SUNDAE: Small Universal Deterministic Authenticated Encryption for the IoT

Subhadeep Banik[1,4], Andrey Bogdanov[2], Atul Luykx[3], Elmar Tischhauser[2]

[1]LASEC, EPFL Switzerland

[2]Technical University of Denmark, Lyngby

[3]Visa Research, USA

[4]NTU, Singapore

Fast Software Encryption 2019, Paris

25th March 2018

Outline

- Introduction
- Specification
- Security
- Implementation

Block Cipher based AE

- Block cipher is an efficient component for lightweight AE.
- SIV (Eurocrypt 2006) mode requires 2 independent keys.
- Some candidates:
  → COPA/EℓMD/COLM: Internal state size atleast 3 times of block length.
  → EAX: Multiple inital block cipher calls.
  → COFB/JAMBU: State size greater than block length.
- GCM-SIV proposed at CCS 2015 .
  → Multiplication in $GF(2^{128})$: not efficient in hardware.

Contributions

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

SUNDAE

- Competes with CLOC/JAMBU in number of block cipher calls for short messages
- Improves COFB and other modes in terms of state size
- Simultaneously offers efficiency on lightweight and high-performance platforms
- Provides maximal robustness to a lack of proper randomness

Characteristics

SUNDAE

- Completely deterministic:
  $\rightarrow$ If input is unique, it maintains both data confidentiality and authenticity.
- Processes inputs of the form (A, M)
  $\rightarrow$ If M is empty, the mode reduces to a MAC.
  $\rightarrow$ If nonce is required, the first x bits of A can serve the purpose.
- Structure is based on SIV, optimized for lightweight settings:
  $\rightarrow$ Uses one key, consists of a cascade of block cipher calls.
  $\rightarrow$ Only additional operations: XOR and multiplication by fixed constants.
- State size of n, where n is blocklength of underlying block cipher.
  $\rightarrow$ CLOC requires $2n$-bits, JAMBU $1.5n$-bits, and COFB $1.5n$-bits.

Characteristics

SUNDAE

- Rate $1/2$ mode:
  $\rightarrow$ 2 block cipher calls per message block.

- Efficient for short messages: for 1 block of nonce, plaintext, AD
  $\rightarrow$ COFB uses 3 block cipher calls, CLOC requires 4, JAMBU 5.
  $\rightarrow$ SUNDAE requires 5 calls (can be reduced to 4, if one call is precomputed).

- Hence efficient in settings where communication outweighs computational costs
  $\rightarrow$ If AD/plaintext is never repeated,
  $\rightarrow$ nonce is no longer needed, and
  $\rightarrow$ communication or synchronization costs are reduced,
  $\rightarrow$ in addition to reducing the block cipher calls to 4

Specification

Algorithm 1: $\mathsf{enc}_K(A, M)$

Input: $K \in \mathsf{K}$, $A \in \{0,1\}^*$, $M \in \{0,1\}^*$
Output: $C \in \{0,1\}^{n+|M|}$

$b_1 \leftarrow |A| > 0\ ?\ 1 : 0$
$b_2 \leftarrow |M| > 0\ ?\ 1 : 0$
$V \leftarrow \mathsf{E}_K\left(b_1 \| b_2 \| 0^{n-2}\right)$
$T \leftarrow V$ `// Initial tag`
if $|A| > 0$ **then**
    $A[1]A[2]\cdots A[\ell_A] \xleftarrow{n} A$
    for $i = 1$ **to** $\ell_A - 1$ **do**
        $V \leftarrow \mathsf{E}_K\left(V \oplus A[i]\right)$
    end
    $X \leftarrow |A[\ell_A]| < n\ ?\ 2 : 4$
    $V \leftarrow \mathsf{E}_K\left(X \times \left(V \oplus \mathsf{pad}(A[\ell_A])\right)\right)$
    $T \leftarrow V$
end
if $|M| > 0$ **then**
    $M[1]M[2]\cdots M[\ell_M] \xleftarrow{n} M$
    for $i = 1$ **to** $\ell_M - 1$ **do**
        $V \leftarrow \mathsf{E}_K\left(V \oplus M[i]\right)$
    end
    $X \leftarrow |M[\ell_M]| < n\ ?\ 2 : 4$
    $V \leftarrow \mathsf{E}_K\left(X \times \left(V \oplus \mathsf{pad}(M[\ell_M])\right)\right)$
    $T \leftarrow V$
    for $i = 1$ **to** ℓ_M **do**
        $V \leftarrow \mathsf{E}_K\left(V\right)$
        $C[i] \leftarrow \lfloor V \rfloor_{|M[i]|} \oplus M[i]$
    end
    return $TC[1]\cdots C[\ell_M]$
end

