Российский стандарт блочного шифрования ГОСТ 28147 - 89

В нашей стране в качестве стандарта используется технология, описанная в ГОСТе 28147-89 "Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования".

Этот ГОСТ был принят в 1989 году и с тех пор практически не изменялся.

Алгоритм шифрования был разработан в КГБ СССР еще в конце 70-х годов, однако он создавался с достаточно большим "запасом прочности". По криптостойкости он на порядок превосходил американский DES, впоследствии замененный на тройной (Triple DES), а потом на AES.

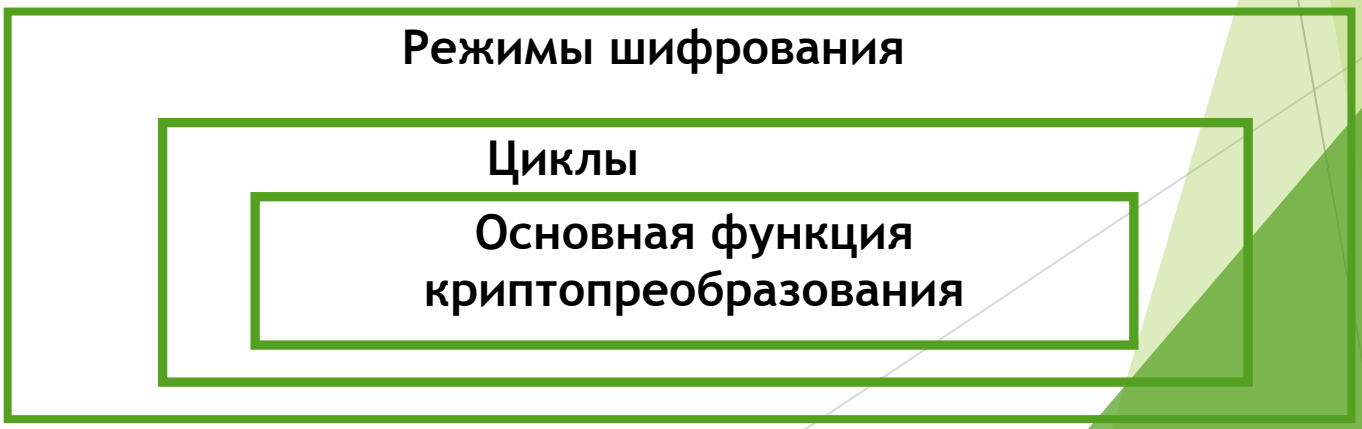
Таким образом, и на сегодняшний день криптостойкость российского стандарта вполне удовлетворяет всем современным требованиям. Вторая причина большого распространения ГОСТа 28147-89 – законодательство.

Государственные организации и многие коммерческие структуры обязаны использовать для защиты данных сертифицированные средства защиты. Однако получение сертификата возможно только в том случае, если "в указанных криптосредствах реализованы криптографические алгоритмы, объявленные государственными или отраслевыми стандартами Российской Федерации".

Иерархия алгоритмов

ГОСТ состоит из алгоритмов трех уровней:

