Some cryptanalytic results on Lizard

Subhadeep Banik[1,4], Takanori Isobe[2], Tingting Cui[3], Jian Guo[4]

[1]LASEC, EPFL Switzerland

[2]University of Hyogo, Japan

[3]Shandong University, China

[4]NTU, Singapore

Fast Software Encryption 2018, Brugges

5th March 2018

Outline

- Introduction
- After Sprout
- Lizard

The Stream Cipher Sprout

Sprout

- Biryukov, Shamir [Asiacrypt 2001] : State size must be 1.5 to 2 times size of Secret Key.
- Radical Departure: Sprout by Armknecht and Mikhalev in FSE 2015.

  → State Size equal to size of Secret Key.

  → Avoids Generic TMD Tradeoff Attacks due to Key mixing in state update.
- Grain like structure: LFSR and NFSR of size 40 bits each.
- Much smaller in area than any known stream cipher.
- Cryptanalysis: 1. Lallemand/Naya-Plascencia [Crypto 2015],
  2. Esgin/Kara [SAC 2015],
  3. Banik [Indocrypt 2015]

The stream cipher Lizard

Lizard

- Stream cipher proposed at IACR TOSC 2017.
- The cipher supports: 120 bit secret key and 64 bit IV.

  $\rightarrow$ However claims only 80 bit security.

  $\rightarrow$ 60 bit security from distinguishing attack.
- State size of 121 bits: two NFSRs of 90 and 31 bits each.
- maximum 2^{18} keystream bits per Key-IV pair.
- Interesting key-IV mixing algorithm.

The stream cipher Lizard

Algebraic Structure

- [Phase 1: Key-IV loading:]

$$b_j^0 = \begin{cases} k_j \oplus v_j, & \text{for } j \in \{0, 1, 2, \ldots, 63\} \\ k_j, & \text{for } j \in \{64, 65, 66, \ldots, 89\} \end{cases}$$

$$s_i^0 = \begin{cases} k_{i+90}, & \text{for } i \in \{0, 1, 2, \ldots, 28\} \\ k_{119} \oplus 1, & \text{for } i = 29 \\ 1, & \text{for } i = 30 \end{cases}$$

The stream cipher Lizard

Algebraic Structure

- [Phase 2: Mixing:]

For $t = 0, 1, 2, \ldots, 127$, we compute:

$$
\begin{aligned}
b_i^{t+1} &= b_{i+1}^t, \quad \text{for} \quad i \in \{0, 1, \ldots, 88\} \\
b_{89}^{t+1} &= z_t \oplus s_0^t \oplus f_2(B^t)
\end{aligned}
$$

$$
\begin{aligned}
s_i^{t+1} &= s_{i+1}^t, \quad \text{for} \quad i \in \{0, 1, \ldots, 29\} \\
s_{30}^{t+1} &= z_t \oplus f_1(S^t)
\end{aligned}
$$

where $f_1(S^t), f_2(B^t)$ and z_t are Boolean functions.

The stream cipher Lizard

Algebraic Structure

[Phase 3: Second key Addition:] After this the 120 bit key is added to the state as follows:

$$b_j^{129} = b_j^{128} \oplus k_j, \qquad \text{for } j \in \{0, 1, 2, \ldots, 89\}$$

$$s_i^{129} = \begin{cases} s_i^{128} \oplus k_{i+90}, & \text{for } i \in \{0, 1, 2, \ldots, 29\} \\ 1, & \text{for } i = 30 \end{cases}$$

The stream cipher Lizard

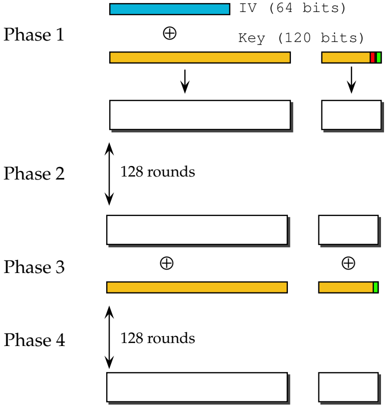
Algebraic Structure

[Phase 4: Diffusion:] For $t = 129, 130, 131, \ldots, 256$, we compute:

$$b_i^{t+1} = b_{i+1}^t, \quad \text{for} \quad i \in \{0, 1, \ldots, 88\}$$
$$b_{89}^{t+1} = s_0^t \oplus f_2(B^t)$$

$$s_i^{t+1} = s_{i+1}^t, \quad \text{for} \quad i \in \{0, 1, \ldots, 29\}$$
$$s_{30}^{t+1} = f_1(S^t)$$

The stream cipher Lizard



Note

- Phase 2 and Phase 4 are individually invertible.
- But Phase 3 makes the whole Initialization procedure non-injective
- And also inefficient to invert.

