Integral Cryptanalysis on Reduced-Round Tweakable TWINE

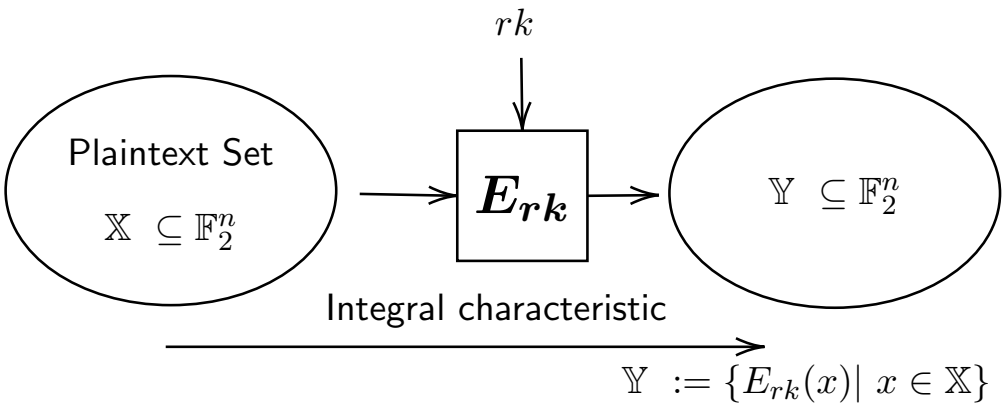
Muhammad ElSheikh, Amr M. Youssef

Concordia Institute for Information Systems Engineering,
Concordia University, Montréal, Québec, Canada

CANS 2020
December 14-16, 2020

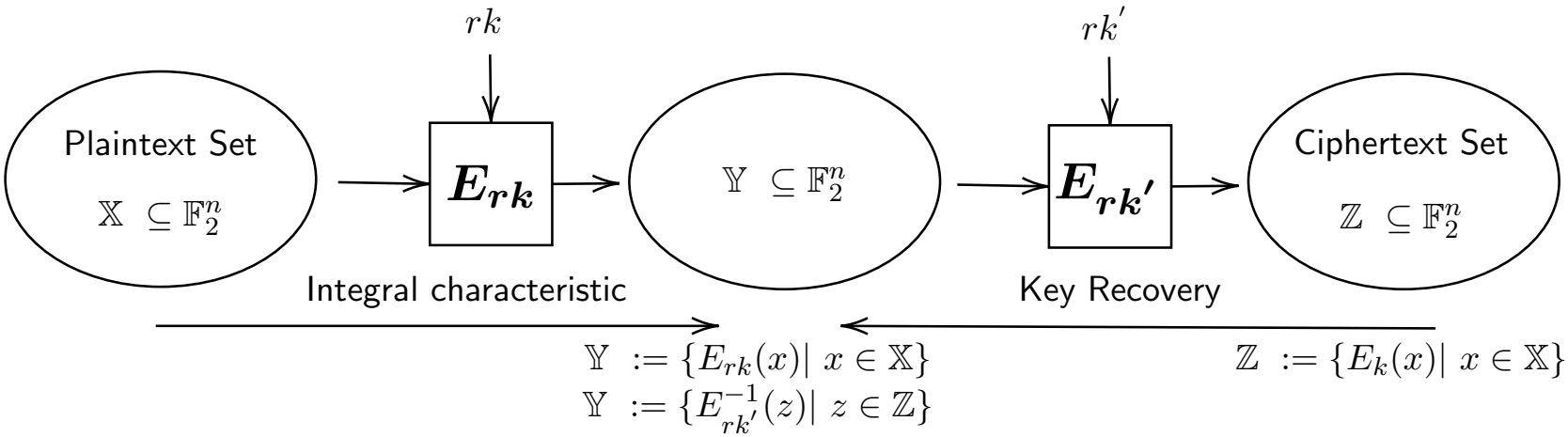
Outline

Integral Cryptanalysis



- E_{rk} has integral characteristic, if $\bigoplus_{x \in \mathbb{X}} E_{rk}(x) = 0$, For all possible rk.

Integral Cryptanalysis



- $\boldsymbol{E_{rk}}$ has integral characteristic, if $\bigoplus_{x \in \mathbb{X}} E_{rk}(x) = 0$, For all possible rk.
- If $\bigoplus_{z \in \mathbb{Z}} E_{rk'}^{-1}(Z) \neq 0$, then the guessed rk' is incorrect.

How to find an integral characteristic?

- Propagation of integral property: ALL ($\mathcal{A}$), BALANCE ($\mathcal{B}$), CONSTANT ($\mathcal{C}$), UNKNOWN ($\mathcal{U}$).
- Exploit algebraic degree.
- Division property.
  - Introduced by Yosuke Todo at Eurocrypt 2015.
  - *bit-based division property*, Todo and Morii - FSE 2016.

Bit-based Division Property [Todo and Morii, FSE 2016]

- bit-product function: $\pi_{\boldsymbol{u}}(\boldsymbol{x}) = \prod_{i=1}^{n} x[i]^{u[i]}$, where $x[i]$, $u[i]$ are the i-th bits of $\boldsymbol{x}$ and $\boldsymbol{u}$ respectively.
- The parity $= \bigoplus_{\boldsymbol{x} \in \mathbb{X}} \pi_{\boldsymbol{u}}(\boldsymbol{x})$.

Bit-based Division Property [Todo and Morii, FSE 2016]

- bit-product function: $\pi_{\boldsymbol{u}}(\boldsymbol{x}) = \prod_{i=1}^{n} x[i]^{u[i]}$, where $x[i]$, $u[i]$ are the i-th bits of $\boldsymbol{x}$ and $\boldsymbol{u}$ respectively.
- The parity $= \bigoplus_{\boldsymbol{x} \in \mathbb{X}} \pi_{\boldsymbol{u}}(\boldsymbol{x})$.

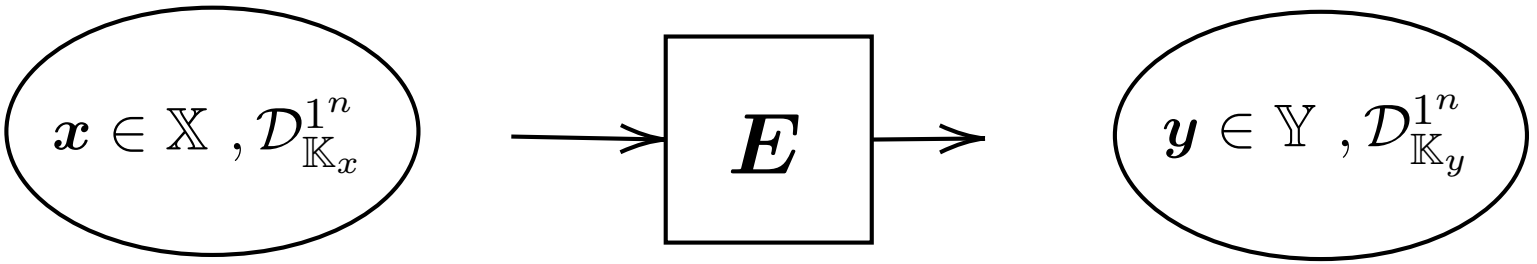
Definition (Bit-based Division Property)

Let $\mathbb{X}$ be a multiset whose elements take a value of $\mathbb{F}_2^n$. When the multiset $\mathbb{X}$ has the division property $\mathcal{D}_{\mathbb{K}}^{1^n}$,where $\mathbb{K}$ denotes a set of n-dimensional vectors whose i-th element takes 0 or 1, it fulfills the following conditions:

$$\bigoplus_{\boldsymbol{x} \in \mathbb{X}} \pi_{\boldsymbol{u}}(\boldsymbol{x}) = \begin{cases} unknown & if\ there\ is\ exist\ \boldsymbol{k} \in \mathbb{K}\ s.t.\ \boldsymbol{u} \succeq \boldsymbol{k}, \\ 0 & otherwise. \end{cases}$$

where $\boldsymbol{u} \succeq \boldsymbol{k}$ if $u[i] \geq k[i]\ \forall i$.

