Reconsidering Generic Composition

Chanathip Namprempre
Thammasat University, Thailand

Phillip Rogaway
University of California, Davis, USA

Tom Shrimpton
Portland State University, USA

DIAC 2013
Directions in Authenticated Encryption
Chicago, Illinois, USA
13 August 2013

Journey back in time to …

[Bellare-Namprempre – ASIACRYPT 2000]

Authenticated Encryption: Relations among Notion
and Analysis of the Generic Composition Paradigm

What did people learn from BN?

1. There are **three ways** to glue together a (privacy-only) **encryption scheme** and a **MAC** to make an AE scheme
   Encrypt-and-MAC **Encrypt-then-MAC** **MAC-then-Encrypt**

2. Of these, only **Encrypt-then-MAC** works well:
   it will **always** be secure (if the underlying primitives are sound),
   while this is untrue for the other two methods

Claim: <u>Not</u> a good summary of [BN]

Why not a good summary?

It doesn't mention **what definitions** BN use

probENC
- probabilistic ENC
- "total": {0,1}* domain
- IND-CPA secure

MAC
- a MAC (eg, a PRF)
- "total": {0,1}* domain
- strongly unforgeable

E&M
EtM
MtE

probAE
- probabilistic AE
- {0,1}* domain
- IND-CPA + INT-CTX

E&M: probENC + MAC $\not\rightarrow$ probAE
MtE: probENC + MAC $\not\rightarrow$ probAE
EtM: probENC + MAC $\rightarrow$ probAE

If you **change** the definitions, the **results might change** (duh…)

And **they do.**

Revised version

1. There are **three ways** to glue together a
   probENC scheme and a **MAC** to make a **probAE scheme :**
   Encrypt-and-MAC **Encrypt-then-MAC** **MAC-then-Encrypt**

2. Of these, only **Encrypt-then-MAC** works well:
   it will **always** be secure (if the underlying primitives are sound),
   while this is untrue for the other two methods

When you state it that way, BN doesn't seem so applicable

1) Standards don't directly provide probabilistic encryption schemes;
   they provide **IV-based encryption scheme** (**ivENC**)
   and not always total.
2) Conventional goal nowadays: **nonce-based AEAD scheme** (**nonceAEAD**)
3) And real-world schemes, like the TLS record protocol, don't respect either abstraction boundary

If you try to directly make
a **nonceAE** scheme
by applying **EtM**
to an **ivENC** scheme and a **MAC**
you might get …

a crocoduck.

Fortunately, nobody would do that.

Well ...

ISO/IEC 19772, Mechanism 5 (Encrypt-then-MAC)

The originator shall perform the following sequence of steps to protext a data string D.

a) A Starting Variable S for use with the selected block cipher mode of operation shall be selected. This variable shall be distinct for every message to be protected during the lifetime of the key, and must be made available to the recipient of the message. Further possible requirements for S are described in the appropriate clauses of ISO/IEC 10116.

b) Let $C' = \varepsilon_{K_1}(D)$.

c) Let $T = f_{K_2}(C')$.

The output of the above process, i.e., the authenticated-encrypted version of D, shall be the bit string

$C = C' \,\|\, T$.

CBC, CFB, OFB, CTR (ISO 9797)

CBC MAC variants (ISO 10116)

Not good.