The stream cipher Lizard

Summary: We will show how to

- For the same key, find 2 IVs that generate same keystream bits.
- Find pairs K_0, IV_0 and K_1, IV_1 that generate same keystream bits.
- Distiguishing attack using slid pairs ($2^{51.5}$ IV trials)
- Key recovery attack on Lizard reduced to 223 rounds.

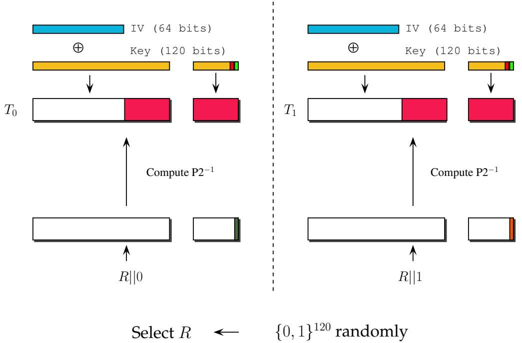
The stream cipher Lizard

Algorithm $P2^{-1}$

1. **Input**: S^t, B^t: The NFSR states at time t
2. **Output**: S^{t-1}, B^{t-1}: The NFSR states at time $t-1$

- $s \leftarrow s^t_{30}, \quad b \leftarrow b^t_{89}.$
- $B' = (b^t_0, b^t_1 \ldots, b^t_{88}), \quad S' = (s^t_0, s^t_1 \ldots, s^t_{29})$
- $\hat{z} = z(S', B')$
- $\hat{s} = s \oplus f'_1(S') \oplus \hat{z}, \quad \hat{b} = b \oplus f'_2(B') \oplus \hat{s} \oplus \hat{z}$
- $S^{t-1} \leftarrow (\hat{s}, s^t_0, s^t_1 \ldots, s^t_{29})$
- $B^{t-1} \leftarrow (\hat{b}, b^t_0, b^t_1 \ldots, b^t_{88})$
- Return S^{t-1}, B^{t-1}

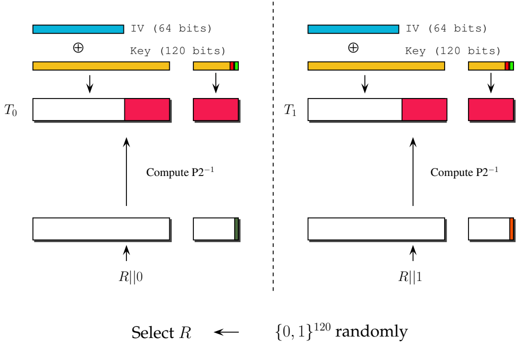
To find IV collisions for same key



Details

- If $T_0[64 \text{ to } 119] = T_1[64 \text{ to } 119]$ and $T_0[120] = T_1[120] = 1$ then we stop.
- Select $\alpha \xleftarrow{R} \{0,1\}^{64}$ randomly.
- Set $K = \alpha \mid\mid T_0[64 \text{ to } 118] \mid\mid T[119] \oplus 1$
- Set $IV_0 = \alpha \oplus T_0[0 \text{ to } 63]$ and $IV_1 = \alpha \oplus T_1[0 \text{ to } 63]$

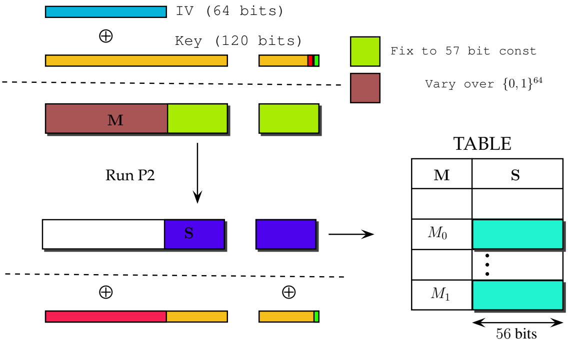
To find IV collisions for same key



Details

- A total of 58 bit conditions need to be satisfied.
- 2^{58} random trials needed.
- Any value of α can be used
- Thus gives us 2^{64} collisions !!!

