

1. Криптография, и ее место в современном обществе

1.1 Человечество изобрело большое число способов секретного письма, многие из которых были известны еще в древности. В некоторых способах тайного письма используются **физические особенности носителей информации**. Например, симпатичные чернила исчезают вскоре после написания или текста или невидимы с самого начала. Но их можно снова сделать видимыми, обработав документ специальным химическим реактивом или осветив лучами определенной части спектра, обычно - ультрафиолетом. В других способах (стенография, шифрование) определенным образом изменяется передаваемое сообщение. **Стенография** предполагает, что передаваемый текст "растворяется" в сообщении большого размера с совершенно "посторонним" смыслом. Но если взять и извлечь из него некоторые символы по определенному закону, например - каждый второй, или третий, и т.д., получим вполне конкретное тайное сообщение. **Шифрование** является преобразованием сообщения по определенным правилам, что делает его бессмысленным набором знаков для непосвященного в тайну шифра человека.

Классифицировать способы засекречивания передаваемых сообщений можно по-разному, однако определяющих факторов всего два: используются ли для засекречивания свойства материальных носителей и материальной среды передачи информации или оно осуществляется независимо от них; прятается ли секретное сообщение или оно просто делается недоступным для всех, кроме получателя.

Методы защиты данных, которые опираются исключительно на свойства самих данных и никак не связаны с особенностями их физического представления - это стенография и шифрование. Принципиально возможны два подхода к защите информации: спрятать информацию в еще большем массиве информации (как иглы в стоге сена) либо, не скрывая факт передачи сообщения, сделать его смысл недоступным для посторонних. Например, мы отправляем нашему корреспонденту почти функцию с расцветкой черно-белой картины, в которой наименее значимый бит в коде яркости каждой точки изображения будет элементом нашего тайного сообщения. Получатель письма извлекает все такие биты и составит из них "истинное" сообщение. В этом и заключается принцип стенографии. Картина, присутствующая здесь только для отвода глаз, так и останется для непосвященных простой картинкой. Стенография полезна, когда необходимо не просто передать секретное сообщение, а секретно передать секретное сообщение, то есть скрыть сам факт передачи секретного сообщения. Такой способ ведения тайной коммуникации, однако, имеет ряд недостатков:

- 1) трудно обосновать его стойкость - может оказаться, что злоумышленнику известен способ подмешивания секретных данных к "болванке" - массиву открытых данных;
- 2) резко увеличивается объем передаваемых или хранимых данных, что отрицательно сказывается на производительности систем их обработки.

Другой подход - не скрывать факт передачи сообщения, но сделать его недоступным посторонним. Для этого сообщение должно быть записано так, чтобы с его содержанием не мог ознакомиться никто за исключением самих корреспондентов - в этом, и заключается суть шифрования. И криптография возникла именно как практическая дисциплина, изучающая и разрабатывающая способы шифрования сообщений.

Криптография - это набор методов защиты **информационных взаимодействий** от отклонений от их нормального, штатного протекания, вызванных злоумышленными действиями различных субъектов, методов, базирующихся на секретных алгоритмах преобразования информации, включая алгоритмы, не являющиеся собственно секретными, но использующие секретные параметры.

В современном мире криптография решает задачи разработки и применения методов защиты процессов информационных взаимодействий различного характера, опирающихся на преобразование данных по секретным алгоритмам, включая алгоритмы, использующие секретные параметры. Термин "информационное взаимодействие" или "процесс информационного взаимодействия" обозначает такой процесс взаимодействия двух и более субъектов, основным содержанием которого является передача и/или обработка информации. По большому счету, криптографической может считаться любая функция преобразования данных, секретная сама по себе или зависящая от секретного параметра S:

$$T = f(T) \text{ или} \\ T = f(T, S).$$

1.2 Чтобы поставить задачу защиты информации, необходимо ответить на следующие вопросы. Что именно необходимо защищать? От чего? От кого?

1) **Что именно необходимо защищать?**

Сюда входит группа вопросов:

- кто является участником информационного процесса;
- каковы задачи участников информационного процесса;
- каким именно образом участники процесса выполняют стоящие перед ними задачи;

Она позволяет построить **модель информационного процесса**, нормальное течение которого обеспечивает явным путем защиты информации

2) **От чего должна быть защищена система?**

Здесь надо найти ответы на следующие вопросы:

- какие критерии нормального процесса информационного взаимодействия;

- какие возможны отклонения от нормы.

Эти вопросы решают задачу выявления отклонения от нормального течения процесса информационного взаимодействия. Они позволяют определить **критерий нормальности процесса** и **список угроз**: возможных отклонений от этой нормальности - ситуаций, которые хотелось бы сделать невозможными.

3) **От кого должна быть защищена система?**

Эта группа вопросов:

- кто может выступать в качестве **злоумышленника**, т.е. относится к тем субъектам, которые могут предпринять те или иные действия, чтобы отклонить процесс от нормы;

- каких целей добиваются злоумышленники;

- какими ресурсами могут использоваться злоумышленники, и какие действия они могут предпринять для достижения своих целей.

Ответы на эти вопросы позволяют построить **модель злоумышленника** и либо оценить надежность используемых методов защиты информации либо сформулировать требования к используемым алгоритмам и методам.

Субъект, препятствующий нормальному протеканию процесса информационного взаимодействия, в криптографии называется злоумышленником. Злоумышленник - это не конкретное лицо, а некая персонифицированная сумма целей и возможностей, для которой справедлив аналог принципа Паули из физики элементарных частиц: два субъекта, имеющие идентичные цели и возможности по их достижению, в криптографии рассматриваются как один и тот же злоумышленник.

Задачи из сферы защиты информации отличаются друг от друга именно ответами на приведенные вопросы. Только после детального и исчерпывающего ответа на них можно сформулировать задачу защиты информационного процесса и приступить к проектированию способов и средств защиты.

2. Задачи, решаемые криптографическими методами

2.1 Предметом **криптографии** является класс методов, предназначенных для защиты процессов **информационного взаимодействия** от отклонений от их **нормального течения**, вызванных целенаправленными воздействиями со стороны **злоумышленника** основанных. На преобразовании информации по **секретным алгоритмам**.

Понятие "секретный алгоритм" трактуется широко - алгоритм, хотя какая-то деталь которого держится в секрете. Оно включает в себя и открытые алгоритмы, зада параметров которых держится в тайне.

Процесс информационного взаимодействия, или **информационным взаимодействием**, или **информационным процессом** называют такой процесс взаимодействия нескольких субъектов, основным содержанием которого является обмен информацией между ними. Взаимодействующих субъекты в криптографии называются **законными** или **легальными участниками** процесса, слово "взаимодействие" предполагает, что их как минимум два.

Нормальное течение процесса информационного взаимодействия, и о возможных **отклонениях** от него. Норма - это соответствие процесса параметрам его течения, запланированным теми субъектами, которые пытаются выполнить стоящие перед ними задачи посредством этого самого информационного процесса. Любое событие, выходящее процесс за рамки предусмотренной для него нормы считается **отклонением**. Криптография рассматривает способы борьбы с отклонениями, вызванными целенаправленными действиями злоумышленников. В ней постулируется, что последние действуют наилучшим возможным с точки зрения достижения своих целей образом в рамках имеющихся в их распоряжении данных и не имеют все необходимые ресурсы. Любое случайное воздействие на информационный процесс не способно отклонить его протекание от нормы больше, чем это сделал бы злоумышленник.

Злоумышленник действует наилучшим возможным в его ситуации образом. В качестве злоумышленников могут выступать как законные участники процесса, так и субъекты, не являющиеся законными участниками процесса, но имеющие доступ к передаваемой и обрабатываемой информации в ходе осуществления информационного взаимодействия и могущие появляться на его протекании.

Если законные участники процесса не могут выступать в качестве злоумышленников, такой процесс называется **информационным взаимодействием со взаимным доверием сторон друг другу**, в противном имеет место **процесс информационного взаимодействия в условиях отсутствия взаимного доверия сторон**. Методы защиты процессов обоих типов существенно отличаются друг от друга, вторая задача сложнее: любую систему намного легче защитить от проникновения извне, чем от злоумышленных действий со стороны ее законных пользователей.

Когда речь идет о **взаимном доверии сторон**, имеется в виду нечто большее, чем простое отношение субъектов информационного взаимодействия друг к другу. При этом должен рассматриваться целый ряд факторов, например среда и окружение, в которых они работают

2.2 Задачи, решаемые криптографическими методами, отличаются друг от друга характером защищаемого информационного взаимодействия, целями и возможностями злоумышленников.

Самая распространенная и до сих пор наиболее важная задача из сферы защиты - защита передаваемой по каналам связи или хранимой в компьютерных системах информации. В последнее время с проникновением электронных технологий в разные сферы жизни человека и общества возникает и принципиально новые проблемы. Например, совместная выработка субъектами сенсуса связи по открытому каналу общего секретного ключа таким образом, чтобы злоумышленники, прослушивающие канал, не смогли получить тот же самый ключ.

