Differential cryptanalysis of the quasigroup cipher

Marko Hassinen, Smile Markovski

Marko.Hassinen@cs.uku.fi, smile@ii.edu.mk

University of Kuopio, Finland

SS Cyril and Methodius University,

Republic of Macedonia

Contents

- Motivation
- Definition of the encryption method
- Differential cryptanalysis
- Results
- Conclusions and Future Work

Quasigroup encryption

- A *groupoid* is a finite set Q that is closed with respect to an operator $*$

Quasigroup encryption

- A *groupoid* is a finite set Q that is closed with respect to an operator $*$
- A *quasigroup* is a groupoid with unique left and right inverses.

Quasigroup encryption

- A *groupoid* is a finite set Q that is closed with respect to an operator $*$
- A *quasigroup* is a groupoid with unique left and right inverses.
- A quasigroup can be characterised with a $Latin\ square$ that is an $n * n$ matrix where each row and column is a permutation of elements of a set

Quasigroup encryption

- A *groupoid* is a finite set Q that is closed with respect to an operator $*$
- A *quasigroup* is a groupoid with unique left and right inverses.
- A quasigroup can be characterised with a $Latin\ square$ that is an $n * n$ matrix where each row and column is a permutation of elements of a set
- The encryption primitive e_l on sequence $x_1x_2 \ldots x_n$ is defined as $e_l(x_1x_2 \ldots x_n) = y_1y_2 \ldots y_n$ where

$$\begin{cases} y_1 = l * x_1, \\ y_{i+1} = y_i * x_{i+1} (i = 1, \ldots n - 1) \end{cases}$$

Encryption cont.

$$\begin{array}{l} \ \ a_1a_2a_3a_4a_5 \ldots \\ \mathrm{l}\ b_1b_2b_3b_4b_5 \ldots \\ \mathrm{l}\ c_1c_2c_3c_4c_5 \ldots \\ \ldots \end{array}$$

Decryption

- Decryption $d_l : A^+ \rightarrow A^+$ is defined as $d_l(y_1y_2 \ldots y_n) = x_1x_2 \ldots x_n$,where

$$\begin{cases} x_1 = l \backslash y_1, \\ x_{i+1} = y_i \backslash y_{i+1} (i = 1, \ldots n - 1) \end{cases}$$

Differential cryptanalysis on a Feistel cipher

- Originally designed for iterated block ciphers (DES)

Differential cryptanalysis on a Feistel cipher

- Originally designed for iterated block ciphers (DES)
- Eli Biham and Adi Shamir

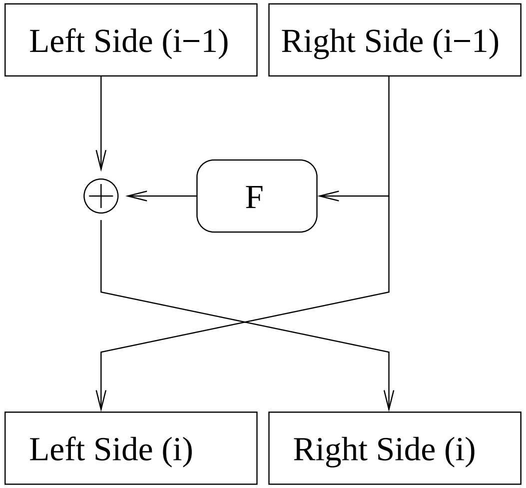
Differential cryptanalysis on a Feistel cipher

- Originally designed for iterated block ciphers (DES)
- Eli Biham and Adi Shamir
- A known plaintext attack

