13. Общие принципы построения современных симметричных криптосистем. Общая характеристика блочных шифров. Криптоалгоритм DES. Криптоалгоритм ГОСТ 28147-89

Симметричные криптосистемы (с секретным ключом – secret key systems) – построены на основе сохранения в тайне ключа шифрования. Процессы зашифрования и расшифрования используют один и тот же ключ. Секретность ключа является постулатом.

Основная проблема при применении симметричных криптосистем для связи заключается в сложности передачи обоим сторонам секретного ключа. Однако данные системы обладают высоким быстродействием. Раскрытие ключа злоумышленником грозит раскрытием только той информации, что была зашифрована на этом ключе. Американский и Российский стандарты шифрования DES и ГОСТ 28.147-89, а также новый стандарт AES Rijndael – все эти алгоритмы являются представителями симметричных криптосистем.

Симметричные криптосистемы в настоящее время принято подразделять на блочные и поточные.

Блочные криптосистемы разбивают текст сообщения на отдельные блоки и затем осуществляют преобразование этих блоков с использованием ключа.

Поточные криптосистемы работают несколько иначе. На основе ключа системы вырабатывается некая последовательность – так называемая гамма, которая затем накладывается на текст сообщения. Таким образом, преобразование текста осуществляется как бы потоком по мере выработки гаммы.

Общая структура использования симметричной криптосистемы представлена на рис.13.

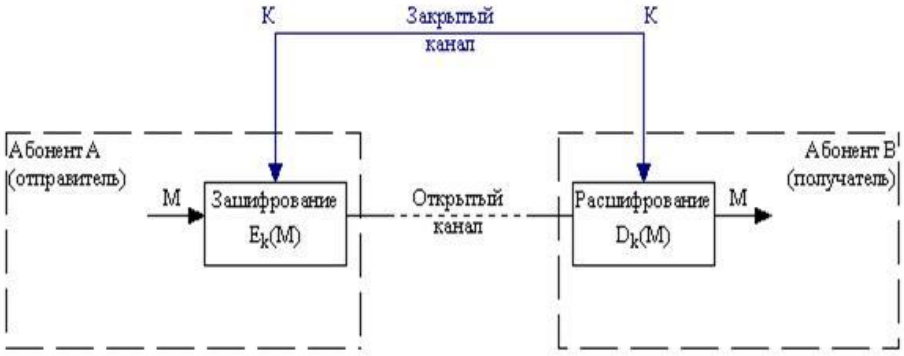

Рис. 13. Структура симметричной криптосистемы

Здесь M – открытый текст, K – секретный ключ, передаваемый по закрытому каналу, Ек(M) – операция зашифрования, а Dk(M) – операция расшифрования.

Все многообразие существующих симметричных криптографических методов можно свести к следующим классам преобразований:

1. **Моно и многоалфавитные подстановки (замены)**

Наиболее простой вид преобразований, заключающийся в замене символов исходного текста на другие (обычно того же алфавита) по более или менее сложному правилу. Для обеспечения высокой криптостойкости требуется использование больших ключей.

2. **Перестановки**

Символы исходного текста переставляются по некоторому правилу. Используется, как правило, в сочетании с другими методами.

3. **Гаммирование**

Этот метод заключается в наложении на исходный текст некоторой псевдослучайной последовательности, генерируемой на основе ключа.

4. **Блочные шифры**

Криптосистемы с секретным ключом подразделяются на два вида: **блочные** (block) и **поточные** (stream). Поточные криптосистемы работают с сообщением как с единым потоком, блочные криптосистемы представляют собой блочные (групповые) шифропреобразования. Блочная криптосистема разбивает открытый текст на последовательные блоки и

зашифровывает каждый блок с помощью одного и того же обратимого преобразования, выбранного с помощью ключа. Любое из них можно рассматривать как последовательность операций, проводимых с элементами ключа и открытого текста, а также производными от них величинами. Произвол в выборе элементов алгоритма шифрования достаточно велик, однако "элементарные" операции должны обладать хорошим криптографическими свойствами и допускать удобную техническую или программную реализацию.