Задачу криптографической задачи защиты данных, передаваемых по открытым каналам связи, в общем случае можно сформулировать следующим образом.

Имеется две легальные стороны - **отправитель** и **получатель**. Информационный процесс заключается в передаче сообщения от первого другому по каналу передачи данных и считается протекающим нормально, если получатель получит сообщение, идентичное тому, что, не связываясь с содержанием сообщения, а стороны не будут выступать против друг друга. Злоумышленник имеет доступ к каналу передачи данных и стремится добиться отклонений от нормального течения процесса. Кроме того, каждая из легальных сторон может предпринять злоумышленные действия в отношении другой стороны.

Возможные угрозы можно классифицировать следующим образом.

Угрозы со стороны злоумышленника.

- 1) Ознакомление с содержанием переданного сообщения.
- 2) Навязывание получателю ложного сообщения либо его полной фабрикации, либо внесением искажений в передаваемое сообщение.
- 3) Изъятие переданного отправителем сообщения из системы таким образом, чтобы получатель не узнал о факте передачи сообщения.
- 4) Создание помех для нормальной работы канала передачи связи, то есть нарушение работоспособности канала связи.

Угрозы со стороны законного отправителя сообщения.

- 1) Разглашение переданного сообщения.
- 2) Отказ от авторства переданного им сообщения.
- 3) Утверждение, что некоторое сообщение отправлено получателю, хотя в действительности отправка не производилась.

Угрозы со стороны законного получателя сообщения.

- 1) Разглашение полученного сообщения.
- 2) Отказ от факта получения некоторого сообщения, хотя в действительности оно было им получено.
- 3) Утверждение, что некоторое сообщение получено от отправителя, хотя в действительности переданное сообщение сфальсифицировано самим получателем.

Угроза нарушения работоспособности канала связи реализуется разрушением физической среды передачи данных (линий передачи и узлов обработки данных), созданием помех, перегрузкой системы большим количеством ложных сообщений (спамом), изъятием сообщений из канала связи. Криптографическими средствами обеспечить эффективную защиту от этих угроз невозможно, поэтому они в работах по криптографии не рассматриваются. Эти проблемы решаются другими методами, например, для устранения угрозы изъятия сообщений обычно используется шифрование – высказка получателем отправителю клятвенный (подтверждения) на полученное сообщение. В рамках криптографии отсутствуют решения, которые устранили бы угрозы разглашения секретных данных одной из легальных сторон.

2.2 Криптография может использоваться для решения одной из следующих задач защиты информации при ее передаче по каналам связи.

1) Защита данных от разглашения и искажения при передаче по открытому каналу связи – классическая задача криптографии.

2) Подпись электронного документа – защита от отказа от авторства сообщения.

3) Вручение заказного письма – защита от отказа от факта получения сообщения.

Эти задачи характеризуются собственными наборами угроз из приведенного выше списка.

1. Защита передаваемых и хранимых секретных данных от разглашения и искажения

Это наиболее важная задача криптографии, в ней учитываются угрозы 1 и 2 со стороны злоумышленника. Задачи хранения и передачи данных отличаются тем, что в первом случае говорят о защите данных при *хранении*, во втором – при *передаче*. Каждый из вариантов задачи имеет свои особенности, методы их решения также могут отличаться. В задаче присутствуют две легальные стороны: *О* – отправитель и *П* – получатель сообщения и одна нелегальная *З* – злоумышленник.

Между *О* и *П* сообщения есть взаимное доверие. *З* может попытаться выполнить одно или несколько из следующих действий:

- 1) чтение сообщения;
- 2) внесение изменений в переданное сообщение "на лету";
- 3) создание нового сообщения и отправка его *П* от имени *О*;
- 4) повторная передача ранее переданного сообщения;
- 5) уничтожение переданного сообщения.

Каждое из этих действий является **атакой** на информационный процесс. Возможности доступа к каналу передачи для выполнения каждой из атак различны, и в конкретной ситуации может оказаться, что одни атаки осуществимы, а другие – нет. Для атаки №1 необходим доступ к каналу передачи данных на чтение, для атаки №3 – на запись, для атаки №4 – на чтение и запись. Для атак №2 и №5 необходим полный контроль над каналом, то есть возможность разорвать его и встроить туда собственные узлы обработки данных, или получить контроль над существующим узлом обработки. Доступность каждой из атак для злоумышленника, зависит от конкретных условий протекания информационного процесса: среды передачи данных, аппаратуры, которой он располагает, и т.д. Например, если средой передачи информации служат радиоканал, осуществимы атаки №2 и частично №5. При использовании оптоволоконной линии связи злоумышленнику может быть доступна только атака №1 – незаметно "врезаться" в оптоволоконную линию практически невозможно.

Итак, рассматриваемую задачу можно сформулировать следующим образом.

Законные стороны информационного процесса – *О* и *П*. Задача первого отправить, а второго получить сообщение.

Процесс считается нормально идущим, если *П* придет именно то сообщение, которое отправлено *О*, и никто кроме *О* не сможет ознакомиться с его содержанием. Отклонения от нормального течения процесса:

- передаваемые данные окажутся известными третьим лицам;
- передаваемые данные искажены.

Между *О* и *П* есть взаимное доверие. Никто из них не осуществляет злоумышленных действий. Злоумышленником является третья сторона *З*, которая ставит перед собой целью ознакомиться с содержанием переданного сообщения или навязать получателю ложное сообщение, полностью сфальсифицировав его самостоятельно или исказив переданное отправителем сообщение. *З* имеет доступ к каналу связи на чтение и на запись. В наилучшей для себя ситуации *З* может захватить полный контроль над каналом, и ему будут доступны любые операции с переданными данными.

2. Подпись электронного документа – задача подтверждения авторства сообщения

В этой задаче принимаются во внимание угрозы 2 со стороны законного отправителя сообщения и угрозы 3 со стороны законного получателя сообщения из приведенного ранее списка угроз. Между *О* и *П* отсутствует взаимное доверие. Каждый из них может совершать злоумышленные действия, направленные против другой стороны, поэтому в системе необходимо наличие независимого арбитража – инстанции, которая выполняет арбитражные функции: в случае конфликта между абонентами решает, кто из них прав, а кто нет. Злоумышленник как отдельный субъект информационного процесса отсутствует.

Рассматриваемую задачу можно сформулировать следующим образом.

Законные стороны информационного процесса – отправитель сообщения (*О*), получатель сообщения (*П*) и независимый арбитр (*А*). Задача *О* отправить, задача *П* получить сообщение и понять его содержание, задача *А* вынести суждение о том, прав в случае возникновения конфликта между *О* и *П*.

Процесс считается проходящим нормально, если *П* получит именно то сообщение, которое отправлено *О*, и стороны не будут предъявлять относительно него претензий друг другу относительно данного сообщения.

Отклонения от нормального течения процесса:

- *О* может отказаться от авторства переданного им сообщения;
- *П* может утверждать о получении им некоторого сообщения от *О*, хотя в действительности тот его не передавал.

Между отправителем и получателем отсутствует взаимное доверие, каждый из них может осуществить злоумышленные действия по отношению к другому.

3. Вручение заказного письма – вручение сообщения под расписку

В этой задаче принимаются во внимание угрозы 3 со стороны законного отправителя сообщения и угрозы 2 со стороны законного получателя сообщения из приведенного ранее списка угроз. Между отправителем и получателем сообщения отсутствует взаимное доверие. Каждый из них может совершать злоумышленные действия, направленные против другой стороны. Злоумышленник как отдельный субъект информационного процесса также отсутствует.

Эту задачу можно сформулировать следующим образом.

Законные стороны информационного процесса – отправитель сообщения (*О*), получатель сообщения (*П*) и независимый арбитр (*А*). Задача *О* отправить, задача *П* получить сообщение и понять его содержание.

Процесс считается проходящим нормально, если получатель ознакомится с содержанием полученного сообщения, и стороны не будут предъявлять претензий друг другу относительно переданных данных. Возможны следующие отклонения от нормального течения процесса:

- *О* будет утверждать, что передал получателю сообщение, хотя в действительности не отправлял его;
 - *П* ознакомится с содержанием сообщения, но будет утверждать, что никакого сообщения не получал.
- Между *О* и *П* отсутствует взаимное доверие, и каждый из них может осуществить злоумышленные действия в отношении другого.

Решением этой задачи может являться такая схема информационного взаимодействия, которая объединяет в одну операцию два следующих действия, не позволяя ни одному из них осуществиться без другого:

О получает и читает сообщение;

П получает подтверждение о том, что *О* получил сообщение.