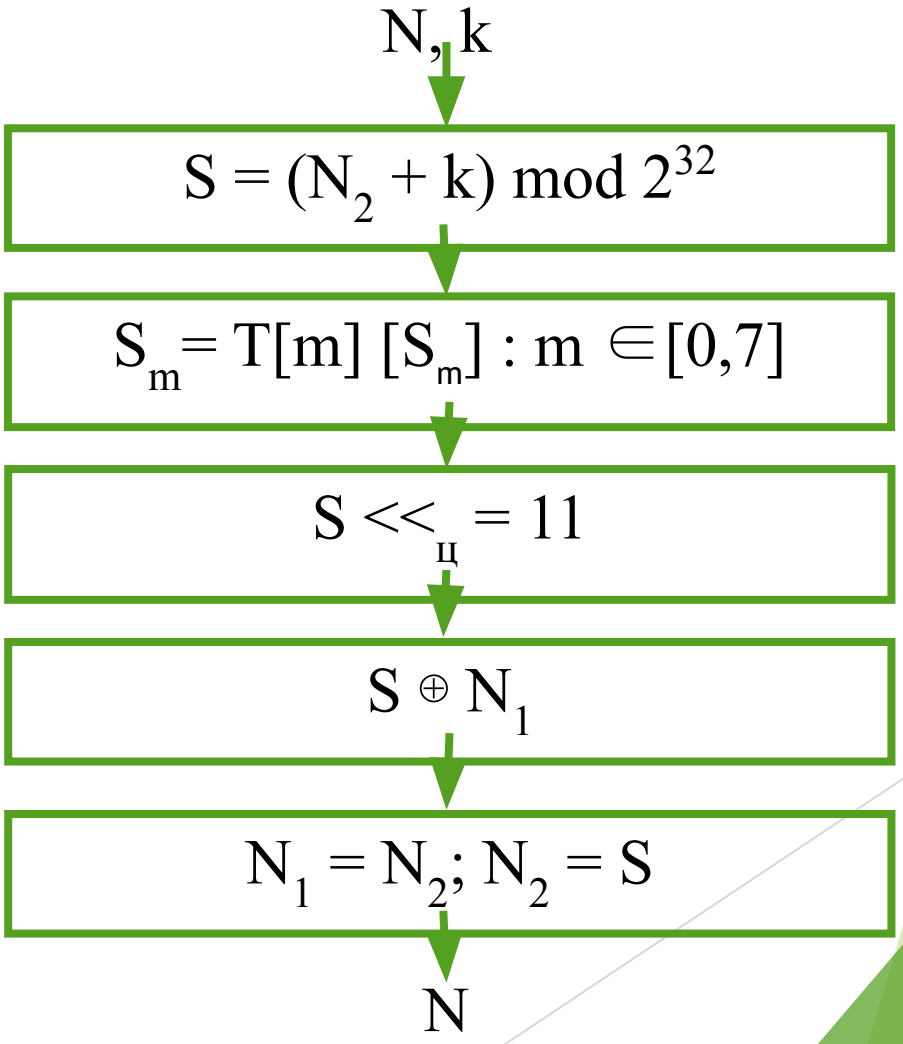
1. Практические алгоритмы, отвечающие непосредственно за шифрование (дешифрование) массивов данных.
2. Алгоритмы более низкого уровня, называемые циклами.
3. Основная функция криптопреобразования.



Ключевая информация

- Ключ является массивом из восьми элементов по 32 бита. Все восемь элементов ключа используются отдельно и рассматриваются как 32-битные целые числа без знака. Размер ключа 32*8=256 бит или 32 байта.
- Таблица замен представляет собой матрицу размером 8×16. Строки матрицы называются узлами замен. Каждый узел замены должен содержать произвольную перестановку набора значений от 0 до 15.
- Сихропосылка – 64 бита (начальное заполнение) передается в открытом виде, но в алгоритме шифрования используется результат преобразования начального заполнения

Основная функция преобразования



Основная функция преобразования

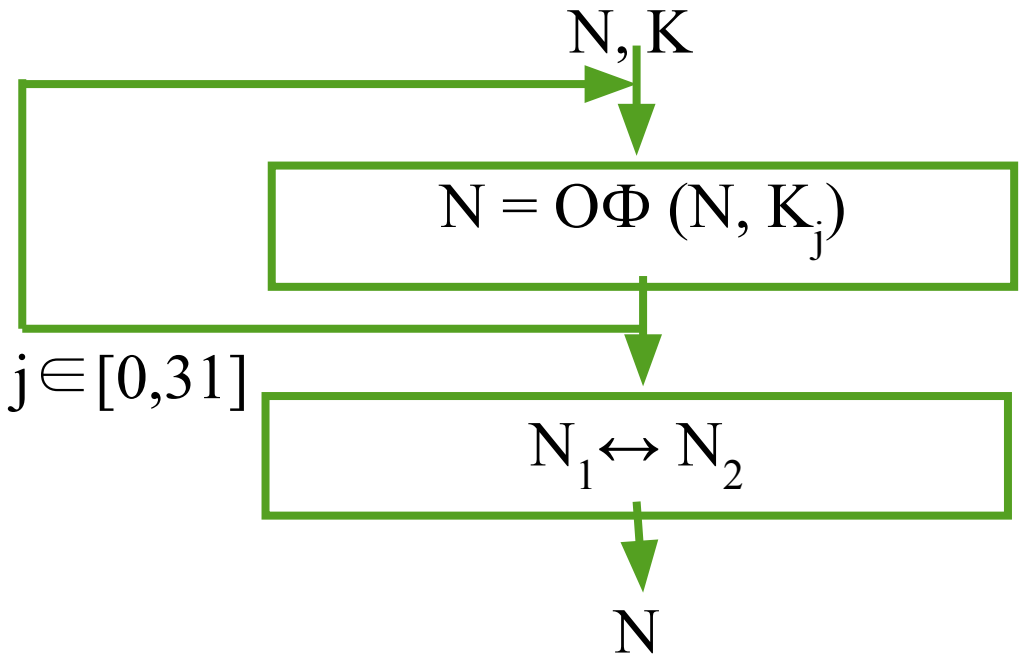
- На входе функции — две величины N и k. N — блок преобразуемой последовательности данных, состоящий из двух частей: старшей (N2) и младшей (N1), k - 32-битный элемент массива ключа.
- Старшая часть преобразуемого блока данных складывается по модулю 2^{32} с элементом ключа.
- Результат предыдущего шага подвергается преобразованию посредством таблицы замен.
- Результат замены подвергается циклическому сдвигу влево на 11 разрядов.
- Значение предыдущего шага побитово складывается посредством операции «XOR» с младшей частью преобразуемого блока.
- Старшая часть преобразуемого блока переходит на место младшей, а на ее место помещается результат предыдущего шага.
- Возврат значения N.

