

Improved Rectangle Attacks on SKINNY and CRAFT

Hosein Hadipour Nasour Bagheri

FSE 2022 - Athens, Greece

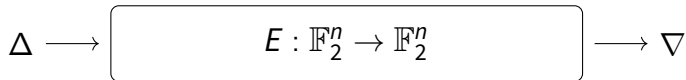
Outline

- 1 Boomerang and Sandwich Distinguishers
- 2 Our Method To Find Sandwich Distinguishers
- 3 BCT Framework and Our New Tools
- 4 Application to CRAFT
- 5 Application to SKINNY
- 6 Conclusion

Boomerang and Sandwich Distinguishers

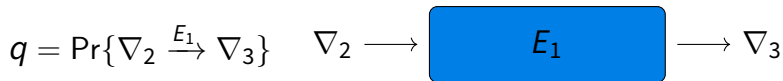
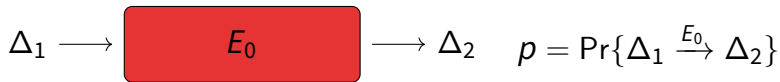


Long Weak Differentials V.S. Two Short Strong Differentials



$$0 \leq \Pr\{\Delta \xrightarrow{E} \nabla\} \lll 2^{-n}$$

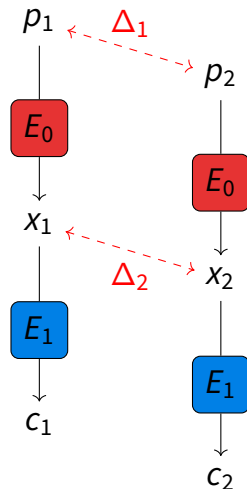
Long Weak Differentials V.S. Two Short Strong Differentials



$$p^2 q^2 \ggg 2^{-n}$$

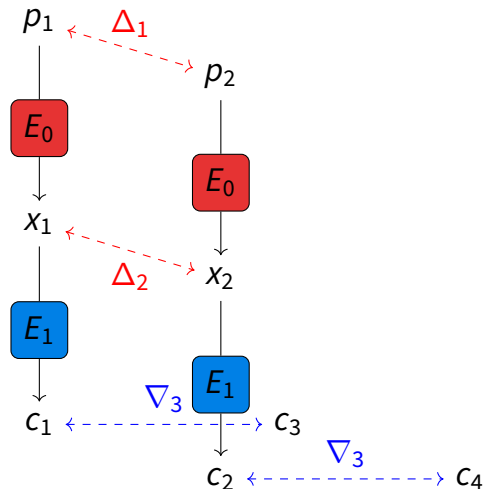
Combine Two Short Differentials in ACPC Setting [Wag99]

$$\Pr\{\Delta_1 \xrightarrow{E_0} \Delta_2\} = p$$



Combine Two Short Differentials in ACPC Setting [Wag99]

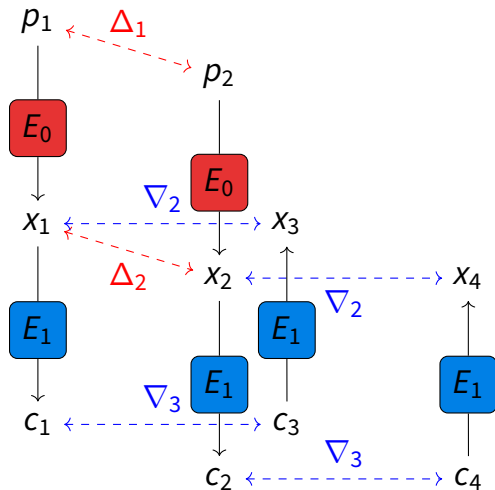
$$\Pr\{\Delta_1 \xrightarrow{E_0} \Delta_2\} = p$$



Combine Two Short Differentials in ACPC Setting [Wag99]

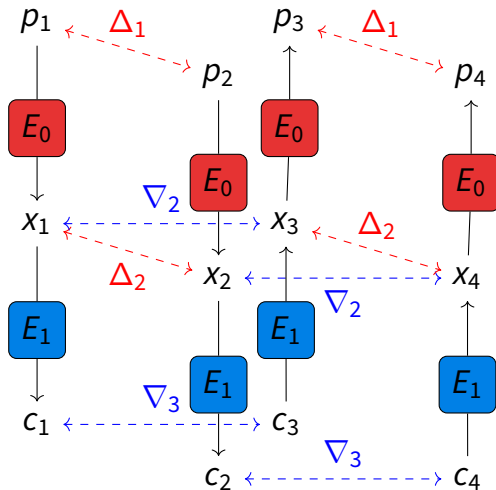
$$\Pr\{\Delta_1 \xrightarrow{E_0} \Delta_2\} = p$$

$$\Pr\{\nabla_2 \xrightarrow{E_1} \nabla_3\} = q$$



Combine Two Short Differentials in ACPC Setting [Wag99]

$$\Pr\{p_3 \oplus p_4 = \Delta_1\} = p^2 \times q^2$$



Upper and Lower Parts are Not Independent in Practice!

From the attacker's perspective:

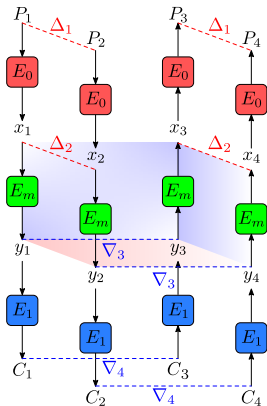
✔ Dependency can have a **positive** effect

- Feistel Switch [Wag99]
- Ladder Switch and S-box Switch [BK09]

⚠ Dependency can have a **negative** effect

- Inconsistency between the upper and lower trail [Mur11]