2.4 Важной областью криптографии является решение задач, связанных с обменом ключевой информацией. Для защиты передаваемых по открытым каналам связи данных применяется их шифрование с применением симметричных шифров, то есть шифров с секретным ключом. В таких системах существует проблема распределения ключевой информации: для ее передачи участникам информационного обмена нужен защищенный канал связи. В системах с большим числом абонентов эта проблема становится весьма острой. В настоящее время известно два способа ее решения:

- применение асимметричных алгоритмов шифрования, в которых для процедур шифрования и дешифрования используются различные ключи; при этом знание ключа шифрования не позволяет определить соответствующий ключ дешифрования, следовательно, ключ шифрования может быть несекретным и передаваться по открытым каналам связи;
- выработка общего секретного ключа в ходе некоторого сеанса информационного взаимодействия по открытому каналу, организованного таким образом, чтобы ключ было невозможно выработать на основе только перехваченных в канале данных.

Первый подход получил название **асимметричного** или **двухключевого шифрования**, второй – **открытого распределения ключей**.

2.5 Помимо перечисленных криптографических методов применяются для решения ряда менее известных задач.

1. **Синхронный обмен** двух сторон сообщениями таким образом, чтобы ни одна из них, получив сообщение другой стороны, не смогла отказаться от передачи своего сообщения либо и не смогла сформулировать его в зависимости от сведений из сообщения другой стороны.

2. **Задача подписи контракта**: требуется организовать обмен подписанными документами в электронной форме между двумя субъектами таким образом, чтобы ни один из них не смог отказаться передать свой подписанный документ, получить подписанный документ от другой стороны.

3. **Передача с забыванием** – организовать передачу сообщения одним субъектом другому таким образом, чтобы вероятность получения сообщения была равно 0,5, и чтобы на эту вероятность никто не мог повлиять.

4. **Доказательство с нулевым раскрытием** – есть субъект, располагающий некоторым секретным элементом данных **a**, ему необходимо продемонстрировать другому субъекту, что он располагает этим элементом данных, не раскрывая его.

5. **Сравнение с нулевым раскрытием** или "проблема двух миллионеров" – два миллионера хотят узнать, кто из них богаче, но при этом никто из них не хочет сообщить другой стороне истинную величину своего состояния. В формальной постановке: каждый из двух субъектов располагает некоторым элементом данных – соответственно **a** и **b**, и оба они которые желают совместно вычислить значение некоторой согласованной функции **f(a,b)**. Требуется организовать процедуру вычисления таким образом, чтобы никто из субъектов не узнал значения параметра другого.

6. **(n,k)-пороговая схема** – имеется некоторый ресурс, например – зашифрованный набор данных (файл), доступ к которому имеют **n** субъектов. Требуется построить процедуру, разрешающую доступ к ресурсу, только если его запросят одновременно не менее **k** субъектов из **n**.

7. **Тайное голосование по телефону** – есть **n** субъектов, голосующих по некоторому вопросу по линиям связи. Требуется организовать процедуру голосования таким образом, чтобы можно было определить ее исход, но при этом результаты голосования каждого из субъектов оставались в тайне.

3. Шифрование и шифры

3.1 Важнейшей задачей криптографии является защита передаваемых по каналам связи или хранящихся в системах обработки информации данных от несанкционированного ознакомления с ними или от преднамеренного их искажения. В криптографии эта задача решается путем применения к записываемым данным двух взаимно обратных преобразований: **шифрования и расшифрования**. Защищаемые данные перед отправлением по линии связи или перед помещением на хранение подвергаются **шифрованию**. Для восстановления исходных данных из зашифрованных применяется процедура **расшифрования**.



Рис. 3.1. Схема преобразования данных при шифровании.

Шифром называется пара алгоритмов, реализующих каждое из указанных преобразований. Секретность второго преобразования обеспечивает недоступность данных для несанкционированного ознакомления, а секретность первого делает невозможным навязывание ложных данных. Получение исходного сообщения из зашифрованного без знания алгоритма расшифрования называется **дешифрованием**. Шифрование может защитить данные от несанкционированной модификации, только если шифруемое сообщение содержит большую избыточность, а алгоритм шифрования обеспечивает хорошо перемешивает структуры единиц сообщения: биты, символы и т.д. Следовательно, в общем случае шифрование не является средством **целостности** – защиты от навязывания ложных данных.

Шифр должен удовлетворять следующим требованиям. Во-первых, процедура расшифрования должна всегда восстанавливать открытое сообщение в его исходном виде. Иными словами, для каждого допустимого сообщения **T** преобразования **шифрования-расшифрования** должны удовлетворять условию:

$$T = D(E(T)).$$

Второе требование, которому должен удовлетворять шифр: зашифрованные данные должны быть непонятными для непосвященного, т.е. не должно существовать легко прослеживаемых связей между исходными и зашифрованными данными. Третьим по счету и, вероятно, первым по важности требованием к шифру должно быть его **криптоустойчивость**, то есть устойчивости к попыткам дешифрования сообщений.

3.2 Для определения меры криптоустойчивости шифра используется энтропийный подход, широко используемый в теории информации и связи.

Отправленное сообщение до его поступления к получателю является неопределенным и для него и для злоумышленника. Пусть возможна отправка сообщений T_1, T_2, \dots, T_n с вероятностями p_1, p_2, \dots, p_n соответственно. Мерой **неопределенности сообщения** для всех, кто обладает этой априорной информацией, может служить величина **энтропии** сообщения – взятое со знаком минус среднее значение логарифма вероятности сообщения; основание логарифма, обычно, выбирают равным 2.

$$H(T) = - \sum_{i=1}^n p_i \log_2 p_i$$

Эта величина определяет количество бит информации, которое в среднем **необходимо** передать, чтобы полностью устранить неопределенность относительно одного сообщения. Если априорной информацией о сообщении является лишь его длина **N** бит, то любой 2^N возможных вариантов считается равновероятным и тогда энтропия (неопределенность) сообщения равна его размеру:

$$H(T) = -2^N \cdot 2^{-N} \log_2 (2^N) = N \lceil T \rceil.$$

Здесь через $\lceil X \rceil$ обозначен размер блока данных **X** в битах. Если об исходном тексте неизвестно вообще ничего (даже его размер), то в этом случае необходимо принять за основу какую-либо модель распределения. Обычно в реальных задачах полагают, что размер шифруемого сообщения априори (до опыта) считается известным злоумышленнику. Там же, где этот размер реально необходимо скрыть, все сообщения перед шифрованием преобразуются в массивы данных одинаковой длины в результате чего снова получается рассматриваемая выше ситуация.

После перехвата шифртекста **T'** неопределенность открытого текста изменится – она определится апостериорной (после опытной) условной энтропией. В качестве условия здесь выступают перехваченное шифрованное сообщение **T'**. Условная энтропия вычисляется по формуле

$$H(T | T') = - \sum_{i=1}^n p(T_i | T') \log_2 p(T_i | T'),$$

где через $p(T_i | T')$ обозначена вероятность того, что исходным сообщением является **T_i** при условии, что зашифрованное сообщение есть **T'**.

Количество информации об исходном тексте, которое злоумышленник может извлечь из перехваченного шифртекста, можно найти как разность между априорной и апостериорной неопределенностью исходного сообщения

$$I = H(T) - H(T | T').$$

Эта величина всегда неотрицательна. Показателем криптоустойчивости шифра является уменьшение неопределенности исходного текста при получении соответствующего шифртекста по сравнению с априорной неопределенностью. В наилучшем для разработчиков шифра случае обе эти неопределенности равны:

$$H(T | T') = H(T),$$

то есть злоумышленник не может извлечь из перехваченного шифртекста никакой полезной информации об открытом тексте: $I = 0$. Иными словами, знание шифртекста не позволяет уменьшить неопределенность соответствующего открытого текста, улучшить его оценку и увеличить вероятность его правильного определения. Шифры, удовлетворяющие данному условию, называются **абсолютно стойкими** или **совершенными шифрами**. Зашифрованные с их применением сообщения не только не могут быть дешифрованы в принципе, но злоумышленник даже не сможет увеличить вероятность их правильного дешифрования.

Абсолютно стойкие шифры существуют. К.Шеннон формально доказал их существование. В процессе доказательства Шеннон получил и необходимое условие абсолютной стойкости шифра: для того, чтобы шифр был абсолютно стойким, необходимо, чтобы энтропия алгоритма шифрования была не меньше энтропии шифруемого сообщения:

$$H(E) \geq H(T).$$

Энтропия алгоритма шифрования определяется точно так же, как и энтропия сообщения – математическое ожидание двоичного логарифма вероятности использования алгоритма со знаком минус. Она имеет смысл только в том случае, если определено множество возможных алгоритмов и заданы вероятности использования каждого из них.

