5: Dedicated AE(AD) Schemes

- Given an IND-CPA-secure ENCR and an UF-CMA-secure MAC, Encrypt-then-MAC with ENCR and MAC guarantees secure AE.
- Though, many researchers studied homogeneous AE schemes.
- Motivation:
  - Single key (instead of two independent K and L)
  - Improved efficiency
  - Fun ;-)
- Many *dedicated* AEAD schemes in the literature
  - Similar to generic composition, with single key (CCM, GCM, EAX, ...)
  - Dedicated schemes (e.g., OCB)

Some Issues for the design of AEAD Schemes

- Efficiency:
  - As fast as possible
  - On a wide range of platforms (from small embedded to high-end)
  - Small chip size when implemented in hardware
  - Parallelism / pipelining
    (e.g., compare Counter and CBC encryption)
- Security:
  - Provable security
  - Take care of lengths
    (e.g., when computing $A \leftarrow \mathsf{MAC}_K(A \,\|\, C)$, an adversary might shift the boundary between associated data A and ciphertext C
- Simplicity:
  - Easy to describe
  - Easy to understand
  - Easy to analyze
  - Easy to implement correctly

5.1: Two-Pass Modes

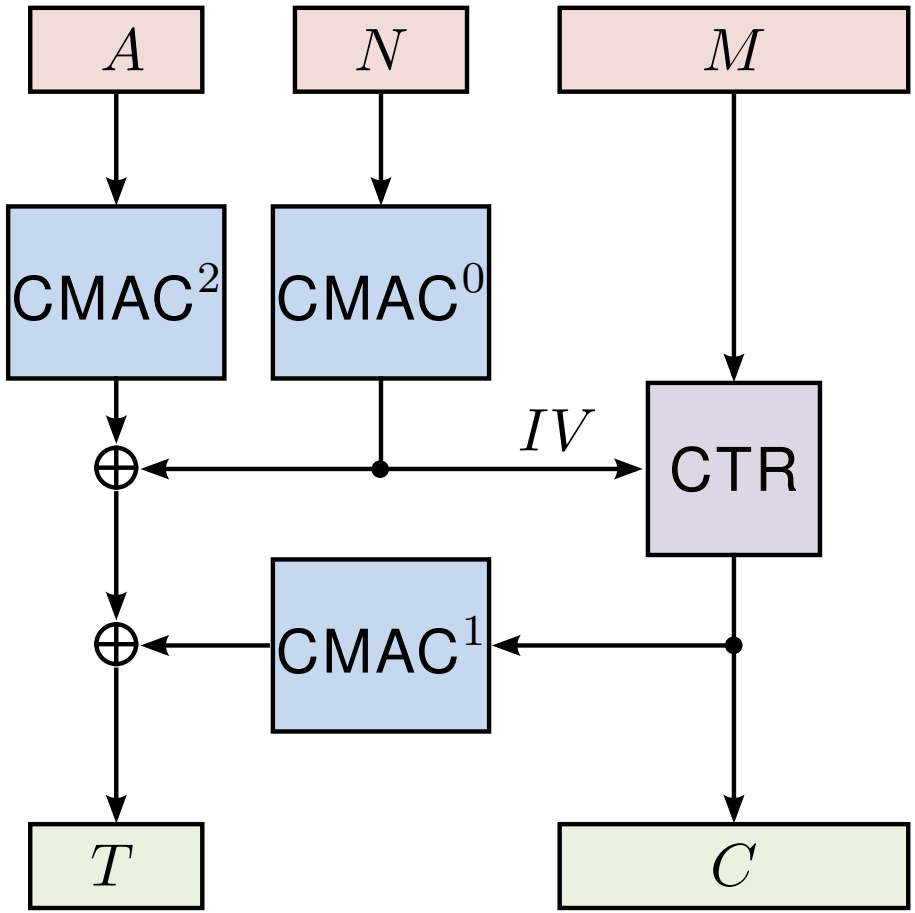
- Block-cipher-based
- “Two-pass”: one pass for encryption, one pass for authentication
- Similar to generic composition (specifically: EtM)
- Single key
- Security up to the birthday bound ($\ll 2^{n/2}$ blocks)
- Core ideas of security proofs rather straightforward

Notation

- Key K
- Nonce N
- Block cipher $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$
- Message $M = (M_1, \ldots, M_m)$
- Associated data $A = (A_1, \ldots, A_s)$
- Ciphertext $C = (C_1, \ldots, C_m)$
- Authentication tag $T \in \{0,1\}^n$ (most modes allow truncation)
- $|M| = |C| = n \cdot (m-1) + |M_m|$
- $|A| = n \cdot (s-1) + |A_s|$