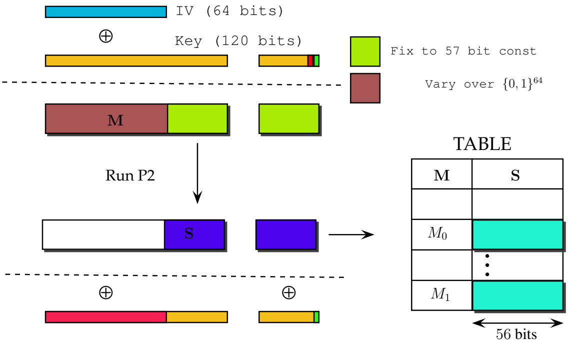
K_0, IV_1 and K_1, IV_1 that generate same keystream



Details

- 64th to 119th bits of $S_0 = F(M_0||L||1)$ and $S_1 = F(M_1||L||1)$ are equal.
- $\alpha \xleftarrow{R} \{0,1\}^{64}$
- $\Delta := S_0[0 \text{ to } 63] \oplus S_1[0 \text{ to } 63]$

K_0, IV_1 and K_1, IV_1 that generate same keystream



Details

- Set $K_0 = \alpha \mid\mid L[0 \text{ to } 54] \mid\mid L[55] \oplus 1$, Set $IV_0 = \alpha \oplus M_0$.
- Set $K_1 = \alpha \oplus \Delta \mid\mid L[0 \text{ to } 54] \mid\mid L[55] \oplus 1$, Set $IV_1 = \alpha \oplus \Delta \oplus M_1$.
- 2^{64} collisions, Complexity $= \sqrt{2^{56}} = 2^{28}$ trials.

K_0, IV_1 and K_1, IV_1 that generate same keystream

$Key - IV$	Keystream
K_0: 0000 0000 0000 0000 6850 8c64 c649 74 IV_0: 724b b286 2f5c f1b2	23f4 9770 0a91 3089 d800 ...
K_1: 1e45 1adc 2ad8 3124 6850 8c64 c649 74 IV_1: 3e18 82d1 d5ac 0376	23f4 9770 0a91 3089 d800 ...

Table: Key-IV pairs that produce identical keystream bits

Distinguishing attack

Questions

- Given a key K, how many pairs of IVs are there that generate same keystream?
- Given a key K, does there exist IVs that produce slid keystream bits ?
- If yes how many ?

Distinguishing attack

Theorem

Let p be an integer greater than zero. Then, for every 120-bit secret key K,

1. *There exists around 2^6* IV Collisions *on average,*
2. *There exists around 2^7 IV pairs (IV_0, IV_1) on average, such that the key-IV pairs K, IV_0 and K, IV_1 produce exactly p-bit shifted keystream sequences.*

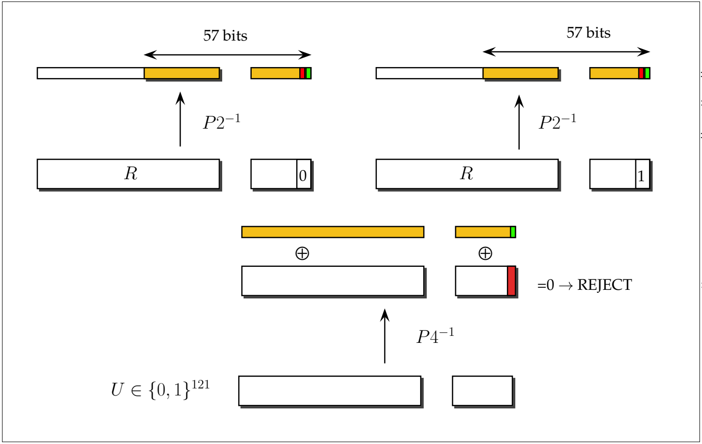
Proof is by construction

Let $G : \mathbb{F}_2^{121} \rightarrow \mathbb{F}_2^{121}$ be the function that maps the input of Phase 4 to its output

Input: A 121 bit string U, a 120-bit key K, Output: The values 0/1/2.
Subroutine $\theta(U, K)$

1. Compute $\hat{U} = (K||0) \oplus G^{-1}(U)$.
2. If $\hat{U}[120] = 0$ then abort and return 0.
3. Compute $U_0' = F^{-1}(\hat{U}[0 \text{ to } 119] \mid\mid 0)$
4. Compute $U_1' = F^{-1}(\hat{U}[0 \text{ to } 119] \mid\mid 1)$
5. Set $r \leftarrow 0$.
6. If $U_0'[64 \text{ to } 120] = K[64 \text{ to } 118] \mid\mid K[119] \oplus 1 \mid\mid 1$, increment $r \leftarrow r + 1$.
7. If $U_1'[64 \text{ to } 120] = K[64 \text{ to } 118] \mid\mid K[119] \oplus 1 \mid\mid 1$, increment $r \leftarrow r + 1$.
8. Return r.