3.3 Стойкость шифров основана на секретности, то есть на неопределенности для злоумышленника алгоритма расшифрования. Чем меньше знает злоумышленник о шифре, тем меньше вероятность дешифрования сообщения. Пусть, например, перехвачена короткая 12-битовая шифровка, имеющая следующее содержание: 100101110101. Предположим, что исходное сообщение имеет ту же длину. Если у злоумышленника нет никаких априорных сведений о зашифрованном сообщении, для него каждый из 2^{12} исходных вариантов равновероятен, и, таким образом, вероятность определить исходное сообщение простым угадыванием равна 2^{-12} . Предположим теперь, что злоумышленнику априори известно, что шифрование является наложением одной и той же 4-битовой маски на каждую 4-битовую группу сообщения с помощью операции **побитового сложения по модулю 2**. Существует $2^4 = 16$ различных вариантов битовой маски, соответственно, возможно 16 различных значений исходного текста (табл.1).

Таблица 1

Маска	Исходный текст
0000	100101110101
0001	100001100100
0010	101101010111
...	...

В этом случае вероятность дешифрования исходного текста возросла до 1/16 за счет знания особенностей использования способа шифрования. Отсюда следует вывод: чем больше неопределенность шифрующего преобразования для постороннего лица тем надежнее шифр. Шифр, полностью неопределенный для злоумышленника ($H(E)=\infty$), является не раскрываемым для него, то есть абсолютно стойким. Надежность шифра зависит исключительно от его секретности и не зависит от прочих его свойств.

Парадокс здесь только кажущийся: на практике невозможно сохранить у злоумышленника полную неопределенность относительно шифра. Он может получить информацию о шифре следующими путями:

- 1) анализировать передаваемые сообщения - практически в его распоряжении всегда имеется определенный набор шифртекстов, для некоторых из них могут иметься и соответствующие открытые тексты, иногда имеется даже возможность получить шифртекст для любого заданного открытого текста;
- 2) получать априорные сведения о шифре из различных источников - например, инструкции по шифрованию или черновика с промежуточными результатами для конкретного текста, фрагмента компьютерного кода или микросхемы, реализующей шифрование аппаратно.

Первая возможность есть у злоумышленника всегда, вторая также очень вероятна - трудно удерживать в секрете от посторонних активно работающий алгоритм. Хороший шифр должен удовлетворять следующим требованиям.

- 1) Анализ зашифрованных данных не должен давать злоумышленнику никаких сведений о внутреннем устройстве шифра. В шифртексте не должно прослеживаться статистических закономерностей - например, статистические тесты не должны выявлять в зашифрованных данных отклонений от равномерного распределения битов (символов) шифртекста.

- 2) Алгоритм должен быть перенастраиваемым. В распоряжении злоумышленника рано или поздно может оказаться описание алгоритма, его программа или аппаратная реализация. Для того, чтобы в этом случае не пришлось заменять алгоритм полностью на всех узлах шифрования, его не используется, он должен содержать легко сменяемую часть.

Второе условие является одним из выражений принципа Кирхгофа, безоговорочно принятого в настоящее время при построении надежных шифров. Согласно ему шифр определяется как параметризованный алгоритм, состоящий из процедурной части и параметров. Процедурная часть включает описание операций над шифруемым данными и последовательности их выполнения. В параметрическую входят параметры различных элементов данных, используемых в преобразованиях. Раскрытие только процедурной части не должно приводить к увеличению вероятности успешного дешифрования сообщения злоумышленником выше допустимого предела. По этой причине, а также в силу того, что рассекречивание этой части достаточно вероятно само по себе, нет особого смысла хранить ее в секрете. В секрете держится некоторая часть параметров алгоритма, которая называется **ключом шифра**:

$$T' = E_K(T),$$

здесь K - ключ шифра.

Использование принципа Кирхгофа позволяет получить преимуществ в построении шифров:

- 1) разложение конкретного шифра (алгоритма и ключа) не приводит к необходимости полной замены реализации всего алгоритма, достаточно заменить скомпрометированный ключ;
- 2) ключи можно отлучать от остальных компонентов системы шифрования - хранить отдельно от реализации алгоритма в более надежном месте и загружать их в шифратор по мере необходимости и только на время выполнения шифрования - это значительно повышает надежность системы в целом;
- 3) появляется возможность для точной оценки степени неопределенности алгоритма шифрования - она просто равна неопределенности используемого ключа: $H(E_K) = H(K)$;
- 4) становится возможным оценить вероятность и трудность успешного дешифрования, объем вычислительной работы, которую необходимо выполнить злоумышленнику для этого.

3.4 Найдем необходимое условие абсолютной стойкости шифра для шифров, построенных в соответствии с принципом Кирхгофа. Если нет никаких априорных данных о шифруемом тексте кроме его длины, то неопределенность исходного текста равна его длине в битах:

$$H(T) = |T|.$$

Максимально возможная неопределенность блока данных заданного размера достигается, когда все возможные значения этого блока равновероятны - в этом случае она равна размеру блока в битах. Таким образом, неопределенность ключа K не превышает его длины:

$$H(K) \leq |K|.$$

С учетом сказанного выше получаем необходимое условие абсолютной стойкости для шифров, удовлетворяющих принципу Кирхгофа:

$$|K| \geq H(K) = H(E_K) = H(E) \geq H(T) = |T|.$$

То есть, чтобы шифр, построенный по принципу Кирхгофа, был абсолютно стойким, необходимо, чтобы размер ключа шифрования был не меньше размера шифруемых данных;

$$|K| \geq |T|.$$

Точное равенство достигается, когда возможные значения ключа равновероятны. Это выполняется, когда все биты ключа равновероятны и статистически независимы друг от друга.

Примером абсолютно стойкого шифра является **одноразовая гамма** Вернама - наложение на открытые данные (T) с помощью некоторой бинарной операции \otimes ключа (K) такого же размера, составленного из статистически независимых битов, принимающих возможные значения с одинаковой вероятностью,

$$T' = T \otimes K.$$

Используемая для наложения гаммы операция должна удовлетворять некоторым условиям, которые можно суммировать следующим образом: уравнение шифрования должно быть однозначно разрешимо относительно открытого данных при известных зашифрованных и ключе, и однозначно разрешимо относительно ключа при известных открытых и зашифрованных данных. В качестве бинарной операции \otimes обычно выбирают **побитовое суммирование по модулю 2 (побитовое исключение или ИЛИ)**. Она обладает следующими свойствами:

- ее реализация требует минимальной по сложности логики, что предельно упрощает аппаратную реализацию шифраторов и дешифраторов;
- она является обратной самой себе, поэтому для шифрования и расшифрования можно применять одну и ту же процедуру и аппаратно.

Для абсолютно стойких шифров необходим ключ, размер которого не меньше размера шифруемых данных. Для однократного пользования таким ключом отправителю и получателю необходим защищенный канал связи. То есть, наряду с потенциально незащищенным каналом для передачи зашифрованных данных необходимо существование зашифрованного канала для передачи такого же по размеру ключа. Такие системы применяются в исключительных случаях для защиты сведений, представляющих особую ценность. Обычно в системах шифрованной связи используются алгоритмы, не обладающие абсолютной стойкостью и поэтому называемые **несовершенными шифрами**.

Для таких шифров актуален вопрос надежности оцен их стойкости. Знание шифртекста позволяет снизить условную энтропию соответствующего открытого текста, что теоретически увеличивает вероятность успешного дешифрования. Однако, из этого вовсе не следует, что такое дешифрование возможно **всегда**.

Это возможно только тогда, когда криптоаналитик располагает достаточным по объему шифртекстом и неограниченными вычислительными возможностями. Повысить вероятность успешного дешифрования и сделать ее равной единице не одно и то же.

Пример. Шифрованию подвергается некий массив двоичной информации, размер ключа один бит и шифрование осуществляется по следующим правилам:

- если ключ $K=0$, то инвертируются нечетные по номеру биты исходного текста;
- если $K=1$, то инвертируются четные по номеру биты исходного текста;

Таким образом, $E_0(01) = 11$, $E_1(01) = 00$. Очевидно, что шифр не обладает абсолютной стойкостью. Предположим, что перехвачена шифровка 10. Каков исходный текст: он может быть как 00 так и 11 в зависимости от значения K , и без дополнительной информации однозначно определить это невозможно, что и требовало доказать. При более сложных шифрах у криптоаналитика будет больше вариантов выбора открытого текста, и никаких указаний на то, какой из них предпочтительнее.

3.5 Вопрос о возможности однозначного дешифрования сообщения, зашифрованного несовершенным шифром, остается открытым. Этот вопрос подробно исследовал в своих работах Шеннон. Он ввел в рассмотрение следующие характеристики шифра.

1. Функция ненадежности ключа - неопределенность ключа, когда известны n бит шифртекста:

$$f(n) = H(K | T^n), \text{ где } T^n = n.$$

$f(n)$ может быть не определена для некоторых n .