EAX

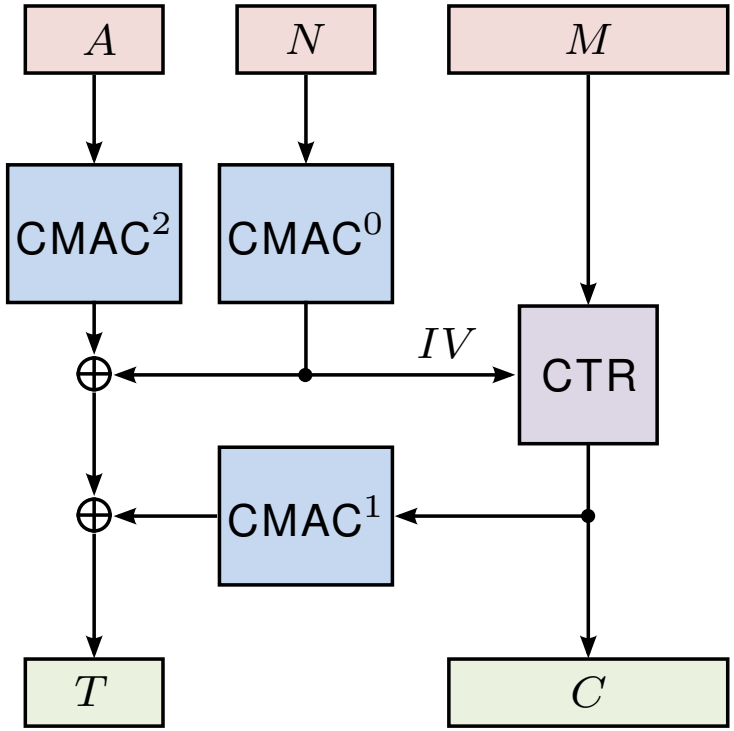
Bellare, Wagner, 2004



EAX: Core Ideas of the Security Proof (1)

IND-CPA-Security up to the Birthday Bound

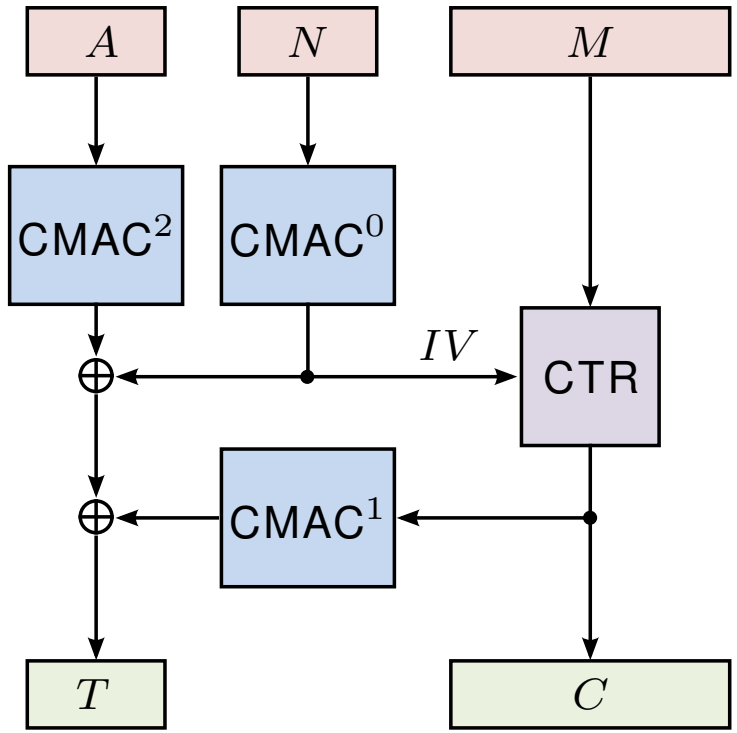
- We observe: CMAC^i-functions $\approx$ independent random functions
- Random IV for the counter mode (for a nonce-respecting adversary!)
- $\Longrightarrow$ each ciphertext is a fresh random value
- Tag: XOR of three independent random values



EAX: Core Ideas for Security Proof (2)

