ESC 2017, Canach, January 16.

On Stream Ciphers with Small State

Willi Meier
joint work with Matthias Hamann, Matthias Krause
(University of Mannheim)
Bin Zhang (Chinese Academy of Sciences, Beijing)

University of Applied Sciences Northwestern Switzerland
School of Engineering

Overview

- Preliminaries
- Stream Ciphers with Small State
- LIZARD design
- A Fast Correlation Attack on Stream Cipher Fruit
- Conclusions

Preliminaries

Common knowledge

Rule: State at least twice key size (or security parameter).

Due to TMDTO state recovery. Based on birthday paradox.

Applies mainly if state update function G is key independent.

eSTREAM finalist stream ciphers obey this rule and have key independent update functions.

Birthday based distinguisher on key stream?

Can work even for key dependent update:

A Note on Distinguishing Attacks (Englund-Hell-Johansson, eSTREAM publication, 2007)

Stream Ciphers with Small State

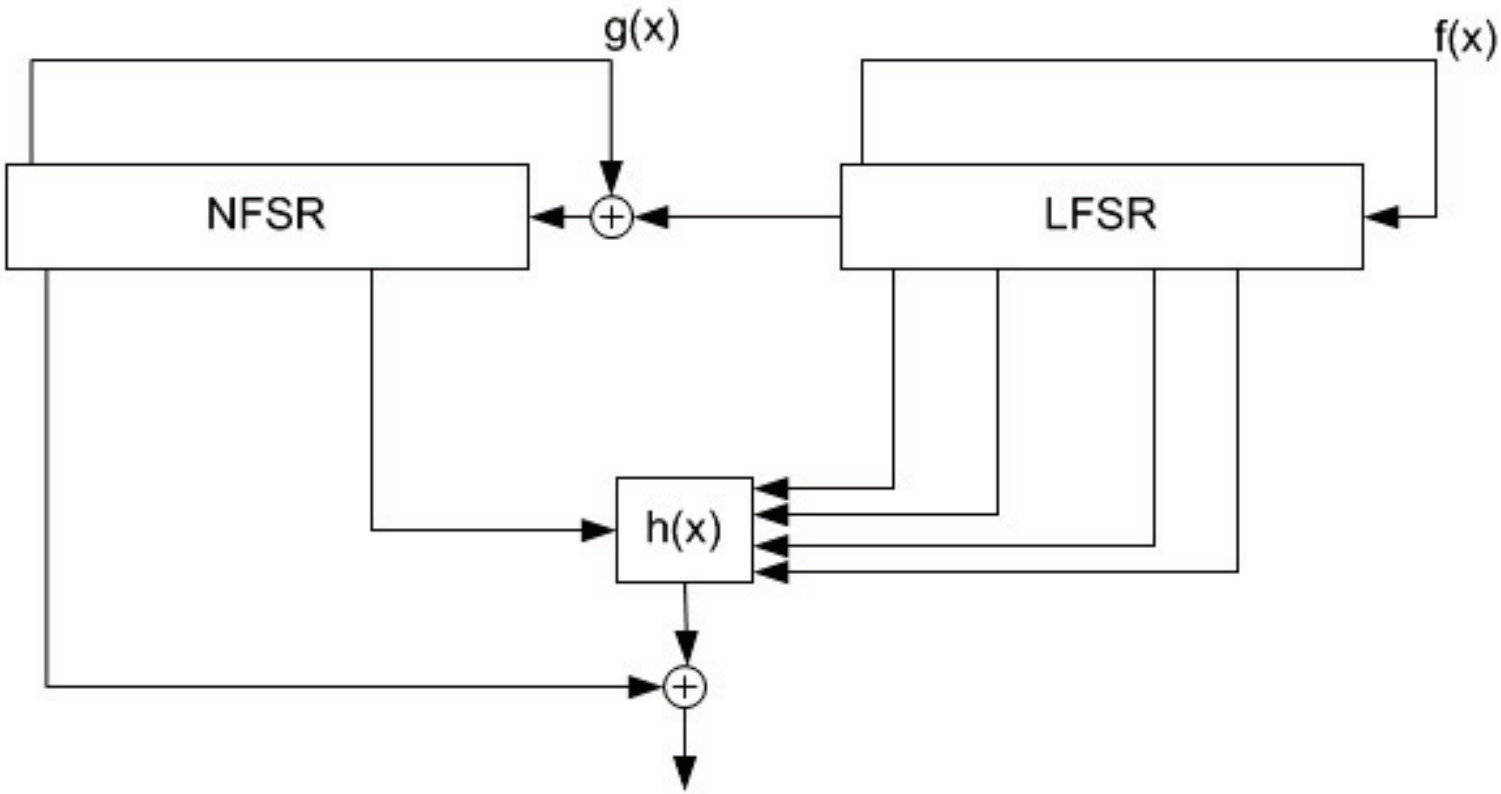
Argue: Allow distinguisher of key stream to some extent (block ciphers have birthday distinguishers as well!)