return T

Specification

Algorithm 2: $\mathsf{dec}_K(A, C)$

Input: $K \in \mathsf{K}$, $A \in \{0,1\}^*$, $C \in \{0,1\}^n \times \{0,1\}^*$

Output: $\perp$ or $M \in \{0,1\}^{|C|-n}$

$C[1]C[2]\cdots C[\ell] \xleftarrow{n} C$

$V \leftarrow C[1]$

for $i = 2$ **to** ℓ **do**

  $V \leftarrow E_K\big(V\big)$

  $M[i-1] \leftarrow \lfloor V \rfloor_{|M[i]|} \oplus C[i]$

end

$M \leftarrow \ell > 1 \ ?\ M[1]M[2]\cdots M[\ell-1] : \varepsilon$

$T \leftarrow \lfloor \mathsf{enc}_K(A, M) \rfloor_n$

if $T \neq C[1]$ **then**

  return $\perp$

return M

Specification

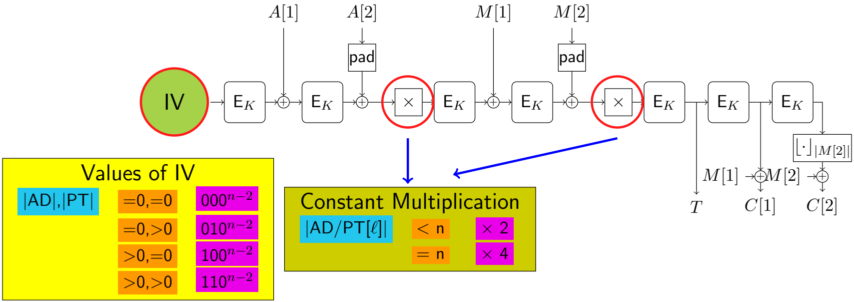


Figure: SUNDAE encryption with associated and plaintext data. The box below the rightmost block cipher call represents truncation.

Specification

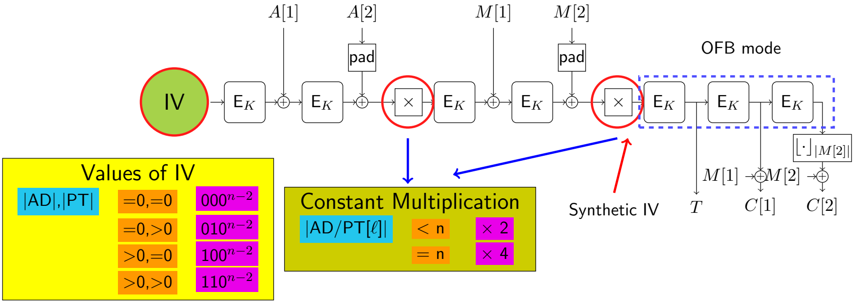


Figure: SUNDAE encryption with associated and plaintext data. The box below the rightmost block cipher call represents truncation.