INT-CTXT-Security Up to the Birthday Bound

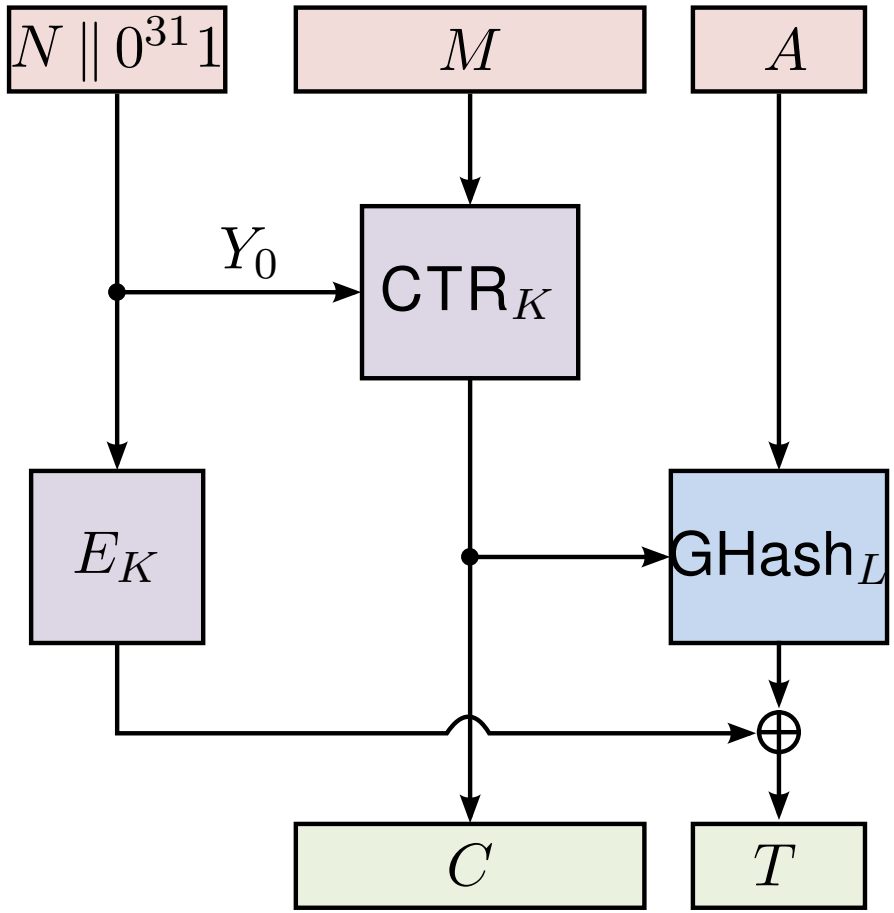
- Eve knows some tuples $(N_i, M_i, A_i, C_i, T_i)$
- **Forgery** = valid (N, A, C, T) with $(N, A, C, T) \notin \{(N_i, A_i, C_i, T_i)\}$
- If $N \notin \{N_i\}$, Eve would need to predict $\mathrm{CMAC}^0(N)$, so assume $N = N_j$ for some j.
- Similarly: $A = A_k$ for some k, and $C = C_\ell$ for some ℓ
- But not $j = k = \ell$, else $(N, A, C, T) = (N_j, A_j, C_j, T_j)$.
- T: Solution of linear equation system
- But: Too few equations and too many random unknowns



Galois Counter Mode (GCM)