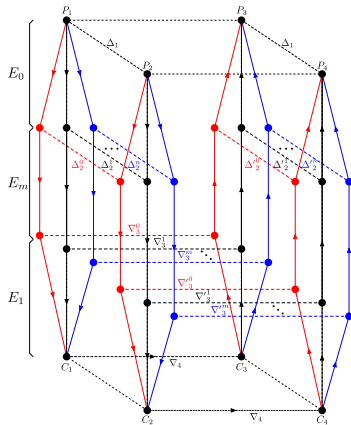
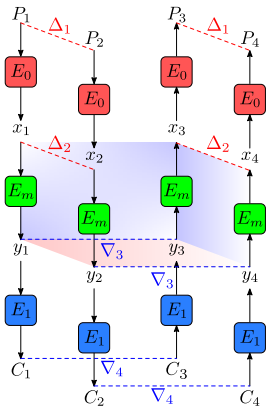
Sandwich Distinguisher [DKS10; DKS14]



$$\Pr(P_3 \oplus P_4 = \Delta_1) \approx p^2 \times r \times q^2$$

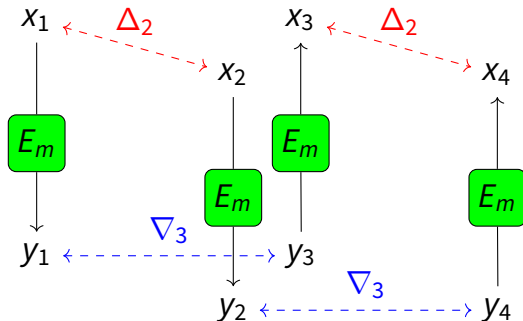
$$r = r(\Delta_2, \nabla_3) = \Pr\{E_m^{-1}(E_m(x) \oplus \nabla_3) \oplus E_m^{-1}(E_m(x \oplus \Delta_2) \oplus \nabla_3) = \Delta_2\}$$

Sandwich Distinguisher [DKS10; DKS14]



$$\Pr(P_3 \oplus P_4 = \Delta_1) = \sum_{\Delta_2, \Delta'_2, \nabla_3, \nabla'_3} p_{\Delta_2} \times p_{\Delta'_2} \times r(\Delta_2, \Delta'_2, \nabla_3, \nabla'_3) \times q_{\nabla_3} \times q_{\nabla'_3}$$

Ladder Switch

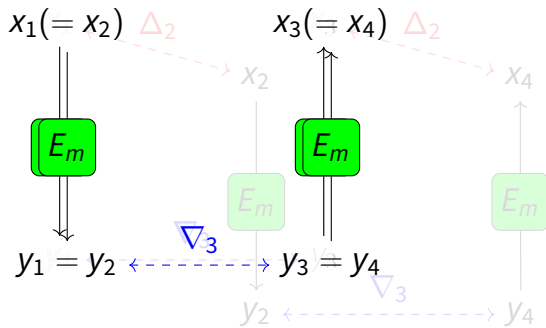


$$r = r(\Delta_2, \nabla_3) = \Pr\{E_m^{-1}(E_m(x) \oplus \nabla_3) \oplus E_m^{-1}(E_m(x \oplus \Delta_2) \oplus \nabla_3) = \Delta_2\}$$

$$\Delta_2 = 0 \implies r = r(0, \nabla_3) = 1$$

$$\nabla_3 = 0 \implies r = r(\Delta_2, 0) = 1$$

Ladder Switch

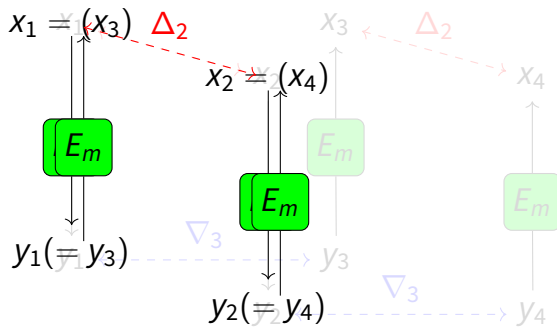


$$r = r(\Delta_2, \nabla_3) = \Pr\{E_m^{-1}(E_m(x) \oplus \nabla_3) \oplus E_m^{-1}(E_m(x \oplus \Delta_2) \oplus \nabla_3) = \Delta_2\}$$

$$\Delta_2 = 0 \implies r = r(0, \nabla_3) = 1$$

$$\nabla_3 = 0 \implies r = r(\Delta_2, 0) = 1$$

Ladder Switch



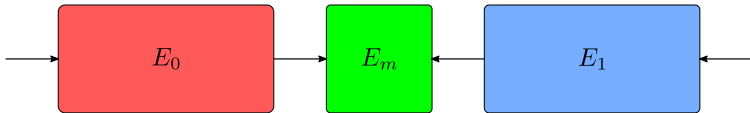
$$r = r(\Delta_2, \nabla_3) = \Pr\{E_m^{-1}(E_m(x) \oplus \nabla_3) \oplus E_m^{-1}(E_m(x \oplus \Delta_2) \oplus \nabla_3) = \Delta_2\}$$

$$\Delta_2 = 0 \implies r = r(0, \nabla_3) = 1$$

$$\nabla_3 = 0 \implies r = r(\Delta_2, 0) = 1$$

Effective Parameters in p^2q^2r for SPN Ciphers

- ✔ p is mostly determined by the number of active S-boxes in E_0
- ✔ q is mostly determined by the number of active S-boxes in E_1
- ✔ r is mostly determined by the number of **common** active S-boxes in E_m
- ⚠ Active S-boxes in E_0, E_1 are more expensive than common active S-boxes in E_m



Our Method To Find Sandwich Distinguishers



Our Method to Find Sandwich Distinguishers

Our method consists of 3 main steps:

- ➡ Find the truncated upper and lower trails minimizing:
 - number of active S-boxes in outer parts
 - and number of common active S-boxes in the middle part
- ➡ Instantiate the discovered truncated trails with concrete differential trails
- ➡ Compute p , q and r to derive the entire probability, i.e., p^2q^2r

Our Method to Find Sandwich Distinguishers

Our method consists of 3 main steps:

- ➔ Find the truncated upper and lower trails minimizing:
 - number of active S-boxes in outer parts
 - and number of common active S-boxes in the middle part
- ➔ Instantiate the discovered truncated trails with concrete differential trails
- ➔ Compute p , q and r to derive the entire probability, i.e., p^2q^2r

Our Method to Find Sandwich Distinguishers

Our method consists of 3 main steps:

- ➔ Find the truncated upper and lower trails minimizing:
 - number of active S-boxes in outer parts
 - and number of common active S-boxes in the middle part
- ➔ Instantiate the discovered truncated trails with concrete differential trails
- ➔ Compute p , q and r to derive the entire probability, i.e., p^2q^2r

