Loong: A family of Involutional Lightweight Block Cipher

YAnonymous

Department of Computer Science
Indian Institute of Technology Bhilai

November 28, 2020

Outline

Why Lightweight Cryptography?

- Lightweight Cryptography become one of the very important field from last few years, provides security with very low computation power
- Deals with RFID tags, wireless sensor network, small internet enabled application that are called **IoT** arises
- Should have less amount of code and easy to implement on hardware.

Why Lightweight Cryptography?

- Lightweight Cryptography become one of the very important field from last few years, provides security with very low computation power
- Deals with RFID tags, wireless sensor network, small internet enabled application that are called **IoT** arises
- Should have less amount of code and easy to implement on hardware.

Why Lightweight Cryptography?

- Lightweight Cryptography become one of the very important field from last few years, provides security with very low computation power
- Deals with RFID tags, wireless sensor network, small internet enabled application that are called **IoT** arises
- Should have less amount of code and easy to implement on hardware.

Lightweight Block Ciphers

- There are two structure of block ciphers
  - Substitution Permutation Network(SPN)
  - Feistel Network
- Feistel Network have same encryption and decryption process but inefficient in terms of number of round, ex. DES
- SPN are efficient in terms of number of round but hard to implement because have different encryption and decryption process. That's leads to
  - big circuits for hardware, needs more space and energy
  - Too much code for smart card, makes it infeasible
  - For Software, cipher and it's inverse needs different code

Lightweight Block Ciphers

- There are two structure of block ciphers
  - Substitution Permutation Network(SPN)
  - Feistel Network
- Feistel Network have same encryption and decryption process but inefficient in terms of number of round, ex. DES
- SPN are efficient in terms of number of round but hard to implement because have different encryption and decryption process. That's leads to
  - big circuits for hardware, needs more space and energy
  - Too much code for smart card, makes it infeasible
  - For Software, cipher and it's inverse needs different code

Lightweight Block Ciphers

- There are two structure of block ciphers
  - Substitution Permutation Network(SPN)
  - Feistel Network
- Feistel Network have same encryption and decryption process but inefficient in terms of number of round, ex. DES
- SPN are efficient in terms of number of round but hard to implement because have different encryption and decryption process. That's leads to
  - big circuits for hardware, needs more space and energy
  - Too much code for smart card, makes it infeasible
  - For Software, cipher and it's inverse needs different code

Lightweight Block Ciphers

- There are two structure of block ciphers
  - Substitution Permutation Network(SPN)
  - Feistel Network
- Feistel Network have same encryption and decryption process but inefficient in terms of number of round, ex. DES
- SPN are efficient in terms of number of round but hard to implement because have different encryption and decryption process. That's leads to
  - big circuits for hardware, needs more space and energy
  - Too much code for smart card, makes it infeasible
  - For Software, cipher and it's inverse needs different code

Lightweight Block Ciphers

- There are two structure of block ciphers
  - Substitution Permutation Network(SPN)
  - Feistel Network
- Feistel Network have same encryption and decryption process but inefficient in terms of number of round, ex. DES
- SPN are efficient in terms of number of round but hard to implement because have different encryption and decryption process. That's leads to
  - big circuits for hardware, needs more space and energy
  - Too much code for smart card, makes it infeasible
  - For Software, cipher and it's inverse needs different code

Lightweight Block Ciphers

- There are two structure of block ciphers
  - Substitution Permutation Network(SPN)
  - Feistel Network
- Feistel Network have same encryption and decryption process but inefficient in terms of number of round, ex. DES
- SPN are efficient in terms of number of round but hard to implement because have different encryption and decryption process. That's leads to
  - big circuits for hardware, needs more space and energy
  - Too much code for smart card, makes it infeasible
  - For Software, cipher and it's inverse needs different code

Loong : A LightWeight Block Cipher

- Loong is a LightWeight Block Cipher that uses SPN structure
- Loong uses SPN construction that have same encryption and decryption process

Lets see how ...

Loong : A LightWeight Block Cipher

- Loong is a LightWeight Block Cipher that uses SPN structure
- Loong uses SPN construction that have same encryption and decryption process

Lets see how ...

Loong : A LightWeight Block Cipher

- Loong is a LightWeight Block Cipher that uses SPN structure
- Loong uses SPN construction that have same encryption and decryption process

Lets see how ...

