Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Supplementary materials for

Xiaowei LI, Jiongjiong REN, Shaozhen CHEN, 2024. Improved deep learning aided key recovery framework: applications to large-state block ciphers. *Front Inform Technol Electron Eng*, 25(10):1406-1420.
https://doi.org/10.1631/FITEE.2300848

Table S1 The algorithm of Gohr's key recovery attack

Require: k neutral bits, neural distinguisher ND
Ensure: Candidate key rk

1: Randomly generate plaintext pairs$(P, P+\varGamma)$, and expand the plaintext pairs into 2^k plaintext structures using the k neutral bits of the pre-difference.
2: Encrypt and obtain the corresponding ciphertext structures.
3: Guess rk, for each of possible kg:
4: Decrypt the 2^k ciphertext pairs by one round using kg.
5: Feed pairs into ND, obtain scores Z_j for $j \in \left[1, 2^k\right]$.
6: Combine the scores using the following formula $v_{kg} := \sum_{j=1}^{2^k} \log_2\left(\frac{Z_j}{1-Z_j}\right)$.
7: If $v_{kg} > c$, c is the threshold, save kg as a possible candidate key.
8: Return 3 and repeat the process until a candidate key is found.

Table S2 The parameters of the SIMON family

Block size	Key size	Word size	Key word	Consent sequence	Round
32	64	16	4	z_0	32
48	72	24	3	z_0	36
	96	24	4	z_1	36
64	96	32	3	z_2	42
	128	32	4	z_3	44
96	96	48	2	z_2	52
	144	48	3	z_3	54
128	128	64	2	z_2	68
	192	64	3	z_3	69
	256	64	4	z_4	72

Algorithm S1 The multi-stage deep learning aided key recovery framework

Require: A section of the key bit set $B_i, i \in [1, x]$; a small constant ϵ; the thresholds on the scores for filtering wrong key guesses $c_i, i \in [1, x]$; the upper bound of the number of kept surviving key guesses $\beta_i, i \in [1, x]$.

Ensure: The guessed value $\overrightarrow{kg_x}$ for rk.

1: **for** $i \in [1, x]$, **do**

2: Launch Stage i by choosing $\frac{\epsilon}{p_i}$ plaintext pairs with difference $\varGamma_i$. Expand the plaintext pairs into plaintext structures using the $\log_2 N_i$ neutral bits of CD_i;

3: **for** $d = 1$ to $\frac{\epsilon}{p_i}$, **do**

4: Encrypt and obtain the corresponding ciphertext structures. Note that each ciphertext structure contains N_i ciphertext pairs;

5: Initialize a list $L_i \longleftarrow \emptyset$;

6: Denote the β_i top-ranked partial key guesses for bits in $\bigcup_{j\in[1,i-1]} B_j$ that were recommended from the previous stages by $\overrightarrow{kg_{i-1}} := kg_{i-1}||\ldots||kg_1$ (for Stage1, $\beta_1 = 1$ and $\overrightarrow{kg_{i-1}} = \emptyset$);

7: **for** $k = 1$ to β_i, **do**

8: The $2^{|B_i|}$ possible value kg_i of the key bits in B_i;

9: **for** $h = 1$ to $2^{|B_i|}$, **do**

10: Denote the concatenation $kg_i||kg_{i-1}||\ldots||kg_1$ by $\overrightarrow{kg_i}$;

11: Partially decrypt the N_i ciphertext pairs by one round using $\overrightarrow{kg_{i-1}}$ to obtain pairs of values for state bits in C_i;

12: Feed N_i partial state pairs into ND_i, obtainN_i scores Z_j for $j \in [1, N_i]$;

13: Combine the scores using the following formula $v_{\overrightarrow{kg_i}} := \sum_{j=1}^{N_i} \log_2 \left(\frac{Z_j}{1-Z_j}\right)$;

14: **end for**

15: **if** $v_{\overrightarrow{kg_i}} > c_i$, **then**

16: Store $(\overrightarrow{kg_i}, v_{\overrightarrow{kg_i}})$ in L_i;

17: **end if**

18: **end for**

19: **if** $L_i \neq \emptyset$, **then**

20: sort L_i according to the scores of the guessed key bits, and take the β_{i+1} top-ranked values as the guessed value for the key-bits in $\bigcup_{j\in[1,i]} B_j$. Go to Step 2;

21: **end if**

22: **end for**

23: If all $\frac{\epsilon}{p_i}$ ciphertext structures have been used and no values of $\overrightarrow{kg_i}$ obtain a score passing c_i, terminate the attack with output $\perp$;

24: **end for**

25: Return the concatenated key bits $\overrightarrow{kg_x}$ with the highest score in the last stage as the guessed value for rk.