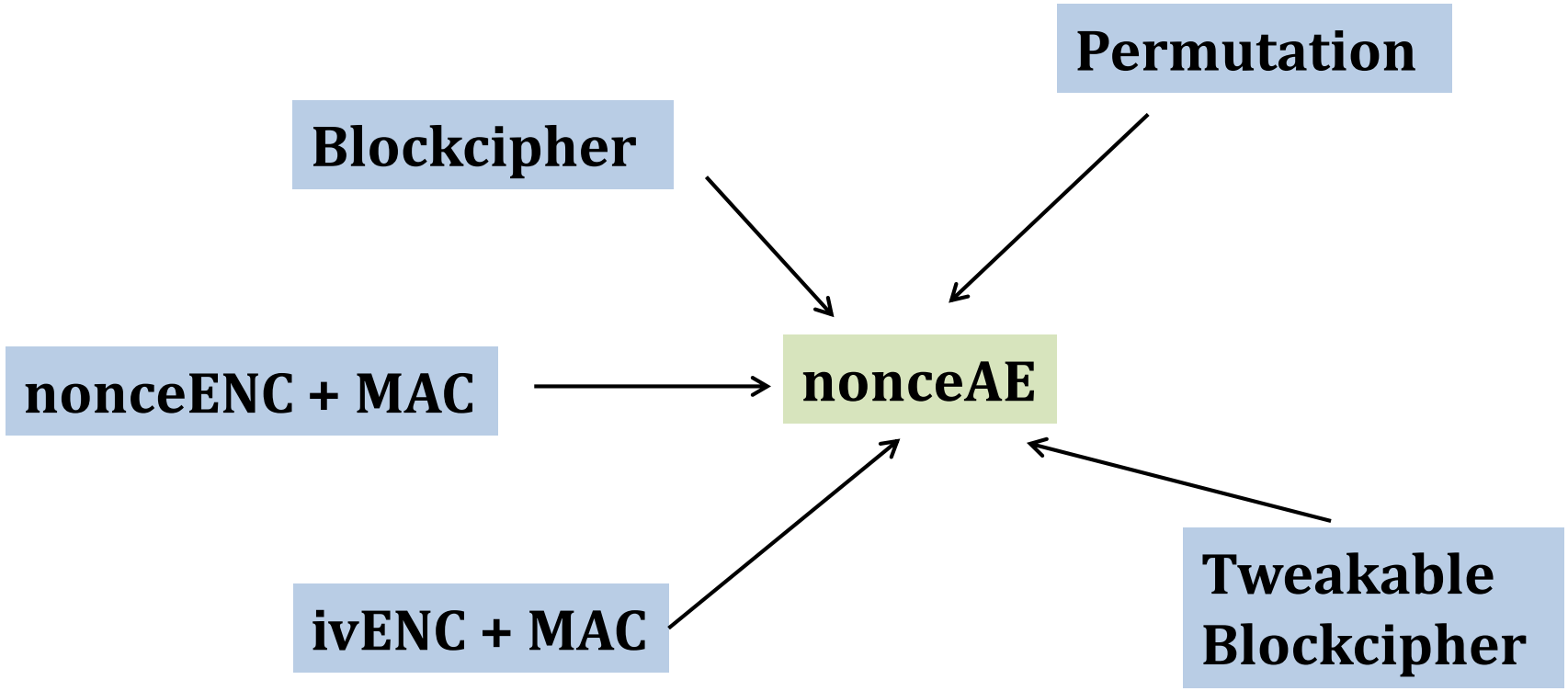
- The SV is not included in the MAC
- Nor is the SV required to be random
- Nor are the underlying encryption modes and MACs total

What is GC about?

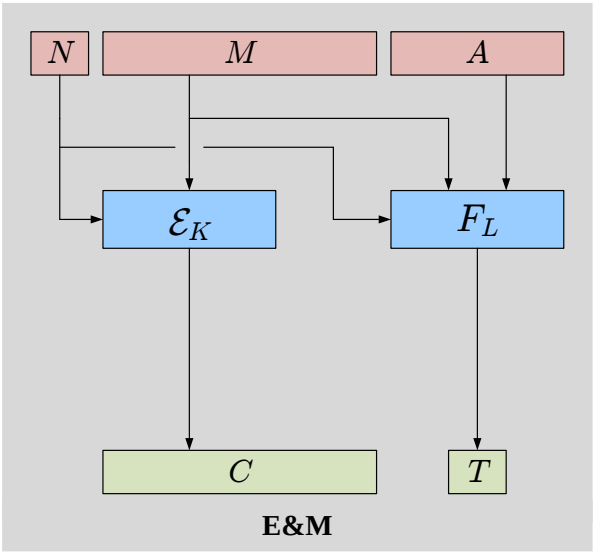
probENC + MAC —BN→ probAE

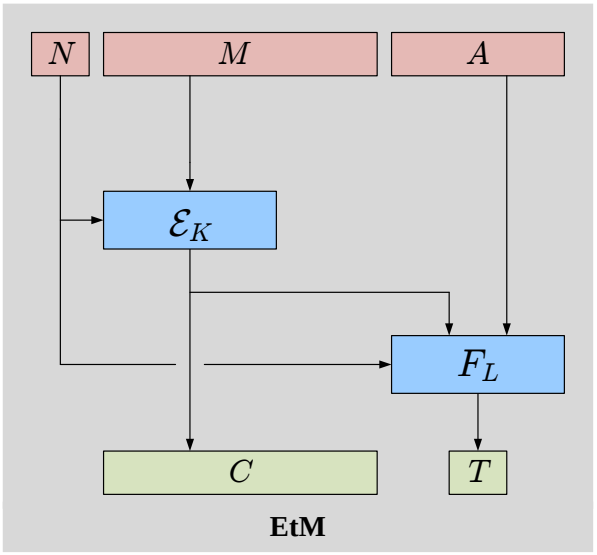
But several starting points and
several ending points are possible