Outline

Specifications

- Three members of Loong cipher: **Loong-64**, **Loong-80** and **loong-128**
- Block size of Loong is 64-bit
- Number of rounds are 16, 20 and 32 for 64-bit, 80-bit and 128-bit key respectively.

Specifications

- Three members of Loong cipher: **Loong-64**, **Loong-80** and **loong-128**
- Block size of Loong is 64-bit
- Number of rounds are 16, 20 and 32 for 64-bit, 80-bit and 128-bit key respectively.

Specifications

- Three members of Loong cipher: **Loong-64**, **Loong-80** and **loong-128**
- Block size of Loong is 64-bit
- Number of rounds are 16, 20 and 32 for 64-bit, 80-bit and 128-bit key respectively.

Encryption

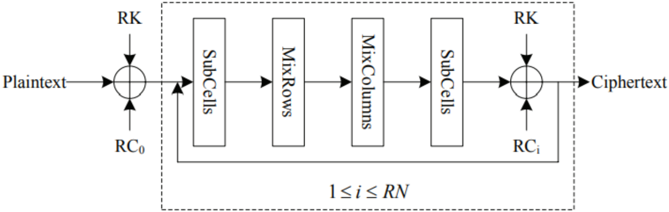


Figure: Encryption Function

Subcells

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(x)	C	A	D	3	E	B	F	7	9	8	1	5	0	2	4	6

Table: S-Box

- Special property of **involutive** in the SBox
- It means the S-Box is its own inverse.
- $x = S(S(x))$

Subcells

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(x)	C	A	D	3	E	B	F	7	9	8	1	5	0	2	4	6

Table: S-Box

- Special property of **involutive** in the SBox
- It means the S-Box is its own inverse.
- $x = S(S(x))$

Subcells

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(x)	C	A	D	3	E	B	F	7	9	8	1	5	0	2	4	6

Table: S-Box

- Special property of **involutive** in the SBox
- It means the S-Box is its own inverse.
- $x = S(S(x))$

MixRows

- In MixRow Operation, the state is post-multiplied by a diffusion matrix M.
- The matrix multiplication is performed in finite field $GF(2^4)$ where the irreducible polynomial is $x^4 + x + 1$.

$$state \leftarrow \begin{pmatrix} state_0 & state_1 & state_2 & state_3 \\ state_4 & state_5 & state_6 & state_7 \\ state_8 & state_9 & state_{10} & state_{11} \\ state_{12} & state_{13} & state_{14} & state_{15} \end{pmatrix} \times \begin{pmatrix} 1 & 4 & 9 & 13 \\ 4 & 1 & 13 & 9 \\ 9 & 13 & 1 & 4 \\ 13 & 9 & 4 & 1 \end{pmatrix}$$

MixRows

- In MixRow Operation, the state is post-multiplied by a diffusion matrix M.
- The matrix multiplication is performed in finite field $GF(2^4)$ where the irreducible polynomial is $x^4 + x + 1$.

$$state \leftarrow \begin{pmatrix} state_0 & state_1 & state_2 & state_3 \\ state_4 & state_5 & state_6 & state_7 \\ state_8 & state_9 & state_{10} & state_{11} \\ state_{12} & state_{13} & state_{14} & state_{15} \end{pmatrix} \times \begin{pmatrix} 1 & 4 & 9 & 13 \\ 4 & 1 & 13 & 9 \\ 9 & 13 & 1 & 4 \\ 13 & 9 & 4 & 1 \end{pmatrix}$$

MixColumns

- In MixColumn Operation, the state is pre-multiplied by a diffusion matrix M'.
- The matrix multiplication is performed in finite field $GF(2^4)$ where the irreducible polynomial is $x^4 + x + 1$.

$$state \leftarrow \begin{pmatrix} 13 & 9 & 4 & 1 \\ 9 & 13 & 1 & 4 \\ 4 & 1 & 13 & 9 \\ 1 & 4 & 9 & 13 \end{pmatrix} \times \begin{pmatrix} state_0 & state_1 & state_2 & state_3 \\ state_4 & state_5 & state_6 & state_7 \\ state_8 & state_9 & state_{10} & state_{11} \\ state_{12} & state_{13} & state_{14} & state_{15} \end{pmatrix}$$

MixColumns

- In MixColumn Operation, the state is pre-multiplied by a diffusion matrix M'.
- The matrix multiplication is performed in finite field $GF(2^4)$ where the irreducible polynomial is $x^4 + x + 1$.

