DES (Data Encryption Standard)

Table of contents

* Introduction
* Goal
* Table of progress
* DES encryption run description
* Conclusion

Introduction :

This algorithm on this report talks about the DES, Data Encryption Standard, based on symmetric-key encryption. Created and developed by IBM in 1970.

With a specific message of 64 bits block (16 hex chars), the DES algorithm will execute different operations to transform the initial block into another block of same size (content changes of course).

Particularity of DES: It uses 56 bits + 8 parity bits, it’s used for the main key to allow data decryption.

This encryption algorithm was not considered as secured in 1999 due to key weakness, related to 56-bit key size. The AES (Advanced Encryption Standard) algorithm replaced DES, and it’s officially the most secured encryption algorithm with secret key of 16, 24 or 32 bits for 128, 192 or 256-bit block.

Goal :

The goal of this LAB is to understand the DES and implement the full steps of this algorithm. I have freedom of the programming language, I choose the C++ with oriented object structure for the implementation.

Table of progress :

The goal of this LAB is to understand the DES and implement the full steps of this algorithm. I have freedom of the programming language, I choose the C++ with oriented object structure for the implementation.

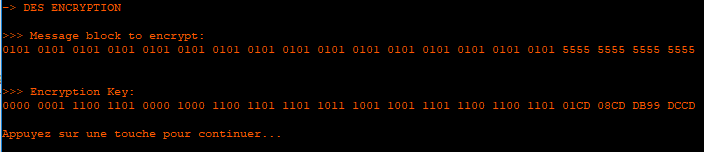
|  |  |  |
| --- | --- | --- |
| Tasks | Not finished | Finished |
| Initial permutation |  | X |
| L0 & R0 |  | X |
| Expansion of Ri |  | X |
| Subkey Generation |  | X |
| SBox output |  | X |
| Result permutation |  | X |
| XOR operations |  | X |
| Results |  | X |
| String message to 64-bit blocks (option for more complete experience) | X |  |

DES Encryption description :

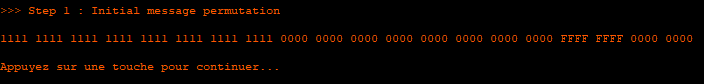
1. The structure:

For the implementation, I have created a DES class (des.h, des.cpp) with private attributes and public methods. The data tables are initialized on data\_tables.h and messages examples on data\_messages\_examples.h, these files were provided with the project subject.

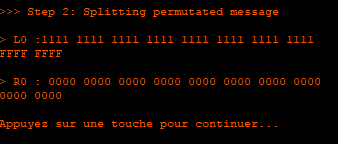
1. Message block and key:



1. Step 1: Initial permutation



1. Step 2: Splitting permutated message



1. Step 3: Expansion of Ri
2. Step 3.1: Subkey generation
3. XOR operation between Ri after expansion and subkey generation