Данный материал является фрагментом электронного учебника
по информационной безопасности и может обновляться.
При цитировании рекомендуется использовать ссылку:

Амелин Р. В. Информационная безопасность. Конспект лекций // Амелин Р.В. Лаборатория преподавателя [Электронный ресурс]. URL: rv-lab.ru (2017).

Лекция 1. Симметричное шифрование

1.1. Основные понятия криптографии

Криптография – наука о методах обеспечения *конфиденциальности* и *аутентичности* информации.

До 70-х годов XX века криптографией называлась область науки и практической деятельности, связанная с изучением и разработкой методов шифрования данных. В настоящее время сфера интересов криптографии включает в себя множество так называемых *криптографических алгоритмов*, к которым, помимо шифрования/дешифрования относятся алгоритмы хэширования, формирования и проверки электронной подписи, распределения ключей и множество других алгоритмов, каждый из которых предназначен для противодействия определенным угрозам информационной безопасности со стороны возможного нарушителя (противника, злоумышленника) или нежелательных воздействий естественного характера. Большинство криптографических алгоритмов строятся на математической основе. Устройства, программы, документы, созданные на основе криптографических алгоритмов, называются *криптографическими средствами*. Система обеспечения информационной безопасности, использующая криптографические средства, называется *криптографической системой*. Можно сказать, что криптография изучает криптографические системы.

Криптография является частью более общей науки – *криптологии*. Вторая часть криптологии – *криптоанализ.* До 70-х годов XX века эта наука занималась оценкой сильных и слабых сторон методов шифрования, а также разработкой методов взлома шифров. В настоящее время ***криптоанализ*** – область науки, занимающаяся изучением криптографических систем защиты в поиске способов нарушения информационной безопасности, которую обеспечивает данная система. Таким образом, криптоанализ изучает методы прочтения зашифрованного текста без ключа, методы подделки электронной цифровой подписи (без знания закрытого ключа автора) и т.д. Криптография и криптоанализ – две сильно взаимодействующие науки с противоположными целями. За последние несколько десятилетий они непрерывно и интенсивно развиваются, причем достижения одной из них заставляют другую быстро реагировать совершенствованием своего аппарата.

Базовым объектом изучения в криптологии обычно является система, состоящая из следующих элементов (рис. 0.1):

A – отправитель (автор) сообщения;

B – получатель сообщения;

M – сообщение (message);

открытый канал связи, по которому передается сообщение;

E – противник (enemy), имеющий доступ к открытому каналу связи и перехватывающий сообщения. В зависимости от возможностей противника различают пассивный и активный перехват. При пассивном перехвате противник только «подслушивает» все со-

общения. Активный перехват предполагает изменение сообщений отправителя, навязывание получателю собственных сообщений, удаление сообщений из канала.

В криптографической литературе участникам информационного обмена по традиции даются имена Алиса и Боб (**A**lice и **B**ob), а противника называют Евой (**E**ve).

Рис. 0.1. Базовая схема криптологии

К приведенной схеме сводятся разные частные случаи. Например, когда Алиса сохраняет на диске зашифрованный файл, можно представить, что она отправляет сообщение самой себе (из будущего) и хочет защитить его конфиденциальность от противника, имеющего доступ к ее жесткому диску.

Когда Алиса и Боб ведут двустороннюю переписку, они по очереди выступают в роли отправителей и получателей сообщения.

Если Алиса и Боб имеют закрытый канал для обмена сообщениями (канал, к которому никто посторонний не может получить доступ ни при каких обстоятельствах; возможно, Алиса и Боб – последние люди на земле), то криптографические средства для защиты этих сообщений им вряд ли понадобятся. Но в реальной жизни очень сложно представить себе гарантированно закрытый канал связи.

1.2. Шифрование

Шифрование – это процесс преобразования исходного сообщения *M* (называемого *открытым текстом*) в форму *M'* (*зашифрованный текст* или *шифртекст*). При этом провести обратное преобразование *M'* в *M* возможно только обладая некоторой дополнительной информацией, называемой *ключом*.