$$state \leftarrow \begin{pmatrix} 13 & 9 & 4 & 1 \\ 9 & 13 & 1 & 4 \\ 4 & 1 & 13 & 9 \\ 1 & 4 & 9 & 13 \end{pmatrix} \times \begin{pmatrix} state_0 & state_1 & state_2 & state_3 \\ state_4 & state_5 & state_6 & state_7 \\ state_8 & state_9 & state_{10} & state_{11} \\ state_{12} & state_{13} & state_{14} & state_{15} \end{pmatrix}$$

Round Constants

- 6-bit affine linear-feedback shift register(LFSR)

$$(rc_5, rc_4, rc_3, rc_2, rc_1, rc_0) \leftarrow (rc_4, rc_3, rc_2, rc_1, rc_0, rc_5 \oplus rc_4 \oplus 1)$$

- Round Constants are intialized to 0.
- The adding(XOR) of the round constants in the AddRoundKey is arranged as follows:

$$\begin{bmatrix} 0 & 0 & 0 & (rc_5 \| rc_4 \| rc_3) \\ 0 & 0 & 1 & (rc_2 \| rc_1 \| rc_0) \\ 0 & 0 & 2 & (rc_5 \| rc_4 \| rc_3) \\ 0 & 0 & 4 & (rc_2 \| rc_1 \| rc_0) \end{bmatrix}$$

Round Constants

- 6-bit affine linear-feedback shift register(LFSR)

  $$(rc_5, rc_4, rc_3, rc_2, rc_1, rc_0) \leftarrow (rc_4, rc_3, rc_2, rc_1, rc_0, rc_5 \oplus rc_4 \oplus 1)$$

- Round Constants are intialized to 0.
- The adding(XOR) of the round constants in the AddRoundKey is arranged as follows:

$$\begin{bmatrix} 0 & 0 & 0 & (rc_5||rc_4||rc_3) \\ 0 & 0 & 1 & (rc_2||rc_1||rc_0) \\ 0 & 0 & 2 & (rc_5||rc_4||rc_3) \\ 0 & 0 & 4 & (rc_2||rc_1||rc_0) \end{bmatrix}$$

Round Constants

- 6-bit affine linear-feedback shift register(LFSR)

$$(rc_5, rc_4, rc_3, rc_2, rc_1, rc_0) \leftarrow (rc_4, rc_3, rc_2, rc_1, rc_0, rc_5 \oplus rc_4 \oplus 1)$$

- Round Constants are intialized to 0.
- The adding(XOR) of the round constants in the AddRoundKey is arranged as follows:

$$\begin{bmatrix} 0 & 0 & 0 & (rc_5||rc_4||rc_3) \\ 0 & 0 & 1 & (rc_2||rc_1||rc_0) \\ 0 & 0 & 2 & (rc_5||rc_4||rc_3) \\ 0 & 0 & 4 & (rc_2||rc_1||rc_0) \end{bmatrix}$$

AddRoundKey

The 64-bit key of Loong-64 is arranged into a round key matrix as:

$$RK \leftarrow \begin{bmatrix} k_0 & k_1 & k_2 & k_3 \\ k_4 & k_5 & k_6 & k_7 \\ k_8 & k_9 & k_{10} & k_{11} \\ k_{12} & k_{13} & k_{14} & k_{15} \end{bmatrix}$$

AddRoundKey

The 80-bit key of Loong-80 is arranged into two round key matrix:

$$RK_0 \leftarrow \begin{bmatrix} k_0 & k_1 & k_2 & k_3 \\ k_4 & k_5 & k_6 & k_7 \\ k_8 & k_9 & k_{10} & k_{11} \\ k_{12} & k_{13} & k_{14} & k_{15} \end{bmatrix} \qquad RK_1 \leftarrow \begin{bmatrix} k_{16} & k_{17} & k_{18} & k_{19} \\ k_0 & k_1 & k_2 & k_3 \\ k_4 & k_5 & k_6 & k_7 \\ k_8 & k_9 & k_{10} & k_{11} \end{bmatrix}$$

AddRoundKey

The 128-bit key of Loong-128 is arranged into two round key matrix as:

$$RK_0 \leftarrow \begin{bmatrix} k_0 & k_1 & k_2 & k_3 \\ k_4 & k_5 & k_6 & k_7 \\ k_8 & k_9 & k_{10} & k_{11} \\ k_{12} & k_{13} & k_{14} & k_{15} \end{bmatrix} \qquad RK_1 \leftarrow \begin{bmatrix} k_{16} & k_{17} & k_{18} & k_{19} \\ k_{20} & k_{21} & k_{22} & k_{23} \\ k_{24} & k_{25} & k_{26} & k_{27} \\ k_{28} & k_{29} & k_{30} & k_{31} \end{bmatrix}$$

