Reconsidering Generic Composition

Chanathip Namprempre
Thammasat University, Thailand

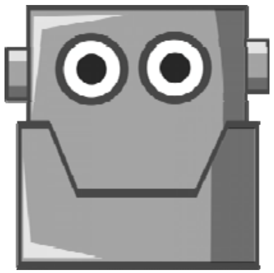
Phillip Rogaway
University of California, Davis, USA

Tom Shrimpton
Portland State University, USA

What is the correct way to build
an authenticated encryption scheme
from an encryption scheme and a MAC?

What is the correct way to build
an authenticated encryption scheme
from an encryption scheme and a MAC?

[Bellare-Namprempre – ASIACRYPT 2000]
Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm



Encrypt-and-MAC

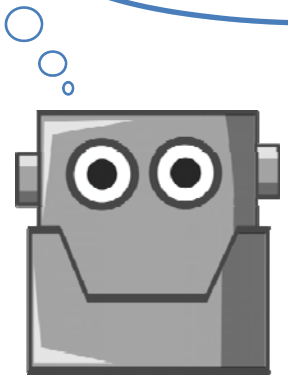
Encrypt-then-MAC

MAC-then-Encrypt

always works if encryption IND-CPA secure and MAC unforgeable

What is the correct way to build
an authenticated encryption scheme
from an encryption scheme and a MAC?

[Bellare-Namprempre – ASIACRYPT 2000]
Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm



Encrypt-and-MAC

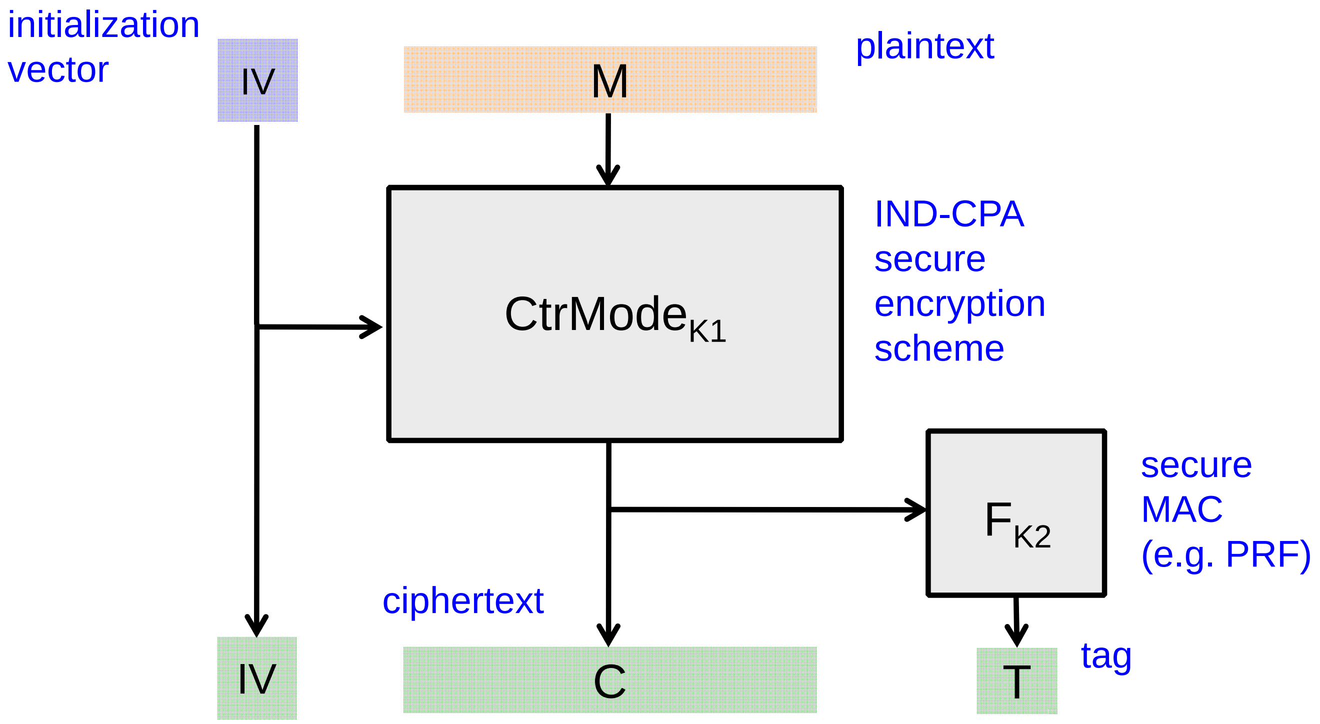
Encrypt-then-MAC

MAC-then-Encrypt

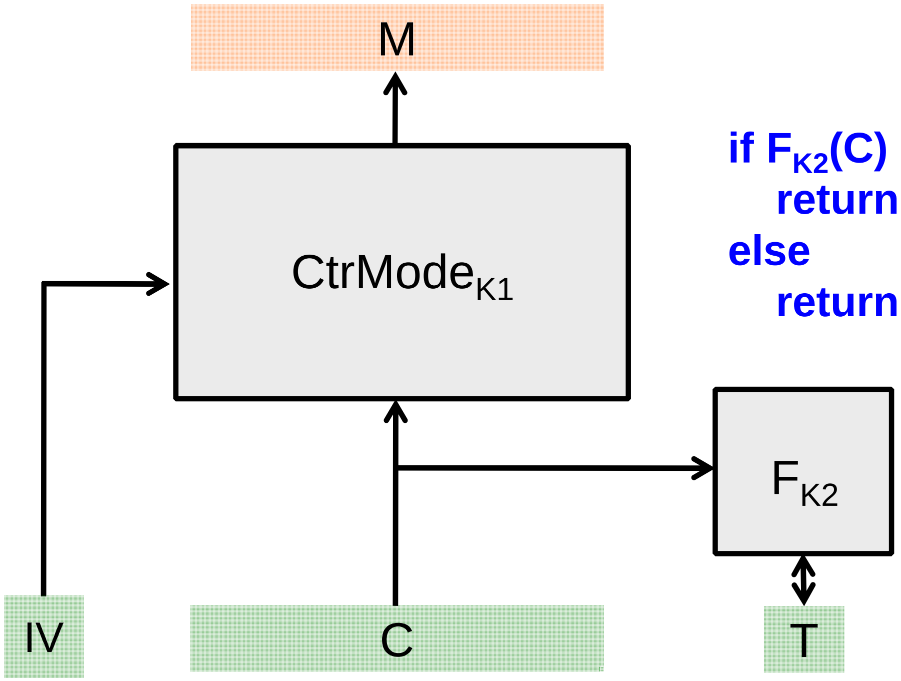
always works if
encryption IND-CPA secure
and MAC unforgeable

This summary of [BN] is incorrect.

