Practical-Time Related-Key Attack on GOST with Secret S-boxes

Orr Dunkelman[1] Nathan Keller[2] Ariel Weizman[2]

1 University of Haifa

2 Bar-Ilan University

Background

- GOST was developed in the USSR in the 1970’s, as an alternative for DES.
- It was the official encryption standard of the USSR, and the Russian Federation (RF) in 1989–2015.
- Since 2015, an instantiation of GOST, named Magma, is one of the two ciphers in the RF encryption standard GOST R 34.12-2015.
- Consequently, GOST is still very widely used in the RF.

GOST Bloch-Cipher

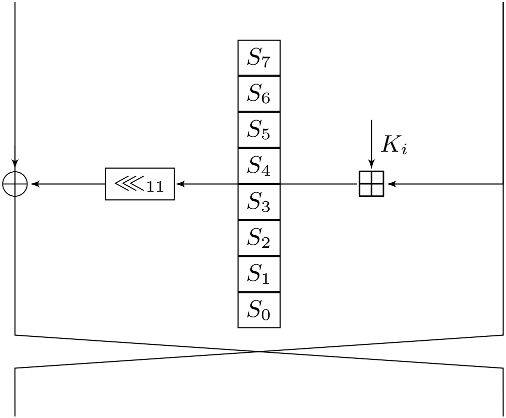
- 64-bit block size.
- 256-bit key size, defined by eight 32-bit words

$$K = (K_1, \ldots, K_8).$$

- 32 Feistel rounds.
- The round function is:

$$F_{K_i}(X_L, X_R) = (X_R, X_L \oplus \lll_{11} (S(X_R \boxplus K_i))).$$

The Round Function of GOST



One GOST Round.

Key Schedule

- Divide the 256-bit key into eight 32-bit subkeys $K_1, \ldots, K_8$. Use the original order in rounds 1–24, and the reverse order in rounds 25–32.

K_1	K_2	K_3	K_4	K_5	K_6	K_7	K_8
K_1	K_2	K_3	K_4	K_5	K_6	K_7	K_8
K_1	K_2	K_3	K_4	K_5	K_6	K_7	K_8
K_8	K_7	K_6	K_5	K_4	K_3	K_2	K_1

S-boxes

- $S_0 \dots, S_7 : \{0,1\}^4 \to \{0,1\}^4$.
- The structure of the S-boxes was kept secret, and different sets were used in different industry branches.
- The banking industry S-boxes were exposed in [S96]:

S_0	4	A	9	2	D	8	0	E	6	B	1	C	7	F	5	3
S_1	E	B	4	C	6	D	F	A	2	3	8	1	0	7	5	9
S_2	5	8	1	D	A	3	4	2	E	F	C	7	6	0	9	B
S_3	7	D	A	1	0	8	9	F	E	4	6	C	B	2	5	3
S_4	6	C	7	1	5	F	D	8	4	A	9	E	0	3	B	2
S_5	4	B	A	0	7	2	1	D	3	6	8	5	9	C	F	E
S_6	D	B	4	1	3	F	5	9	0	A	E	7	6	8	2	C
S_7	1	F	D	0	5	7	A	4	9	2	3	E	6	B	8	C

Contributions

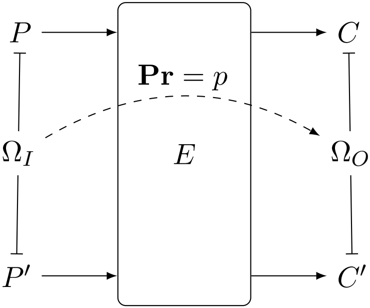
- Previous attacks assume at least one of the following:
  - Specific S-boxes.
  - Less than 32-rounds.
  - A small fraction of weak keys.
- Our attack:
  - Secret S-boxes.
  - Entire cipher.
  - All of the keys.

Comparison: Our Results vs Previous Attacks

No. of Rounds	Fraction of Keys	Secret S-boxes?	Data	Time	Technique and Source
21	all	no	2^{56} CP	2^{56}	RK Diff. [SK00]
24	all	no	?	?	RK Diff. [KSW96]
25	all	no	5 CP	2^{32}	RK Diff. [P11]
32	all	no	2^{36} CP	2^{36}	RK Diff. [KHLLK04]
32	all	no	2^{38} CP	2^{38}	Complementation [BN13]
32	all	no	2^{10} ACPC	2^{71}	RK Boom. [R11]
32	all	no	? ACPC	?	RK Boom. [PK13]
24	all	yes	2^{63} CP	2^{63}	Slide [BBDK18]
32	2^{-224}	yes	2^{32} CP	2^{32}	Slide [S98]
32	2^{-128}	yes	2^{40} CP	2^{40}	Slide [BBDK18]
32	**all**	**yes**	2^{27} CP	2^{27}	**RK Diff. (Sec. 4)**

Differential Cryptanalysis [BS91]

- **Differential cryptanalysis** analyzes block ciphers by tracking the development of differences through the encryption process of a pair of plaintexts.



