Separable Statistics and Multidimensional Linear Cryptanalysis

Stian Fauskanger   Igor Semaev

FFI, Norway
Univ. of Bergen, Norway

28 March 2019, FSE, Paris

Briefly

- Matsui's Linear Cryptanalysis is based on the distribution of
$$x_1 \oplus \ldots \oplus x_s \oplus y_1 \oplus \ldots \oplus y_t,$$
where $X = x_1, \ldots, x_s$ some plain-text bits and $Y = y_1, \ldots, y_t$ some cipher-text bits in Algorithm1
- In Algorithm2 the bits X are inputs to the second round, and Y to the last round
- Starting point in **our work**: method for computing joint distribution of $(X, Y) = (x_1, \ldots, x_s, y_1, \ldots, y_t)$
- The distributions (both Matsui's and our's) are approximate
- They depend on small sets of the cipher key-bits or linear combinations
- Algorithm2-like cryptanalysis is then applied

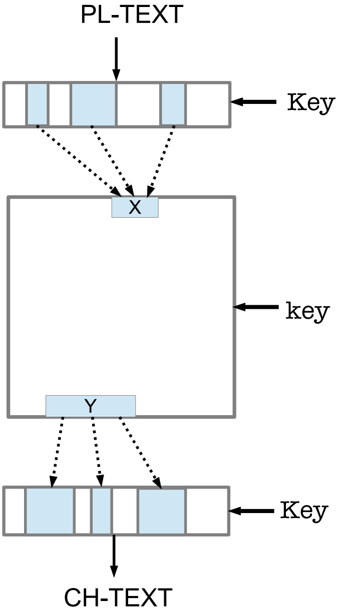
Outline

- Matsui's Algorithm2 and LLR statistic
- New Statistic Construction
- Optimisation Problem and Search Algorithm
- Implementation for 16-round DES
- Multidimensional Distributions in Feistel Ciphers
- Conclusions

Outline

- Matsui's Algorithm2 and LLR statistic
-
-
-
-
-

Round Cipher Cryptanalysis with Algorithm2



Logarithmic Likelihood Ratio(LLR) Statistic

- To distinguish two distributions with densities $P(x), Q(x)$
- By independent observations $\nu_1, .., \nu_n$
- Most powerful test(Neyman-Pearson lemma):
- Accept $P(x)$ if

$$\sum_{i=1}^{n} \ln \frac{P(\nu_i)}{Q(\nu_i)} > threshold$$

- Left hand side function is called LLR statistic

Algorithm2 Cryptanalisis with LLR statistic

- Distribution of (X, Y) depends on key-bits *key*
- Observation on (X, Y) depends on key-bits *Key*
- LLR statistic depends on $key \cup Key$
- Distinguish correct and incorrect $key \cup Key$ with LLR statistic
- by computing $2^{|key \cup Key|}$ values of LLR
- For large (X, Y) the number of the key-bits involved $|key \cup Key|$ may be too large
- Not efficient

New Statistic

- Instead of $2^{|key \cup Key|}$ computations of LLR-values
- **Our work**: $<< 2^{|key \cup Key|}$($\approx 10^3$ times faster in DES)
- By using a new statistic
- Which reflects the structure of the round function
- That has a price to pay, but trade-off is positive

Outline

-
- New Statistic Construction
-
-
-
-

LLRs for Projections

- $(h_1, .., h_m)$ some subvectors (projections) of (X, Y) such that
- Distribution and Observation for h_i depend on a lower number of the key-bits $key_i \cup Key_i$
- LLR_i is a LLR-statistic for h_i
- Vector $(LLR_1, .., LLR_m)$ asymptotically distributed
- m-variate $\mathbf{N}(n\mu, nC)$ if $key \cup Key$ is correct
- Close to $\mathbf{N}(-n\mu, nC)$ if $key \cup Key$ is incorrect
- Mean vector μ, covariance matrix C, number of plain-texts n