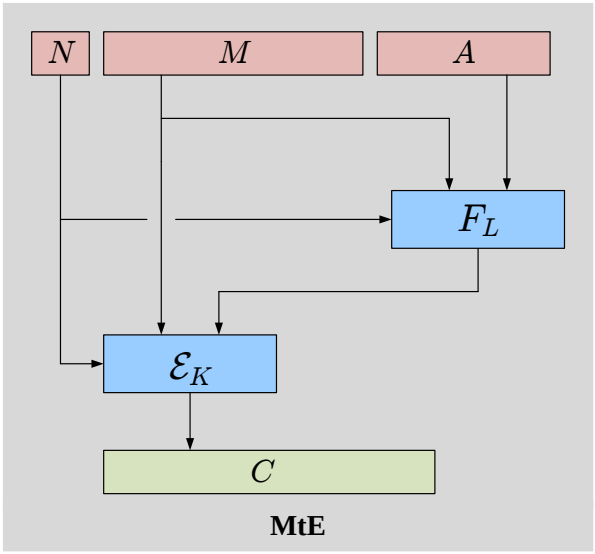
They are not all equal



nonceENC + MAC ⟶ **nonceAE**







All three methods work correctly.

$$\mathcal{E}_K(N, A, M) = \mathcal{C} \rightarrow \mathcal{D}_K(N, A, \mathcal{C}) = M \quad ?$$

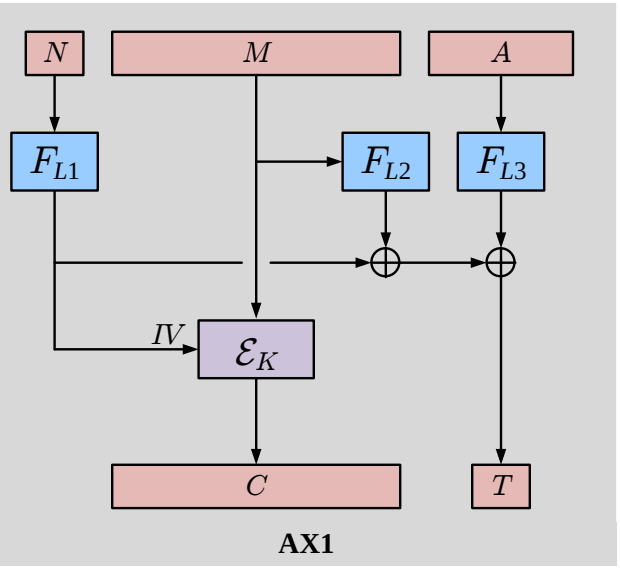
$$\mathcal{D}_K(N, A, \mathcal{C}) = M \rightarrow \mathcal{E}_K(N, A, M) = \mathcal{C} \quad ?$$

if assume both.

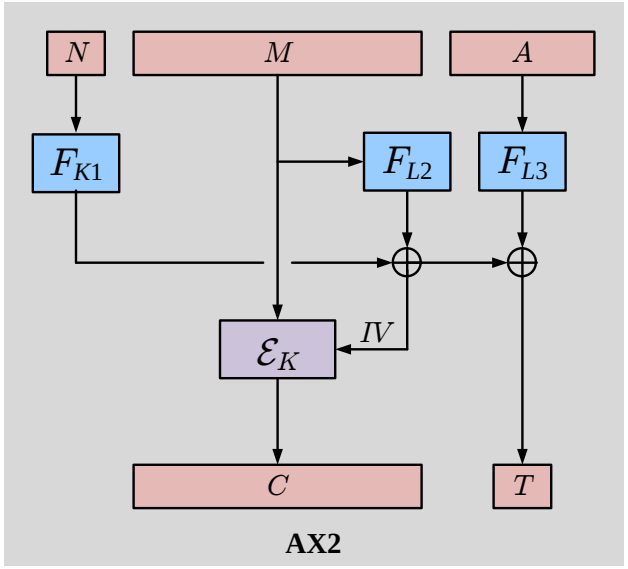
Bellare-Tackmann recently pointed out that, without this, E&M and MtE are insecure, making wrong my claim for MtE security in my CCS02 and FSE04 papers.