Encrypt-then-MAC

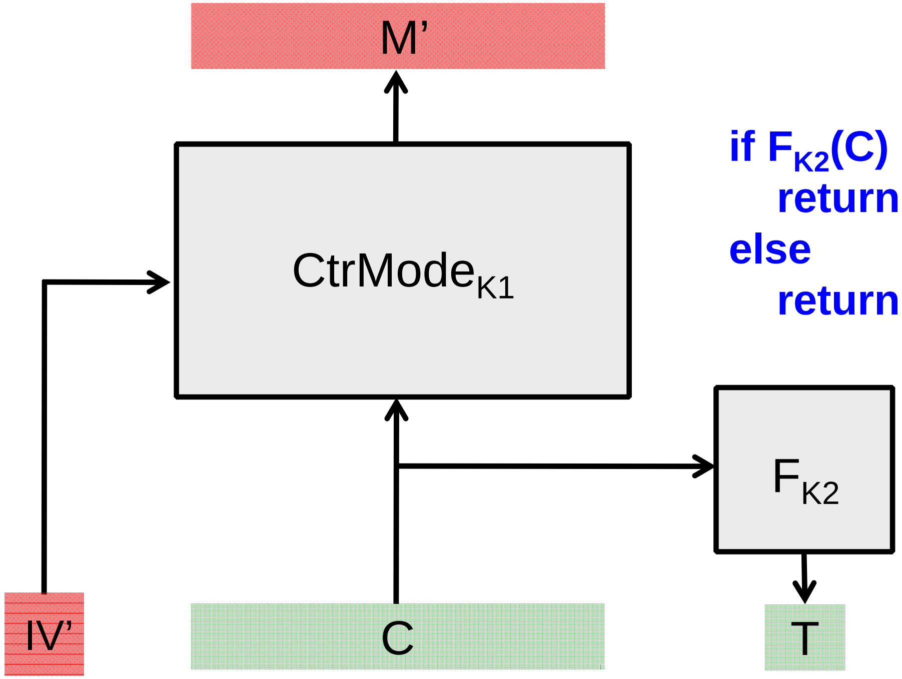


Encrypt-then-MAC



```
if $F_{K2}(C) = T$ then
    return $\text{Dec}_{K1}(IV,C)$
else
    return “invalid”
```

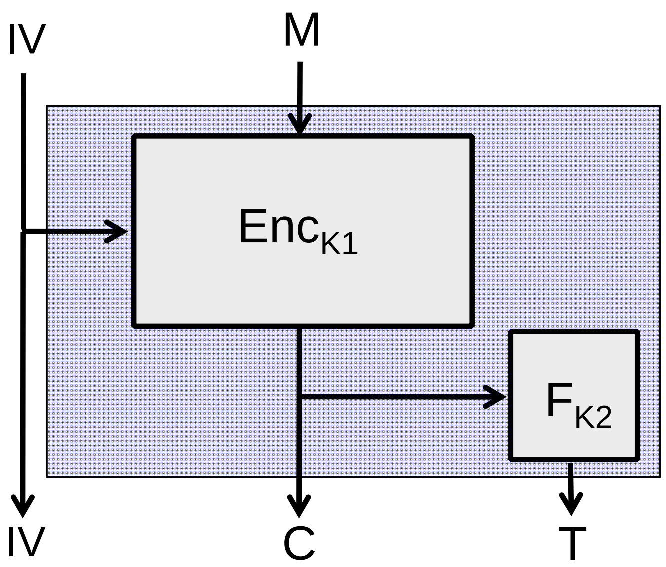
Encrypt-then-MAC

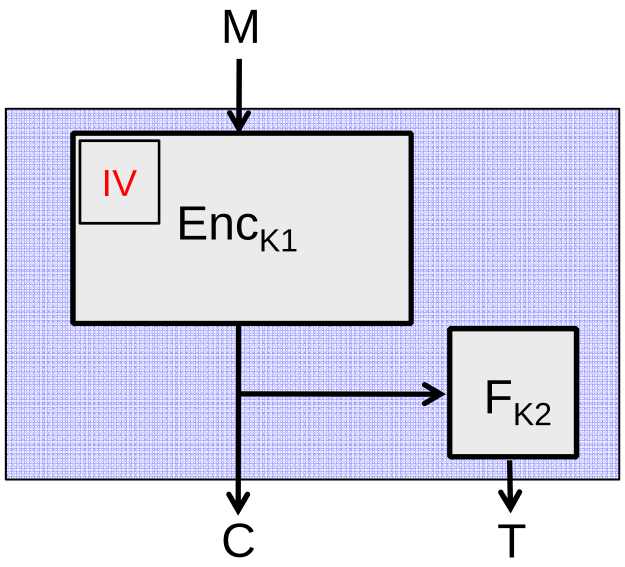


```
if $F_{K2}(C)$ = T then
    return $\text{Dec}_{K1}(IV,C)$
else
    return “invalid”
```

But… [BN] says… ???

“Encrypt-then-MAC” vs. Encrypt-then-MAC





IV-based AE scheme built from an **IV-based encryption** scheme and a MAC

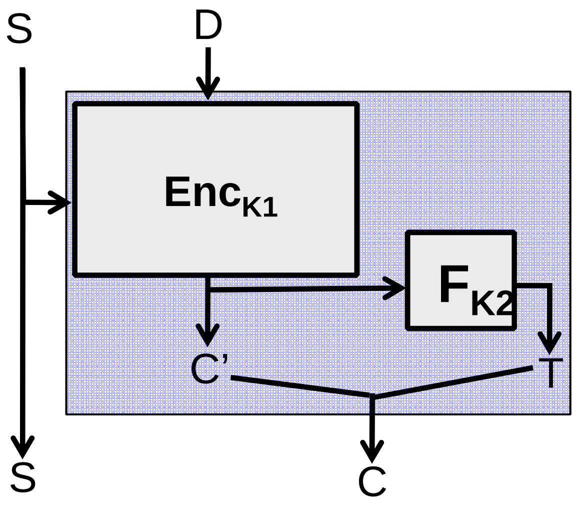
Probabilistic AE scheme built from a **probabilistic encryption** scheme and a MAC

[BN] is about this setting only.

Different starting primitives, different final primitives, different security

Incorrect summary of [BN], in practice

ISO/IEC 19772, Mechanism 5 (Encrypt-then-MAC)
Information Security – Security Techniques – Authenticated Encryption



S required to be a nonce (but not random)

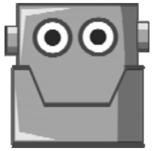
"Enc" = CBC, CTR, OFB, CFB blockcipher modes

-- not all have {0,1}* domains

-- some require S to be random for IND-CPA

S not covered by tag

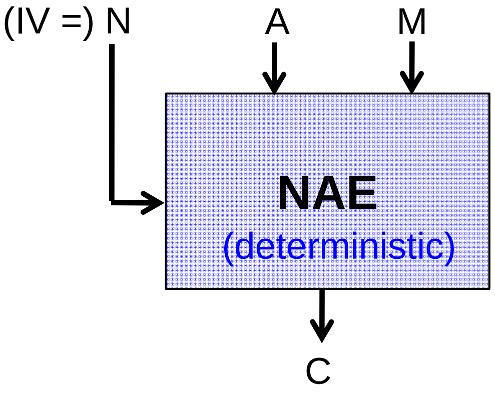
Appeals to [BN] to justify security of a nonce-based scheme
built from IV-based encryption.