Преобразование по таблице замен

Блок данных (32 бита) разбивается на восемь блоков по 4 бита: $S=S_0...S_7$.

Далее каждый блок S_i меняется на значение таблицы замен, находящееся на i строке, в S_i столбце.

Циклы криптопреобразования



Пошаговая схема циклов «З» и «Р»

Циклы зашифрования (расшифрования)

- На входе цикла две величины N и K.
  N — 64-битный блок преобразуемой последовательности данных.
  K — 256-битная ключевая последовательность.
- На вход основной функции преобразования подается кодируемая последовательность и 32 битный блок ключа. Этот шаг выполняется 32 раза.
- Старшая и младшая часть кодируемой последовательности меняются местами.
- Возврат 64-битного блока N.
- Отличие циклов «З» и «Р» друг от друга состоит в последовательности перебора частей ключа.
- Для цикла «З» эта последовательность такова:

  S[32] = {0, 1, 2, 3, 4, 5, 6, 7, 0, 1, 2, 3, 4, 5, 6, 7,0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0}
- Для цикла «Р» такова:

  S[32] = {0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0,7, 6, 5, 4, 3, 2, 1, 0, 7, 6, 5, 4, 3, 2, 1, 0,}

Цикл выработки имитовставки

Цикл «И» (имитовставка) отличается от описанных выше циклов только по трем пунктам:

- основная функция преобразования вызывается не 32 раза, а только 16;
- отсутствует шаг 3 — обмен значениями между старшей и младшей частью кодируемого блока не происходит;
- последовательность перебора частей ключа для выработки имитовставки выглядит так:

  S [16] = (0, 1, 2, 3, 4, 5, 6, 7, 0, 1, 2, 3, 4, 5, 6, 7).