2. Расстояние единственности шифра - такое значение n , при котором функция ненадежности (неопределенность ключа) становится близкой к 0.

$$U(E) = n, \text{ когда } f(n) \sim 0.$$

Шеннон показал, что обе эти характеристики зависят от избыточности открытого текста, причем расстояние прямо пропорционально размеру ключа единственности и обратно пропорционально избыточности:

$$U(E) \sim \frac{H(K)}{R} = \frac{|K|}{R},$$

где $R = 1 - \frac{H(T)}{|T|}$ - избыточность исходного текста.

Таким образом, полное устранение избыточности открытого текста позволяет достигнуть невозможности его однозначного дешифрования на основе знания только соответствующего шифртекста, даже если криптоаналитик располагает неограниченными вычислительными возможностями. Неопределенность исходного текста будет определяться неопределенностью ключа, т.е. его размером:

$$H(T) = H(K) = |K|.$$

Для снижения избыточности исходного текста, в реальной практике перед зашифрованием данные можно "сжать" архиватором. Конечно, полная безыбыточность исходного текста недостижима, однако такая процедура очень сильно затруднит некоторые виды криптоанализа.

Если в распоряжении криптоаналитика есть не только шифртекст, но и соответствующий открытый текст, следовательно характеристики стойкости шифра не будут зависеть от избыточности исходных сообщений. В этом случае рас-

стояние единственности шифра близко к размеру его ключа, то есть весьма мало. Такой шифр легко вскрывается при неограниченных вычислительных ресурсах аналитика.

3.6 Существуют абсолютно стойкие (*совершенные*) шифры, которые невозможно раскрыть в принципе. Цена такой стойкости - необходимость использовать ключевую информацию, по объему не меньшую, чем защищаемые данные. При использовании совершенных шифров однократное дешифрование возможно сообщения только при выполнении одного из следующих условий:

в распоряжении аналитика есть фрагмент шифртекста и соответствующего ему открытого текста примерно равные по размеру длине ключа ($|K|$);

исходный текст обладает некоторой избыточностью ($R, 0 \leq R < 1$), а в распоряжении криптоаналитика есть фрагмент шифртекста объемом не менее $\frac{|K|}{R}$

На практике часто эти условия выполняются, поэтому для реальных шифров злоумышленник, располагающий неограниченными вычислительными ресурсами, временем и шифртекстом достаточного размера, может однозначно дешифровать сообщение. Поэтому стойкость всех современных шифров, не являющихся совершенными, базируется на вычислительной сложности дешифрования. Реально вычислительные возможности аналитика всегда ограничены, и для них может быть получена верхняя оценка. Хорошо спроектированным считается шифр, который невозможно вскрыть с вычислительными затратами, осуществимыми за разумное время и запасом в несколько порядков для прогнозируемой упрямости прогресса вычислительной техники.

В качестве меры трудоемкости раскрытия таких шифров обычно используется количество *элементарных операций* (И) некоторого типа, необходимых для дешифрования сообщения или определения ключа. Под элементарной операцией в различных случаях обычно понимают операцию, выполняемую на конкретной аппаратуре за один шаг ее работы. Например, операцию типа "сложение", для универсальных процессоров, или цикл проверки открытого ключа для специальных аппаратных схем переноса ключей. Трудоемкость дешифрования зависит от того, какая информация есть в распоряжении аналитика, и сколько ее у него. Обычно различают следующие виды криптоанализа:

анализ на основе только шифртекста - у аналитика имеется только зашифрованное сообщение размером N :

$$w = W_{CT}(n);$$

анализ на основе заданного открытого текста - аналитик располагает зашифрованным сообщением размером N и соответствующим ему открытым текстом:

$$w = W_{CT}(n);$$

анализ на основе произвольно выбранного шифртекста - в распоряжении аналитика есть возможность получить шифртекст для произвольно выбранного им массива открытых данных размером N :

$$w = W_{CT}(n);$$

анализ на основе произвольно выбранного шифртекста - в распоряжении аналитика есть возможность получить результат дешифрования для произвольно выбранного им зашифрованного сообщения размером N :

$$w = W_{CT}(n);$$

Предполагается, что криптоаналитик использует наилучший из доступных ему способов анализа. Очевидно, что между величинами трудоемкости различных видов криптоанализа выполняются следующие соотношения:

$$W_{CT}(n) \geq W_{CT}(n) \geq W_{CT}(n), W_{CT}(n).$$

Все рассмотренные характеристики трудоемкости имеют нижние границы $w_{\min} = \inf W_{CT}(n)$.

Эти границы достигаются при некоторых конечных значениях параметра N , потому что при неограниченном увеличении N трудоемкость анализа (если принимать во внимание все имеющиеся в наличии данные) будет неограниченно возрастать.

$$w_{\min} = W_{CT}(n_{\min}).$$

3.7 Таким образом, для каждого вида криптоанализа (XX) существует свой оптимальный объем необходимых данных (N_{XX}), при возрастании объема имеющихся данных до этого значения трудоемкость анализа снижается до своего граничного значения ($W_{CT}(n_{\min})$), а при дальнейшем возрастании - увеличивается. Эти критические объемы данных и соответствующие им трудоемкости анализа представляют особый интерес для специалистов-криптографов. Реально трудоемкость анализа зависит не только от объема анализируемых данных, но и от самих этих данных. По этой причине все приведенные выше соотношения являются оценочными, а соответствующие величины считаются заданными с точностью до одного-двух порядков.

Не существует способа получить точное значение трудоемкости анализа, все оценки базируются на проверках устойчивости шифров к известным на текущий момент видам криптоанализа, и нет гарантии, что в ближайшем или более отдаленном будущем не будут разработаны новые методы анализа, существенно ее снижающие.

Стойкость абсолютно всех шифров, за исключением *совершенных*, в настоящее время не может быть доказательно обоснована. Вместо этого она обосновывается эмпирически как устойчивости к известным на сегодняшний день видам криптоанализа, однако нет гарантии того, что завтра не будет изобретен вид криптоанализа, успешный именно для данного конкретного шифра. Поэтому не стоит доверять "новейшим шифрам" - они не прошли проверку временем. По той же самой причине не является разумным доверять криптоалгоритмам, которые держатся их авторами в секрете - даже при отсутствии злонамеренно оставленных там "люков" нет совершенно никакой гарантии того, что алгоритм был исследован со всей необходимой тщательностью.

Сказанное не означает, что использование секретных алгоритмов шифрования вовсе лишено смысла. Оно является допустимым и разумным при выполнении двух следующих условий:

между разработчиками и пользователями алгоритма существует уровень доверия, исключающий намерение разработчика нанести ущерб пользователю, предоставив ему недостаточно качественный шифр или шифр с оставленными в нем "люками";

специалисты, разработавшие алгоритм, имеют достаточно высокий уровень компетентности в этой области; эти условия выполняются, например, для спецслужб ведущих государств мира, разрабатывающих собственные шифры для "внутреннего потребления".

Классификация шифров

3.8 Рассмотрение следующей нашей темы - классификации шифров - начнем с двух требований, предъявляемых к практическим алгоритмам шифрования:

шифр должен быть технически применим для закрытия массивов данных произвольного объема; шифр должен быть реализуем в виде устройства, имеющего ограниченный объем памяти, и его реализация должна быть эффективна при этом.

Попытка совместить оба требования неизбежно приводит к криптоалгоритму, в котором шифрование проводится пошагово (порциями) - массив данных разбивается на n блоков T_i ($i=1...n$) ограниченного размера, и за один шаг шифруется один блок $T = (T_1, T_2, ..., T_n)$. Для всех i $|T_i| \leq N$, где N - максимальный размер блока.

От размера шифруемого массива данных зависит только количество шагов шифрования, но не сами шаги. Ради удобства измерения размер блока обычно полагают постоянным.

$$|T_1| = |T_2| = ... = |T_n| = |T_n|, N_n \leq N.$$

Для обеспечения криптоустойчивости размер блока не должен значительно превышать размер ключа, лучше, если он будет меньше или равен ему.

Известны два принципиально различающихся подхода к построению шифров с секретным ключом, соответственно им можно выделить два типа шифров - *блочные* и *поточковые шифры*.

В *блочных шифрах* результат шифрования очередного блока зависит только от него самого и не зависит от других блоков шифруемого массива данных:

$$T_i' = E(T_i).$$

Из этого следует, что в результате шифрования двух одинаковых блоков открытого текста всегда получаются идентичные блоки шифртекста:

$$T_i = T_j \rightarrow E(T_i) = E(T_j).$$

3.9 В *поточных* или *поточковых шифрах* результат зашифрования очередного блока зависит от него самого и, в общем случае, от всех предыдущих блоков массива данных:

$$T_i' = E(T_1, T_2, ..., T_i).$$

Сюда же относится важный частный случай, когда результат зашифрования очередного блока зависит этого блока и от его номера:

$$T_i' = E(i, T_i).$$

В современной криптологии понятия блочного и поточкового шифра иногда используют в несколько отличном от сказанного выше смысле - поточковыми называют только такие шифры, в которых шифруемый за один шаг блок имеет размер один бит или один символ текста, а шифры с большим размером блока приписывают к блочным. Такие поточковые шифры хорошо подходят для зашифрования асинхронного информационного потока.