“Okay, fine: EtM + secure prob. Enc + secure MAC = secure prob. AE”

The thing is…

1. Typical goal nowadays is **nonce-based AE with associated data (NAE),** not probabilistic AE



N= nonce (“number used once”, e.g. sequence number)

A = associated data, bound to plaintext/ciphertext, not private

M = plaintext, private

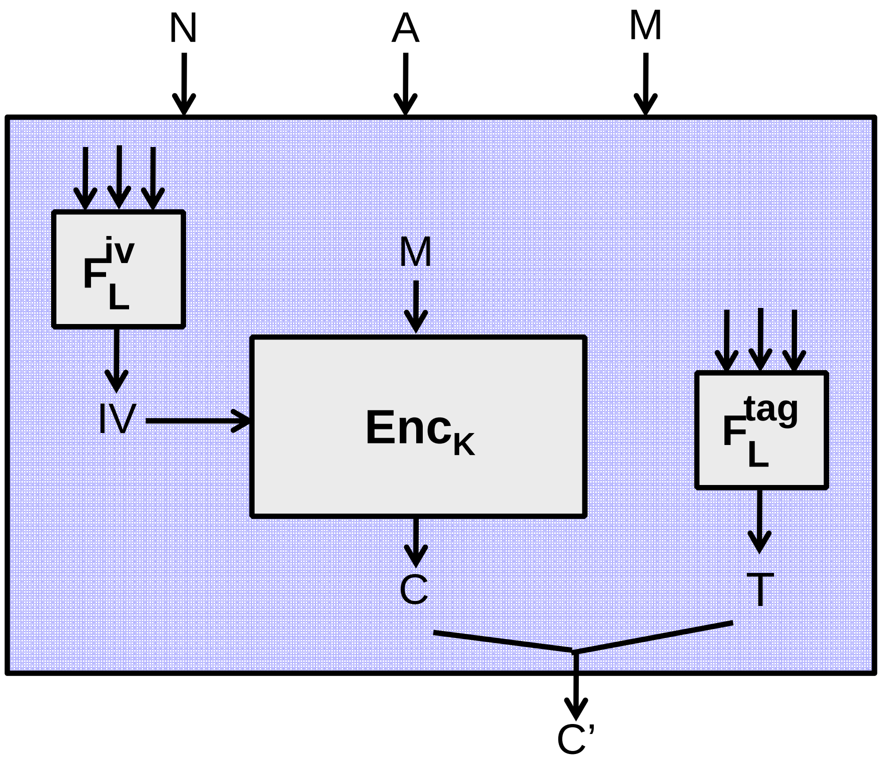
2. Standards and common crypto libraries **don’t provide probabilistic encryption** schemes, they provide **IV-based encryption**

```
int encrypt(unsigned char *plaintext,
            int plaintext_len,
            unsigned char *key,
            unsigned char *iv,
            unsigned char *ciphertext)
```

openSSL encryption API

What are the correct ways to compose
a secure IV-based encryption scheme
and a secure PRF in order to build
a nonce-based AE(AD) scheme?

Our basic NAE forms

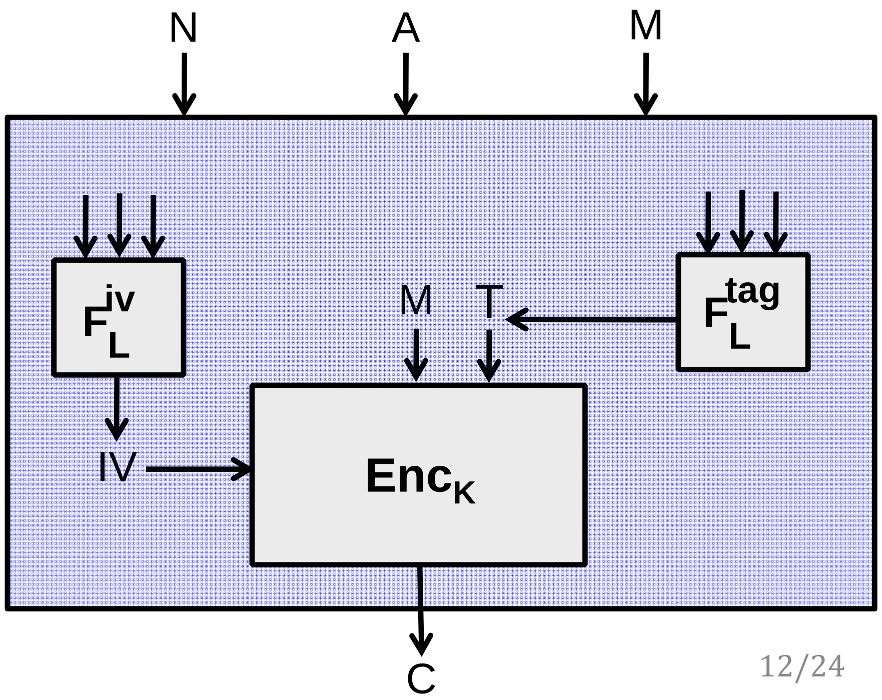


F^{iv} inputs: (N or □ , A or □, M or □)

F^{tag} inputs: (N or □ , A or □, M or □) "E&M"

or (N or □ , A or □, C or □) "EtM"

□ = "missing"



F^{iv} inputs: (N or □ , A or □, M or □)

F^{tag} inputs: (N or □ , A or □, M or □)

160 possible constructions analyzed, resulting in:

8 "favored" schemes --- generically secure, good security bounds

1 "transitional" scheme --- generically secure, inferior bound

3 "elusive" schemes --- despite LOADS of effort, unable to find proofs
using only IND$-CPA and PRF security of components,
unable to find counterexamples

All other schemes --- we find counterexamples (many trivial, some not)

