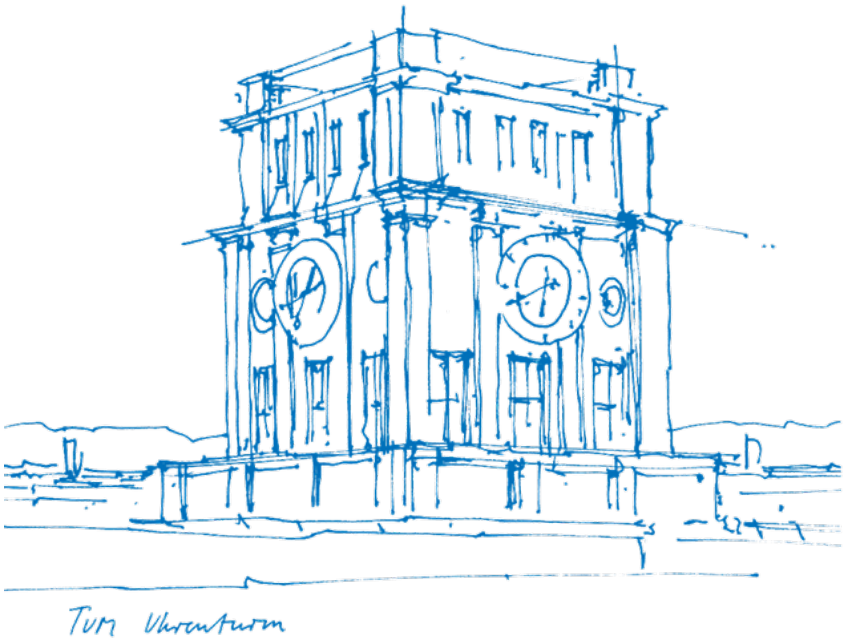
Persistent Fault Analysis of OCB, DEOXYS and COLM

Michael Gruber, Matthias Probst, Michael Tempelmeier

Department of Electrical and Computer Engineering,
Technical University of Munich