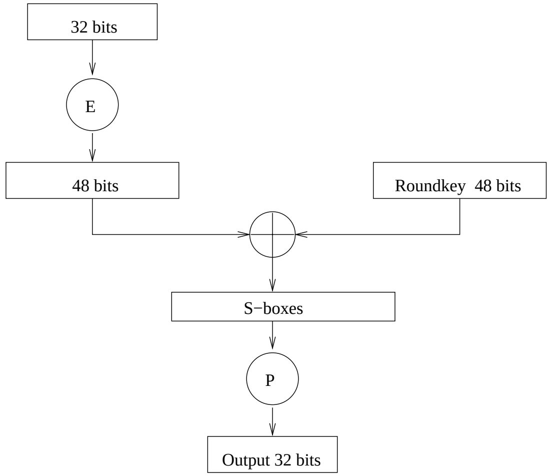
Differential cryptanalysis on a Feistel cipher

- Originally designed for iterated block ciphers (DES)
- Eli Biham and Adi Shamir
- A known plaintext attack
- A large amount of ciphertext - plaintext pairs is used

The Feistel structure



The Feistel structure



Differential cryptanalysis on a Feistel cipher

- We define a $charasteristic$ as follows. X causes Y with probability p, marked $X \to Y$, if for fraction $\frac{1}{p}$ of input pairs whose XOR is X the output XOR is Y.

Differential cryptanalysis on a Feistel cipher

- We define a $charasteristic$ as follows. X causes Y with probability p, marked $X \to Y$, if for fraction $\frac{1}{p}$ of input pairs whose XOR is X the output XOR is Y.
- From analyzing the crypto component we obtain *difference distribution table*

Differential cryptanalysis on a Feistel cipher

- We define a $charasteristic$ as follows. X causes Y with probability p, marked $X \rightarrow Y$, if for fraction $\frac{1}{p}$ of input pairs whose XOR is X the output XOR is Y.
- From analyzing the crypto component we obtain *difference distribution table*
- Input XOR $\Delta X = x_1 \oplus x_2$

Differential cryptanalysis on a Feistel cipher

- We define a $charasteristic$ as follows. X causes Y with probability p, marked $X \rightarrow Y$, if for fraction $\frac{1}{p}$ of input pairs whose XOR is X the output XOR is Y.
- From analyzing the crypto component we obtain *difference distribution table*
- Input XOR $\Delta X = x_1 \oplus x_2$
- Output difference of the component $\Delta Z = (Y_1 \oplus K) \oplus (Y_2 \oplus K))$

Differential cryptanalysis on a Feistel cipher

- We define a $charasteristic$ as follows. X causes Y with probability p, marked $X \rightarrow Y$, if for fraction $\frac{1}{p}$ of input pairs whose XOR is X the output XOR is Y.
- From analyzing the crypto component we obtain *difference distribution table*
- Input XOR $\Delta X = x_1 \oplus x_2$
- Output difference of the component $\Delta Z = (Y_1 \oplus K) \oplus (Y_2 \oplus K))$
- $\Delta Z = Y_1 \oplus Y_2$, since $(Y_1 \oplus K) \oplus (Y_2 \oplus K)$ = $Y_1 \oplus Y_2 \oplus K \oplus K$.

Differential analysis of a quasigroup

```
for (a_1 := 0 ... Quasigroupsize)
  for (a_2 := 0 ... Quasigroupsize)
    for (leader := 0 ... Quasigroupsize)
      c_1 := e_transformation(leader,a_1)
      c_2 := e_transformation(leader,a_2)
      input_xor := a_1 ⊕ a_2
      output_xor := c_1 ⊕ c_2
      distributions[input_xor][output_xor]++
    endfor
  endfor
endfor
```

