Cryptanalysis of GOST2: Can Updated Key Schedule Solve all of GOST's Problems?

Orr Dunkelman
(joint work with Achiya Bar-On and Tomer Ashur)

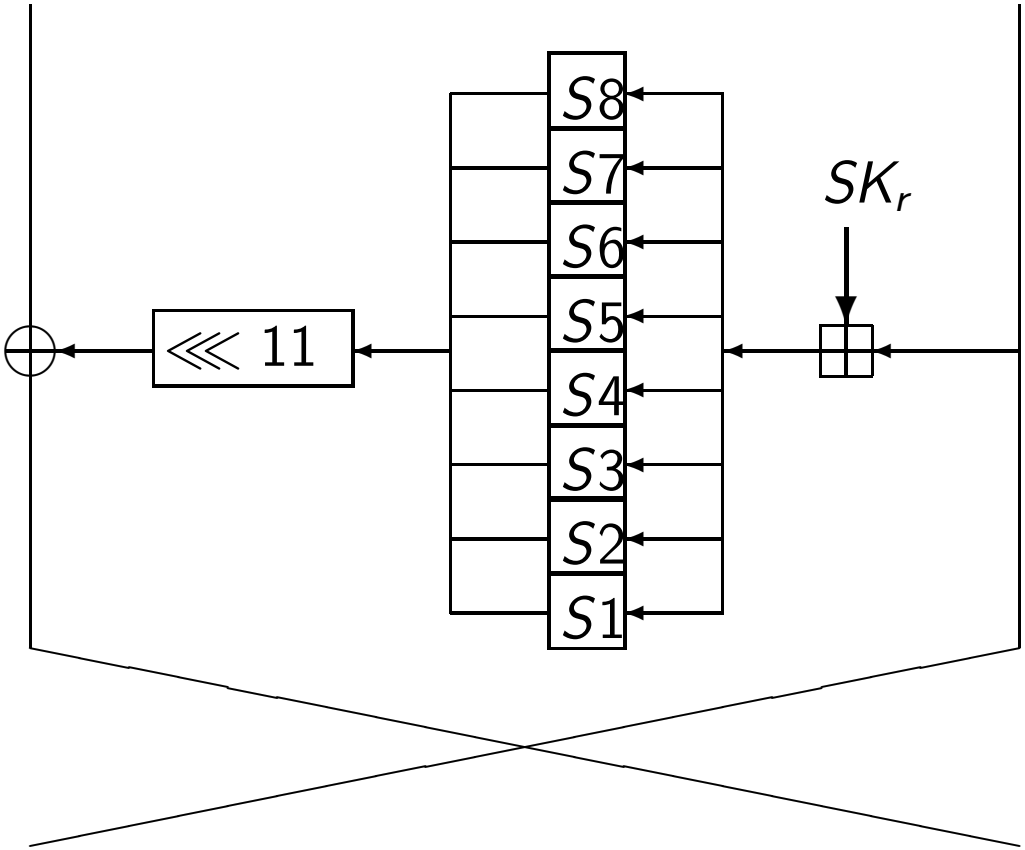
University of Haifa

June 29, 2016

History of the GOST Block Cipher

- GOST 28147-89 defined a block cipher (A.K.A. Magma these days)
- 64-bit block, 256-bit key
- 32-round Feistel
- With different secret S-boxes for each industry (a few leaked)

The GOST Block Cipher



The GOST Key Schedule

- The key schedule takes a 256-bit key (eight 32-bit words — $K_0, K_2, K_3, \ldots, K_7$) and uses them according to:

K_0	K_1	K_2	K_3	K_4	K_5	K_6	K_7
K_0	K_1	K_2	K_3	K_4	K_5	K_6	K_7
K_0	K_1	K_2	K_3	K_4	K_5	K_6	K_7
K_7	K_6	K_5	K_4	K_3	K_2	K_1	K_0

- The descending order — probably to defeat slide attacks

Attacks on GOST (Short and Partial History)