AddRoundKey

For 64-bit key, the AddRoundKey operation is as:

$$state \leftarrow state \oplus RK \oplus RC_i \quad (0 \leq i \leq RN) \tag{1}$$

For 80-bit and 128-bit key, the AddRoundKey Operation is as:

$$state \leftarrow state \oplus RK_{i \bmod 2} \oplus RC_i \quad (0 \leq i \leq RN) \tag{2}$$

Decryption

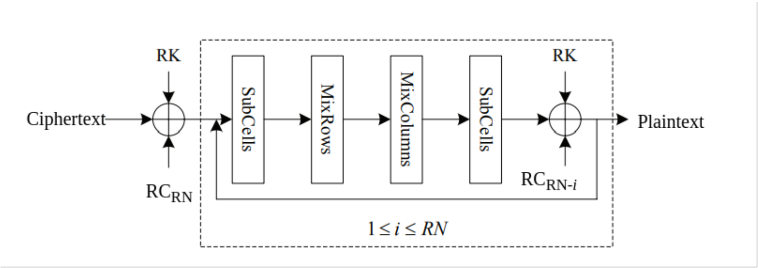


Figure: Decryption Function

Comparison Between Encryption & Decryption

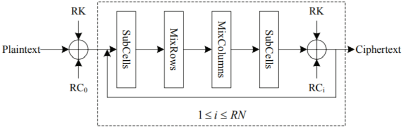


Figure: Encryption Process

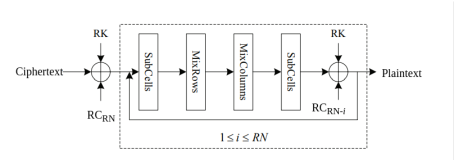


Figure: Decryption Process

Figure: Similar Encryption and decryption Process

The process of Decryption is same as encryption in Loong. The only difference is round constants are used in reverse order.

Why Loong?

- Encryption and decryption process is same.
- Easy to implement on hardware.
- Provides Two times confusion in one round of cipher.
- Highly security against cryptanalysis, especially the differential attack and linear attack.

Why Loong?

- Encryption and decryption process is same.
- Easy to implement on hardware.
- Provides Two times confusion in one round of cipher.
- Highly security against cryptanalysis, especially the differential attack and linear attack.

Why Loong?

- Encryption and decryption process is same.
- Easy to implement on hardware.
- Provides Two times confusion in one round of cipher.
- Highly security against cryptanalysis, especially the differential attack and linear attack.

Why Loong?

- Encryption and decryption process is same.
- Easy to implement on hardware.
- Provides Two times confusion in one round of cipher.
- Highly security against cryptanalysis, especially the differential attack and linear attack.

Outline

DDT-Difference Distribution Table

I/O	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	16	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1	-	2	4	-	2	2	2	-	2	-	-	-	-	-	2	-
2	-	4	-	-	4	-	-	-	2	2	-	-	2	2	-	-
3	-	-	-	-	2	-	4	2	-	4	-	-	2	-	-	2
4	-	2	4	2	2	2	-	-	-	2	2	-	-	-	-	-
5	-	2	-	-	2	-	-	4	2	-	2	2	2	-	-	-
6	-	2	-	4	-	-	-	2	2	-	-	-	-	4	2	-
7	-	-	-	2	-	4	2	-	-	-	-	2	2	2	-	2
8	-	2	2	-	-	2	2	-	-	2	-	2	2	-	2	-
9	-	-	2	4	2	-	-	-	2	2	-	2	2	-	-	-
10	-	-	-	-	2	2	-	-	-	-	2	2	-	4	-	4
11	-	-	-	-	-	2	-	2	2	2	2	2	-	2	-	2
12	-	-	2	2	-	2	-	2	2	2	-	-	2	-	2	-
13	-	-	2	-	-	-	4	2	-	-	4	2	-	-	2	-
14	-	2	-	-	-	-	2	-	2	-	-	-	2	2	2	4
15	-	-	-	2	-	-	-	2	-	-	4	2	-	-	4	2