3	12	6	14	8	9	13	11	15	4	1	5	10	7	0	2
4	10	14	9	0	12	7	5	11	8	3	15	1	6	2	13
12	9	1	3	14	11	2	8	13	5	6	0	7	15	10	4
15	5	10	11	7	14	4	13	3	0	2	1	12	8	9	6
0	11	3	10	13	5	8	14	1	15	12	9	6	2	4	7
10	1	8	12	11	0	5	3	9	13	4	7	2	14	6	15
6	4	15	13	1	7	14	9	8	10	5	2	11	3	12	0
5	15	13	2	9	10	1	12	0	6	7	14	4	11	3	8
13	6	7	1	2	8	9	10	14	3	15	4	0	5	11	12
14	7	11	4	3	2	15	0	12	9	8	6	5	10	13	1
9	13	2	0	15	4	10	7	6	12	11	3	14	1	8	5
8	14	5	15	6	1	0	4	10	2	9	11	13	12	7	3
11	2	9	6	5	13	12	15	4	7	10	8	3	0	1	14
1	3	0	8	10	15	6	2	7	14	13	12	9	4	5	11
7	0	12	5	4	6	3	1	2	11	14	10	8	13	15	9
2	8	4	7	12	3	11	6	5	1	0	13	15	9	14	10

An example quasigroup of order 16

$I \backslash O$	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000	256	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001	0	10	24	22	24	14	20	14	18	20	14	16	10	16	10	24
0010	0	20	20	26	10	28	10	22	12	12	14	20	12	22	14	14
0011	0	26	14	16	12	22	6	12	28	16	24	24	18	18	10	10
0100	0	14	10	18	20	16	20	22	20	12	14	24	10	12	30	14
0101	0	18	20	20	18	14	18	16	10	18	18	24	12	18	16	16
0110	0	20	14	16	20	22	10	18	26	18	14	12	8	14	24	20
0111	0	8	14	18	24	16	24	16	14	24	22	16	10	12	16	22
1000	0	16	26	22	14	18	12	12	14	18	14	18	28	20	12	12
1001	0	16	16	20	8	20	16	16	12	12	20	12	24	12	24	28
1010	0	24	28	8	18	18	18	22	8	20	16	8	14	18	14	22
1011	0	24	20	6	10	20	14	14	16	22	22	18	18	18	20	14
1100	0	12	12	18	18	10	20	18	14	14	12	26	26	34	14	8
1101	0	10	20	12	22	16	22	18	20	18	20	24	10	12	14	18
1110	0	14	6	20	20	8	22	18	18	18	20	8	34	12	20	18
1111	0	24	12	14	18	14	24	18	26	14	12	6	22	18	18	16

An example difference distribution table

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0
2	3	4	5	6	7	8	9	10	11	12	13	14	15	0	1
3	4	5	6	7	8	9	10	11	12	13	14	15	0	1	2
4	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3
5	6	7	8	9	10	11	12	13	14	15	0	1	2	3	4
6	7	8	9	10	11	12	13	14	15	0	1	2	3	4	5
7	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6
8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7
9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8
10	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9
11	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10
12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11
13	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12
14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13
15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Another example quasigroup of order 16

$I \backslash O$	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000	256	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001	0	128	0	64	0	0	0	32	0	0	0	0	0	0	0	32
0010	0	0	128	0	0	0	64	0	0	0	0	0	0	0	64	0
0011	0	64	0	64	0	32	0	32	0	0	0	0	0	32	0	32
0100	0	0	0	0	128	0	0	0	0	0	0	0	128	0	0	0
0101	0	0	0	32	0	64	0	32	0	0	0	32	0	64	0	32
0110	0	0	64	0	0	0	64	0	0	0	64	0	0	0	64	0
0111	0	32	0	32	0	32	0	32	0	32	0	32	0	32	0	32
1000	0	0	0	0	0	0	0	0	256	0	0	0	0	0	0	0
1001	0	0	0	0	0	0	0	32	0	128	0	64	0	0	0	32
1010	0	0	0	0	0	0	64	0	0	0	128	0	0	0	64	0
1011	0	0	0	0	0	32	0	32	0	64	0	64	0	32	0	32
1100	0	0	0	0	128	0	0	0	0	0	0	0	128	0	0	0
1101	0	0	0	32	0	64	0	32	0	0	0	32	0	64	0	32
1110	0	0	64	0	0	0	64	0	0	0	64	0	0	0	64	0
1111	0	32	0	32	0	32	0	32	0	32	0	32	0	32	0	32