Шифрование нередко путают с *кодированием*, но между двумя этими процессами есть значительная разница. Кодирование также представляет собой преобразование исходного сообщения в другую форму, но цель этого преобразования – удобство обработки или передачи сообщения. Например, символьный текст кодируется в двоичный (каждый

символ заменяется последовательностью нулей и единиц) для того, чтобы его можно было хранить и обрабатывать в ЭВМ, а двоичный текст преобразовывается в последовательность электрических импульсов, для того, чтобы стала возможной его передача по кабелю. Цель шифрования – противоположная. Текст зашифровывается для того, чтобы посторонние лица, не обладающие ключом, не могли бы воспринять заложенную в нем информацию, даже располагая этим зашифрованным текстом. Таким образом, шифрование является *средством обеспечения конфиденциальности* информации.

Алгоритмы шифрования делятся на две большие группы:

1. Симметричное (традиционное шифрование).
2. Шифрование с открытым ключом.

1.3. Симметричное шифрование

В симметричных алгоритмах шифрования *один и тот же ключ* K используется для того, чтобы зашифровать сообщение и для его последующей расшифровки. Таким образом, и *отправитель* и *получатель* сообщения должны располагать одним и тем же ключом. Схематично это можно записать в виде:

$M' = E(M, K)$

$M = D(M', K)$,

где E – функция шифрования (encrypt), а D – функция дешифрования (decrypt), обе используют ключ K в качестве одного из параметров.

Исторически симметричное шифрование появилось первым. Более того, до середины XX века это была единственная разновидность шифрования. Симметричные алгоритмы широко применяются и в настоящее время.

Далее мы рассмотрим ряд простых алгоритмов симметричного шифрования, на примере которых легко проанализировать такие их характеристики, как устойчивость к различным видам криптоанализа, а также некоторые базовые принципы криптографии. Затем будут рассмотрены алгоритмы, используемые в современных информационных системах.

Все алгоритмы симметричного шифрования можно разделить на три класса:

1. Подстановочные алгоритмы.
2. Перестановочные алгоритмы.
3. Алгоритмы, использующие подстановку и перестановку (практически все современные алгоритмы, разработанные для защиты информации в ЭВМ).

Кроме того, среди симметричных алгоритмов шифрования можно выделить:

1. Алгоритмы, предполагающие *одноразовое использование ключа* (т.е. для шифрования каждого нового сообщения используется новый ключ).

2. Алгоритмы *с многократно используемым ключом*. Очевидно, такие алгоритмы должны быть более надежными (и пройти более серьезные проверки на безопасность), поскольку у противника появляются дополнительные возможности криптоанализа, о чем будет сказано далее.

1.4. Обзор классических алгоритмов шифрования

Подстановочные алгоритмы шифрования работают по следующему принципу: каждый символ (или последовательность символов) исходного сообщения заменяются другим символом (или другой последовательностью символов).

Рассмотрим конкретные примеры.

Шифр Цезаря

Самым древним и самым простым из известных подстановочных шифров является шифр, использовавшийся Юлием Цезарем. В этом шифре каждая буква исходного сообщения заменяется буквой, находящейся в алфавите на три позиции после нее.

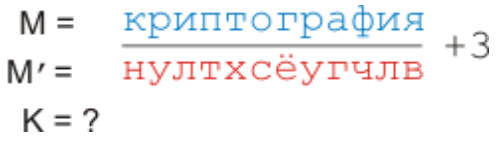

Рис. 0.2. Пример шифрования по Цезарю

Особенностью шифра Цезаря, как несложно заметить, является отсутствие ключа. Число 3 в данном случае ключом не является, поскольку не выбирается отправителем сообщения, а используется всегда. Во времена Юлия Цезаря это не было слабостью шифра (поскольку сама идея сокрытия информации путем преобразования текста была незнакомой его противникам), но в настоящее время первым правилом криптографии является следующее допущение:

Стойкость любого шифра определяется в предположении, что противнику полностью известен механизм шифрования и единственной информацией, которой он не располагает, является ключ.