С точки зрения реализуемости криптоалгоритма устройством с конечным числом возможных состояний наиболее общей моделью поточковых шифров является конечный автомат, описываемый множеством состояний \mathcal{S} , входным \mathcal{X} и выходным \mathcal{E} алфавитами, и правилами перехода и выхода $\mathcal{S} \rightarrow \mathcal{S}$ соответственно.

$$X_{i+1} = \mathcal{X}(X_i, T_i), T_i' = \mathcal{E}(X_i, T_i), \quad X \in \mathcal{X}, T \in \mathcal{X}, T_i' \in \mathcal{E}.$$

Множество состояний и алфавиты автомата являются конечными, а правила перехода и выхода могут быть записаны в виде двумерной таблицы, которые иногда называют таблицами переходов и выходов соответственно. Автомат работает следующим образом: каждый символ, поступивший на его вход, вызывает изменение состояний автомата и порождение одного выходного символа. В результате входное слово преобразуется в слово точно такой же длины, составленное из символов выходного алфавита. Работа конечного автомата зависит от его начального состояния: в общем случае два идентичных автомата преобразуют одно и то же входное слово в разные выходные, если начнут свою работу с разных состояний.

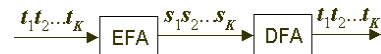


Рис. 3.2. Конечный автомат EFA и его обращение DFA

Чтобы процедура шифрования была обратима, для шифрующего автомата должен существовать обратный ему автомат. Конечный автомат является обратным и называется обращением другого автомата, если он преобразует любую выходную последовательность этого автомата в его входную последовательность. На рис 3.2 конечный автомат DFA преобразует выходную последовательность $y_1y_2...y_k$ конечного автомата EFA в его входную последовательность $l_1l_2...l_k$, и в силу этого является обращением автомата EFA.

Поскольку работа конечного автомата зависит от его начального состояния, то для каждого допустимого начального состояния автомата должно существовать парное ему начальное состояние его обращения, при котором эти автоматы, будучи соединенными последовательно, работают как тривиальный автомат, без изменения выдающийся на выход все то, что поступило на его вход. Следовательно, для корректного расшифрования шифртекста автомат, выполняющий эту процедуру, должен быть синхронизирован с шифрующим автоматом - он должен начать свою работу с состоянием, парного начального состоянию последнего. Сигнал синхронизации (*синхропосылка*) передается до начала передачи шифрованных данных всякий раз при установлении соединения или восстановлении его после сбоя в канале связи. Синхропосылка может передаваться только в открытом виде - на момент ее получения автомат расшифрования на принимающей стороне не готов к работе.

3.10 В настоящее время одним из наиболее популярных видов потоковых шифров является шифр гаммирования, в котором соответствующий конечный автомат является безঝовым и используется для выработки последовательности элементов гаммы.

Для наложения гаммы на данные используется *операция побитового суммирования по модулю 2* или *побитового исключающего ИЛИ* (см. п.3.4).

$$X_{i+1} = \oplus(X_i), \quad G_i = \mathcal{S}(X_i),$$

$$T'_i = T_i \oplus G_i.$$

Такой шифр называют двоичным аддитивным.

С точки зрения надежности шифра все допустимые операции наложения гаммы одинаковы. Условием стойкости шифра гаммирования является невозможность определить по известному фрагменту гаммы другие ее части или восстановить структуру порождающего ее конечного автомата. Для стороннего наблюдателя, обладающего лишь ограниченными вычислительными возможностями, выработанная гамма должна быть неотличима от случайной последовательности.

3.11 В блочных шифрах результат шифрования каждого блока зависит только от его содержания, естественно, не считая секретного ключа:

$$T'_i = E_K(T_i).$$

Как следствие, при шифровании двух одинаковых блоков данных получаются идентичные блоки шифртекста:

$$T_i = T_j \rightarrow T'_i = T'_j.$$

Из этой особенности блочных шифров следует очевидный способ их анализа - статистический. Если известен закон распределения блоков открытого текста, то, анализируя статистику блоков шифртекста, можно установить соответствие между ними. Чтобы исчерпать подобную возможность, размер блока выбирается достаточно большим. Например, при длине блока в один байт анализ шифра можно выполнить вручную без использования вычислительной техники; при длине блока в два байта такой анализ легко реализуется на персональной ЭВМ и занимает несколько секунд; при длине блока в 32 бита компьютерный анализ осуществим, хотя требует больше времени и объема шифрованных данных. При дальнейшем увеличении размера блока статистический анализ становится все менее осуществимым на практике. В большинстве современных шифров размер блока выбирается равным 64 битам. Для такого размера исчерпывающий анализ шифртекста практически неосуществим из-за невозможности набрать требуемую статистику. При дальнейшем увеличении длины блока усложняется не только криптоанализ, но и сам алгоритм шифрования. Для шифров очень распространенной на сегодняшний день архитектуры, называемой "сбалансированной сетью Фейстеля" (balanced Feistel network) условием эффективной реализации в виде программы для ЭВМ является равенство половины длины блока криптоалгоритма величине машинного слова. В настоящее время подавляющее количество компьютеров в мире 32 битовые, и по этой причине выбирать размер блока для упомянутой архитектуры шифров больше 64 бит бессмысленно, а с точки зрения эффективности реализации вредно.

Хотя для блочных шифров с достаточно большим размером блока провести исчерпывающий статистический анализ в общем случае невозможно, тем не менее, анализируя зашифрованные данные, легко обнаружить наличие одинаковых блоков в исходных данных, что позволяет выявить стабильные паттерны, имеющиеся в них. Если данные содержат массивы повторяющейся информации, их рекомендуется рандомизировать - модифицировать с использованием случайных или псевдослучайных процедур.

Условия, которым должен удовлетворять стойкий блочный шифр, сформулировал Шеннон в своих работах по теории шифрования: такой шифр должен обладать свойствами *рассеивания* и *перемешивания*.

Рассеивание - это свойство шифра, при котором один символ исходного текста влияет на несколько символов шифртекста (идеально - на все символы одного блока). Если это условие выполнено, то при шифровании двух блоков данных, отличающихся хотя бы одним символом, должны получиться совершенно непохожие друг на друга блоки шифртекста. Такая же картина должна иметь место и для зависимости шифртекста от ключа - один символ ключа должен влиять на несколько символов шифртекста.

Перемешивание - это свойство шифра скрывать зависимость между символами исходного текста и шифртекста. Если шифр достаточно хорошо "перемешивает" биты исходного текста, то соответствующий шифртекст не содержит никаких статистических, и, тем более, функциональных закономерностей (для стороннего наблюдателя, обладающего ограниченными вычислительными ресурсами).

Если шифр в достаточной степени обладает обоими указанными свойствами, то любые изменения в блоке открытого текста приводят к тому, что для наблюдателя все символы в зашифрованном блоке получают новые значения,

равновероятные в области их определения и независимые друг от друга. Такой шифр невозможно вскрыть способом, менее затратным с точки зрения количества необходимых операций, чем полный перебор по множеству возможных значений ключа. Данное условие является обязательным для шифра рассматриваемого типа, претендующего на то, чтобы считаться хорошим.

4. Архитектура блочных шифров

В настоящем выпуске мы с вами начнем рассмотрение архитектуры блочных шифров, безоговорочно доминирующей в традиционной криптографии на нынешнем этапе ее развития. Из предыдущих выпусков нам стало известно, что сильный блочный шифр должен обладать свойствами рассеивания и перемешивания, и что главная идея в построении таких шифров - использование последовательности большого числа простых шифрующих преобразований. Поэтому настоящим выпуск начнем с рассмотрения этих преобразований - основных строительных кирпичиков серьезных шифров.

Под простым шифрующим преобразованием здесь и далее понимается такое преобразование, который реализует аппаратно относительно несложной логической схемой или программно несколькими компьютерными командами. Можно выделить следующие группы простых шифров:

- шифр перестановок - заключается в перестановках структурных элементов шифруемого блока данных - битов, символов, цифр;
- шифр замен - заключается в замене одних значений на другие по индексной таблице, замене подвергаются группы элементов шифруемого блока - битов или символов;
- шифр функциональных преобразований - заключается в выполнении сдвигов, логических и арифметических операций над элементами данных.