Another example difference distribution table

Brute force attack

- Known quasigroup operations

Brute force attack

- Known quasigroup operations
- A brute force against leaders $l_1 \dots l_n, l_i \in Q$

Brute force attack

- Known quasigroup operations
- A brute force against leaders $l_1 \dots l_n, l_i \in Q$
- $|Q|$ different leaders

Brute force attack

- Known quasigroup operations
- A brute force against leaders $l_1 \dots l_n, l_i \in Q$
- $|Q|$ different leaders
- We need in average $\frac{(|Q|*k)^n}{2}$ trials

Brute force attack

- Known quasigroup operations
- A brute force against leaders $l_1 \dots l_n, l_i \in Q$
- $|Q|$ different leaders
- We need in average $\frac{(|Q|*k)^n}{2}$ trials
- k is the number of different operations used

$\|Q\|$	5	10	20	30	40
4	$1.53 * 2^{21}$	$1.16 * 2^{43}$	$1.36 * 2^{86}$	$1.58 * 2^{129}$	$1.84 * 2^{172}$
8	$1.53 * 2^{26}$	$1.16 * 2^{53}$	$1.36 * 2^{106}$	$1.58 * 2^{159}$	$1.84 * 2^{212}$
16	$1.53 * 2^{31}$	$1.16 * 2^{63}$	$1.36 * 2^{126}$	$1.58 * 2^{189}$	$1.84 * 2^{252}$
32	$1.53 * 2^{36}$	$1.16 * 2^{73}$	$1.36 * 2^{146}$	$1.58 * 2^{219}$	$1.84 * 2^{292}$
64	$1.53 * 2^{41}$	$1.16 * 2^{83}$	$1.36 * 2^{166}$	$1.58 * 2^{249}$	$1.84 * 2^{332}$
128	$1.53 * 2^{46}$	$1.16 * 2^{93}$	$1.36 * 2^{186}$	$1.58 * 2^{279}$	$1.84 * 2^{372}$
256	$1.53 * 2^{51}$	$1.16 * 2^{103}$	$1.36 * 2^{206}$	$1.58 * 2^{309}$	$1.84 * 2^{412}$

Amount of tries needed for a quasigoup of size $|Q|, k = 5$.

In colums are values calculated for 5, 10, 20, 30 and 40 iterations of the cipher.

Brute force attack against unknown quasigroup operations

- In a brute force attack we need to

Brute force attack against unknown quasigroup operations

- In a brute force attack we need to
- Find out how many encryptions have been done

Brute force attack against unknown quasigroup operations

- In a brute force attack we need to
- Find out how many encryptions have been done
- Find the quasigroup(s) used to encrypt the message

Brute force attack against unknown quasigroup operations

- In a brute force attack we need to
- Find out how many encryptions have been done
- Find the quasigroup(s) used to encrypt the message
- Find out the leader(s) used to encrypt the message

Brute force attack against unknown quasigroup operations

- In a brute force attack we need to
- Find out how many encryptions have been done
- Find the quasigroup(s) used to encrypt the message
- Find out the leader(s) used to encrypt the message
- Find out the order in which the quasigroups were used to encrypt the message

Brute force attack against unknown quasigroup operations

- Aim for differential analysis is to gain some non neglible advantage over brute force attack.

Brute force attack against unknown quasigroup operations

- Aim for differential analysis is to gain some non neglible advantage over brute force attack.
- The amount of different latin squares of order k is $\geq \prod_{k=1}^{n} k!$

Brute force attack against unknown quasigroup operations

- Aim for differential analysis is to gain some non neglible advantage over brute force attack.
- The amount of different latin squares of order k is $\geq \prod_{k=1}^{n} k!$
- There is no (known) formula for deciding the amount of latin squares of certain order.

Brute force attack against unknown quasigroup operations

- Aim for differential analysis is to gain some non neglible advantage over brute force attack.
- The amount of different latin squares of order k is $\geq \prod_{k=1}^{n} k!$
- There is no (known) formula for deciding the amount of latin squares of certain order.
- Experiments show that there are 576 latin squares of order 4, more than 10^{90} of order 16.

