Authenticated Encryption

Andrey Bogdanov
Technical University of Denmark
June 2, 2014

Scope

- Main focus on modes of operation for block ciphers
- Permutation-based designs briefly mentioned

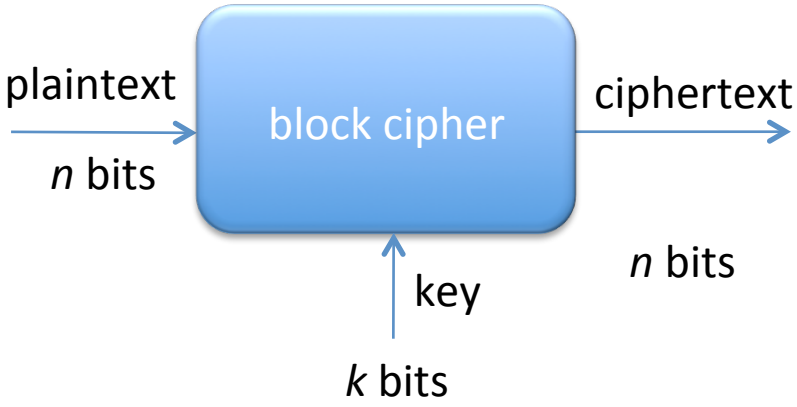
Outline

- Block ciphers
- Basic modes of operation
- AE and AEAD
- Nonce-based AE modes and features
- Nonce-based AE: Implementation properties
- Nonce-free AE modes and features
- Nonce-free AE: Implementation properties
- Permutation-based AE

Outline

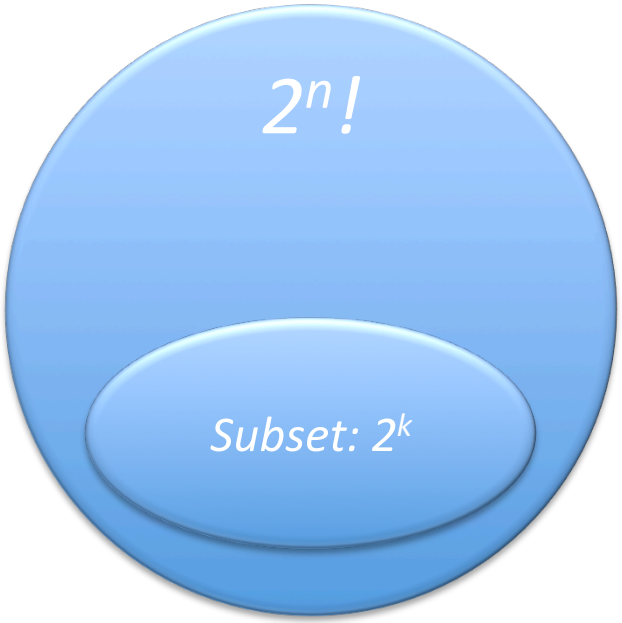
- **Block ciphers**
- Basic modes of operation
- AE and AEAD
- Nonce-based AE modes and features
- Nonce-based AE: Implementation properties
- Nonce-free AE modes and features
- Nonce-free AE: Implementation properties
- Permutation-based AE

Block ciphers

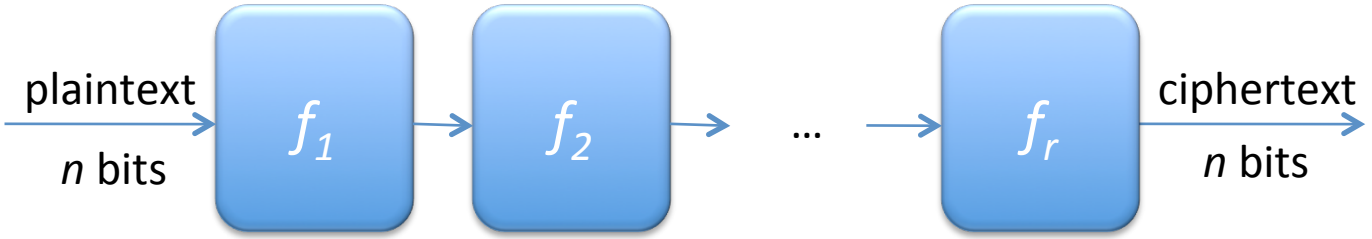


Block cipher

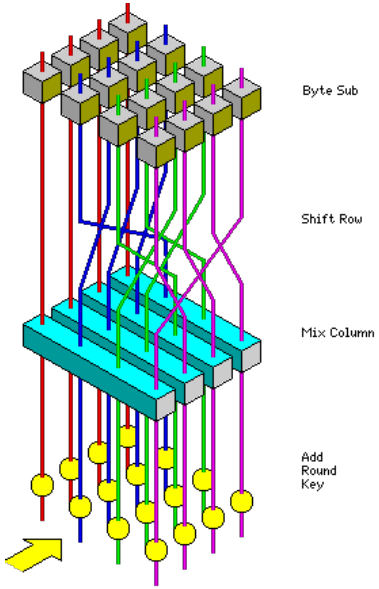
A block cipher with n-bit block and k-bit key is a subset of 2^k permutations among all $2^n!$ permutations on n bits.



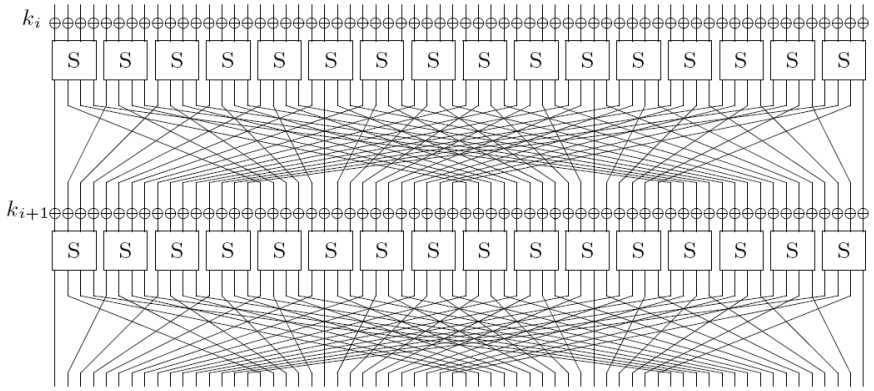
Some standard block ciphers



AES



PRESENT



Visualization of a round transform

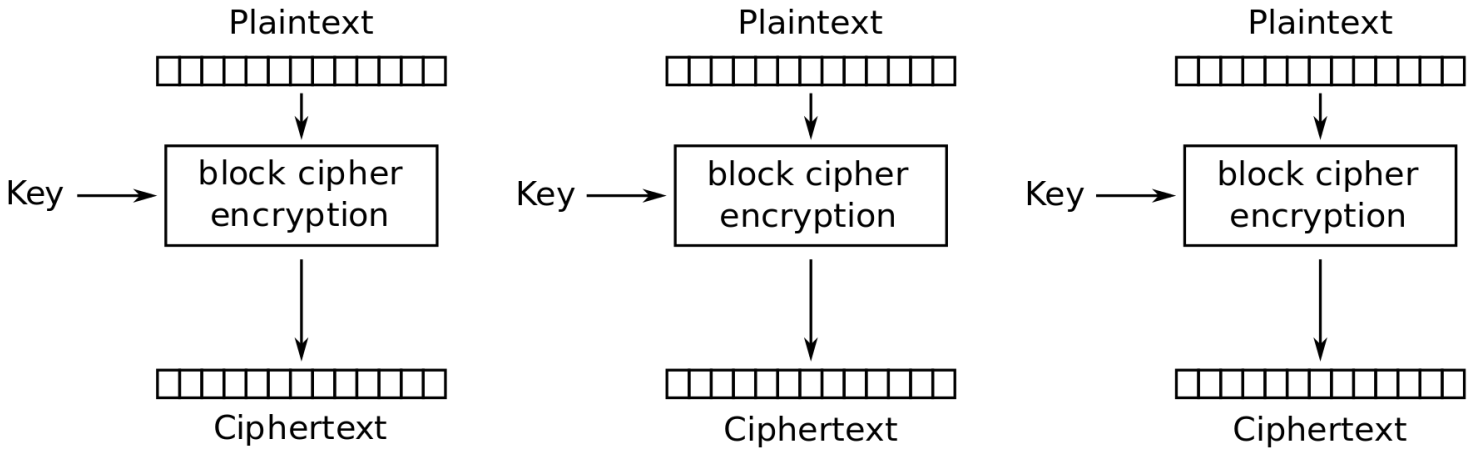
Why block ciphers?

- Most basic security primitive in nearly all security solutions, e.g. used for constructing
  - stream ciphers,
  - hash functions,
  - message authentication codes,
  - **authenticated encryption algorithms,**
  - entropy extractors, …
- Probably the best understood cryptographic primitives
- U.S. symmetric-key encryption standards and recommendations have block ciphers at their core: DES, AES

Modes of operation

- The block cipher itself only encrypts one block of data
  - Standard and efficient block ciphers such as AES
- To encrypt data that is not exactly one block
  - Switch a block cipher into a mode of operation

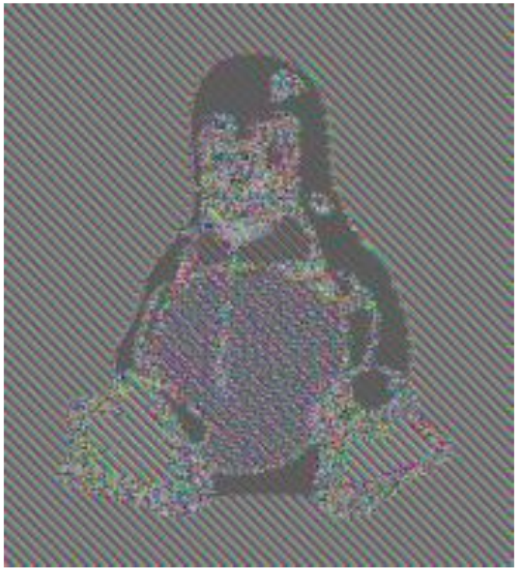
Electronic Code Book (ECB)



