Key Guessing Strategies for Linear Key-Schedule Algorithms in Rectangle Attacks

Xiaoyang Dong Lingyue Qin Siwei Sun Xiaoyun Wang

[1]Institute for Advanced Study, BNRist, Tsinghua University, Beijing, China
[2]School of Cryptology, University of Chinese Academy of Sciences, Beijing, China
[3]State Key Laboratory of Cryptology, Beijing, China
[4]Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan, China
[5]School of Cyber Science and Technology, Shandong University, Qingdao China

May 21, 2022

Outline

1. Background and Motivations
2. Key-Guessing Strategies in the Rectangle Attack
3. Automatic Model For SKINNY
4. Improved Rectangle Attacks on SKINNY
5. Conclusion and Further Disscussion

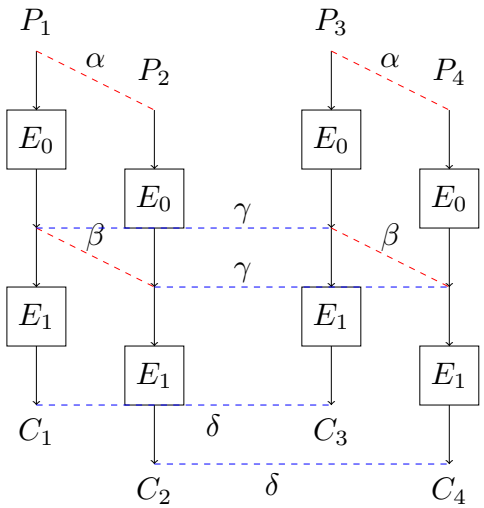
Outline

1. Background and Motivations
2. Key-Guessing Strategies in the Rectangle Attack
3. Automatic Model For SKINNY
4. Improved Rectangle Attacks on SKINNY
5. Conclusion and Further Disscussion

Boomerang and Retangle Attack

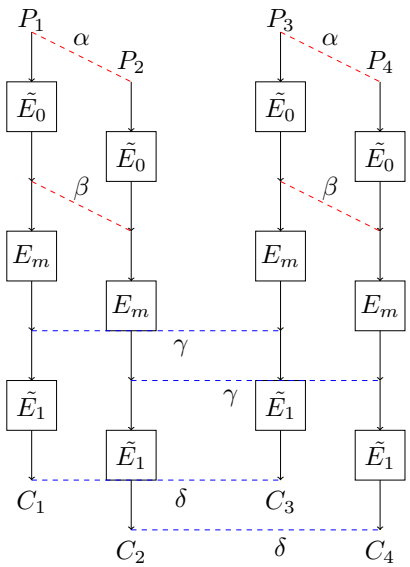
- Boomerang Attack proposed by Wagner [Wag99]
  - $E_d = \underbrace{E_1}_{\gamma \xrightarrow{q} \delta} \circ \underbrace{E_0}_{\alpha \xrightarrow{p} \beta}$
  - $Pr[E_d^{-1}(E_d(x) \oplus \delta) \oplus E_d^{-1}(E_d(x \oplus \alpha) \oplus \delta) = \alpha] = p^2 q^2$
- Adaptive chosen plaintext and ciphertext $\rightarrow$ chosen-plaintext attack:
  - amplified boomerang attack [KKS00]
  - rectangle attack [BDK01]
  - probability $2^{-n}\hat{p}^2\hat{q}^2$

$$\hat{p} = \sqrt{\sum_i Pr^2(\alpha \rightarrow \beta_i)}, \hat{q} = \sqrt{\sum_i Pr^2(\gamma_i \rightarrow \delta)}$$

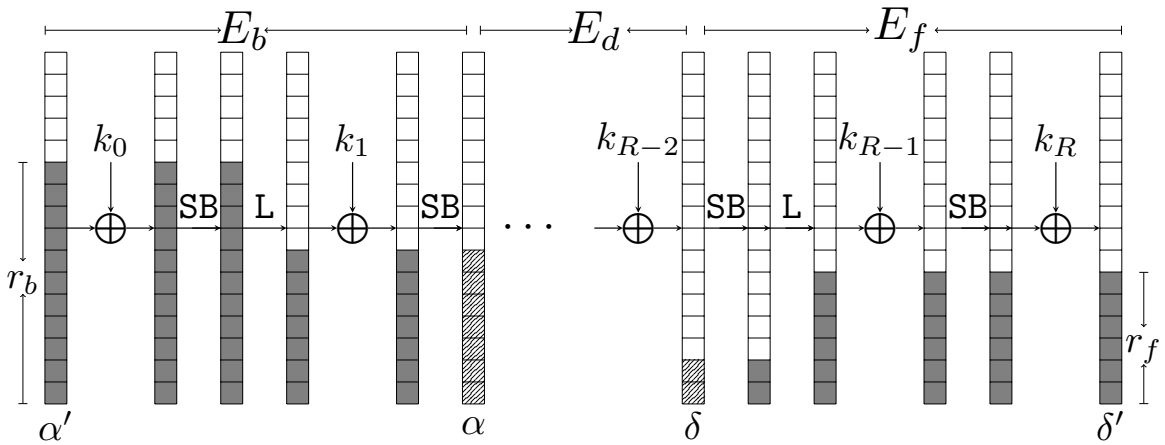


Boomerang and Retangle Attack

- Improvements based on the dependence between two differentials
  - Boomerang switch [Mur11, BK09, BDK05]
  - Sandwich attack [DKS14]
    - $E_d = \underbrace{\tilde{E}_1}_{\gamma \xrightarrow{\tilde{q}} \delta} \circ \underbrace{E_M}_{\beta \xrightarrow{t} \gamma} \circ \underbrace{\tilde{E}_0}_{\alpha \xrightarrow{\tilde{p}} \beta}$
    - probability $2^{-n}\tilde{p}^2\tilde{q}^2 t$
  - Boomerang connectivity Table [CHP+18]
  - BCT in multiple rounds [WP19, SQH19]



Generalized Key-Recovery Algorithms for Rectangle Attack



- `Attack I`: Biham-Dunkelman-Keller's Attack at EC01 [BDK01]
- `Attack II`: Biham-Dunkelman-Keller's Attack at FSE02 [BDK02]
- `Attack III`: Zhao *et al.*'s Related-Key Attack at ToSC19 and DCC20 [ZDJ19, ZDM+20]