Security Statement

Theorem

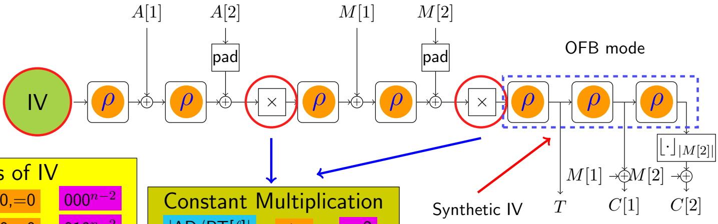
Let $\mathbf{A}$ *be an adversary making at most* q enc_K *and* q_v dec_K *queries with block length costs of at most* σ_A, σ_P, *and* σ_C *for associated, plaintext, and ciphertext data, respectively, then*

$$\mathsf{DAE}(\mathbf{A}) \leq \frac{N_{\mathsf{E}}^2}{2^{n+1}} + \frac{q_v}{2^n} + \frac{q^2}{2^n} + \frac{qq_v}{2^n} + \frac{(\sigma_P + \sigma_C)^2}{2^{n+1}} + \frac{4(\sigma_P + \sigma_C)}{2^n} + \frac{(4 + \sigma_A + \sigma_P + \sigma_C)^2}{2^n} + \frac{4(q + q_v)^2}{2^n} + \mathsf{PRP}_{\mathsf{E}}(\mathbf{A}_{\mathsf{E}}) \,. \quad (1)$$

where

$$N_{\mathsf{E}} := 4 + \sigma_A + 2\sigma_P + 2\sigma_C \quad (2)$$

Proof Intuition: Step 1 (Switching to URF)



Values of IV

|AD|,|PT|		
	=0,=0	000^{n-2}
	=0,>0	010^{n-2}
	>0,=0	100^{n-2}
	>0,>0	110^{n-2}

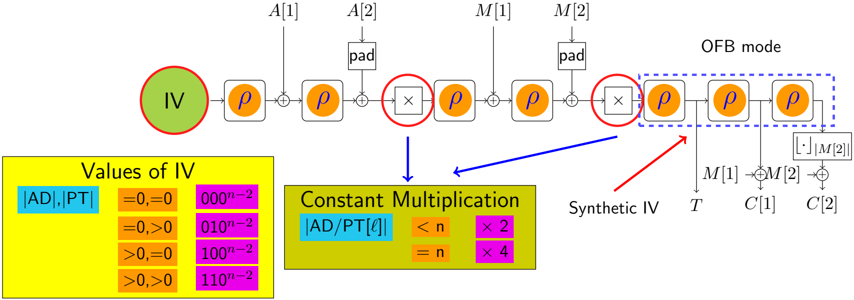
Constant Multiplication

$\|AD/PT[\ell]\|$		
	$<$ n	× 2
	$=$ n	× 4

$$\mathsf{DAE}(\mathbf{A}) := \underset{\mathbf{A}}{\Delta}(\mathsf{enc}_K, \mathsf{dec}_K\,;\,\$, \bot)$$

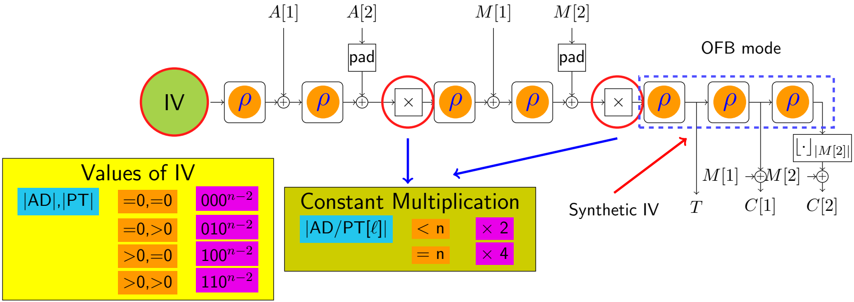
$$:= \underset{\mathbf{A}}{\Delta}(\mathsf{enc}[\rho], \mathsf{dec}[\rho]\,;\,\$, \bot) + \frac{N_{\mathsf{E}}^2}{2^{n+1}} + \mathsf{PRP}_{\mathsf{E}}(\mathbf{A}_{\mathsf{E}}),$$

Proof Intuition: Step 1 (Switching to URF)



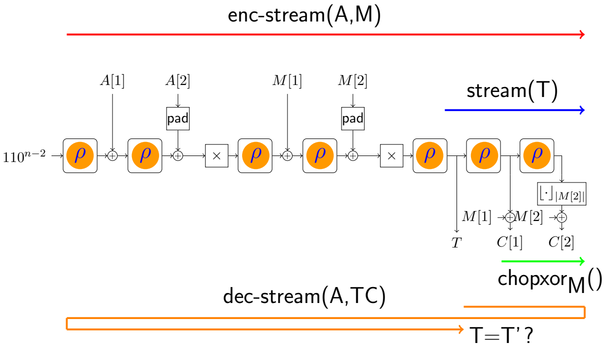
- We use stream cipher OFB, unpredictable SIV → confidentiality.
- Confidentiality will be maintained if the tag is unpredictable.
- AD/PT is processed similarly, we argue that the domain separation works.