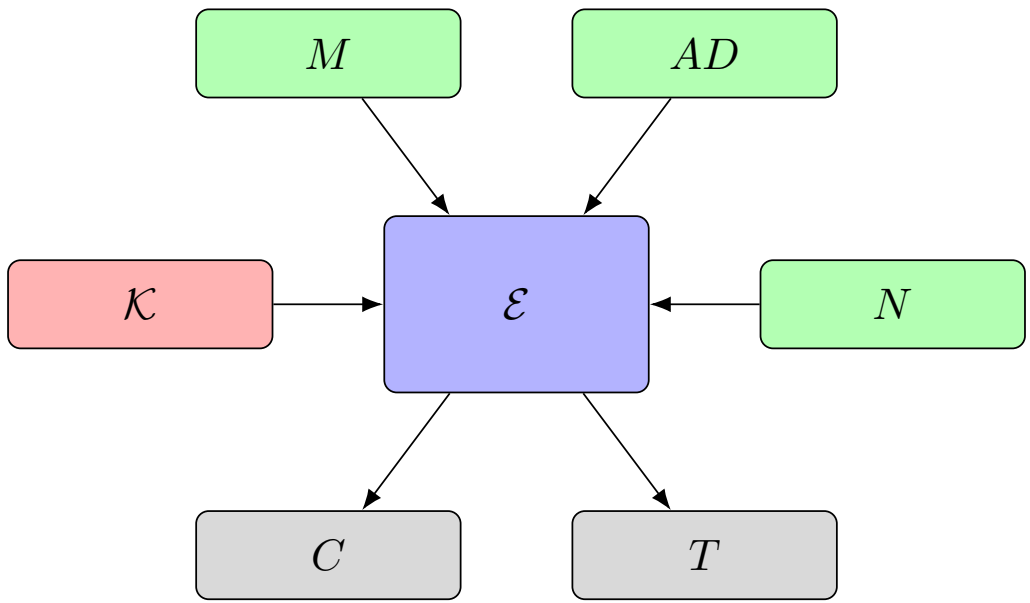
FDTC 2019



Overview

Authenticated Encryption

- Authenticity
- Confidentiality

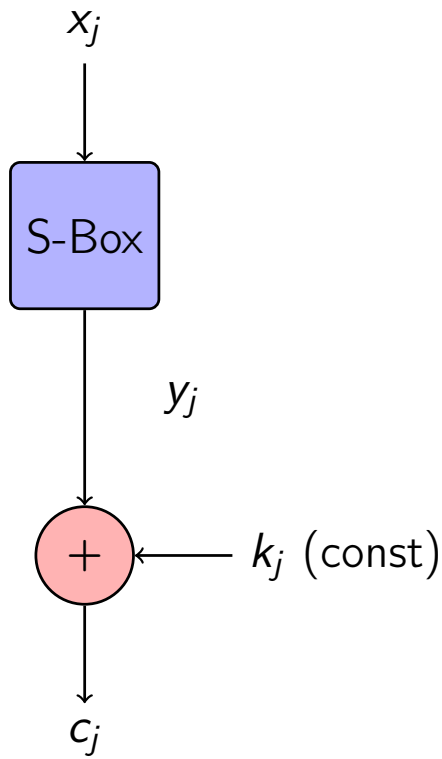


Overview

Persistence Fault Analysis I

- Introduced by Zhang et al. [6]
- Fault with persistent effect
- Modified constants (S-Box)