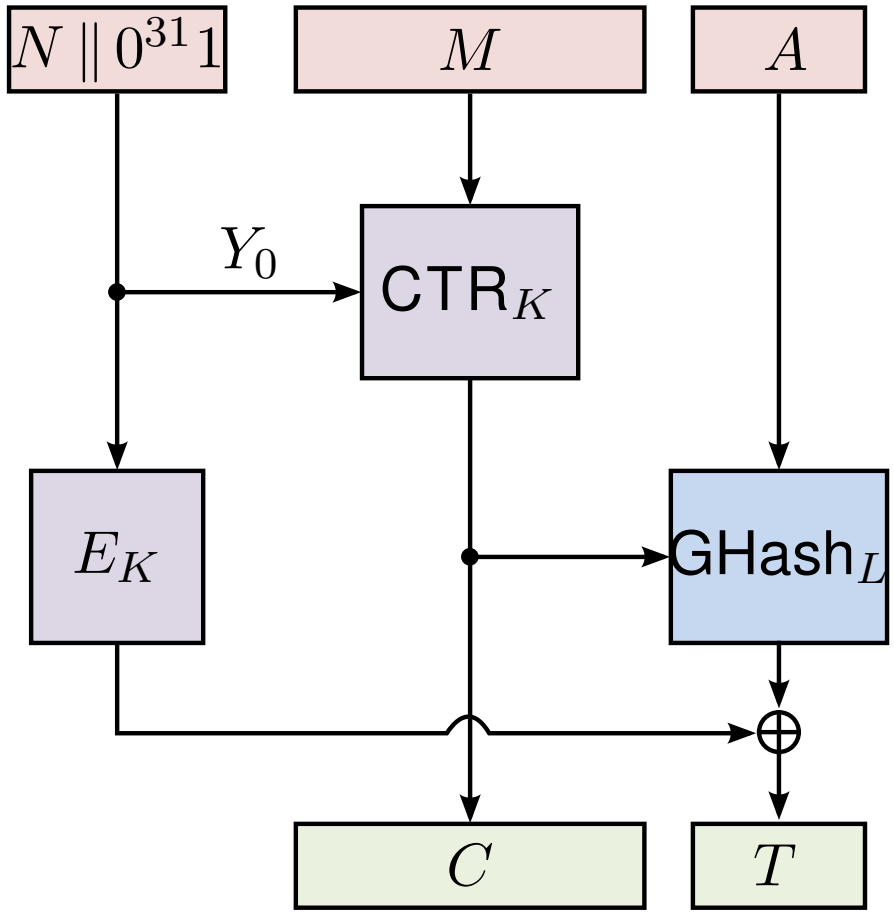
McGrew, Viega, 2004

- Single key K
- Derives hash key: $L := E_K(0^n)$
- Polynomial Hash (GHash)
- GHash appends a length-encoding block $|A| \,\|\, |M|$ for domain separation, encoded as $n/2$-bit values



Properties of GCM

- If N is nonce: Inputs to E never repeat
- 96-bit nonce N and 32-bit block counter
  $\Longrightarrow$ $|M| \leq 2^{32}$ blocks
- Since $Y_0 \neq 0^n$: $E_K(0^n)$ never used again
- Carter-Wegman MAC:
  $E_K(N \,\|\, 0^{31}1) \oplus$
  $\mathsf{GHash}_L(C, A)$



GCM: GHash

Definition 40 (Almost-XOR-Universal Hash Functions)

Let $\mathcal{H} = \{H \mid H : \{0,1\}^* \to \{0,1\}^n\}$ be a family of hash functions. $\mathcal{H}$ is called **ϵ-almost-XOR-universal** (ϵ-**AXU**) iff for $H \xleftarrow{\$} \mathcal{H}$ and all distinct $X, X' \in \{0,1\}^*$ and $\Delta \in \{0,1\}^n$

$$\Pr\left[H(X) \oplus H(X') = \Delta\right] \leq \epsilon.$$

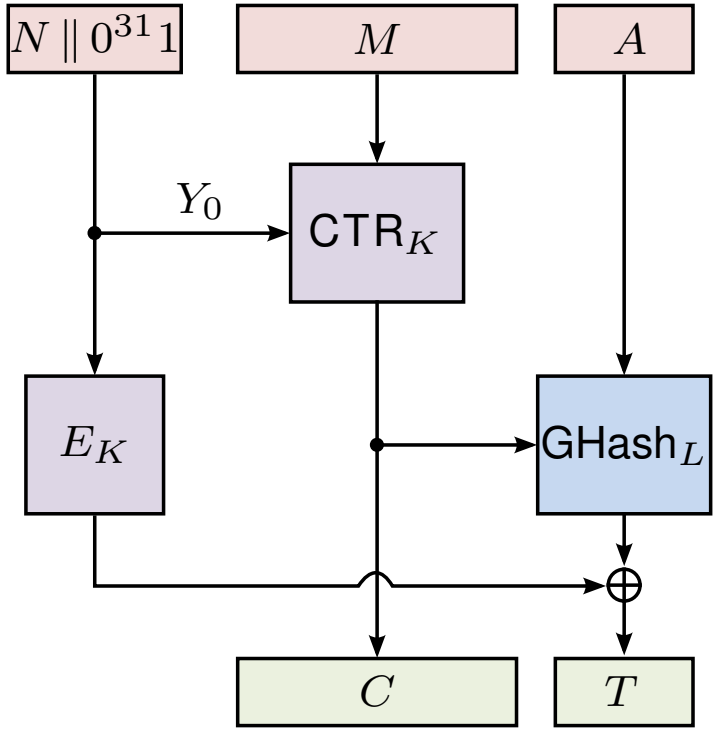
- AXU is equivalent to AΔU when computing in GF(2^n))
- Secret key $L \in$ GF(2^n), fixed input length of m blocks:

$$\mathsf{GHash}_L(X_1, \ldots, X_m) = \bigoplus_{i=1}^{m} X_i \cdot L^i$$

- Fundamental Theorem $\Rightarrow$ GHash is ϵ-AXU with $\epsilon \leq m/2^n$.