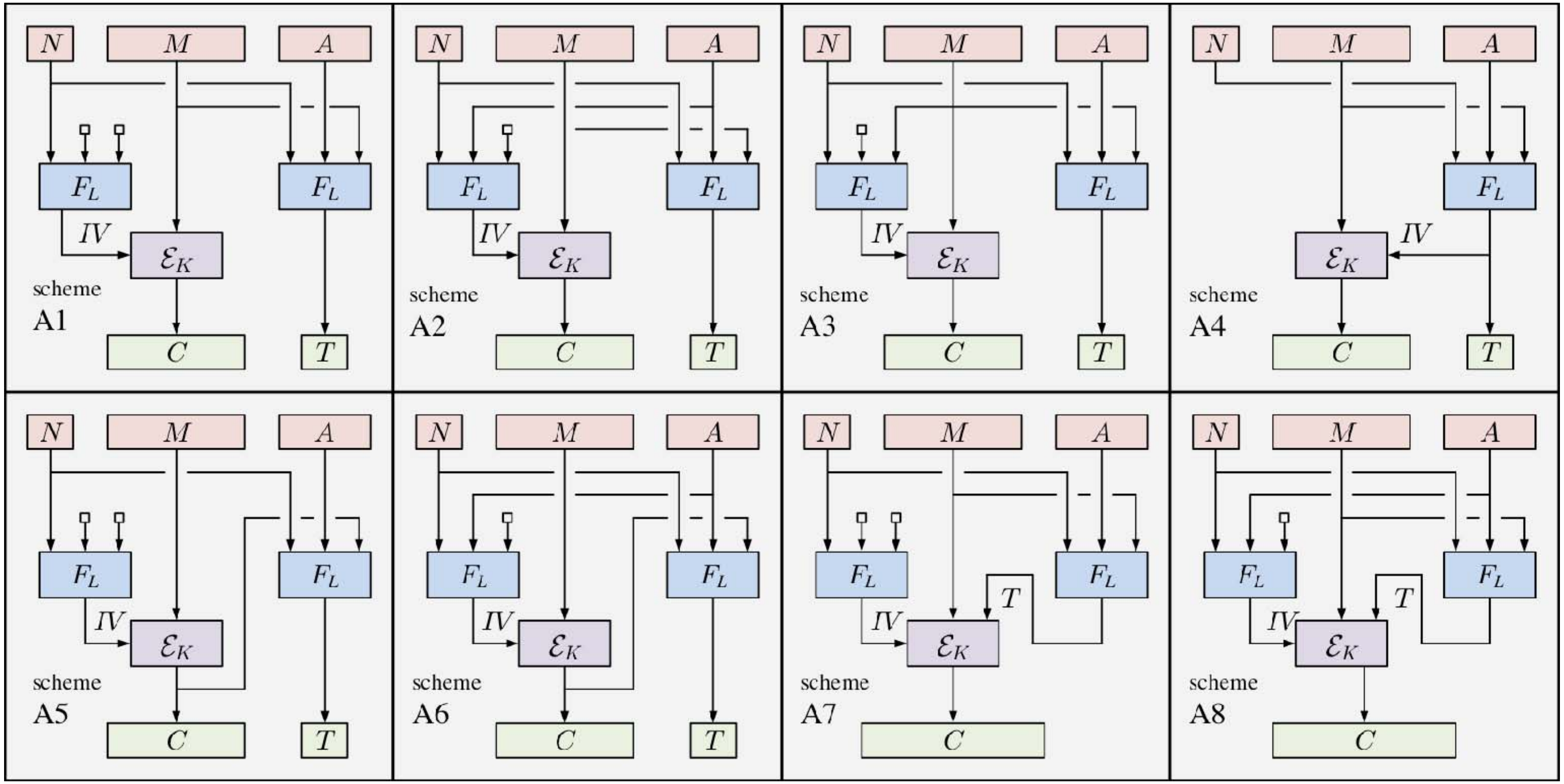
What security notion?

$$\mathbf{Adv}_{\Pi}^{\mathrm{nAE}}(\mathcal{A}) = \Pr\left[\mathcal{A}^{\mathcal{E}(\cdot,\cdot,\cdot),\,\mathcal{D}(\cdot,\cdot,\cdot)} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\$(\cdot,\cdot,\cdot),\,\perp(\cdot,\cdot,\cdot)} \Rightarrow 1\right]$$

we target an "all-in-one" AE notion [RS06],
equivalent to IND$-CPA + INT-CTXT

The favored eight

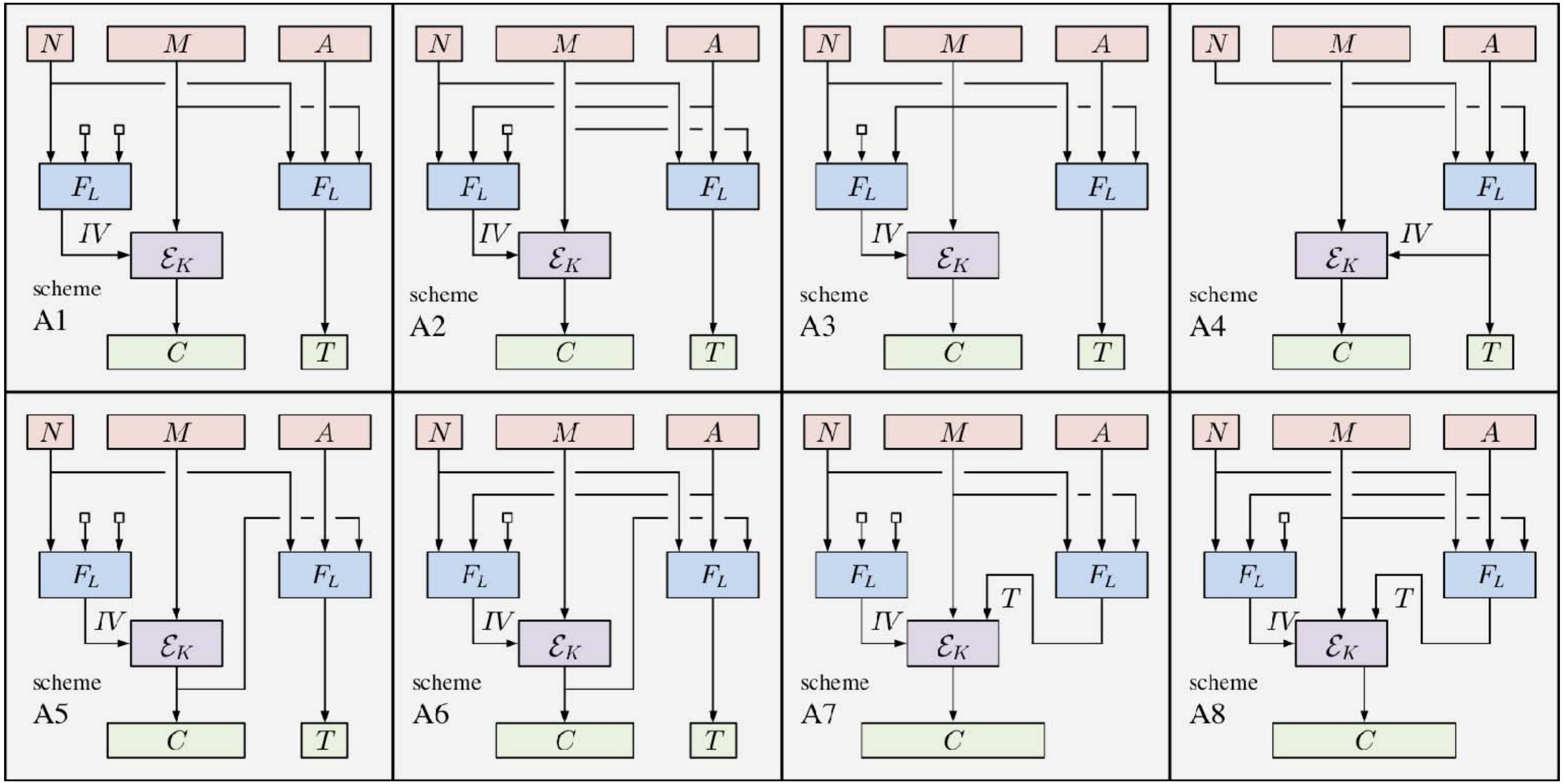
“E and M” “E and M” “E and M” SIV mode [RS06]