Integral Characteristic and Bit-based Division Property



Let $n = 4$ and $\boldsymbol{y} = (y_0, y_1, y_2, y_3)$,
To check if the bit y_3 is balanced over $\mathbb{Y}$:

$$\bigoplus_{\boldsymbol{y} \in \mathbb{Y}} y_3 = \bigoplus_{\boldsymbol{y} \in \mathbb{Y}} \pi_{\boldsymbol{v}}(\boldsymbol{y})|_{\boldsymbol{v}=(0,0,0,1)} \stackrel{?}{=} 0$$

From the definition of the bit-based division property,
the bit y_3 is balanced iff $\boldsymbol{v} = (0, 0, 0, 1) \notin \mathbb{K}_y$

Bit-based division property practical problem

- For n-bit cipher, the complexity of utilizing bit-based division property is roughly equal to 2^n.
- For $n > 32$, propagation includes large number of vectors in $\mathbb{K}$ and it will be infeasible to handle.
- Solved by Xiang *et al.* at Asiacrypt 2016 by defining a new notation called *Division Trail*.
  With the *Division Trail*, it becomes easy to model the bit-based division property propagation using Mixed Integer Linear Programming (MILP)

Propagation rules for COPY, XOR, AND

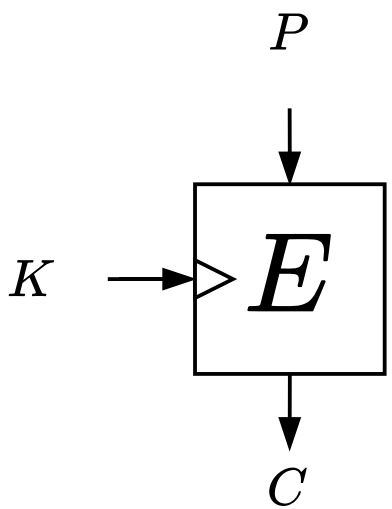
Operation	MILP Model
$(a) \xrightarrow{COPY} (b_1, b_2)$	$a - b_1 - b_2 = 0$
$(a_1, a_2) \xrightarrow{XOR} (b)$	$a_1 + a_2 - b = 0$
$(a_1, a_2) \xrightarrow{AND} (b)$	$\begin{cases} b - a_1 \geq 0, \\ b - a_2 \geq 0 \end{cases}$

MILP Model for S-box

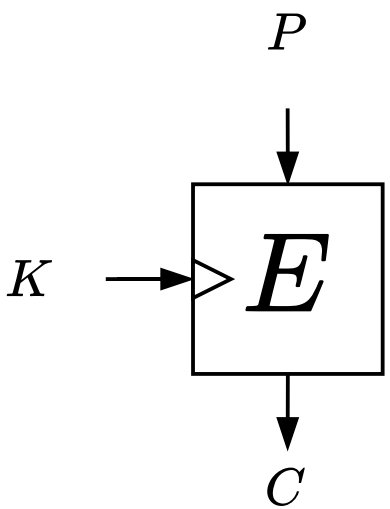
1. Represent the S-Box using its algebraic normal form.
2. Represent the division trail though an n-bit S-box as a set of $2n$-dimensional binary vectors.
3. Get the H-Representation as a set of linear inequalities that describe these vectors.
4. Use this set of inequalities as MILP constraints to present the division trail though the S-box.

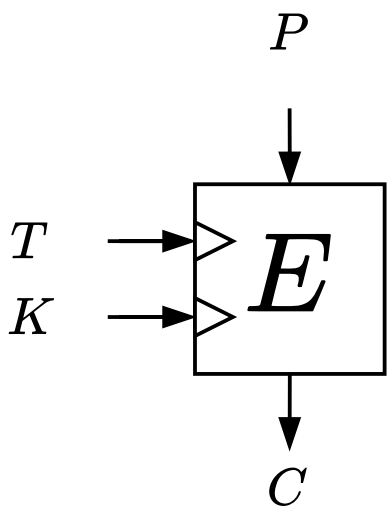
Outline

Block cipher - Tweakable



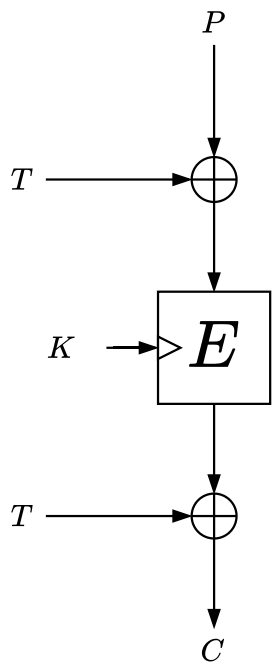
Block cipher - Tweakable





Block cipher - Tweakable

Mode of Operation



XEX Mode of operation

Dedicated Structure

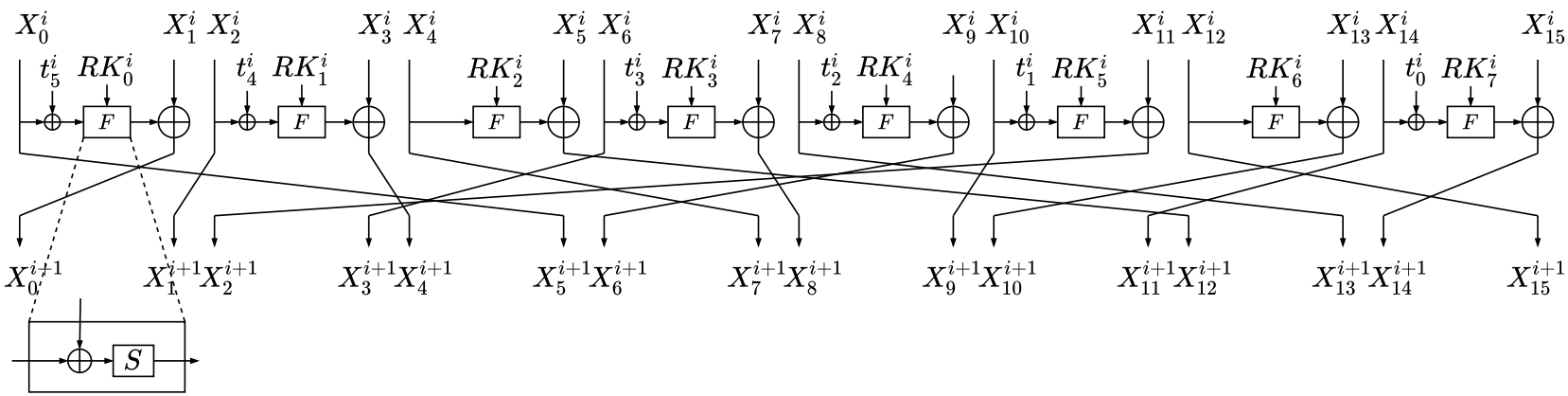
- SKINNY
- QARMA
- CRAFT
- T-TWINE

T-TWINE block cipher

1. The first lightweight dedicated TBC that is built on Generalized Feistel Structure (GFS).
2. An extension of the conventional block cipher TWINE.
3. Two variants namely, T-TWINE-80 and T-TWINE-128.
4. Block size of 64 bits, a tweak of 64 bits, and a variable key length of 80 and 128 bits.