Proof Intuition: Step 1 (Authenticity)



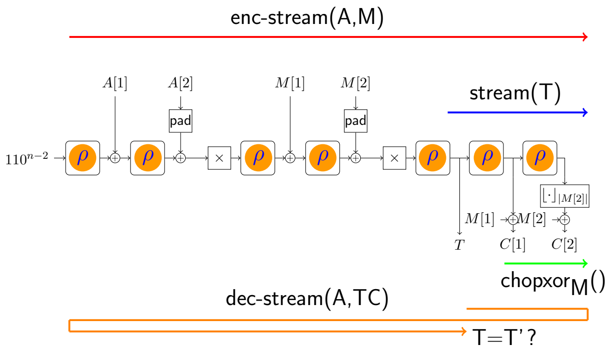
- Adversary forges $(C, T) \rightarrow$ output of MAC for $dec(C, T)$- call equals T
- By defn, C was never before output of previous enc query.
- Equivalent to producing pre-image/2nd pre-image of underlying MAC.

Proof Intuition: Step 2 (eliminate chopxor)



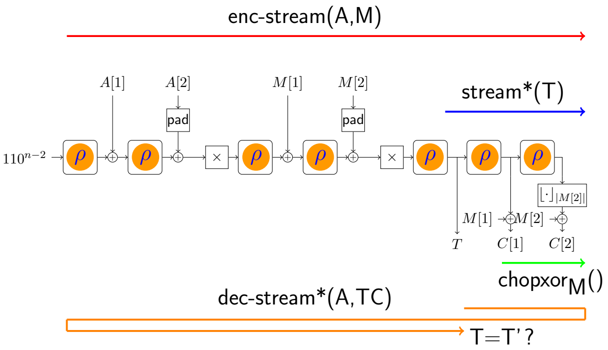
- TC = enc(A,M)=chopxor$_M$ ∘ enc-stream(A,M)
- M'=chopxor$_C$ ∘ stream(T). Compute T'= 1st block of enc-stream(A,M')
- If T=T', dec-stream(A,TC)= stream(T) else ⊥.
- M = dec(A,TC) = chopxor$_C$ ∘ dec-stream(A,TC)

Proof Intuition: Step 2 (eliminate chopxor)



- $\mathsf{DAE}(\mathbf{A}) := \Delta_{\mathbf{A}}\left(\mathsf{enc}[\rho], \mathsf{dec}[\rho]\,;\, \$, \perp\right) + \frac{N_{\mathsf{E}}^2}{2^{n+1}} + \mathsf{PRP}_{\mathsf{E}}(\mathbf{A}_{\mathsf{E}})$
- $\Delta_{\mathbf{A}}\left(\mathsf{enc}[\rho], \mathsf{dec}[\rho]\,;\, \$, \perp\right) \leq \Delta_{\mathbf{A}_{\mathsf{chopxor}}}\left(\mathsf{enc\text{-}stream}, \mathsf{dec\text{-}stream}\,;\, \$^s, \perp\right)$
- Where $\$^s$ returns random string of length $(\ell_M + 1) * n$

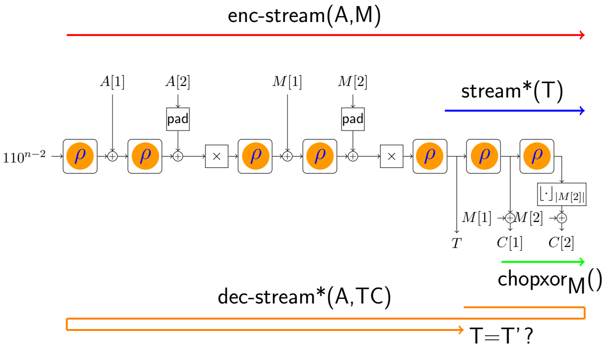
Proof Intuition: Step 3 (introduce stream*/decstream*)