Differential

Differential Cryptanalysis [BS91]

- A differential with probability p of E is a statistical property of the form

$$\Pr[E(P) \oplus E(P') = \Omega_O \mid P \oplus P' = \Omega_I] = p.$$

- Denoted by

$$\Omega_I \xrightarrow[E]{p} \Omega_O.$$

Related-Key (RK) Attacks [B94, K92]

- Related-key (in short, RK) attacks were introduced by Biham and by Knudsen, independently.
- In this model, the adversary may obtain the encryption of plaintexts under several **related unknown keys**, where the relation between the keys is known.

RK Differential Cryptanalysis [KSW96]

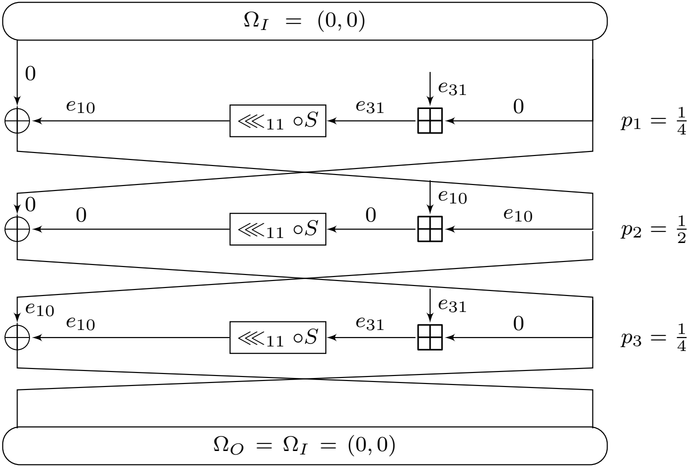
- A RK differential with probability p of E under the key difference Ω_K is a statistical property of the form

$$\Pr[E_K(P) \oplus E_{K'}(P') = \Omega_O \mid P \oplus P' = \Omega_I, K \oplus K' = \Omega_K] = p.$$

- Denoted by

$$\Omega_I \xrightarrow[\Omega_K]{p} \Omega_O.$$

3-Round Iterative RK Differential (bank S-boxes)



Extension of the 3-Round Iterative RK Differential

- Consider the previous **3-round** RK char.
- We can append 5-round RK char. with 0 difference.
- We obtain an **8-round** RK iterative char. $(0,0) \xrightarrow{2^{-5}} (0,0)$, which we concatenate to itself 3 times and obtain **24-round** one: $(0,0) \xrightarrow{2^{-15}} (0,0)$.

What Happen in the Last 8 Rounds?

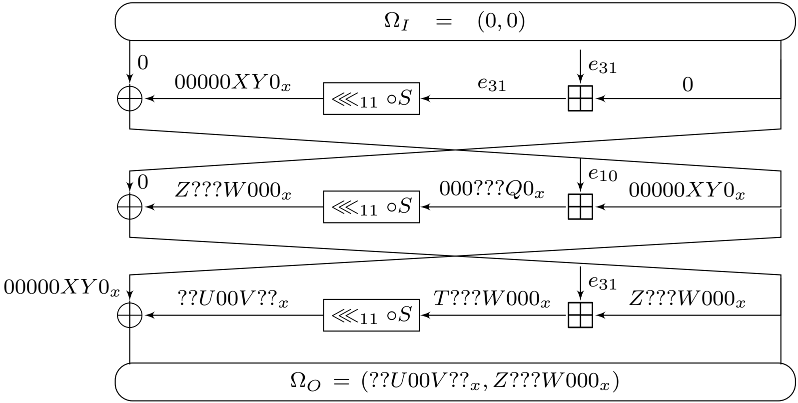
- Start with key difference

$$\Omega_K = (e_{31}, e_{10}, e_{31}, 0, 0, 0, 0, 0).$$

- Means: 0 sub-key difference in rounds 25–29!
- Therefore we get the **29-rounds** RK differential

$$(0, 0) \xrightarrow[\Omega_K]{2^{-15}} (0, 0).$$

Differences Development in the Last 3 Rounds



Where $X, Z, V \in \{0, \ldots, 7\}, Q, Y, W, U \in \{0, 8\}, T \in \{8, \ldots, 15\}$.