Outline

1 Background and Motivations

2 Key-Guessing Strategies in the Rectangle Attack

3 Automatic Model For SKINNY

4 Improved Rectangle Attacks on SKINNY

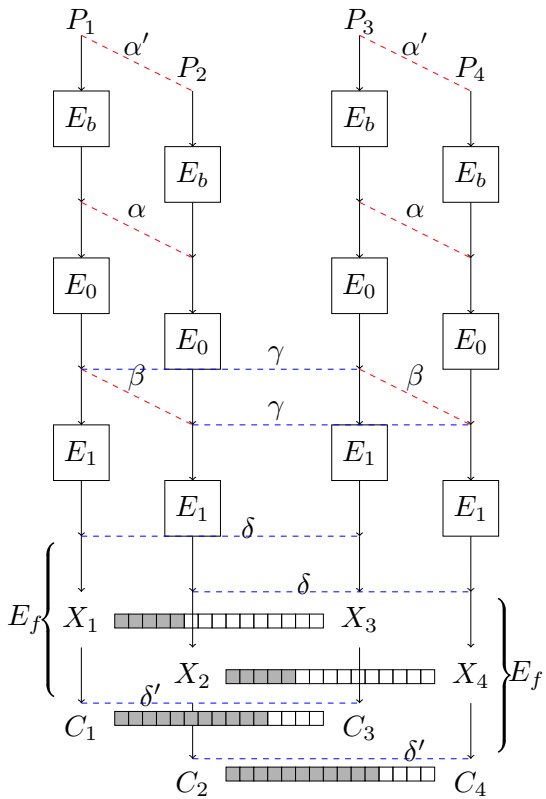
5 Conclusion and Further Disscussion

Related-key Rectangle Attack with Linear Key Schedule

- Invalid quartet: (P_1, P_2, P_3, P_4) meet (α', δ') but never suggest any key candidates. Guess k_{1b}, other k_{2b}, k_{3b}, k_{4b}, are fixed

$$\begin{cases} S(k_{1b} \oplus P_1) \oplus S(k_{2b} \oplus P_2) = \alpha, \\ S(k_{3b} \oplus P_3) \oplus S(k_{4b} \oplus P_4) = \alpha \end{cases}$$

- A tradeoff of the rectangle attack framework with linear key schedule
  - guess k_b in E_b and part of k_f in E_f
  - gain more inactive bits (or fixed differences) from the internal state

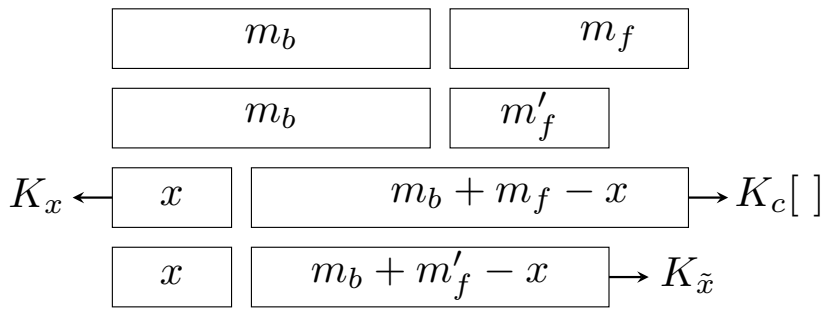


Algorithm 1 (Attack IV)

Construct y structures of 2^{rb} plaintexts each. For structure i ($1 \leq i \leq y$), query the 2^{rb} plaintexts by encryption under K_1, K_2, K_3 and K_4 and store them in $L_1[i]$, $L_2[i]$, $L_3[i]$ and $L_4[i]$
for *each of the* x*-bit key* K_x*, which is a part of* $(m_b + m'_f)$*-bit* K_1 **do**
  $K_c \leftarrow [\,]$
  for *each of* $(m_b + m'_f - x)$*-bit* $K_{\tilde{x}}$ *of* K_1 *involved in* E_b *and* E_f **do**
    for i *from* 1 *to* y **do**
      for $(P_1, C_1) \in L_1[i]$ **do**
        $P_2 = E^{-1}_{b[K_1 \oplus \Delta K]}(E_{b_{K_1}}(P_1) \oplus \alpha), S_1 \leftarrow (P_1, C_1, P_2, C_2)$
      end
      for $(P_3, C_3) \in L_3[i]$ **do**
        $P_4 = E^{-1}_{b[K_1 \oplus \Delta K \oplus \nabla K]}(E_{b[K_1 \oplus \nabla K]}(P_3) \oplus \alpha), S_2 \leftarrow (P_3, C_3, P_4, C_4)$
      end
    end
    for $(P_1, C_1, P_2, C_2) \in S_1$ **do**
      $X_1[1, \cdots, h_f] = E^{-1}_{f[K_1]}(C_1)$, $X_2[1, \cdots, h_f] = E^{-1}_{f[K_1 \oplus \Delta K]}(C_2)$
      $\tau = (X_1[1,\cdots,h_f], X_2[1,\cdots,h_f], C_1[1,\cdots,n - r_f], C_2[1,\cdots,n - r_f])$,
      $H[\tau] \leftarrow (P_1, C_1, P_2, C_2)$
    end
    for $(P_3, C_3, P_4, C_4) \in S_2$ **do**
      $X_3[1, \cdots, h_f] = E^{-1}_{f[K_1 \oplus \nabla K]}(C_3)$, $X_4[1, \cdots, h_f] = E^{-1}_{f[K_1 \oplus \Delta K \oplus \nabla K]}(C_4)$
      $\tau' = (X_3[1,\cdots,h_f], X_4[1,\cdots,h_f], C_3[1,\cdots,n - r_f], C_4[1,\cdots,n - r_f])$
      Access $H[\tau']$ to find (P_1, C_1, P_2, C_2) to generate quartet (C_1, C_2, C_3, C_4)
      for *each generated quartet* **do**
        Determine other $(m_f - m'_f)$-bit key k''_f, $K_c[K_{\tilde{x}} \| k''_f] \leftarrow K_c[K_{\tilde{x}} \| k''_f] + 1$
      end
    end
  end
  Select top $2^{m_b + m_f - x - h}$ hits in K_c. Guess remaining $k - (m_b + m_f)$-bit key to check the full key
end