Algorithm S2 Key bit sensitivity test

Require: A cipher with a word size(round key size) of n;a neural distinguisher ND^t; a test dataset consisting of $\frac{M}{2}$ positive samples and $\frac{M}{2}$ negative samples.

Ensure: An array sen that saves the bit sensitivity of m ciphertext bits.

Test the distinguishing accuracy of ND^t on the test dataset. Save it to $sen\,[m]$;
Generate $t+1$-round key ks;
Encrypt 1 round test with $ks[t]$, $enc\_one\_round\,(C, ks\,[t])$, denote as C';
for $j = 0$ to $n-1$, **do**
    $ks\,[t] = ks\,[t] \wedge 2^j$;
    Decrypt 1 round test with $ks[t]$, $dec\_one\_round\,(C', ks\,[t])$, and generate new test dataset denote as C'';
    Test the distinguishing accuracy of ND^t on the new test dataset C'', denote as cp;
    $sen\,[j] = sen\,[n] - cp$;
end for
return sen.

Table S3 The parameters of the SPECK family

Block size	Key size	Word size	Key word	(α, β)	Round
32	64	16	4	$(7, 2)$	22
48	72	24	3	$(8, 3)$	22
	96	24	4	$(8, 3)$	23
64	96	32	3	$(8, 3)$	26
	128	32	4	$(8, 3)$	27
96	96	48	2	$(8, 3)$	28
	144	48	3	$(8, 3)$	29
128	128	64	2	$(8, 3)$	32
	192	64	3	$(8, 3)$	33
	256	64	4	$(8, 3)$	34

Table S4 9-round neural distinguisher combination for SPECK128

	Chen's 9-round SPECK128				Improved 9-round SPECK128			
ND_i	Δ_i	B_i	C_i	Accuracy	Δ_i	B_i	C_i	Accuracy
ND_1	$\Delta_{[64]}$	$\{14 \sim 0\}$	$\{22 \sim 18\}$ $\{14 \sim 9\}$	0.559	$\Delta_{[64]}$	$\{14 \sim 0\}$	$\{22 \sim 17\}$ $\{14 \sim 9\}$	0.609
ND_2	$\Delta_{[76]}$	$\{26 \sim 15\}$	$\{34 \sim 30\}$ $\{26 \sim 21\}$	0.586	$\Delta_{[76]}$	$\{26 \sim 15\}$	$\{34 \sim 30\}$ $\{26 \sim 19\}$	0.624
ND_3	$\Delta_{[90]}$	$\{40 \sim 27\}$	$\{48 \sim 44\}$ $\{40 \sim 34\}$	0.609	$\Delta_{[90]}$	$\{40 \sim 27\}$	$\{48 \sim 44\}$ $\{40 \sim 32\}$	0.622
ND_4	$\Delta_{[105]}$	$\{55 \sim 41\}$	$\{63 \sim 59\}$ $\{55 \sim 49\}$	0.616	$\Delta_{[105]}$	$\{55 \sim 41\}$	$\{63 \sim 59\}$ $\{55 \sim 47\}$	0.623
ND_5	$\Delta_{[117]}$	$\{63 \sim 56\}$	$\{11, 7, 4\}$ $\{3, 0\}$	0.559	$\Delta_{[117]}$	$\{63 \sim 56\}$	$\{63, 62, 11, 8\}$ $\{7, 4 \sim 0\}$	0.644