Goals:

- Lower area and power consumption than for existing designs.
- Fast access to key in non-volatile memory (stream cipher Sprout).
- Understanding security achievable by stream ciphers with small state.

Stream Ciphers with Small State

Grain v1: State size 160 bits, key size 80 bits.



Stream Ciphers with Small State

For 80 bit security, can we go lower than 160 bit state size?

One idea: Make state update key-dependent.

Cannot prevent distinguishers of keystream, but possibly key recovery.

Sprout: State size only 80 bits. Modelled on stream cipher Grain v1.

Has been broken by several methods, including TMDTO and use of k-normality of Boolean functions, by Bin Zhang (Asiacrypt 2015).

Fruit (on eprint): A tweak of Sprout stream cipher.

LIZARD Design

LIZARD modelled on Grain v1 as well.

State update independent of key, but initialization mechanism so that key recovery is provably prevented.

Security:

- Against key recovery: 2^{80}
- Complexity of generic distinguisher: 2^{60}

Use in packet mode: No (severe) TMDTO distinguishers. 16% reduced power consumption over Grain v1.

LIZARD comes with a security proof against key recovery based on generic TMDTO.

LIZARD Design

Beyond-the-birthday-bound security level of $\frac{2}{3}n$ w.r.t. generic TMDTO's aiming at key recovery.

Security proof: Theoretical work by Hamann and Krause. Based on formal ideal primitive model.

Information-theoretic $\frac{2}{3}n$ security bound, which is tight.

LIZARD Design

Packet length of LIZARD limited to 2^{18} bits:

Lower bound for security complexity theoretic, in spirit of work on security of Even-Mansour ciphers.

Gives only asymptotic bounds, but suggests that instantiation by LIZARD is meaningful.

Packet length chosen conservatively, to fit in application scenarios.

LIZARD Design

Design of LIZARD uses well established components.

Differences to Grain v1:

- Smaller state size (121 compared to 160 bits).
- Key size: 120 bit (rather than 80 bits): necessary assumption for security proof.
- Key is introduced not only once, but twice in initialization.
- Quite different output function: Similar to FLIP stream cipher, uses many inputs.
- Both register feedbacks nonlinear.

LIZARD Design

Components:

Two NFSR's, NFSR1 and NFSR2, of length 31 and 90, resp.

NFSR1 has guaranteed period $2^{31} - 1$ (from ACHTERBAHN stream cipher).

NFSR2 keeps same cryptographic properties as NFSR in Grain-128a.

LIZARD Design

Output function:

Depends on 53 inputs, shared carefully between NFSR1 and NFSR2:

If one of the registers asssumed to be known, the remaining function satisfies cryptographic properties well.

In particular, remaining function still nonlinear.

In contrast to Grain v1, where the output function becomes linear in NFSR state bits if LFSR state assumed to be known.

As in FLIP, output function is a sum of linear, quadratic and triangular functions.

Fulfills all known design criteria for output function of a stream cipher.

