Revamped Differential-Linear Cryptanalysis on Reduced Round ChaCha

Sabyasachi Dey[1], Hirendra Kumar Garai[1], Santanu Sarkar[2], Nitin Kumar Sharma[1]

[1] Birla Institute of Technology and Science Pilani,
[2]Indian Institute of Technology Madras, India

EUROCRYPT 2022
Trondheim, Norway

Outline of the talk

1. Description of ChaCha
2. Attack ideas
    a. Differential attack & our refinements
    b. Probabilistic Neutral Bits (PNBs) & our approach
    c. Key recovery attack
    d. Our results
    e. Conclusion

Description of ChaCha

ChaCha

$$X^{(0)} = \begin{pmatrix} X_0 & X_1 & X_2 & X_3 \\ X_4 & X_5 & X_6 & X_7 \\ X_8 & X_9 & X_{10} & X_{11} \\ X_{12} & X_{13} & X_{14} & X_{15} \end{pmatrix} = \begin{pmatrix} c_0 & c_1 & c_2 & c_3 \\ k_0 & k_1 & k_2 & k_3 \\ k_4 & k_5 & k_6 & k_7 \\ t_0 & v_0 & v_1 & v_2 \end{pmatrix}$$

Each word is of 32 bits. Total 16 words.

For ChaCha256:
$c_0 = 0x61707865$; $c_1 = 0x3320646e$; $c_2 = 0x79622d32$;
$c_3 = 0x6b206574$.

For ChaCha128:
$c_0 =$ `0x61707865`, $c_1 =$ `0x3120646e`, $c_2 =$ `0x79622d36`, $c_3 =$ `0x6b206574` and $k_{i+4} = k_i$ for $0 \leq i \leq 3$.

Quarterround Function (QRF)

$$\begin{pmatrix} X_0 & X_1 & X_2 & X_3 \\ X_4 & X_5 & X_6 & X_7 \\ X_8 & X_9 & X_{10} & X_{11} \\ X_{12} & X_{13} & X_{14} & X_{15} \end{pmatrix}$$

- Input of quaterround function is (a, b, c, d) and the corresponding output is (a'', b'', c'', d'').

$$\begin{aligned} a' &= a \boxplus b; & d' &= ((d \oplus a') \lll 16); \\ c' &= c \boxplus d'; & b' &= ((b \oplus c') \lll 12); \\ a'' &= a' \boxplus b'; & d'' &= ((d' \oplus a'') \lll 8); \\ c'' &= c' \boxplus d''; & b'' &= ((b' \oplus c'') \lll 7); \end{aligned}$$

- ChaCha applies the quaterround function along columns and diagonals alternately.
- Output is $Z = X^{(0)} \boxplus X^{(r)}$

Quarterround Function (QRF)

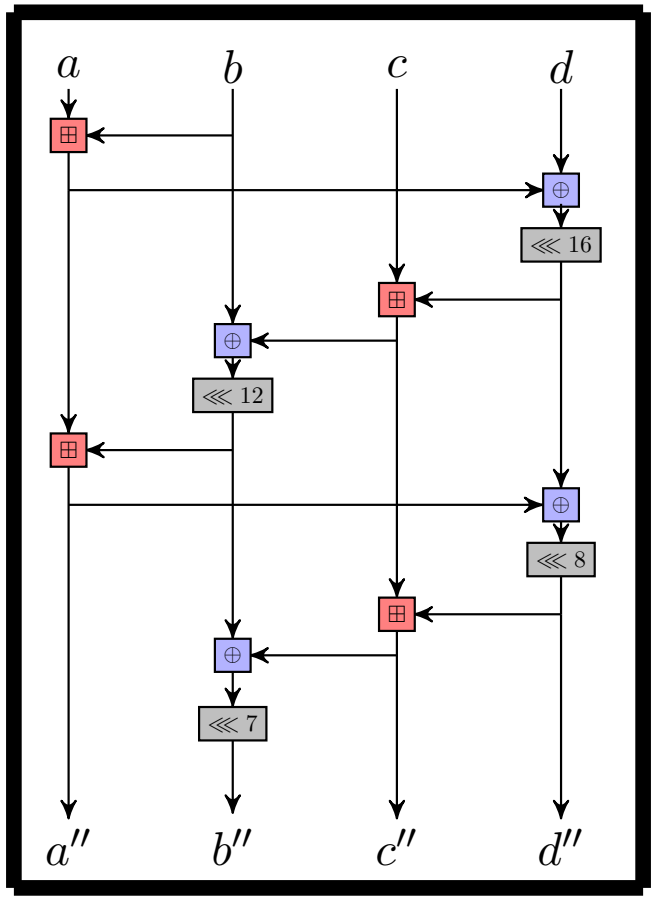


Figure: One quarterround function in ChaCha

QRF is reversible.