T-TWINE Specifications

Data Processing



T-TWINE Specifications

Key Scheduling Function

Algorithm 1: Key Schedule of T-TWINE-80

Data: The 80-bit master key K

Result: The round keys $RK = RK^1||RK^2||\cdots||RK^{36}$

$k_0||k_1||\cdots||k_{19} \leftarrow K$;

for $i \leftarrow 1$ *to* 35 **do**

$RK^i \leftarrow k_1||k_3||k_4||k_6||k_{13}||k_{14}||k_{15}||k_{16}$;

$k_1 \leftarrow k_1 \oplus S(k_0)$;

$k_4 \leftarrow k_4 \oplus S(k_{16})$;

$k_7 \leftarrow k_7 \oplus (0||CON_H^i)$;

$k_{19} \leftarrow k_{19} \oplus (0||CON_L^i)$;

$k_0||\cdots||k_3 \leftarrow Rot4(k_0||\cdots||k_3)$;

$k_0||\cdots||k_{19} \leftarrow Rot16(k_0||\cdots||k_{19})$;

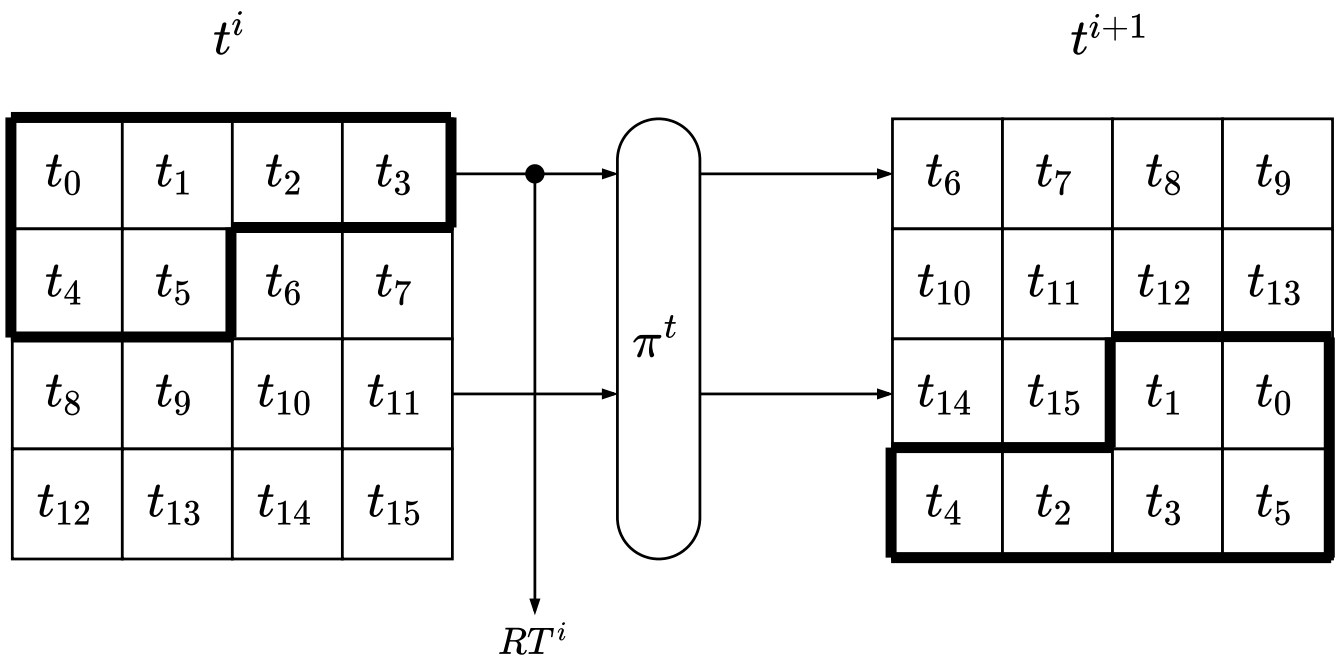
end

$RK^{36} \leftarrow k_1||k_3||k_4||k_6||k_{13}||k_{14}||k_{15}||k_{16}$;

$RK \leftarrow RK^1||RK^2||\cdots||RK^{36}$;

T-TWINE Specifications

Tweak Schduleing Function



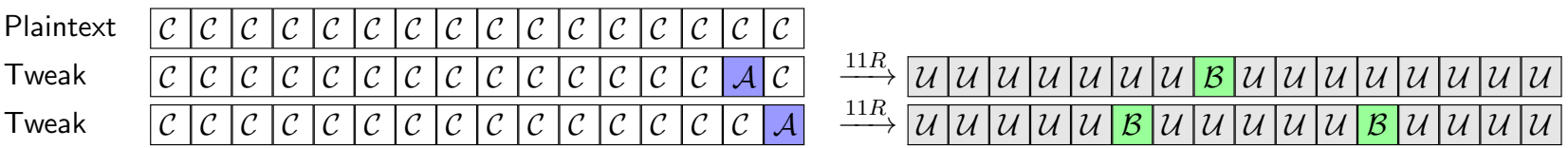
Outline

Integral distinguishing attack

1. TWINE has 16-round integral distinguisher.
2. Three attack settings: chosen tweak, chosen tweak-plaintext, and chosen tweak-ciphertext.
3. We use the following notation to present the status of each nibble of the tweak, plaintext, and ciphertext
   - $\mathcal{C}$ each bit of the nibble is fixed to constant.
   - $\mathcal{A}$ all bits of the nibble are active.
   - $\tilde{\mathcal{A}}$ all bits of the nibble are active except one bit is constant.
   - $\mathcal{B}$ each bit of the nibble is balanced.
   - $\mathcal{U}$ a nibble with unknown status.