Table: DDT of S-Box

The maximum differential probability of the SBox is $\frac{4}{16} = \frac{1}{4}$

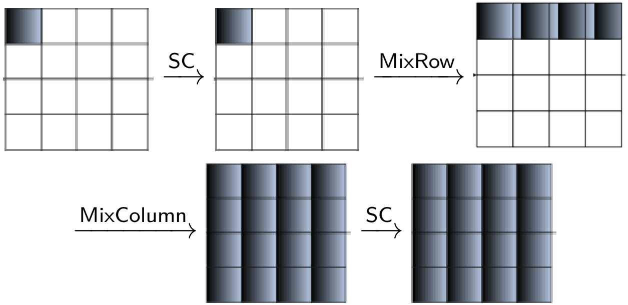
LAT-Linear Approximation Table

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

Table: LAT of S-Box

The highest absolute value observed in the LAT Table is 4 indicating it occurs with probability $\frac{12}{16} = \frac{3}{4}$.

Diffusion Spread in Loong



After one round diffusion spread in whole 64-bit block.

Outline

Differential & Linear Attack

- Finding total number of active S-Box
- Construct a mixed integer linear programming (MILP).
- All variables corresponding to the inputs of the SubCells operations are summed in the objective function, so this corresponds to the number of active S-boxes[1]

[1]N. Mouha, Q. Wang, D. Gu, and B. Preneel, "Differential and linear cryptanalysis using mixed-integer linear programming," in Proc. 7th Int. Conf. Inf. Secur. Cryptol., vol. 7537, 2011, pp. 57–76.

Differential & Linear Attack

- Finding total number of active S-Box
- Construct a mixed integer linear programming (MILP).
- All variables corresponding to the inputs of the SubCells operations are summed in the objective function, so this corresponds to the number of active S-boxes[1]

[1] N. Mouha, Q. Wang, D. Gu, and B. Preneel, "Differential and linear cryptanalysis using mixed-integer linear programming," in Proc. 7th Int. Conf. Inf. Secur. Cryptol., vol. 7537, 2011, pp. 57–76.

Differential & Linear Attack

- Finding total number of active S-Box
- Construct a mixed integer linear programming (MILP).
- All variables corresponding to the inputs of the SubCells operations are summed in the objective function, so this corresponds to the number of active S-boxes[1]

[1] N. Mouha, Q. Wang, D. Gu, and B. Preneel, "Differential and linear cryptanalysis using mixed-integer linear programming," in Proc. 7th Int. Conf. Inf. Secur. Cryptol., vol. 7537, 2011, pp. 57–76.

Differential & Linear Attack

$$\begin{bmatrix} x_0 & x_4 & x_8 & x_{12} \\ x_1 & x_5 & x_9 & x_{13} \\ x_2 & x_6 & x_{10} & x_{14} \\ x_3 & x_7 & x_{11} & x_{15} \end{bmatrix} \xrightarrow{\text{SC}} \begin{bmatrix} x_0 & x_4 & x_8 & x_{12} \\ x_1 & x_5 & x_9 & x_{13} \\ x_2 & x_6 & x_{10} & x_{14} \\ x_3 & x_7 & x_{11} & x_{15} \end{bmatrix} \xrightarrow{\text{Mix Row}} \begin{bmatrix} x_{16} & x_{17} & x_{18} & x_{19} \\ x_{20} & x_{21} & x_{22} & x_{23} \\ x_{24} & x_{25} & x_{26} & x_{27} \\ x_{28} & x_{29} & x_{30} & x_{31} \end{bmatrix}$$

$$\xrightarrow{\text{Mix column}} \begin{bmatrix} x_{32} & x_{36} & x_{40} & x_{44} \\ x_{33} & x_{37} & x_{41} & x_{45} \\ x_{34} & x_{38} & x_{42} & x_{46} \\ x_{35} & x_{39} & x_{43} & x_{47} \end{bmatrix} \xrightarrow{\text{SC}} \begin{bmatrix} x_{32} & x_{36} & x_{40} & x_{44} \\ x_{33} & x_{37} & x_{41} & x_{45} \\ x_{34} & x_{38} & x_{42} & x_{46} \\ x_{35} & x_{39} & x_{43} & x_{47} \end{bmatrix}$$