- No fault injection at runtime
- Faulty ciphertext only

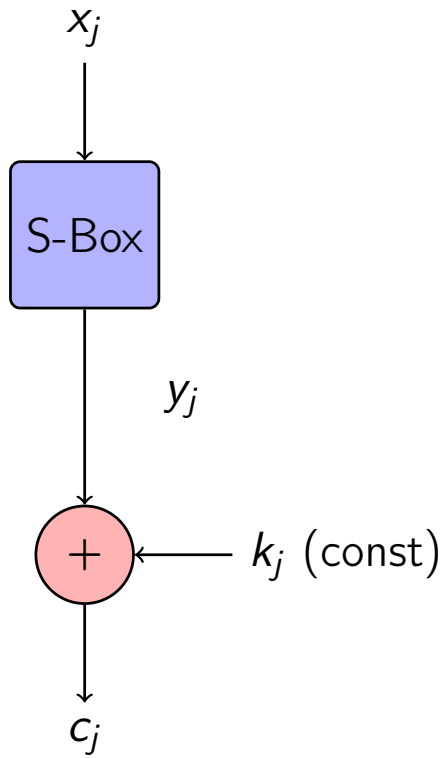


Persistence Fault Analysis II

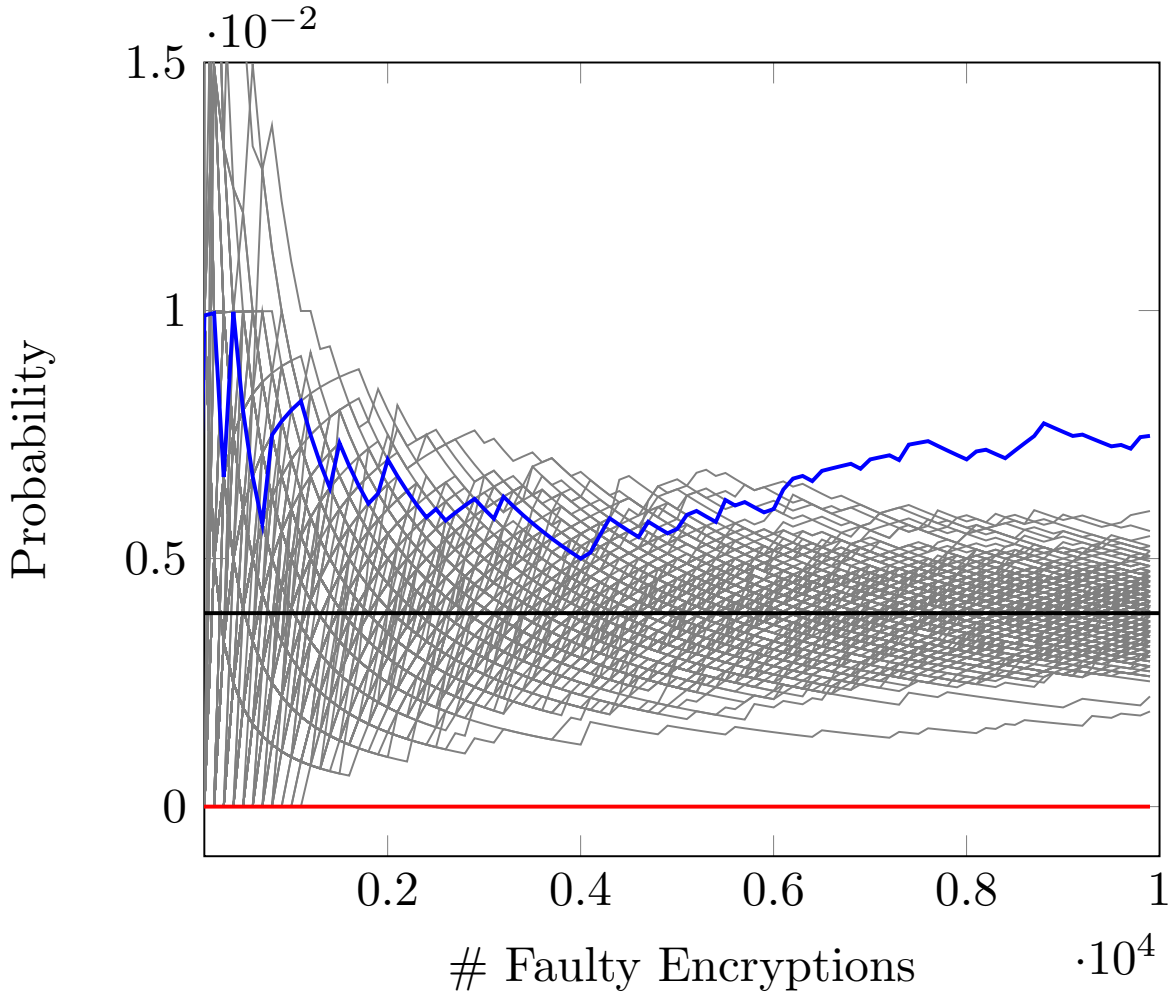
- $c_j = k_j + y_j$
- $y_j = S[x_j]$
- $P(y_j = i) = 2^{-b} \; \forall \; i \in [0, (2^b - 1)]$

- $S[0] = v$
- $S^*[0] = v^* \mid v \neq v^*.$
- $P(y_j \neq \{v, v^*\}) = 2^{-b} = \frac{1}{256}$

- $P(y_j = v) = 0$
- $P(y_j = v^*) = 2^{1-b} = \frac{2}{256}$



Persistence Fault Analysis III



Application of Persistence Fault Analysis

- Prerequisites
  - Faulty S-Box (persistent fault)
  - Sufficient #ciphertexts
  - Access to output of substitution layer

Application of Persistence Fault Analysis

- Prerequisites
  - Faulty S-Box (persistent fault)
  - Sufficient #ciphertexts
  - Access to output of substitution layer

- Some AEAD schemes limit access to the substitution layer output

Application of Persistence Fault Analysis

- Prerequisites
  - Faulty S-Box (persistent fault)
  - Sufficient #ciphertexts
  - Access to output of substitution layer

- Some AEAD schemes limit access to the substitution layer output

Application of PFA on AEAD schemes?

Application of Persistence Fault Analysis

- Prerequisites
  - Faulty S-Box (persistent fault)
  - Sufficient #ciphertexts
  - Access to output of substitution layer