Related-key Rectangle Attack with Linear Key Schedule

- The guessed keys in Algorithm 1



  - m_b-bit k_b and m_f-bit k_f in total
  - guess m_b-bit k_b and m'_f-bit k'_f before generating quartets
  - guess x-bit K_x before initializing the key counter $K_c[\ ]$

Complexity

- set $(y \cdot 2^{2r_b})^2 \cdot 2^{-2r_b} \cdot 2^{-n}\hat{p}^2\hat{q}^2 = s$, s is the expected number of right quartets
- Data complexity: $4y \cdot 2^{r_b} = \sqrt{s} \cdot 2^{n/2+2}/\hat{p}\hat{q}$
- Time complexity:
  - **Time I** : $T_1 = 2^{x+m_b+m'_f-x} \cdot y \cdot 2^{r_b} \cdot 2 = \sqrt{s} \cdot 2^{m_b+m'_f+n/2+1}/\hat{p}\hat{q}$
  - **Time II** : $T_2 = (s \cdot 2^{m_b+m'_f-n+2r_f-2h_f}/\hat{p}^2\hat{q}^2) \cdot \varepsilon$
  - **Time III** : $T_3 = 2^x \cdot 2^{m_b+m_f-x-h} \cdot 2^{k-(m_b+m_f)} = 2^{k-h}$
- Memory complexity of the key counters and data structures : $2^{m_b+m_f-x} + 4y \cdot 2^{r_b} = 2^{m_b+m_f-x} + \sqrt{s} \cdot 2^{n/2+2}/\hat{p}\hat{q}$
- Success Probability: $P_s = \Phi(\frac{\sqrt{sS_N}-\Phi^{-1}(1-2^{-h})}{\sqrt{S_N+1}})$ [Sel08]
  - $h < m_b + m_f - x$, $x \leq m_b + m'_f$

Outline

1 Background and Motivations

2 Key-Guessing Strategies in the Rectangle Attack

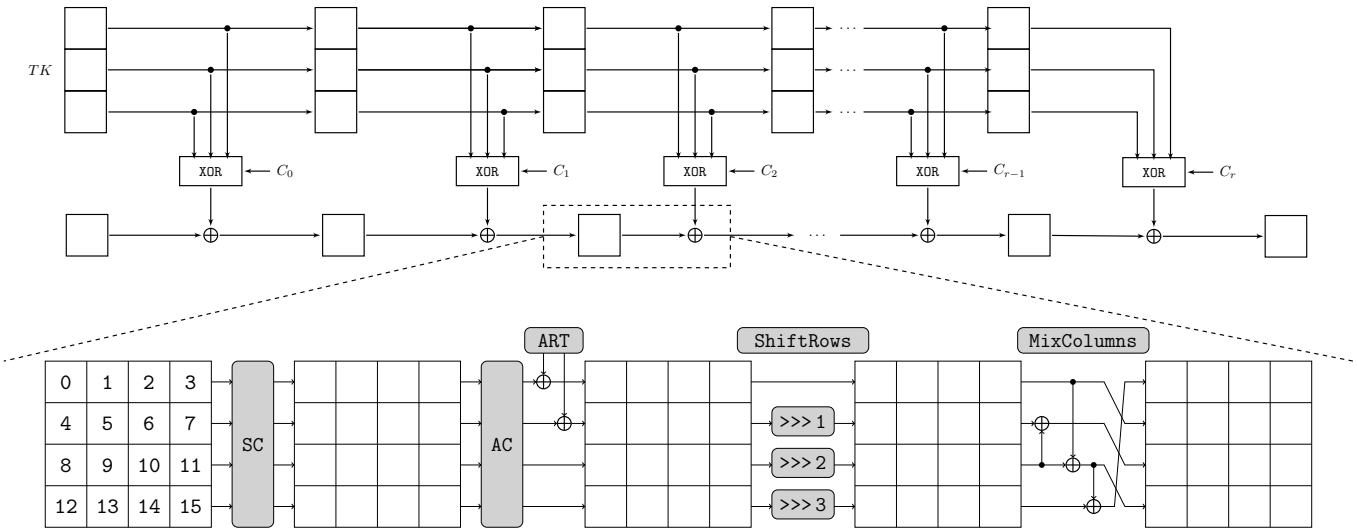
3 Automatic Model For SKINNY

4 Improved Rectangle Attacks on SKINNY

5 Conclusion and Further Disscussion

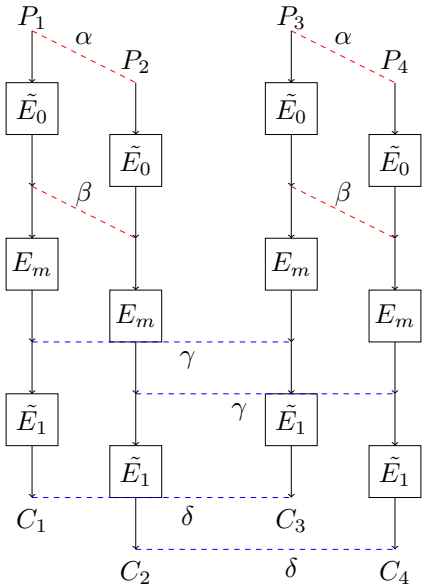
Introduction to SKINNY

- Proposed by Beierle et al. [BJK+16]
- SKINNY-n-t: block size $n \in \{64, 128\}$, tweakey size $t = n, 2n$ or $3n$



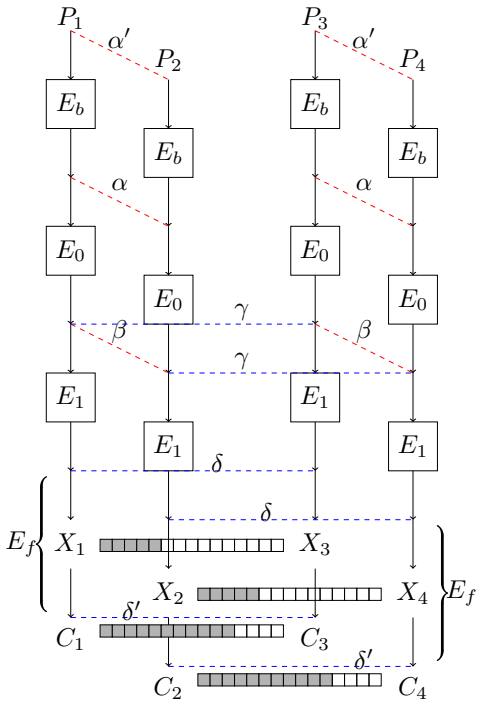
Previous Automatic Search Models for Boomerang Distinguishers on SKINNY

