Fault-based Attacks on the Bel-T Block Cipher Family

Philipp Jovanovic and Ilia Polian

{philipp.jovanovic,ilia.polian}@uni-passau.de

UNIVERSITÄT PASSAU
Fakultät für Informatik und Mathematik

Overview

- First **differential fault analysis** of the Bel-T block cipher family.
- **Bel-T**:
  - A 128-bit block cipher.
  - Supports key sizes of 128-bit, 192-bit, 256-bit.
  - Based on the Lai-Massey scheme.
  - National standard of the Republic of Belarus since 2011.
- **Full key recovery** using **4** (128-bit), **7** (192-bit), and **10** (256-bit) fault injections.
- Attacks based on realistic fault models.
- Extensive simulation-based experiments for verification of the developed methods.
- Low computational costs of analysis (feasible on common hardware).

Bel-T Specification

Key Setup (with 32-bit values θ_i):

- Bel-T-256: $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7, \theta_8$.
- Bel-T-192: $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7 := \theta_1 \oplus \theta_2 \oplus \theta_3, \theta_8 := \theta_4 \oplus \theta_5 \oplus \theta_6$.
- Bel-T-128: $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5 := \theta_1, \theta_6 := \theta_2, \theta_7 := \theta_3, \theta_8 := \theta_4$.

Key Usage (during *i*-th round):

	i	K_{7i-6}	K_{7i-5}	K_{7i-4}	K_{7i-3}	K_{7i-2}	K_{7i-1}	K_{7i}	
Encryption ↓	1	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	↑ Decryption
	2	θ_8	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	
	3	θ_7	θ_8	θ_1	θ_2	θ_3	θ_4	θ_5	
	4	θ_6	θ_7	θ_8	θ_1	θ_2	θ_3	θ_4	
	5	θ_5	θ_6	θ_7	θ_8	θ_1	θ_2	θ_3	
	6	θ_4	θ_5	θ_6	θ_7	θ_8	θ_1	θ_2	
	7	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8	θ_1	
	8	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7	θ_8	
	i	K_{7i}	K_{7i-1}	K_{7i-2}	K_{7i-3}	K_{7i-4}	K_{7i-5}	K_{7i-6}	

Encryption and Decryption:

$\mathcal{E}_\theta(X)$
Inputs:
$\theta \in \mathbb{F}_2^{256}, X \in \mathbb{F}_2^{128}$
Outputs:
$Y \in \mathbb{F}_2^{128}$
Algorithm:
1. $K \leftarrow \mathsf{setup\_keys}(\theta)$
2. $a \parallel b \parallel c \parallel d \leftarrow X$
3. **for** $i \in \{1, \ldots, 8\}$ **do**
4. $b \leftarrow b \oplus G_5(a \boxplus K_{7i-6})$
5. $c \leftarrow c \oplus G_{21}(d \boxplus K_{7i-5})$
6. $a \leftarrow a \boxminus G_{13}(b \boxplus K_{7i-4})$
7. $e \leftarrow G_{21}(b \boxplus c \boxplus K_{7i-3}) \oplus \langle i \rangle_{32}$
8. $b \leftarrow b \boxplus e$
9. $c \leftarrow c \boxminus e$
10. $d \leftarrow d \boxplus G_{13}(c \boxplus K_{7i-2})$
11. $b \leftarrow b \oplus G_{21}(a \boxplus K_{7i-1})$
12. $c \leftarrow c \oplus G_5(d \boxplus K_{7i})$
13. swap a and b
14. swap c and d
15. swap b and c
16. **end**
17. $Y \leftarrow b \parallel d \parallel a \parallel c$
18. **return** Y