Electronic Codebook (ECB) mode encryption

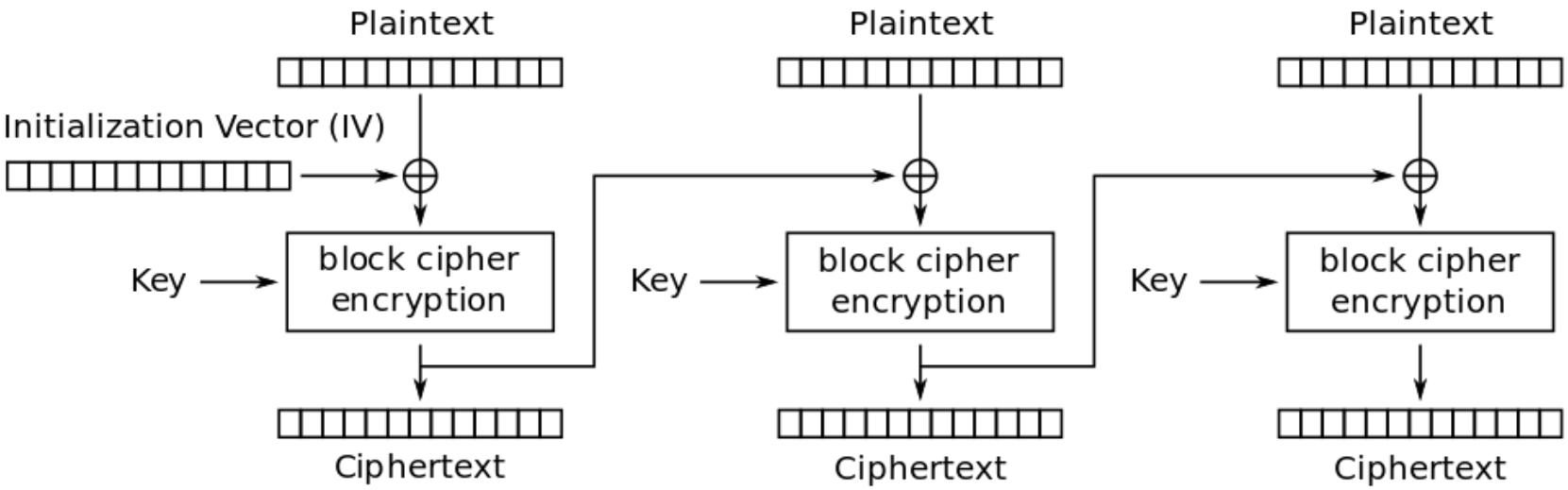
WhiteTimberwolf, Wikimedia Commons

Electronic Code Book (ECB)



- Good performance and parallelizability
- But retains patterns and repetitions
- Limited number of applications

Cipher Block Chaining (CBC)

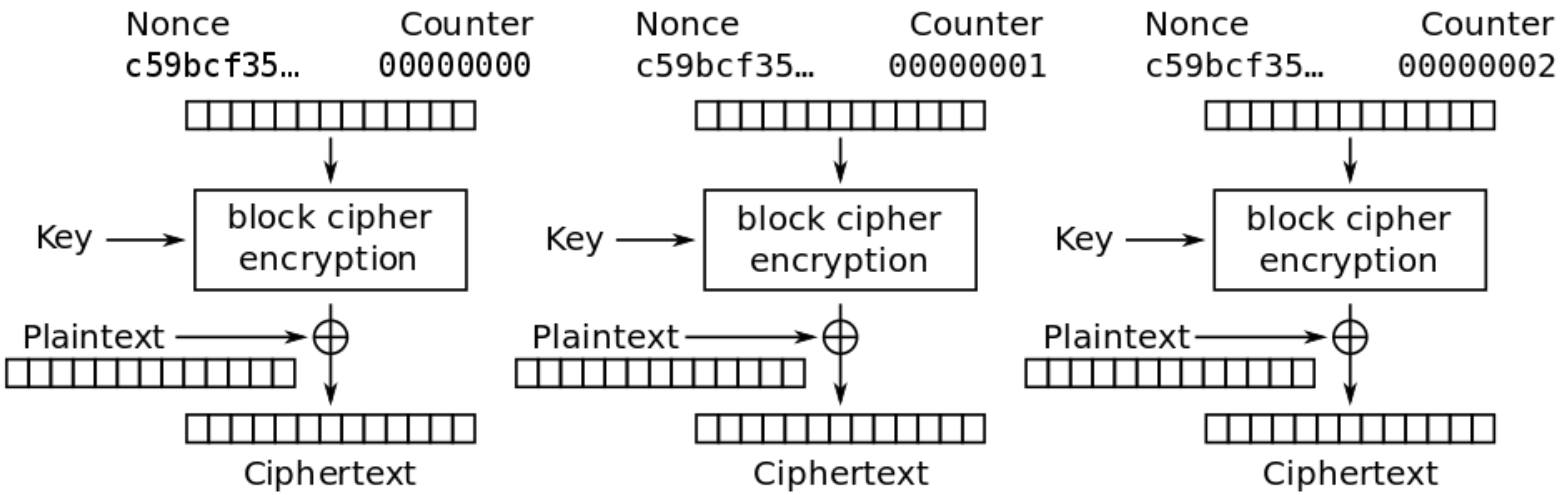


Cipher Block Chaining (CBC) mode encryption

- Serial encryption
- Parallel decryption
- Needs both cipher enc and dec for both enc and dec
- A MAC can be constructed from CBC: CBC-MAC

WhiteTimberwolf, Wikimedia Commons

Counter mode (CTR)



Counter (CTR) mode encryption

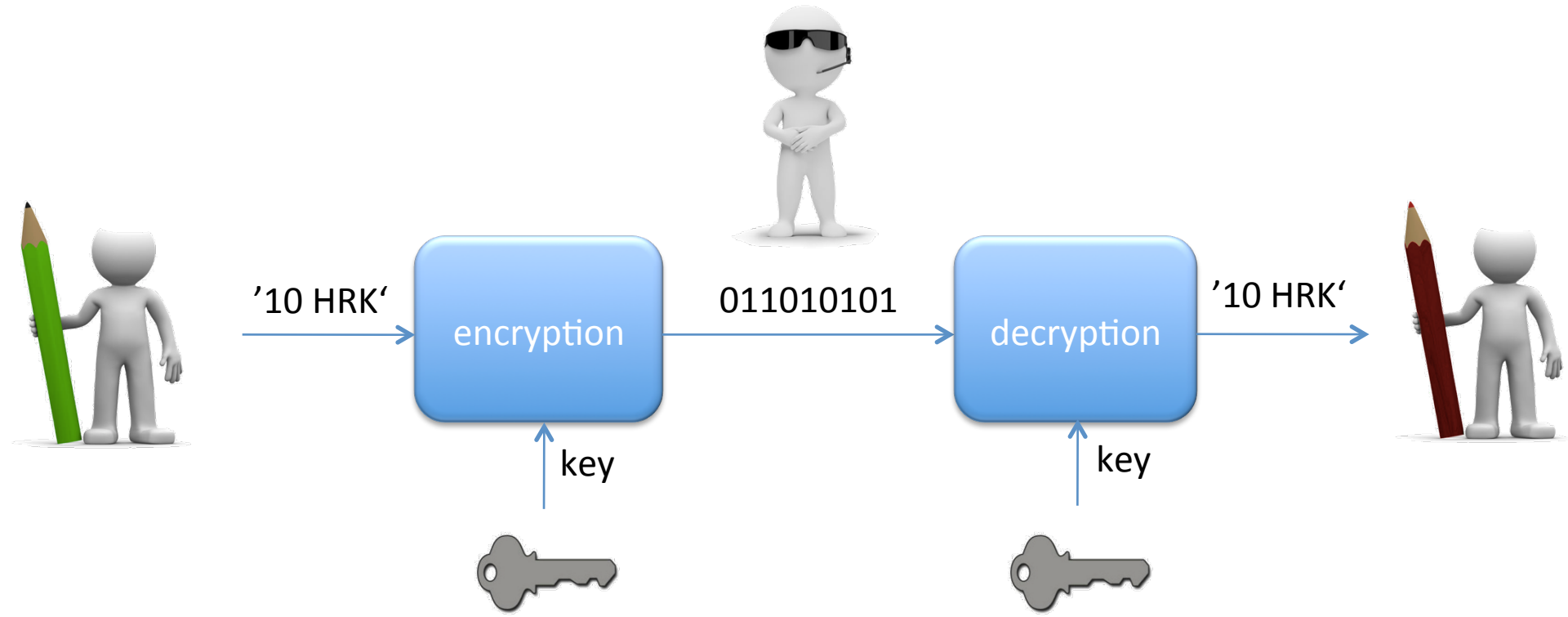
- Essentially relies on IV
- Parallel encryption and decryption
- Needs only cipher enc for both enc and dec

WhiteTimberwolf, Wikimedia Commons

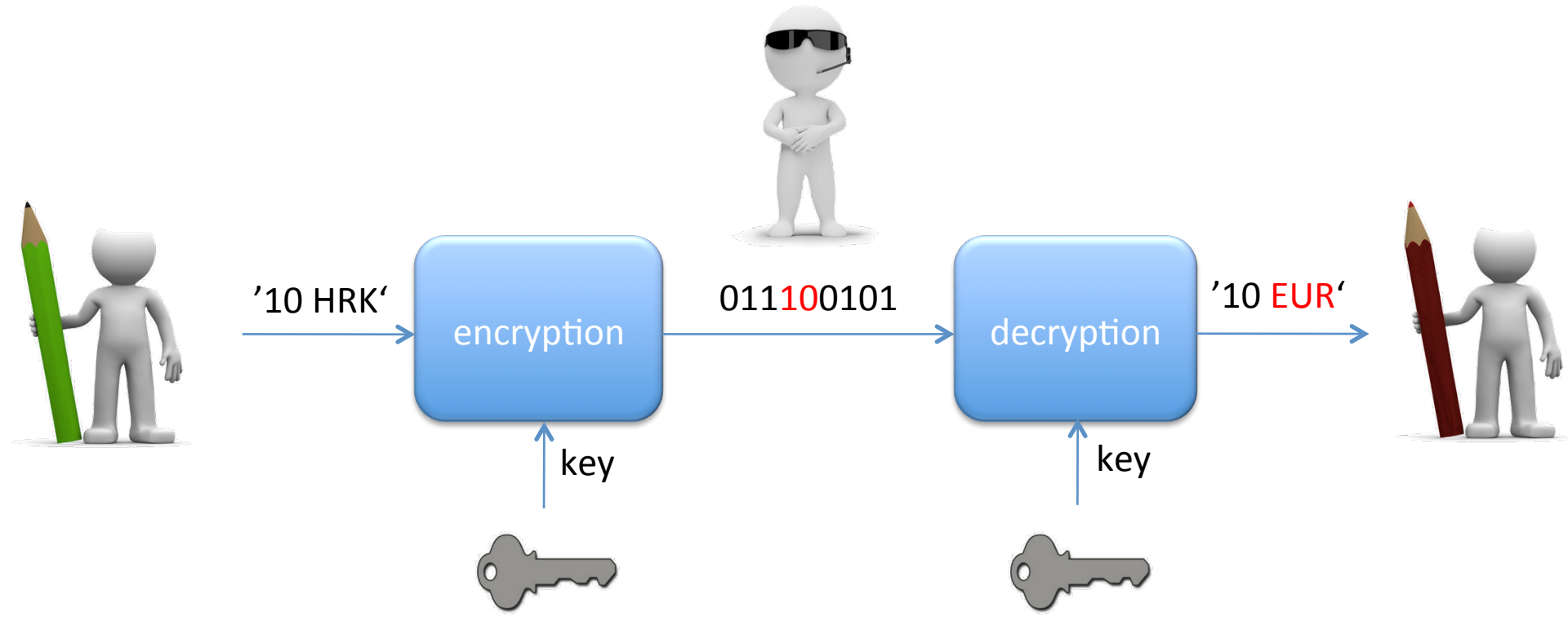
Outline

- Block ciphers
- Basic modes of operation
- **AE and AEAD**
- Nonce-based AE modes and features
- Nonce-based AE: Implementation properties
- Nonce-free AE modes and features
- Nonce-free AE: Implementation properties
- Permutation-based AE

OK, what about authenticity?