ivENC + MAC ⟶ **nonceAE**

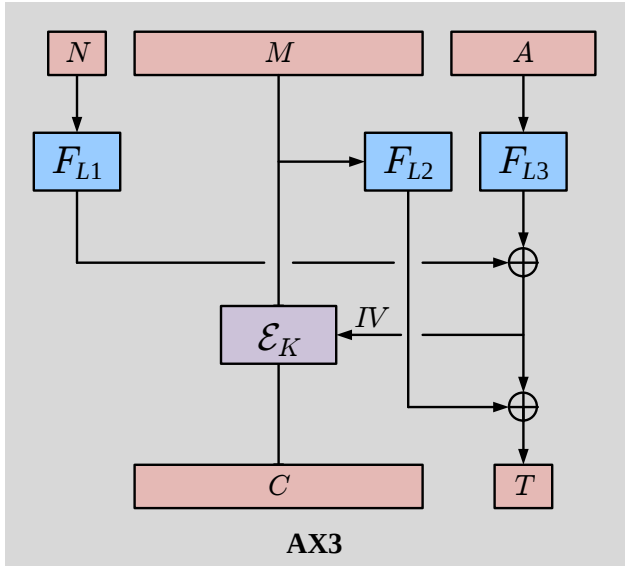
single-input MAC + XOR setting



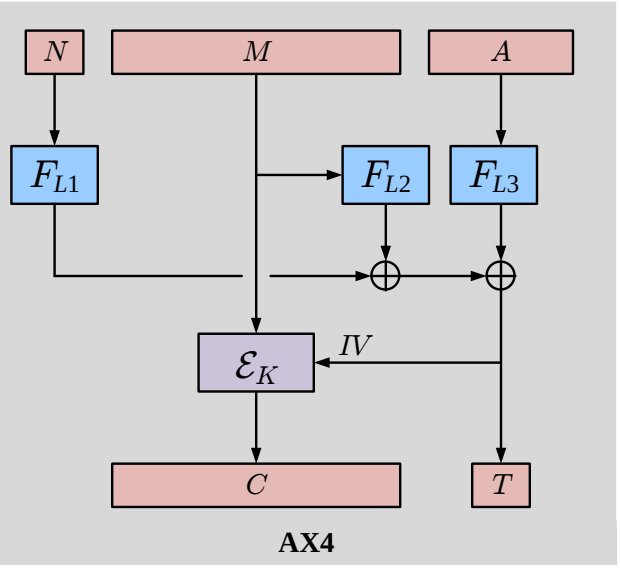
AX1