Нижее дана подробная характеристика каждого из упомянутых типов преобразований:

Шифры перестановок чрезвычайно просто реализуются аппаратно - разводкой проводников на плате или в кристалле, при этом совсем не требуется каких-либо дополнительных затрат, как при проводниках, связывающие регистры аппаратуры, так или иначе присутствуют в схеме. В то же самое время эти преобразования очень неэффективно реализуются программно на процессорах общего назначения. Как правило, вычислительные затраты составляют не менее двух машинных циклов на каждый двоичный разряд в модифицируемом блоке, если только в перестановках нет согласованности. Этой причиной, в частности, объясняется тот факт, что многие шифры, широко использующие операции данного типа, имеют при прочих равных условиях существенно менее эффективные реализации по сравнению с шифрами, их не использующими. Например, американский стандарт шифрования криптоалгоритм DES при вдвое меньшем количестве шагов в цикле шифрования по сравнению с Российским стандартом (16 против 32) имеет примерно вдвое более медленную оптимальную реализацию для процессоров Intel x86.

Общие виды замен аппаратно реализуются с помощью запоминающих устройств, программно - индексированным чтением из оперативной памяти, что, по сути, одно и то же: замена для элемента данных X берется из *вектора* или *узла* *замен* V , являющегося массивом заменяющих значений, индексированным заменяемым элементом данных: X заменяется на

$$y = V[X].$$

Программно такая операция реализуется за одну команду, не считая операции загрузки индекса в соответствующий регистр. Размер памяти, необходимый для хранения вектора заменяющих, определяется следующим соотношением:

$$|V| = 2^{|X|} |Y|,$$

где $|X|$ и $|Y|$ - размеры заменяемого и заменяющего блоков в битах соответственно, размер вектора *замен* V также получается в битах. Из приведенной формулы видно, что он растет экспоненциально с ростом размера заменяемого блока. В силу этого выполнение подстановки в масштабах всего шифруемого блока невозможно - потребовался бы слишком большой объем памяти для хранения вектора. Поэтому преобразуемый блок данных разделяют на фрагменты, обычно одинакового размера, и выполняют замену в этих подблоках независимо друг от друга. Для повышения стойкости шифра замену различных частей шифруемого блока следует выполнять с использованием разных векторов *замен*, которые все вместе составляют *таблицу подстановок* или *таблицу замен*. Для хранения этой таблицы требуется участок памяти следующего размера:

$$|S| = n_p |V| = n_p 2^{|X|} |Y|,$$

где n_p - число подблоков размера $|X|$, в которых производится подстановка. Как уже отмечалось выше, размер таблицы подстановок быстро увеличивается с ростом размера заменяемого, и, особенно, заменяемого блока, что влечет за собой возрастание требований к необходимому для реализации шифра объему памяти. С другой стороны, увеличение этих размеров усложняет криптоанализ и, тем самым, повышает стойкость шифра, поэтому на практике их

следует выбирать на границе разумности, ведь криптоалгоритм проектируется на достаточно длительный срок, а возможности электронной техники увеличиваются очень быстро. В алгоритме DES суммарный объем блоков подстановки равен $|S_{DES}| = 82^2 \cdot 4 = 2^{11} \text{бит} = 256 \text{байт}$. В отечественном стандарте это величина того же порядка: $|S_{ГОСТ}| = 82^2 \cdot 4 = 2^{11} \text{бит} = 256 \text{байт}$. Следует помнить, что указанные шифры разрабатывались в семидесятые годы, когда понятие "микросхема" еще только начинало входить в наш обиход, обычная емкость микросхемы запоминающего устройства составляла несколько десятков, максимум сотни битов, а объем оперативной памяти 32Кбайта считался совсем неплохим вариантом для компьютера. Вполне естественно, что созданные в то время криптоалгоритмы отражали суровые реалии тех дней. Сейчас эта проблема практически отсутствует, и потому современные шифры гораздо более свободны в данном отношении. Так, в криптоалгоритме BLOWFISH подстановки производятся следующим образом: каждый из 4-х байтов, составляющих 32-битовое слово, заменяется на 4-байтовое слово, полученные слова преобразуются в одно с помощью логических и арифметических операций. Соответственно размер одной таблицы замен в этом алгоритме равен $|S_{BLOWFISH}| = 42^8 \cdot 32 = 2^{15} \text{бит} = 4 \text{Кбайт}$.

Функциональные преобразования - это унарные и бинарные логические и арифметические операции, реализуемые аппаратно логическими схемами, а программно - одной-двумя компьютерными командами. Теоретически, возможно использовать любую операцию, которая может быть сформулирована в терминах логических функций. Однако на практике дело всегда ограничивается теми из них, которые имеются в наборах команд универсальных процессоров и реализованы аппаратно в виде микросхем. Из логических операций это основные логические функции - инверсия, и бинарные - побитовые И, ИЛИ, ИСКЛЮЧАЮЩЕЕ ИЛИ, из арифметических - изменение знака (переход к дополнению коду), и бинарные - сложение, вычитание, умножение, деление по модулю некоторого числа, из битовых манипуляций - циклические сдвиги.

Как же построить надежный шифр из элементарных операций указанного типа? Наиболее очевидная идея - каскадировать их, как это показано на рисунке 1. Символы P, S, F на нем обозначены операции перестановок (Permutation), замен (Substitution), функциональных преобразований (Function) соответственно. Ключевые элементы (K_i) могут комбинироваться с преобразуемыми данными в операциях подстановок и функциональных преобразований.

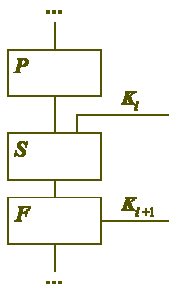


Рис. 1. Фрагмент составного шифра - комбинация большого числа элементарных шифрующих преобразований.

Не имеет смысла комбинировать две однотипные операции подряд. Если чередовать процедуры различного типа, сложность результирующего преобразования (степень перемешивания и рассеивания) будет выше. Это очень легко объяснить: при комбинировании двух операций их сложности складываются за вычетом некоего "дефекта сложности", который тем больше, чем более схожи две операции. Например, суперпозиция (результат последовательного выполнения) двух битовых перестановок может быть выражена одной перестановкой. То же справедливо для двух подстановок, выполняемых в одних и тех же границах заменяемых подблоков. Прибавление к блоку данных двух ключевых элементов равносильно прибавлению одного, равного им сумме. Во всех рассмотренных случаях добавление к операции еще одной такой же вообще не приводит к возрастанию сложности, а следовательно и стойкости преобразования.

Даже если комбинировать две различные операции, принадлежащие одной и той же группе, сложность полученного преобразования меньше, чем могла бы быть, если бы они были разделены операцией другого типа. На рисунке 2 изображены два трехэлементных шифрующих преобразования, составленных из одних и тех же операций. Сложность и стойкость преобразования, изображенного справа, по только что изложенным соображениям выше, чем у левого.

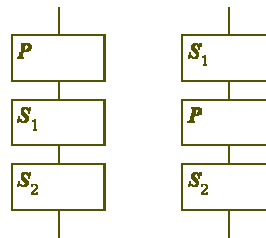


Рис. 2. Пара шифрующих преобразований из трех элементарных операций, использованных в различном порядке.

Каким же условиям должен удовлетворять шифр, не только обладающий необходимой стойкостью, но, вдобавок к этому удобный в реализации и использовании. Рассмотрение начнем с требований, которые были особенно актуальными четверть века назад, когда возможности микросистемных приборов были весьма ограничены и соображения экономичности и самой возможности реализации шифров на имеющейся элементной базе играли определяющую роль. Сейчас их актуальность заметно меньше, но, тем не менее, они остаются в достаточной степени важными для того, чтобы учитываться и в современных разработках.

1. Операции за- и расшифровки должны быть близкими настолько, чтобы могли быть выполнены одним и тем же аппаратным или программным модулем - это диктуется требованием экономичности реализации.
2. Объем ключевой информации должен быть относительно небольшим. Разумным является такой размер ключа, при котором невозможно его нахождение путем перебора по всему ключевому пространству, с определенным запасом на возможный прогресс электронной техники. В настоящее время граница практической осуществимости подбора ключа находится где-то в районе 60-64 бит. Соответственно, разумным может считаться размер ключа 80-256 бит. Данное требование вытекает из необходимости хранить ключи на любых носителях, включая нетрадиционные, например - на персональных миниатюрных магнитных карточках.
3. Реализация шифра (код программы и постоянные данные) должна быть достаточно компактной для того, чтобы "уместиться" на микроконтроллерах с относительно невысоким объемом запоминающего устройства - последнее требование также диктуется соображениями экономичности реализации.