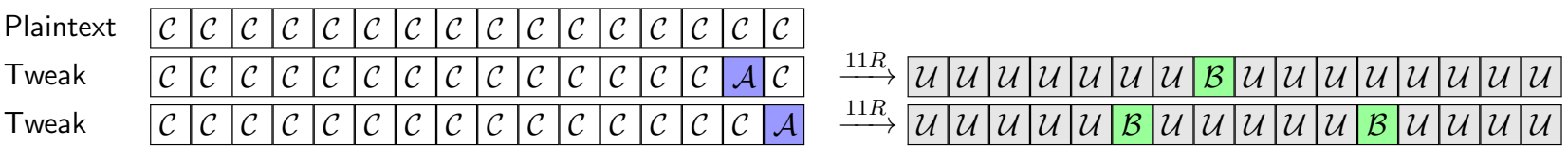
Chosen Tweak Setting

- The 11-round distinguisher

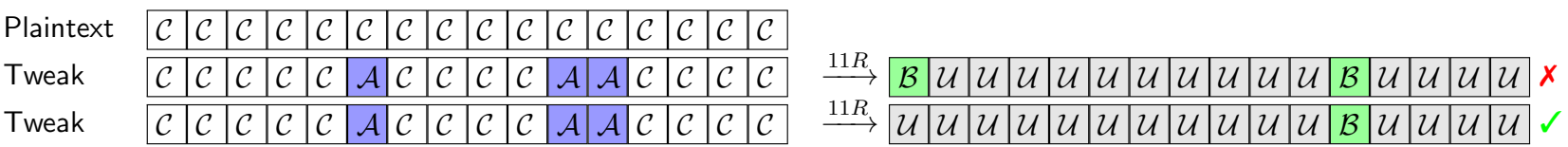


Chosen Tweak Setting

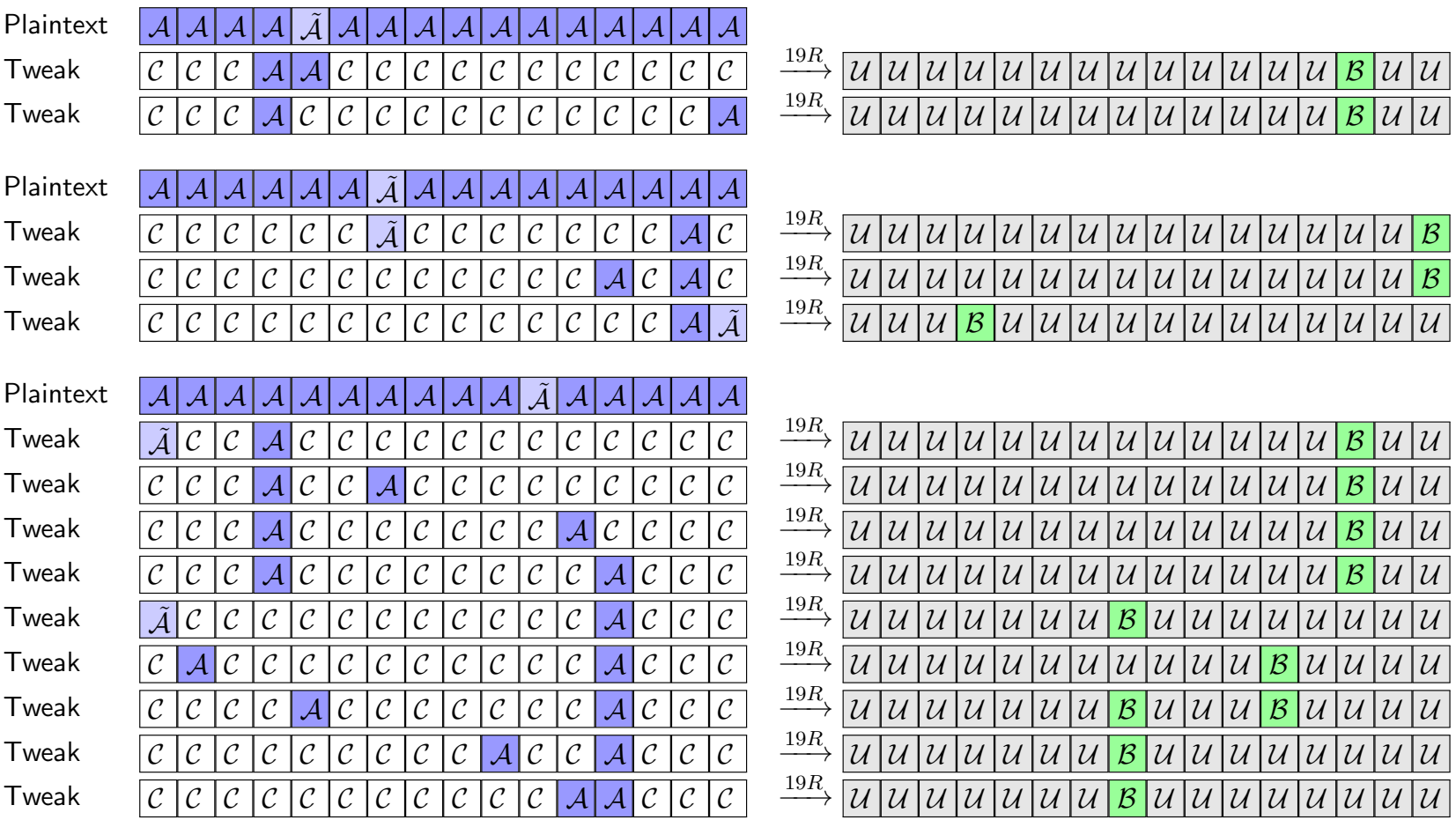
- The 11-round distinguisher



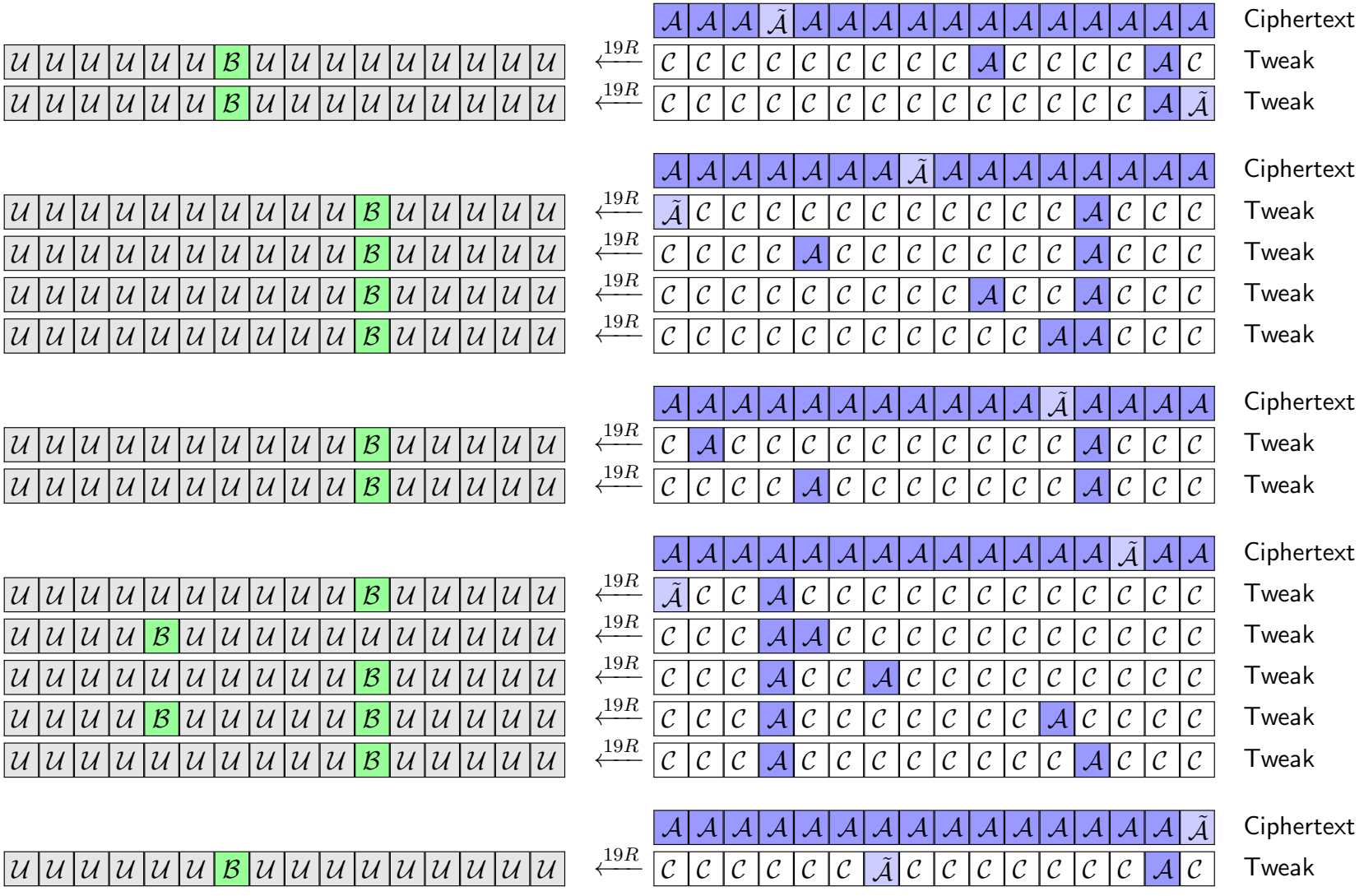
- The 11-round distinguisher by the designers