GCM Security (1)

- Encrypt-then-MAC with "independent" keys K and L ($E_K(0^n)$ can never occur elsewhere)
- Iwata, Ohashi, Minematsu (2012):
  - q queries of σ blocks in total, maximum input length ℓ
  - Probability to break GHash $\leq q\ell/2^n$ ($\rightarrow$ Thm. **??**)



$$\mathbf{Adv}^{\text{IND-CPA}}_{\text{GCM}}(q,\ell,\sigma) \leq \frac{(q+\sigma+1)^2}{2^{n+1}}$$

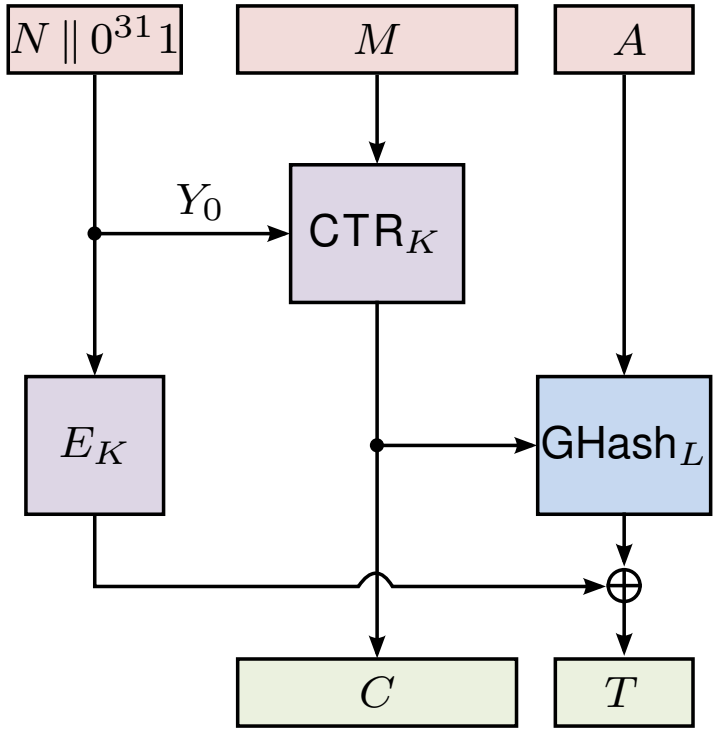
$$\mathbf{Adv}^{\text{INT-CTXT}}_{\text{GCM}}(q,\ell,\sigma) \leq \frac{(q+\sigma+1)^2}{2^{n+1}} + \frac{q\ell}{2^n}$$

GCM Security (2)

- GCM also supports smaller nonces:

$$N \leftarrow \mathsf{GHash}_L(N') \bmod 2^{96}$$

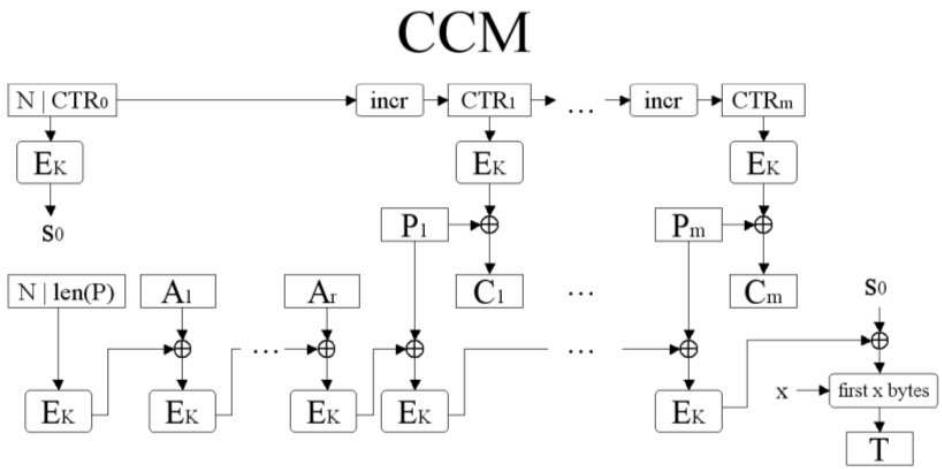
- But: Security bounds become **much worse**
- Hardly ever used in practice (fortunately)