Данное допущение особенно актуально для настоящего времени, когда сложность шифров достигла такого уровня, что зашифровывать и расшифровывать сообщения вручную просто невозможно. Для этих целей используется программное обеспечение, которое заинтересованные лица могут детально проанализировать и, таким образом, полностью восстановить алгоритм шифрования.

Это правило имеет тенденцию нарушаться в тех областях, когда криптографическое программное обеспечение не предназначено для широкого распространения. Например, алгоритмы, использующиеся в системах электронного голосования, правительственной связи и др. Разработчики этих систем считают сокрытие алгоритмов шифрования фактором, усиливающим безопасность. *Ошибочность этого предположения научно установлена*. Злоумышленник, серьезно заинтересованный в том, чтобы взломать криптографическую защиту и нарушить конфиденциальность данных, почти наверняка найдет способ получить доступ к самой программе, которая по определению не может быть также хорошо защищена, как обрабатываемые ею данные, и изучить используемые алгоритмы. Сокрытие алгоритмов и деталей архитектуры таких систем лишь препятствует их изучению

независимыми исследователями и увеличивает опасность того, что алгоритмы, положенные в их основу, будут недостаточно надежными[1].

Рассмотрим вариацию шифра Цезаря, при которой сдвиг в алфавите не обязательно равен 3, а выбирается произвольно, согласно договоренности между отправителем и получателем сообщения. В этом случае шифр Цезаря становится полноценным шифром с ключом K (потенциально неизвестном противнику).

Однако легко заметить, что в качестве ключа могут быть выбраны лишь числа в диапазоне от 1 до 32 (для русского алфавита). Действительно, шифр является циклическим: Я + 1 = А, но и Я + 34 = А. То есть, число 34, выбранное в качестве ключа, будет эквивалентно ключу 1, ключ 35 – ключу 2 и т.д. Противнику ничего не стоит перебрать все 32 возможных ключа и обнаружить нужный.

Таким образом, модифицированный шифр Цезаря является неустойчивым к взлому методом перебора возможных ключей по причине их малого диапазона или, другими словами, малой длины ключа.

Термин ***длина ключа*** отражает особенности современных алгоритмов шифрования. Современные алгоритмы шифрования разрабатываются для ЭВМ, поэтому открытый текст, шифртекст и ключ представляют собой двоичные последовательности бит (иногда удобно представлять их как числа). Длина ключа – это длина соответствующей двоичной последовательности, причем, предполагается, что для шифрования может быть выбран любой ключ с равной вероятностью. Таким образом, общее число ключей равно 2^n, где n – длина ключа, а вероятность «угадать» ключ с одной попытки составляет $1/2^n$. С ростом n сложность подбора ключа растет экспоненциально. В настоящее время надежной считается длина ключа 128 бит.

С этой точки зрения длина ключа шифра Цезаря равна неполным 5 битам, что явно небезопасно.

Основная классификация методов криптоанализа базируется на возможностях противника. Мы будем рассматривать эти атаки по мере знакомства с алгоритмами шифрования, после чего обобщим полученные сведения.

Атака на основе шифртекста – метод криптоанализа, при котором криптоаналитик располагает только зашифрованным сообщением или несколькими сообщениями, зашифрованными с использованием одного ключа. Целью криптоаналитика является восстановление открытых текстов и/или восстановление ключа.

[1] Так, например, в 2007 году в результате утечки информации оказались обнародованы алгоритмы, используемые в популярной технологии KeeLoq — системы охраны, применяемой в противоугонных средствах автомобильной защиты. После исследования, проведенного независимыми экспертами, оказалось, что при сравнительно скромных затратах времени и вычислительных ресурсов (два дня работы компьютеров общей ценой около 10 000 евро) злоумышленники могут вскрыть сначала секретный ключ какой-либо конкретной машины, а на его основе — но теперь уже за секунды — цифровой ключ любого другого автомобиля этой же компании. Двумя годами раньше аналогичный скандал возник с системой электронного голосования, использовавшейся в США, исходные коды которой оказались случайно выложенными в общий доступ на сайте компании. Исследование показало уязвимость системы к целому спектру популярных атак.

