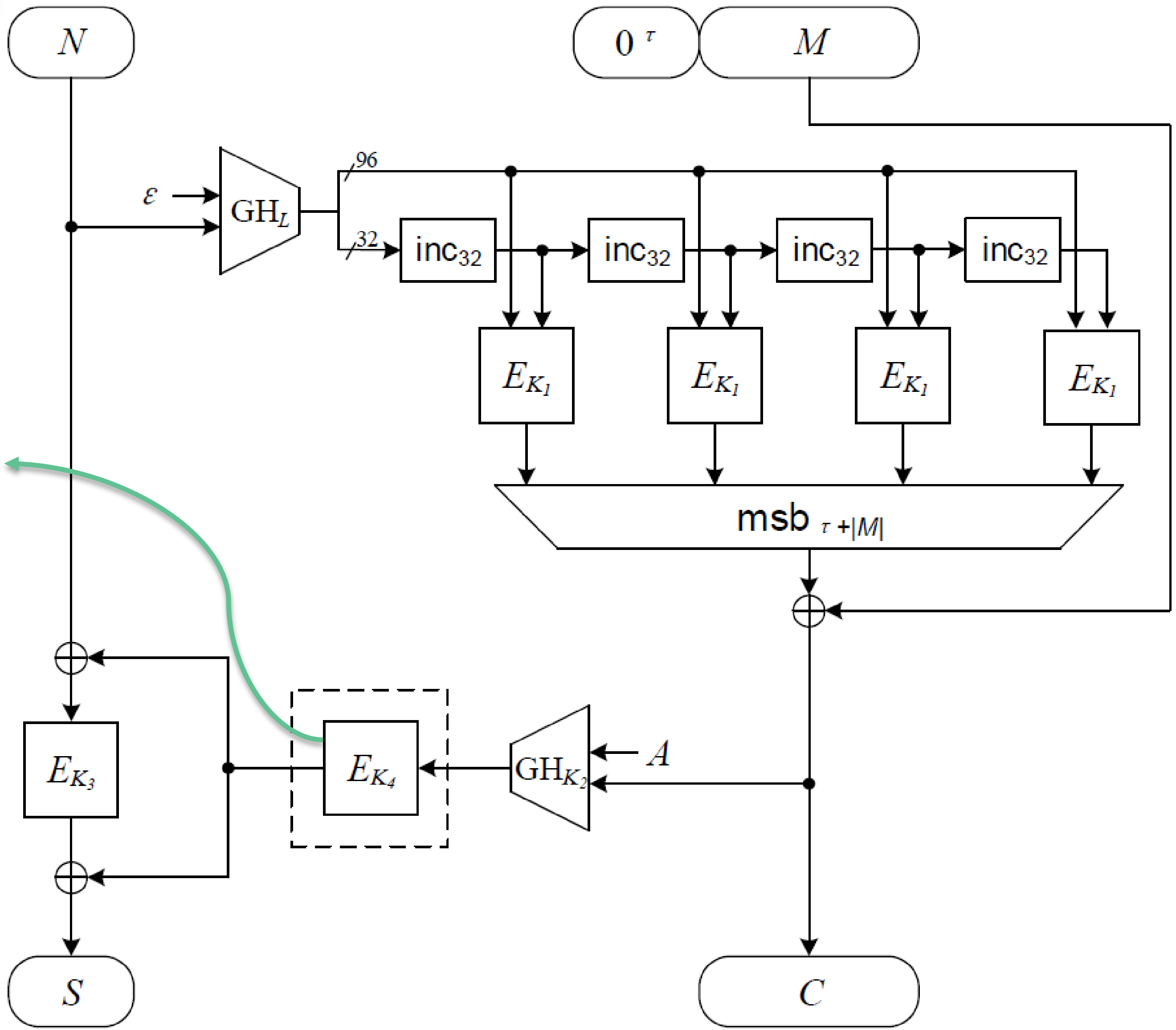
- Make $2^{n/2}$ queries (N_i, A, M) for fixed A and M with $|M| = |M^*|$, and receive the ciphertexts (S_i, C_i)
- Compute $G_i = GHASH(A, C_i)$ and receive inputs and outputs to E_{K_3}: $E_{K_3}(N_i \oplus G_i) = S_i \oplus G_i$
- For each N_i, build the corresponding C_i' from M^* and C_i as above
- Check whether $N_i \oplus GHASH(A^*, C_i')$ is in the set of known inputs to E_{K_3}
- If so, find $N_i = N_j$ satisfying $N_i \oplus GHASH(A^*, C_i') = N_j \oplus G_j$, and then we deduce a forgery using $S' = S_j \oplus G_j \oplus GHASH(A^*, C_i')$

RSAConference2020

Variant of GCM-RUP

A Variant of GCM-RUP to Avoid Our Attack

Avoid key leakage from known difference.



Conclusion

- Birthday-bound attack against authentication part of GCM-RUP.
- Bound is tight but drastic break at security bound, unlike GCM.
- Minor modification can avoid this attack.

If you have any question please contact Professor Meiqin Wang at

mqwang@sdu.edu.cn