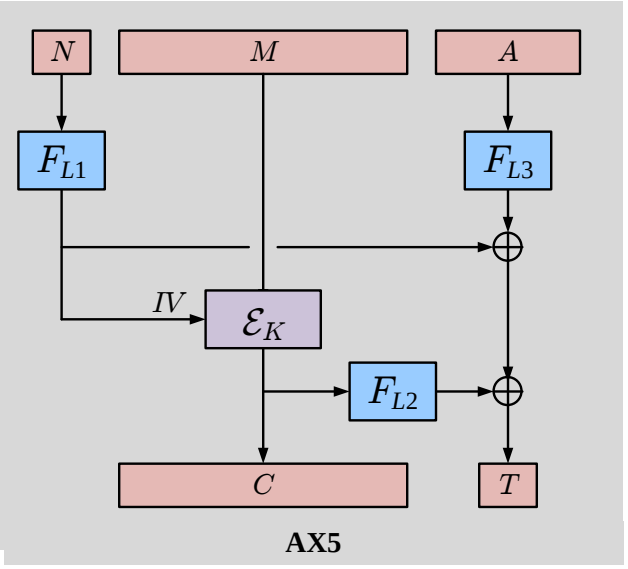
AX2



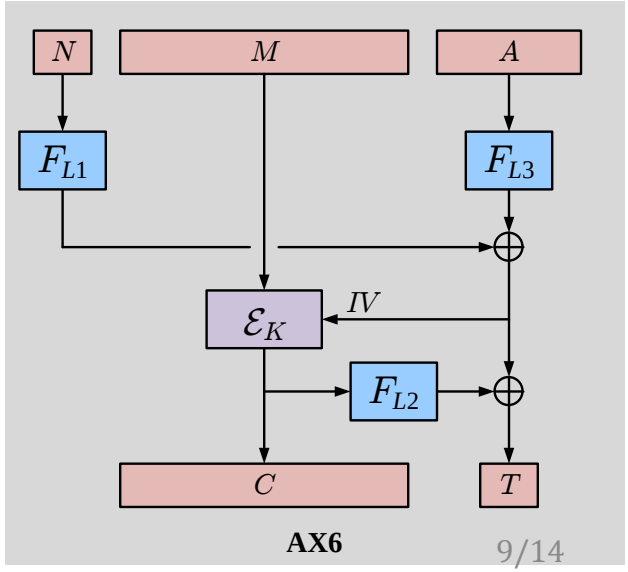
AX3



AX4

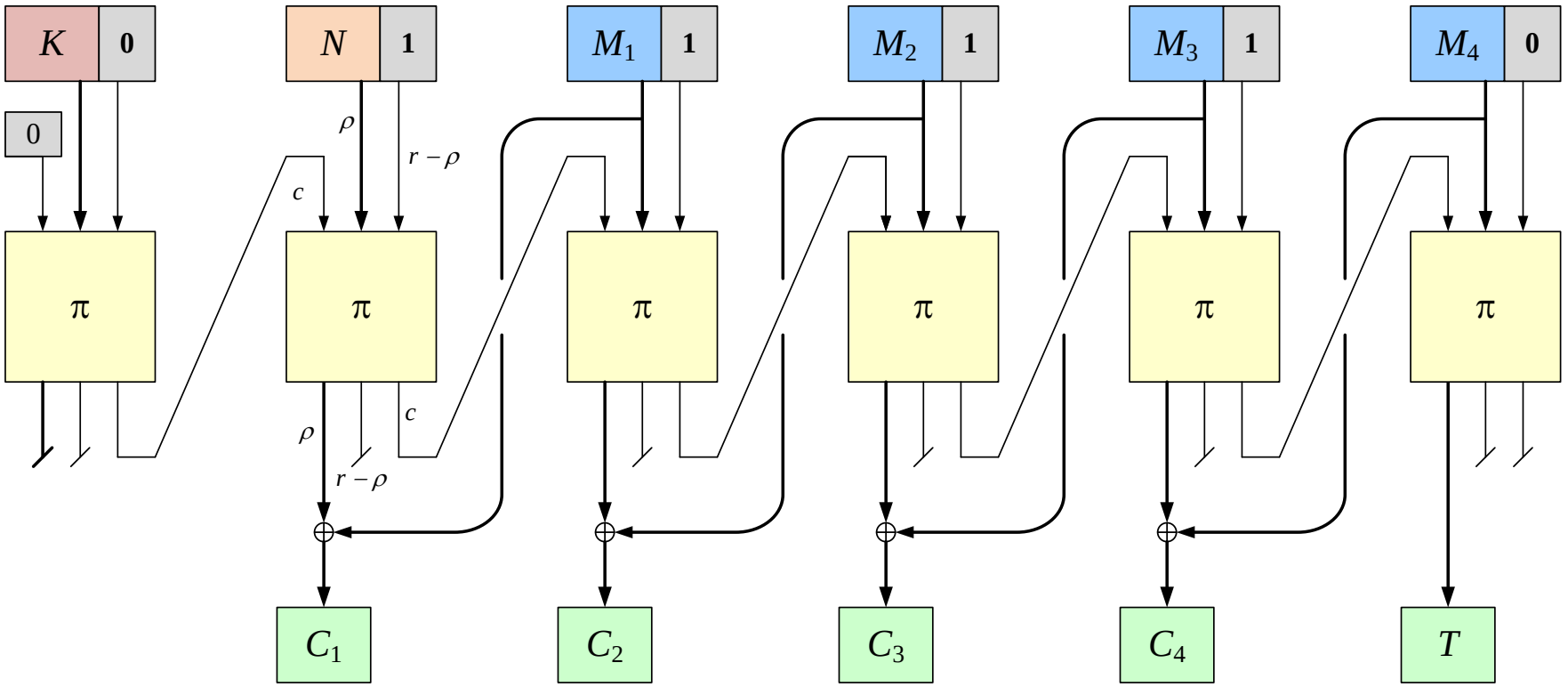


AX5