- MILP, Beierle et al. [BJK+16]
- MILP, Liu et al. [LGS17]
- MILP+SAT, Hadipour et al. [HBS21]
  - considering the switching effect in multiple rounds
- MILP+CP, Delaune et al. [DDV20]
  - automatically handling the middle rounds
- MILP+CP, Qin et al. [QDW+21]
  - search the entire $(N_b + N_d + N_f)$-round attack

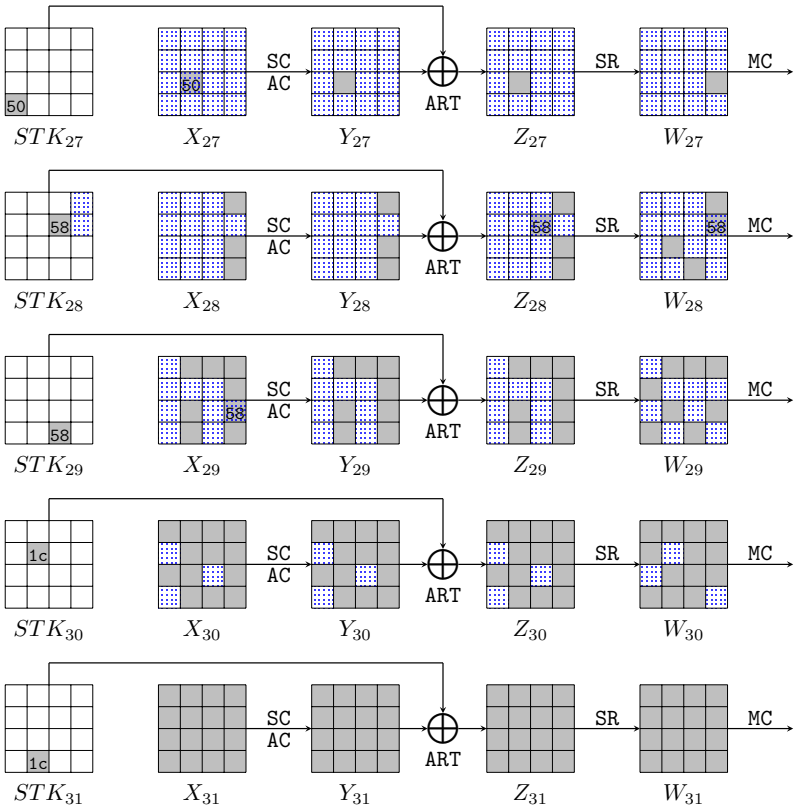


Our Model to Determine the Optimal Distinguisher

- search the entire $(N_b + N_d + N_f)$-round attack for our new rectangle framework
  - based on [HBS21, DDV20, QDW+21]
  - previous constraints: m_b, r_f, $\tilde{p}^2\tilde{q}^2 t$
  - new constraints : m_f, m'_f, h_f, h
  - objective : $\min\{T_1, T_2, T_3\}$

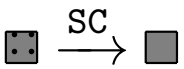


Modelling Propagation of Cells with Known Differences in E_f



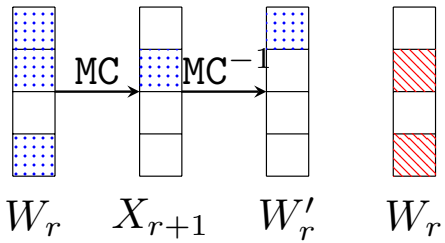
- `DXFixed(DWFixed)`: cells with known differences in $X_r(W_r)$, marked by ⚀ and ⚅
- the impact of `SC` and `SR` : $\texttt{DWFixed}[r][i] = \neg\texttt{DXL}[r_m + r_1 + r][P_{\texttt{SR}}[i]]$
- the impact of `MC` : $\texttt{DXFixed}[r+1][i] = \texttt{DWFixed}[r][i] \wedge \texttt{DWFixed}[r][i+8] \wedge \texttt{DWFixed}[r][i+12]$

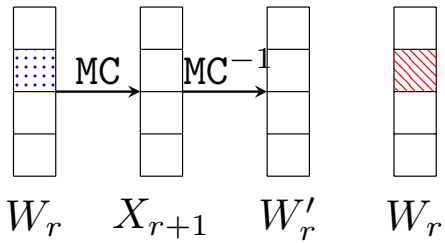
Modeling Cells that could be Used to Filter Quartets in E_f

- `DXFilter`: cells can be used as filters in X_r
  - cells with nonzero fixed difference: 

$\texttt{DXFixed}[r][i]$	$\texttt{DXL}[r_m + r_1 + r][i]$	$\texttt{DXFilter}[r][i]$
0	1	0
1	0	0
1	1	1

- `DWFilter`: cells can be used as filters in W_r
  - $r = N_f - 1$, $\texttt{DWFilter}[N_f - 1][i] = \texttt{DWFixed}[N_f - 1][i]$
  - $0 \leq r \leq N_f - 2$, extra cells with fixed differences in W_r than W'_r (deduced from X_{r+1}) can act as filters





Modeling m'_f-bit Subtweakeys in E_f for Generating Quartets

- $\texttt{DXGuess(DWGuess)}$: cells need to know in $X_r(W_r)$ in decryption from ciphertexts to the filters
- $0 \leq r \leq N_f - 1$, the cells in W_r need to know involve two types
  - cells to be known from X_r over the $\texttt{SR}$ operation
  - cells used as filters in W_r

$$\texttt{DWGuess}[r][i] = \texttt{DWisFilter}[r][i] \vee \texttt{DXGuess}[r][P_{\texttt{SR}}[i]]$$

- $0 \leq r \leq N_f - 2$, the cells in X_{r+1} need to know involve two types:
  - cells need to known from W_r over the $\texttt{MC}$ operation
  - cells used as filters in X_{r+1}