- stream*(T) outputs completely random values of required length.
- If T=T$_i$ for some i, dec-stream*(A,TC) outputs stream*(T$_i$) else $\perp$

$$\Delta_{\mathbf{A}_{\text{chopxor}}}(\text{enc-stream}, \text{dec-stream}\,;\,\$^s, \perp) \leq \Delta_{\mathbf{A}_{\text{chopxor}}}(\text{enc-stream}, \text{dec-stream}\,;\,\$^s, \text{dec-stream}^*) + \Delta_{\mathbf{A}_{\text{chopxor}}}(\$^s, \text{dec-stream}^*\,;\,\$^s, \perp)$$

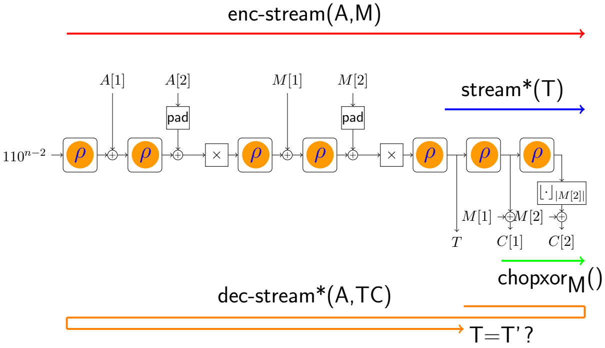
Proof Intuition: Step 3 (introduce stream*/decstream*)



- $\Delta_{\mathbf{A}_{\text{chopxor}}}(\$^s, \text{dec-stream}^*\,;\,\$^s, \perp) =$ prob that decstream* outputs non-$\perp$
- Same as finding pre-image/second pre-image for $\lfloor \$^s \rfloor_n$

$$\Delta_{\mathbf{A}_{\text{chopxor}}}(\$^s, \text{dec-stream}^*\,;\,\$^s, \perp) \leq \frac{q_v}{2^n} + \frac{q^2}{2^n} + \frac{qq_v}{2^n}\,. \qquad (3)$$

Proof Intuition: Step 3 (introduce stream*/decstream*)



- Remaining term $\Delta_{\mathbf{A}_{\text{chopxor}}}(\text{enc-stream}, \text{dec-stream}\,;\, \$^s, \text{dec-stream}^*)$
- We will try to bound using H-coefficient technique.

Proof Intuition: Step 4 (message to function)

- Split A and M into blocks, if non-empty, to get

$$A[1]\cdots A[\ell_A] \xleftarrow{n} A \text{ and } M[1]\cdots M[\ell_M] \xleftarrow{n} M\,. \tag{4}$$

- Each block augmented with a bit to indicate if it is a final block or not.

$$\Big((0, A[1]), \ldots, (1, A[\ell_A]), (0, M[1]), \ldots, (1, M[\ell_M])\Big)\,. \tag{5}$$

- The augmented blocks are used as parameter in the function

$$f : \Big(\{0,1\} \times \{0,1\}^{\leq n}\Big) \times \mathsf{B} \to \mathsf{B}\,, \tag{6}$$

where f is defined as

$$f((\delta, X), Y) := \begin{cases} X \oplus Y & \text{if } \delta = 0 \\ 2 \times (\mathsf{pad}(X) \oplus Y) & \text{if } \delta = 1 \text{ and } |X| < n \\ 4 \times (X \oplus Y) & \text{otherwise} \end{cases}\,. \tag{7}$$

Proof Intuition: Step 4 (message to function)

- If $A \neq \varepsilon$ and $M \neq \varepsilon$, we have that $(f((\delta, X), Y)$ and $f_{\delta,X}(Y)$ are equiv)

$$I(A, M) := \Big(110^{n-2}, f_{0,A[1]}, \cdots, f_{0,A[\ell-1]}, f_{1,A[\ell_A]}, f_{0,M[1]}, \cdots, f_{0,M[\ell-1]}, f_{1,M[\ell_M]}\Big), \quad (8)$$