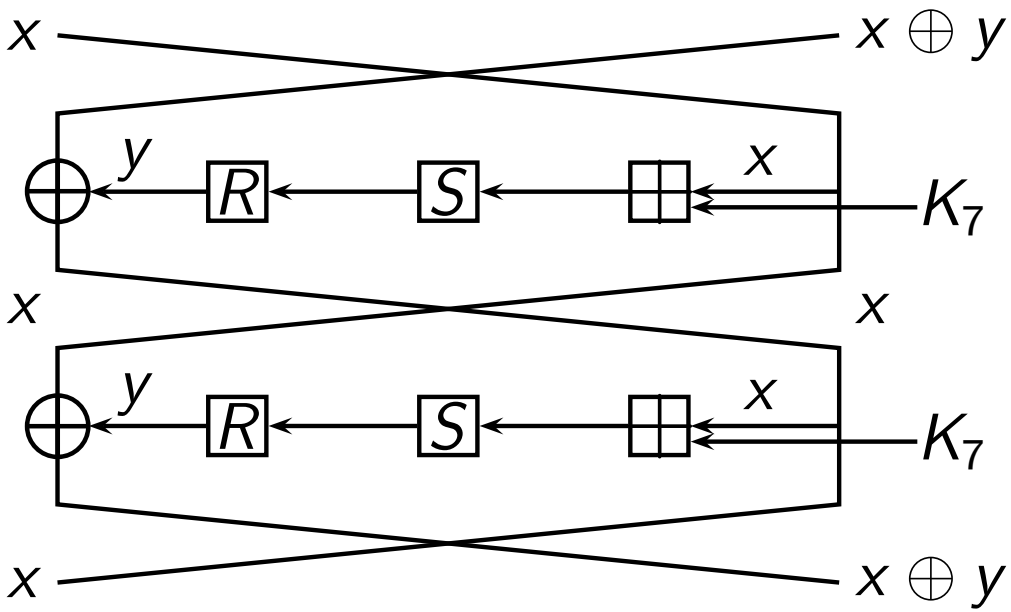
- Related-key differential attacks on reduced-round GOST (specific S-boxes) [KSW96]
- Chosen-key S-box recovery attacks [S99]
- Related-key differential attacks on reduced-round GOST [KS00]
- Related-key differential attacks on full GOST [K+04]
- Slide attacks on first 24 rounds [BW00]
- Slide attacks on full GOST for a weak key class of 2^{128} keys [BW00]
- Slide attacks on first 30 rounds [BDK07]

Attacks on GOST (Short and Partial History)

Attack	Data	Memory	Time	S-boxes
Reflection [I11]	2^{32}CP	2^{64}	2^{224}	Bijective
Fixed point/Algebraic [C11]	2^{64}KP	2^{64}	2^{248}	Russian Banks
Differential [CM11]	2^{64}KP	2^{64}	2^{226}	Russian Banks
Fixed point [DDS12]	2^{64}KP	2^{36}	2^{192}	any
Fixed point [DDS12]	2^{64}KP	2^{19}	2^{204}	any
Reflection [DDS12]	2^{32}KP	2^{36}	2^{224}	any
Reflection [DDS12]	2^{32}KP	2^{19}	2^{236}	any

Very Quick Summary of the Reflection Attack

- Assume that at the entrance to round 25, the intermediate encryption value is (x, x)
- Then round 25 cancels round 24, round 26 cancels round 23, etc.



Very Quick Summary of the Reflection Attack

- Isobe noticed that for a reflection point, the intermediate encryption value after 16 rounds is equal to the ciphertext
- This allows for attacking 16-round GOST (using meet in the middle, or any attack you wish for)

ISO SC27 (Parallel Work)

- The Russian federation has submitted GOST (Magma) for standardization in 2010 to ISO SC27 (18033)
- Several issues spotted:
  - S-boxes were not defined
  - Related-key attacks
- By the time they were "addressed", Isobe's attack came out

In Mother Russia, Cipher Encrypts You!

- Following the failure of standardizing GOST, a new cipher was suggested
- Kuznyechik (Grasshopper) — 128-bit block, 256-bit key SPN
- Secret design process
- Interesting properties revealed by [BP15,BPU16] about how the S-box was designed
- And then came a new proposal...

The GOST2 Block Cipher

- Dmukh, Dygin, and Marshalko offered a variant of GOST on eprint report 2015/065
- Two main changes with respect to GOST:
  - S-boxes are fully specified
  - Key schedule changed to:

K_0	K_1	K_2	K_3	K_4	K_5	K_6	K_7
K_3	K_4	K_5	K_6	K_7	K_0	K_1	K_2
K_5	K_6	K_7	K_0	K_1	K_2	K_3	K_4
K_6	K_5	K_4	K_3	K_2	K_1	K_0	K_7

The Security Claims

Both Isobe and Dinur-Dunkelman-Shamir attacks exploit the reflection property for the last 16 iterations. For the proposed algorithm the probability of the corresponding event is negligible: $P\{K_0 = K_2 = K_4 = K_6, K_1 = K_3 = K_5 = K_7)\} = 2^{-192}$ *(if keys are selected at random).*

The first Dinur-Dunkelman-Shamir method works if $K_0 = K_2 = K_4 = K_6 = K_1 = K_3 = K_5 = K_7$. *The probability of such event is* 2^{-224}.