Modeling the Advantage h in the Key-recovery Attack

- $\texttt{Adv}$: the advantage h determines the exhaustive search time T_3
- $\texttt{KnownDec}$: the cells in Y_r need to know in the decryption from ciphertext to δ
- $h \leq m_b + m_f - x$ and $x \leq m_b + m'_f$:

$$\begin{cases} \texttt{x} \leq \sum\limits_{0 \leq r \leq N_b - 2,\ 0 \leq i \leq 7} \texttt{KnownEnc}[r][i] + \sum\limits_{0 \leq r \leq N_f - 1,\ 0 \leq i \leq 7} \texttt{DXGuess}[r][i] \\ \texttt{Adv} + \texttt{x} \leq \sum\limits_{0 \leq r \leq N_b - 2,\ 0 \leq i \leq 7} \texttt{KnownEnc}[r][i] + \sum\limits_{0 \leq r \leq N_f - 1,\ 0 \leq i \leq 7} \texttt{KnownDec}[r][i] \end{cases}$$

The Objective Function

- Time complexities: **Time I (T_1)**, **Time II (T_2)** and **Time III (T_3)**
  - $\texttt{DXU} \xrightarrow[w_0]{\text{upper differential}} \tilde{p}$, $\texttt{DXL} \xrightarrow[w_1]{\text{lower differential}} \tilde{q}$, $\texttt{DXU}\wedge\texttt{DXL} \xrightarrow[w_m]{\text{middle part}} t$
  - $\texttt{KnownEnc} \xrightarrow[w_{m_b}]{} m_b$, $\texttt{DXGuess} \xrightarrow[w_{m_f}]{} m_f'$
  - $\texttt{DXisFilter+DWisFilter} \xrightarrow[w_{h_f}]{} h_f$
  - $\texttt{Adv} \rightarrow h$
- The uniformed objective:

$$\text{Minimize } obj,\ obj \geq T_1,\ obj \geq T_2,\ obj \geq T_2.$$

New Distinguishers of SKINNY

- MILP+CP model (based on [DDV20, HBS21, QDW+21])

Version	N_d	Probability $\tilde{p}^2\tilde{q}^2t$	$N_b + N_d + N_f$	Ref.
64-128	17	$2^{-29.78}$	-	[SQH19]
	17	$2^{-48.72}$	21	[LGS17]
	19	$2^{-51.08}$	23	[HBS21]
	19	$2^{-54.36}$	-	[DDV20]
	18	$2^{-55.34}$	24	[QDW+21]
	18	$2^{-55.34}$	25	Ours
64-192	22	$2^{-42.98}$	-	[SQH19]
	22	$2^{-54.94}$	26	[LGS17]
	23	$2^{-55.85}$	29	[HBS21]
	23	$2^{-57.93}$	-	[DDV20]
	22	$2^{-57.73}$	30	[QDW+21]
	22	$2^{-57.56}$	31	Ours
128-256	18	$2^{-77.83}$	-	[SQH19]
	18	$2^{-103.84}$	22	[LGS17]
	20	$2^{-85.77}$	-	[DDV20]
	21	$2^{-116.43}$	24	[HBS21]
	19	$2^{-116.97}$	25	[QDW+21]
	18	$2^{-108.51}$	25	Ours
	19	$2^{-121.07}$	26	Ours
128-384	22	$2^{-48.30}$	-	[SQH19]
	23	2^{-112}	27	[LGS17]
	23	2^{-112}	28	[ZDM+20]
	24	$2^{-86.09}$	-	[DDV20]
	25	$2^{-116.59}$	30	[HBS21]
	22	$2^{-101.49}$	30	[QDW+21]
	23	$2^{-115.09}$	32	Ours

Outline

1 Background and Motivations

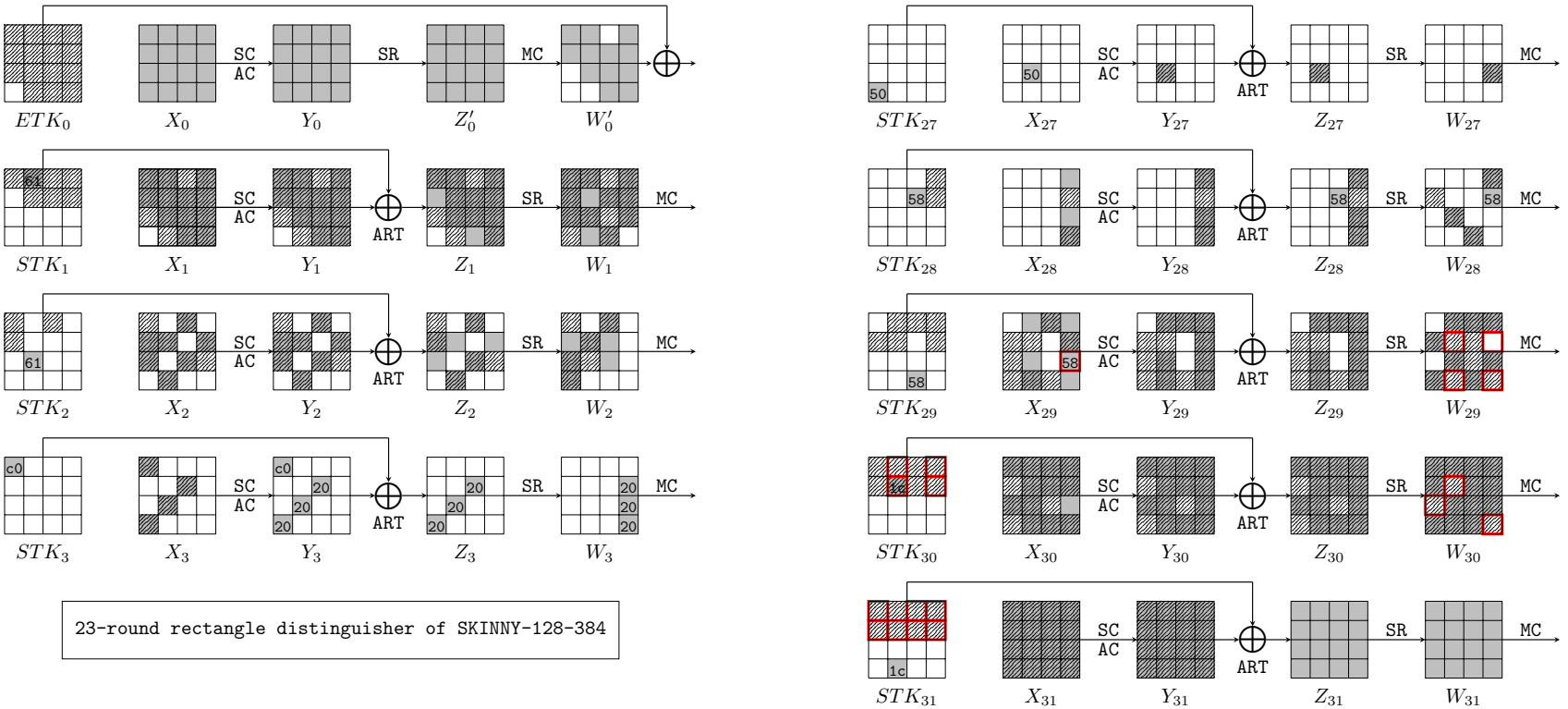
2 Key-Guessing Strategies in the Rectangle Attack