Режимы использования

Прямая замена

Гаммирование

Гаммирование с зацеплением

Прямая замена

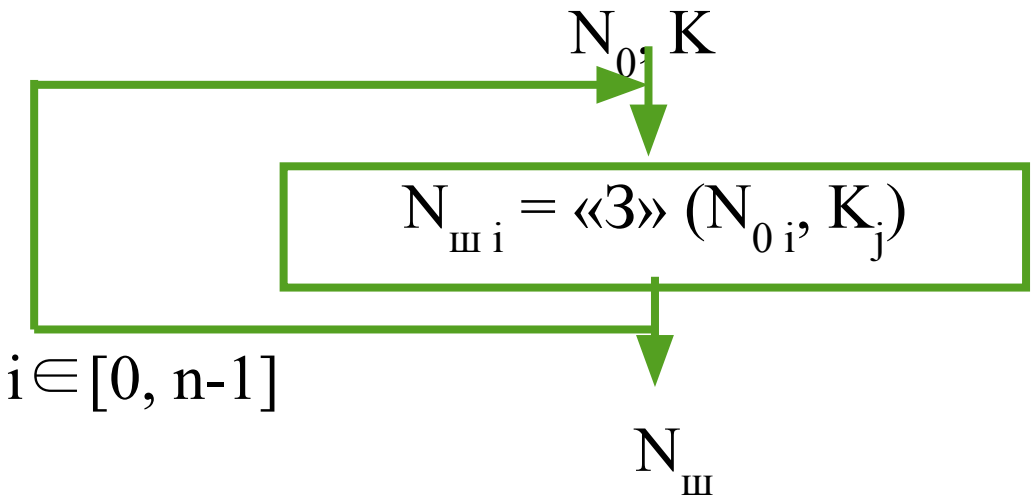


Схема шифрования
в режиме прямой замены

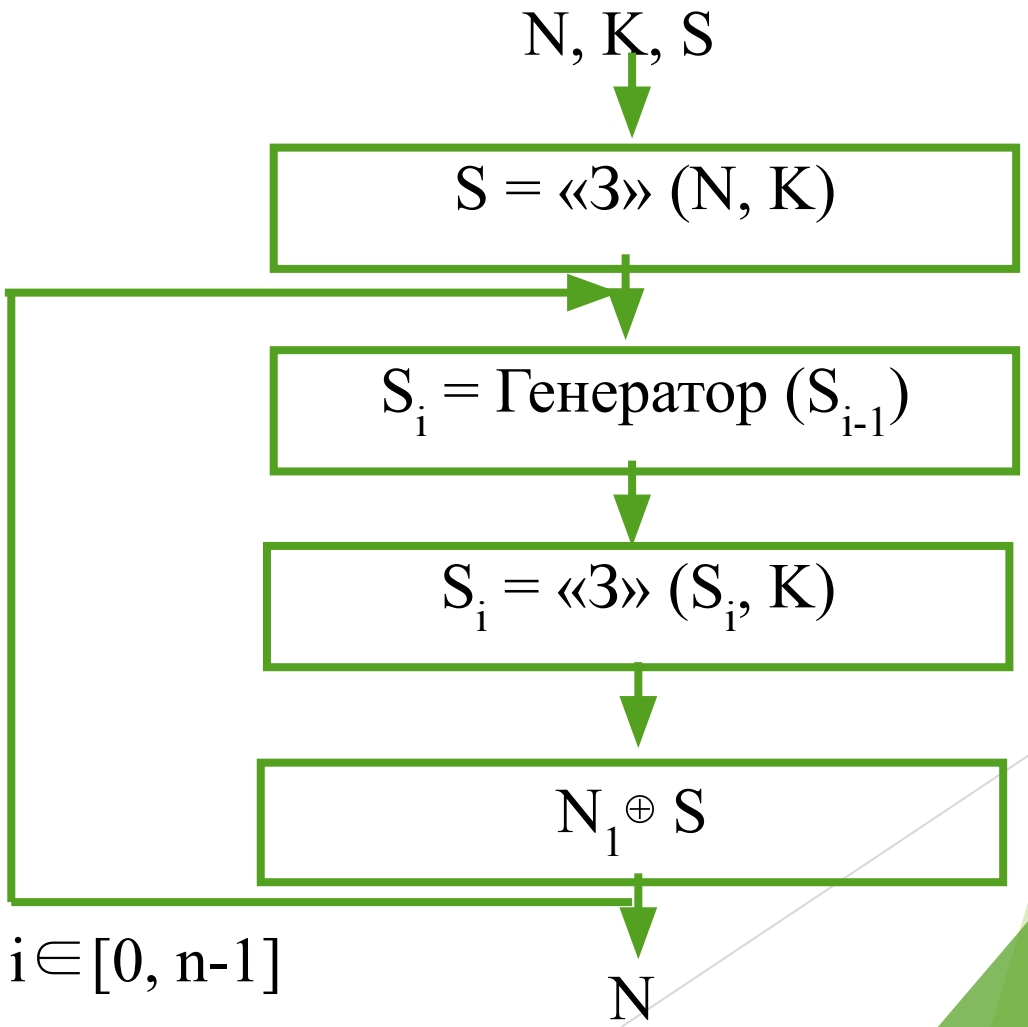
Прямая замена

- На входе: N_0 — последовательность открытых данных и К— ключ.
- Последовательный перебор всех 64-битных блоков исходного массива данных, подача их на вход цикла «3», последовательная запись полученных зашифрованных блоков в массив выходной информации Nш.
- Возврат массива зашифрованных данных Nш.