Differential attack & our refinements

Differential Analysis on ChaCha

- Start with two ChaCha states
  - Same at most of the positions
  - Different in a very few positions (may be just one bit)
- Apply a few rounds
- Check whether any correlation can be obtained between two states
- Same key, different IV.

Differential State Matrix

Given two states $X^{(r)}, X'^{(r)}$ after r rounds, we represent the differential state matrix as

$$\Delta^{(r)} = \begin{pmatrix} \Delta_0^{(r)} & \Delta_1^{(r)} & \Delta_2^{(r)} & \Delta_3^{(r)} \\ \Delta_4^{(r)} & \Delta_5^{(r)} & \Delta_6^{(r)} & \Delta_7^{(r)} \\ \Delta_8^{(r)} & \Delta_9^{(r)} & \Delta_{10}^{(r)} & \Delta_{11}^{(r)} \\ \Delta_{12}^{(r)} & \Delta_{13}^{(r)} & \Delta_{14}^{(r)} & \Delta_{15}^{(r)} \end{pmatrix},$$

where $\Delta_i^{(r)} = X_i^{(r)} \oplus X_i'^{(r)}$.

Differential Bias

- Let $X^{(0)}$ and $X'^{(0)}$ be two initial state matrix with a given input difference ($\mathcal{ID}$) $\Delta_{i,j}^{(0)} = 1$, where $\Delta_{i,j}^{(0)}$ is j-th bit of i-th word of the initial differential state matrix.
- After that we observe a high bias ϵ_d in an output difference (OD) $\Delta_{p,q}^{(r)}$ after r ChaCha rounds.Thus

$$Pr(\Delta_{p,q}^{(r)} = 1 \mid \Delta_{i,j}^{(0)} = 1) = \frac{1}{2}(1 + \epsilon_d),$$

where $\Delta^{(r)} = X^{(r)} \oplus X'^{(r)}$.

Right Pair

$$X = \begin{pmatrix} c_0 & c_1 & c_2 & c_3 \\ k_0 & k_1 & k_2 & k_3 \\ k_4 & k_5 & k_6 & k_7 \\ t_0 & v_0 & v_1 & v_2 \end{pmatrix}$$

Give one bit difference in the last row of X to create X'

Choose IV such that after applying one round on X, X', Hamming weight of differential state matrix is 10.

Choosing a Right Pair: our observation

- Beierle, Leander, Todo (CRYPTO 2020) mentioned that in a column of ChaCha, there are approximately 30% keys which do not have any IVs to form a right pair.
- They showed out of 2^5 IVs, on average one gives 10 differences after one round.
- If we allow 12 differences, we need 9 IVs.

Hamming weight	$= 10$	$\leqslant 12$
ϵ_d	0.00317	0.0021
Percentage of weak keys	70	100
Probability of satisfy ($\mathtt{Pr}$)	2^{-5}	$2^{-3.17}$
$\mathtt{Pr} \cdot \epsilon_d^2$	$2^{-21.602}$	$2^{-20.96}$

Table: Comparison between the two criteria for right pair on the Hamming weights of the output difference after one round

Use of memory

$$X = \begin{pmatrix} c_0 & c_1 & c_2 & c_3 \\ k_0 & k_1 & k_2 & k_3 \\ k_4 & k_5 & k_6 & k_7 \\ t_0 & v_0 & v_1 & v_2 \end{pmatrix}$$

No more required to run $\mathtt{Pr}^{-1}$ random IVs to get one IV to construct a right pair.

Table size: 0.7×2^{64}

We partition the set of key bit positions of the $\mathcal{ID}$ column into two subsets K_{mem} and K_{nmem}.

Store all possible values corresponding to K_{mem} positions and their corresponding IVs.