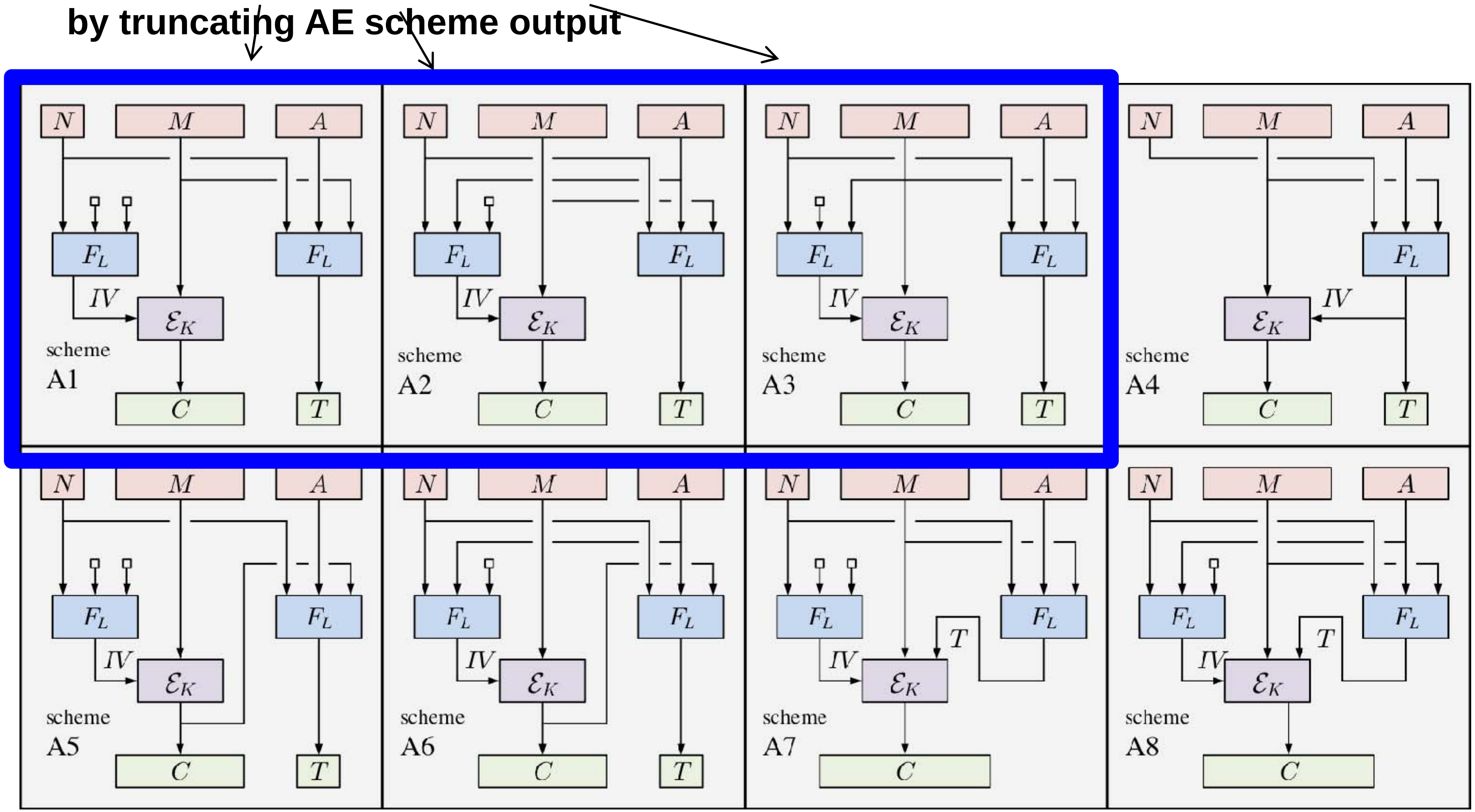
“E then M” “E then M” “M then E” “M then E”

The favored eight all have the same (good, tight) AE security.

Which should I use?

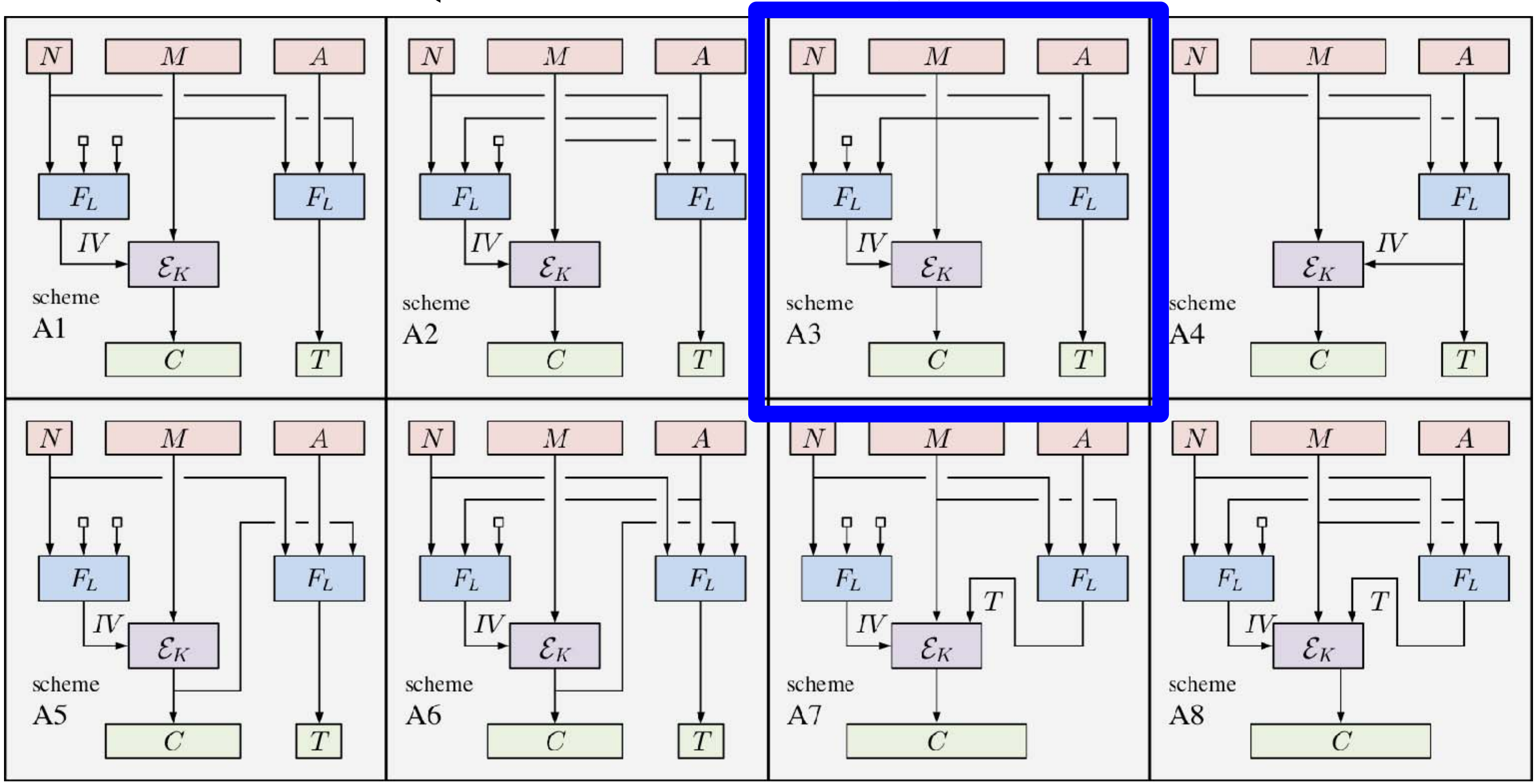


Encryption can compute C,T in parallel
Can truncate the tag
by truncating AE scheme output



Encryption can compute C,T in parallel
Can truncate the tag
by truncating AE scheme output