Chosen Tweak-Plaintext Setting



Chosen Tweak-Ciphertext Setting

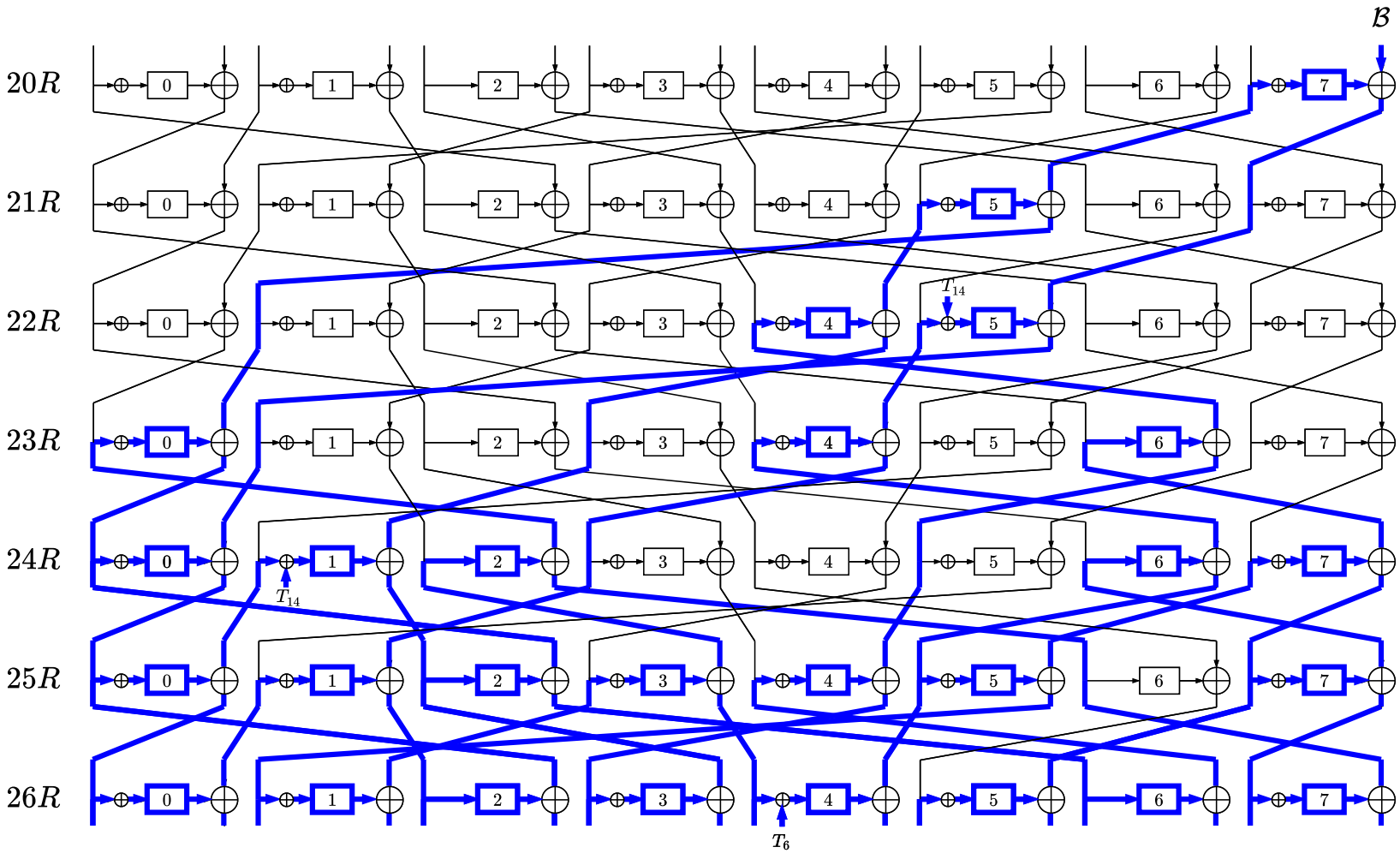


Outline

Integral Distinguisher

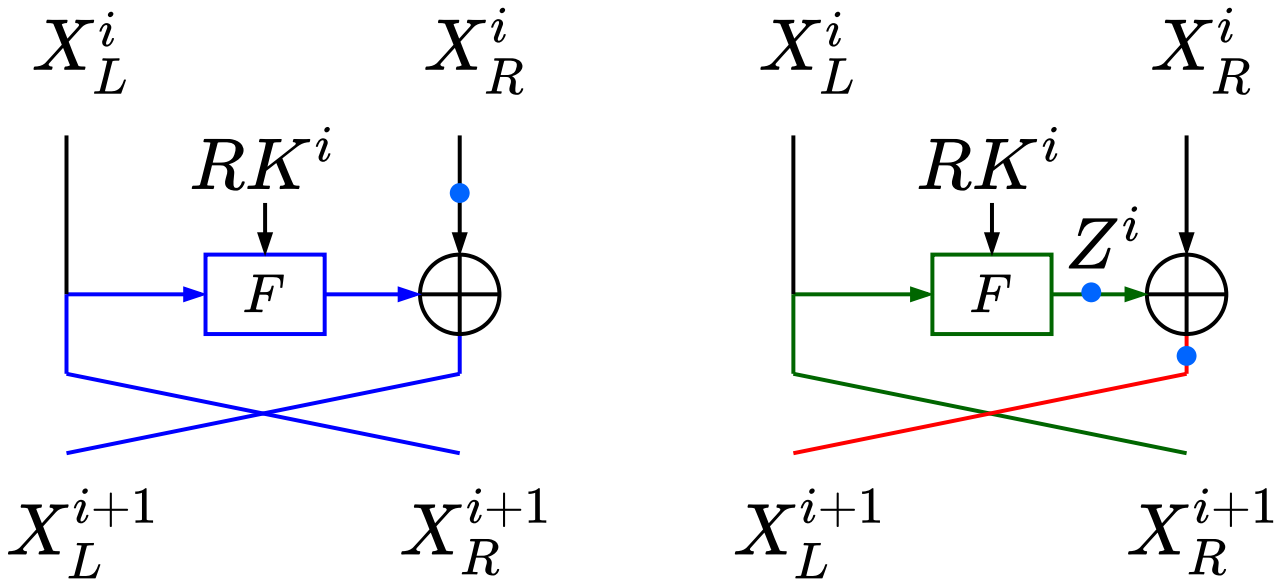
$$\begin{aligned}
\text{Plaintext} &: (\mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{A}_3, \mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{A}) \\
\text{Tweak} &: (\mathcal{C}, \mathcal{C}, \mathcal{C}, \mathcal{C}, \mathcal{C}, \mathcal{C}, \mathcal{A}_{1,3}, \mathcal{C}, \mathcal{C}, \mathcal{C}, \mathcal{C}, \mathcal{C}, \mathcal{C}, \mathcal{C}, \mathcal{A}, \mathcal{C}) \\
&\downarrow 19R \\
&(\mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{U}, \mathcal{B})
\end{aligned}$$