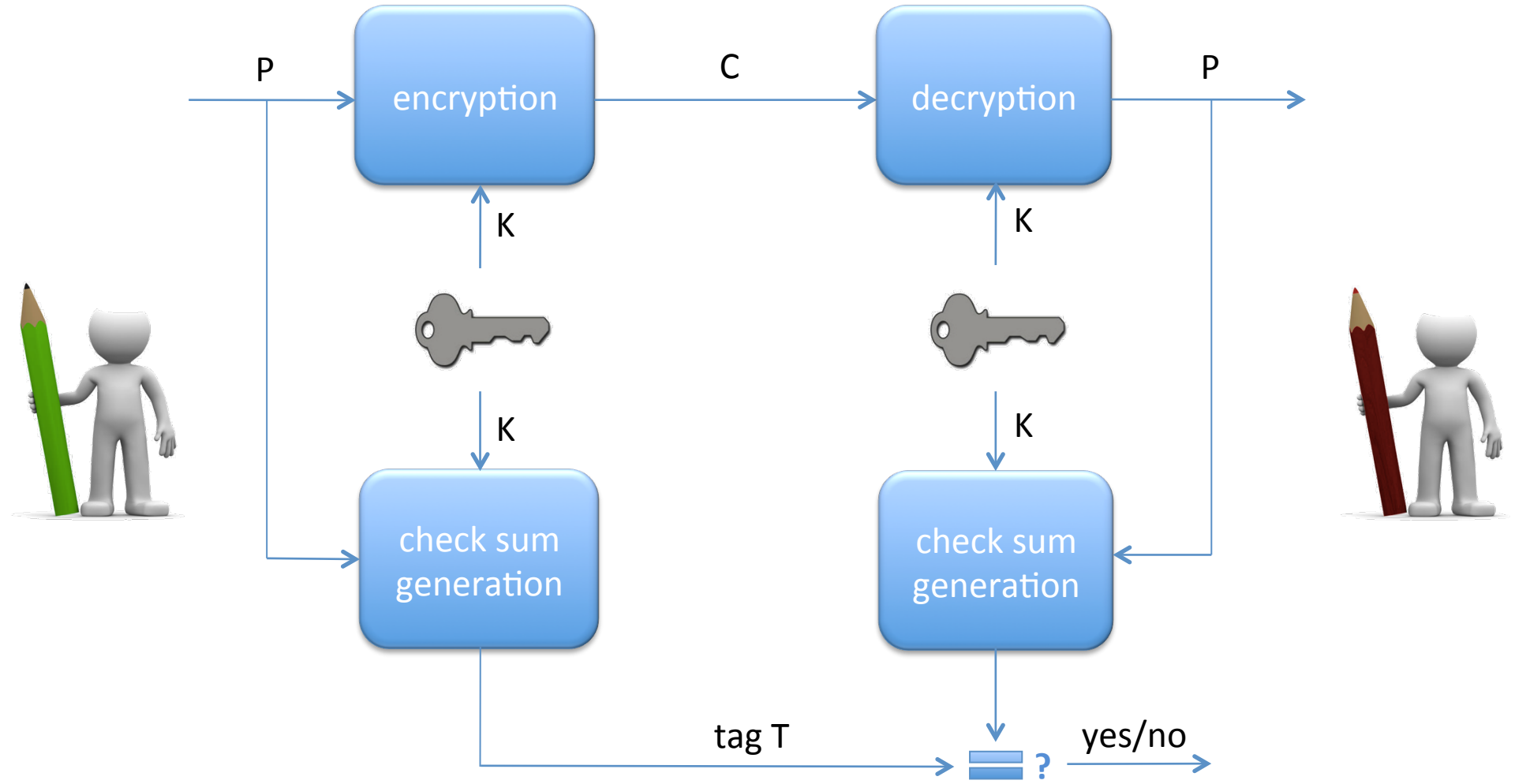
OK, what about authenticity?



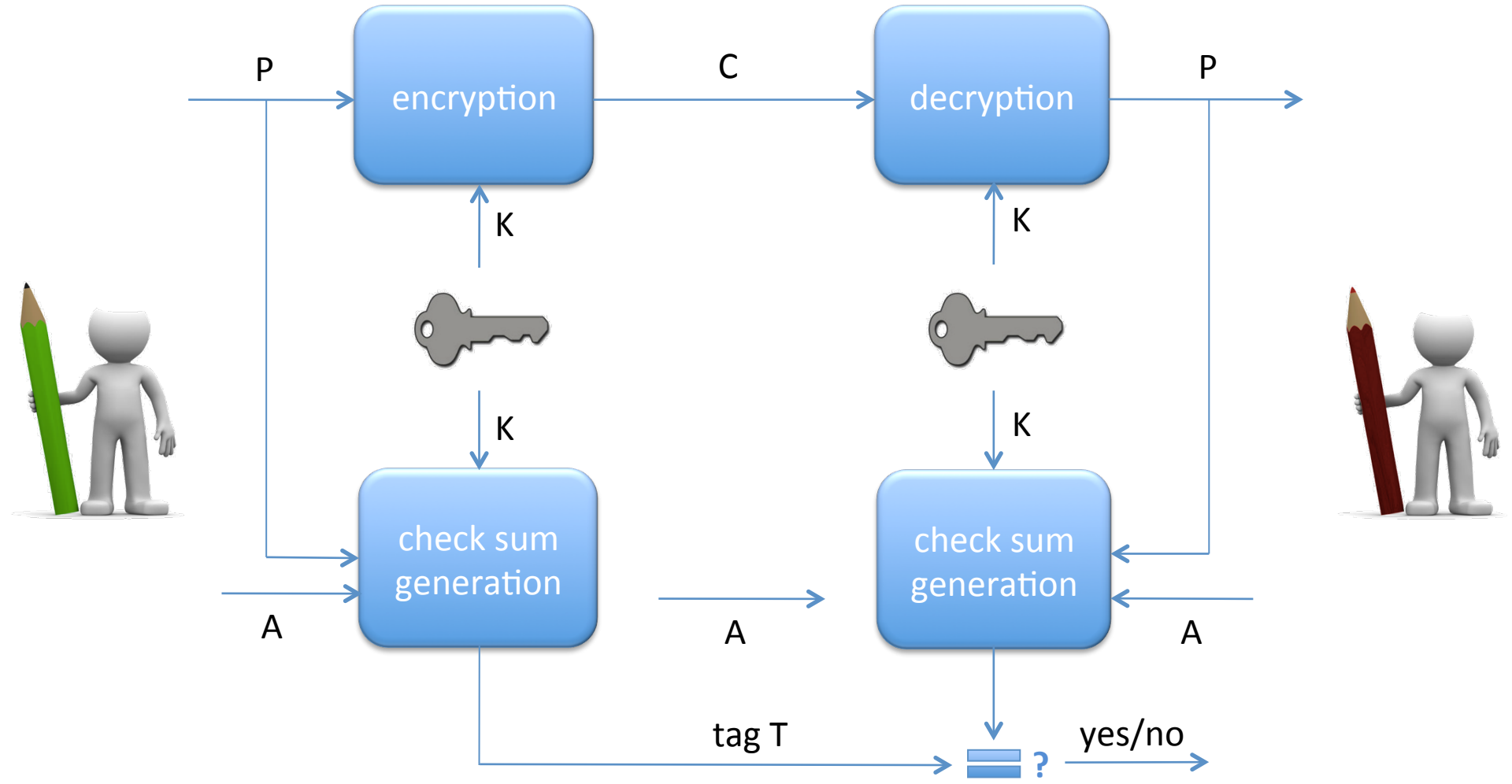
Authenticity

- Is cryptography about secrets?
  - Yes, but not only!
  - Encryption alone is not enough
  - **Authenticity is essential**

Authenticated Encryption (AE)



Authenticated Encryption with Associated Data (AEAD)



AE vs AEAD

- Authenticated encryption

  AE: (P,K) -> (C,T)
  where T is authentication tag

- Authenticated encryption with associated data

  AEAD: (A,P,K) -> (A,C,T)
  where A is associated data transmitted in plaintext

- What is the use of associated data?
  - Routing information
  - Packet headers

CAESAR competition for authenticated encryption

- **CAESAR** = *Competition for Authenticated Encryption: Security, Applicability, and Robustness*
- Following NIST AES, EU NESSIE, EU eStream, and NIST SHA-3
- Submissions were due: March 2014
- Co-funded by US NIST
- Need for efficient authenticated encryption

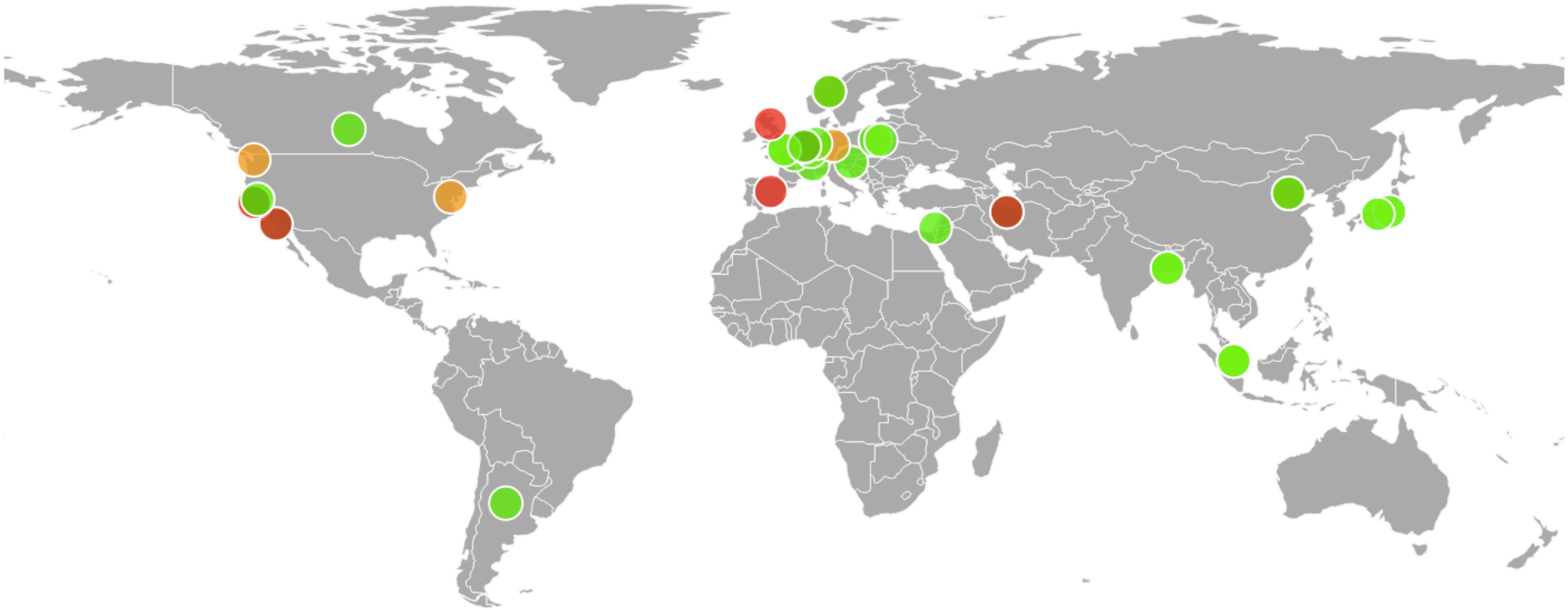
CAESAR competition for authenticated encryption

- 57 submissions from all around the world
- serious flaws identified for 11 submissions
- less serious issues for 5 more submissions
- 8 submission withdrawn, fully or partly

CAESAR competition for authenticated encryption

- 57 submissions from all around the world
- serious flaws identified for 11 submissions
- less serious issues for 5 more submissions
- 8 submission withdrawn, fully or partly
- names of 2 submissions are Pokemon names!