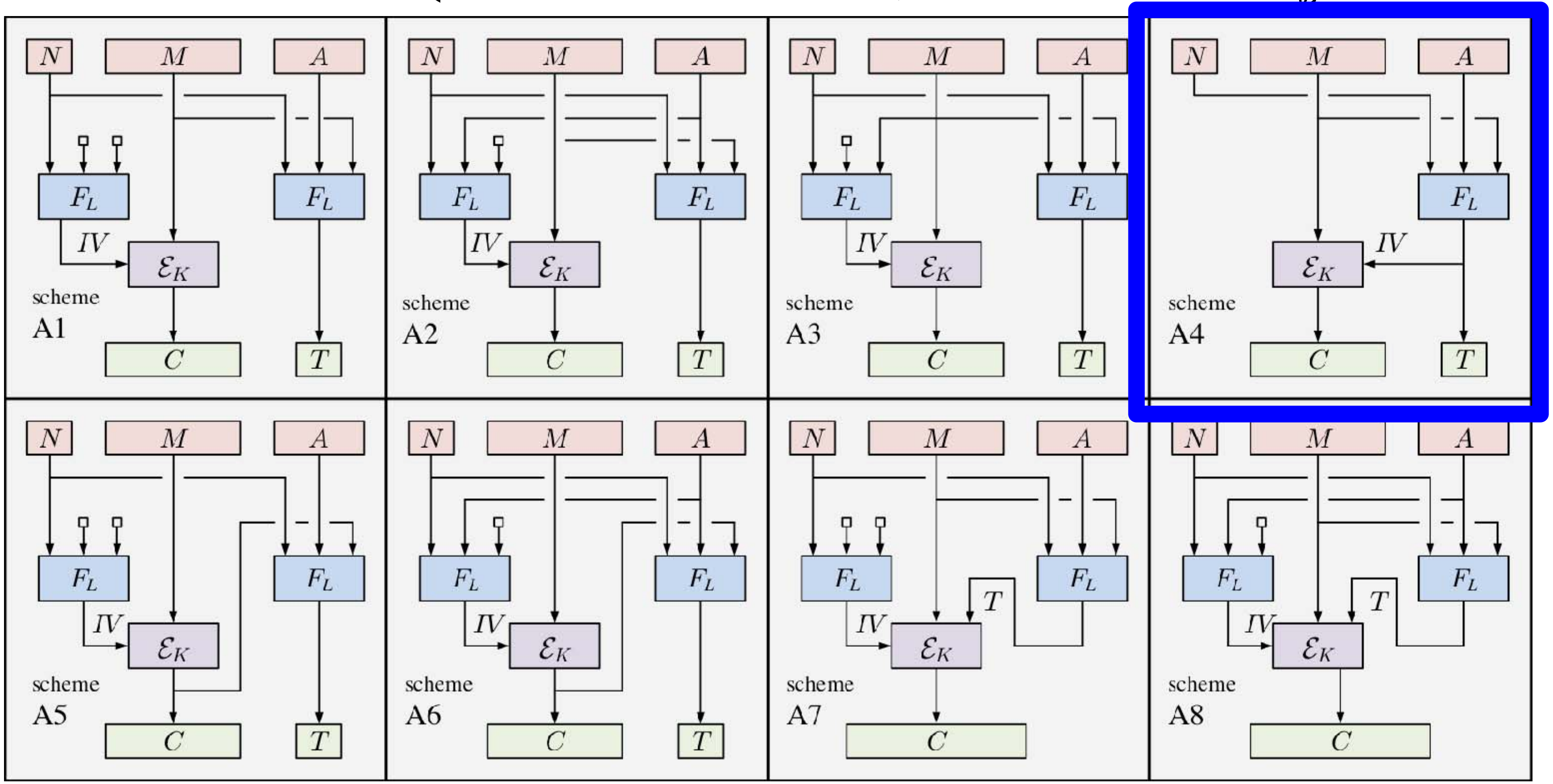
IV must be recoverable from C,T



Encryption can compute C,T in parallel
Can truncate the tag
by truncating AE scheme output

IV must be recoverable from C,T

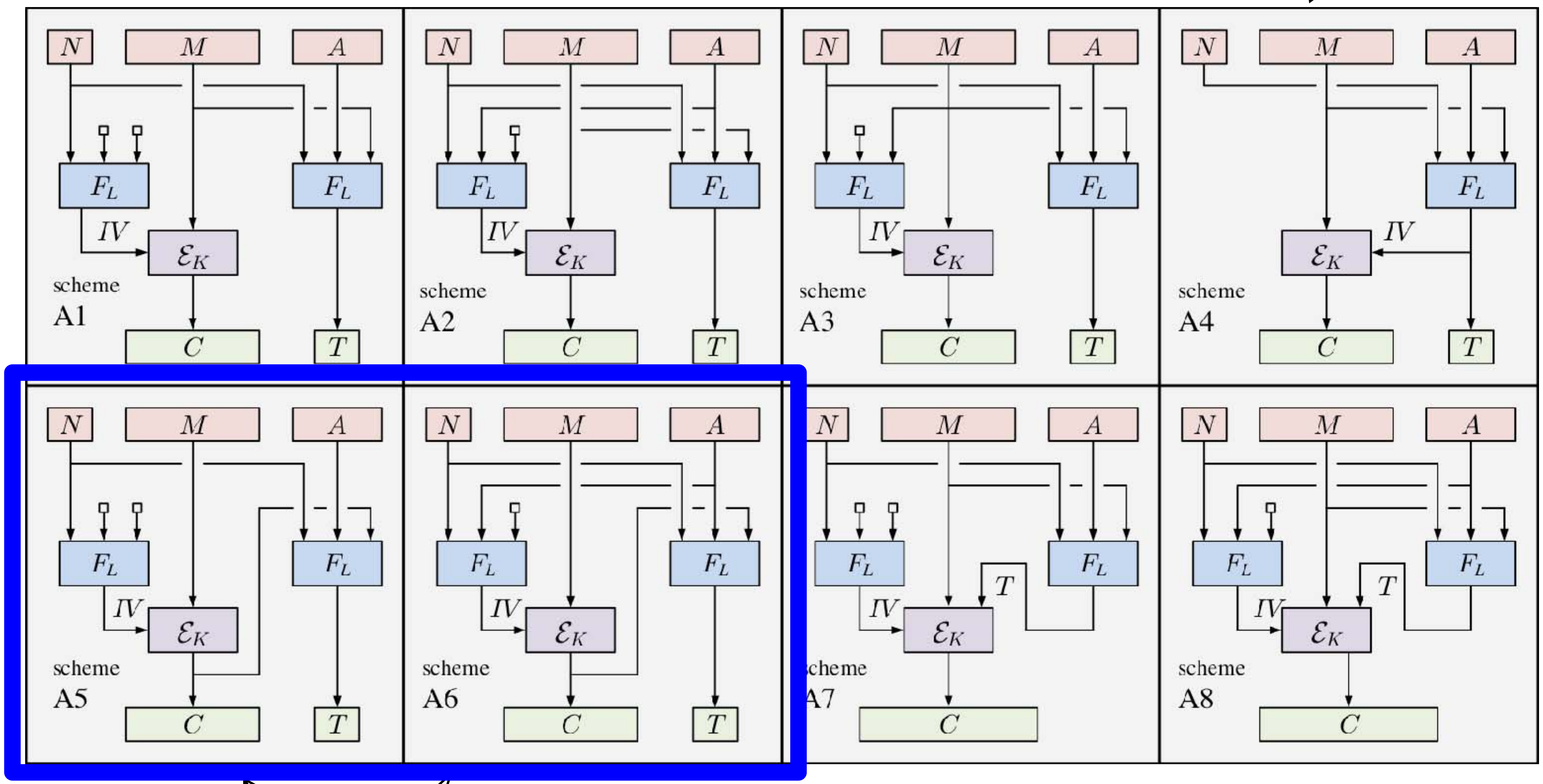
Nonce-misuse resistan
Cannot truncate



Encryption can compute C,T in parallel
Can truncate the tag
by truncating AE scheme output