Our Method to Find Sandwich Distinguishers

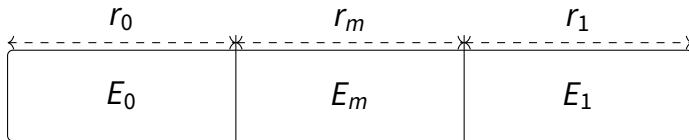
Our method consists of 3 main steps:

- ➔ Find the truncated upper and lower trails minimizing:
 - number of active S-boxes in outer parts
 - and number of common active S-boxes in the middle part
- ➔ Instantiate the discovered truncated trails with concrete differential trails
- ➔ Compute p , q and r to derive the entire probability, i.e., p^2q^2r

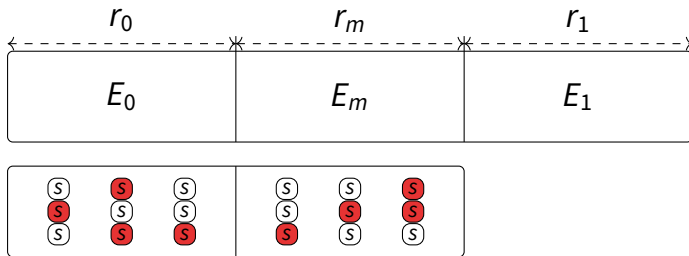
Finding Appropriate Truncated Upper and Lower Trails

E

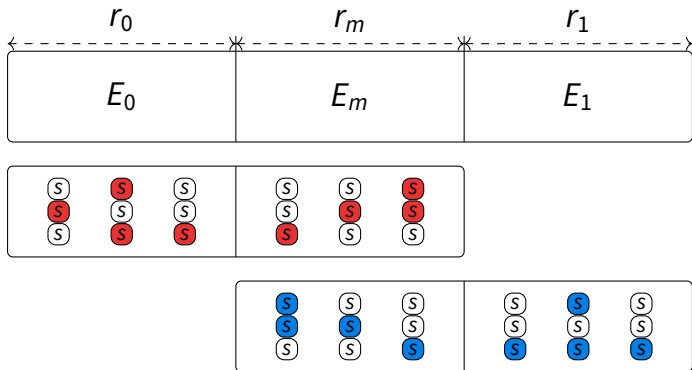
Finding Appropriate Truncated Upper and Lower Trails



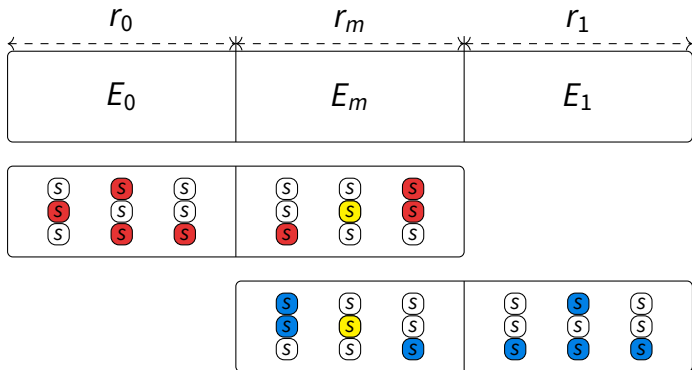
Finding Appropriate Truncated Upper and Lower Trails



Finding Appropriate Truncated Upper and Lower Trails

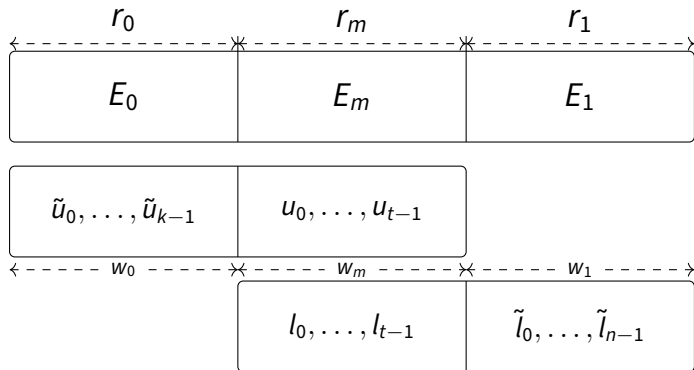


Finding Appropriate Truncated Upper and Lower Trails



$$u_i - s_i \geq 0, \quad l_i - s_i \geq 0, \quad -u_i - l_i + s_i \geq -1$$

Finding Appropriate Truncated Upper and Lower Trails

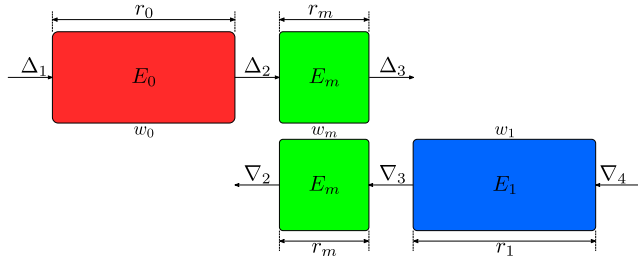


$$\min \sum_{i=0}^{k-1} w_0 \cdot \tilde{u}_i + \sum_{j=0}^{t-1} w_m \cdot s_j + \sum_{k=0}^{n-1} w_1 \cdot \tilde{l}_k.$$

$$u_i - s_i \geq 0, \quad \ell_i - s_i \geq 0, \quad -u_i - \ell_i + s_i \geq -1.$$

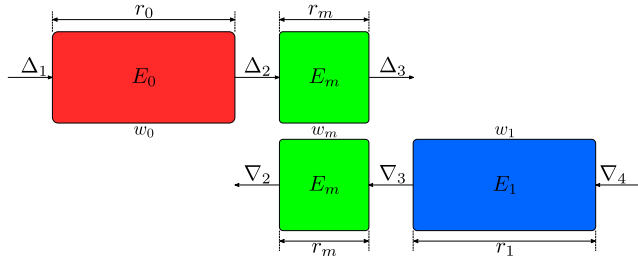
Instantiating Truncated Trails with Concrete Differentials