CAESAR competition for authenticated encryption



Authenticated Encryption Zoo: https://aezoo.compute.dtu.dk

CAESAR competition for authenticated encryption

- Jan 2015: Announcement of round-2 candidates
- Dec 2015: Announcement of round-3 candidates
- Dec 2016: Announcement of finalists
- Dec 2017: Announcement of final portfolio

Some features of AE modes

- Nonce-based vs nonce-free
- Parallel vs serial
- One BC call vs two BC calls per block
- One-pass vs two-pass
- Online vs offline
- Deterministic vs common prefix
- Many more

[M12]

AE: some modes and standards

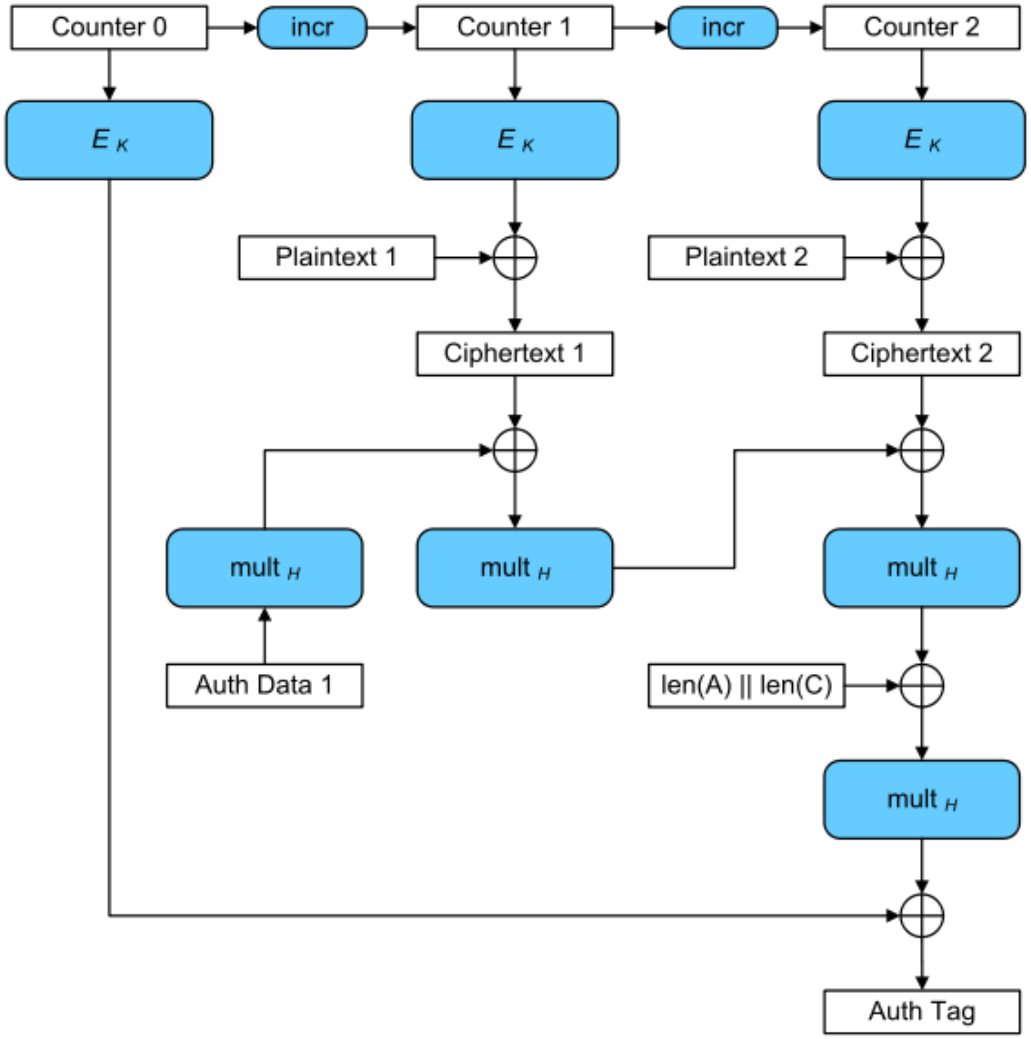
	Algorithms	Standards
1999	IAPCBC	
2000	IACBC, AE	
2001	OCB, AEAD	
2002	CCM	802.11
2003		
2004	GCM	802.1
2005		IPsec
2006		FC-SP, 1619.1, LTO-4
2007		
2008		RFC5116
2009	SIV	TLSv1.2, IKE, XMLsec, SSH
2010		
2011	OCBv3	
2012	CBC+HMAC	SRTP, *JOSE*

Outline

- Block ciphers
- Basic modes of operation
- AE and AEAD
- **Nonce-based AE modes and features**
- Nonce-based AE: Implementation properties
- Nonce-free AE modes and features
- Nonce-free AE: Implementation properties
- Permutation-based AE

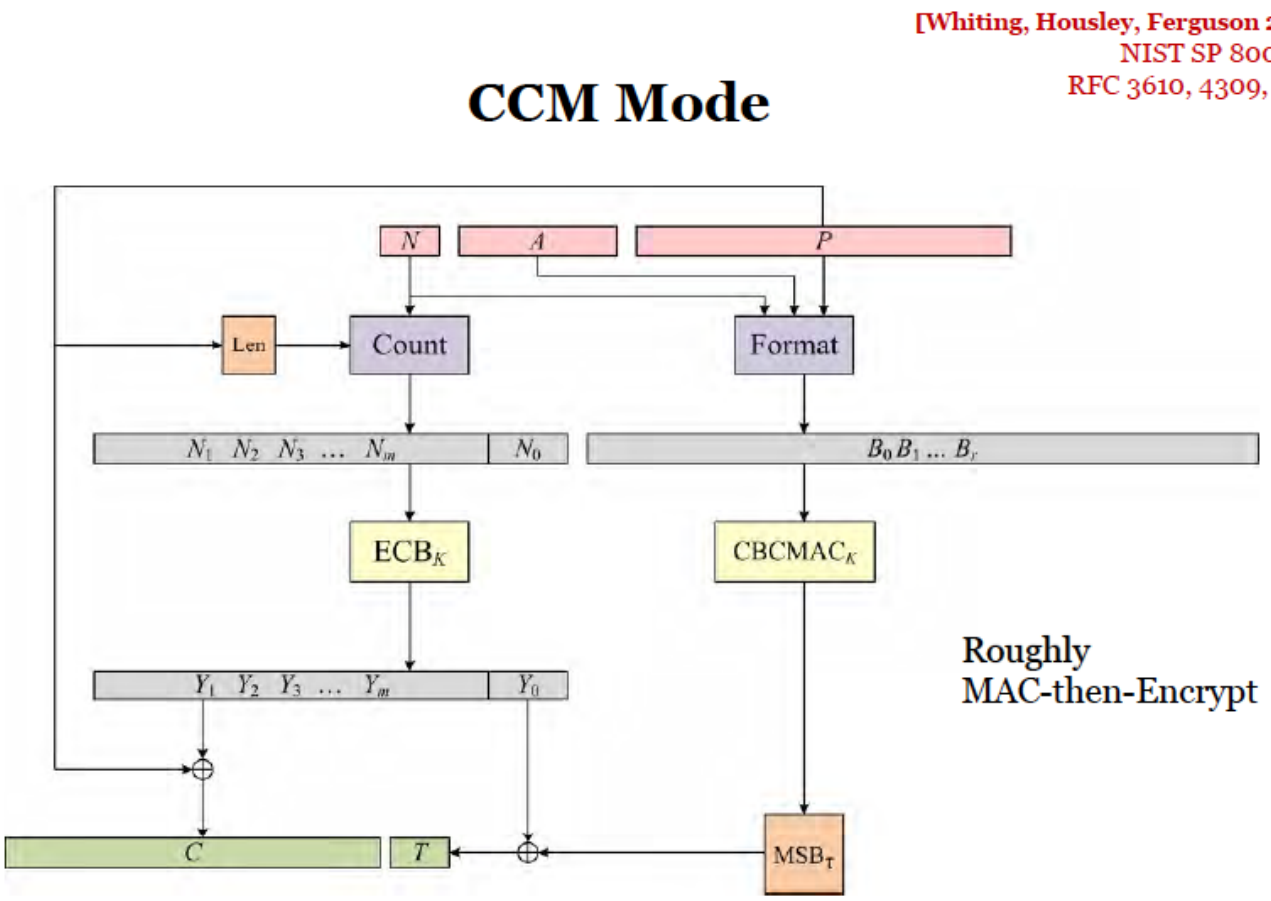
[MV04]