$\mathcal{ID}$ difference at (13,6)
$K_{nmem} =$
$\{39, 47, 48, 168, 191, 163, 164, 165, 171, 172, 173, 174, 175, 176, 183, 184, 185, 186\}$

Table size: 0.62×2^{46}

Probabilistic Neutral Bits (PNBs) & our approach

Ideas of PNB

Output $Z = X \boxplus X^{(R)}$

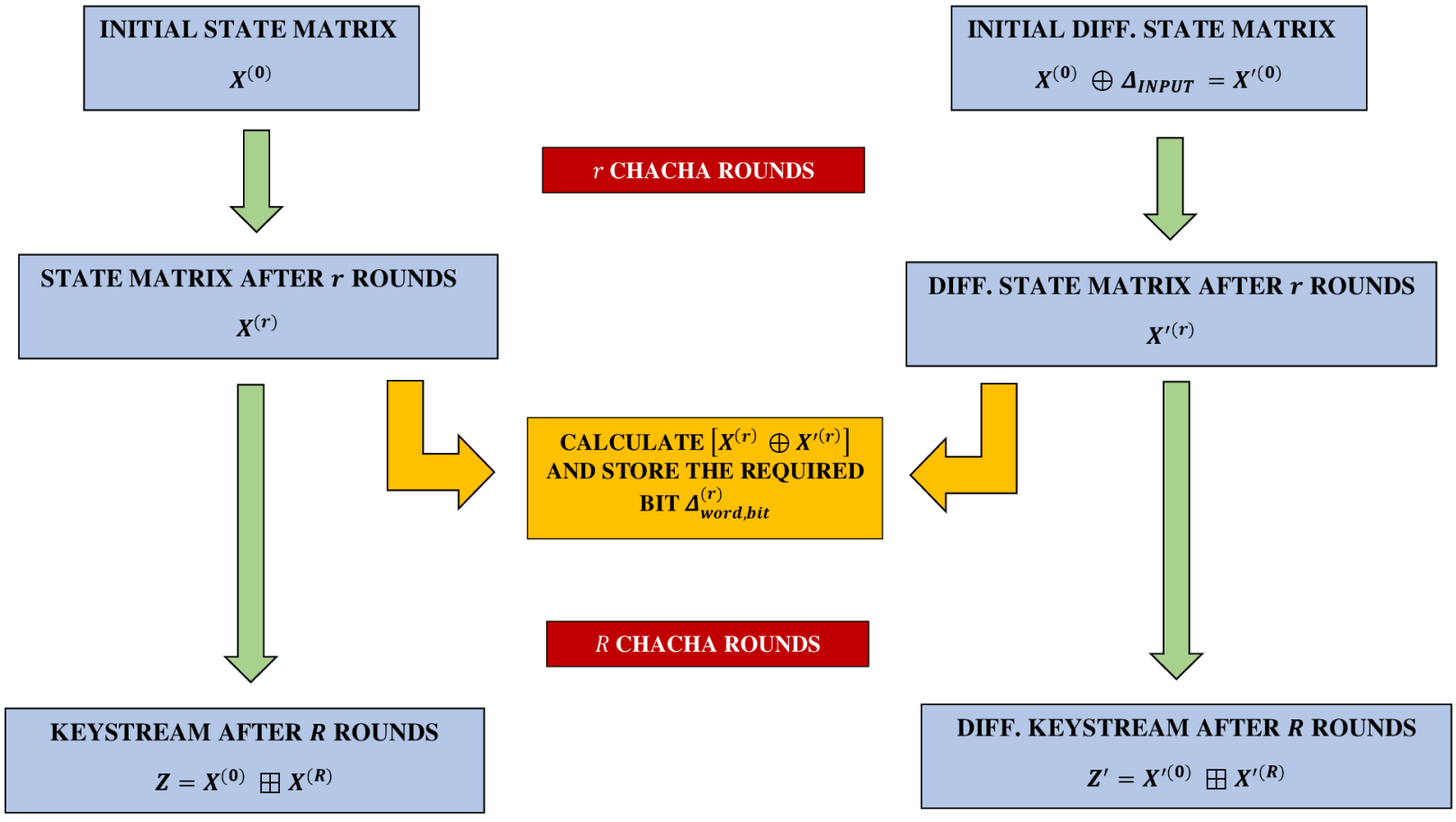
In X and X', a particular key bit position k is complemented to yield the states $\widetilde{X}$ and $\widetilde{X'}$

Next, one can reverse the states $Z - \widetilde{X}$ and $Z' - \widetilde{X'}$ by $R - r$ rounds to yield the states Y and Y' respectively.