3 Automatic Model For SKINNY

4 Improved Rectangle Attacks on SKINNY

5 Conclusion and Further Disscussion

Improved Attack on 32-round SKINNY-128-384



- 4-round E_b+ 23-round distinguisher+5-round E_f
  - W'_0 →plaintext, Z_{31} →ciphertext
  - $r_b = 12c$, $m_b = 18c$, $r_f = 16c$, $m_f = 24c$

Internal State Used for Filtering and Involved Subtweakeys

	Round	Filter	Involved subtweakeys
1	30	$\boldsymbol{\Delta W_{30}[5] = 0}$	$STK_{31}[5]$
2		$\boldsymbol{\Delta W_{30}[8] = 0}$	$STK_{31}[4]$
3		$\boldsymbol{\Delta W_{30}[15] = 0}$	$STK_{31}[3]$
4	29	$\boldsymbol{\Delta W_{29}[5] = 0}$	$STK_{30}[5]$, $STK_{31}[0,6,7]$
5		$\boldsymbol{\Delta W_{29}[7] = 0}$	$STK_{30}[7]$, $STK_{31}[2,4,5]$
6		$\Delta W_{29}[10] = 0$	$STK_{30}[6]$, $STK_{31}[1,7]$
7		$\boldsymbol{\Delta W_{29}[13] = 0}$	$STK_{30}[1]$, $STK_{31}[0,5]$
8		$\boldsymbol{\Delta W_{29}[15] = 0}$	$STK_{30}[3]$, $STK_{31}[2,7]$
9		$\boldsymbol{\Delta X_{29}[11]} = 0\text{x}58$	$STK_{30}[5]$, $STK_{31}[0,6]$
10	28	$\Delta W_{28}[5] = 0$	$STK_{29}[5]$, $STK_{30}[0,6,7]$, $STK_{31}[1,2,3,4,7]$
11		$\Delta W_{28}[11] = 0$	$STK_{29}[7]$, $STK_{30}[2,4]$, $STK_{31}[1,3,5,6]$
12		$\Delta W_{28}[13] = 0$	$STK_{29}[1]$, $STK_{30}[0,5]$, $STK_{31}[3,4,6]$
13		$\Delta W_{28}[15] = 0$	$STK_{29}[3]$, $STK_{30}[2,7]$, $STK_{31}[1,4,6]$
14	27	$\Delta W_{27}[7] = 0$	$STK_{28}[7]$, $STK_{29}[2,4,5]$, $STK_{30}[0,1,3,5,6]$, $STK_{31}[0,1,2,3,4,5,6,7]$
15		$\Delta W_{27}[15] = 0$	$STK_{28}[3]$, $STK_{29}[2,7]$, $STK_{30}[1,4,6]$, $STK_{31}[0,3,5,6,7]$
16		$\Delta X_{27}[9] = 0\text{x}50$	$STK_{28}[7]$,$STK_{29}[2,4]$, $STK_{30}[1,3,5,6]$, $STK_{31}[0,1,2,5,6,7]$

- $k'_f = \{STK_{30}[1,3,5,7], STK_{31}[0,2,3,4,5,6,7]\}$ $(m'_f = 11c)$
- $h_f = \{X_{29}[11], W_{29}[5,7,13,15], W_{30}[5,8,15]\}$ $(h_f = 8c)$

Tweakey Recovery for 32-round SKINNY-128-384

Step	Internal state	Involved subtweakeys
A	$Z_{30}[6]$	$STK_{31}[7]$
	$Z_{30}[14]$	$STK_{31}[1]$
B	$Z_{29}[1]$	$STK_{30}[5]$, $STK_{31}[6]$
	$Z_{29}[9]$	$STK_{30}[7]$, $STK_{31}[2,4]$
	$Z_{29}[13]$	$STK_{30}[0]$, $STK_{31}[3,4]$
C	$Z_{29}[3]$	$STK_{30}[7]$, $STK_{31}[4]$
	$Z_{29}[7]$	$STK_{30}[4]$, $STK_{31}[3,5,6]$
	$Z_{29}[15]$	$STK_{30}[2]$, $STK_{31}[1,6]$
D	$Z_{28}[3]$	$STK_{29}[7]$, $STK_{30}[4]$, $STK_{31}[3,5,6]$
	$Z_{28}[11]$	$STK_{29}[5]$, $STK_{30}[0,6]$, $STK_{31}[1,3,4,7]$
	$Z_{28}[15]$	$STK_{29}[2]$, $STK_{30}[1,6]$, $STK_{31}[0,5,7]$
E	$X_{27}[9]$	$STK_{28}[7]$,$STK_{29}[2,4]$, $STK_{30}[1,3,5,6]$, $STK_{31}[0,1,2,5,6,7]$

- set $s = 1$, $h = 40$, $x = 208$
- Time complexity: $2^{354.99}$
  - $T_1 = \sqrt{s} \cdot 2^{m_b+m'_f+n/2+1} / \sqrt{\tilde{p}^2 t \tilde{q}^2} = 2^{354.54}$
  - $T_3 = s \cdot 2^{m_b+m'_f-2h_f-n+2r_f} / (\tilde{p}^2 t \tilde{q}^2) \cdot \varepsilon = 2^{353.1}$
  - $T_3 = 2^{k-h} = 2^{344}$
- Success probability [Sel08]: $P_s = \Phi(\frac{\sqrt{sS_N} - \Phi^{-1}(1-2^{-h})}{\sqrt{S_N+1}}) = 82.1\%$