GCM: Galois/Counter Mode



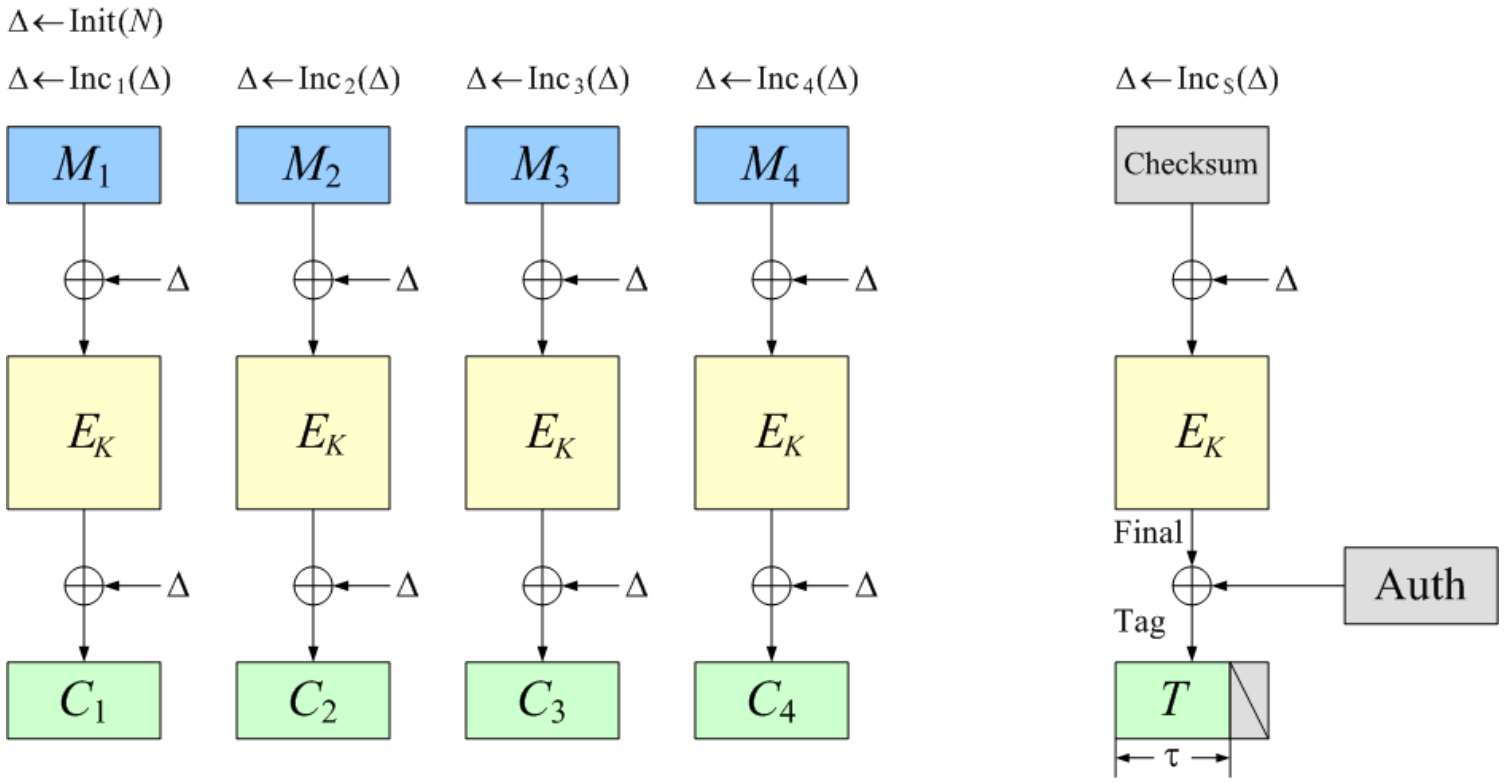
CCM: Counter with CBC-MAC

[Whiting, Housley, Ferguson 2002]
NIST SP 800-38C
RFC 3610, 4309, 5084



[RBBK01]
[BR02]
[R02]
[R04]
[KR11]

OCB: Offset Codebook Mode



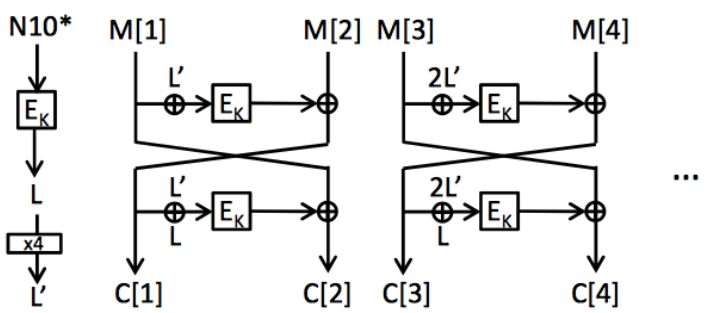
+

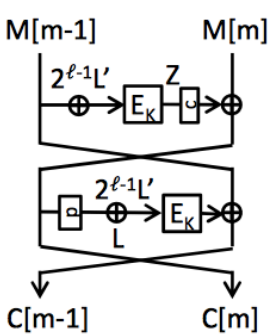
- 1 AES-128 call per block
- well parallelizable
- associated data
- online scheme

-

- enc/dec different
- state 4x128 bits
- (patents)

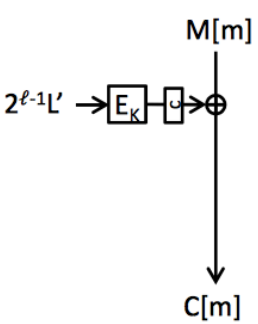
OTR: Offset Two Round





When m is even

$L_{last} = 2^{\ell-1}L' \oplus L$

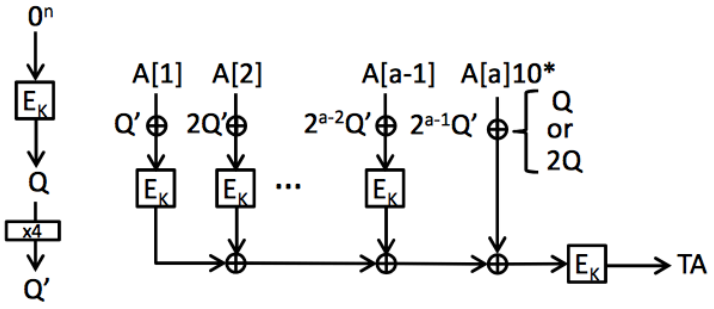


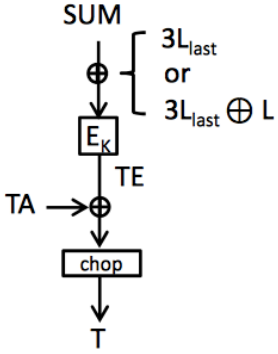
When m is odd

$L_{last} = 2^{\ell-1}L'$

SUM = M[2] ⊕ M[4] ⊕ ... ⊕ M[m-2] ⊕ Z ⊕ C[m] 10* if m even

= M[2] ⊕ M[4] ⊕ ... ⊕ M[m-1] ⊕ M[m]10* if m odd





+

- 1 AES-128 call per block
- well parallelizable
- only BC enc is needed for both OTR enc & dec

-

- enc/dec different
- Feistel limits parallelizability

Outline

- Block ciphers
- Basic modes of operation
- AE and AEAD
- Nonce-based AE modes and features
- **Nonce-based AE: Implementation properties**
- Nonce-free AE modes and features
- Nonce-free AE: Implementation properties
- Permutation-based AE

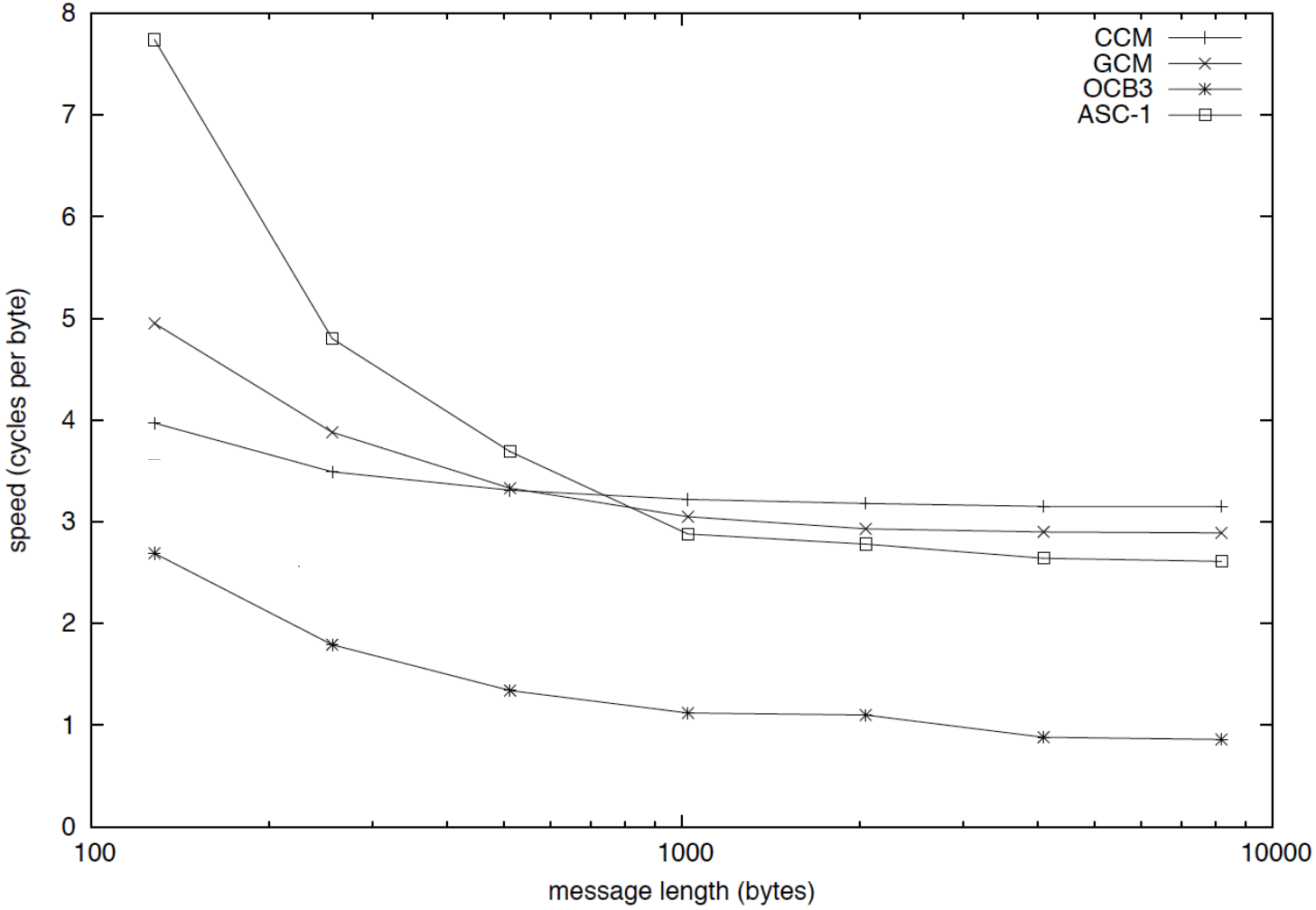
Nonce-based AE

- OCB by Rogaway et al. is hard to beat in software!
  - parallelizable (extremely fast with AES instructions)
  - virtually single cipher call per block
  - small overhead (especially with stretching)

[BMRRT'13]

Nonce-based AE in software

AES-NI Sandy Bridge, cycles per byte



Software implementation of OCB

High-speed data links: up to 100 Gbit/s AE needed (McGrew)

Standard Sandy Bridge desktop CPUs with 6 cores and AES-NI @ 3.1 GHz available

Thus, for 1KByte as average size of a message

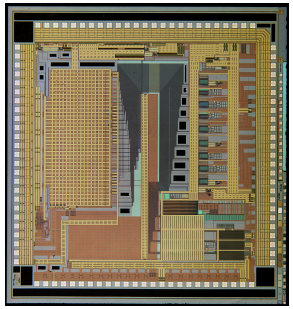
- OCB3: 132.8 Gbit/s

Do we really need it faster?