Эта атака *наиболее легко реализуема*, поскольку все, что необходимо для ее осуществления – перехват зашифрованного текста, который, как уже отмечалось, передается по открытому каналу связи. В то же время – это *наиболее слабый и неудобный вид атаки*, поскольку возможности криптоаналитика серьезно ограничены.

Модифицированный шифр Цезаря, очевидно, является неустойчивым к атаке на основе шифртекста. Опробуя все возможные 32 ключа, противник быстро видит, что получается в результате дешифрования – бессмысленный набор символов (следовательно, ключ не подходит), или что-то, похожее на настоящий текст. Для этого достаточно расшифровать только начало сообщения.

На примере шифра Цезаря мы увидели и первую из многочисленных техник криптоанализа – ***метод грубой силы*** или *полный перебор ключей*. Этот метод служит своеобразной оценкой прочности шифра сверху. Если длина ключа мала, то не спасут никакие конструктивные особенности шифра – ведь криптоаналитик всегда может просто перебрать ключи и с вероятностью 50% найдет нужный после 2^{n-1} попыток. Шифр Цезаря противостоять методу грубой силы не может.

2. Моноалфавитный шифр (шифр простой замены)

Один из хорошо известных подстановочных шифров. Каждому символу алфавита открытого текста ставится в соответствие некоторый символ другого алфавита (эти алфавиты могут и совпадать).

Ключом к данному шифру будет являться таблица соответствий, которую удобно представить в виде символов, выписанных в алфавитном порядке тех букв, которые они заменяют. Другими словами, ключом является перестановка символов алфавита зашифрованного текста.

При шифровании каждый символ открытого текста заменяется другим символом в соответствии с ключом.

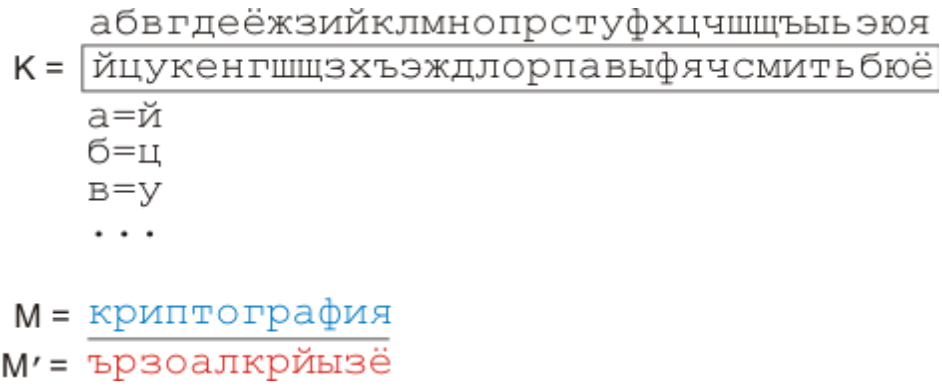

Рис. 0.3. Пример шифрования шифром простой замены

В данном случае число возможных ключей равно числу возможных перестановок из 33 букв, то есть, 33!. Даже при использовании миллиона компьютеров, проверяющих миллион возможных ключей в секунду, перебор всех вариантов займет больше миллиона

лет. Таким образом, моноалфавитный шифр является стойким к взлому методом перебора возможных ключей.

Однако данный шифр достаточно просто поддается криптоанализу, который начинается с подсчета каждого символа шифртекста и определения частоты его встречаемости. Для достаточно длинного сообщения (порядка 4–5 предложений) этой информации будет достаточно, чтобы сопоставить ее с таблицей частоты встречаемости букв языка.