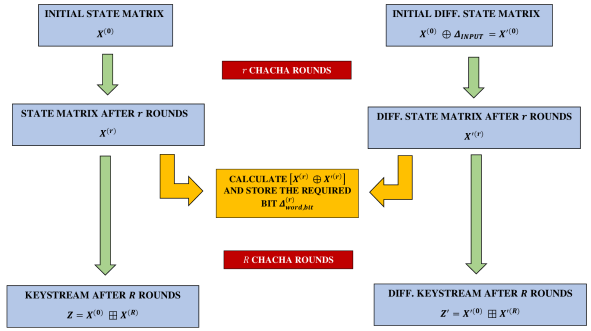
Let $\Gamma_{p,q} = Y_{p,q} \oplus Y'_{p,q}$.

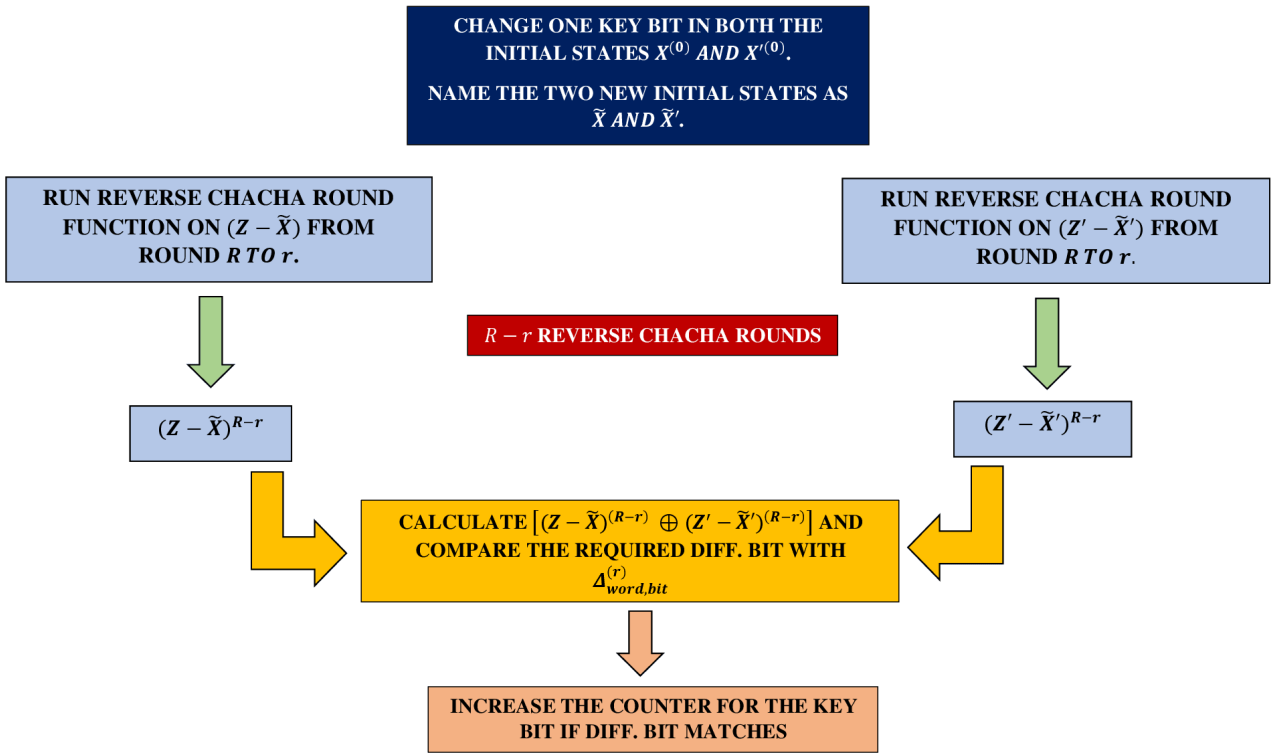
If $\Delta^{(r)}_{p,q} = \Gamma_{p,q}$ holds with high probability, then we call the key bit k a *probabilistic neutral bit* (PNB)