Недостатки режима прямой замены

- Шифрование двух одинаковых блоков даст идентичный результат.
- Если длина шифруемого блока данных не кратна 64 битам, ее надо дополнить до требуемой длины.

Гаммирование



Гаммирование

- На входе: N - исходная последовательность данных, K – ключ и S - синхропосылка.
- Синхропосылка подвергается шифрованию в режиме прямой замены с ключом.
- Генерируем очередное значение гаммы по рекуррентным формулам на основе ее предыдущего значения (для первого раза это синхропосылка). Сгенерированное значение гаммы шифруется методом прямой замены с ключом.
- Очередной 64-битный блок исходных данных складывается посредством операции «XOR». При этом старшая часть блока складывается со старшей частью гаммы, а младшая — с младшей.
- Возврат обработанного блока данных.

Гаммирование

- Для старшей части формула генератора гаммы такова:

$$G_{i+1} = (G_i + 16843009) \bmod 2^{32}$$

- Для младшей части формула генератора гаммы такова:

$$G_{i+1} = (G_i + 16843009) \bmod (2^{32} - 1) + 1$$

Гаммирование с зацеплением

Данный режим отличается от режима простого гаммирования способом выработки гаммы. Очередной ее элемент не генерируется по рекуррентной формуле, а получается из предыдущего блока зашифрованных данных посредством шифрования его методом прямой замены с ключом.

Первый же элемент гаммы получается из синхропосылки посредством прямой замены.

Гаммирование с зацеплением

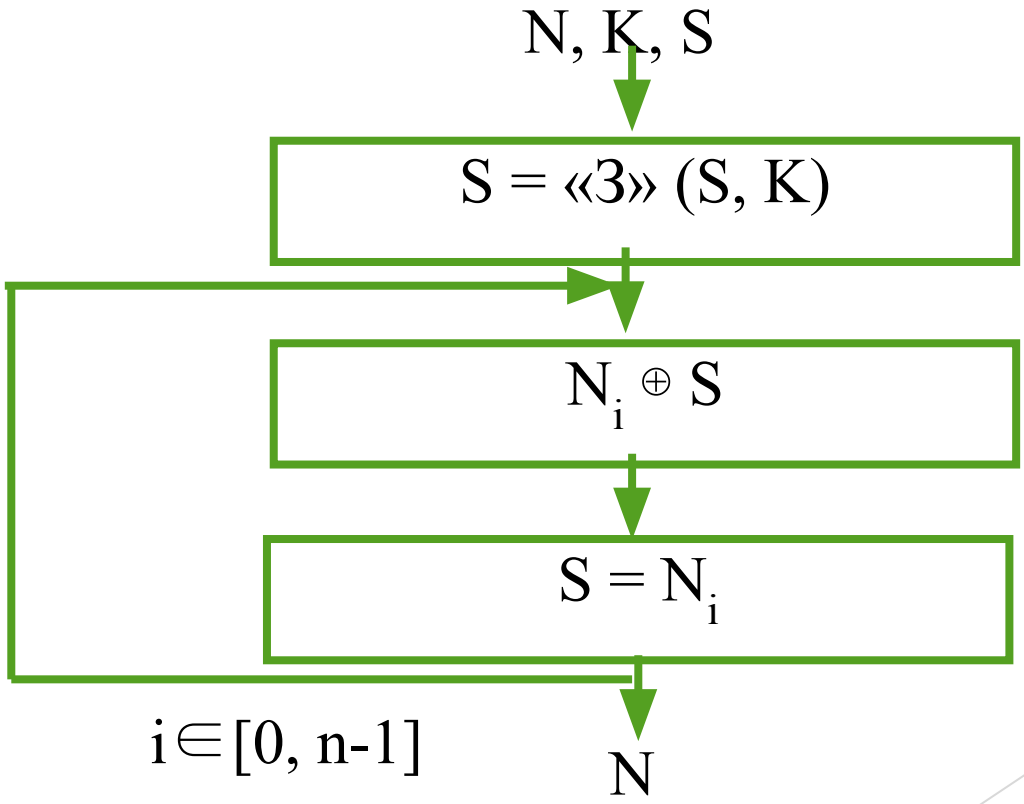


Схема шифрования в режиме гаммирования с зацеплением

Гаммирование с зацеплением

- На входе величины N - исходная последовательность данных. K - ключ и S - синхропосылка.
- Синхропосылка подвергается шифрованию алгоритмом прямой замены с ключом, в результате чего получается значение гаммы.
- Очередной 64-битный блок исходных данных складывается посредством операции «XOR» с гаммой.
- Результат предыдущего шага используется для получения очередного элемента гаммы.
- Возврат обработанного блока данных.