Все естественные языки имеют характерное частотное распределение символов. Например, буква «О» - встречается в русском языке чаще других, а буква «Ф» – самая редкая (см. табл. 1).

Табл. 1. Таблица частот встречаемости букв русского языка.

Символ	Вероятность	Символ	Вероятность	Символ	Вероятность
пробел	0.175	К	0.028	Ч	0.012
О	0.089	М	0.026	Й	0.010
Е	0.072	Д	0.025	Х	0.009
А	0.062	П	0.023	Ж	0.007
И	0.062	У	0.021	Ю	0.006
Н	0.053	Я	0.018	Ш	0.006
Т	0.053	Ы	0.016	Ц	0.004
С	0.045	З	0.016	Щ	0.003
Р	0.040	Ь	0.014	Э	0.003
В	0.038	Б	0.014	Ф	0.002
Л	0.03	Г	0.013		

На основе частоты встречаемости символов зашифрованного текста можно сделать предположения о некоторых, наиболее часто встречающихся из них, а затем, опираясь на эти предположения, постепенно восстанавливать слова текста, начиная с самых коротких – предлогов и союзов. Так, в английском языке достаточно легко идентифицируется артикль the – самая часто встречающаяся комбинация из трех букв.

Таким образом, моноалфавитные шифры уязвимы к атаке на основе шифртекста, а конкретно – к анализу статистических особенностей исходного текста. Идеальный шифр не позволит злоумышленнику сделать какие-то заключения об открытом тексте на основе зашифрованного текста.

Нетрудно заметить, что шифр Цезаря также является моноалфавитным шифром.

3. Шифр Гронсфельда

Рассмотрим шифр, представляющий собой модификацию шифра Цезаря. В качестве ключа используется последовательность цифр произвольной фиксированной длины. Каждая цифра этой последовательности записывается под одним символом открытого текста,

причем если длина ключа меньше длины текста, ключ циклически повторяется. Зашифруем слово «информатика» ключом «123».

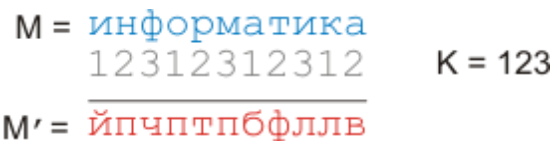

Рис. 0.4. Пример использования шифра Гронсфельда

Данный шифр (описанный Жюль Верном в романе «Жангада») относится к семейству *многоалфавитных шифров* (или *шифров сложной замены*). В многоалфавитном шифре 1-й символ открытого текста шифруется с помощью моноалфавитного шифра, ключом к которому является перестановка K_1, 2-й символ – ключом K_2 и т.д., n-й символ – ключом K_n, а n+1-й – снова ключом K_1, где n – количество используемых алфавитов (или шифров простой замены). В приведенном примере n=3.

Особенности многоалфавитных шифров хорошо демонстрирует шифр Гронсфельда (и, в частности, приведенный пример). Мы видим, что одна и та же буква *«и»* превращается то в *«й»*, то в *«л»* в зависимости от того, какая цифра ключа использовалась для шифрования, а буква *«а»* может быть зашифрована как *«б»* или *«в»*. Более того, буква *«п»*, встречающаяся в зашифрованном тексте три раза, каждый раз означает разные буквы.

Таким образом, хотя статистические особенности исходного текста и будут проявляться с цикличностью n, где n – длина ключа, при достаточно большом n (десять и более цифр, при том, что противнику эта длина неизвестна), таблицы частот дают гораздо большую погрешность, не говоря уже о том, что проверять предположения, восстанавливая фрагменты по смыслу, становится практически невозможным.

Главная слабость шифра Гронсфельда в том, что каждая буква зашифрованного текста отстоит от «своей» буквы открытого текста не более чем на девять позиций в алфавите, а это дает противнику возможность легко проверять различные предположения. Например, предположив, что начало зашифрованного текста «йпчптпбфллв» расшифровывается как «крипто», уже на первом символе, противник отбросит этот вариант, поскольку буква *«к»* не может превратиться в *«й»* после сдвига.