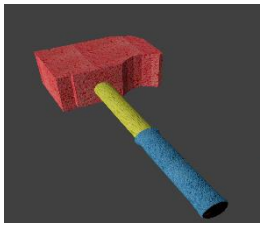
AX6

permutation → **nonceAE**

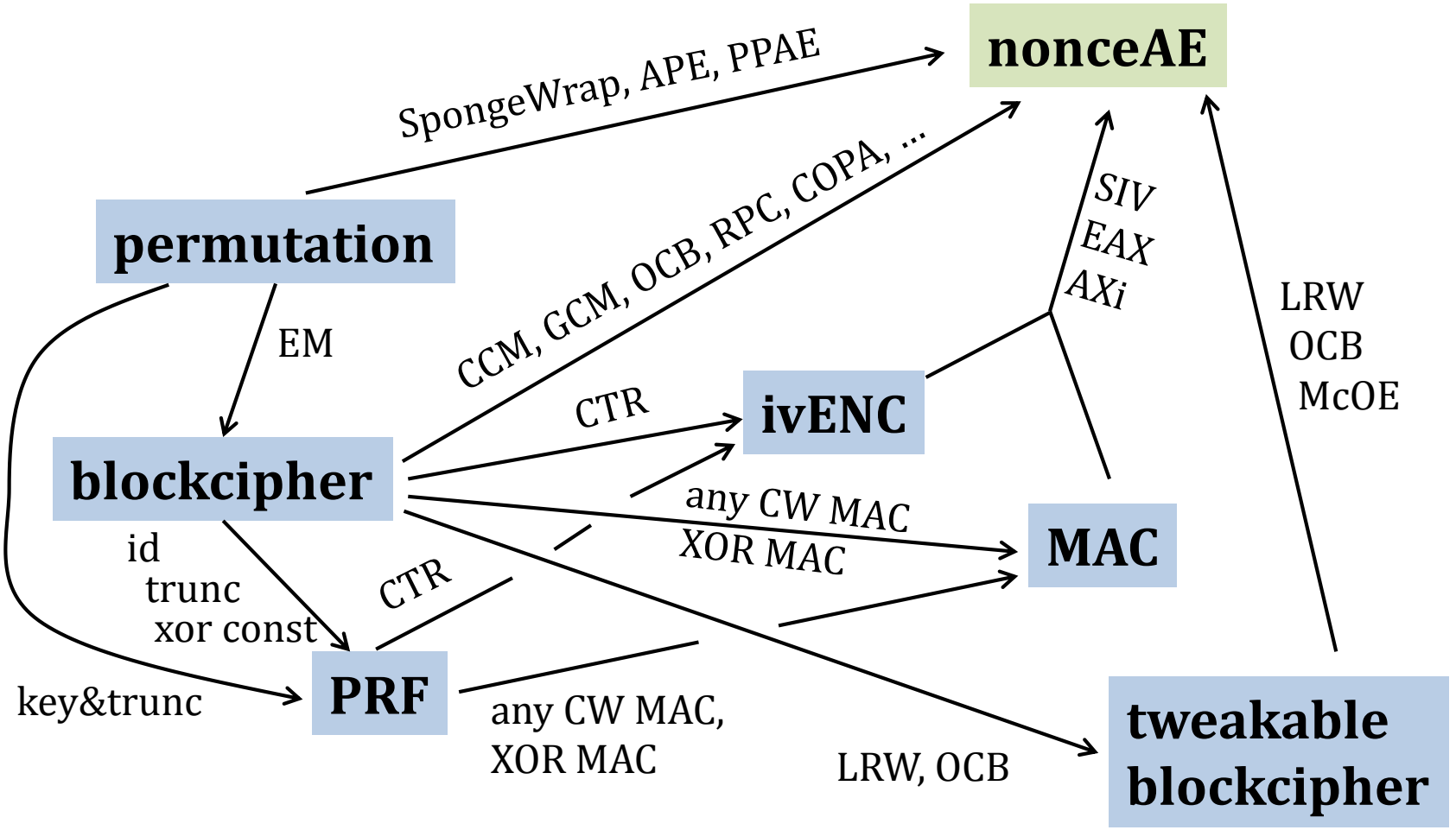


SpongeWrap
[Bertoni, Daemen, Peeters, Van Aasche 2011]