- Some AEAD schemes limit access to the substitution layer output

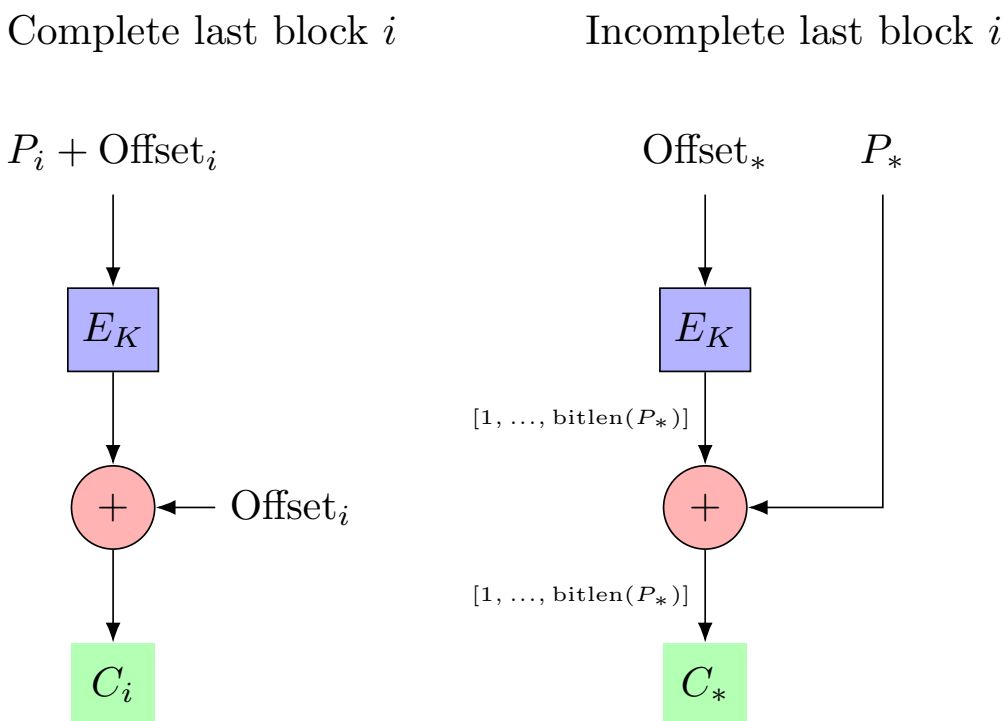
Application of PFA on AEAD schemes?

$\Rightarrow$ **Reduce AEAD schemes to "block cipher invocations"**

Overview

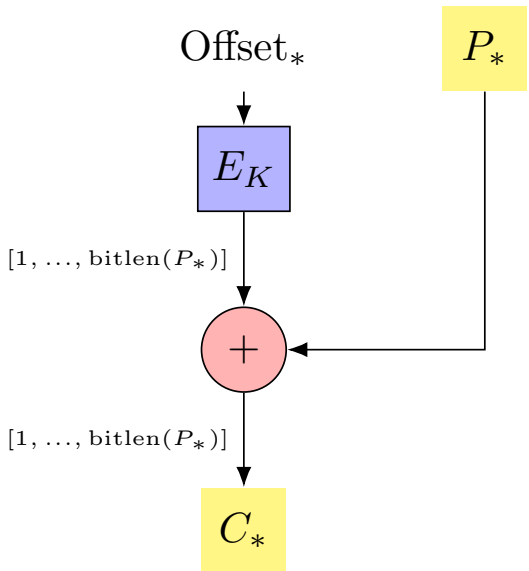
OCB

- Introduced by Krovetz and Rogaway [5]
- Mode of operation of AES



OCB - Attack - Message Encryption

- Similar, Dobraunig et al. [2]
- Masks on full message blocks
- Last incomplete message block does not involve Masks