Since the new key schedule could be represented as a concatenation of different shifts of $(K_0, \ldots, K_7)$, *2-GOST (together with original GOST) is subjected to related-key attacks. At the same time, such attacks are difficult for practical implementation, since the probabilities of relations are negligible (see, for example, [5]), when keys are selected randomly.*

. . .

Eprint report 2015/065

The Security Claims

A Reflection Property for GOST2 (Weak Key Class)

- Consider the key schedule of rounds 18–31, when $K_5 = K_6$:

K_5	K_6	K_7	K_0	K_1	K_2	K_3 K_3	K_4 K_4
K_6 K_6	K_5 K_5	K_4 K_4	K_3 K_3	K_2	K_1	K_0	K_7

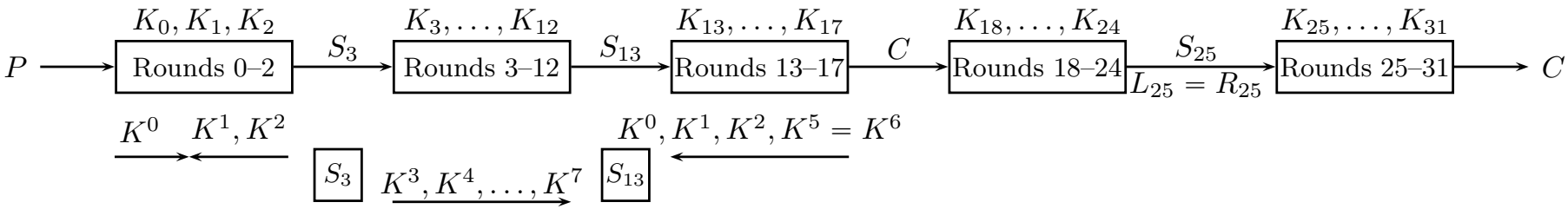
- Hence, if the intermediate encryption value after 25 rounds is (x, x), the ciphertext is equal to the value after 18 rounds

A Reflection Attack on GOST2 (Weak Key Class)

Require: 2^{32} pairs of known plaintexts and ciphertexts - $\{P_i, C_i\}$.

for $S_3, K_5 = K_6$ **do**
  for $(P_i, C_i), K_0$ **do**
    $K_1, K_2 \leftarrow \text{Solve}(P_i, S_3, K_0)$
    $S_{13} \leftarrow R^{-1}_{SK_{13}}(R^{-1}_{SK_{14}}(R^{-1}_{SK_{15}}(R^{-1}_{SK_{16}}(R^{-1}_{SK_{17}}(C_i = S_{18})))))$
    $T[S_{13}] \leftarrow (P_i, K_0, K_1, K_2)$
  end for
  for K_3, K_4, K_7 **do**
    $S_{13} \leftarrow$
    $R_{SK_{12}}(R_{SK_{11}}(R_{SK_{10}}(R_{SK_9}(R_{SK_8}(R_{SK_7}(R_{SK_6}(R_{SK_5}(R_{SK_4}(R_{SK_3}(S_3)))))))))$
    $(P_i, K_0, K_1, K_2) \leftarrow T[S_{13}]$
    $\text{TRY}(K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7)$
  end for
end for

A Reflection Attack on GOST2 (Weak Key Class)



- Data complexity: 2^{32} KPs
- Memory complexity: 2^{64} blocks
- Time complexity: 2^{192}
- Weak Key Size: 2^{224}
- Attack can be transformed into an impossible reflection attack for all other keys (data increased to 2^{64}, saves a factor of 5.4 on exhaustive search)

A Fixed Point Property for GOST2

- Consider the key schedule of rounds 10–22:

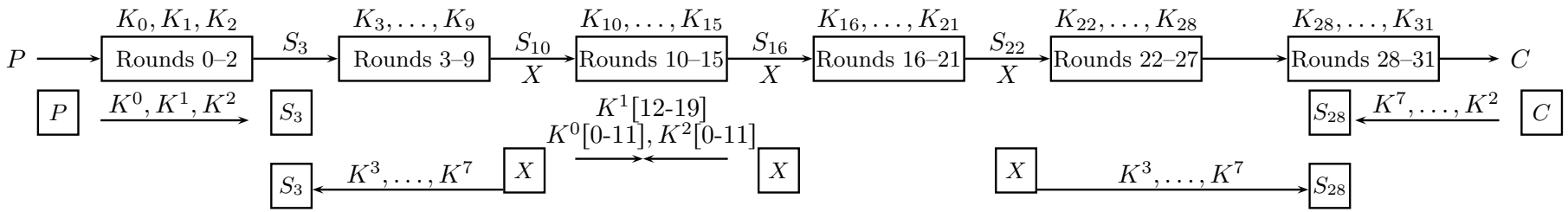
K_3	K_4	K_5	K_6	K_7	K_0	K_1	K_2
K_5	K_6	K_7	K_0	K_1	K_2	K_3	K_4

- The keys of rounds 10–15 are the same as 16–21
- Hence, a fixed point of rounds 10–15 is a fixed point for rounds 10–21

A Fixed-Point Attack on GOST2

Require: 2^{64} pairs of known plaintexts and ciphertexts.
  for $(P_i, C_i), SK_0, SK_1, SK_2, SK_7$ **do**
    $S_{28} \leftarrow R^{-1}_{SK_{28}}(R^{-1}_{SK_{29}}(R^{-1}_{SK_{30}}(R^{-1}_{SK_{31}}(C_i))))$
    $S_3 \leftarrow R_{K_2}(R_{K_1}(R_{K_0}(P_i)))$
    $T[S_3||S_{28}] \leftarrow (K_0, K_1, K_2, K_7)$
  end for
  for $S_{10} = S_{16} = S_{22}, K_3, K_4, K_5, K_6, K_7$ **do**
    $S_{13} \leftarrow R_{SK_{12}}(R_{SK_{11}}(R_{SK_{10}}(S_{10})))$
    for $K_0[0\text{–}11], K_2[0\text{–}11], K_1[10]$ **do**
      $(K_0[0\text{–}11], K_1[12\text{–}19], K_2[0\text{–}11]) \leftarrow$
      $\mathrm{SOLVE}(S_{16}, S_{13}, K_0[0\text{–}11], K_2[0\text{–}11], \mathrm{Carry})$
    end for
    $S_3 \leftarrow R^{-1}_{SK_3}(R^{-1}_{SK_4}(R^{-1}_{SK_5}(R^{-1}_{SK_6}(R^{-1}_{SK_7}(R^{-1}_{SK_8}(R^{-1}_{SK_9}(S_{10})))))))$
    $S_{28} \leftarrow R_{SK_{27}}(R_{SK_{26}}(R_{SK_{25}}(R_{SK_{24}}(R_{SK_{23}}(R_{SK_{22}}(S_{22}))))))$
    $(K_0, K_1, K_2, K_7) \leftarrow T[S_3||S_{28}]$
    $\mathrm{Filter}(K_0, K_1, K_2, K_7)$
    $\mathrm{TRY}(K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7)$
  end for

A Fixed-Point Attack on GOST2



- Data complexity: 2^{64} KPs
- Memory complexity: 2^{160} blocks
- Time complexity: 2^{237}

We are working on reducing memory consumption.

Summary

- New GOST2 does not offer full security against fixed-point and reflection attacks
- Same related-key attacks can be applied (including complementation property)
- Simple ways to handle these issues exist

Summary of Attacks

Type of attack	Time	Data	Memory (blocks)	No. of keys
Fixed point	2^{237}	2^{64}KP	2^{160}	All
Reflection	2^{192}	2^{32}KP	2^{64}	2^{224}
Impossible reflection	$2^{253.56}$	2^{63}CP	2^{160}	$2^{256} - 2^{224}$
Impossible reflection	$2^{254.56}$	2^{64}KP	2^{160}	$2^{256} - 2^{224}$

Some Aftermath

- We posted our results (not including some optimizations we now have) on eprint (report 2016/532)
- And we got an interesting email from Grigory Marshalko:

  . . . It was clear from the very beginning that with such a slight change of the key schedule it would be impossible to fully protect the cipher from these attacks since the reflection property still exists. Nevertheless the figures you obtained shows that it is really possible to mitigate the security threats in a way. . . .

Summary 2

Wait!

- The security analysis does not really say that there are no shortcut attacks
- It just implies that fact
- and the designer admits they assumed security will not be perfect
- Let's leave the conspiracy theorists what they think of that...

Questions?

Thank you

for your Attention!