IV must be recoverable
from C,T

Nonce-misuse resistan
Cannot truncate

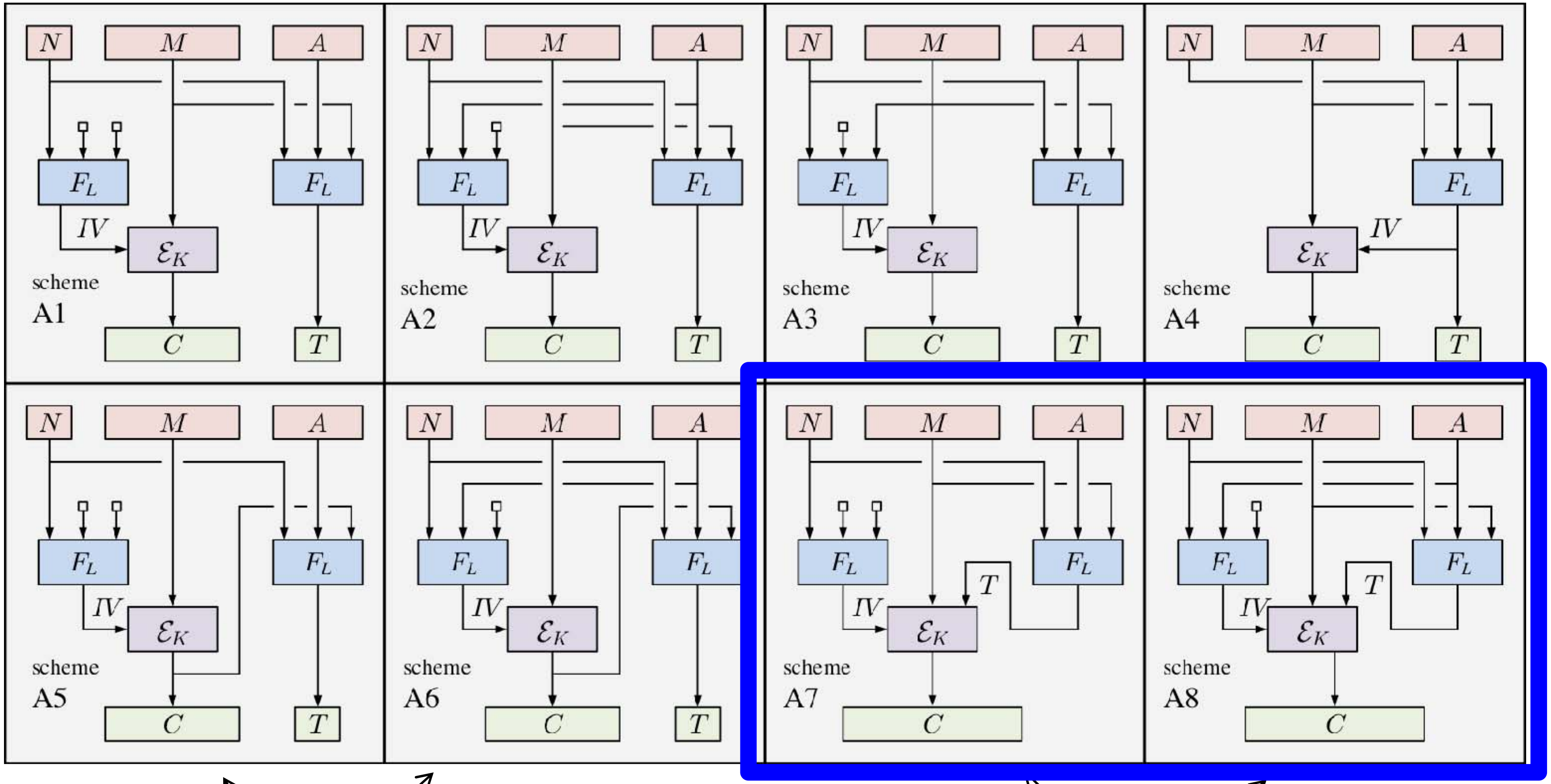


Decryption can compute M, check T in parallel
Can truncate

Encryption can compute C,T in parallel
Can truncate the tag
by truncating AE scheme output

IV must be recoverable from C,T

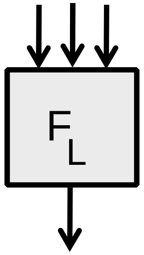
Nonce-misuse resistan
Cannot truncate



Decryption can compute M, check T in parallel
Can truncate

Cannot truncate
“MtE” style schemes have history
of problems in practice

What are these “vector input” PRFs?
Real PRFs (e.g. HMAC-SHA) take a string!



Can be instantiated in many ways. We use the **three-xor construction**.