Other Two-Pass Modes

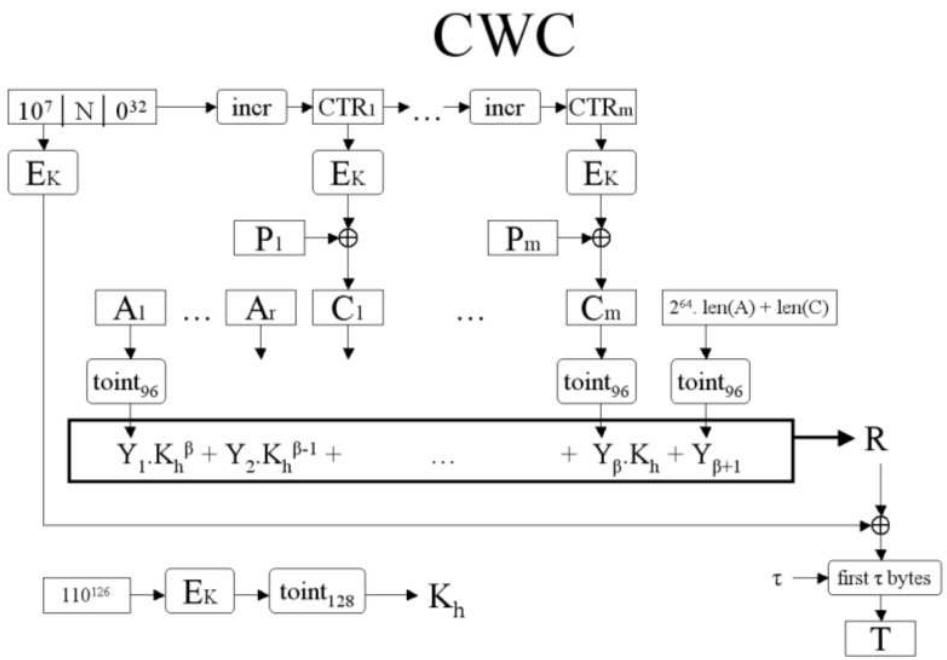
CCM: Counter with CBC-MAC
(Whiting, Housley, Ferguson, 2003)

- MAC-then-Encrypt
- Two-pass, $|M|$ required early



CWC: Counter with Carter-Wegman
(Kohno, Viega, Whiting, 2004)