NOTE:

State-of-the-art 16-lane PCI Express 3.0 slot has a capacity of about 120 Gbit/s

[BMRRT'13]



Nonce-based AE in lightweight hardware

Design	Area (GE)	Net per 128-bit block (clock cycles)	Overhead per message (clock cycles)	Power (uW)
AES-ECB	2,435	226	-	87.84
AES-OCB2	4,563	226	452	165.21
AES-OCB2 e/d	5,783	226	452	201.32
ASC-1 A	4,793	370	904	169.11
ASC-1 A e/d	4,964	370	904	193.71
ASC-1 B	5,517	235	904	199.02
ASC-1 B e/d	5,632	235	904	207.13
AES-CCM	3,472	452	-	128.31
AES-CCM e/d	3,765	452	-	162.15

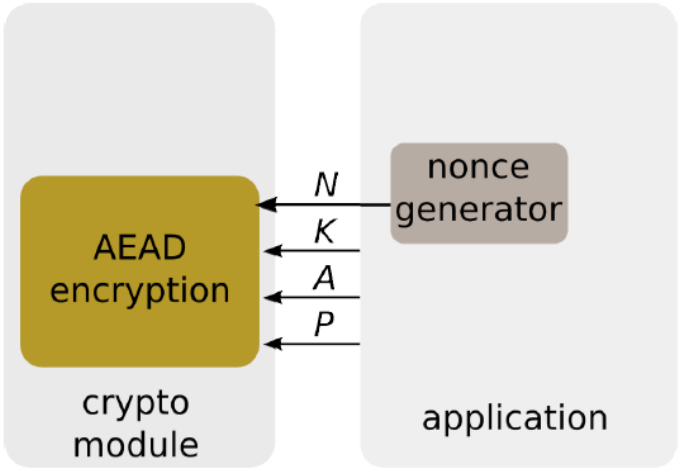
STMicroelectronics 65 nm CMOS LP-HVT, Synopsis 2009.06, 20 MHz

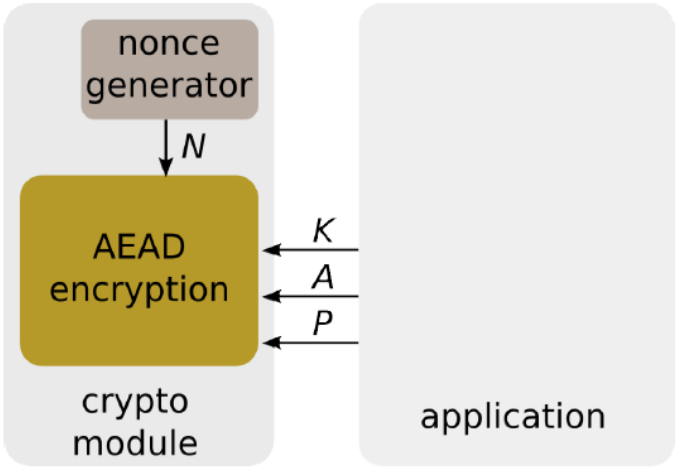
Outline

- Block ciphers
- Basic modes of operation
- AE and AEAD
- Nonce-based AE modes and features
- Nonce-based AE: Implementation properties
- **Nonce-free AE modes and features**
- Nonce-free AE: Implementation properties
- Permutation-based AE

Nonce-free vs nonce-based

- Nonce N = number used once, freshness
- Nice but might be difficult to enforce sometimes



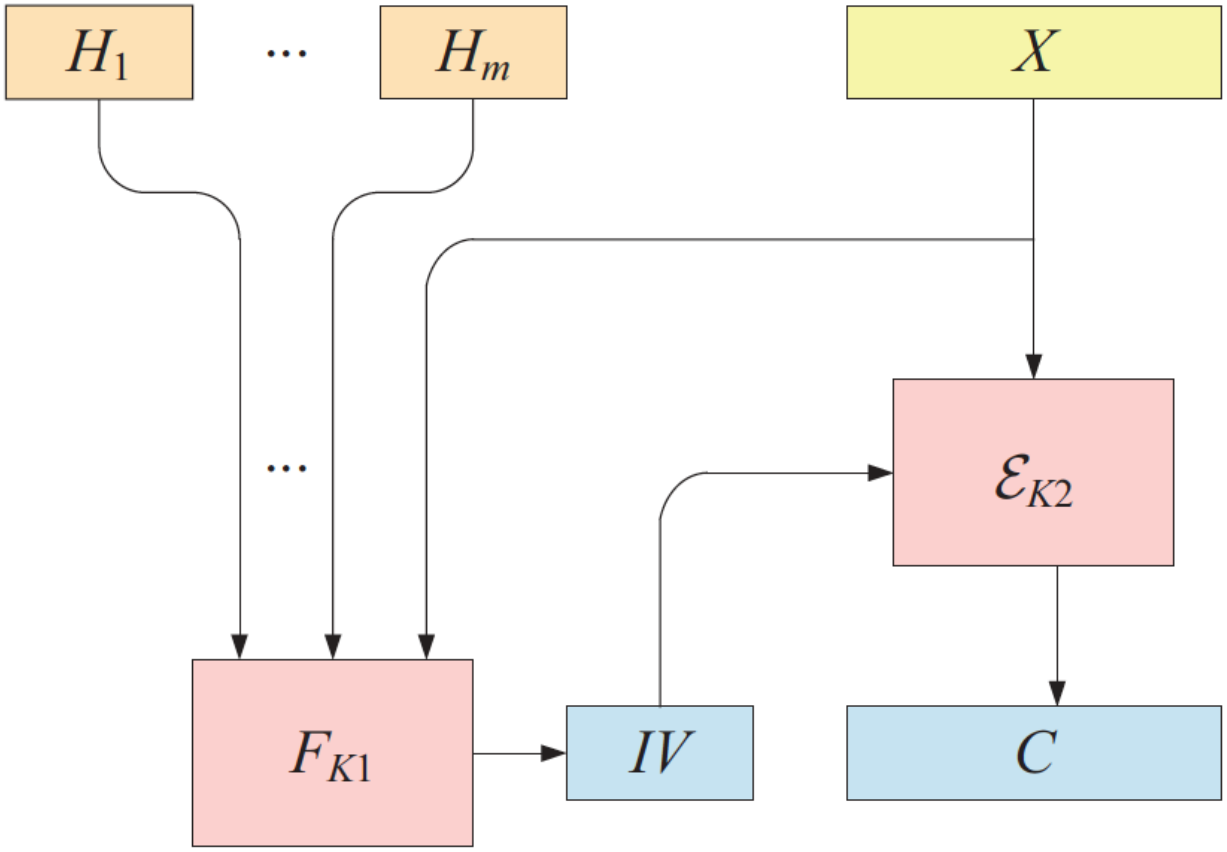


- Good news: Nonce can be “just” a counter!

Nonce-free AE(AD)

- There are two flavours
  - Single-pass
  - Double-pass
- Double-pass, SIV [RS06] as a good example
  - Processes data twice
  - Might be inefficient/prohibitive in some applications
  - Same plaintext means same ciphertext and tag
- Single-pass, MCoE-G [FFLW12] as a good example
  - Process the data one time both for auth and enc
  - Inherent limitation: common prefix in P translates to common prefix in C

SIV: Nonce-free two-pass



H = header, or AD

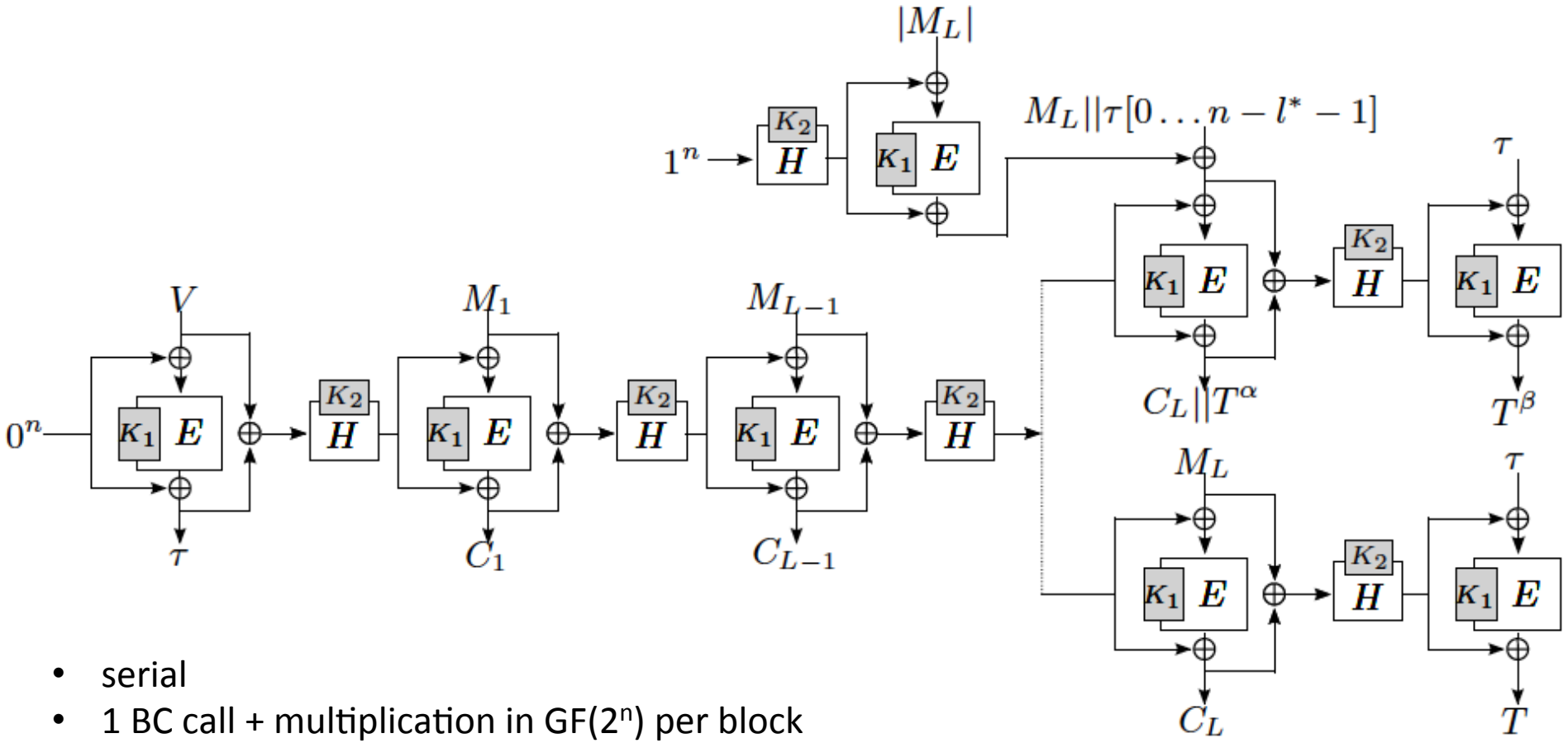
F = MAC

X = P

E = enc, e.g. AES-CTR

[FFLW12]