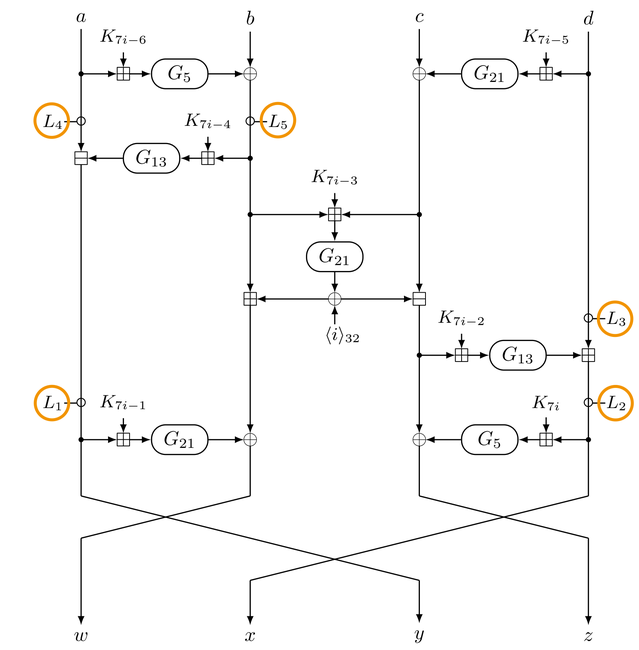
$\mathcal{D}_\theta(X)$
Inputs:
$\theta \in \mathbb{F}_2^{256}, X \in \mathbb{F}_2^{128}$
Outputs:
$Y \in \mathbb{F}_2^{128}$
Algorithm:
1. $K \leftarrow \mathsf{setup\_keys}(\theta)$
2. $a \parallel b \parallel c \parallel d \leftarrow X$
3. **for** $i \in \{1, \ldots, 8\}$ **do**
4. $b \leftarrow b \oplus G_5(a \boxplus K_{7i})$
5. $c \leftarrow c \oplus G_{21}(d \boxplus K_{7i-1})$
6. $a \leftarrow a \boxminus G_{13}(b \boxplus K_{7i-2})$
7. $e \leftarrow G_{21}(b \boxplus c \boxplus K_{7i-3}) \oplus \langle i \rangle_{32}$
8. $b \leftarrow b \boxplus e$
9. $c \leftarrow c \boxminus e$
10. $d \leftarrow d \boxplus G_{13}(c \boxplus K_{7i-4})$
11. $b \leftarrow b \oplus G_{21}(a \boxplus K_{7i-5})$
12. $c \leftarrow c \oplus G_5(d \boxplus K_{7i-6})$
13. swap a and b
14. swap c and d
15. swap b and c
16. **end**
17. $Y \leftarrow b \parallel d \parallel a \parallel c$
18. **return** Y

Substitution Layer *G* (with 8-bit SBox *H*):

$$G_r(u) = (H(u_1) \parallel H(u_2) \parallel H(u_3) \parallel H(u_4)) \lll r$$

Differential Fault Analysis

Random Fault Model (RFM): inject random fault values.
Chosen Fault Model (CFM): inject chosen fault values (here: value = 0).



Bel-T round function with fault locations $L_1, \ldots, L_5$

Attack on Bel-T-128 (4 RFM faults):

- Obtain $\theta_7 = \theta_3$ by injecting a single RFM-fault at L_1 during encryption:

$$G_{21}(L_1 \boxplus \theta_7) \oplus G_{21}((L_1 \oplus f_1) \boxplus \theta_7) = w \oplus w'$$

- Equivalently for $\theta_8 = \theta_4$ and L_2.
- Repeat attack for decryption and obtain $\theta_2 = \theta_6$ (L_1) and $\theta_1 = \theta_5$ (L_2).