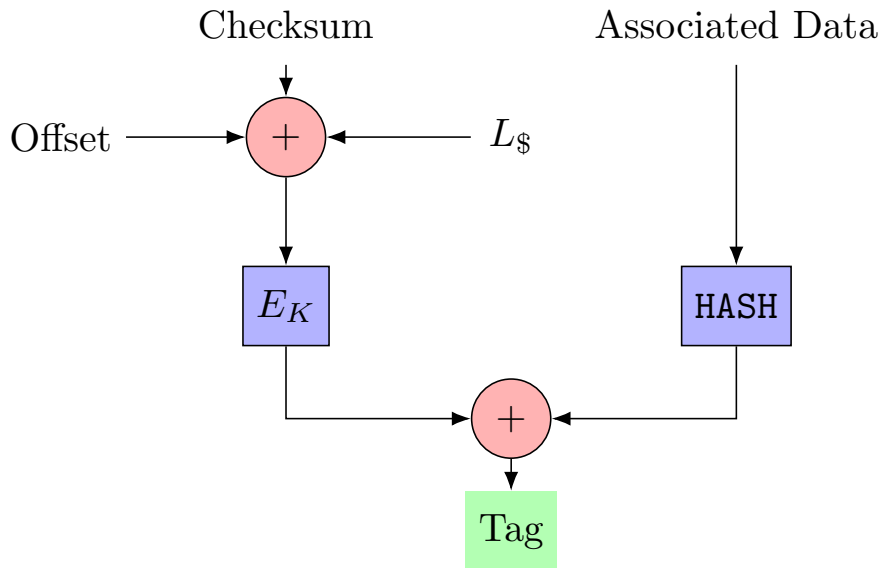


- Requirements:
  - Nonce
  - Plaintext
  - Ciphertext

OCB - Attack - Tag Generation

- If AD empty HASH returns 0^{128}
- Apply standard PFA

- Requirements:
  - Nonce
  - Tag
  - AD empty



Overview

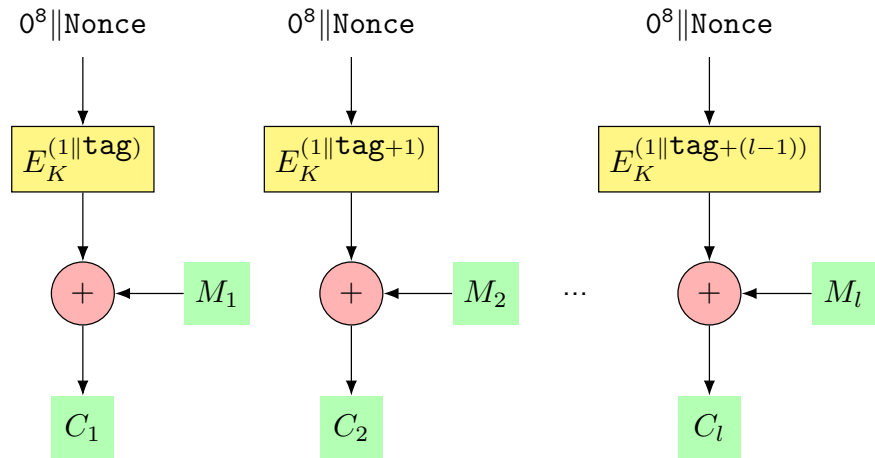
DEOXYS II

- Introduced by Jean et al. [4]
- Mode of Operation of Deoxys-BC
  - Round function similiar to AES
  - Tweakey Framework
- Applicable to Deoxys-II-128-128
- Problems
  - MB not omitted during last round
  - Tweakey Schedule

DEOXYS-II - Attack - Encryption

- Encryption of $0^8||$Nonce
- Key constant
- Tweak depends on tag and block number *l*

- Requirements:
  - Tweak
  - Tag
  - Plain/Cipher-text



DEOXYS Modified Round Function

- MB spreads fault over 4 bytes
- Reformulate last round i.e. exchange MB, ATK
- Both functions are linear
- Same behaviour as the original round function

