**ЛАБОРАТОРНАЯ РАБОТА № 5**

**Алгоритм шифрования ГОСТ 28147-89**

**1. Теоретическая часть**

**ГОСТ 28147-89** - это стандарт, принятый в 1989 году и установивший алгоритм шифрования данных, составляющих гостайну. Данный алгоритм является действующим и настоящее время на территории Российской Федерации. Он рекомендован к использованию для защиты любых данных, представленных в виде двоичного кода. Данный стандарт формировался с учетом мирового опыта, и в частности, были приняты во внимание недостатки и нереализованные возможности алгоритма DES, поэтому использование стандарта ГОСТ предпочтительнее.

**ГОСТ 28147-89** является блочным алгоритмом шифрования с *256-битным ключом* и *32 раундами*(циклами преобразования), оперирующий *64-битными блоками.* Основа алгоритма шифрования – *сеть Фейстеля* (рис.1).



Рис. 1. Схема шифрования блока данных, согласно ГОСТ 28147-89

ГОСТ 28147-89 предусматривает 3 режима шифрования (простая замена, гаммирование, гаммирование с обратной связью) и один режим выработки имитовставки. Первый из режимов шифрования предназначен для шифрования ключевой информации и не может использоваться для шифрования других данных, для этого предусмотрены два других режима шифрования. Режим выработки *имитовставки* (криптографической контрольной комбинации) предназначен для *имитозащиты* шифруемых данных, то есть для их защиты от случайных или преднамеренных несанкционированных изменений.

**1.1. Выполнение отдельного раунда шифрования** Согласно ГОСТ 28147-89, в режиме простой замены шифрование данных в каждом раунде выполняется над 64-битным блоком исходных данных с использованием 32-битного элемента ключа (подключа). Блок-схема алгоритма отдельного раунда приведена на рис. 2.



Рис. 2. Блок-схема отдельного раунда шифрования алгоритма ГОСТ 28147-89

**Шаг 0**. Определяются исходные данные для раунда шифрования:

- *N* – преобразуемый 64-битовый блок данных. В ходе выполнения раунда его младшая (*A*) и старшая (*B*) части обрабатываются как отдельные 32-битовые целые числа без знака, таким образом, можно записать *N* = (*A*, *B*);

- *X* – 32-битовый подключ (элемент ключа, получение подключей описано в п. 1.2).

**Шаг 1**. *Сложение с ключом*. Младшая половина преобразуемого блока (*А*) складывается по модулю 232 с используемым в раунде подключом *X*, результат передается на следующий шаг.

**Шаг 2**. *Поблочная замена*. 32-битовое значение, полученное на предыдущем шаге, интерпретируется как массив из восьми 4-битовых блоков кода: *S* = (*S0*, *S1*, *S2*, *S3*, *S4*, *S5*, *S6*, *S7*), причем *S0* содержит 4 самых младших, а *S7* – 4 самых старших бита *S*.

Далее значение каждого из восьми блоков заменяется новым, которое выбирается по *таблице замен* (*S-блоки*, см. таблицу 1) следующим образом: значение блока *Sm* меняется на *Sm* -ый по порядку элемент (нумерация с нуля) *m*-го узла замены (т.е. *m*-ой строки таблицы замен, нумерация также с нуля). Другими словами, в качестве замены для значения блока выбирается элемент из таблицы замен с номером строки, равным номеру заменяемого блока, и номером столбца, равным значению заменяемого блока как 4-битового целого неотрицательного числа. Отсюда становится понятным размер таблицы замен: число строк в ней равно числу 4-битовых элементов в 32-битовом блоке данных, то есть восьми, а число столбцов равно числу различных значений 4-битового блока данных, то есть 24 = 16. Пример поблочной замены приведен в п. 1.3.

После замены все восемь блоков объединяются в 32-битное слово

*S* = (*S0*, *S1*, *S2*, *S3*, *S4*, *S5*, *S6*, *S7*).

**Шаг 3**. *Циклический сдвиг на 11 бит влево*. Результат предыдущего шага сдвигается циклически на 11 бит влево (в сторону старших разрядов) и передается на следующий шаг.

**Шаг 4**. *Побитовое сложение.* Значение, полученное на шаге 3, побитно складывается по модулю 2 (она же операция «XOR», «ИСКЛЮЧАЮЩЕЕ ИЛИ») со старшей половиной преобразуемого блока (*В*).