Attack idea


INITIAL STATE MATRIX
$X^{(0)}$
INITIAL DIFF. STATE MATRIX
$X^{(0)} \oplus \Delta_{INPUT} = X'^{(0)}$
r CHACHA ROUNDS
STATE MATRIX AFTER r ROUNDS
$X^{(r)}$
DIFF. STATE MATRIX AFTER r ROUNDS
$X'^{(r)}$
CALCULATE $[X^{(r)} \oplus X'^{(r)}]$ AND STORE THE REQUIRED BIT $\Delta^{(r)}_{word,bit}$
R CHACHA ROUNDS
KEYSTREAM AFTER R ROUNDS
$Z = X^{(0)} \boxplus X^{(R)}$
DIFF. KEYSTREAM AFTER R ROUNDS
$Z' = X'^{(0)} \boxplus X'^{(R)}$

Attack idea


INITIAL STATE MATRIX
$X^{(0)}$
INITIAL DIFF. STATE MATRIX
$X^{(0)} \oplus \Delta_{INPUT} = X'^{(0)}$
r CHACHA ROUNDS
STATE MATRIX AFTER r ROUNDS
$X^{(r)}$
DIFF. STATE MATRIX AFTER r ROUNDS
$X'^{(r)}$
CALCULATE $[X^{(r)} \oplus X'^{(r)}]$ AND STORE THE REQUIRED BIT $\Delta^{(r)}_{word,bit}$
R CHACHA ROUNDS
KEYSTREAM AFTER R ROUNDS
$Z = X^{(0)} \boxplus X^{(R)}$
DIFF. KEYSTREAM AFTER R ROUNDS
$Z' = X'^{(0)} \boxplus X'^{(R)}$


CHANGE ONE KEY BIT IN BOTH THE INITIAL STATES $X^{(0)}$ AND $X'^{(0)}$.
NAME THE TWO NEW INITIAL STATES AS $\tilde{X}$ AND $\tilde{X}'$.
RUN REVERSE CHACHA ROUND FUNCTION ON $(Z - \tilde{X})$ FROM ROUND R TO r.
RUN REVERSE CHACHA ROUND FUNCTION ON $(Z' - \tilde{X}')$ FROM ROUND R TO r.
$R - r$ REVERSE CHACHA ROUNDS
$(Z - \tilde{X})^{R-r}$
$(Z' - \tilde{X}')^{R-r}$
CALCULATE $[(Z - \tilde{X})^{(R-r)} \oplus (Z' - \tilde{X}')^{(R-r)}]$ AND COMPARE THE REQUIRED DIFF. BIT WITH $\Delta^{(r)}_{word,bit}$
INCREASE THE COUNTER FOR THE KEY BIT IF DIFF. BIT MATCHES

Improving the PNBs

- Beierle et al. received 74 using conventional method.
- We provide a three-step strategy.

