**ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ**

Ордена Трудового Красного Знамени

федеральное государственное бюджетное образовательное учреждение высшего образования

**Московский технический университет связи и информатики**

Кафедра информационной безопасности

Курсовая работа

Студент:

Масленников П. В.

Факультет: ИТ

Группа: БВТ1501

Москва 2019

1. MASLENN

2. Двоичное представление

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Символ | M | A | S | L | E | N | N |
| Двоичн. | 01001101 | 01000001 | 01010011 | 01001100 | 01000101 | 01001110 | 01001110 |

3. Циклический XOR

Шаг 1:

00000000 xor 01001101 = 01001101

Шаг 2:

01001101 rol 1 = **10011010**

Шаг 3:

**10011010** xor 01000001 = 00001100

Шаг 4:

00001100 rol 1 = **00011000**

Шаг 5:

**00011000** xor 01010011 = 01001011

Шаг 6:

01001011 rol 1 = **10010110**

Шаг 7:

**10010110** xor 01001100 = 11011010

Шаг 8:

11011010 rol 1 = **10110101**

Шаг 9:

**10110101** xor 01000101 = 11110000

Шаг 10:

11110000 rol 1 = **11100001**

Шаг 11:

**11100001** xor 01001110 = 10101111

Шаг 12:

10101111 rol 1 = **01011111**

Шаг 13:

**01011111** xor 01001110 = 00010001

Ответ: 01001101 01000001 01010011 01001100 01000101 01001110 01001110 00010001

4. DEA

Пароль:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Символ | A | L | M | A | N | E | N |
| Двоичн. | 01000001 | 01001100 | 01001101 | 01000001 | 01001110 | 01000101 | 01001110 |

Делаем перестановку бит в 64-разрядном блоке исходных данных:

00100110 10010110 00000000 00000000 11101001 10011110 00100001 11111110

Шаг 1 итерация 1:

Делим полученный блок на две 32-разрядные части:

L1 = 00100110 10010110 00000000 00000000

R1 = 11101001 10011110 00100001 11111110

Шаг 2 итерация 1:

Расширяем 56-битный ключ до 64-битного, вставляя каждые 7 бит дополнительный бит чётности:

Ключ: 01000001 01001100 01001101 01000001 01001110 01000101 01001110

Ответ: 01000001 10100110 00010010 10101001 00010100 01110010 00010100 10011100

Шаг 3 итерация 1:

Делаем Выбор 1 из таблицы:

01000001 10100110 00010010 10101001 00010100 01110010 00010100 10011100

Ответ: 1001000 0011001 0001001 0110001 1000010 0010101 0000101 1110001

Шаг 4 итерация 1:

Делим результат из шага 3 на 2 части:

C0 = 1001000 0011001 0001001 0110001

D0 = 1000010 0010101 0000101 1110001

Шаг 5 итерация 1:

Делаем битовый сдвиг каждой из частей согласно таблице битовых сдвигов:

C1 = 0010000 0110010 0010010 1100011

D1 = 0000100 0101010 0001011 1100011

Шаг 6 итерация 1:

Соединяем две части:

0010000 0110010 0010010 1100011 0000100 0101010 0001011 1100011

Шаг 7 итерация 1:

Делаем Выбор 2 из таблицы:

001110 000100 000001 110101 000110 110001 100010 000010

Шаг 8 итерация 1:

Расширяем L1 до 48 бит согласно таблице расширения:

L1 = 00100110 10010110 00000000 00000000

Ответ: 00000000 00000000 00000000 00110101 00101011 00001000

Шаг 9 итерация 1:

Делаем XOR с данными из Шаг 7 итерация 1 и Шаг 8 итерация 1:

00000000 00000000 00000000 00110101 00101011 00001000 XOR

00111000 01000000 01110101 00011011 00011000 10000010

00111000 01000000 01110101 00101110 00110011 10001010

Шаг 10 итерация 1:

Разделим на 6 разрядов:

001110 000100 000001 110101 001011 100011 001110 001010

S-матрица:

8 8 0 5 11 3 13 6

Двоичный вид:

10001000 00000101 10110011 11010110

Шаг 11 итерация 1:

Делаем перестановку разрядов согласно таблице:

11000011 00001001 11011011 11000000

Шаг 12 итерация 1:

Делаем XOR с результатом из Шаг 11 итерация 1 и R1:

11000011 00001001 11011011 11000000 XOR

11101001 10011110 00100001 11111110

00101010 10010111 11111010 00111110

Шаг 13 итерация 1:

Формируем результат текущего шага:

11101001 10011110 00100001 11111110 00101010 10010111 11111010 00111110

Шаг 1 итерация 2:

Делим полученный блок на две 32-разрядные части:

L2 = 11101001 10011110 00100001 11111110

R2 = 00101010 10010111 11111010 00111110

Шаг 2 итерация 2:

Делаем битовый сдвиг каждой из частей из первой итерации согласно таблице битовых сдвигов:

C2 = 0100000 1100100 0100101 1000110

D2 = 0001000 1010100 0010111 1000110

Шаг 3 итерация 2:

Соединяем две части:

0100000 1100100 0100101 1000110 0001000 1010100 0010111 1000110

Шаг 4 итерация 2:

Делаем Выбор 2 из таблицы:

010000 100011 011001 010001 101001 100101 010000 000100

Шаг 5 итерация 2:

Расширяем L2 до 48 бит согласно таблице расширения:

L2 = 00100110 10010110 00000000 00000000

Ответ: 101111 111111 110000 001000 001111 110011 110010 101110

Шаг 6 итерация 2:

Делаем XOR с данными из Шаг 4 итерация 2 и Шаг 5 итерация 2:

01000010 00110110 01010001 10100110 01010100 00000100 XOR

10111111 11111100 00001000 00111111 00111100 10101110

11111101 11001010 01011001 10011001 01101000 10101010

Шаг 7 итерация 2:

Разделим на 6 разрядов:

111111 011100 101001 011001 100110 010110 100010 101010

S-матрица:

13 5 6 1 11 7 4 12

Двоичный вид:

11010101 01100001 10110111 01001100

Шаг 8 итерация 2:

Делаем перестановку разрядов согласно таблице:

11001001 00100011 00011111 01001111

Шаг 9 итерация 2:

Делаем XOR с результатом из Шаг 8 итерация 2 и R2:

11001001 00100011 00011111 01001111 XOR

00101010 10010111 11111010 00111110

11100011 10110100 11100101 01110001

Шаг 10 итерация 2:

Формируем результат текущего шага:

00101010 10010111 11111010 00111110 11100011 10110100 11100101 01110001

Шаг 1 итерация 3:

Делим полученный блок на две 32-разрядные части:

L3 = 00101010 10010111 11111010 00111110

R3 = 11100011 10110100 11100101 01110001

Шаг 2 итерация 3:

Делаем битовый сдвиг каждой из частей из второй итерации согласно таблице битовых сдвигов:

C3 = 0000011 0010001 0010110 0011001

D3 = 0100010 1010000 1011110 0011000

Шаг 3 итерация 3:

Соединяем две части:

0000011 0010001 0010110 0011001 0100010 1010000 1011110 0011000

Шаг 4 итерация 3:

Делаем Выбор 2 из таблицы:

101001 000011 111100 001000 000110 011010 000000 001011

Шаг 5 итерация 3:

Расширяем L3 до 48 бит согласно таблице расширения:

L3 = 00101010 10010111 11111010 00111110

Ответ: 001111 111000 001011 111111 111101 010010 101010 101000

Шаг 6 итерация 3:

Делаем XOR с данными из Шаг 4 итерация 3 и Шаг 5 итерация 3:

10100100 00111111 00001000 00011001 10100000 00001011 XOR

00111111 10000010 11111111 11110101 00101010 10101000

10011011 10111101 11110111 11101100 10001010 10100011

Шаг 7 итерация 3:

Разделим на 6 разрядов:

100110 111011 110111 110111 111011 001000 101010 100011

S-матрица:

8 5 3 11 4 9 3 1

Двоичный вид:

10000101 00111011 01001001 00110001

Шаг 8 итерация 3:

Делаем перестановку разрядов согласно таблице:

00110101 11001100 00001101 00011001

Шаг 9 итерация 2:

Делаем XOR с результатом из Шаг 8 итерация 3 и R3:

00110101 11001100 00001101 00011001 XOR

11100011 10110100 11100101 01110001

11010110 01111000 11101000 01101000

Шаг 10 итерация 3:

Формируем результат текущего шага:

11100011 10110100 11100101 01110001 11010110 01111000 11101000 01101000

Шаг 1 итерация 4:

Делим полученный блок на две 32-разрядные части:

L4 = 11100011 10110100 11100101 01110001

R4 = 11010110 01111000 11101000 01101000

Шаг 2 итерация 4:

Делаем битовый сдвиг каждой из частей из второй итерации согласно таблице битовых сдвигов:

C3 = 0001100 1000100 1011000 1100100

D3 = 0001010 1000010 1111000 1100001

Шаг 3 итерация 4:

Соединяем две части:

0001100 1000100 1011000 1100100 0001010 1000010 1111000 1100001

Шаг 4 итерация 4:

Делаем Выбор 2 из таблицы:

101010 000110 101000 110010 011000 010100 011011 010000

Шаг 5 итерация 4:

Расширяем L4 до 48 бит согласно таблице расширения:

L4 = 11100011 10110100 11100101 01110001

Ответ: 110001 011101 010100 001110 100101 011011 111000 001111

Шаг 6 итерация 4:

Делаем XOR с данными из Шаг 4 итерация 4 и Шаг 5 итерация 4:

11000101 11010101 00001110 10010101 10111110 00001111 XOR

10101000 01101010 00110010 01100001 01000110 11010000

10100100 00110111 01100010 10001000 10111001 11011111

Шаг 7 итерация 4:

Разделим на 6 разрядов:

101001 000011 011101 100010 100010 001011 100111 011111

S-матрица:

4 13 15 6 2 12 8 2

Двоичный вид:

01001101 11110110 00101100 10000010

Шаг 8 итерация 4:

Делаем перестановку разрядов согласно таблице:

00010110 00100110 11110100 10001101

Шаг 9 итерация 4:

Делаем XOR с результатом из Шаг 8 итерация 4 и R4:

00010110 00100110 11110100 10001101 XOR

11010110 01111000 11101000 01101000

11000000 01011110 00011100 11100101

Шаг 10 итерация 4:

Формируем результат текущего шага:

11010110 01111000 11101000 01101000 11000000 01011110 00011100 11100101

Делаем инверсную перестановку разрядов:

01100011 11101111 11101000 00011110 10111100 01010110 00000110 01000000

Ответ: 01100011 11101111 11101000 00011110 10111100 01010110 00000110 01000000

Теперь интерпретируем каждые 8 бит как ASCII-символ:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| c | ï | è | RS | ¼ | V | ACK | @ |

RS - Record Separtator (Разделитель записей).

ACK - Acknowledge (Подтверждение).

Данные символы являются управляющими и не выводятся на экран.

Вывод

Данное шифрование нужно считать на ЭВМ или специализированном устройстве.