- Encrypt-then-MAC
- Two-pass
- Uses polynomial hash
- Modulo $2^{128} - 1$



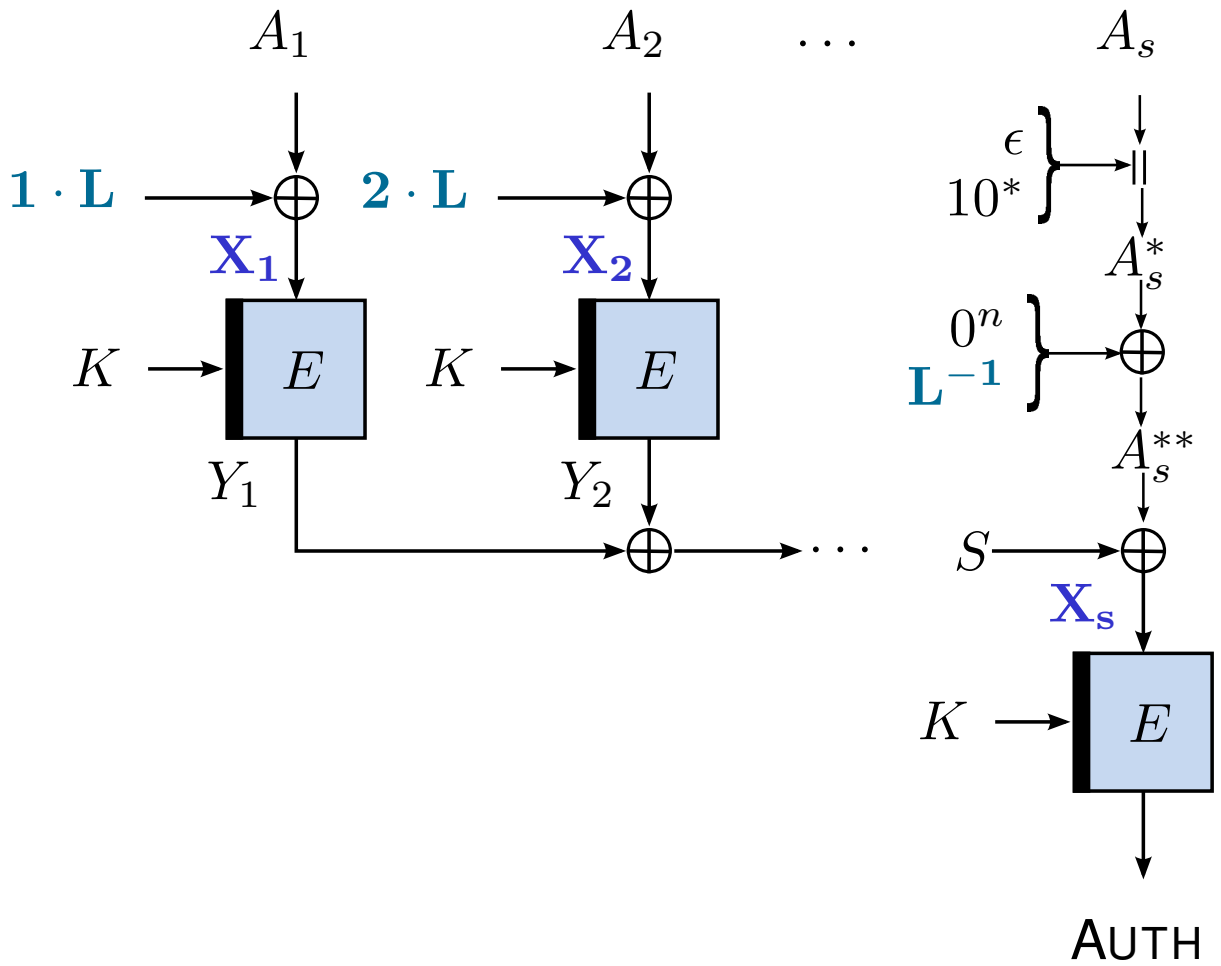
5.2: OCB: A Single-Pass Mode

“Offset Codebook” Rogaway et al, 2001, 2004, 2010; One-Pass

- Actually predates most two-pass modes
- Complex patent situation (three different parties claim a patent)
- A bit more complex than single-pass modes, but more efficient
- Three versions:
  - Three different ways to compute the “offsets”
  - OCB1 did not yet support associated data
  - Beyond that not many differences
  - Below: we will focus on OCB3