LLR for Two Normal Distributions

- LLR statistic S to distinguish two normal distributions $\mathbf{N}(n\mu, nC)$ and $\mathbf{N}(-n\mu, nC)$
- S degenerates to linear:
- $S(key \cup Key, \nu) = \sum_{i=1}^{m} S_i(key_i \cup Key_i, \nu_i)$,
- where $S_i = \omega_i \, LLR_i$ weighted LLR statistic for h_i
- ν observation on (X, Y) and ν_i observation on h_i
- S is **separable**
- For polynomial distributions the theory of separable statistics was developed by Ivchenko, Medvedev,.. in 1970-s

Distribution

- S distributed 1-variate $\mathbf{N}(u, u)$ if $key \cup Key$ correct
- Close to $\mathbf{N}(-u, u)$ if incorrect
- for an explicit positive u

Cryptanalysis

- Find $key \cup Key$ s.t.

$$S(key \cup Key, \nu) > threshold$$

- without brute forcing $key \cup Key$
- Can be done as
- $S(key \cup Key, \nu) = \sum_{i=1}^{m} S_i(key_i \cup Key_i, \nu_i)$
- and $|key_i \cup Key_i|$ is much smaller than $|key \cup Key|$
- $|key \cup Key| = 54$ and $|key_i \cup Key_i| \approx 20$ in DES
- By solving efficiently an optimisation problem with a Search Algorithm

Outline

-
-
- Optimisation Problem and Search Algorithm
-
-
-

Optimisation Problem Example

S_1	0.1	0.2	0.3	0.1
$x_1 \oplus x_3$	0	0	1	1
x_2	0	1	0	1

S_2	0.5	0.1
$x_1 \oplus x_2$	0	1

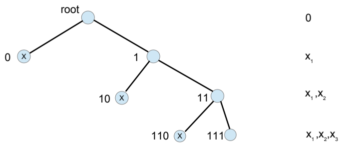
S_3	0.4	0.5	0.7	0.1
x_1	0	0	1	1
$x_2 \oplus x_3$	0	1	0	1

find binary x_1, x_2, x_3 s.t.

$$S(x_1, x_2, x_3) = S_1(x_1 \oplus x_3, x_2) + S_2(x_1 \oplus x_2) + S_3(x_1, x_2 \oplus x_3) > 1.3$$

Threshold is 1.3, solution 111

Search Tree



- One walks over a search tree and checks if the inequality $S_1(x_1 \oplus x_3, x_2) + S_2(x_1 \oplus x_2) + S_3(x_1, x_2 \oplus x_3) > 1.3$
- feasible under current fixation
- Cut if not feasible. Continue if feasible
- One is to check 6 linear inequalities. Brute force takes 8
- Same way one solves

$$S(key \cup Key, \nu) = \sum_{i=1}^{m} S_i(key_i \cup Key_i, \nu_i) > \text{threshold}$$

Success Probability& Number of $(\mathit{key} \cup \mathit{Key})$-candidates

- Search tree output is $(\mathit{key} \cup \mathit{Key})$-candidates for the final brute force
- The distribution of $S(\mathit{key} \cup \mathit{Key}, \nu)$ is known
- So one can compute success probability and
- The number of wrong solutions, that is $(\mathit{key} \cup \mathit{Key})$-candidates

Outline

-
-
-
- Implementation for 16-round DES
-
-

Two 14-bit vectors

- $\mathrm{DES}_K(X_0, X_1) = (X_{17}, X_{16})$
- Matsui's best linear approximation

$$X_2\{24, 18, 7\} \oplus X_{15}\{15\} \oplus X_{16}\{24, 18, 7, 29\}$$

- We use two 14-bit vectors

$$X_2[24, 18, 7, 29], X_{15}[16, 15, .., 11], X_{16}[24, 18, 7, 29]$$
$$X_1[24, 18, 7, 29], X_2[16, 15, .., 11], X_{15}[24, 18, 7, 29]$$

- Considered independent as they incorporate different bits
- Computing their distributions took a few seconds