LIZARD Design

State initialization:

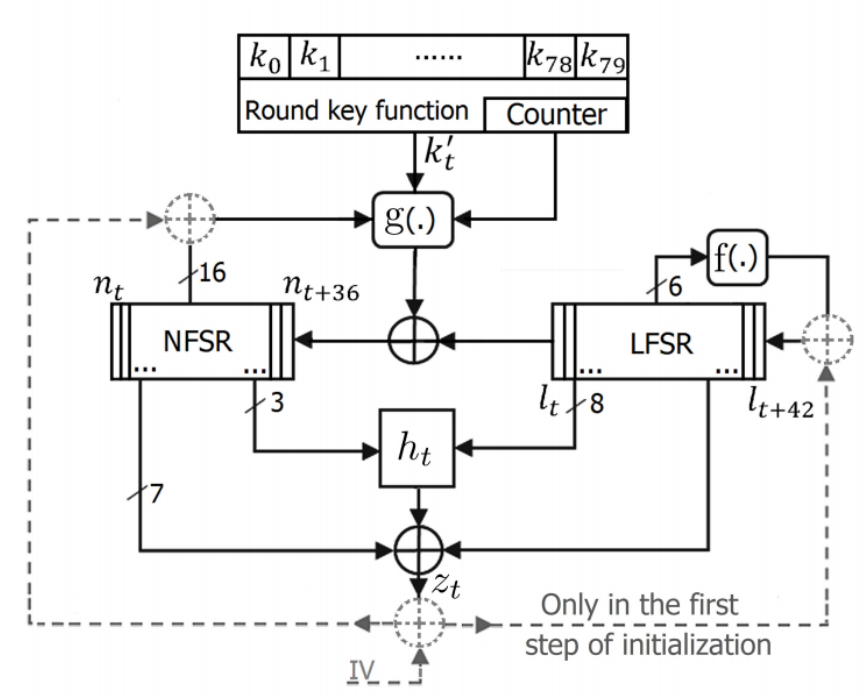
Consists of 4 phases:

- Key and IV loading (IV of 64 bits)
- Grain-like mixing
- Second key addition (hardening)
- Final diffusion

All zero state prevented.

Fruit (description)

Fruit has similar global structure as Grain.



In addition two counters and a round key function are used.

Fruit (description)

- LFSR of length 43, denoted by $L^t = (l_t, ..., l_{t+42})$ at time t.
- NFSR of length 37, denoted by $N^t = (n_t, ..., n_{t+36})$.
- 80-bit key, generate round key bit k'_t at time t.
- k'_t sparse quadratic function of 6 key bits.

Fruit (description)

- 7-bit counter $C_r = (c_t^0, ..., c_t^6)$ for round key function (kept secret).
- 8-bit counter $C_c = (c_t^7, ..., c_t^{14}$ for NFSR-update.
- NFSR updating: $n_{t+37} = k'_t \oplus l_t \oplus c_t^{10} \oplus g(N^t)$.
- Function g in NFSR update has 64 best linear approximations with bias $\epsilon = 2^{-4.6}$.
- Fruit allows for 2^{43} key stream bits with same key and IV.

A preliminary observation

- Given the internal state (L^t, N^t) at time t, the keystream bit z_t is generated as

$$z_t = n_{t+1} l_{t+15} \oplus l_{t+1} l_{t+22} \oplus n_{t+35} l_{t+27} \oplus n_{t+33} l_{t+11} \oplus l_{t+6} l_{t+33} l_{t+42} \oplus l_{t+38} \oplus n_t \oplus n_{t+7} \oplus n_{t+13} \oplus n_{t+19} \oplus n_{t+24} \oplus n_{t+29} \oplus n_{t+36}$$

  i.e., the restriction of the output function on L^t is a linear Boolean function on the variables from N^t.

- If the initial state $L^0 = (l_0, l_1, ..., l_{42})$ of the LFSR is known, the filter generator of Fruit can be interpreted as a linearly filtered NFSR involving the secret key information with a known cycle of 128.

A preliminary observation

- Use the method by Berbain-Gilbert-Joux (SAC 2008): Each NFSR state variable n_i can be expressed as a linear combination of the initial state variables $N^0 = (n_0, n_1, ..., n_{36})$ and of some keystream bits using the output function of Fruit.

$$
\begin{aligned}
n_{37} &= z_1 \oplus l_{16} n_2 \oplus l_{12} n_{34} \oplus l_{28} n_{36} \oplus n_1 \oplus n_8 \oplus n_{14} \oplus n_{20} \oplus n_{25} \\
&\quad \oplus n_{30} \oplus l_{39} \oplus l_2 l_{23} \oplus l_7 l_{34} l_{43} \\
n_{38} &= z_2 \oplus l_{17} n_3 \oplus l_{13} n_{35} \oplus l_{29} n_{37} \oplus n_2 \oplus n_9 \oplus n_{15} \oplus n_{21} \oplus n_{26} \\
&\quad \oplus n_{31} \oplus l_{40} \oplus l_3 l_{24} \oplus l_8 l_{35} l_{44} \\
&\quad \vdots
\end{aligned}
$$

The variable n_{38} depends on n_{37}.