Analysis Rounds



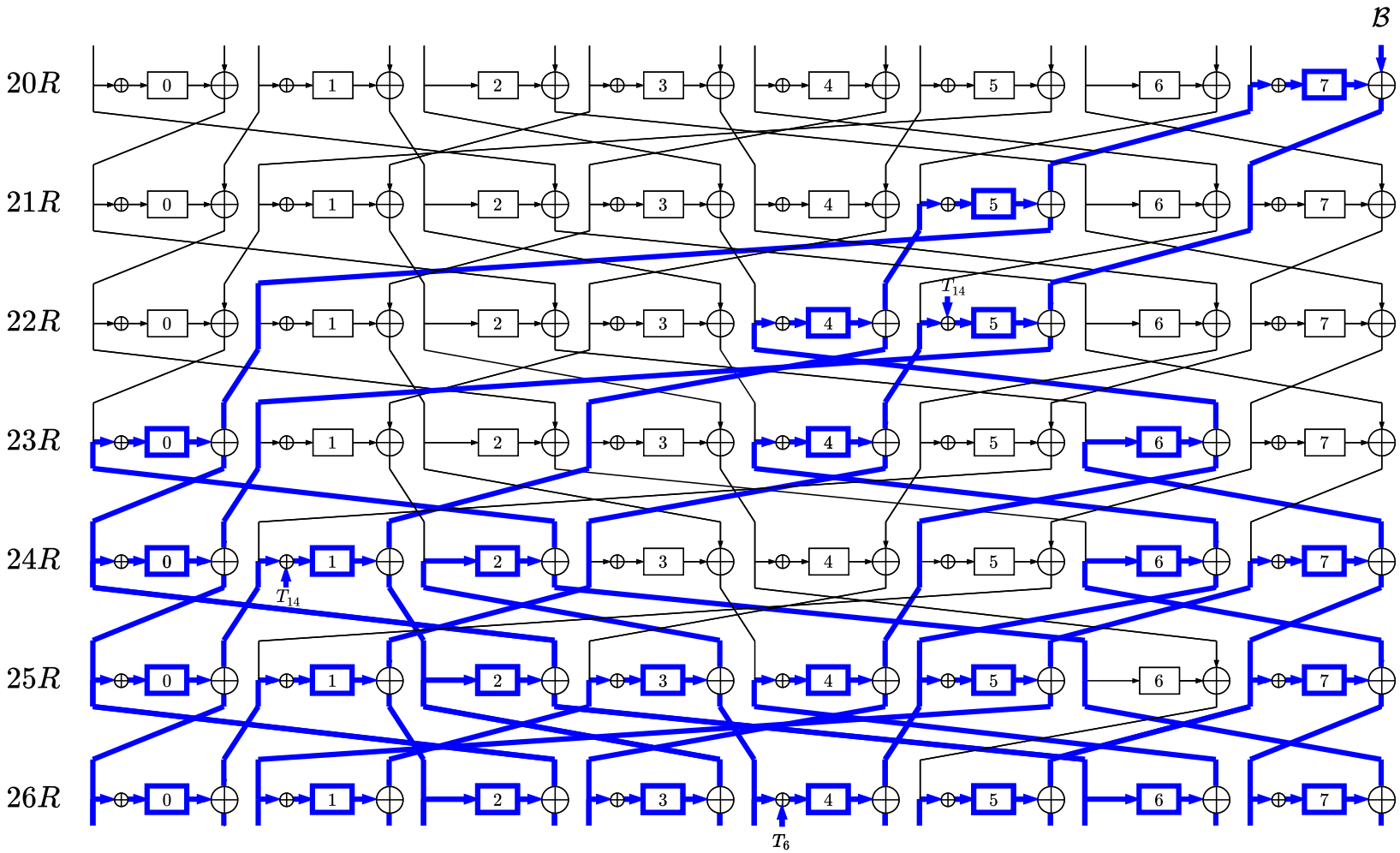
table

Meet-in-the-Middle Technique



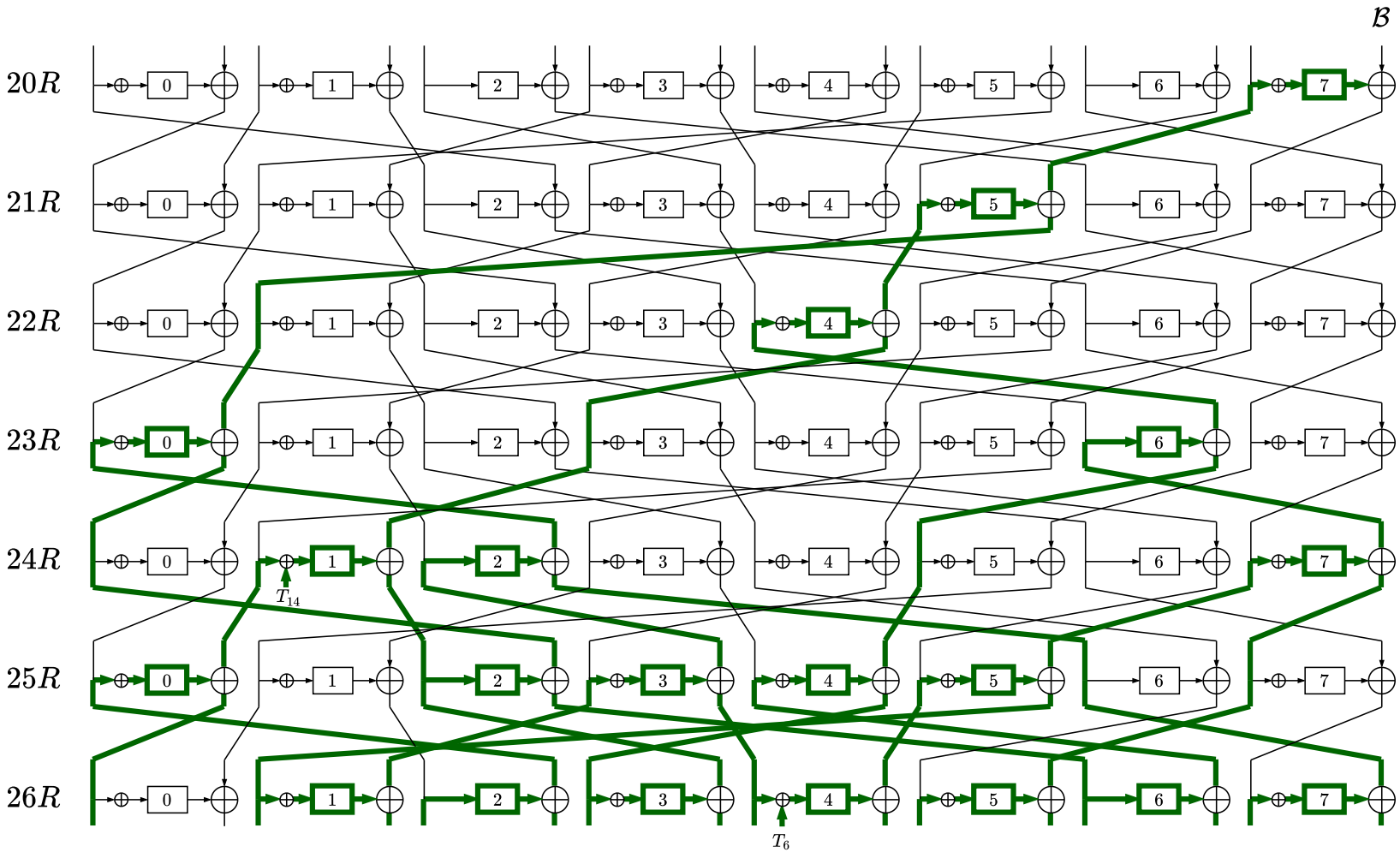
$$\bigoplus X_R^i = 0 \Longrightarrow \bigoplus Z^i = \bigoplus X_L^{i+1}$$

Analysis Rounds



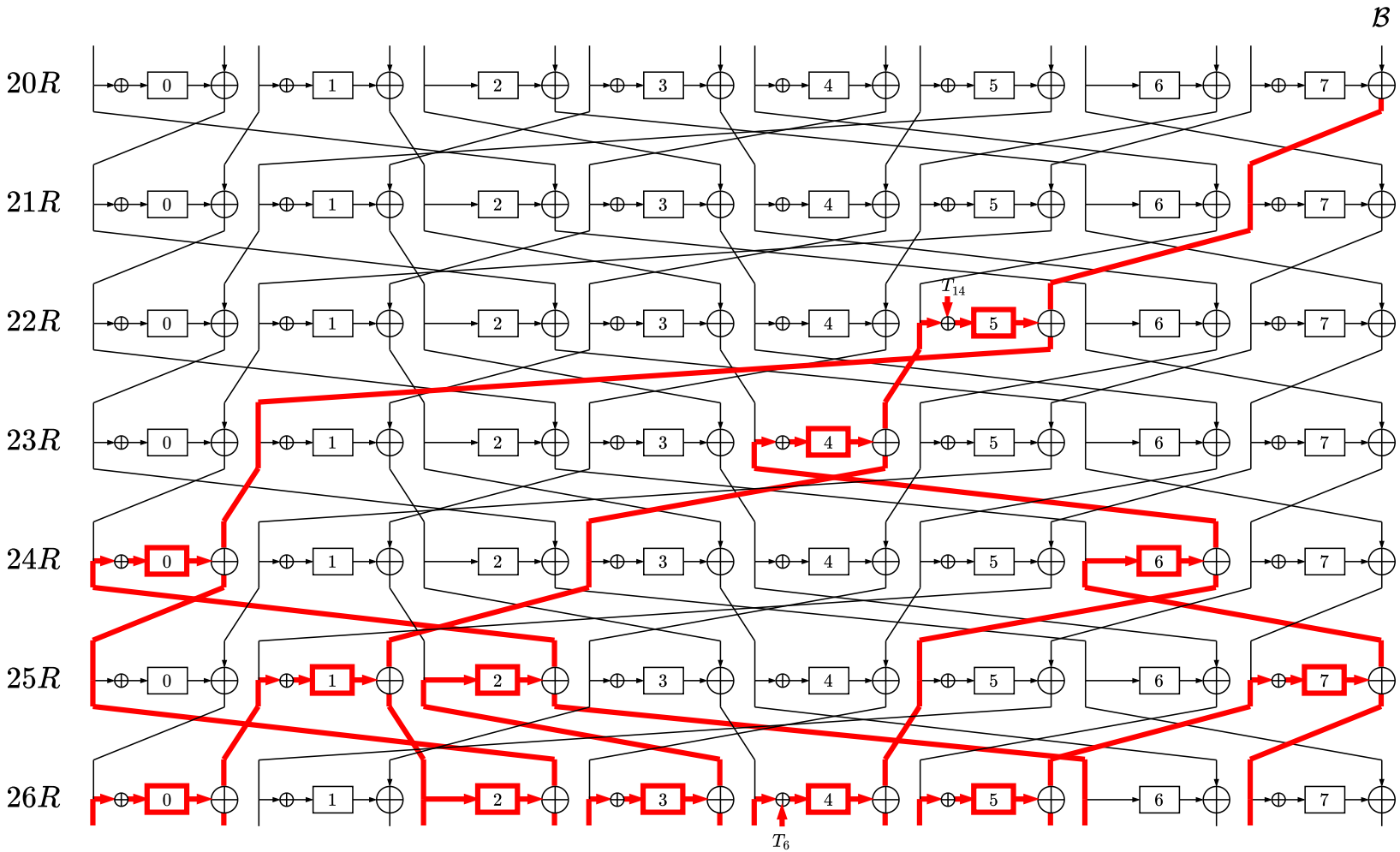
table

Path to Z_7^{20}



table

Path to X_{14}^{21}



table

Attack Procedure against 26 rounds of T-TWINE-80

1. Initialize two empty hash tables H_Z and H_X with 2^{56} and 2^{40} entries to store the values of $\bigoplus Z_7^{20}$ and $\bigoplus X_{14}^{21}$, respectively, indexed by the round keys used during the computations.