$$F_{L1,L2,L3}(N,A,M) = f_{L1}(N) \oplus f_{L2}(A) \oplus f_{L3}(M)$$

$$F_{L1,L2,L3}(N,\square,M) = f_{L1}(N) \oplus 0^n \oplus f_{L3}(M)$$

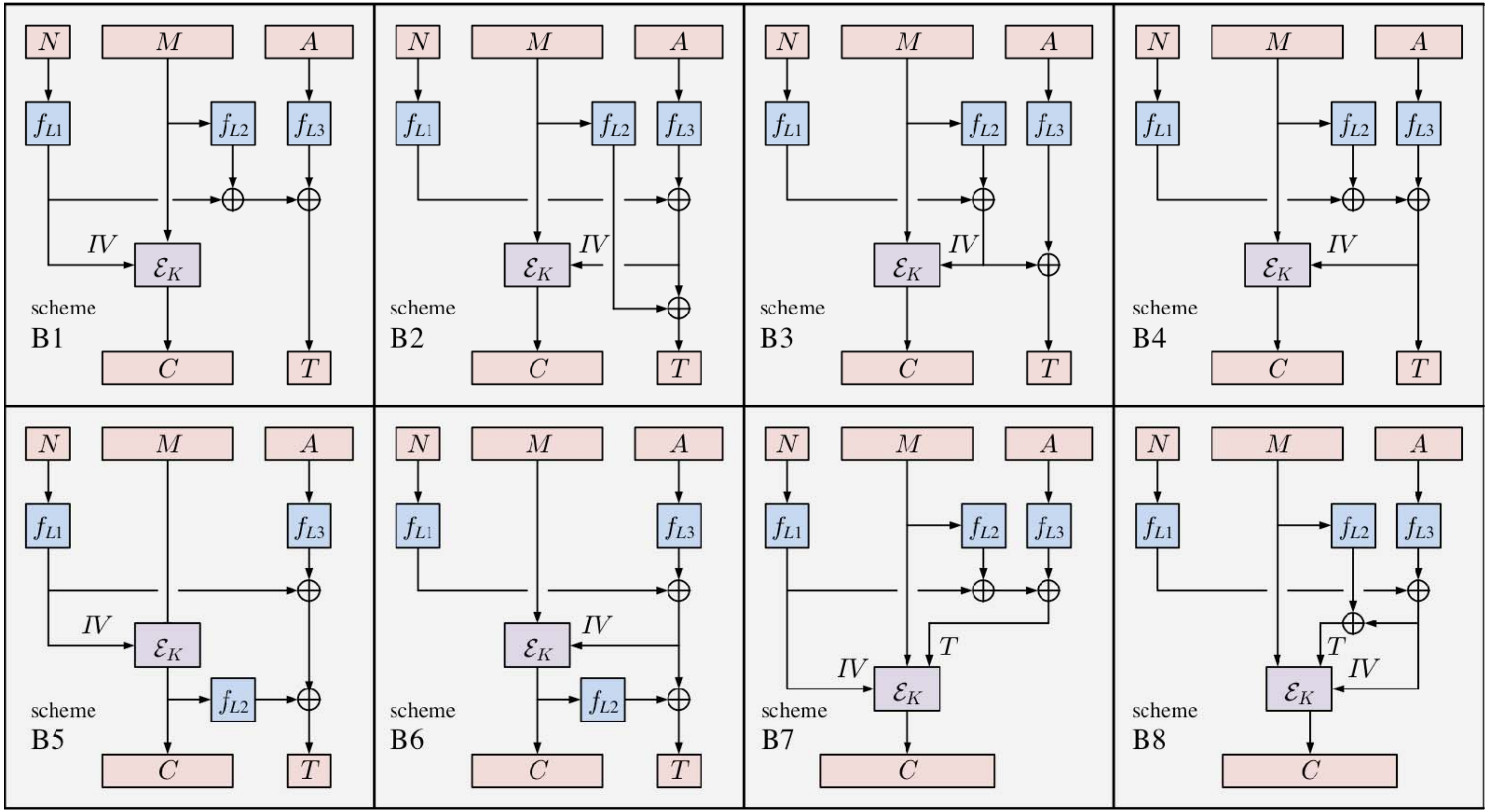
$$F_{L1,L2,L3}(\square,\square,M) = 0^n \oplus 0^n \oplus f_{L3}(M)$$

etc.

The favored eight, based on a string-input PRF

(using the three-XOR construction)

EAX2
[Bellare, R, Wagner'04]



Also in the paper

Building NAE from tidy **nonce-based encryption** and a PRF:
Three secure options, one elusive.

Proofs of security for elusive schemes under new “knowledge of tags” assumption

An ISO standard that uses [BN] to justify an NAE design = Broken

Discussion of “tidiness” as a syntactic property of deterministic encryption

High-level Summary

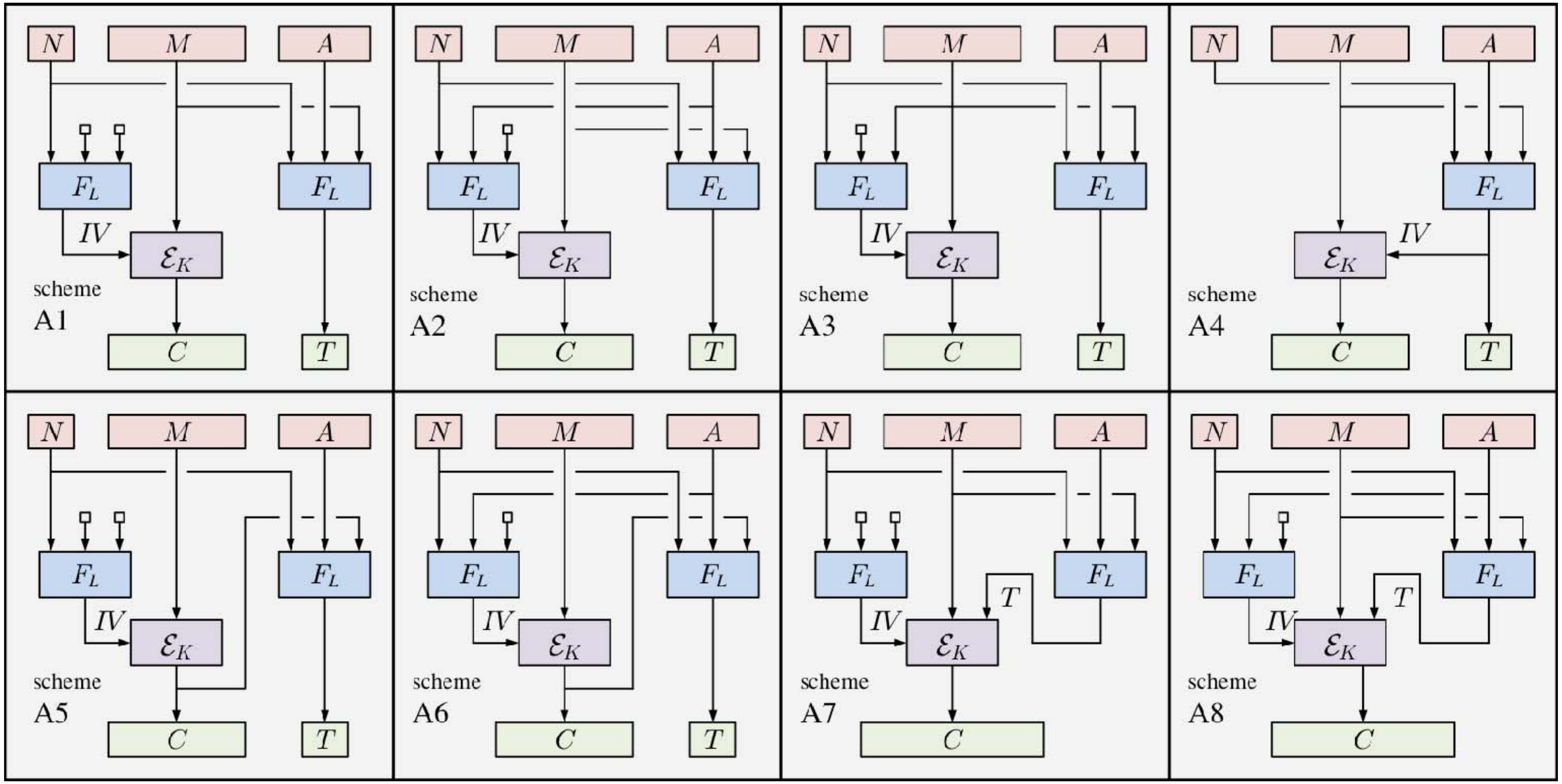
[BN] is fine, but people’s “understanding” of it over-generalizes,
leading to problems in practice

E&M, EtM, MtE taxonomy / security characterization is specific
to building probabilistic AE from probabilistic encryption

GC story is much more nuanced when building nonce-based AE

Thank you!

“E and M” “E and M” “E and M” SIV mode [RS06]



“E then M” “E then M” “M then E” “M then E”

--- END ---