A preliminary observation

- The variable n_{38} depends on n_{37}. By a simple substitution, we get

$$\begin{aligned} n_{38} = {} & z_2 \oplus l_{29} z_1 \oplus l_{29} l_{16} n_2 \oplus l_{17} n_3 \oplus l_{29} l_{12} n_{34} \oplus l_{13} n_{35} \oplus l_{29} l_{28} n_{36} \\ & \oplus l_{29} n_1 \oplus n_2 \oplus l_{29} n_8 \oplus n_9 \oplus l_{29} n_{14} \oplus n_{15} \oplus l_{29} n_{20} \oplus n_{21} \\ & \oplus l_{29} n_{25} \oplus n_{26} \oplus l_{29} n_{30} \oplus n_{31} \oplus l_{29}(l_{39} \oplus l_2 l_{23} \oplus l_7 l_{34} l_{43}) \\ & \oplus l_{40} \oplus l_3 l_{24} \oplus l_8 l_{35} l_{44} \end{aligned}$$

  Thus, n_{38} is expressed linearly as a combination of N^0 and of the keystream bits z_1, z_2, under the condition that L^0 is known.
- Continually, n_{37}, n_{38},n_{39}, n_{40},...,n_{37+N-1} can be expressed as a linear combination of N^0 and of the keystream bits $z_1, z_2, \ldots, z_N$.
- The cost is $\mathcal{O}(2^{43} \cdot 37 \cdot N)$ to express N consecutive NFSR variables through the induction process.
- Denote by lc_t the linear expression associated with n_t.

Fast correlation attack on Fruit

Fast correlation attack on NFSR part.

- Preprocessing (parity checks)
- Processing (solving probabilistic linear equations for state bits)

Parity-checks

- Express N NFSR variables n_{37+i} for $i = 0, 1, ..., N-1$.
- Then use the NFSR update function g.
  - Denote by $\mathcal{A}_g^{(\vec{a}^k, b_k)}(\cdot)$ the linear approximation for $g(\cdot)$ and by ϵ its bias.
  - Write $\mathcal{A}_g^{(\vec{a}^k, b_k)}(N^i) = (n_i, n_{1+i}, ..., n_{36+i}) \cdot (\vec{a}^k)^T \oplus b_k$.
  - 64 best linear approximations for g having the same bias $\epsilon = 2^{-4.6}$.
- With probability $\frac{1}{2} + \epsilon$, we have

$$n_{37+i} = k'_i \oplus c_i^{10} \oplus l_i \oplus (n_i, n_{1+i}, ..., n_{36+i}) \cdot (\vec{a}^k)^T \oplus b_k.$$

- Replace n_t by its linear expression lc_t, with the same probability

$$lc_{37+i} \oplus (lc_i, lc_{1+i}, ..., lc_{36+i}) \cdot (\vec{a}^k)^T \oplus b_k \oplus l_i = k'_i \oplus c_i^{10}.$$

- c_i^{10} has a cycle of length 32, k'_i has a cycle of length 128, $\Rightarrow k'_i \oplus c_i^{10}$ has a cycle of length 128.

Parity-checks

- $k'_i \oplus c_i^{10}$ has a cycle of length 128., i.e.,

  $$k'_{i+128j} \oplus c_{i+128j}^{10} = k'_i \oplus c_i^{10},\ i = 0, 1, ..., 127, j = 0, 1, ...$$

  By choosing $i = 0$, we obtain

  $$lc_{37+128j} \oplus (lc_{128j}, ..., lc_{36+128j}) \cdot (\vec{a}^k)^T \oplus b_k \oplus l_{128j} = k'_0 \oplus c_0^{10}.$$

- Let $N = 128(m-1)+1$, we can obtain $m' \triangleq 64m$ equations of the above form, for $k = 1, ..., 64$, $j = 0, 1, ..., m-1$.