**Шаг 5**. *Обмен частей блока*. Младшая часть преобразуемого блока сдвигается на место старшей, а на ее место помещается результат выполнения предыдущего шага (*В* = *А*, *А* = *S*).

**Шаг 6**. Полученное значение преобразованного блока возвращается как результат выполнения раунда алгоритма шифрования.

**1.2. Генерация подключей**

Для генерации подключей исходный 256-битный ключ разбивается на восемь 32-битных блоков: *K1*…*K8* (нумеруются от младших битов к старшим). Подключи *K9*…*K24* являются циклическим повторением подключей *K1*…*K8*. Подключи *K25*…*K32* являются подключами *K1*…*K8*, идущими в обратном порядке.

**1.3 Таблица замен (S-блоки)** Все восемь S-блоков могут быть различными. Фактически, они могут являться дополнительным ключевым материалом, но чаще являются параметром схемы, общим для определенной группы пользователей. Таблица замен, приведенная в таблице 1, используется в криптографических приложениях Центрального Банка РФ.



*Пример использования S-блоков:* Если значение нулевого блока данных *S0* равно 1, то он заменяется на значение 10 (нулевая строка, первый столбец). Если значение блока данных *S1* равно 3, то он заменяется на значение 12 (первая строка, третий столбец). Если значение блока данных *S6* равно 15, то он заменяется на значение 12 (шестая строка, пятнадцатый столбец).

**1.4. Расшифровывание данных** Расшифровывание выполняется так же, как и шифрование, но инвертируется порядок подключей *Ki*.

**1.5. Пример расчета первого раунда шифрования по ГОСТ 28147-89**

Исходные данные – текст «ШИФРОВКА». Ключ – «ИВАНОВ».

Для шифрования по ГОСТ 28147-89 блок данных должен иметь 64 бита. Если исходный текст записать в кодировке ASCII (1 символ – 1 байт), то он составит 64 бита (8 символов \* 8 бит = 64). Ключ, по ГОСТ 28147-89, должен иметь размер 256 бит (32 байта). Следовательно, исходный ключ «ИВАНОВ» требуется расширить до 32 символов. Самый простой способ – повторить ключ до 32 символов («ИВАНОВИВАНОВИВАНОВИВАНОВ ИВАНОВИВ»).

**Шаг 0**. Выделим младшую и старшую 32-битные части блока исходных данных: *А* – «ШИФР», *В* – «ОВКА». Запишем исходный текст в виде двоичных ASCII кодов (таблицы 2 и 3). Для удобства дальнейших преобразований порядок символов в таблицах 2 и 3 - обратный (справа налево).





В качестве подключа на первый раунд *К1* берется первые 4 символа (32 бита) расширенного ключа – «ИВАН». Запишем подключ в виде двоичных ASCII кодов - таблица 4 (порядок символов обратный).

**

**Шаг 1**. Сложение с ключом: *S* = (*A* + *К1*) mod 232. Примечание: сложение по модулю 232 – это обычное двоичное сложение, только при условии, что если в результате сложения происходит переполнение, то все что выходит за разрядную сетку (больше 32 бит) отбрасывается.



**Шаг 2**. Поблочная замена. 32-битовое значение *S*, полученное на шаге 1, интерпретируется как массив из восьми 4-битовых блоков кода: *S* = (*S0*, *S1*, *S2*, *S3*, *S*4, *S*5, *S*6, *S*7) (таблица 6).



Используя таблицу замен (таблица 1) произведем замены блоков данных *Sm* (*m* = 0.. 7).

*S0* = 0. Смотрим 0-ю строку таблицы 1 и находим в ней 0-е по счету число – 4.

*S1* = 10. Смотрим 1-ю строку таблицы 1 и находим в ней 10-е по счету число – 8.

*S2* = 11. Смотрим 2-ю строку таблицы 1 и находим в ней 11-е по счету число – 7.

*S3* = 8. Смотрим 3-ю строку таблицы 1 и находим в ней 8-е по счету число – 14.

*S4* = 5. Смотрим 4-ю строку таблицы 1 и находим в ней 5-е по счету число – 15.