Brute force attack against unknown quasigroup operations

- Aim for differential analysis is to gain some non neglible advantage over brute force attack.
- The amount of different latin squares of order k is $\geq \prod_{k=1}^{n} k!$
- There is no (known) formula for deciding the amount of latin squares of certain order.
- Experiments show that there are 576 latin squares of order 4, more than 10^{90} of order 16.
- For simplicity we assume that we know how many encryptions has been done (pherhaps we can use timing attack to find this out).

Brute force attack against unknown quasigroup operations

- With chosen plaintext attack we can try to find out information about the orders of the quasigroup(s) used to encrypt the message.

Brute force attack against unknown quasigroup operations

- With chosen plaintext attack we can try to find out information about the orders of the quasigroup(s) used to encrypt the message.

- A brute force attack after this consists of trying all the possible quasigroups with all the possible leaders, succeeding within average of half the the possibities tried.

Petrozavodsk 8-9.6.2004 – p.19/31

Brute force attack against unknown quasigroup operations

- With chosen plaintext attack we can try to find out information about the orders of the quasigroup(s) used to encrypt the message.
- A brute force attack after this consists of trying all the possible quasigroups with all the possible leaders, succeeding within average of half the the possibities tried.

- Unfortunately we do not know how many possibilities there are for quasigroups of order higher than ?

Petrozavodsk 8-9.6.2004 – p.19/31

Statistical analysis of difference distributions

- In statistical analysis our aim is to find *structure* of the quasigroup(s) used.

Statistical analysis of difference distributions

- In statistical analysis our aim is to find *structure* of the quasigroup(s) used.
- General structure of a latin square of order 4 would look like

Statistical analysis of difference distributions

- In statistical analysis our aim is to find *structure* of the quasigroup(s) used.
- General structure of a latin square of order 4 would look like

$$\begin{array}{cccc} \hline x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \\ \hline \end{array}$$

Statistical analysis of difference distributions

- In statistical analysis our aim is to find *structure* of the quasigroup(s) used.
- General structure of a latin square of order 4 would look like

$$\begin{array}{cccc} \hline x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \\ \hline \end{array}$$

- where for example x_{11} = x_{22} = x_{34} = x_{43} = 01

Finding bit difference patterns (1 round)

After 1 round of encryption we can find bit differences as follows:

$$a_1 a_2 a_3 a_4 a_5 a_6 \dots$$

$$b_1 b_2 b_3 b_4 b_5 b_6 b_7 \dots$$

After 2 rounds of encryption we can find bit differences as follows:

$$a_1 a_2 a_3 a_4 a_5 a_6 \dots$$

$$b_1 b_2 b_3 b_4 b_5 b_6 b_7 \dots$$

$$c_1 c_2 c_3 c_4 c_5 c_6 c_7 \dots$$

Finding the quasigroup

- An example difference distribution on a quasigroup of order 4 might look like

Finding the quasigroup

- An example difference distribution on a quasigroup of order 4 might look like

$I\backslash O$	00	01	10	11
00	16	0	0	0
01	0	8	8	0
10	0	8	8	0
11	0	0	0	16

Finding the quasigroup

- An example difference distribution on a quasigroup of order 4 might look like

$I\backslash O$	00	01	10	11
00	16	0	0	0
01	0	8	8	0
10	0	8	8	0
11	0	0	0	16

- We can now construct the quasigroup structure using the difference table.

Finding the quasigroup

- Here we have a charasteristic of 11 $\rightarrow$ 11 with probability 1, which gives us

Finding the quasigroup

- Here we have a charasteristic of 11 → 11 with probability 1, which gives us

00	x_{12}	x_{13}	11
01	x_{22}	x_{23}	10
10	x_{32}	x_{33}	01
11	x_{42}	x_{43}	00

Finding the quasigroup

- Here we have a charasteristic of 11 → 11 with probability 1, which gives us

00	x_{12}	x_{13}	11
01	x_{22}	x_{23}	10
10	x_{32}	x_{33}	01
11	x_{42}	x_{43}	00

- For difference 01 we have two charasteristics with equal probability, namely 01 → 01 and 01 → 10.