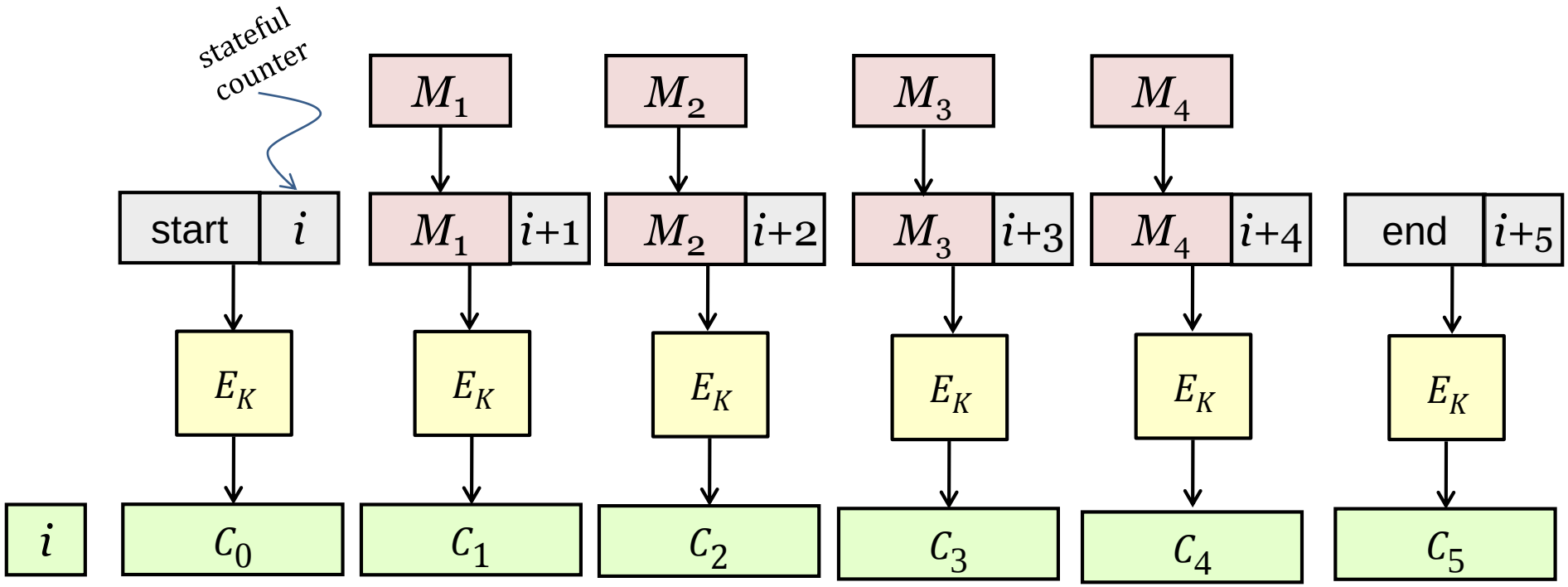
Not parallelizable, rate << 1, poor bounds, RPM …
Maybe a sponge not the tool for this job.



Can make permutation-based AE by <u>composition</u>



permutation → **nonceAE**



RPC Mode

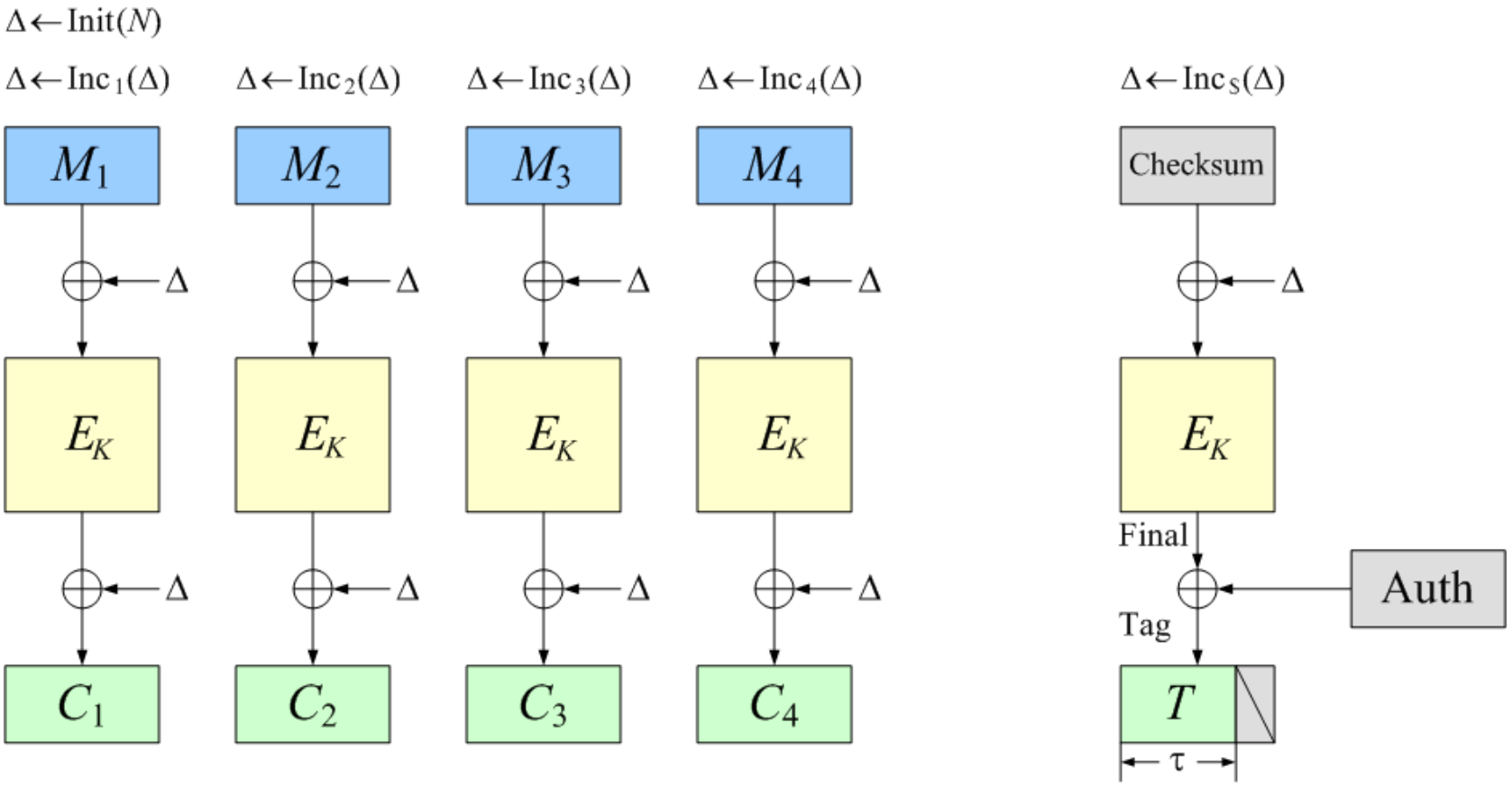
[Katz, Yung 2000]