Algorithm 1 DEOXYS-Round

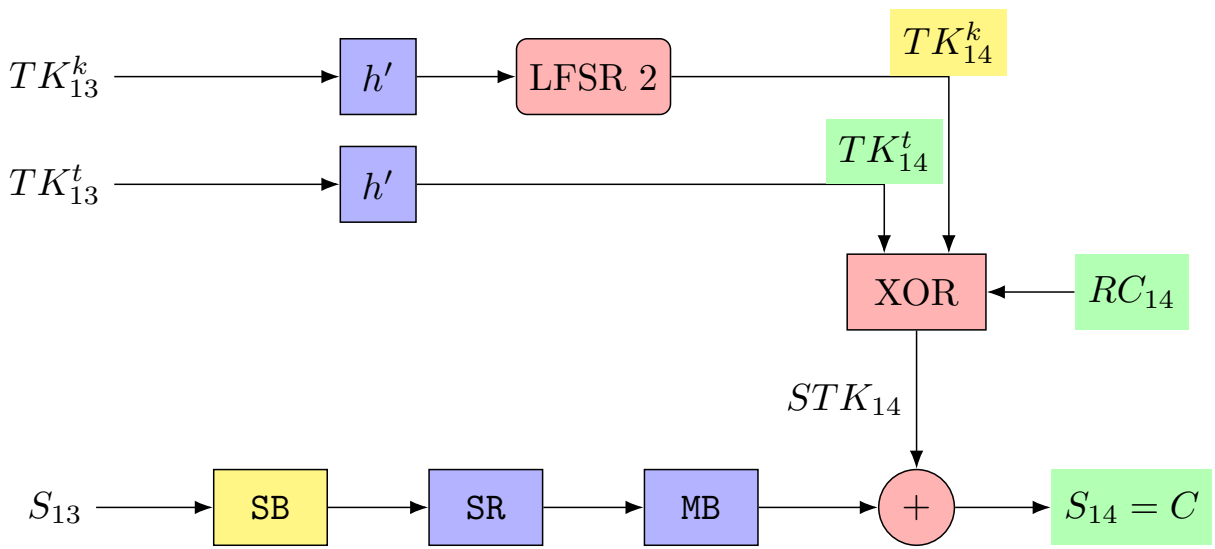
1: $ATK(State, STK_{13})$
2: $STK_{14} \leftarrow TKS(STK_{13})$
3: $SR(State)$
4: $SB(State)$
5: $MB(State)$
6: $Cipher \leftarrow ATK(State, STK_{14})$

Algorithm 2 DEOXYS-Round-MOD

1: $ATK(State, STK_{13})$
2: $STK_{14} \leftarrow TKS(STK_{13})$
3: $SR(State)$
4: $SB(State)$
5: $ATK(State, inverseMB(STK_{14})$
6: $Cipher \leftarrow MB(State)$

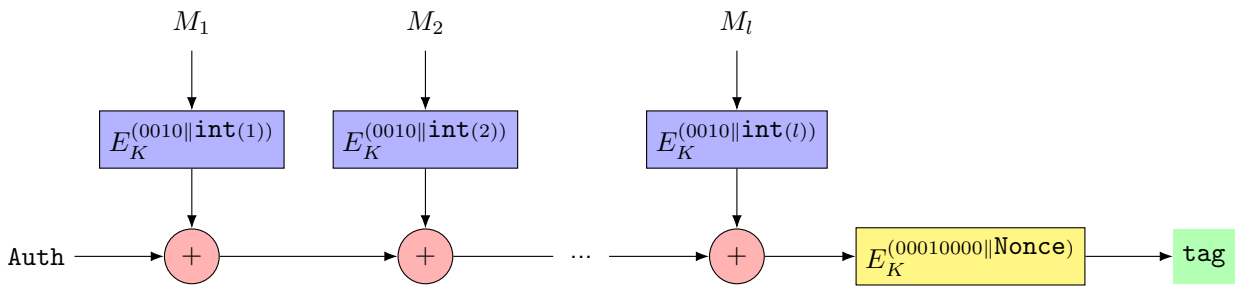
DEOXYS TweaKey-Schedule

- Introduced by Jean et al. [3]
- Round key derived from key and tweak
- h', $LFSR\ 2$ invertible
- $TK_{14}^k = TK_{14}^t + RC_{14} + STK_{14}$