00	01	x_{13}	11
01	x_{22}	x_{23}	10
10	x_{32}	x_{33}	01
11	x_{42}	x_{43}	00

00	10	x_{13}	11
01	x_{22}	x_{23}	10
10	x_{32}	x_{33}	01
11	x_{42}	x_{43}	00

Finding the quasigroup

- For determining the value of x_{22} we have to do the same ending up with four possible tables:

00	01	10	11
01	00	11	10
10	x_{32}	x_{33}	01
11	x_{42}	x_{43}	00

00	10	01	11
01	00	11	10
10	x_{32}	x_{33}	01
11	x_{42}	x_{43}	00

00	01	10	11
01	11	00	10
10	x_{32}	x_{33}	01
11	x_{42}	x_{43}	00

00	10	01	11
01	11	00	10
10	x_{32}	x_{33}	01
11	x_{42}	x_{43}	00

Finding the quasigroup

- For x_{32} we have two choices
- This means eight possible structures four of which would violate the definition of latin squares

00	01	10	11
01	00	11	10
10	11	00	01
11	10	01	00

00	10	01	11
01	00	11	10
10	11	00	01
11	01	10	00

00	01	10	11
01	11	00	10
10	00	11	01
11	10	01	00

00	10	01	11
01	11	00	10
10	00	11	01
11	01	10	00

Attack with known structure

- Knowing the structure of the latin square reduces brute force complexity to n!

Attack with known structure

- Knowing the structure of the latin square reduces brute force complexity to n!
- For example for order 4 this is 24 while amount of latin squares of order 4 is 576.

Petrozavodsk 8-9.6.2004 – p.26/31

Attack with known structure

- Knowing the structure of the latin square reduces brute force complexity to n!
- For example for order 4 this is 24 while amount of latin squares of order 4 is 576.
- The task of finding the latin square from difference distributions comes more difficult as the order increases.

Petrozavodsk 8-9.6.2004 – p.26/31

Attack with known structure

- Knowing the structure of the latin square reduces brute force complexity to n!
- For example for order 4 this is 24 while amount of latin squares of order 4 is 576.
- The task of finding the latin square from difference distributions comes more difficult as the order increases.
- Some cases are "simple" and some can be impossible

Petrozavodsk 8-9.6.2004 – p.26/31

Using several quasigroups

- When several squares are used the success of the attack depends on the distributions.

Using several quasigroups

- When several squares are used the success of the attack depends on the distributions.
- Using two latin squares with distribution tables of, for example, form

	00	01	10	11
00	16	0	0	0
01	0	8	0	8
10	0	0	16	8
11	0	8	0	8

	00	01	10	11
00	16	0	0	0
00	0	16	0	0
00	0	0	8	8
00	0	0	8	8

Using several quasigroups

- will give the most uniform distribution, such as

Using several quasigroups

- will give the most uniform distribution, such as

00	12	0	0	0
01	0	3	2	2
10	0	3	4	5
11	0	1	2	3

Using several quasigroups

- will give the most uniform distribution, such as

00	12	0	0	0
01	0	3	2	2
10	0	3	4	5
11	0	1	2	3

- which reveals nothing about the structures of the groups.

Conclusions

- In some cases it is possible to gain considerable advantage with differential analysis compared to straight brute force attack
- It is useful to consider a difference distribution of a quasigroup before using it
- If small group is used, one should use more than one group
- These groups should be selected so that combined they produce difference distribution that has no charasteristics with probability 1.

Future work

- One could generate a general algorithm for finding the quasigroup based on the difference distribution
- What happens if one uses different, but isomorphic quasigroups (ie. quasigroups with same structure) for encryption and decryption?

Thank you

Questions?