Рассмотрим, каким образом можно построить шифр, удовлетворяющий указанным требованиям. Начнем с условия обратимости процедуры зашифрования. Из него вытекает, что все преобразования, непосредственно модифицирующие шифруемые данные, должны быть обратимыми, то есть при выполнении не должно теряться информация. Перестановка обратима по определению. Непараметризованная замена имеет обратную операцию если она соръективна, то есть если каждое возможное заменяемое значение встречается в соответствующем источнике замен ровно один раз. Параметризованная, то есть зависящая от значения ключевого элемента, замена обратима в том случае, если при каждом фиксированном значении параметра соответствующая простая замена обратима. Бинарная функциональная операция обратима, если при каждом фиксированном значении второго, модифицирующего, аргумента задаваемое ей отображение сюръективно, это равносильно условию, что уравнение модификации элемента данных $Y = f(X, K)$ всегда однозначно разрешимо относительно модифицируемого элемента (X). Унарные функциональные операции можно рассматривать как некоторые бинарные с фиксированным вторым операндом. Из простых обратимых унарных и бинарных логических операций над числами конечной разрядности следует отметить инверсию и операцию побитового исключающего или, из арифметических - изменение знака числа, сложение или вычитание в пределах разрядной сетки числа, умножение и деление по модулю простого числа. Если шифрующее преобразование определено как цепочка описанных выше элементарных операций, то достаточно просто построить обратное ему, если только все элементарные операции в цепочке обратимы.

Для этого достаточно выполнить перечисленные ниже требования, которые, хотя и не являются абсолютно необходимыми, тем не менее исчерпывают практически все случаи эффективно реализуемых преобразований, и потому на практике всегда принимаются во внимание:

1. Все шифрующие преобразования должны принимать на входе и выдавать на выходе блок данных одного и того же размера, не считая дополнительных входов для параметра замены и второго операнда функционального преобразования.
2. Замены, применяемые непосредственно к шифруемому данным, должны быть обратимыми, параметризованные замены должны быть обратимыми при каждом фиксированном значении параметра.

3. Уравнение функционального преобразования шифруемых данных с помощью бинарной операции должно быть всегда однозначно разрешимо относительно преобразуемого (первого) операнда.

В этом случае для составного шифрующего преобразования, имеющего линейную структуру, можно очевидным образом сконструировать обратное преобразование: оно строится комбинированием обращений его составных частей в порядке, обратном тому, в котором они использовались в исходном преобразовании, как это показано на рисунке 3.

Если прямое преобразование определяется формулой

$$T = T_{n, X_n} T_{n-1, X_{n-1}} \dots T_{2, X_2} T_{1, X_1}$$

то обратное ему преобразование задается следующей формулой:

$$T^{-1} = T_{1, X_1}^{-1} T_{2, X_2}^{-1} \dots T_{n-1, X_{n-1}}^{-1} T_{n, X_n}^{-1}$$

В данных формулах T_{i, X_i} обозначает одну из перечисленных выше простых операций преобразования (перестановку, подстановку или функциональную перацію), возможно, зависящую от параметра - ключевого элемента K_i . Операции группируются справа налево, то есть $(T_i T_j)(X)$ обозначает $T_j(T_i(X))$.

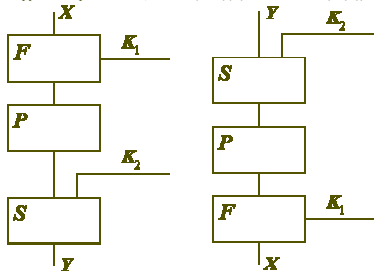


Рис. 3. Шифрующее преобразование с линейной структурой и обратное ему шифрующее преобразование.

Для того, чтобы прямое и обратное преобразование было возможно реализовать в одном аппаратном блоке или программном модуле, они должны быть идентичны с точностью до используемых ключевых элементов. Это означает, что шифрующее преобразование должно быть "антисимметрично" самому себе - для каждого его шага, находящегося на определенном расстоянии от начала преобразования, на точно таком же расстоянии от его конца должен располагаться обратный ему шаг, использующий тот же самый ключевой элемент:

$$T^{-1} = T_i$$

если для всех i справедливо следующее условие:

$$T_{i, X_i}^{-1} = T_{n-i+1, X_{n-i+1}} \text{ для всех } K_i, \text{ или } T_i^{-1} \equiv T_{n-i+1}$$

Если данное условие выполняется, процедуры за-и расшифрования могут осуществляться одним и тем же программным и аппаратным модулем и отличаются только порядком использования ключевых элементов.

Теперь рассмотрим требование относительного небольшого размера ключа - как было отмечено выше, он не должен быть намного больше размера, достаточного для исключения практической возможности его нахождения полным перебором по всему ключевому пространству. Так как это "критический" размер составляет в настоящее время величину порядка восьми байт, разумный размер ключа не превышает 256 бит. Ясно, что для получения необходимой стойкости шифра придется использовать достаточно большое количество элементарных шагов преобразования, нуждающихся в наборе ключевых элементов, намного (в разы) превосходящем по размеру ключ. Поэтому во всех шифрах подобного типа применяется процедура "развертывания", с помощью которой из небольшого ключа строится массив ключевых элементов нужного размера.

Процедура "развертывания" ключа должна удовлетворять следующим требованиям:

Биты (символы) каждого ключевого элемента должны быть равновероятны и статистически независимы друг от друга.

Биты (символы) каждого ключевого элемента должны быть статистически независимы от битов (символов) нескольких соседних ключевых элементов. Это условие должно выполняться в пределах такого количества шагов шифрования, на котором еще можно проследить статистические зависимости между битами (символами) шифруемых блоков.

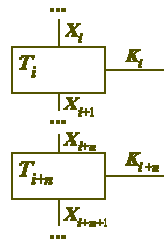


Рис. 4. Шаги шифрующего преобразования.

Данное требование проиллюстрировано на рисунке 4. Для любой пары шагов шифрующего образования, не обязательно смежных, допускается тем меньшая коррелированность используемых ими ключевых элементов, чем больше коррелированность выхода первого из них со входом второго:

$$|C(X_{i+1}, X_{i+n})| \leq |C(K_i, K_{i+n})|$$

где C - некоторая мера коррелированности. Понятно, что данная зависимость не является количественно строгой, она отражает суть вопроса лишь качественно. Опять же следует отметить, что данное требование не является абсолютно необходимым. Нет ничего страшного, если оно выполняется не в полной мере для отдельных пар шагов преобразования. Однако систематическое игнорирование этого правила приводит к тому, что криптоанализ шифра значительно облегчается.

Криптостойкость последовательности ключевых элементов является неизбежным, но весьма полезным ее свойством, так как сама по себе гарантирует выполнение двух вышеприведенных требований. Возможны различные подходы к выработке ключевых элементов для шагов шифрования - от самых простых, до обладающих сложностью, сопоставимой со сложностью самого шифра. Например, в качестве ключевых элементов для шагов шифрования можно просто брать фрагменты ключа, как это делается в отечественном стандарте шифрования. Можно вырабатывать ключевые элементы с помощью генератора псевдослучайных чисел. Здесь спектр возможных решений чрезвычайно широк - от сравнительно несложных схем выработки гаммы на основе сдвиговых регистров с обратной связью до генерации последовательности элементов с помощью того же самого криптоалгоритма. Последний подход реализован, например, в шифре BLOWFISH. Конечно, он значительно увеличивает стойкость шифра, но и существенно затрудняет его эффективную реализацию. Например, в упомянутом шифре BLOWFISH построение массива ключевых элементов вычислительно эквивалентно выполнению более 500 циклов шифрования, что делает его непригодным для реального практического использования всюду за исключением систем, в которых смена ключа происходит достаточно редко, и объемы массивов, шифруемых на одном ключе, велики.

Наконец, требование компактности реализации алгоритма с точки зрения кода и используемых постоянных данных приводит к идеологии его построения из одинаковых групп преобразований, повторенных нужное число раз. В этом случае реализация алгоритма работает итеративно - результат каждой итерации, за исключением последней, является входом для следующей итерации. Кроме того, очевидно, что каждая повторяющаяся группа преобразований, из которых построен криптоалгоритм, должна удовлетворять рассмотренному выше требованию антисимметричности прямого и обратного криптопреобразований, которое в данном случае должно выполняться на уровне отдельных групп.

Требование компактности вспомогательных массивов данных можно выполнить, используя на разных итерациях преобразования один и тот же комплект векторов замен.

Таким образом, в настоящем выпуске мы с вами рассмотрели базовые требования, в соответствии с которыми строятся современные шифры:

- построение шифров из простых преобразований - перестановок, подстановок, элементарных функциональных операций и чередование операций различного типа;
- циклическая, итеративная структура алгоритма - одна итерация состоит из нескольких простых преобразований, итерации отличаются друг от друга только используемыми ключевыми элементами;
- антисимметричная структура алгоритма, позволяющая достичь структурной эквивалентности процедур за-и расшифрования, они отличаются друг от друга только последовательностью использования ключевых элементов;
- использование ключа относительно небольшого размера и выработка из него массива ключевых элементов для шагов преобразования с помощью процедуры "развертывания" ключа.