Summary of Attacks on Skinny in Related-tweakey Setting

Version	Rounds	Data	Time	Memory	Approach	Setting	Ref.
64-128	22	$2^{63.5}$	$2^{110.9}$	$2^{63.5}$	Rectangle	RK	[LGS17]
	23	$2^{62.47}$	$2^{125.91}$	2^{124}	ID	RK	[LGS17]
	23	$2^{62.47}$	2^{124}	$2^{77.47}$	ID	RK	[SMB18]
	23	$2^{71.4}$	2^{79}	$2^{64.0}$	ID	RK	[ABC$^+$]
	23	$2^{60.54}$	$2^{120.7}$	$2^{60.9}$	Rectangle	RK	[HBS21]
	24	$2^{61.67}$	$2^{96.83}$	2^{84}	Rectangle	RK	[QDW$^+$21]
	25	$2^{61.67}$	$2^{118.43}$	$2^{64.26}$	Rectangle	RK	Ours
64-192	27	$2^{63.5}$	$2^{165.5}$	2^{80}	Rectangle	RK	[LGS17]
	29	$2^{62.92}$	$2^{181.7}$	2^{80}	Rectangle	RK	[HBS21]
	30	$2^{62.87}$	$2^{163.11}$	$2^{68.05}$	Rectangle	RK	[QDW$^+$21]
	31	$2^{62.78}$	$2^{182.07}$	$2^{62.79}$	Rectangle	RK	Ours
128-256	22	2^{127}	$2^{235.6}$	2^{127}	Rectangle	RK	[LGS17]
	23	$2^{124.47}$	$2^{251.47}$	2^{248}	ID	RK	[LGS17]
	23	$2^{124.41}$	$2^{243.41}$	$2^{155.41}$	ID	RK	[SMB18]
	24	$2^{125.21}$	$2^{209.85}$	$2^{125.54}$	Rectangle	RK	[HBS21]
	25	$2^{124.48}$	$2^{226.38}$	2^{168}	Rectangle	RK	[QDW$^+$21]
	25	$2^{120.25}$	$2^{193.91}$	2^{136}	Rectangle	RK	Ours
	26	$2^{126.53}$	$2^{254.4}$	$2^{128.44}$	Rectangle	RK	Ours
128-384	27	2^{123}	2^{331}	2^{155}	Rectangle	RK	[LGS17]
	28	2^{122}	$2^{315.25}$	$2^{122.32}$	Rectangle	RK	[ZDM$^+$20]
	30	$2^{125.29}$	$2^{361.68}$	$2^{125.8}$	Rectangle	RK	[HBS21]
	30	2^{122}	$2^{341.11}$	$2^{128.02}$	Rectangle	RK	[QDW$^+$21]
	32	$2^{123.54}$	$2^{354.99}$	$2^{123.54}$	Rectangle	RK	Ours

Outline

1 Background and Motivations

2 Key-Guessing Strategies in the Rectangle Attack

3 Automatic Model For SKINNY

4 Improved Rectangle Attacks on SKINNY

5 Conclusion and Further Disscussion

Application to ForkSkinny, Deoxys-BC and GIFT

Version	Rounds	Data	Time	Memory	Approach	Setting	Ref.
ForkSkinny							
128-256 (256-bit key)	26	2^{125}	$2^{254.6}$	2^{160}	ID	RK	[BDL20]
	26	2^{127}	$2^{250.3}$	2^{160}	ID	RK	[BDL20]
	28	$2^{118.88}$	$2^{246.98}$	2^{136}	Rectangle	RK	[QDW+21]
	28	$2^{118.88}$	$2^{224.76}$	$2^{118.88}$	Rectangle	RK	Ours
Deoxys-BC							
128-384	13	2^{127}	2^{270}	2^{144}	Rectangle	RK	[CHP+17]
	14	2^{127}	$2^{286.2}$	2^{136}	Rectangle	RK	[ZDJ19]
	14	$2^{125.2}$	$2^{282.7}$	2^{136}	Rectangle	RK	[ZDJM19]
	14	$2^{125.2}$	2^{260}	2^{140}	Rectangle	RK	Ours
GIFT							
64-128	25	$2^{63.78}$	$2^{120.92}$	$2^{64.1}$	Rectangle	RK	[JZZD20]
	26	$2^{60.96}$	$2^{123.23}$	$2^{102.86}$	Differential	RK	[SWW21]
	26	$2^{63.78}$	$2^{122.78}$	$2^{63.78}$	Rectangle	RK	Ours

Application in Single-key Setting

- Rectangle attack in single-key model: set differences of keys to 0
  - set $y = \sqrt{s} \cdot 2^{n/2-r_b+1.5}/\hat{p}\hat{q}$, data complexity $y \cdot 2^{r_b}$
  - $T_1 = 2^{m_b+m_f'} \cdot y \cdot 2^{r_b} = \sqrt{s} \cdot 2^{m_b+m_f'+\frac{n}{2}+1.5}/\hat{p}\hat{q}$
  - $T_2 = (s \cdot 2^{m_b+m_f'-n+2r_f-2h_f}/\hat{p}^2\hat{q}^2) \cdot \varepsilon$
- Application to Serpent:
  - reuse the rectangle distinguisher by Biham, Dunkelman and Keller [BDK01]

Key size	Time	Data	Memory	Approach	Ref.
256	$2^{173.8}$	$2^{126.3}$	$2^{126.3}$	rectangle	[BDK02]
256	2^{164}	$2^{126.3}$	$2^{126.3}$	rectangle	ours

Overall Analysis of the Four Attack Models

- For generating quartets:
  - `Attack I`: guesses all the $(m_b + m_f)$-bit key at once
  - `Attack II`: does not guess the key involved in E_b and E_f
  - `Attack III`: guesses m_b-bit key in E_b
  - Our new attack: guesses m_b-bit key in E_b and m'_f-bit key in E_f
- Time complexities:
  - `Attack I`: $T_{\mathtt{I}} = 2^{m_b+m_f+n/2+t+2}$
  - `Attack II`: $T_{\mathtt{II}} = 2^{m_b+r_b+2r_f-n+2t} + 2^{m_f+2r_b+r_f-n+2t}$
  - `Attack III`: $T_{\mathtt{III}} = 2^{m_b+2r_f-n+2t} \cdot \varepsilon$
  - `Attack IV`: $T_{\mathtt{IV}} = 2^{m_b+2r_f-n+m'_f-2h_f+2t} \cdot \varepsilon$

Thanks for Your Attention!