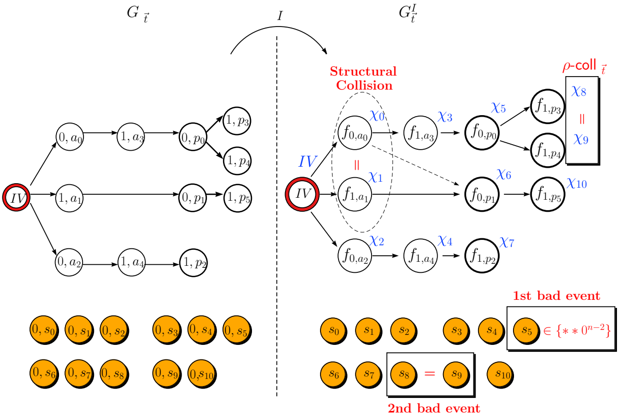
where values $X \in \{0,1\}^n$ are interpreted as constant functions mapping any element in B to X.

- Given $\vec{x} = (x_1, x_2, \ldots, x_\ell)$ where each x_i is a function, define

$$\widehat{\rho}(x_1, x_2, \ldots, x_\ell) = \rho \circ x_\ell \circ \rho \circ x_{\ell-1} \circ \cdots \circ \rho \circ x_3 \circ \rho \circ x_2 \circ \rho \circ x_1 \,. \quad (9)$$

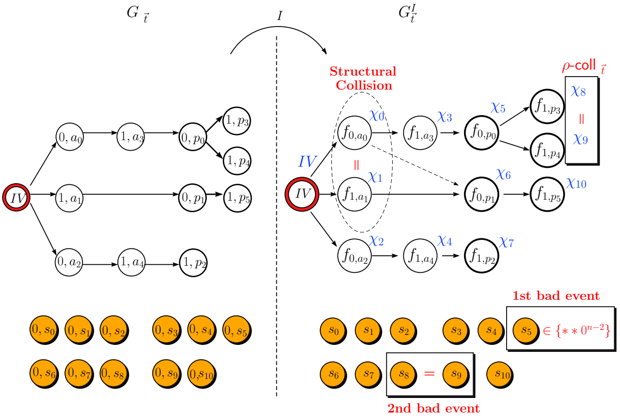
It is easy to see enc-stream$(A, M) :=$ stream$_{\ell_M}(\widehat{\rho}(I(A, M)))$

Proof Intuition: Step 5 (function to graph)



- Convert transcript to a graph, respecting prefix rules.
- Output streams exist as independent, unconnected nodes.
- Very natural to transform values to functions.
- Each edge becomes application of ρ, each node has label χ_i.

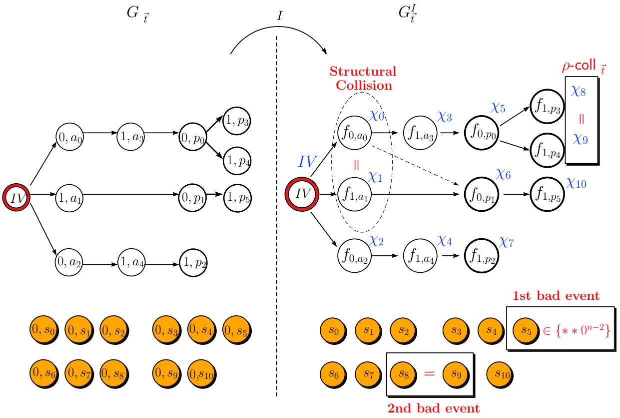
Proof Intuition: Step 5 (function to graph)



- Define T_{bad} for all transcripts that lead to events 1,2
- Allows trivial forgery.
- Concentrate on T_{good}