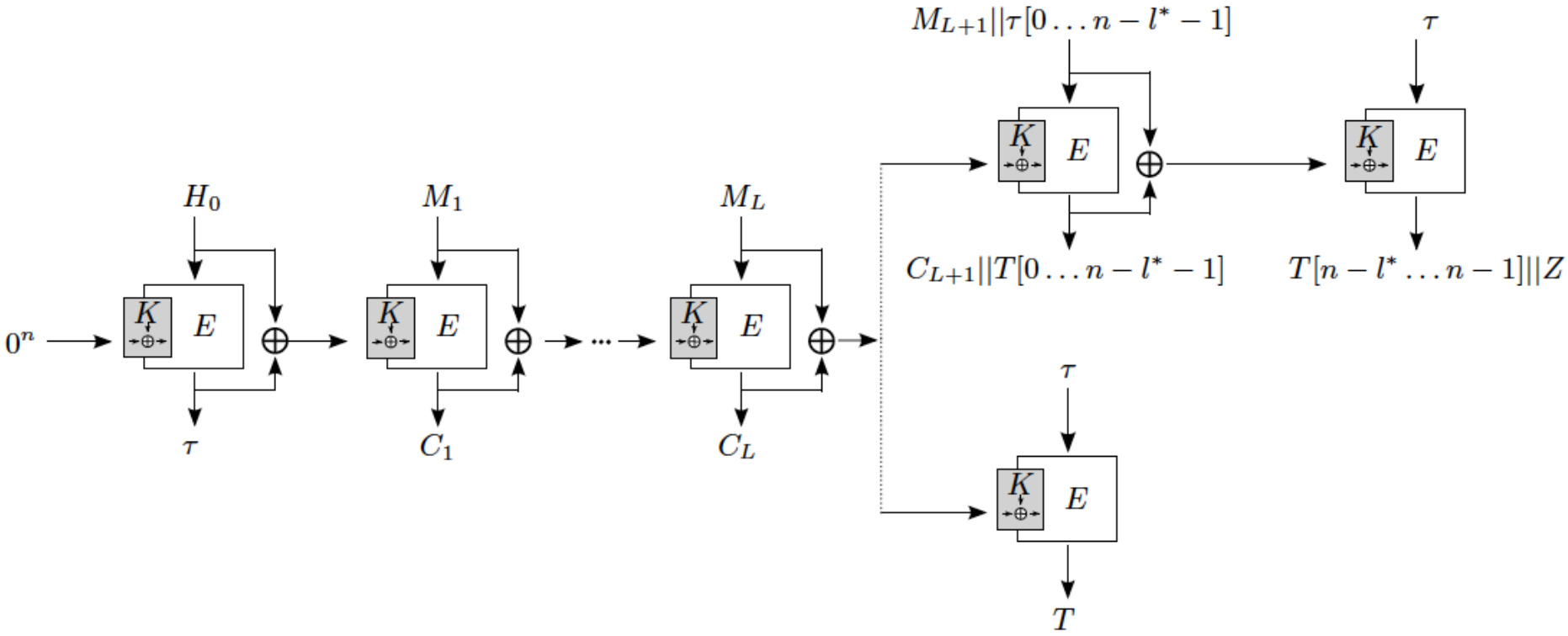
MCoE-G: Nonce-free one-pass



- serial
- 1 BC call + multiplication in GF(2^n) per block
- nonce-free
- one-pass
- common-prefix preservation

[MMRT12]

MCoE-X: how things can go wrong



- serial
- 1 BC call + key schedule per block
- nonce-free
- one-pass
- common-prefix preservation

[MMRT12]

MCoE-X: Key recovery in birthday complexity

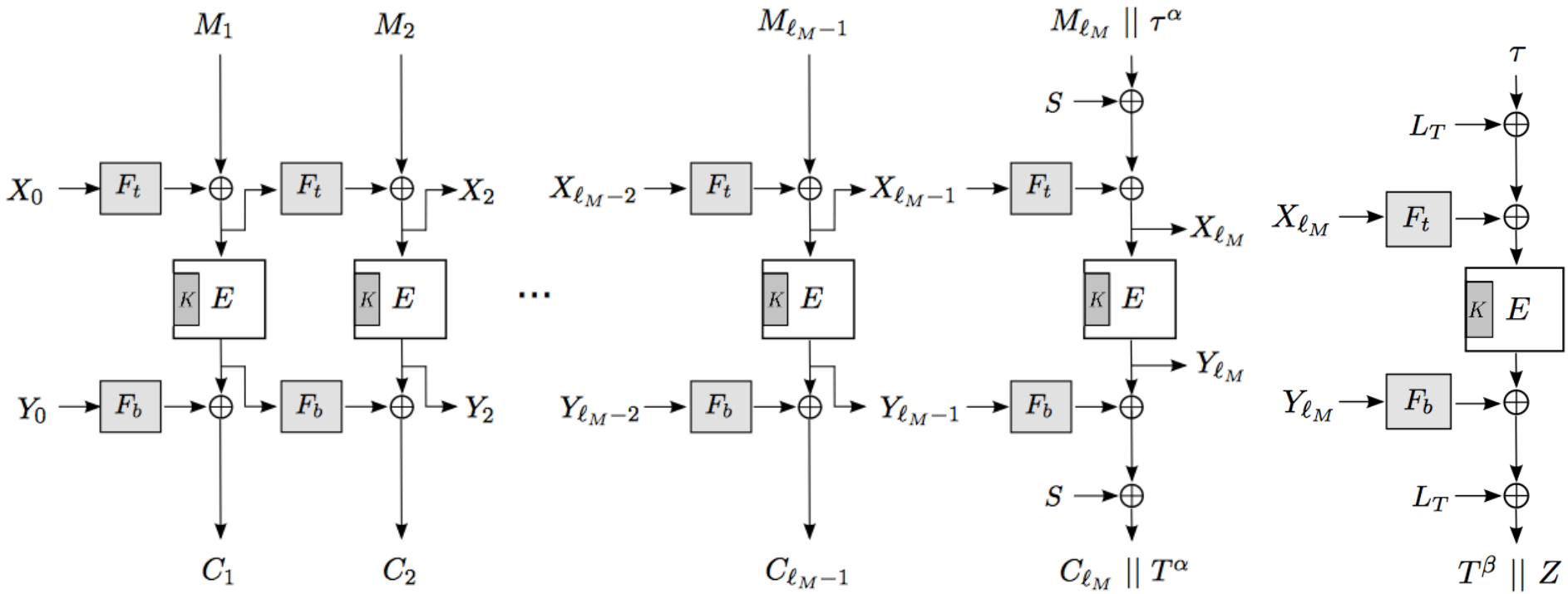
A simple attack (key collision):

1. Choose an arbitrary value a.
2. For ℓ values k compute $b = E(k, a)$ and save the pair (b, k) in a list L.
3. Choose an arbitrary x and set $M_1 = x$ and $M_2 = a$ such that $m = x \| a$ and ask for the ciphertext/tag pair (c, T) with $c = C_1 \| C_2$.
4. Check if C_2 is in the list L to get K.
   - If C_2 is in the list L then a candidate for the key is found. Compute $K = k \oplus M_1 \oplus C_1$,
   - Else go back to step 3.

After repeating steps 3-4 about $2^n/\ell$ times one expects to find the correct key with complexity of about $2^n/\ell + \ell$.

[AFFFLLMW14]

POET: Nonce-free one-pass



- 1 BC call + 2 universal hash function" per block (3 BC calls per block)
- nonce-misuse resistant
- one-pass
- common-prefix preservation
- "pipelinable" but rather serial

Nonce-misuse resistant: Going parallel

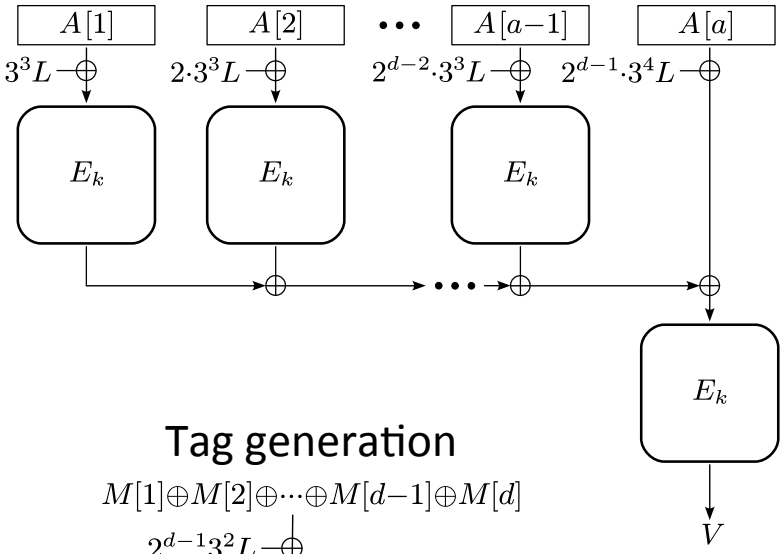
- McOE-G and POET are single-pass but (practically) serial
- SIV is two-pass

Is it possible to build a nonce-free AEAD that is one-pass and parallel?

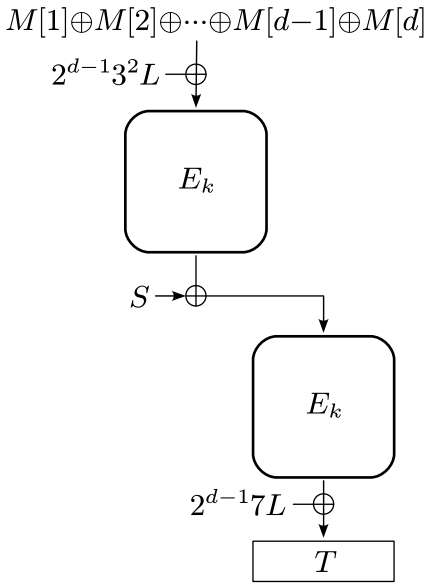
COPA:
Parallelizable single-pass nonce-free AE

[ABLMTY'13]

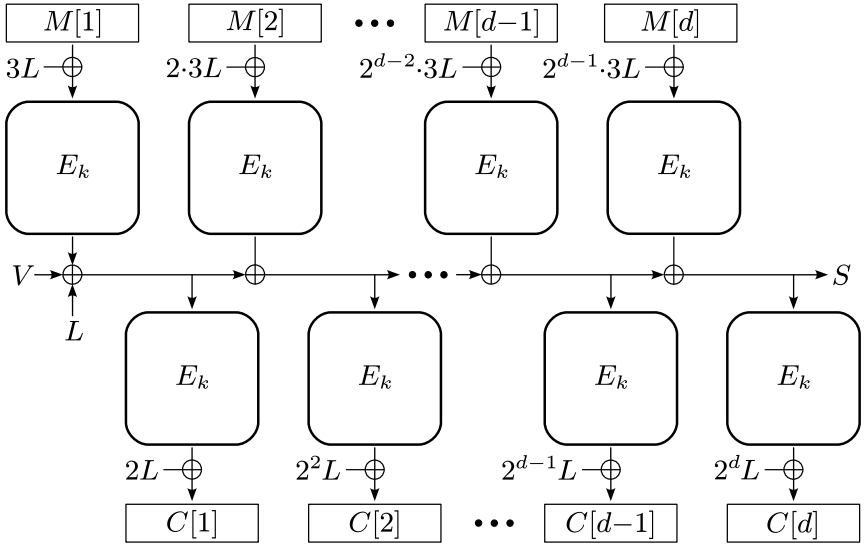
Associated data



Tag generation



Message (COPE)



- well parallelizable
- two BC calls per block
- nonce-free
- one-pass
- common-prefix preservation