Differential & Linear Attack

- The MixRows can be constrained by the following linear equations:

$$x_0 + x_1 + x_2 + x_3 + x_{16} + x_{20} + x_{24} + x_{28} - 5d_0 \geq 0 \quad (3)$$

$$x_4 + x_5 + x_6 + x_7 + x_{17} + x_{21} + x_{25} + x_{29} - 5d_1 \geq 0 \quad (4)$$

$$x_8 + x_9 + x_{10} + x_{11} + x_{18} + x_{22} + x_{26} + x_{30} - 5d_2 \geq 0 \quad (5)$$

$$x_{12} + x_{13} + x_{14} + x_{15} + x_{19} + x_{23} + x_{27} + x_{31} - 5d_3 \geq 0 \quad (6)$$

Differential & Linear Attack

- The MixColumns can be constrained by the following linear equations:

$$x_{16} + x_{17} + x_{18} + x_{19} + x_{32} + x_{33} + x_{34} + x_{35} - 5d_4 \geq 0 \quad (7)$$

$$x_{20} + x_{21} + x_{22} + x_{23} + x_{36} + x_{37} + x_{38} + x_{39} - 5d_5 \geq 0 \quad (8)$$

$$x_{24} + x_{25} + x_{26} + x_{27} + x_{40} + x_{41} + x_{42} + x_{43} - 5d_6 \geq 0 \quad (9)$$

$$x_{28} + x_{29} + x_{30} + x_{31} + x_{44} + x_{45} + x_{46} + x_{47} - 5d_7 \geq 0 \quad (10)$$

Differential & Linear Attack

Ciphers	DC/LC	Rounds								
		1	2	3	4	5	6	7	8	9
Loong-64	DC	8	16	24	32	40	48	56	64	**72**
	LC	8	16	24	32	40	48	56	64	**72**
AES-128	DC	1	5	9	25	26	30	34	50	51
	LC	1	5	9	25	26	30	34	50	51
PRESENT-80	DC	1	2	4	6	10	12	14	16	18
	LC	1	2	3	4	5	6	7	8	9
GIFT-64	DC	1	2	3	5	7	10	13	16	18
	LC	1	2	3	5	7	9	12	15	18
RECTANGLE	DC	1	2	3	4	6	8	11	13	14
	LC	1	2	3	4	6	8	10	12	14

Figure: The number of active S-boxes.

- Loong has more active S-Box Compare to other Block ciphers.

Differential & Linear Attack

Ciphers	DC/LC	Rounds								
		1	2	3	4	5	6	7	8	9
Loong-64	DC	8	16	24	32	40	48	56	64	**72**
	LC	8	16	24	32	40	48	56	64	**72**
AES-128	DC	1	5	9	25	26	30	34	50	51
	LC	1	5	9	25	26	30	34	50	51
PRESENT-80	DC	1	2	4	6	10	12	14	16	18
	LC	1	2	3	4	5	6	7	8	9
GIFT-64	DC	1	2	3	5	7	10	13	16	18
	LC	1	2	3	5	7	9	12	15	18
RECTANGLE	DC	1	2	3	4	6	8	11	13	14
	LC	1	2	3	4	6	8	10	12	14

Figure: The number of active S-boxes.

- Loong has more active S-Box Compare to other Block ciphers.

Differential & Linear Attack

- Loong-64 has 16 rounds, and its differential probability is 2^{-256} and its bias of linear probability is 2^{-129} .
- Loong-80 has 20 rounds, and its differential probability is 2^{-320} and its bias of linear probability is 2^{-161} .
- And Loong-128 has 32 rounds, and its differential probability is 2^{-512} and its bias of linear probability is 2^{-257}

Therefore, Loong has high security and we believe Loong is enough to resist against differential and linear attacks.

Differential & Linear Attack

- Loong-64 has 16 rounds, and its differential probability is 2^{-256} and its bias of linear probability is 2^{-129} .
- Loong-80 has 20 rounds, and its differential probability is 2^{-320} and its bias of linear probability is 2^{-161} .
- And Loong-128 has 32 rounds, and its differential probability is 2^{-512} and its bias of linear probability is 2^{-257}

Therefore, Loong has high security and we believe Loong is enough to resist against differential and linear attacks.