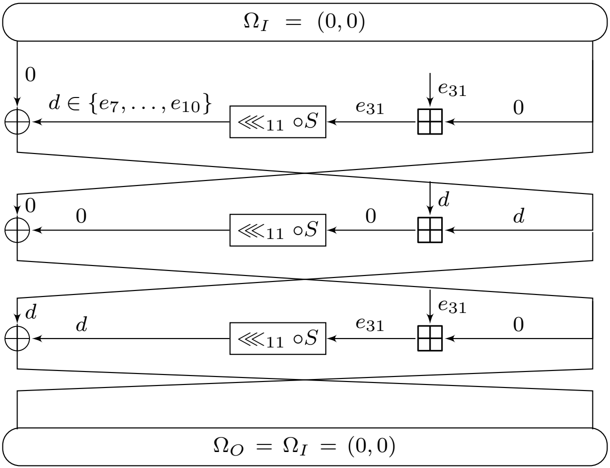
Conclusion: 32-Rounds Truncated differential

- We get the following **32-rounds** truncated differential:

$$(0,0) \xrightarrow[\Omega_K]{2^{-15}} (??U00V??_x, Z???W000_x).$$

- Note: It is possible to concatenate the 8-round RK differential 4 times to obtain 32-round RK differential $(0,0) \xrightarrow[\Omega_K]{2^{-20}} (0,0)$.
- However, the use of truncated differential allows us to recover key material.

Adjustments for Secret S-boxes



$p_1 \in \left\{\frac{1}{8}, \frac{1}{4}, \frac{3}{8}, \frac{1}{2}\right\}$

$p_2 = \frac{1}{2}$

$p_3 = p_1$

S-box Recovery

- Goal: Recovering a secret S-box $S : \{0,1\}^n \to \{0,1\}^n$, up to an XOR with a constant.
- Assumptions:
  1. $S(0) = 0$. It's OK since we recover S only up to an XOR with a constant ($S(0)$ is the constant).
  2. We have m triples (v_i, v'_i, d_i) (sorted by v_i) where (v_i, v'_i) is a pair of input values to S, and $d_i = S(v_i) \oplus S(v'_i)$ is the corresponding output difference (we can achieve such triples using out distinguisher).

S-box Recovery

Recovering process:

1. Look for pairs of the form $(v_i, v_i') = (0, x)$. The assumption $S(0) = 0$ implies $S(x) = d_i$.
2. Look for pairs of the form $(v_j, v_j') = (x, y)$. Therefore:

$$d_j = S(x) \oplus S(y) \Rightarrow S(y) = d_j \oplus S(x).$$

3. And so on.

Achieving the Triples

- Using our distinguisher, we expect to find, by generating at mots 2^{24} plaintexts, a plaintext P s.t.

$$C \oplus C' = E_K(P) \oplus E_{K'}(P) = (??U00V??_x, Z???W000_x).$$

- Given one right pair, we can find (using neutral bits) additional 2^8 right pairs, using additional 2^{24} plaintexts.

Attack Stages

- In the first three stages we use three different variants of the RK differential, in which:
  1. Recover the S-boxes S_4, S_5.
  2. Recover the S-boxes S_1, S_2.
  3. Recover the S-box S_7.
- In the forth stage we reuse the ciphertexts obtain in the first three stages to **fully recover all of the S-boxes**.
- In all of the stages we eliminate wrong candidates of the sub-key K_1.

Summary Table

- The success rate and the complexities of the attack, using 7 RK and 256 right pairs for each characteristic.

	1st stage	2nd stage	3rd stage	4th stage	Overall
Success rate	$97/100$	$94/97$	$92/94$	$88/92$	88%
Data	$2^{22.2}$	$2^{22.4}$	$2^{23.2}$	0	$2^{24.2}$
Time	$2^{22.2}$	$2^{22.4}$	$2^{23.2}$	negligible	$2^{24.2}$
Memory	$2^{9.5}$	2^{9}	2^{9}	0	$3 \cdot 2^{9} = 2^{10.6}$

Right Pairs vs Success Rate and Complexity

- The effect of the num. of right pairs on the success rate and the complexity of the full attack.

Num. of right pairs	128	192	256	384	512
Success rate	84%	88%	88%	91%	83%
Data and time	$2^{23.9}$	$2^{24.2}$	$2^{24.2}$	$2^{24.5}$	2^{25}

Thank you for your attention!

Questions?