Обычно используются операции:

- побитового сложения по модулю 2 двоичных векторов (XOR)
- сложение или умножение целых чисел по некоторому модулю
- перестановка битов двоичных векторов;
- табличная замена элементов двоичных векторов.

Блочные шифры

Блочные шифры оперируют с блоками открытого текста и используют простую замену блоков.

Основные процедуры, используемые при получении таких шифров, сводятся к следующему:

- **рассеивание (diffusion)** – изменение любого знака открытого текста или ключа влияет на большое число знаков шифротекста, что скрывает статистические свойства открытого текста;
- **перемешивание (confusion)** – использование преобразований, затрудняющих получение статистических зависимостей между шифротектстом и открытым текстом.

В блочных шифрах, когда длина блока достаточно велика, таблица замены становится необозримой и саму замену приходится задавать не таблицей, а неким алгоритмом преобразования.

Практически все современные блочные шифры являются композиционными – то есть состоят из композиции простых преобразований. Само по себе преобразование может и не обеспечивать нужных свойств, но их цепочка позволяет получить необходимый результат.

Американский стандарт шифрования данных DES. Стандарт шифрования данных DES (Data Encryption Standard) опубликован в 1977 г. Национальным бюро стандартов США. Он предназначен для защиты от несанкционированного доступа к важной, но не секретной информации в государственных и коммерческих организациях США.

Основные достоинства алгоритма DES:

- используется только один ключ длиной 56 бит;
- относительная простота алгоритма обеспечивает высокую скорость обработки;
- достаточно высокая стойкость алгоритма.

Алгоритм DES основан на комбинировании методов подстановки и перестановки и состоит из чередующейся последовательности блоков перестановки и подстановки. DES осуществляет шифрование 64-битовых блоков данных с помощью 64-битового ключа, в котором значащими являются 56 бит (остальные 8 бит – проверочные биты для контроля на четность). Расшифрование в DES является операцией, обратной шифрованию, и выполняется путем повторения операций шифрования в обратной последовательности. Обобщенная схема процесса шифрования в алгоритме DES показана на рисунке 6.5.

Рисунок 6.5 – Обобщенная схема шифрования в алгоритме DES

Процесс шифрования заключается в начальной перестановке битов 64-битового блока, шестнадцати циклах шифрования и, наконец, в конечной перестановке битов.

Стандарт шифрования данных (ГОСТ 28147-89). Алгоритм криптографического преобразования данных для систем обработки информации в сетях ЭВМ, отдельных вычислительных комплексах и ЭВМ был разработан в СССР и опубликован в виде государственного стандарта ГОСТ 28147-89 в 1989 году. Алгоритм криптографического преобразования данных предназначен для аппаратной и программной реализации, удовлетворяет криптографическим требованиям и не накладывает ограничений на степень секретности защищаемой информации. Алгоритм шифрования данных представляет собой 64-битовый блочный алгоритм с 256-битовым ключом.

Стандарт шифрования гост 28147-89

Краткое описание шифра

ГОСТ 28147-89 — советский и российский стандарт симметричного шифрования, введённый в 1990 году, также является стандартом СНГ. Полное название — «ГОСТ 28147-89 Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования». Блочный шифроалгоритм. При использовании метода шифрования с гаммированием, может выполнять функции поточного шифроалгоритма.

ГОСТ 28147-89 — блочный шифр с 256-битным ключом и 32 циклами преобразования, оперирующий 64-битными блоками. Основа алгоритма шифра — Сеть Фейстеля. Базовым режимом шифрования по ГОСТ 28147-89 является режим простой замены (определены также более сложные режимы гаммирование, гаммирование с обратной связью и режим имитовставки).

Принцип работы алгоритма

Алгоритм принципиально не отличается от DES. В нем также происходят циклы шифрования (их 32) по схеме Фейстеля (Рис. 2.9.).