Эта проблема исчезает в модификации шифра Гронсфельда, где в качестве ключа выступает не цифровая, а буквенная последовательность. Порядковый номер буквы открытого текста складывается с записанной под ней буквой ключа и получается порядковый номер буквы зашифрованного текста (который берется по модулю мощности алфавита, т.е. Г + Э = 4 + 31 = 35; 35 mod 33 = 2 = Б; Г + Э = Б).

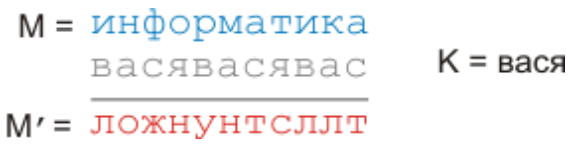

Рис. 0.5. Модифицированный шифр Гронсфельда

Но модифицированный шифр Гронсфельда не является идеальным. Поскольку ключ циклически повторяется, противник как минимум, может *проверять предположения* относительно открытого текста. Допустим, у противника имеется зашифрованный текст, а также предполагаемый открытый текст (полученный с помощью какого-либо метода криптоанализа). Как проверить, действительно ли этот открытый текст соответствует зашифрованному? Очень просто: нужно «вычесть»[2] из зашифрованного текста открытый текст. Если результат представляет собой циклически повторяющуюся последовательность символов, то все правильно. Вероятность получить такую закономерность для другого (неправильного) осмысленного текста ничтожно мала.

Что касается метода получения предполагаемого открытого текста, то противник может использовать атаку на основе шифртекста, точнее, нескольких шифртекстов, полученных с использованием одного и того же ключа.

Пусть c_1 – первая буква первого шифртекста, а c_2 – первая буква второго шифртекста. По определению шифра Гронсфельда эти буквы были получены следующим образом:

$c_1 = m_1 + k; c_2 = m_2 + k,$

где m_1 – первая буква первого открытого текста, а m_2 – первая буква второго открытого текста, k – первый символ ключевой последовательности (которая будет идентичной в обоих случаях).

Что будет если вычесть из одного зашифрованного текста другой? Рассмотрим разность первых букв шифртекстов:

$c_1 - c_2 = m_1 + k - m_2 – k = m_1 - m_2$

Другими словами, вычитая из одного шифртекста другой, мы получаем разность открытых текстов, не зависящую от использованного ключа. А это дает аналитику массу зацепок – например, он может подбирать слова одного из текстов и проверять, получается ли что-то осмысленное в другом тексте.

Кроме того, если противник располагает открытым текстом и соответствующим ему зашифрованным текстом, он может получить ключ путем вычитания и попробовать применить его к другим зашифрованным текстам (поскольку нередко один ключ используется для многих сообщений). Такая атака называется ***атакой с известным открытым текстом*** и, очевидно, шифр Гронсфельда (и, конечно, все ранее рассмотренные шифры) уязвим по отношению к ней.

Чтобы получить пару «открытый текст – зашифрованный текст» противник может вынудить отправителя сообщения зашифровать определенную информацию (например, в истории разведки не раз применялся прием, когда с этой целью нарочно организовывалась утечка информации). Кроме того, противник может делать предположения о словах, которые могут встречаться в тексте исходного сообщения (особенно продуктивны попытки угадать первое слово), и, пробуя эти слова, вычислять фрагменты возможного ключа. Далее эти фрагменты пробуются в других участках зашифрованного текста и, если в резуль-

[2] При вычитании одного текста из другого отсчет идет по алфавиту назад. Если получилось отрицательное число, к нему прибавляется мощность (число букв) алфавита. Математически происходит вычитание порядковых номеров букв по модулю мощности алфавита.

тате обратного преобразования получается осмысленная последовательность, криптоаналитик на верном пути.