Parity-checks

- Write these equations in another form (separate NFSR state bits from known information, like keystream bits z, and values coming from LFSR):

$$\bar{Z}_{k,128j} \oplus (n_0, n_1, ..., n_{36}) \cdot (\vec{\bar{u}}^k_{128j})^T \oplus \bar{d}_{k,128j} \oplus b_k = k'_0 \oplus c_0^{10}$$

- Let $Z_{k,j} \triangleq \bar{Z}_{k,128j}$, $\vec{u}^k_j \triangleq \vec{\bar{u}}^k_{128j}$, and $d_{k,j} \triangleq \bar{d}_{k,128j}$, we further obtain

$$Z_{k,j} \oplus (n_0, n_1, ..., n_{36}) \cdot (\vec{u}^k_j)^T \oplus d_{k,j} \oplus b_k = k'_0 \oplus c_0^{10} \oplus e_{k,j},$$

where $e_{k,j}$ is the noise introduced from the linear approximation $\mathcal{A}_g^{(\vec{a}^k, b_k)}$ for g, and $\Pr(e_{k,j} = 0) = 1/2 + \epsilon$.

Parity-checks

- The above system of equations can be equivalently written as

$$\vec{Z}^k = (n_0, n_1, ..., n_{36}) \cdot \mathbf{U}^k \oplus \vec{d}^k \oplus b_k \cdot \vec{1} \oplus (k'_0 \oplus c_0^{10}) \cdot \vec{1} \oplus \vec{e}^k,$$

for $k = 1, ..., 64$, where

$$\vec{Z}^k = (Z_{k,0}, Z_{k,1}, ..., Z_{k,m-1}),\ \mathbf{U}^k = [(\vec{u}_0^k)^T, (\vec{u}_1^k)^T, ..., (\vec{u}_{m-1}^k)^T]$$
$$\vec{d}^k = (d_{k,0}, d_{k,1}, ..., d_{k,m-1}),\ \vec{e}^k = (e_{k,0}, e_{k,1}, ..., e_{k,m-1})$$

- Putting all the $m' = 64m$ equations in a single system,

$$(\vec{Z}^1, ..., \vec{Z}^{64}) = (n_0, n_1, ..., n_{36}) \cdot \left[\mathbf{U}^1, ..., \mathbf{U}^{64}\right] \oplus (\vec{d}^1, \vec{d}^2, ..., \vec{d}^{64}) \oplus (b_1 \cdot \vec{1}, ..., b_{64} \cdot \vec{1}) \oplus (k'_0 \oplus c_0^{10}) \cdot \vec{1} \oplus (\vec{e}^1, ..., \vec{e}^{64}).$$

or equivalently
$\vec{Z} \oplus \vec{b} = (n_0, n_1, ..., n_{36}) \cdot \mathbf{U} \oplus \vec{d} \oplus \vec{e} \oplus (k'_0 \oplus c_0^{10}) \cdot \vec{1}.$

Outline of the Attack

- Suppose $z_1, z_2, ..., z_N$ are available, $N = 128(m-1)+1$.
- Exhaustively search over the LFSR initial state . For each one, express the NFSR variables n_{37+i} $(i = 0, ..., N-1)$, and derive a system of m' equations, i.e., $\vec{Z} \oplus \vec{b} = (n_0, n_1, ..., n_{36}) \cdot \mathbf{U} \oplus \vec{d} \oplus \vec{e} \oplus (k'_0 \oplus c_0^{10}) \cdot \vec{1}$.
  - $\vec{Z}$ is obtained from the given keystream $z_1, z_2, ..., z_N$.
  - $\vec{b}$ is a constant vector determined by the 64 linear approximations for g.
  - $\mathbf{U}$ and $\vec{d}$ are closely related with the LFSR state bits.
  - $\vec{e}$ is the noise vector with the bias $\epsilon = 2^{-4.6}$.
- Time complexity $2^{43} \cdot 37 \cdot N + 2^{43} \cdot m'$.