Reference I


Ralph Ankele, Subhadeep Banik, Avik Chakraborti, Eik List, Florian Mendel, Siang Meng Sim, and Gaoli Wang.
Related-key impossible-differential attack on reduced-round SKINNY.
In *ACNS 2017*, volume 10355, pages 208–228.


Eli Biham, Orr Dunkelman, and Nathan Keller.
The rectangle attack - rectangling the serpent.
In *EUROCRYPT 2001, Proceeding*, volume 2045, pages 340–357, 2001.


Eli Biham, Orr Dunkelman, and Nathan Keller.
New results on boomerang and rectangle attacks.
In *FSE 2002, Revised Papers*, volume 2365, pages 1–16, 2002.


Eli Biham, Orr Dunkelman, and Nathan Keller.
Related-key boomerang and rectangle attacks.
In *EUROCRYPT 2005, Proceedings*, volume 3494, pages 507–525, 2005.


Augustin Bariant, Nicolas David, and Gaëtan Leurent.
Cryptanalysis of Forkciphers.
IACR Trans. Symmetric Cryptol., 2020(1):233–265, 2020.

Reference II


Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim.
The SKINNY family of block ciphers and its low-latency variant MANTIS.
In *CRYPTO 2016, Proceedings, Part II*, pages 123–153, 2016.


Alex Biryukov and Dmitry Khovratovich.
Related-key cryptanalysis of the full AES-192 and AES-256.
In *ASIACRYPT 2009*, volume 5912, pages 1–18, 2009.


Carlos Cid, Tao Huang, Thomas Peyrin, Yu Sasaki, and Ling Song.
A security analysis of Deoxys and its internal tweakable block ciphers.
IACR Trans. Symmetric Cryptol., 2017(3):73–107, 2017.


Carlos Cid, Tao Huang, Thomas Peyrin, Yu Sasaki, and Ling Song.
Boomerang connectivity table: A new cryptanalysis tool.
In *EUROCRYPT 2018, Proceedings, Part II*, volume 10821, pages 683–714, 2018.


Stéphanie Delaune, Patrick Derbez, and Mathieu Vavrille.
Catching the fastest boomerangs application to SKINNY.
IACR Trans. Symmetric Cryptol., 2020(4):104–129, 2020.

Reference III


Orr Dunkelman, Nathan Keller, and Adi Shamir.
A practical-time related-key attack on the KASUMI cryptosystem used in GSM and 3G telephony.
J. Cryptology, 27(4):824–849, 2014.


Hosein Hadipour, Nasour Bagheri, and Ling Song.
Improved rectangle attacks on SKINNY and CRAFT.
IACR Trans. Symmetric Cryptol., 2021(2):140–198, 2021.


Fulei Ji, Wentao Zhang, Chunning Zhou, and Tianyou Ding.
Improved (related-key) differential cryptanalysis on GIFT.
Accepted to SAC 2020, 2020.


John Kelsey, Tadayoshi Kohno, and Bruce Schneier.
Amplified boomerang attacks against reduced-round MARS and Serpent.
In *FSE 2000*, volume 1978, pages 75–93, 2000.


Guozhen Liu, Mohona Ghosh, and Ling Song.
Security analysis of SKINNY under related-tweakey settings.
IACR Transactions on Symmetric Cryptology, 2017(3):37–72, 2017.

Reference IV


Sean Murphy.
The return of the cryptographic boomerang.
IEEE Transactions on Information Theory, 57(4):2517–2521, 2011.


Lingyue Qin, Xiaoyang Dong, Xiaoyun Wang, Keting Jia, and Yunwen Liu.
Automated search oriented to key recovery on ciphers with linear key schedule applications to boomerangs in SKINNY and ForkSkinny.
IACR Trans. Symmetric Cryptol., 2021(2):249–291, 2021.


Ali Aydin Selçuk.
On probability of success in linear and differential cryptanalysis.
J. Cryptology, 21(1):131–147, 2008.


Sadegh Sadeghi, Tahereh Mohammadi, and Nasour Bagheri.
Cryptanalysis of reduced round SKINNY block cipher.
IACR Transactions on Symmetric Cryptology, 2018(3):124–162, 2018.


Ling Song, Xianrui Qin, and Lei Hu.
Boomerang connectivity table revisited. application to SKINNY and AES.
IACR Transactions on Symmetric Cryptology, 2019(1):118–141, 2019.

Reference V


Ling Sun, Wei Wang, and Meiqin Wang.
Accelerating the search of differential and linear characteristics with the SAT method.
IACR Trans. Symmetric Cryptol., 2021(1):269–315, 2021.


David A. Wagner.
The boomerang attack.
In *FSE '99, Proceedings*, volume 1636, pages 156–170, 1999.


Haoyang Wang and Thomas Peyrin.
Boomerang switch in multiple rounds. application to AES variants and deoxys.
IACR Trans. Symmetric Cryptol., 2019(1):142–169, 2019.


Boxin Zhao, Xiaoyang Dong, and Keting Jia.
New related-tweakey boomerang and rectangle attacks on Deoxys-BC including BDT effect.
IACR Trans. Symmetric Cryptol., 2019(3):121–151, 2019.


Boxin Zhao, Xiaoyang Dong, Keting Jia, and Willi Meier.
Improved related-tweakey rectangle attacks on reduced-round Deoxys-BC-384 and Deoxys-I-256-128.
In *INDOCRYPT 2019, Proceedings*, pages 139–159, 2019.

Reference VI

Boxin Zhao, Xiaoyang Dong, Willi Meier, Keting Jia, and Gaoli Wang.
Generalized related-key rectangle attacks on block ciphers with linear key schedule: applications to SKINNY and GIFT.
Designs, Codes and Cryptography, 88(6):1103–1126, 2020.