- ➡ We Instantiate the first and last parts with concrete bit-wise differentials
- ➡ To compute p , q and r we fix the differences at only four positions
- ⚠ Our distinguishers do not rely on differential characteristics for E_0, E_1, E_m



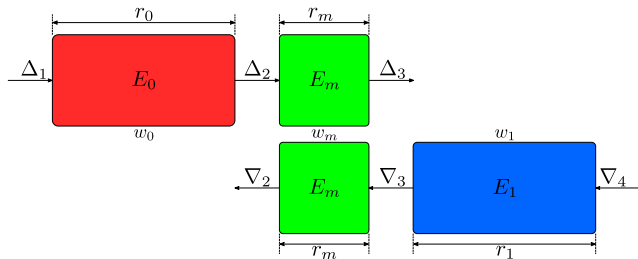
Instantiating Truncated Trails with Concrete Differentials

- ➡ We Instantiate the first and last parts with concrete bit-wise differentials
- ➡ To compute p , q and r we fix the differences at only four positions
- ⚠ Our distinguishers do not rely on differential characteristics for E_0, E_1, E_m



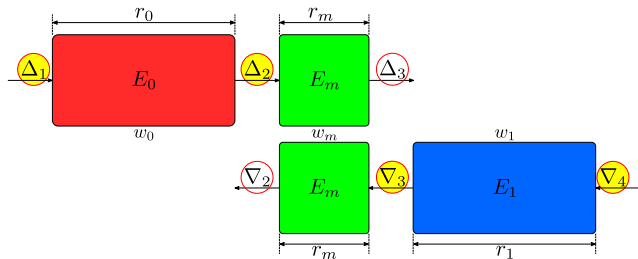
Instantiating Truncated Trails with Concrete Differentials

- ➡ We Instantiate the first and last parts with concrete bit-wise differentials
- ➡ To compute p , q and r we fix the differences at only four positions
- ⚠ Our distinguishers do not rely on differential characteristics for E_0, E_1, E_m



Instantiating Truncated Trails with Concrete Differentials

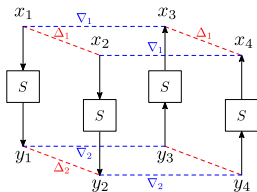
- ➡ We Instantiate the first and last parts with concrete bit-wise differentials
- ➡ To compute p , q and r we fix the differences at only four positions
- ⚠ Our distinguishers do not rely on differential characteristics for E_0, E_1, E_m



BCT Framework And Our New Tools



BCT Framework



Δ_1

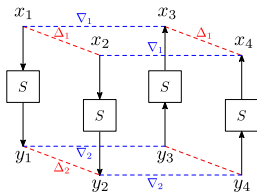
Δ_2

∇_1

∇_2

- ✓ $\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2) = \{x : S(x) \oplus S(x \oplus \Delta_1) = \Delta_2\}, \quad \text{DDT}(\Delta_1, \Delta_2) = \#\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)$
- ✓ $\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) = \{x : S^{-1}(S(x) \oplus \nabla_2) \oplus S^{-1}(S(x \oplus \Delta_1) \oplus \nabla_2) = \Delta_1\}, \quad \text{BCT}(\Delta_1, \nabla_2) = \#\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \text{ [Cid+18]}$
- ✓ $\text{UBCT}(\Delta_1, \Delta_2, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)\} \quad [\text{WP19}]$
- ✓ $\text{LBCT}(\Delta_1, \nabla_1, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\nabla_1, \nabla_2)\} \quad [\text{SQH19}]$

BCT Framework

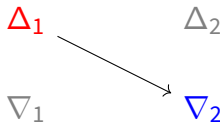
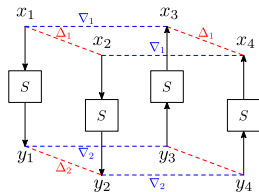


$$\Delta_1 \longrightarrow \Delta_2$$

$$\nabla_1 \longrightarrow \nabla_2$$

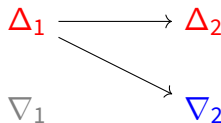
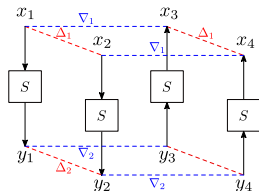
- ✓ $\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2) = \{x : S(x) \oplus S(x \oplus \Delta_1) = \Delta_2\}, \quad \text{DDT}(\Delta_1, \Delta_2) = \#\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)$
- ✓ $\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) = \{x : S^{-1}(S(x) \oplus \nabla_2) \oplus S^{-1}(S(x \oplus \Delta_1) \oplus \nabla_2) = \Delta_1\}, \quad \text{BCT}(\Delta_1, \nabla_2) = \#\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \text{ [Cid+18]}$
- ✓ $\text{UBCT}(\Delta_1, \Delta_2, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)\} \quad [\text{WP19}]$
- ✓ $\text{LBCT}(\Delta_1, \nabla_1, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\nabla_1, \nabla_2)\} \quad [\text{SQH19}]$

BCT Framework



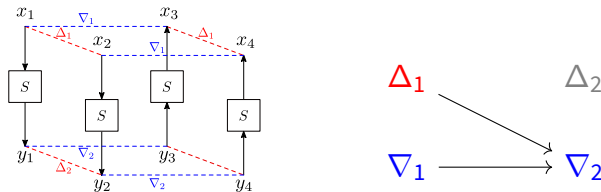
- ✓ $\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2) = \{x : S(x) \oplus S(x \oplus \Delta_1) = \Delta_2\}, \quad \text{DDT}(\Delta_1, \Delta_2) = \#\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)$
- ✓ $\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) = \{x : S^{-1}(S(x) \oplus \nabla_2) \oplus S^{-1}(S(x \oplus \Delta_1) \oplus \nabla_2) = \Delta_1\}, \quad \text{BCT}(\Delta_1, \nabla_2) = \#\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \text{ [Cid+18]}$
- ✓ $\text{UBCT}(\Delta_1, \Delta_2, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)\} \quad [\text{WP19}]$
- ✓ $\text{LBCT}(\Delta_1, \nabla_1, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\nabla_1, \nabla_2)\} \quad [\text{SQH19}]$