Table S5 7-round neural distinguisher combination for SPECK96

	Chen's 7-round SPECK96				Improved 7-round SPECK96			
ND_i	Δ_i	B_i	C_i	Accuracy	Δ_i	B_i	C_i	Accuracy
ND_1	$\Delta_{[53]}$	$\{11 \sim 0\}$	$\{19 \sim 8\}$	0.633	$\Delta_{[53]}$	$\{11 \sim 0\}$	$\{19 \sim 8\}$	0.633
ND_2	$\Delta_{[65]}$	$\{23 \sim 12\}$	$\{31 \sim 20\}$	0.621	$\Delta_{[65]}$	$\{23 \sim 12\}$	$\{31 \sim 20\}$	0.621
ND_3	$\Delta_{[77]}$	$\{35 \sim 24\}$	$\{43 \sim 32\}$	0.628	$\Delta_{[77]}$	$\{35 \sim 24\}$	$\{43 \sim 29\}$	0.690
ND_4	$\Delta_{[89]}$	$\{47 \sim 36\}$	$\{47 \sim 44\}$ $\{7 \sim 0\}$	0.634	$\Delta_{[89]}$	$\{47 \sim 36\}$	$\{47 \sim 44\}$ $\{7 \sim 0\}$	0.634

Table S6 6-round neural distinguisher combination for SPECK64

	Chen's 6-round SPECK64				Improved 6-round SPECK64			
ND_i	Δ_i	B_i	C_i	Accuracy	Δ_i	B_i	C_i	Accuracy
ND_1	$\Delta_{[42]}$	$\{9 \sim 0\}$	$\{17 \sim 8\}$	0.613	$\Delta_{[42]}$	$\{9 \sim 0\}$	$\{17 \sim 8\}$	0.613
ND_2	$\Delta_{[47]}$	$\{21 \sim 10\}$	$\{29 \sim 18\}$	0.677	$\Delta_{[47]}$	$\{21 \sim 10\}$	$\{29 \sim 15\}$	0.728
ND_3	$\Delta_{[33]}$	$\{31 \sim 22\}$	$\{31, 30\}$ $\{7 \sim 0\}$	0.653	$\Delta_{[33]}$	$\{31 \sim 22\}$	$\{31, 30, 14\}$ $\{13, 7 \sim 0\}$	0.725

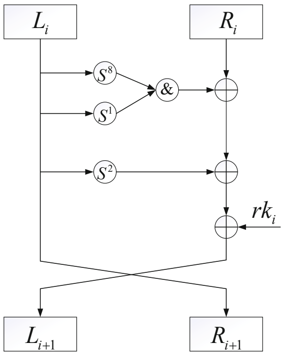

Fig. S1 The round transformation of SIMON

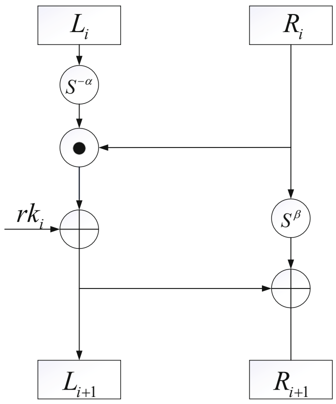

Fig. S2 The round transformation of SPECK

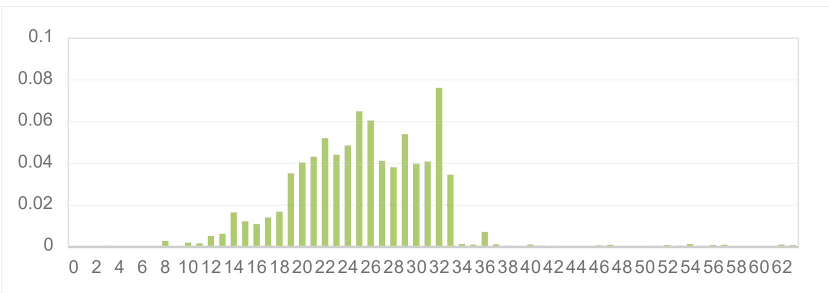

Fig. S3 The outcomes of KBST on the neural distinguisher for 15-round SIMON128

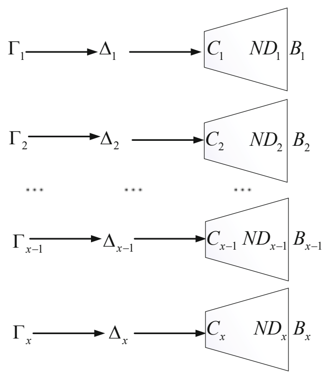

Fig. S4 The schematic of the improved multi-stage key recovery framework for large-state block ciphers. A total of x neural distinguishers ND_i are used, whose input differences are Δ_i, each ND_i is prepended with a CD_i, and CD_i is defined as $\Gamma_i \to \Delta_i, i \in [1, x]$. Each ND_i is trained on partial state bits C_i, and is used to recover partial key bits $B_i, i \in [1, x]$