Attack on Bel-T-192 (4 RFM + 3 CFM faults):

- Obtain θ_1, θ_2, θ_7, θ_8 as above and moreover $\theta_3 := \theta_1 \oplus \theta_2 \oplus \theta_7$.
- Obtain θ_6 by injecting a single CFM-fault at L_3 during encryption:

$$G_{13}(s \boxplus \theta_6) \boxplus 0 = x' \qquad s = G_5(x \boxplus \theta_8) \oplus z$$

- Obtain θ_4 by injecting dual CFM-faults at L_4 and L_5 during encryption:

$$0 \boxminus G_{13}(0 \boxplus \theta_4) = y'$$

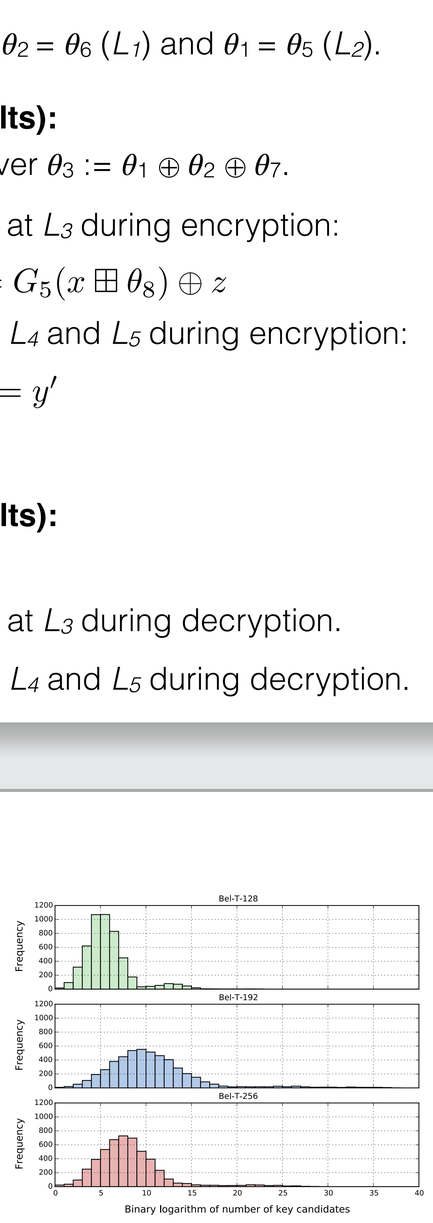
- Finally, $\theta_5 := \theta_4 \oplus \theta_6 \oplus \theta_8$.

Attack on Bel-T-256 (4 RFM + 6 CFM faults):

- Obtain θ_1, θ_2, θ_4, θ_6, θ_7, θ_8 as above.
- Obtain θ_3 by injecting a single CFM-fault at L_3 during decryption.
- Obtain θ_5 by injecting dual CFM-faults at L_4 and L_5 during decryption.

Experimental Results

- Evaluation of **5000** attack runs.
- Values denote binary logarithms for the number of key candidates in the sets Θ_i.
- Analysis of one instance: **148.0** (Bel-T-128), **287.0** (Bel-T-192), and **687.0** (Bel-T-256) seconds on a common workstation.



		Θ	Θ_1	Θ_2	Θ_3	Θ_4	Θ_5	Θ_6	Θ_7	Θ_8
Bel-T-128	min	0.00	0.00	0.00	0.00	0.00	-	-	-	-
	max	22.00	10.00	10.58	17.00	10.58	-	-	-	-
	avg	5.11	3.32	3.17	5.64	3.00	-	-	-	-
	med	4.58	1.00	1.00	1.00	1.00	-	-	-	-
Bel-T-192	min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-	-
	max	40.00	10.32	10.00	17.58	0.00	19.17	9.58	-	-
	avg	10.06	3.32	3.00	7.71	0.00	11.26	2.81	-	-
	med	9.17	1.00	1.00	3.58	0.00	2.00	1.00	-	-
Bel-T-256	min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	max	39.00	10.00	10.00	10.00	0.00	0.00	10.58	16.00	10.58
	avg	7.63	3.17	3.17	3.17	0.00	0.00	3.32	4.46	3.32
	med	7.00	1.00	1.00	1.00	0.00	0.00	1.00	1.00	1.00

Design, Automation & Test in Europe – DATE 15, 9 - 13 March, 2015, Grenoble, France