Eg : Permutation-based RPC

via [EM]: $E_K(X) = K \oplus \pi(x \oplus K)$

Parallelizable, standard-model security claim

permutation ⟶ **nonceAE**



OCB Mode

[RBBK 2000 +]

Eg : Permutation-based OCB

via [EM]: $E_K(X) = K \oplus \pi(x \oplus K)$

Parallelizable, rate-1, standard-model security claim

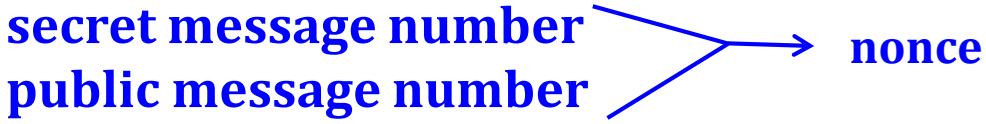
Summary and additional work

- **E&M / EtM / MtE** is **specific** to **one setting** — **probENC + MAC → probAE**
- Attack on **ISO 19772** as a symptom of over-generalization
- **nonceAE definitional exploration** — implications and separations
- Advocate **ind$**-style definition
  - Implies everything one expects, tightly, including **anonymity**
- "All" **nonceENC + MAC → nonceAE**
- "All" **ivENC + MAC → nonceAE** via **computer-aided search**
  - **multi-input MACs** $\mathcal{E}_{KL}(N,M,A)=\mathcal{E}_K(IV,\ M \,||\, S) \,||\, T$

    $IV = F_L(\mathsf{C}_{\text{iv}}, N \,|\, \Diamond, M \,|\, \Diamond, A \,|\, \Diamond)$

    $S = F_L(\mathsf{C}_{\text{in}}, N \,|\, \Diamond, M \,|\, \Diamond, A \,|\, \Diamond)$

    $T = F_L(\mathsf{C}_{\text{out}}, N \,|\, \Diamond, M \,|\, \Diamond, A \,|\, \Diamond, C \,|\, \Diamond)$
  - **single-input MACs + XORs** — *analogous*
- For **perm → nonceAE**, making a **compositional approaches** sensible



A scheme makes a **nonce confidentiality** choice:
- **no**: nonces are not incorporated into the ciphertext
- **yes**: nonces are incorporated into the ciphertext

```
int crypto_aead_decrypt(
const unsigned char *k
unsigned char *nonce,
const unsigned char *ad, unsigned long long adlen,
const unsigned char *c, unsigned long long clen,
unsigned char *m, unsigned long long *mlen,
)
```

```
unsigned char *nsec,
const unsigned char *npub,
```