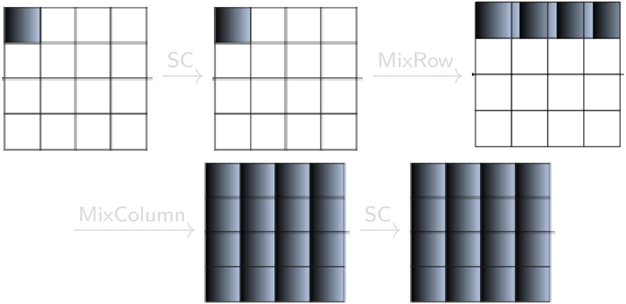
Differential & Linear Attack

- Loong-64 has 16 rounds, and its differential probability is 2^{-256} and its bias of linear probability is 2^{-129} .
- Loong-80 has 20 rounds, and its differential probability is 2^{-320} and its bias of linear probability is 2^{-161} .
- And Loong-128 has 32 rounds, and its differential probability is 2^{-512} and its bias of linear probability is 2^{-257}

Therefore, Loong has high security and we believe Loong is enough to resist against differential and linear attacks.

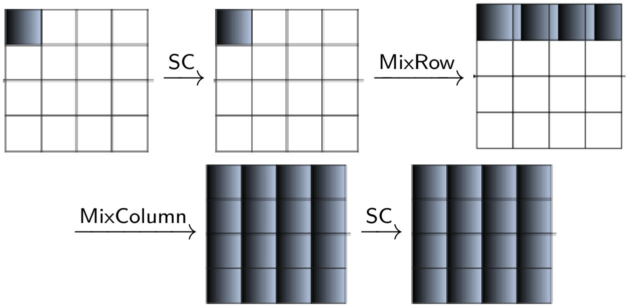
Related-key Attack

- Analyze the probability of related-key differential characteristics.
- Differences are inserted in the round key input the round key will be active in Loong-64.



Related-key Attack

- Analyze the probability of related-key differential characteristics.
- Differences are inserted in the round key input the round key will be active in Loong-64.



Related-key Attack

- Loong-64 has at least $(RN\ /2) \times 8$ active S-boxes in related-key differential path.
- And Loong 8-round differential probability is 2^{-64}

Related-key Attack

- Loong-64 has at least $(RN/2) \times 8$ active S-boxes in related-key differential path.
- And Loong 8-round differential probability is 2^{-64}

Algebraic Attack

- Used to find multi-variable equation for non-linear component
- Loong uses sub-cell two times in one round
- Each sub-cell uses 16(block size) s-boxes
- Loong uses 32(16*2) sboxes in each round
- Loong-64 uses 512 sboxes
- Loong-80 uses 640 sboxes
- Loong-128 uses 1024 sboxes

Algebraic Attack

- Used to find multi-variable equation for non-linear component
- Loong uses sub-cell two times in one round
- Each sub-cell uses 16(block size) s-boxes
- Loong uses 32(16*2) sboxes in each round
- Loong-64 uses 512 sboxes
- Loong-80 uses 640 sboxes
- Loong-128 uses 1024 sboxes

Algebraic Attack

- Used to find multi-variable equation for non-linear component
- Loong uses sub-cell two times in one round
- Each sub-cell uses 16(block size) s-boxes
- Loong uses 32(16*2) sboxes in each round
- Loong-64 uses 512 sboxes
- Loong-80 uses 640 sboxes
- Loong-128 uses 1024 sboxes

Algebraic Attack

- Used to find multi-variable equation for non-linear component
- Loong uses sub-cell two times in one round
- Each sub-cell uses 16(block size) s-boxes
- Loong uses 32(16*2) sboxes in each round
- Loong-64 uses 512 sboxes
- Loong-80 uses 640 sboxes
- Loong-128 uses 1024 sboxes

Algebraic Attack

- Used to find multi-variable equation for non-linear component
- Loong uses sub-cell two times in one round
- Each sub-cell uses 16(block size) s-boxes
- Loong uses 32(16*2) sboxes in each round
- Loong-64 uses 512 sboxes
- Loong-80 uses 640 sboxes
- Loong-128 uses 1024 sboxes

Algebraic Attack

- Used to find multi-variable equation for non-linear component
- Loong uses sub-cell two times in one round
- Each sub-cell uses 16(block size) s-boxes
- Loong uses 32(16*2) sboxes in each round
- Loong-64 uses 512 sboxes
- Loong-80 uses 640 sboxes
- Loong-128 uses 1024 sboxes

Algebraic Attack

- Used to find multi-variable equation for non-linear component
- Loong uses sub-cell two times in one round
- Each sub-cell uses 16(block size) s-boxes
- Loong uses 32(16*2) sboxes in each round
- Loong-64 uses 512 sboxes
- Loong-80 uses 640 sboxes
- Loong-128 uses 1024 sboxes