Outline of the Attack

- a divide-and-conquer attack
  - restore the initial state of both the LFSR and NFSR, i.e., L^0 and N^0,
  - recover the round key bits within one cycle (128-bit),
  - recover the original 80-bit secret key (in guess-and-determine manner).
- After guessing the initial state L^0 of the LFSR, divide the NFSR initial state $N^0 = (n_0, n_1, ..., n_{36})$ into two parts as follows.

$$(\underbrace{n_0, n_1, ..., n_{x-1}}_{x}, \underbrace{n_x, n_{x+1} ..., n_{36}}_{(y=)37-x})$$

Outline of the Attack

- How to recover the LFSR and NFSR initial state?
  - Exhaustively search over the LFSR initial states, and pass them to the next steps.
  - Then proceed to determine the first part of the initial state of the NFSR (x-bit length) conditioned on both the LFSR initial state candidates and the keystream bits.
  - Finally determine the last part of the initial state of the NFSR (y-bit length) conditioned on the LFSR initial state candidates, the first part of the initial state of the NFSR and the keystream bits.

Outline of the Attack Process: Preprocessing Stage

$$Z_i \oplus b'_i = (n_0, n_1, ..., n_{36}) \cdot \vec{u}_i^T \oplus d_i \oplus e_i \oplus (k'_0 \oplus c_0^{10}),\ i = 1, 2, ..., m'.$$

- Regard the column vectors $\vec{u}_i^T$ as random vectors.
- Look for pairs $(\vec{u}_{i_1}^T, \vec{u}_{i_2}^T)$ satisfying $\mathrm{Low}_y(\vec{u}_{i_1}^T \oplus \vec{u}_{i_2}^T) = (0, ..., 0)^T$.
  - Sort-and-merge procedure. First m' vectors $\vec{u}_i^T$ are sorted into 2^y equivalence classes according to their values on the most significant y positions, thus any two vectors in the same equivalence class will have the same value on these positions.
  - Next, look at each pair of vectors $(\vec{u}_{i_1}^T, \vec{u}_{i_2}^T)$ in each equivalence class, deriving that $\mathrm{Low}_y(\vec{u}_{i_1}^T \oplus \vec{u}_{i_2}^T) = (0, ..., 0)^T$.
- The expected number of pairs is $\Omega = \binom{m'}{2} \cdot 2^{-y} \approx m'^2 \cdot 2^{-(y+1)}$.
- This can be finished in time $2^{43} \cdot (m' + \Omega)$.

Outline of the Attack Process: Preprocessing Stage

- Denote the indices of the t-th pair by $(i_1^{(t)}, i_2^{(t)})$, $t = 1, 2, ..., \Omega$.
- Let $\vec{u}^T_{i_1^{(t)}} \oplus \vec{u}^T_{i_2^{(t)}} = (a_0^{(t)}, a_1^{(t)}, ..., a_{x-1}^{(t)}, 0, ..., 0)^T$, Then we have

$$(Z_{i_1^{(t)}} \oplus Z_{i_2^{(t)}}) \oplus (b'_{i_1^{(t)}} \oplus b'_{i_2^{(t)}}) = a_0^{(t)} n_0 \oplus a_1^{(t)} n_1 \oplus ... \oplus a_{x-1}^{(t)} n_{x-1} \oplus (d_{i_1^{(t)}} \oplus d_{i_2^{(t)}}) \oplus (e_{i_1^{(t)}} \oplus e_{i_2^{(t)}}).$$

- Let $\mathcal{Z}_t = Z_{i_1^{(t)}} \oplus Z_{i_2^{(t)}}$, $\mathcal{B}_t = b'_{i_1^{(t)}} \oplus b'_{i_2^{(t)}}$, $\mathcal{D}_t = d_{i_1^{(t)}} \oplus d_{i_2^{(t)}}$, $\mathcal{E}_t = e_{i_1^{(t)}} \oplus e_{i_2^{(t)}}$, and $\vec{\mathcal{U}}_t = \text{High}_x \left(\vec{u}_{i_1^{(t)}} \oplus \vec{u}_{i_2^{(t)}} \right)$, rewrite it as

$$\mathcal{Z}_t \oplus \mathcal{B}_t = (n_0, n_1, ..., n_{x-1}) \cdot \vec{\mathcal{U}}_t^T \oplus \mathcal{D}_t \oplus \mathcal{E}_t, \ t = 1, 2, ..., \Omega$$

Recovery of the Initial State of the LFSR

$$\mathcal{Z}_t \oplus \mathcal{B}_t = (n_0, n_1, ..., n_{x-1}) \cdot \vec{\mathcal{U}}_t^T \oplus \mathcal{D}_t \oplus \mathcal{E}_t, \ t = 1, 2, ..., \Omega$$