Attack Procedure against 26 rounds of T-TWINE-80

1. Initialize two empty hash tables H_Z and H_X with 2^{56} and 2^{40} entries to store the values of $\bigoplus Z_7^{20}$ and $\bigoplus X_{14}^{21}$, respectively, indexed by the round keys used during the computations.
2. For each 56-bit round-key, obtain $\bigoplus Z_7^{20}$ and store it in H_Z.

Attack Procedure against 26 rounds of T-TWINE-80

1. Initialize two empty hash tables H_Z and H_X with 2^{56} and 2^{40} entries to store the values of $\bigoplus Z_7^{20}$ and $\bigoplus X_{14}^{21}$, respectively, indexed by the round keys used during the computations.
2. For each 56-bit round-key, obtain $\bigoplus Z_7^{20}$ and store it in H_Z.
3. For each 40-bit round-key, obtain $\bigoplus X_{14}^{21}$ and store it in H_X.

Attack Procedure against 26 rounds of T-TWINE-80

1. Initialize two empty hash tables H_Z and H_X with 2^{56} and 2^{40} entries to store the values of $\bigoplus Z_7^{20}$ and $\bigoplus X_{14}^{21}$, respectively, indexed by the round keys used during the computations.
2. For each 56-bit round-key, obtain $\bigoplus Z_7^{20}$ and store it in H_Z.
3. For each 40-bit round-key, obtain $\bigoplus X_{14}^{21}$ and store it in H_X.
4. Consider a 76-bit key as a candidate if the two entries in the hash tables indexed by its two parts are equal.

Attack Procedure against 26 rounds of T-TWINE-80

1. Initialize two empty hash tables H_Z and H_X with 2^{56} and 2^{40} entries to store the values of $\bigoplus Z_7^{20}$ and $\bigoplus X_{14}^{21}$, respectively, indexed by the round keys used during the computations.
2. For each 56-bit round-key, obtain $\bigoplus Z_7^{20}$ and store it in H_Z.
3. For each 40-bit round-key, obtain $\bigoplus X_{14}^{21}$ and store it in H_X.
4. Consider a 76-bit key as a candidate if the two entries in the hash tables indexed by its two parts are equal.
5. For each 76-bit candidate, drive 2^4 80-bit candidates of the master key.

Attack Procedure against 26 rounds of T-TWINE-80

1. Initialize two empty hash tables H_Z and H_X with 2^{56} and 2^{40} entries to store the values of $\bigoplus Z_7^{20}$ and $\bigoplus X_{14}^{21}$, respectively, indexed by the round keys used during the computations.
2. For each 56-bit round-key, obtain $\bigoplus Z_7^{20}$ and store it in H_Z.
3. For each 40-bit round-key, obtain $\bigoplus X_{14}^{21}$ and store it in H_X.
4. Consider a 76-bit key as a candidate if the two entries in the hash tables indexed by its two parts are equal.
5. For each 76-bit candidate, drive 2^4 80-bit candidates of the master key.
6. Check each 80-bit candidate using 2 pairs of plaintext/ciphertext.