*S5* = 9. Смотрим 5-ю строку таблицы 1 и находим в ней 9-е по счету число – 6.

*S6* = 14. Смотрим 6-ю строку таблицы 1 и находим в ней 14-е по счету число – 2.

*S7* = 9. Смотрим 7-ю строку таблицы 1 и находим в ней 9-е по счету число – 2.

Таким образом, после замены *S* примет вид, как в таблице 7.



**Шаг 3.** Циклический сдвиг на 11 бит влево. Результат шага 2 сдвигается циклически на 11 бит влево (в сторону старших разрядов). Результат сдвига приведен в таблице 8.



**Шаг 4**. Побитовое сложение по модулю 2. Значение *S*, полученное на шаге 3, побитно складывается по модулю 2 со старшей частью шифруемого блока (*В*). Результат сложения приведен в таблице 9.



**Шаг 5.** Обмен частей блока: *В* = *А*, *А* = *S*. Результат обмена приведен в таблице 10.



**Шаг 6.** Полученное значение преобразованного блока возвращается как результат выполнения раунда алгоритма шифрования (в таблицах 11, 12 приведены представления блоков в тестовый вид).





**2. Задание на лабораторную работу**

***Задание 1****:* для заданных 64 битов данных – в текстовом виде «ШИФРОВКА» (или «D8C8D4D0CEC2 CAC0» в шестнадцатеричном коде), с помощью ключа состоящему из Вашей фамилии (повторенное до размера в 32 символа (256 бит)), произвести один основной шаг (раунд) криптопреобразования по ГОСТ 28147-89. Привести пошаговые расчеты произведенных криптопреобразований в электронной или бумажной форме.

***Задание 2:***разработайте программу, выполняющую шифрование данных алгоритмом ГОСТ 28147-89. Программа должна выполнять шифрование данных (произвольного вида) хранящихся в указанном пользователем файле.

**Таблица 1.**  **Символы кириллицы (альтернативная кодовая таблица ASCII)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Сим-л | Дес. | Двоич. | Сим-л | Дес. | Двоич. |
| А | 192 | 11000000 | а | 224 | 11100000 |
| Б | 193 | 11000001 | б | 225 | 11100001 |
| В | 194 | 11000010 | в | 226 | 11100010 |
| Г | 195 | 11000011 | г | 227 | 11100011 |
| Д | 196 | 11000100 | д | 228 | 11100100 |
| Е | 197 | 11000101 | е | 229 | 11100101 |
| Ж | 198 | 11000110 | ж | 230 | 11100110 |
| З | 199 | 11000111 | з | 231 | 11100111 |
| И | 200 | 11001000 | и | 232 | 11101000 |
| Й | 201 | 11001001 | й | 277 | 11101001 |
| К | 202 | 11001010 | к | 234 | 11101010 |
| Л | 203 | 11001011 | л | 235 | 11101011 |
| М | 204 | 11001100 | м | 236 | 11101100 |
| Н | 205 | 11001101 | н | 237 | 11101101 |
| О | 206 | 11001110 | о | 238 | 11101110 |
| П | 207 | 11001111 | п | 239 | 11101111 |
| Р | 208 | 11010000 | р | 240 | 11110000 |
| С | 209 | 11010001 | с | 241 | 11110001 |
| Т | 210 | 11010010 | т | 242 | 11110010 |
| У | 211 | 11010011 | у | 243 | 11110011 |
| Ф | 212 | 11010100 | ф | 244 | 11110100 |
| Х | 213 | 11010101 | х | 245 | 11110101 |
| Ц | 214 | 11010110 | ц | 246 | 11110110 |
| Ч | 215 | 11010111 | ч | 247 | 11110111 |
| Ш | 216 | 11011000 | ш | 248 | 11111000 |
| Щ | 217 | 11011001 | щ | 249 | 11111001 |
| Ъ | 218 | 11011010 | ъ | 250 | 11111010 |
| Ы | 219 | 11011011 | ы | 251 | 11111011 |
| Ь | 220 | 11011100 | ь | 252 | 11111100 |
| Э | 221 | 11011101 | э | 253 | 11111101 |
| Ю | 222 | 11011110 | ю | 254 | 11111110 |
| Я | 223 | 11011111 | я | 255 | 11111111 |
|  |  |  | пробел | 32 | 00010000 |