- If we exhaustively search all the possible values of $(l_0, l_1, ..., l_{42})$ and $(n_0, n_1, ..., n_{x-1})$, then from the above, we have

$$\mathcal{Z}_t \oplus (n'_0, n'_1, ..., n'_{x-1}) \cdot \vec{\mathcal{U}}_t'^T \oplus \mathcal{D}'_t \oplus \mathcal{B}_t$$
$$= (n_0, n_1, ..., n_{x-1}) \cdot \vec{\mathcal{U}}_t^T \oplus (n'_0, n'_1, ..., n'_{x-1}) \cdot \vec{\mathcal{U}}_t'^T \oplus \mathcal{D}_t \oplus \mathcal{D}'_t \oplus \mathcal{E}_t,$$

where $(n'_0, n'_1, ..., n'_{x-1})$ is the guessed value of the first x-bit of the NFSR, and $\vec{\mathcal{U}}'_t$, $\mathcal{D}'_t$ are obtained from the guessed value $(l'_0, l'_1, ..., l'_{42})$ of the LFSR.

Recovery of the Initial State of the LFSR

$$\Delta(i_1^{(t)}, i_2^{(t)}) =$$
$$(n_0, n_1, ..., n_{x-1}) \cdot \vec{\mathcal{U}}_t^T \oplus (n'_0, n'_1, ..., n'_{x-1}) \cdot \vec{\mathcal{U}}_t'^T \oplus \mathcal{D}_t \oplus \mathcal{D}'_t \oplus \mathcal{E}_t.$$

Need to discuss the distribution of Δ in 4 situations.

- **Case 1.** If both $(l_0, l_1, ..., l_{42})$ and $(n_0, n_1, ..., n_{x-1})$ are correctly guessed, we have $\vec{\mathcal{U}}'_t = \vec{\mathcal{U}}_t$, $\mathcal{D}'_t = \mathcal{D}_t$, and $\Delta(i_1^{(t)}, i_2^{(t)}) = \mathcal{E}_t$. Since $\mathcal{E}_t = e_{i_1^{(t)}} \oplus e_{i_2^{(t)}}$, and $e_{i_1^{(t)}}$, $e_{i_2^{(t)}}$ are independent random variables, from the piling-up lemma, we have

$$\Pr(\Delta = 0) = \frac{1}{2} + 2\epsilon^2 \triangleq \frac{1}{2}(1 + \epsilon_f),$$

where $\epsilon = 2^{-4.6}$ and $\epsilon_f = 4\epsilon^2 = 2^{-7.2}$.

Recovery of the Initial State of the LFSR

Remaining steps (omitted here):

- Discuss other cases
- Distinguish right and wrong candidate states: Compute sums of Δ's using Fast Walsh Transform.
- Determine last part of initial state of NFSR.
- Complexity analysis for suitable parameters.
- Check whether a state candidate is correct, and if so, further recover the 128 round key bits within one cycle.

Complexity Analysis

Complexity analysis done in general.

A set of suitable parameters are chosen as follows:

- $x = 21$, $y = 16$, $\Omega = 2^{21.30}$ pairs.
- Since $\Omega = \binom{m'}{2} \cdot 2^{-y} \approx m'^2 \cdot 2^{-(y+1)}$, we get data complexity

  $m' = \sqrt{\Omega \cdot 2^{y+1}} = 2^{19.65}$, and $N = 128(\frac{m'}{64} - 1) + 1 = 2^{20.15}$.

- Preprocessing time complexity
  $= 2^{43}(37N + 2m' + \Omega)) \approx 2^{69}$;
- Processing time complexity: About 2^{69} basic operations.

  Resulting complexity for state and round key recovery about 2^{69} basic operations.

Conclusions

- Presented LIZARD, a new design of a small state stream cipher.
- Comes with a provable security against generic TMDTO state recovery attacks.
- Cryptanalysis of Fruit using correlation attack: New design crieria for Grain-like small state stream ciphers:
  - Output function: Strong even when one of the registers is known
  - Feedback function of NFSR: Of high enough nonlinearity (to prevent good linear approximations).
- Underlines necessity of strong output function for stream ciphers with small state.
- More analysis on such stream ciphers necessary for understanding achievable security bounds in practice.