BCT Framework



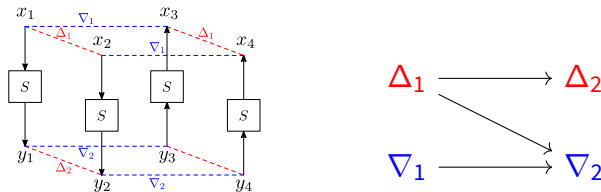
- ✓ $\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2) = \{x : S(x) \oplus S(x \oplus \Delta_1) = \Delta_2\}, \quad \text{DDT}(\Delta_1, \Delta_2) = \#\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)$
- ✓ $\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) = \{x : S^{-1}(S(x) \oplus \nabla_2) \oplus S^{-1}(S(x \oplus \Delta_1) \oplus \nabla_2) = \Delta_1\}, \quad \text{BCT}(\Delta_1, \nabla_2) = \#\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \text{ [Cid+18]}$
- ✓ $\text{UBCT}(\Delta_1, \Delta_2, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)\} \quad [\text{WP19}]$
- ✓ $\text{LBCT}(\Delta_1, \nabla_1, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\nabla_1, \nabla_2)\} \quad [\text{SQH19}]$

BCT Framework



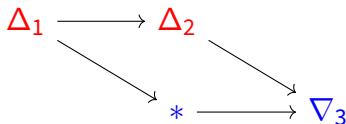
- ✓ $\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2) = \{x : S(x) \oplus S(x \oplus \Delta_1) = \Delta_2\}, \quad \text{DDT}(\Delta_1, \Delta_2) = \#\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)$
- ✓ $\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) = \{x : S^{-1}(S(x) \oplus \nabla_2) \oplus S^{-1}(S(x \oplus \Delta_1) \oplus \nabla_2) = \Delta_1\}, \quad \text{BCT}(\Delta_1, \nabla_2) = \#\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \text{ [Cid+18]}$
- ✓ $\text{UBCT}(\Delta_1, \Delta_2, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)\} \quad [\text{WP19}]$
- ✓ $\text{LBCT}(\Delta_1, \nabla_1, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\nabla_1, \nabla_2)\} \quad [\text{SQH19}]$

BCT Framework



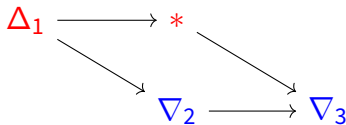
- ✓ $\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2) = \{x : S(x) \oplus S(x \oplus \Delta_1) = \Delta_2\}, \quad \text{DDT}(\Delta_1, \Delta_2) = \#\mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)$
- ✓ $\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) = \{x : S^{-1}(S(x) \oplus \nabla_2) \oplus S^{-1}(S(x \oplus \Delta_1) \oplus \nabla_2) = \Delta_1\}, \quad \text{BCT}(\Delta_1, \nabla_2) = \#\mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \text{ [Cid+18]}$
- ✓ $\text{UBCT}(\Delta_1, \Delta_2, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\Delta_1, \Delta_2)\} \quad [\text{WP19}]$
- ✓ $\text{LBCT}(\Delta_1, \nabla_1, \nabla_2) = \#\{x : x \in \mathcal{X}_{\text{BCT}}(\Delta_1, \nabla_2) \cap \mathcal{X}_{\text{DDT}}(\nabla_1, \nabla_2)\} \quad [\text{SQH19}]$

Double Boomerang Connectivity Table (DBCT)



- ✓ $\text{DBCT}^{\perp}(\Delta_1, \Delta_2, \nabla_3) = \sum_{\nabla_2} \text{UBCT}(\Delta_1, \nabla_2, \Delta_2) \cdot \text{LBCT}(\Delta_2, \nabla_3, \nabla_2)$
- ✓ $\text{DBCT}^{\perp}(\Delta_1, \nabla_2, \nabla_3) = \sum_{\Delta_2} \text{UBCT}(\Delta_1, \nabla_2, \Delta_2) \cdot \text{LBCT}(\Delta_2, \nabla_3, \nabla_2).$
- ✓ $\text{DBCT}(\Delta_1, \nabla_3) = \sum_{\Delta_2} \text{DBCT}^{\perp}(\Delta_1, \Delta_2, \nabla_3) = \sum_{\nabla_2} \text{DBCT}^{\perp}(\Delta_1, \nabla_2, \nabla_3).$

Double Boomerang Connectivity Table (DBCT)

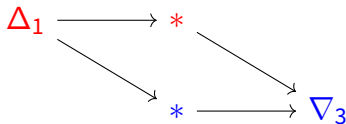


✓ $\text{DBCT}^{\perp}(\Delta_1, \Delta_2, \nabla_3) = \sum_{\nabla_2} \text{UBCT}(\Delta_1, \nabla_2, \Delta_2) \cdot \text{LBCT}(\Delta_2, \nabla_3, \nabla_2)$

✓ $\text{DBCT}^{\perp}(\Delta_1, \nabla_2, \nabla_3) = \sum_{\Delta_2} \text{UBCT}(\Delta_1, \nabla_2, \Delta_2) \cdot \text{LBCT}(\Delta_2, \nabla_3, \nabla_2).$

✓ $\text{DBCT}(\Delta_1, \nabla_3) = \sum_{\Delta_2} \text{DBCT}^{\perp}(\Delta_1, \Delta_2, \nabla_3) = \sum_{\nabla_2} \text{DBCT}^{\perp}(\Delta_1, \nabla_2, \nabla_3).$

Double Boomerang Connectivity Table (DBCT)