DEOXYS-II - Attack - Tag

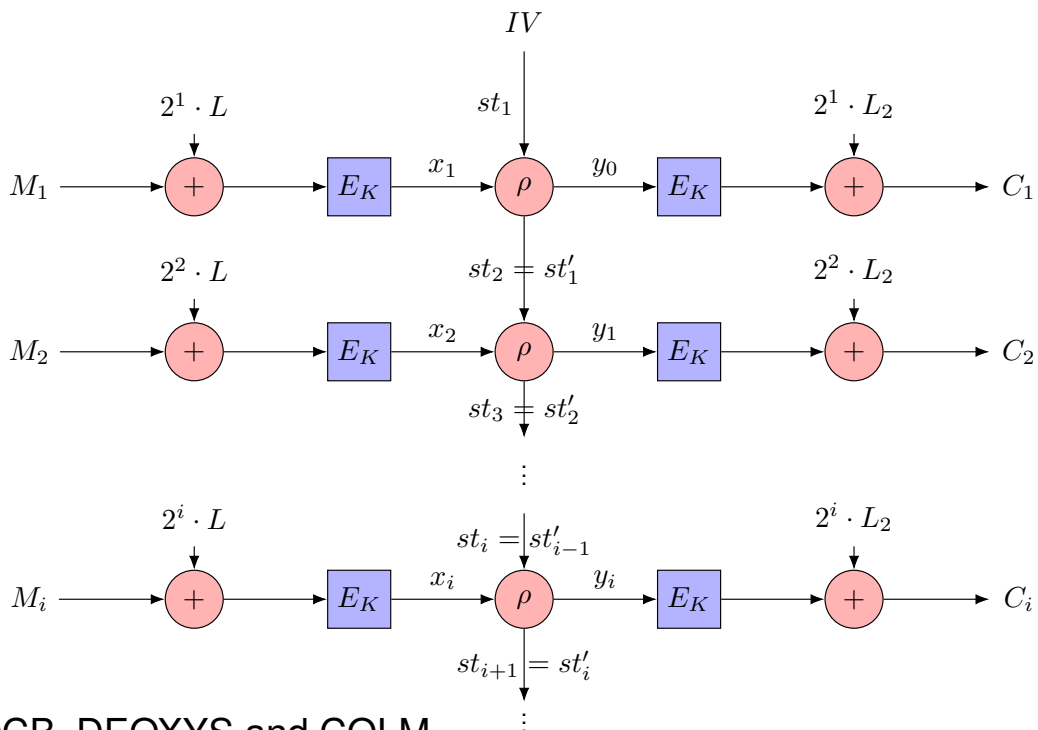
- Requirements:
  - Nonce
  - Tweak
  - Tag



Overview

COLM

- Introduced by Andreeva et al. [1]
- encrypt-linear-mix-encrypt mode of AES
- COLM_0, defense in depth
- Addition of masks $L = E_K(0)$, $L_2 = 3^2 \cdot L$



COLM - Attack - Encryption

Recover sum (R_i) of mask and key $R_i = k + 2^i \cdot L_2$

$$R_x + R_y = (k + 2^x \cdot L_2) + (k + 2^y \cdot L_2) = (2^x + 2^y) \cdot L_2 \tag{1}$$