Proof is by construction



Function $\theta(U)$

= 2 if both states

= 1 if only state

= 0 otherwise

= 0 here

Proof

- #IV collision is the no. of times the Subroutine returns 2 over 2^{121} values of U
- 115 bit conditions need to be satisfied: $2^{121-115} = 2^6$

Slid pairs

Slid pairs

- Let g be the function that maps one Phase 4 iteration.
- Number of times $\theta(U,K)$ and $\theta(g^p(U),K)$ both return non-zero.

$$\begin{aligned}
\mathsf{Pr}[\theta(U,K)\neq 0] &= \mathsf{Pr}[\theta(U,K)=2 \mid A]\cdot \mathsf{Pr}[A] + \mathsf{Pr}[\theta(U,K)\neq 0 \mid A^c]\cdot \mathsf{Pr}[A^c] \\
&= 0\cdot \frac{1}{2} + \mathsf{Pr}[B\vee C \mid A^c]\cdot \frac{1}{2} \\
&= \frac{1}{2}\cdot (\mathsf{Pr}[B \mid A^c] + \mathsf{Pr}[C \mid A^c]) \\
&= \frac{1}{2}\cdot (2^{-57}+2^{-57}) = 2^{-57}
\end{aligned}$$

- Assuming the distributions are i.i.d total probability is 2^{-114}.
- #Slid pairs $= 2^{121-114} = 2^7$

Constructing Distinguisher

Using Slid pairs

- Generate 2^{18} keystream bits $[z_0,\ z_1,\ldots,z_{2^{18}-1}]$ for the unknown key K and some randomly generated Initial Vector IV.

- For $i = 0$ to $2^{18} - 121$

  $\rightarrow$ Store $[z_i, z_{i+1},\ldots,z_{i+120}]$ in a Hash table along with the IV.

  $\rightarrow$ Continue the above steps with more randomly generated IVs

  $\rightarrow$ Stop either IV Collision or p-bit shifted keystream (for $1 \leq p \leq 2^{18} - 121$).

Constructing Distinguisher

Complexity

- Space of Initial Vectors as an undirected Graph $G = (W, E)$, all the IVs are nodes.

- An edge $(IV_1, IV_2) \in E$ iff (K, IV_1) and (K, IV_2) produce either an IV collision or p-bit shifted keystream(for $1 \leq p \leq 2^{18} - 80$).

- Cardinality of edge-set E is expected to be $(2^{18} - 121) \cdot 2^7 + 2^6 \approx 2^{25}$.

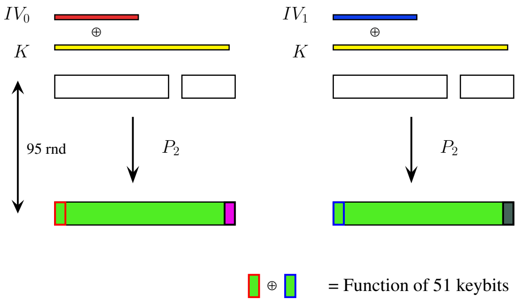
- By Birthday bound $\binom{N}{2} \cdot 2^{25} = \binom{2^{64}}{2} \to N \approx 2^{51.5}$.

Impossible Collision attack

Similar to Impossible Differential attack

- 2^6 IV collisions per key on average.
- In phase 1, attacker exhausts entire codebook of IVs (2^{64})
- Gets 2^6 IV pairs which produce same keystream.

Impossible Collision attack



Details

- The algebraic expression of $B^{95}[0] \oplus \hat{B}^{95}[0]$ has 51 key bits.
- Possible to search over smaller space,

Impossible Collision attack

Impossible Collision Attack

1. Given around 2^6 colliding pair of IVs.
2. For each guess of the 51-bit key

   $\rightarrow$ Compute $\delta = B^{95}[0] \oplus \hat{B}^{95}[0]$ for the next colliding IV pair.

   $\rightarrow$ If $\delta = 1$, reject the key and start with another key guess

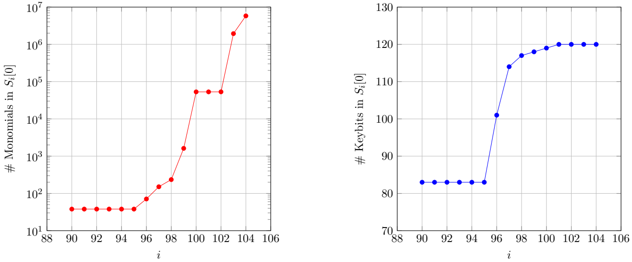
   $\rightarrow$ **Else** go to the previous step and try out another IV pair.

Impossible Collision attack

Complexity of Attack

- Start with 2^{64} encryptions to find all the colliding pairs.
- The filtering algorithm for 2^{51} keys takes at most 2^{6} computations of δ per key guess
- So 2^{57} calculations of δ.
- Brute force search over the remaining 69 keybits.

Impossible Collision attack



(a) (A) (b) (B)

Figure: Plot of (A) # Monomials, (B) # Keybits in $B^i[0]$

More rounds

- Can be extended to 3 more rounds...

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

THANK YOU