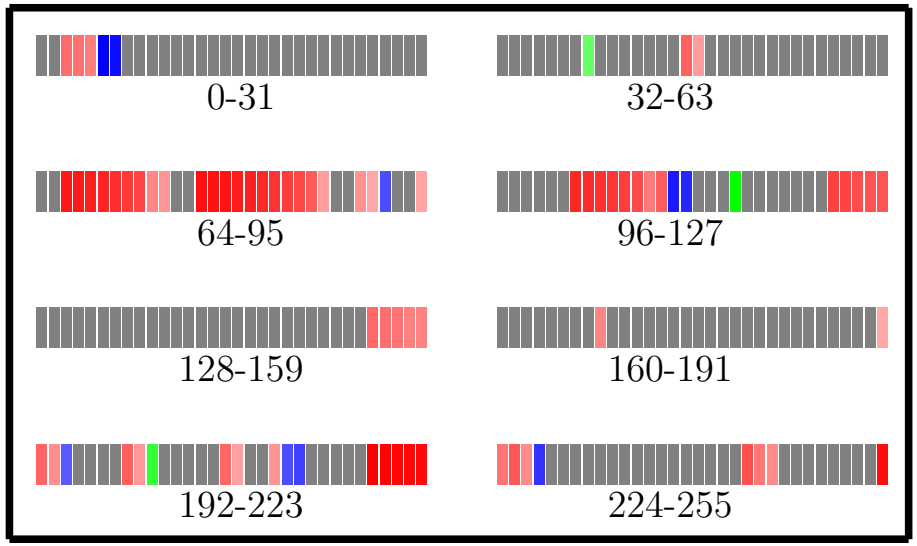


Fig. 5: : Non-PNBs, : Stage 1 PNBs, : Stage 2 PNBs, : Stage 3 PNBs

Stage 1: 68

Stage 2: 8

Stage 3: 3

Key recovery attack

Attack Idea

- Let the number of PNBs be n and therefore the number of non-PNB bits are $m = 256 - n$.
- The main idea behind the key recovery is to search these two sets separately.
- Recover non-PNBs by considering a distinguisher.

Attack complexity (Aumasson et al. FSE 2008)

1. The selected non-PNBs are correct but not detected. The probability of this event is $\Pr_{nd}$.
2. The selected non-PNBs are incorrect but give significant bias. The probability of the event is $\Pr_{fa}$.

Number of samples

$$N \approx \left(\frac{\sqrt{\alpha \log 4} + 3\sqrt{1 - \epsilon_a \epsilon_d{}^2}}{\epsilon_a \epsilon_d} \right)^2$$

gives $\Pr_{fa} \leq 2^{-\alpha}$ and $\Pr_{nd} \leq 1.3 \times 10^{-3}$.

The complexity of the attack is $2^m N + 2^{(256-\alpha)}$.

Our results

Attack on 7-round ChaCha256

- Input difference at $\Delta X_{13}^{(0)}[6]$
- Output difference at $\Delta X_{2}^{(4)}[0] \oplus \Delta X_{7}^{(4)}[7] \oplus \Delta X_{8}^{(4)}[0]$
- Bias $2^{-8.3} = 0.00317$.
- Here we will go 4-rounds forward and 3-rounds backward.

PNB set:

{219, 220, 221, 222, 223, 255, 77, 78, 79, 66, 67, 80, 68, 81, 69, 102, 82, 103, 70, 104, 83, 105, 71, 84, 106, 123, 124, 72, 85, 107, 125, 244, 126, 127, 225, 86, 109, 199, 47, 192, 207, 155, 2, 156, 3, 157, 224, 245, 108, 4, 158, 159, 168, 73, 246, 226, 193, 90, 211, 74, 200, 48, 87, 208, 95, 91, 191, 5, 6, 110, 212, 111, 227, 213, 92, 194, 115, 201, 39}.

Backward bias $\epsilon_a = 0.00057$ for these 79 PNBs
Time complexity $2^{221.95}$
Data complexity $2^{90.2}$
Memory complexity $2^{47.31}$

Attack complexities

Cipher	Rounds	Time	Data	Memory	Ref.
ChaCha128		2^{128}	0	0	Brute-force attack
		2^{107}	2^{30}	0	Aumasson et al. FSE 2008
		2^{105}	2^{28}	0	Shi et al. ICISC 2012
	6	$2^{84.39}$	$2^{38.66}$	0	[our work]
		$2^{81.58}$	$2^{43.59}$	0	[our work]
	6.5	$2^{123.04}$	$2^{66.94}$	2^{31}	[our work]
ChaCha256		2^{256}	0	0	Brute-force attack
		2^{248}	2^{27}	0	Aumasson et al. FSE 2008
		$2^{246.5}$	2^{27}	0	Shi et al. ICISC 2012
		$2^{238.9}$	2^{96}	2^{96}	Maitra DAM 2016
	7	$2^{237.7}$	2^{96}	2^{96}	Choudhuri et al. ToSC 2016
		$2^{235.22}$	-	-	Dey et al. DAM 2017
		$2^{230.86}$	$2^{48.83}$	0	Beierle et al. Crypto 2020
		$2^{221.95}$	$2^{90.20}$	$2^{47.31}$	[our work]

Table: Known full key recovery attacks.

Conclusion

Conclusion

1. Use memory to reduce time complexity.
2. Present a new technique to construct a set of PNBs.
3. Get around $2^{8.91}$ times better time complexity than the previous work for 7-round of ChaCha256.
4. Obtain $2^{23.42}$ times better complexity for 6-round ChaCha128 than the existing work
5. Report first-time cryptanalysis for 6.5-round ChaCha128.


Thank You!