$$L_2 = (2^x + 2^y)^{-1} \cdot (R_x + R_y) \tag{2}$$

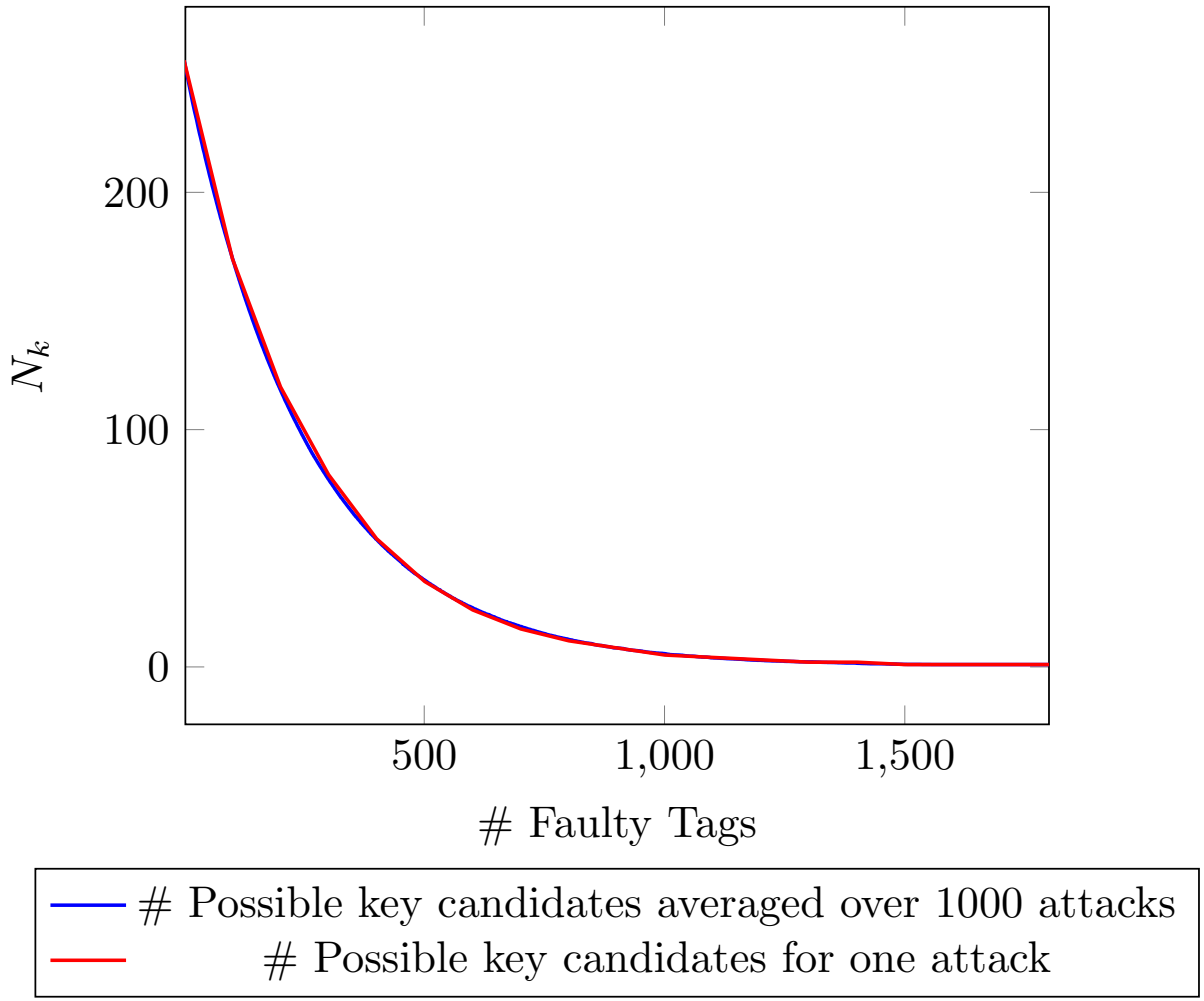
As L_2 is known, the last round key k can be calculated as shown in eq. (3).

$$k = R_i + 2^i \cdot L_2 \tag{3}$$

Invert key schedule of AES to calculate the master key K.

Overview

Results - Simulation- DEOXYS - TAG



Results - Summary

Cipher-Family	Version	Applicable
Deoxys-II	Deoxys-II-128-128	✓
	Deoxys-II-256-128	✗
OCB	with AES-128, all tag-sizes*	✓
	with AES-192, all tag-sizes	✗
	with AES-256, all tag-sizes	✗
COLM	COLM_0	✓
	COLM_{127}	✓

* Attack on Tag-generation: Brute force effort $2^{128-\text{Taglen}}$

Results - Requirements

Cipher	Attacked Function	Requirements
Deoxys-II	Tag-Generation Message-Encryption	Faulty tags, nonce Tags, faulty cipher and plain-texts
OCB	Tag with AD empty Incomplete Message-Block	Faulty tags Incomplete cipher- and corresponding plain-texts
COLM	Message-Encryption	Faulty cipher-texts

Conclusion

- PFA is applicable to AEAD schemes
- PFA of OCB, DEOXYS and COLM is feasible
- Masks can hinder the application of PFA

Thank you for your attention!

m.gruber@tum.de
https://www.sec.ei.tum.de

References

[1] E. Andreeva et al. “COLM v1”. In: *Submission to the CAESAR Competition* (2016).

[2] C. Dobraunig et al. “Practical Fault Attacks on Authenticated Encryption Modes for AES.”. In: *IACR Cryptology ePrint Archive* 2016 (2016), p. 616.

[3] J. Jean, I. Nikolić, and T. Peyrin. “Tweaks and Keys for Block Ciphers: The TWEAKEY Framework”. In: *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2014, pp. 274–288.

[4] J. Jean et al. “Deoxys v1.41”. In: *Submitted to the CAESAR Competition* (2016).

[5] T. Krovetz and P. Rogaway. “OCB v1.1”. In: *Submission to the CAESAR Competition* (2016).

[6] F. Zhang et al. “Persistent Fault Analysis on Block Ciphers”. en. In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* Volume 2018 (2018), Issue 3–.