Выработка имитовставки

- ***Имитовставка*** — двоичная контрольная комбинация, которая зависит от открытых данных и ключевой информации.
- Цель использования имитовставки — обнаружение искажений в массиве шифруемых данных.

Выработка имитовставки

Имитовставка обладает двумя свойствами:

- без ключевой информации невозможно нахождение имитовставки для заданного открытого массива данных;
- без ключевой информации невозможен подбор данных под заданную имитовставку.

Эти свойства позволяют использовать имитовставку для контроля целостности данных.

Выработка имитовставки

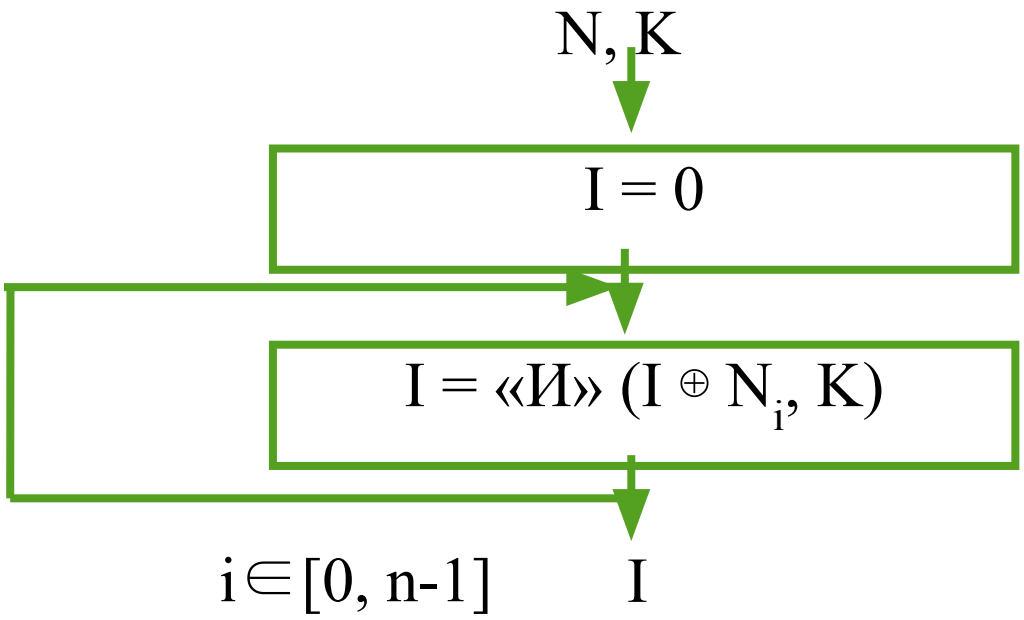


Схема алгоритма выработки имитовставки

Выработка имитовставки

- На входе величины: N — исходная последовательность данных, К—ключ.
- Начальное значение имитовставки обнуляется.
- Имитовставка складывается с очередным 64-битным блоком входных данных посредством операции «XOR». Результат подвергается преобразованию циклом «И» с ключом. Результат присваивается имитовставке.
- Возврат значения имитовставки.

Основные различия между DES и ГОСТ 28147-89

Параметры	DES	ГОСТ 28147
Процедура создания подключей	более сложная	очень проста
Ключ	56-битный	256-битный
Входы/выходы у S-boxes	6-битовые входы и 4-битовые выходы	4-битовые входы и выходы
Кол-во S-boxes	8	8
Размер S-box	больше	меньше
Перестановки	нерегулярные	11-битный циклический сдвиг влево
Изменение 1 входного бита	влияет на 1 S-box 1 раунда, который затем влияет на 2 S-boxes следующего раунда, 3 S-boxes следующего и т.д. Нужно только 5 раундов	требуется 8 раундов прежде, чем изменение одного входного бита повлияет на каждый бит результата
Кол-во раундов	16 раундов	32 раунда Более стойкий к дифференциальному и линейному криптоанализу