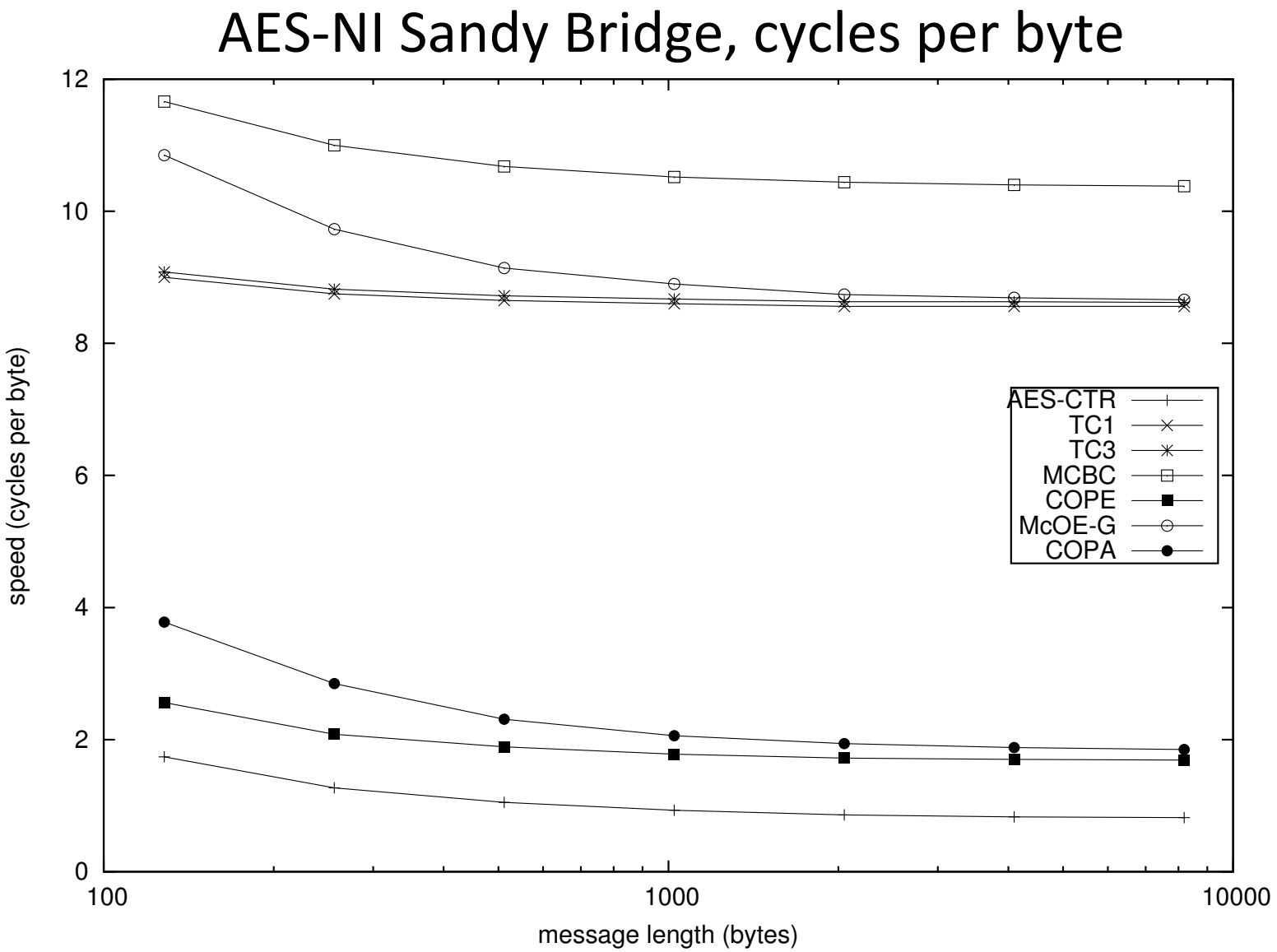
Collaboration with Elena Andreeva, Atul Luykx, Bart Mennink, Elmar Tischhauser, Kan Yasuda

Outline

- Block ciphers
- Basic modes of operation
- AE and AEAD
- Nonce-based AE modes and features
- Nonce-based AE: Implementation properties
- Nonce-free AE modes and features
- **Nonce-free AE: Implementation properties**
- Permutation-based AE

[ABLMTY'13]

Nonce-free single-pass AE in software



Outline

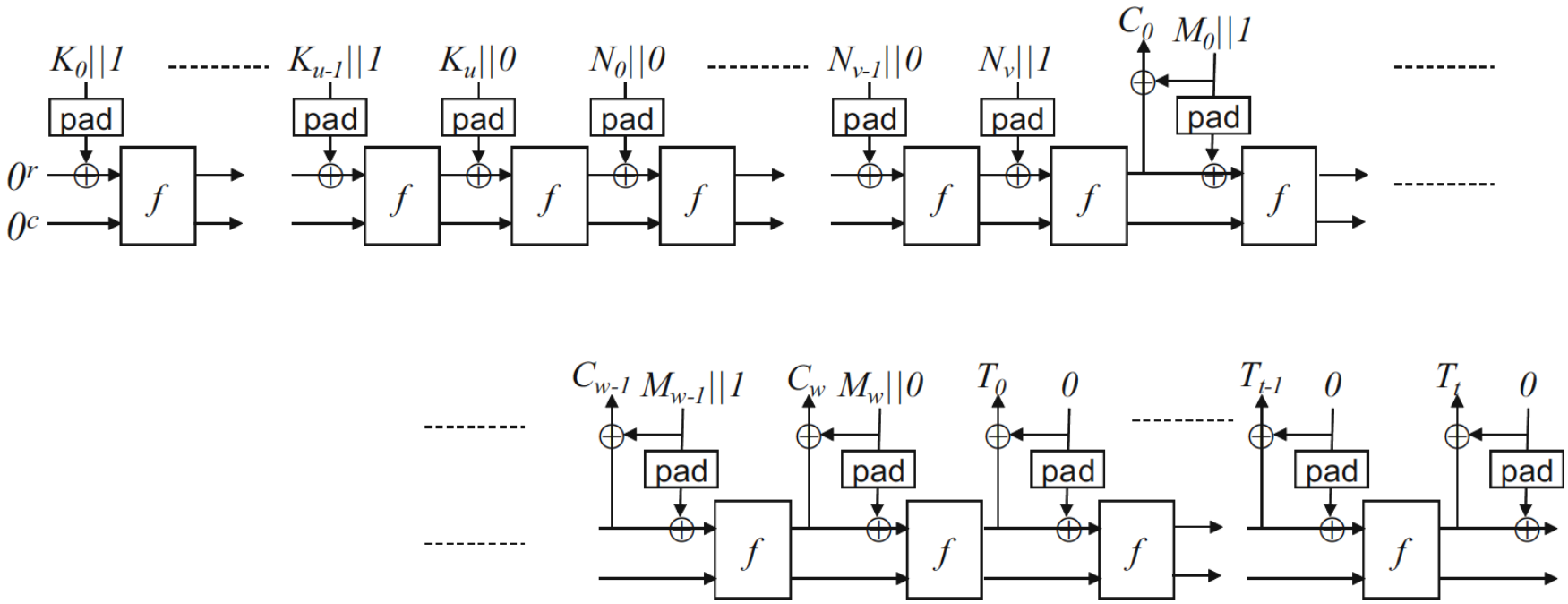
- Block ciphers
- Basic modes of operation
- AE and AEAD
- Nonce-based AE modes and features
- Nonce-based AE: Implementation properties
- Nonce-free AE modes and features
- Nonce-free AE: Implementation properties
- Permutation-based AE

Outline

- Block ciphers
- Basic modes of operation
- AE and AEAD
- Nonce-based AE modes and features
- Nonce-based AE: Implementation properties
- Nonce-free AE modes and features
- Nonce-free AE: Implementation properties
- **Permutation-based AE**

[BDPV'10]

SpongeWrap: Nonce- and permutation-based AE

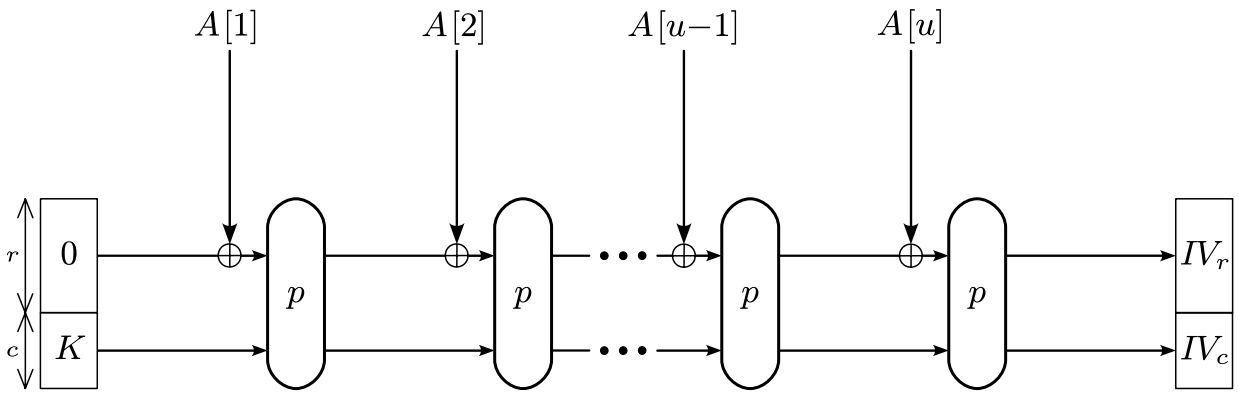


- Permutation-based
- Requires nonce
- Essentially serial but there are parallelization tricks (see e.g. Keyak and NORX)
- Strong potential for lightweight

APE:
Nonce-free permutation-based AE

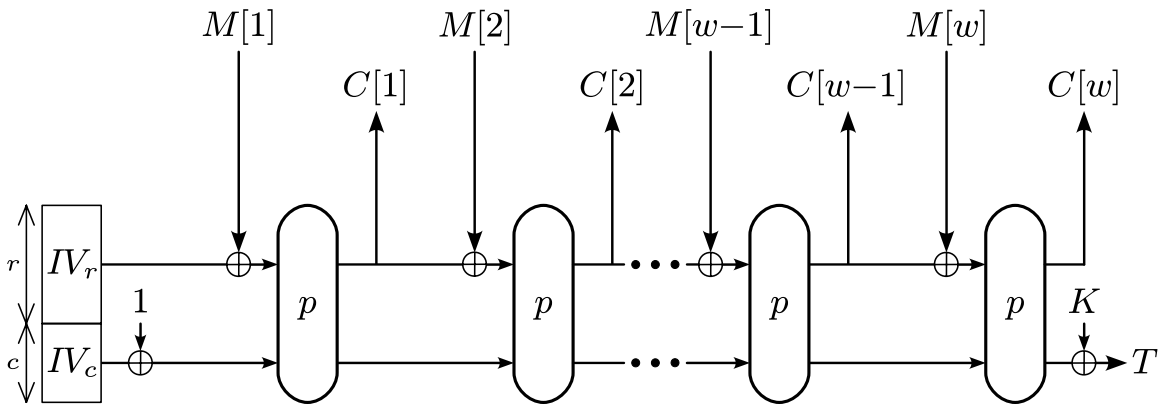
[ABLMMY'14]

Associated data:



- Permutation-based
- Nonce misuse resistant
- Serial
- Lightweight
- Dec backwards
- A mode for a sponge construction such as Photon or Spongent

Message processing and tag generation:



Collaboration with Elena Andreeva, Atul Luykx, Bart Mennink, Nicky Mouha, Kan Yasuda

Wrap-up

Block-cipher based AE:

- OCB if reliable nonce and SW (patent!)
- SIV if no nonce, perfect security is needed, and two-pass is possible
- COPA if no nonce and single-pass required

Permutation-based AE:

- SpongeWrap if nonce is reliable
- APE if no nonce and decrypting backwards is possible
- (Both rather lightweight in HW)