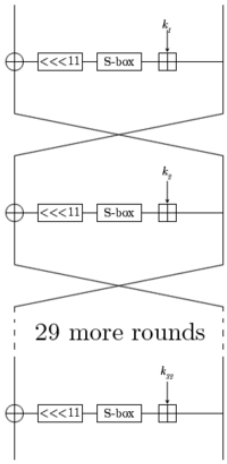

Рис. 2.9. Раунды шифрования алгоритма ГОСТ 28147-89.

Для генерации подключей исходный 256-битный ключ разбивается на восемь 32-битных блоков: $k_1 \ldots k_8$. Ключи $k_9 \ldots k_{24}$ являются циклическим повторением ключей $k_1 \ldots k_8$ (нумеруются от младших битов к старшим). Ключи $k_{25} \ldots k_{32}$ являются ключами $k_1 \ldots k_8$, идущими в обратном порядке.

После выполнения всех 32 раундов алгоритма, блоки A_{33} и B_{33} склеиваются (следует обратить внимание на то, что старшим битом становится A_{33}, а младшим - B_{33}) – результат есть результат работы алгоритма.

Функция $f(A_i,K_i)$ вычисляется следующим образом: A_i и K_i складываются по модулю 2^{32}, затем результат разбивается на восемь 4-битовых подпоследовательностей, каждая из которых поступает на вход своего узла таблицы замен (в порядке возрастания старшинства битов), называемого ниже S-блоком. Общее количество S-блоков ГОСТа — восемь, т. е. столько же, сколько и подпоследовательностей. Каждый S-блок представляет собой

перестановку чисел от 0 до 15. Первая 4-битная подпоследовательность попадает на вход первого S-блока, вторая — на вход второго и т. д. Выходы всех восьми S-блоков объединяются в 32-битное слово, затем всё слово циклически сдвигается влево (к старшим разрядам) на 11 битов. Все восемь S-блоков могут быть различными.

Фактически, они могут являться дополнительным ключевым материалом, но чаще являются параметром схемы, общим для определенной группы пользователей. В тексте стандарта указывается, что поставка заполнения узлов замены (S-блоков) производится в установленном порядке, т.е. разработчиком алгоритма. Сообщество российских разработчиков СКЗИ согласовала используемые в Интернет узлы замены. Расшифрование выполняется так же, как и зашифрование, но инвертируется порядок подключей k_i.

Режимы работы алгоритма ГОСТ 28147-89

Алгоритм ГОСТ 28147-89 имеет четыре режима работы.

1. **Режим простой замены**

   Принимает на вход данные, размер которых кратен 64-м битам. Результатом шифрования является входной текст, преобразованный блоками по 64 бита в случае зашифрования циклом «32-З», а в случае расшифрования — циклом «32-Р».

2. **Режим гаммирования**

   Принимает на вход данные любого размера, а также дополнительный 64-битовый параметр — синхропосылку. В ходе работы синхропосылка преобразуется в цикле «32-З», результат делится на две части. Первая часть складывается по модулю 2^{32} с постоянным значением 1010101_{16}. Если вторая часть равна 2^{32}-1, то её значение не меняется, иначе она складывается по модулю 2^{32}-1 с постоянным значением 1010104_{16}.

   Полученное объединением обеих преобразованных частей значение, называемое гаммой шифра, поступает в цикл «32-З», его результат порязрядно складывается по модулю 2 с 64-разрядным блоком входных данных. Если последний меньше 64-х разрядов, то лишние разряды полученного значения отбрасываются.

   Полученное значение подаётся на выход. Если ещё имеются входящие данные, то действие повторяется: составленный из 32-разрядных частей блок преобразуется по частям и так далее.

3. **Режим гаммирования с обратной связью**

   Также принимает на вход данные любого размера и синхропосылку. Блок входных данных поразрядно складывается по модулю 2 с результатом преобразования в цикле «32-З» синхропосылки. Полученное значение подаётся на выход. Значение синхропосылки заменяется в случае зашифрования выходным блоком, а в случае расшифрования — входным, то есть зашифрованным.