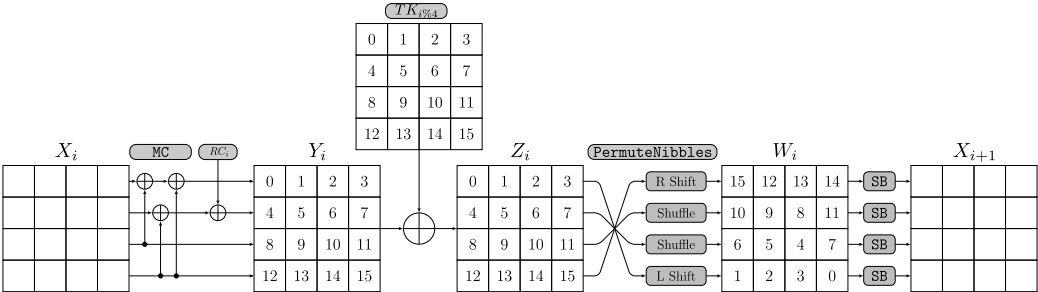


- ✓ $\text{DBCT}^{\perp}(\Delta_1, \Delta_2, \nabla_3) = \sum_{\nabla_2} \text{UBCT}(\Delta_1, \nabla_2, \Delta_2) \cdot \text{LBCT}(\Delta_2, \nabla_3, \nabla_2)$
- ✓ $\text{DBCT}^{\perp}(\Delta_1, \nabla_2, \nabla_3) = \sum_{\Delta_2} \text{UBCT}(\Delta_1, \nabla_2, \Delta_2) \cdot \text{LBCT}(\Delta_2, \nabla_3, \nabla_2).$
- ✓ $\text{DBCT}(\Delta_1, \nabla_3) = \sum_{\Delta_2} \text{DBCT}^{\perp}(\Delta_1, \Delta_2, \nabla_3) = \sum_{\nabla_2} \text{DBCT}^{\perp}(\Delta_1, \nabla_2, \nabla_3).$

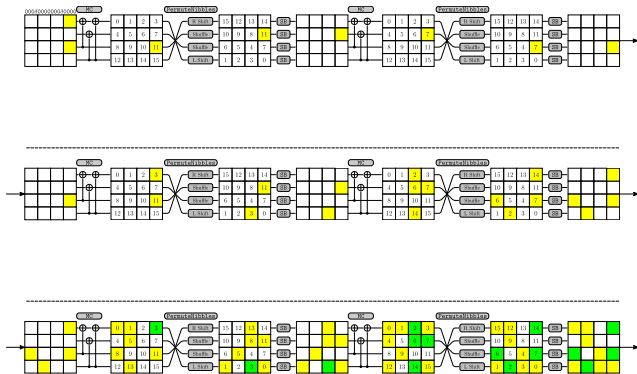
Application to CRAFT



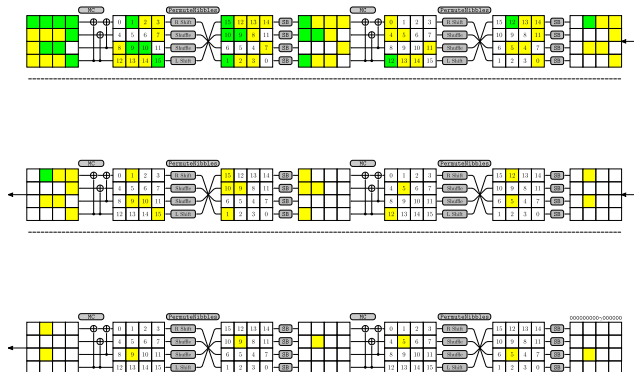
CRAFT [Bei+19]



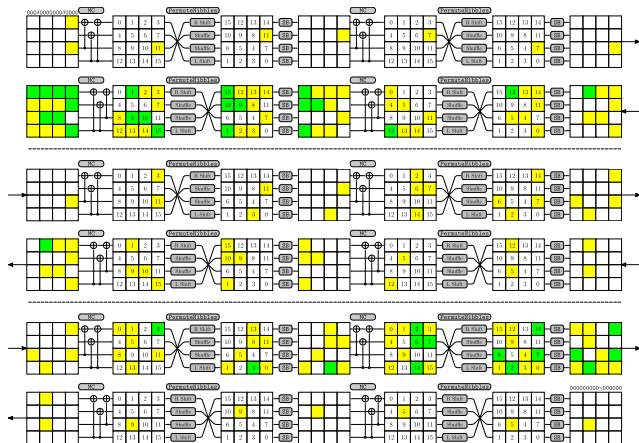
A 6-round ST Deterministic Distinguisher for CRAFT