4. Многобуквенные шифры

В определении подстановочных шифров сказано, что при шифровании может заменяться не каждый отдельный символ исходного сообщения, а сразу группа символов – на другую группу символов. Такие шифры называются ***многобуквенными***. Рассмотрим *шифр Плейфейера*, в котором единицей шифрования является биграмма (пара букв), заменяемая другой парой букв.

Шифр предназначен для английского алфавита. Ключом является кодовая фраза, которая записывается в первые ячейки квадратной решетки 5x5 (повторяющиеся буквы пропускаются). Затем квадрат в алфавитном порядке заполняется буквами, которые не вошли в кодовую фразу. I и J считаются одной буквой.

Заполним решетку с ключевым словом MONARCHY.

```
MONAR
CHYBD
EFGIK
LPQST
UVWXZ
```

Рис. 0.6. Кодовая решетка для шифра Плейфейера

Исходный текст разбивается на биграммы. При этом если две одинаковые буквы открытого текста при разбиении образуют одну биграмму, между ними вставляется символ X. Т.е. вместо BALLOON шифруем BALXLOON.

Если буквы биграммы стоят в одной строке (столбце), они заменяются их правыми (нижними) соседями с учетом циклического сдвига. В нашем примере OR шифруется как NM, а OP – как HV.

Если буквы биграммы оказываются в разных строках и столбцах, то каждая из пары букв заменяется буквой, находящейся на пересечении ее строки и столбца, в котором находится вторая буква. Например, BE шифруется как CI, а OS – как AP.

Слово INFORMATION будет зашифровано как GAPHMORSFAAW.

Данный шифр сохраняет статистические особенности исходного текста в том смысле, что можно построить таблицу частот биграмм для языка и проанализировать частоты биграмм зашифрованного текста. Однако если букв в английском языке 26, то биграмм уже $26^2 = 676$, поэтому задача существенно усложняется и без значительных объемов зашифрованного текста обречена на провал.

Конечно, атака с известным открытым текстом по-прежнему будет эффективной.

Другой интересный пример многобуквенного шифра – *шифр Хилла*. Он представляет собой систему из m линейных уравнений с m коэффициентами. Шифр заменяет каждые m букв открытого текста на m букв зашифрованного текста. Например, при $m = 3$ имеем систему уравнений (где n – мощность алфавита):

$$c_1 = (k_{11}\, p_1 + k_{12}\, p_2 + k_{13}\, p_3) \bmod n$$
$$c_2 = (k_{21}\, p_1 + k_{22}\, p_2 + k_{23}\, p_3) \bmod n$$
$$c_3 = (k_{31}\, p_1 + k_{32}\, p_2 + k_{33}\, p_3) \bmod n$$

Ключом шифрования будет являться матрица коэффициентов K, а само шифрование можно представить как произведение вектора, составленного из порядковых номеров букв открытого текста (первая буква имеет номер 0) на ключевую матрицу K:

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}$$

Для расшифровки зашифрованный текст необходимо умножить на матрицу, обратную к K (K^{-1}).

Шифр Хилла также хорошо скрывает частоту символов исходного сообщения, но уязвим к криптоанализу с известным открытым текстом. Поскольку система линейная, противнику понадобится m^2 пар «открытый текст – зашифрованный текст», чтобы вычислить ключевую матрицу. Однако наиболее эффективный метод криптоанализа для данного шифра – ***криптоанализ с избранным открытым текстом***, когда противник имеет возможность получать зашифрованные образцы специально подобранного им открытого текста[3]. Подумайте, что будет представлять собой зашифрованный текст, если зашифровать последовательность БАА ($p_1 = 1$, $p_2 = 0$, $p_3 = 0$).

В ***перестановочных алгоритмах*** символы открытого текста изменяют порядок следования в соответствии с правилом, которое определяется ключом.

Простейший пример перестановки: символы открытого текста не слева направо, а сверху вниз, при этом длина столбца ограничена. Результатом будет текст, выписанный по строкам.