   Если последний блок входящих данных меньше 64 разрядов, то лишние разряды гаммы (выхода цикла «32-З») отбрасываются. Если ещё имеются входящие данные, то действие повторяется: из результата зашифрования заменённого значения образуется гамма шифра и т.д.

4. **Режим выработки имитовставки**

   Принимает на вход данные, размер которых составляет не меньше двух полных 64-разрядных блоков, а возвращает 64-разрядный блок данных, называемый имитовставкой. Временное 64-битовое значение устанавливается в 0, далее, пока имеются входные данные, оно поразрядно складывается по модулю 2 с результатом выполнения цикла «16-З», на вход которого подаётся блок входных данных. После окончания входных данных временное значение возвращается как результат.

Криптоанализ шифра

В шифре ГОСТ 28147-89 используется 256-битовый ключ и объем ключевого пространства составляет 2^{256}. Ни на одном из существующих в настоящее время компьютере общего применения нельзя подобрать ключ за время, меньшее многих сотен лет. Российский стандарт ГОСТ 28147-89 проектировался с большим запасом и по стойкости на много порядков

превосходит американский стандарт DES с его реальным размером ключа в 56 бит и объемом ключевого пространства всего 2^{56}.

Существуют атаки и на полнораундовый ГОСТ 28147—89 без каких-либо модификаций. Одна из первых открытых работ, в которых был проведен анализ алгоритма, использует слабости процедуры расширения ключа ряда известных алгоритмов шифрования. В частности, полнораундовый алгоритм ГОСТ 28147—89 может быть вскрыт с помощью дифференциального криптоанализа на связанных ключах, но только в случае использования слабых таблиц замен. 24-раундовый вариант алгоритма (в котором отсутствуют первые 8 раундов) вскрывается аналогичным образом при любых таблицах замен, однако, сильные таблицы замен делают такую атаку абсолютно непрактичной.

Отечественные ученые А.Г. Ростовцев и Е.Б. Маховенко в 2001 г. предложили принципиально новый метод криптоанализа путем формирования целевой функции от известного открытого текста, соответствующего ему шифртекста и искомого значения ключа и нахождения ее экстремума, соответствующего истинному значению ключа. Они же нашли большой класс слабых ключей алгоритма ГОСТ 28147—89, которые позволяют вскрыть алгоритм с помощью всего 4-х выбранных открытых текстов и соответствующих им шифротекстов с достаточно низкой сложностью.

В 2004 году группа специалистов из Кореи предложила атаку, с помощью которой, используя дифференциальный криптоанализ на связанных ключах, можно получить с вероятностью 91,7% 12 бит секретного ключа. Для атаки требуется 2^{35} выбранных открытых текстов и 2^{36} операций шифрования. Как видно, данная атака практически бесполезна для реального вскрытия алгоритма.

Таблица замен является долговременным ключевым элементом, то есть действует в течение гораздо более длительного срока, чем отдельный ключ. Предполагается, что она является общей для всех узлов шифрования в рамках одной системы криптографической защиты. От качества этой таблицы зависит качество шифра.

При "сильной" таблице замен стойкость шифра не опускается ниже некоторого допустимого предела даже в случае ее разглашения. И наоборот, использование "слабой" таблицы может уменьшить стойкость шифра до недопустимо низкого предела. Никакой информации по качеству таблицы замен в открытой печати России не публиковалось, однако существование "слабых" таблиц не вызывает сомнения - примером может служить "тривиальная" таблица замен, по которой каждое значение заменяется на него самого.

В ряде работ ошибочно делается вывод о том, что секретные таблицы замен алгоритма ГОСТ 28147-89 могут являться частью ключа и увеличивать его эффективную длину (что несущественно, поскольку алгоритм обладает весьма большим 256-битным ключом).