A 6-round ST Deterministic Distinguisher for CRAFT

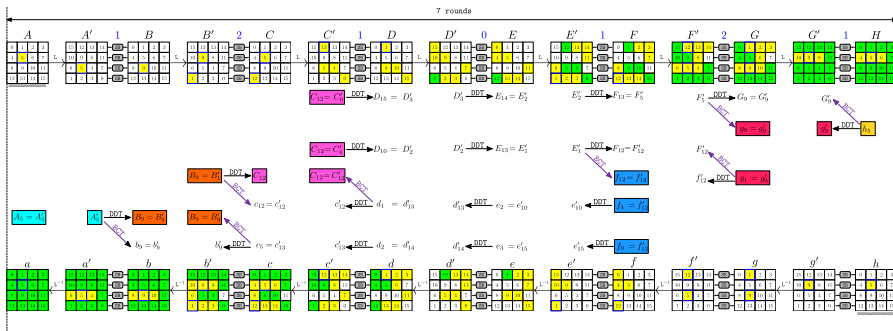


A 6-round ST Deterministic Distinguisher for CRAFT

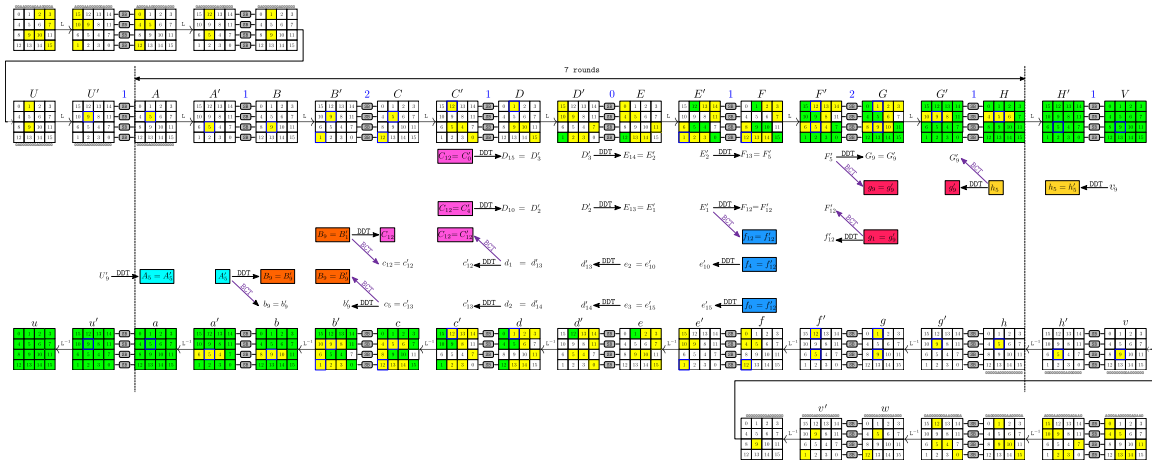


$\text{Pr} = 1$

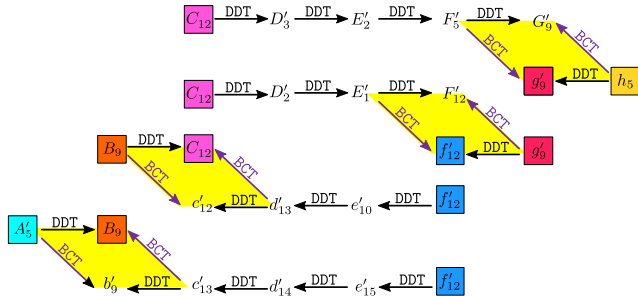
A 7-round Distinguisher (Extendable up to 14 rounds)



A 7-round Distinguisher (Extendable up to 14 rounds)



A 7-round Distinguisher (Extendable up to 14 rounds)



$$\text{DBCT}_{\text{total}} = \text{DBCT}^{\perp}(A_5, B_9, c_5) \cdot \text{DBCT}^{\perp}(B_9, C_{12}, d_1) \cdot \text{DBCT}^{\perp}(E'_1, f'_{12}, g'_9) \cdot \text{DBCT}^{\perp}(F'_5, g'_9, h_5)$$

$$\text{Pr}_{\text{total}} = \Pr(d_1 \xleftarrow{2 \text{ DDT}} f'_{12}) \cdot \Pr(c_5 \xleftarrow{3 \text{ DDT}} f'_{12}) \cdot \Pr(C_{12} \xrightarrow{2 \text{ DDT}} E'_1) \cdot \Pr(C_{12} \xrightarrow{3 \text{ DDT}} F'_5)$$

$$r = 2^{-8 \cdot n} \cdot \sum_{B_9} \sum_{C_{12}} \sum_{g'_9} \sum_{f'_{12}} \sum_{c_5} \sum_{d_1} \sum_{E'_1} \sum_{F'_5} \text{DBCT}_{\text{total}} \cdot \text{Pr}_{\text{total}}.$$

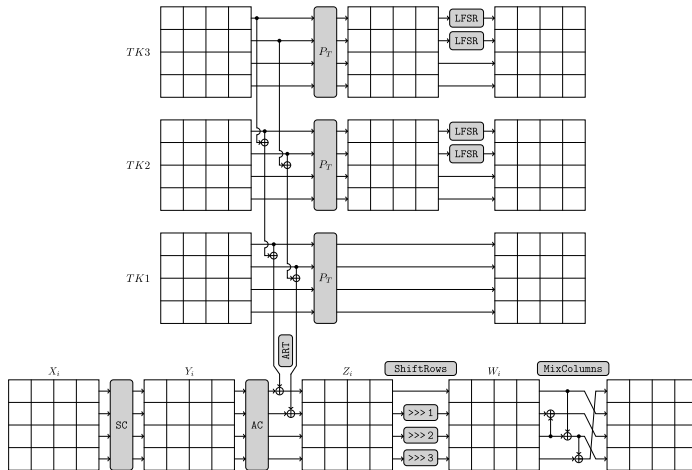
Summary of Our Distinguishers for CRAFT

Distinguisher Type	# Rounds	Probability	Reference
<i>ST-Differential</i>	9	$2^{-40.20}$	[Had+19]
	10	$2^{-44.89}$	
	11	$2^{-49.79}$	
	12	$2^{-54.48}$	
	13	$2^{-59.13}$	
	14	$2^{-63.80}$	
<i>ST-Boomerang</i>	6	1	This Paper
	7	2^{-4}	
	8	2^{-8}	
	9	$2^{-14.76}$	
	10	$2^{-19.83}$	
	11	$2^{-24.90}$	
	12	$2^{-34.89}$	
	13	$2^{-44.89}$	
	14	$2^{-55.85}$	

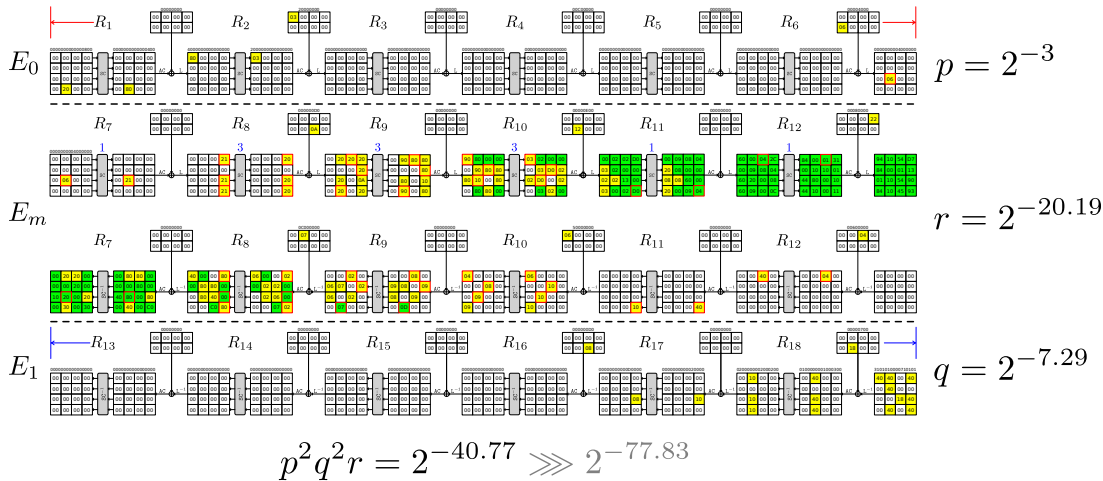
Application to SKINNY



SKINNY [Bei+16]