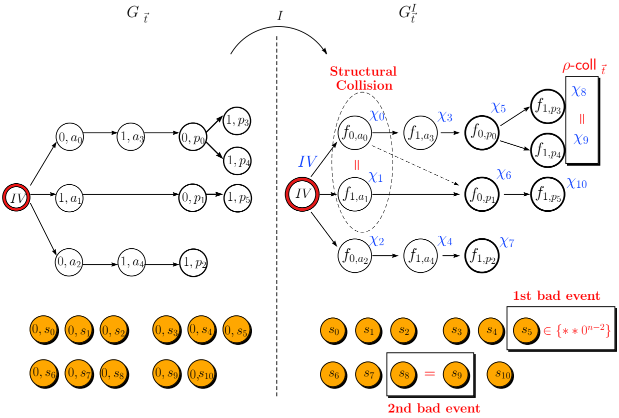
Proof Intuition: Step 5 (function to graph)

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE



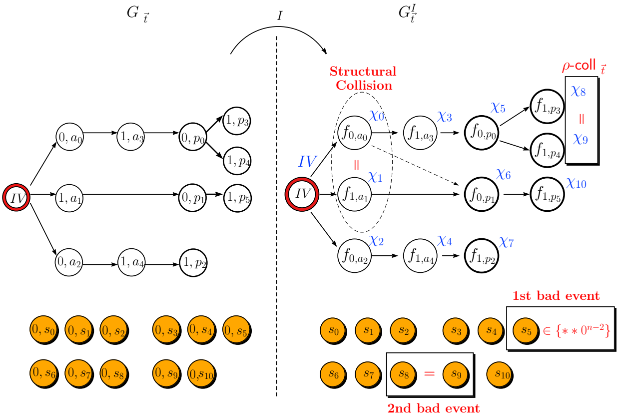
- Structural collision: when two unequal values lead to same function.
- Natural isomorphism between the 2 graphs no longer maintained.
- This can never happen in SUNDAE. Mapping from $\delta, X \rightarrow f_{\delta,X}$ is injective.

Proof Intuition: Step 5 (function to graph)



- The next event is ρ-coll$_{\vec{t}}$: if labels of 2 nodes become equal.
- May occur due to randomness introduced by the URF ρ.
- We use graph-theoretic arguments to bound prob of ρ-coll$_{\vec{t}}$.

Proof Intuition: Step 5 (function to graph)



- Now straightforward to apply H-coeffs. Adding we get bound in Thm 1.

$$\Delta_{A_{\text{chopxor}}}(\text{enc-stream}, \text{dec-stream}; \$^s, \text{dec-stream}^*) \leq$$

$$\frac{(\sigma_P + \sigma_C)^2}{2^{n+1}} + \frac{4(\sigma_P + \sigma_C)}{2^n} + \frac{(4 + \sigma_A + \sigma_P + \sigma_C)^2}{2^n} + \frac{4(q + q_v)^2}{2^n}. \quad (10)$$

Performance

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Software

- Platforms: Cortex-A57 core of a Samsung Exynos 7420 CPU (ARMv8 platform), Intel Core i7-6700 CPU (Skylake)
- Message lengths: $\ell = 2^b$ bytes, with $6 \leq b \leq 11$, with comb scheduling.

- On Intel, SUNDAE is around 3% slower than two passes of CBC; on ARM, 7%.
- For short messages onlyaround 11% worse than for longer messages.

- Compared to the single-pass COFB, SUNDAE has an overhead of 60% for short and 80% for long messages on Intel
- And 35% for short and 80% for long messages on ARM.

Performance

Table: ARMv8 platform (embedded)