Algebraic Attack

- A 4×4 S-box can be described by 21 quadratic equations of 8 input/output-bit variables over[2]

[2] N. T. Courtois and J. Pieprzyk, "Cryptanalysis of block ciphers with overdefined systems of equations," in Proc. 8th Int. Conf. Theory Appl. Cryptol. Inf. Secur., vol. 2501, 2002, pp. 267–287

Algebraic Attack

- Using that fact here is the comparison of Loong with other Lightweight ciphers

The Algebraic Comparison information in different ciphers				
Ciphers	Rounds	S-boxes	Quadratic equations	Variables
Loong-64	16	512	10752	4096
Loong-80	20	**640**	**13440**	**5120**
Loong-128	32	**1024**	**21504**	**8192**
KLEIN-64	12	240	5040	1920
MIBS-80	32	320	6720	2560
PRESENT-80	31	527	11067	4216

Meet-in-The-Middle Attack

- Known Plain-text Attack (KPA)
- Analytical method to balancing time and memory
- Loong round function has faster diffusion effect
- Fully diffused in two round of encryption
- In the worst case, meet in the middle attack can analyze up to 7(2+2+3=7) rounds.

Secure against MITM Attack

Meet-in-The-Middle Attack

- Known Plain-text Attack (KPA)
- Analytical method to balancing time and memory
- Loong round function has faster diffusion effect
- Fully diffused in two round of encryption
- In the worst case, meet in the middle attack can analyze up to 7(2+2+3=7) rounds.

Secure against MITM Attack

Meet-in-The-Middle Attack

- Known Plain-text Attack (KPA)
- Analytical method to balancing time and memory
- Loong round function has faster diffusion effect
- Fully diffused in two round of encryption
- In the worst case, meet in the middle attack can analyze up to 7(2+2+3=7) rounds.

Secure against MITM Attack

Meet-in-The-Middle Attack

- Known Plain-text Attack (KPA)
- Analytical method to balancing time and memory
- Loong round function has faster diffusion effect
- Fully diffused in two round of encryption
- In the worst case, meet in the middle attack can analyze up to 7(2+2+3=7) rounds.

Secure against MITM Attack

Meet-in-The-Middle Attack

- Known Plain-text Attack (KPA)
- Analytical method to balancing time and memory
- Loong round function has faster diffusion effect
- Fully diffused in two round of encryption
- In the worst case, meet in the middle attack can analyze up to 7(2+2+3=7) rounds.

Secure against MITM Attack

Outline

Properties of Loong

- S-box involutive in nature
- By storing one matrix we can use Mix-Column and Mix-Row, as
  $M * M' = I$

Properties of Loong

- S-box involutive in nature
- By storing one matrix we can use Mix-Column and Mix-Row, as
  $M * M' = I$

Outline

Loong is Lightweight

- Easy to implement, need less code
- Smart card can use it as need less code
- Hardware can use it as encryption and decryption circuit is same

Loong is Lightweight

- Easy to implement, need less code
- Smart card can use it as need less code
- Hardware can use it as encryption and decryption circuit is same

Loong is Lightweight

- Easy to implement, need less code
- Smart card can use it as need less code
- Hardware can use it as encryption and decryption circuit is same

Loong is secure (Cryptanalysis)

- Loong is new but secure for every know attack
- provide 2 times more confusion then AES(Advance encryption standard)
- from the same matrix can do mixRow and mixColumn

Loong is secure (Cryptanalysis)

- Loong is new but secure for every know attack
- provide 2 times more confusion then AES(Advance encryption standard)
- from the same matrix can do mixRow and mixColumn

Loong is secure (Cryptanalysis)

- Loong is new but secure for every know attack
- provide 2 times more confusion then AES(Advance encryption standard)
- from the same matrix can do mixRow and mixColumn

Loong is efficient

- The inverse cipher of Loong can absolutely reuse the codes and circuitries to reduce resources in software and hardware implementation
- Can be used in both serialize and round-based architecture

Loong is efficient

- The inverse cipher of Loong can absolutely reuse the codes and circuitries to reduce resources in software and hardware implementation
- Can be used in both serialize and round-based architecture

Thank You

Team Members

- Anmol Sagar(11840180)
- Ashutosh Garg(11840250)
- Prince Kumar Pansari(11840860)

Implementation Info

- Github Link: https://github.com/princepansari/loong_cipher