18-round Practical Sandwich Distinguisher for SKINNY-128-256



Summary of Our Distinguishers for SKINNY

Version	n	#Rounds	Probability	
			Our Distinguisher	[SQH19]
SKINNY- $n-2n$	64	17	$2^{-26.54}$ (II)	$2^{-29.78}$
		18	$2^{-37.90}$ (II)	$2^{-45.14}$
		19	$2^{-51.08}$ (II)	$2^{-65.62}$
	128	18	$2^{-40.77}$ (II)	$2^{-77.83}$
		19	$2^{-58.33}$ (II)	$2^{-97.53}$
		20	$2^{-85.31}$ (I)	$2^{-128.65}$
		21	$2^{-114.07}$ (II)	$2^{-171.77}$
SKINNY- $n-3n$	64	22	$2^{-38.84}$ (I)	$2^{-42.98}$
		23	$2^{-52.84}$ (I)	$2^{-67.36}$
	128	22	$2^{-40.57}$ (I)	$2^{-48.30}$
		23	$2^{-56.47}$ (I)	$2^{-75.86}$
		24	$2^{-87.39}$ (I)	$2^{-107.86}$
		25	$2^{-116.59}$ (I)	$2^{-141.66}$

Summary of Our Key Recovery Attacks

Scheme	#rounds	Data	Memory	Time	Attack	P_s	Reference
SKINNY-64-128	23/36	$2^{60.54}$	$2^{60.9}$	$2^{120.7}$	Rectangle	0.977	This paper
SKINNY-64-192	29/40	$2^{61.42}$	2^{80}	2^{178}	Rectangle	0.977	This paper
SKINNY-128-256	24/48	$2^{125.21}$	$2^{125.54}$	$2^{209.85}$	Rectangle	0.977	This paper
SKINNY-128-384	30/56	$2^{125.29}$	$2^{125.8}$	$2^{361.68}$	Rectangle	0.977	This paper
CRAFT	18/32	$2^{60.92}$	2^{84}	$2^{101.7}$	Rectangle	0.977	This paper
SKINNY-64-128	23/36	$2^{62.47}$	2^{124}	$2^{125.91}$	Impossible	1	[LGS17]
SKINNY-64-192	27/40	$2^{63.5}$	2^{80}	$2^{165.5}$	Rectangle	0.916	[LGS17]
SKINNY-128-256	23/48	$2^{124.47}$	2^{248}	$2^{251.47}$	Impossible	1	[LGS17]
SKINNY-128-384	28/56	2^{122}	$2^{122.32}$	$2^{315.25}$	Rectangle	0.8315	[Zha+20]

Conclusion



Our Main Contributions

- ✔ We introduced a heuristic method to search for sandwich distinguishers
- ✔ We introduced new tools in BCT framework (DBCT, ...)
- ✔ We significantly improved the rectangle attacks on SKINNY and CRAFT

Thanks for your attention!

<https://github.com/hadipourh/Boomerang>

Bibliography I

- [Bei+16] Christof Beierle et al. **The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS**. CRYPTO (2). Vol. 9815. Lecture Notes in Computer Science. Springer, 2016, pp. 123–153.
- [Bei+19] Christof Beierle et al. **CRAFT: Lightweight Tweakable Block Cipher with Efficient Protection Against DFA Attacks**. *IACR Trans. Symmetric Cryptol.* 2019.1 (2019), pp. 5–45.
- [BK09] Alex Biryukov and Dmitry Khovratovich. **Related-Key Cryptanalysis of the Full AES-192 and AES-256**. ASIACRYPT. Vol. 5912. Lecture Notes in Computer Science. Springer, 2009, pp. 1–18.
- [Cid+18] Carlos Cid et al. **Boomerang Connectivity Table: A New Cryptanalysis Tool**. EUROCRYPT (2). Vol. 10821. Lecture Notes in Computer Science. Springer, 2018, pp. 683–714.
- [DKS10] Orr Dunkelman, Nathan Keller, and Adi Shamir. **A Practical-Time Related-Key Attack on the KASUMI Cryptosystem Used in GSM and 3G Telephony**. CRYPTO. Vol. 6223. Lecture Notes in Computer Science. Springer, 2010, pp. 393–410.
- [DKS14] Orr Dunkelman, Nathan Keller, and Adi Shamir. **A Practical-Time Related-Key Attack on the KASUMI Cryptosystem Used in GSM and 3G Telephony**. *J. Cryptol.* 27.4 (2014), pp. 824–849.

Bibliography II

- [Had+19] Hosein Hadipour et al. **Comprehensive security analysis of CRAFT**. *IACR Trans. Symmetric Cryptol.* 2019.4 (2019), pp. 290–317.
- [LGS17] Guozhen Liu, Mohona Ghosh, and Ling Song. **Security Analysis of SKINNY under Related-Tweakey Settings**. *IACR Transactions on Symmetric Cryptology* 2017.3 (Sept. 2017).
<https://tosc.iacr.org/index.php/ToSC/article/view/765>, pp. 37–72. DOI:
[10.13154/tosc.v2017.i3.37-72](https://doi.org/10.13154/tosc.v2017.i3.37-72).
- [Mur11] Sean Murphy. **The Return of the Cryptographic Boomerang**. *IEEE Trans. Inf. Theory* 57.4 (2011), pp. 2517–2521.
- [SQH19] Ling Song, Xianrui Qin, and Lei Hu. **Boomerang Connectivity Table Revisited. Application to SKINNY and AES**. *IACR Trans. Symmetric Cryptol.* 2019.1 (2019), pp. 118–141.
- [Wag99] David A. Wagner. **The Boomerang Attack**. FSE. Vol. 1636. Lecture Notes in Computer Science. Springer, 1999, pp. 156–170.
- [WP19] Haoyang Wang and Thomas Peyrin. **Boomerang Switch in Multiple Rounds. Application to AES Variants and Deoxys**. *IACR Trans. Symmetric Cryptol.* 2019.1 (2019), pp. 142–169.

Bibliography III

- [Zha+20] Boxin Zhao et al. **Generalized related-key rectangle attacks on block ciphers with linear key schedule: applications to SKINNY and GIFT.** *Designs, Codes and Cryptography* 88.6 (2020), pp. 1103–1126.