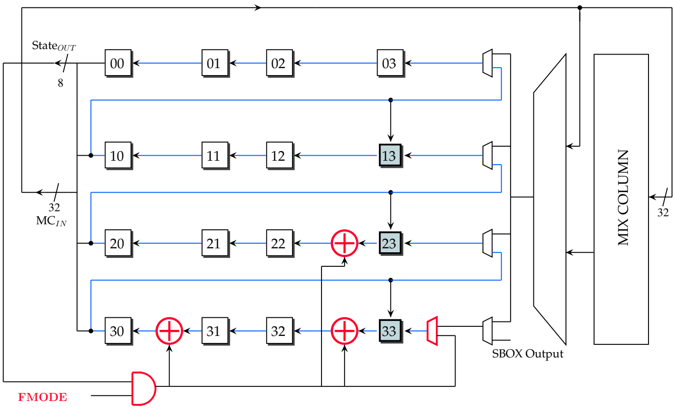
	message length (bytes)						
Algorithm	64	128	256	512	1024	2048	mix
CBC (S)	2.69	2.54	2.39	2.30	2.26	2.25	2.38
CBC (P)	1.42	1.14	1.02	0.95	0.92	0.90	1.00
COFB (S)	3.99	3.34	2.96	2.78	2.72	2.71	2.98
COFB (P)	2.98	1.89	1.49	1.32	1.25	1.22	1.52
SUNDAE (S)	5.42	5.14	5.02	4.92	4.86	4.84	4.97
SUNDAE (P)	3.16	2.95	2.85	2.80	2.78	2.76	2.84

Performance

Table: Intel Skylake platform (server)

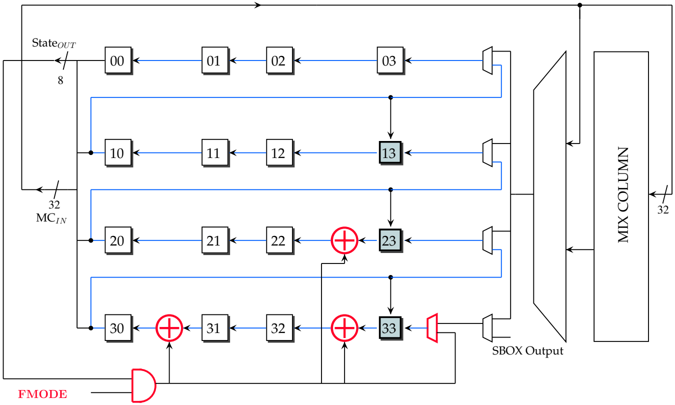
	message length (bytes)						
Algorithm	64	128	256	512	1024	2048	mix
CBC (S)	2.90	2.75	2.68	2.63	2.60	2.59	2.67
CBC (P)	0.64	0.64	0.63	0.63	0.63	0.63	0.64
COFB (S)	3.71	3.32	3.12	3.02	2.97	2.96	3.12
COFB (P)	1.03	0.95	0.90	0.87	0.86	0.85	0.90
SUNDAE (S)	6.00	5.71	5.57	5.46	5.40	5.37	5.52
SUNDAE (P)	1.36	1.31	1.29	1.27	1.26	1.26	1.28

On ASIC



- Replace 2x on $GF(2^{128}) \rightarrow$ eight 2x over $GF(2^{16})/ < x^{16} + x^5 + x^3 + x + 1 >$
- If $c_0, c_1, \ldots, c_{15}$ denote the individual bytes
- i^{th} bits of each byte is an element of $GF(2^{16})$
- We have: $f(c_0, \ldots, c_{15}) = c_1, c_2, \ldots, c_{11} \oplus c_0, c_{12}, c_{13} \oplus c_0, c_{14}, c_{15} \oplus c_0, c_0$

On ASIC



- Fits well into the bytewise AES circuit: only few gates required.
- Mapping from $\delta, X \rightarrow f_{\delta,X}$ is still injective.
- No change in security guarantees.
- No additional state needs to be stored/updated.

Performance On ASIC

Mode	Underlying Cipher	Blocksize/ Keysize	Area (GE)	Power (μW)
CLOC (A)	AES-128	128/128	3110	131.1
CLOC (C)	AES-128	128/128	4310	156.6
SILC (A)	AES-128	128/128	3110	131.0
SILC (C)	AES-128	128/128	4220	155.6
AES-OTR (A)	AES-128	128/128	4720	164.3
AES-OTR (C)	AES-128	128/128	6770	205.4
AES-SUNDAE	AES-128	128/128	**2524**	126.1
Present-SUNDAE	Present	64/80	**1452**	50.9

Table: Implementation results for CLOC, SILC, AES-OTR, and SUNDAE. (Power reported at 10 MHz, A: Aggressive, C: Conservative

THANK YOU