The summary of the procedure to obtain $\bigoplus Z_7^{20}$

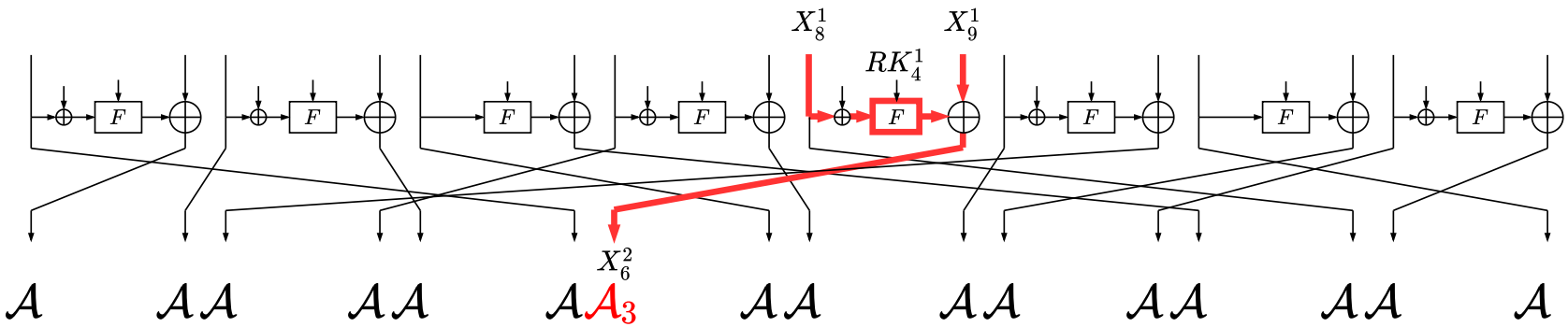
Step	Key	Size	The State ($\mathcal{S}_i$)	Complexity
0	-	2^{66}	$X_0^{27}, X_2^{27}, X_3^{27}, \boxed{X_4^{27}}, \boxed{X_5^{27}}, X_6^{27}, X_7^{27}, X_8^{27}, X_9^{27}, X_{10}^{27}, X_{11}^{27}, X_{12}^{27}, X_{13}^{27}, X_{14}^{27}, X_{15}^{27}, T_6, T_{14}$	$2^{66}\,MA$
1	RK_2^{26}	2^{62}	$X_0^{27}, X_2^{27}, X_3^{27}, X_5^{26}, X_6^{27}, X_7^{27}, X_8^{27}, X_9^{27}, \boxed{X_{10}^{27}}, \boxed{X_{11}^{27}}, X_{12}^{27}, X_{13}^{27}, X_{14}^{27}, X_{15}^{27}, T_6, T_{14}$	$2^4 \times 2^{66} = 2^{70}$
2	RK_5^{26}	2^{58}	$X_0^{27}, X_2^{27}, X_3^{27}, X_5^{26}, X_6^{27}, X_7^{27}, X_8^{27}, X_9^{27}, X_{11}^{26}, X_{12}^{27}, X_{13}^{27}, \boxed{X_{14}^{27}}, \boxed{X_{15}^{27}}, T_6, T_{14}$	$2^8 \times 2^{62} = 2^{70}$
3	RK_7^{26}	2^{54}	$\boxed{X_0^{27}}, X_2^{27}, X_3^{27}, \boxed{X_5^{26}}, X_6^{27}, X_7^{27}, X_8^{27}, X_9^{27}, X_{11}^{26}, X_{12}^{27}, X_{13}^{27}, X_{15}^{26}, T_6, T_{14}$	$2^{12} \times 2^{58} = 2^{70}$
4	RK_0^{25}	2^{50}	$X_1^{25}, X_2^{27}, X_3^{27}, \boxed{X_6^{27}}, \boxed{X_7^{27}}, X_8^{27}, X_9^{27}, X_{11}^{26}, X_{12}^{27}, X_{13}^{27}, X_{15}^{26}, T_6, T_{14}$	$2^{16} \times 2^{54} = 2^{70}$
5	RK_3^{26}	2^{50}	$\boxed{X_1^{25}}, X_2^{27}, X_3^{27}, X_6^{26}, \boxed{X_7^{26}}, X_8^{27}, X_9^{27}, X_{11}^{26}, X_{12}^{27}, X_{13}^{27}, X_{15}^{26}, T_6, \boxed{T_{14}}$	$2^{20} \times 2^{50} = 2^{70}$
6	RK_1^{24}	2^{46}	$X_3^{24}, X_2^{27}, X_3^{27}, X_6^{26}, X_7^{26}, \boxed{X_8^{27}}, \boxed{X_9^{27}}, X_{11}^{26}, X_{12}^{27}, X_{13}^{27}, X_{15}^{26}, \boxed{T_6}$	$2^{20} \times 2^{50} = 2^{70}$
7	RK_4^{26}	2^{44}	$X_3^{24}, \boxed{X_2^{27}}, \boxed{X_3^{27}}, X_6^{26}, X_7^{26}, X_8^{26}, X_9^{26}, X_{11}^{26}, X_{12}^{27}, X_{13}^{27}, X_{15}^{26}$	$2^{24} \times 2^{46} = 2^{70}$
8	RK_1^{26}	2^{44}	$X_3^{24}, X_2^{26}, \boxed{X_3^{26}}, X_6^{26}, X_7^{26}, \boxed{X_8^{26}}, X_9^{26}, X_{11}^{26}, X_{12}^{27}, X_{13}^{27}, X_{15}^{26}$	$2^{28} \times 2^{44} = 2^{72}$
9	RK_3^{25}	2^{40}	$X_3^{24}, \boxed{X_2^{26}}, X_6^{26}, X_7^{26}, X_7^{25}, \boxed{X_9^{26}}, X_{11}^{26}, X_{12}^{27}, X_{13}^{27}, X_{15}^{26}$	$2^{32} \times 2^{44} = 2^{76}$
10	RK_5^{25}	2^{36}	$X_3^{24}, X_6^{26}, X_7^{26}, X_7^{25}, \boxed{X_{11}^{25}}, \boxed{X_{11}^{26}}, X_{12}^{27}, X_{13}^{27}, X_{15}^{26}$	$2^{36} \times 2^{40} = 2^{76}$
11	RK_7^{24}	2^{32}	$X_3^{24}, X_6^{26}, X_7^{26}, \boxed{X_7^{25}}, X_{12}^{27}, X_{13}^{27}, X_{15}^{24}, \boxed{X_{15}^{26}}$	$2^{40} \times 2^{36} = 2^{76}$
12	RK_2^{24}	2^{28}	$X_3^{24}, X_5^{24}, X_6^{26}, X_7^{26}, \boxed{X_{12}^{27}}, \boxed{X_{13}^{27}}, X_{15}^{24}$	$2^{40} \times 2^{32} = 2^{72}$
13	RK_6^{26}	2^{28}	$X_3^{24}, X_5^{24}, \boxed{X_6^{26}}, X_7^{26}, X_{12}^{26}, \boxed{X_{13}^{26}}, X_{15}^{24}$	$2^{44} \times 2^{28} = 2^{72}$
14	RK_4^{24}	2^{24}	$X_3^{24}, X_5^{24}, X_7^{26}, \boxed{X_9^{25}}, X_{12}^{26}, \boxed{X_{15}^{24}}$	$2^{48} \times 2^{28} = 2^{76}$
15	RK_6^{23}	2^{20}	$\boxed{X_3^{24}}, X_5^{24}, X_7^{26}, X_{12}^{26}, \boxed{X_{13}^{23}}$	$2^{48} \times 2^{24} = 2^{72}$
16	RK_4^{22}	2^{16}	$X_5^{24}, \boxed{X_7^{26}}, X_9^{22}, \boxed{X_{12}^{26}}$	$2^{48} \times 2^{20} = 2^{68}$
17	RK_2^{25}	2^{12}	$\boxed{X_5^{24}}, \boxed{X_5^{25}}, X_9^{22}$	$2^{52} \times 2^{16} = 2^{68}$
18	RK_0^{23}	2^8	$\boxed{X_1^{23}}, \boxed{X_9^{22}}$	$2^{56} \times 2^{12} = 2^{68}$
19	RK_5^{21}	2^4	$\boxed{X_{11}^{21}}$	$2^{56} \times 2^8 = 2^{64}$
20	RK_7^{20}	1	$\bigoplus Z_7^{20} = \bigoplus S(X_{11}^{21} \oplus RK_7^{20})$	$2^{56} \times 2^4 = 2^{60}$

figure

Attack Complexity

	Data	Time Complexity	Memory
1	2^{69}	$2^{69} + 1 \times \frac{2^{78.13}+2^{59.91}}{8\times 26} + \frac{2^{76}}{8\times 26} + \frac{145\times 2^{72}}{8\times 26} + 2^{76} + 2^{12} \approx 2^{76.11}$	$2^{66.04}$
2	2^{70}	$2^{70} + 2 \times \frac{2^{78.13}+2^{59.91}}{8\times 26} + \frac{2^{76}+2^{72}}{8\times 26} + \frac{145\times 2^{68}}{8\times 26} + 2^{72} + 2^{8} \approx 2^{73.03}$	$2^{67.04}$
3	$2^{70.58}$	$2^{70.58} + 3 \times \frac{2^{78.13}+2^{59.91}}{8\times 26} + \frac{2^{76}+2^{72}+2^{68}}{8\times 26} + \frac{145\times 2^{64}}{8\times 26} + 2^{68} + 2^{4} \approx 2^{72.62}$	$2^{67.62}$
4	2^{71}	$2^{71} + 4 \times \frac{2^{78.13}+2^{59.91}}{8\times 26} + \frac{2^{76}+2^{72}+2^{68}+2^{64}}{8\times 26} + \frac{145\times 2^{60}}{8\times 26} + 2^{64} \approx 2^{72.95}$	$2^{68.04}$

Attack one more round



$$X_6^2[3] = X_9^1[3] \oplus S(X_8^1 \oplus RK_4^1 \oplus RT_2^1)[3]$$

Attack results on T-TWINE

	Attack	#Rounds	Data	Time	Memory	Reference
T-TWINE-80	Imp. diff.	25	$2^{65.5}$ CTP	$2^{70.86}$	2^{66}	Tolba *et al.* 2020
	Integral	26	$2^{70.58}$ CTP	$2^{72.62}$	$2^{67.62}$	This work
		27	$2^{70.95}$ CTP	$2^{75.79}$	$2^{71.08}$	This work
T-TWINE-128	Imp. diff.	27	2^{64} CTP	$2^{120.83}$	2^{118}	Tolba *et al.* 2020
	Integral	27	$2^{71.58}$ CTP	$2^{109.54}$	$2^{90.58}$	This work
		28	$2^{72.27}$ CTP	$2^{113.38}$	$2^{94.32}$	This work

Outline

Conclusion

- We have studied the security of T-TWINE against the integral cryptanalysis.

Conclusion

- We have studied the security of T-TWINE against the integral cryptanalysis.
- We have showed that adding a tweak to the round function structure gives the attacker more room to target a large number of rounds in T-TWINE comparing to the conventional TWINE.

Conclusion

- We have studied the security of T-TWINE against the integral cryptanalysis.
- We have showed that adding a tweak to the round function structure gives the attacker more room to target a large number of rounds in T-TWINE comparing to the conventional TWINE.
- We were able to construct several integral distinguishers that cover 19 rounds of T-TWINE whereas the longest distinguisher covers only 16 rounds of TWINE.

Conclusion

- We have studied the security of T-TWINE against the integral cryptanalysis.
- We have showed that adding a tweak to the round function structure gives the attacker more room to target a large number of rounds in T-TWINE comparing to the conventional TWINE.
- We were able to construct several integral distinguishers that cover 19 rounds of T-TWINE whereas the longest distinguisher covers only 16 rounds of TWINE.
- Using one of these distinguishers, we launched key recovery attacks against 27 and 28 of T-TWINE-80 and T-TWINE-128, respectively. The presented attacks are the best-published attacks against both variants of T-TWINE.

Thank You

For Questions: m_elshei@encs.concordia.ca