Projections

- 28 projections

$$X_2[24, 18, 7, 29], X_{15}[i, j], X_{16}[24, 18, 7, 29]$$
$$X_1[24, 18, 7, 29], X_2[i, j], X_{15}[24, 18, 7, 29]$$

- For each projection LLR depends on ($\leq$21) key-bits
- 54 key-bits overall
- Two separable statistics for two independent bunches of the projections
- Search Algorithm combines ($\leq$ 21)-bit values to find 54-bit candidates
- Those candidates are brute forced

One Particular Projection

- projection h_1:

$$X_2[24, 18, 7, 29], X_{15}[16, 15], X_{16}[24, 18, 7, 29]$$

- $key_1 \cup Key_1$ incorporates 20 unknowns

$$x_{63}, x_{61}, x_{60}, x_{53}, x_{46}, x_{42}, x_{39}, x_{36}, x_{31},$$
$$x_{30}, x_{27}, x_{26}, x_{25}, x_{22}, x_{21}, x_{12}, x_{10}, x_{7}, x_{5},$$
$$x_{57} + x_{51} + x_{50} + x_{19} + x_{18} + x_{15} + x_{14}$$

  x_i key-bits of 56-bit DES key
- 2^{20} values of $S_1 = \omega_1 LLR_1$
- Similar for other 27 projections

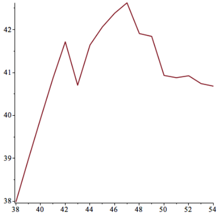
Key-variables Order for the Search Tree

- One needs $key \cup Key$ ordered to run a tree search
- x_2 appears in 14(maximal number) of $key_i \cup Key_i$, etc

$$x_2, x_{19}, x_{60}, x_{34}, x_{10}, x_{17}, x_{59}, x_{36}, x_{42}, x_{27}, x_{25},$$
$$x_{52}, x_{11}, x_{33}, x_{51}, x_9, x_{23}, x_{28}, x_5, x_{55}, x_{46}, x_{22},$$
$$x_{62}, x_{15}, x_{37}, x_{47}, x_7, x_{54}, x_{39}, x_{31}, x_{29}, x_{20}, x_{61},$$
$$x_{63}, x_{30}, x_{38}, x_{26}, x_{50}, x_1, x_{57}, x_{18}, x_{14}, x_{35}, x_{44},$$
$$x_3, x_{21}, x_{41}, x_{13}, x_4, x_{45}, x_{53}, x_6, x_{12}, x_{43}$$

Search Tree Algorithm Run

- We fixe desirable success rate 0.83
- solve equation $n = |\text{keys to brute force}|$ in n
- got $n = 2^{41.8}$



- The number of tree nodes is shown, $\log_2$ scale
- $|(key \cup Key)\text{-candidates}| = 2^{39.8}$, $|\text{keys to brute force}| = 2^{41.8}$
- Number of nodes is $2^{45.5} << 2^{54}$. Constructing the nodes is faster (in bit operations) than final brute force
- Improves Matsui's result on DES($n = 2^{43}$, 0.85)

Outline

-
-
-
-
- Multidimensional Distributions in Feistel Ciphers
-

r-Round DES

- $\text{DES}_K(X) = Y$, where X random, $\mathcal{E}$ any event
- We want to compute $\mathbf{Pr}(\mathcal{E})$ in r-round DES. Let's formalise
- $X_0, X_1, \ldots, X_{r+1}$ random independently generated 32-bit blocks. Event $\mathcal{C}$ defines DES:

$$X_{i-1} \oplus X_{i+1} = F_i(X_i, K_i), \quad i = 1, \ldots, r$$

- $K_1, \ldots, K_r$ fixed round keys. We need

$$\mathbf{Pr}(\mathcal{E}|\mathcal{C}) = \frac{\mathbf{Pr}(\mathcal{E}\mathcal{C})}{\mathbf{Pr}(\mathcal{C})} = 2^{32r}\mathbf{Pr}(\mathcal{E}\mathcal{C})$$