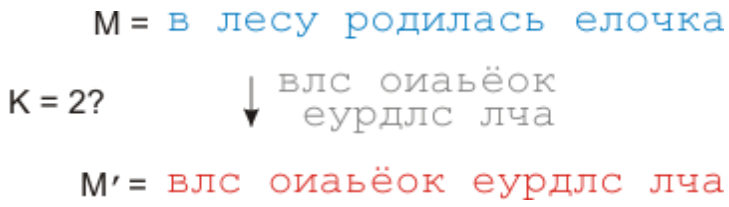

Рис. 0.7. Простейший перестановочный шифр

Такой шифр будет весьма уязвим к перебору ключей (в качестве ключа будет выступать длина столбца), поскольку ключ в любом случае не может быть длиннее, чем длина самого сообщения.

Рассмотрим более интересный пример: *квадратичная решетка*. Ключом к этому шифру является квадратная решетка, стороны которой содержат четное число ячеек. Чет-

[3] Например, имеется система передачи данных по каналу связи в зашифрованном виде, которая использует один и тот же ключ для шифрования всех данных. При этом противник может перехватывать передающиеся по каналу связи зашифрованные сообщения, а также передавать свои собственные. В этом случае он может получить интересующие его пары «открытый текст — зашифрованный текст» и с их помощью провести криптоанализ ключа.

верть ячеек решетки вырезаются по следующему принципу: если некоторая ячейка вырезана, то нельзя вырезать те ячейки, в которые она переходит при повороте решетки на 90, 180 и 270 градусов.

Чтобы зашифровать текст, решетка с прорезями накладывается на расчерченный квадрат, после чего буквы текста последовательно записываются в прорези. Когда все прорези заполнены, решетка поворачивается на 90 градусов, причем, согласно принципу построения решетки, прорези при этом окажутся на месте незаполненных ячеек. В прорези записывается продолжение текста, после чего решетка снова поворачивается и, таким образом, процедура повторяется еще два раза. Если текст не поместился в один квадрат, таким же образом заполняется следующий. Оставшиеся пустыми ячейки последнего квадрата заполняют случайными символами.

Рис. 0.8. Пример шифрования с помощью квадратичной решетки

Квадратичная решетка, очевидно, уязвима к криптоанализу с известным открытым текстом, причем для двоичного алфавита эта уязвимость значительно меньше, чем для естественно-языковых алфавитов.

1.5. Одноразовый блокнот

Выше были рассмотрены классические алгоритмы симметричного шифрования, применявшиеся, в докомпьютерную эпоху. Каждый из этих алгоритмов уязвим для определенных видов криптоанализа. Самые совершенные из современных алгоритмов со временем также обнаруживают свои слабые стороны.

В настоящее время признано существование единственного алгоритма шифрования, который невозможно вскрыть. Этот шифр известен как *шифр Вернама* или *одноразовый блокнот*. Открытый текст складывается с *абсолютно случайным ключом, совпадающим с ним по размеру*[4]. После этого ключ уничтожается (т.е. не используется для шифрования других текстов). Абсолютная криптостойкость шифра доказана Клодом Шенноном.

На практике одноразовые блокноты применяются очень редко (лишь для сообщений высшей секретности). Во-первых, изготовление такого блокнота достаточно дорого (т.к. абсолютно случайная последовательность не может генерироваться алгоритмически), а блокнот предназначен лишь для одноразового использования. Во-вторых, возникает *про-*

[4] Сложение происходит по модулю мощности алфавита. Если зашифровывается текст, представленный в двоичном виде, то операция шифрования представляет собой исключающее или (XOR), примененное к ключу и открытому тексту.

блема передачи ключа: единственный надежный вариант – личная встреча. Действительно, предположим у отправителя и получателя сообщения есть надежный канал обмена информацией. Тогда почему бы не передавать по этому каналу незашифрованные сообщения. Если же отправить ключ, зашифровав его другим алгоритмом, вся система окажется надежной не более, чем этот алгоритм.