OCB is the AE-sibling of PMAC

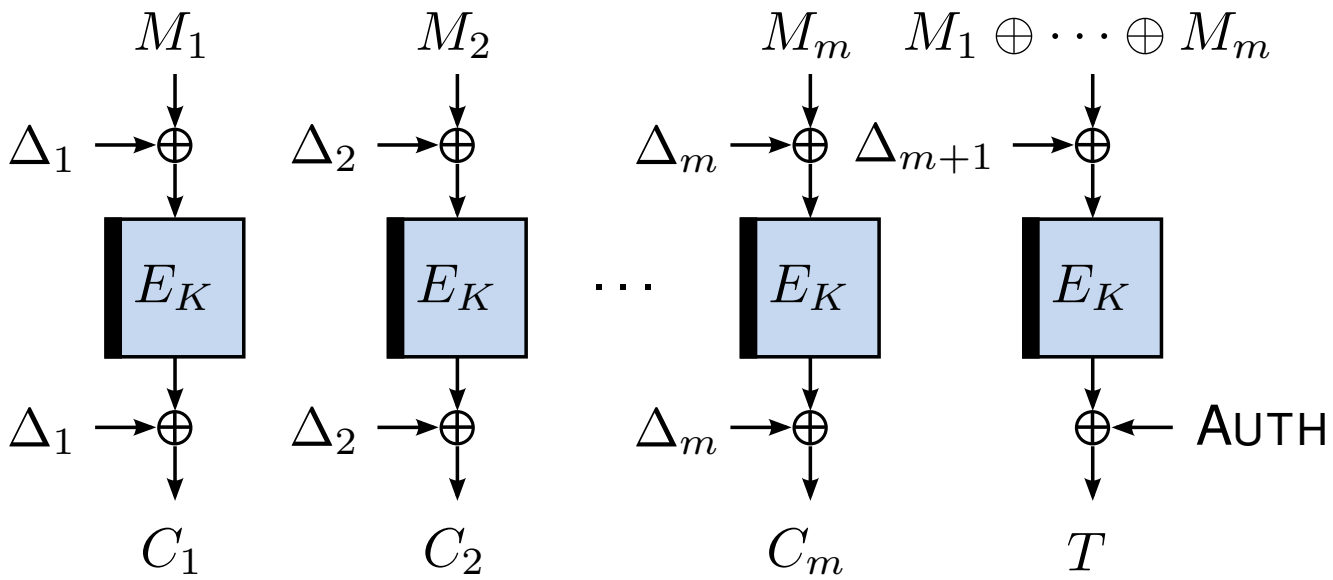
Simplified



OCB

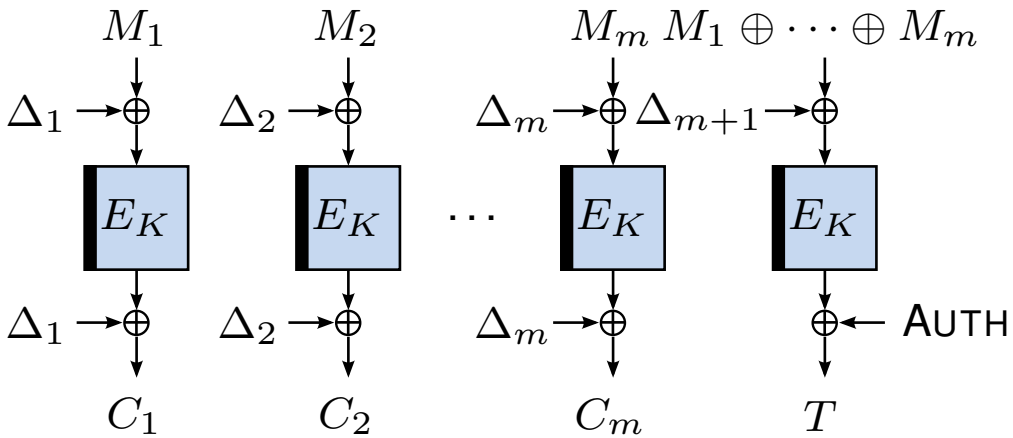
Ignoring the Case of Partial M_m

- $L \leftarrow E_K(0^n)$
- $L_0 \leftarrow 2^2 L$
- $L_i \leftarrow 2L_{i-1}$
- $\Delta_0 \leftarrow F_K(\mathsf{N}, |T|)$
- $\Delta_i \leftarrow \Delta_{i-1} \oplus L_{\mathsf{ntz}(i)}$



Remarks

- Like the two-pass modes we presented, OCB is secure up to the birthday bound.
- Security analysis is similar to that of PMAC
- The multiples of $\mathbf{L}$ depend only on the key
- Like PMAC, $\mathbf{i} \cdot \mathbf{L}$ actually is $\mathbf{g_i} \cdot \mathbf{L}$, where $\mathbf{g_i}$ is a grey code
- $\boldsymbol{\Delta}$ depends on the key and the nonce
- When the nonce is a counter, the computation of $\boldsymbol{\Delta}$ only needs a block-cipher call for every 64-th nonce



5.3: EAX-Prime

- A *minor modification* of EAX (needs slightly less storage)
- Adapted as the AE mode of operation for the proposed smart grid standard from ANSI,
- **But**: security proof is no longer applicable
- **And**: EAX′ was actually broken ($\rightarrow$ separate slides).

This is yet another shocking case of so-called “practitioners” neglecting cryptographers’ knowledge:

> *I still think standards are preferable in theory, but only if they’re promulgated by reasonable standards bodies. And we seem to have a shortage of those.*
>
> *[. . .]*
>
> *How can I advocate for crypto standards when standards bodies will casually throw away something as wonderful as a security proof?*

– Matthew Green

5.4: Discussion: Which Mode is the Best?

- **OCB**:
  - Very fast, almost fully parallelizable
  - A bit more complex to understand and implement
  - Patent situation may be an issue
- **GCM**:
  - Has become quite popular
  - Fastest two-pass mode on most machines
- **EAX**:
  - May be the easiest to implement
- In any case, take great care, if you think of creating your own!
- At the end: All the modes presented here are OK.
- Except for **EAX**$'$, of course.
- But all these modes break apart when used improperly
  ($\rightarrow$ next chapter)