- infeasible as $\mathcal{C}$ depends on all key-bits

Relax $\mathcal{C}$

- One chooses a larger event $\mathcal{C}_\alpha$ (that is $\mathcal{C}$ implies $\mathcal{C}_\alpha$)

$$X_{i-1}[\alpha_i] \oplus X_{i+1}[\alpha_i] = F_i(X_i, K_i)[\alpha_i], \quad i = 1, \ldots, r$$

- where $\alpha = (\alpha_1, \ldots, \alpha_r)$. Then

$$\mathbf{Pr}(\mathcal{C}_\alpha) = 2^{-\sum_{i=1}^{r} |\alpha_i|}$$

- Let's accept

$$\mathbf{Pr}(\mathcal{E}|\mathcal{C}) \approx \mathbf{Pr}(\mathcal{E}|\mathcal{C}_\alpha) = \frac{\mathbf{Pr}(\mathcal{E}\mathcal{C}_\alpha)}{\mathbf{Pr}(\mathcal{C}_\alpha)} = 2^{\sum_{i=1}^{r} |\alpha_i|}\mathbf{Pr}(\mathcal{E}\mathcal{C}_\alpha)$$

- $\mathcal{C}_\alpha$ depends on a lower number of the key-bits. Now feasible and may be computed exactly

Regular Trails

- To compute the distribution of

$$Z = X_0[\alpha_1], X_1[\alpha_2 \cup \beta_1], X_r[\alpha_{r-1} \cup \beta_r], X_{r+1}[\alpha_r]$$

- One chooses event $\mathcal{C}_\alpha$, where $\alpha = (\alpha_1, \ldots, \alpha_r)$, and the trail

$$X_i[\beta_i], F_i[\alpha_i], \quad i = 1, \ldots, r$$

- The trail is called regular if

$$\gamma_i \cap (\alpha_{i-1} \cup \alpha_{i+1}) \subseteq \beta_i \subseteq \gamma_i, \quad i = 1, \ldots, r$$

  where $X_i[\gamma_i]$ input bits relevant to $F_i[\alpha_i]$
- For a regular trail $\mathbf{Pr}(Z = A|\mathcal{C}_\alpha)$ is computed with a convolution-type formula, only depends on α_i

Convolution Formula

- $Z = X_0[\alpha_1], X_1[\alpha_2 \cup \beta_1], X_r[\alpha_{r-1} \cup \beta_r], X_{r+1}[\alpha_r]$
- Then $\mathbf{Pr}(Z = A_0, A_1, A_r, A_{r+1} | \mathcal{C}_\alpha) =$

$$\frac{2^{\sum_{i=2}^{r-1} |\alpha_i|}}{2^{\sum_{i=1}^{r} |(\alpha_{i-1} \cup \alpha_{i+1}) \setminus \beta_i|}} \sum_{A_2,\dots,A_{r-1}} \prod_{i=1}^{r} \mathbf{q_i}(A_i[\beta_i], (A_{i-1} \oplus A_{i+1})[\alpha_i], k_i)$$

- probability distribution on round sub-vectors

$$\mathbf{q_i}(b, a, k) = \mathbf{Pr}(X_i[\beta_i] = b, F_i[\alpha_i] = a | K_i[\delta_i] = k)$$

- $K_i[\delta_i]$ key-bits relevant to $F_i[\alpha_i]$
- May be computed iteratively by splitting encryption into two parts. A few seconds for 14-round DES

Outline

-
-
-
-
-
- Conclusions

Conclusions

- Method of computing joint distribution of encryption internal bites X, Y(for Feistel ciphers) is found
- Conventional LLR statistic is inefficient for large X, Y. New statistic reflects round function structure
- We computed its distribution and able to predict success probability and the size of the final brute force
- Efficient Search Algorithm to find key-candidates which fall into critical region is presented

- Got an improvement over Matsui's results in DES (at least in bit operations)
- Predicted correctly success probability(8-round DES) and the number of final key-candidates(16-round DES)
- Search Algorithm is 10^3 times faster than brute forcing all key-bits which affect the statistic