Информационная безопасность

Лебедева Т.Ф.

Криптографические методы и средства для защиты информации:
Основные термины и понятия криптографии

Термин	Определение
открытый текст (plaintext)	исходное, незашифрованное сообщение
зашифрованный текст, шифротекст (ciphertext)	данные, полученные после применения криптосистемы (обычно — с некоторым указанным ключом).
шифрование	процесс преобразования открытого текста в зашифрованный с целью сделать непонятным его смысл
расшифрование (дешифрование)	процесс обратного преобразования шифротекста в открытый текст
криптография (от др.-греч. κρυπτός — скрытый и γράφω — пишу)	наука, которая изучает методы сохранения содержания сообщения в тайне и дает возможность преобразовать исходную информацию так, что ее восстановление возможно только при знании ключа
криптоанализ	наука, изучающая математические методы нарушения конфиденциальности и целостности информации, наука о вскрытии шифра
криптология (kryptos - тайный, logos - наука)	наука – объединяющая криптографию и криптоанализ и занимающуюся вопросами обратимого преобразования информации с целью защиты от несанкционированного доступа, оценкой надежности систем шифрования и анализом стойкости шифров или наука о создании и взломе шифров

Термин	Определение
шифр	совокупность заранее оговоренных способов преобразования исходного секретного сообщения с целью его защиты
система шифрования, или **шифросистема**	это любая система, которую можно использовать для обратимого изменения текста сообщения с целью сделать его непонятным для всех, кроме тех, кому оно предназначено.
криптографическая стойкость	способность криптографического алгоритма противостоять криптоанализу
криптоаналитическая атака	использование специальных методов для раскрытия ключа шифра и/или получения открытого текста
криптосистема	алгоритм шифрования и множество всевозможных ключей
пространство ключей	множество всех возможных ключей, доступных для использования в алгоритме.
ключ	параметр шифра, определяющий выбор конкретного преобразования данного текста. В современных шифрах криптографическая стойкость шифра целиком определяется секретностью ключа (Принцип Керхкофа).

Криптографические методы и средства для защиты информации: криптография древности

Около 4000 лет назад египтяне заменяли в важных текстах одни иероглифы на другие

Греция: Лисандр предложил скиталы – кожаная полоска наматывается на цилиндр определенной толщины

Пример моноалфавитного шифра

Леон Батисти Альберти. Первый полиалфавитный шифр: внутри код, снаружи сообщение. Время от времени двигаем внутренний диск, тем самым меняя код

1. *Конфиденциальность:* как сохранить информацию в секрете от всех, кроме имеющих доступ:
   - при передаче по незащищенному каналу связи
   - хранение данных на общедоступных носителях
2. *Целостность*: как обеспечить передачу данных в целостности и сохранности. Получатель сообщения должен быть в состоянии проверить внесены ли какие-либо изменения в ходе передачи сообщения;
3. *Аутентификация*: Получателю необходимо убедиться в том, что сообщение исходит от конкретного пользователя;
4. *Неоспоримость:* Отправитель сообщения должен быть лишен возможности отрицать, что именно он является автором сообщения.

Надежность криптографического алгоритма обеспечивается за счет сохранения в тайне ключей, если же в тайне необходимо сохранять суть самого алгоритма, то такой алгоритм называется ограниченным.

В современной криптографии проблемы секретности решаются с помощью ключей. Ключ **K** должен выбираться среди значений, принадлежащих множеству, называемому ключевым пространством.

Пусть **P** – открытый текст, **C** – шифротекст. После шифрования **C** передают по каналам связи. **E** – функция шифрования, **D** – функция расшифрования, тогда

$$E(P)=C, \qquad D(C)=P, \qquad D(E(P))=P$$

Функции шифрования и расшифрования зависят от ключей:

$$Ek_1(P)=C, \qquad Dk_2(C)=P.$$

Надежность алгоритма шифрования с использованием ключей достигается за счет их надлежащего выбора и последующего хранения в строжайшем секрете. В современной криптологии принято считать, что надежность шифра определяется только секретностью ключей.

Правило Керхкофа (1903) гласит, что весь механизм шифрования за исключением ключей предположительно известен противнику.

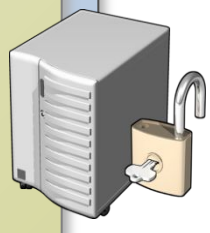
Криптографические методы и средства для защиты информации: классификация криптосистем

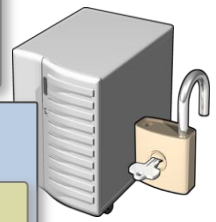
Симметричная криптосистема:

- один и тот же ключ (хранящийся в секрете) используется и для шифрования, и для *расшифрования*
- проблема распространения ключей

Асимметричная криптосистема:

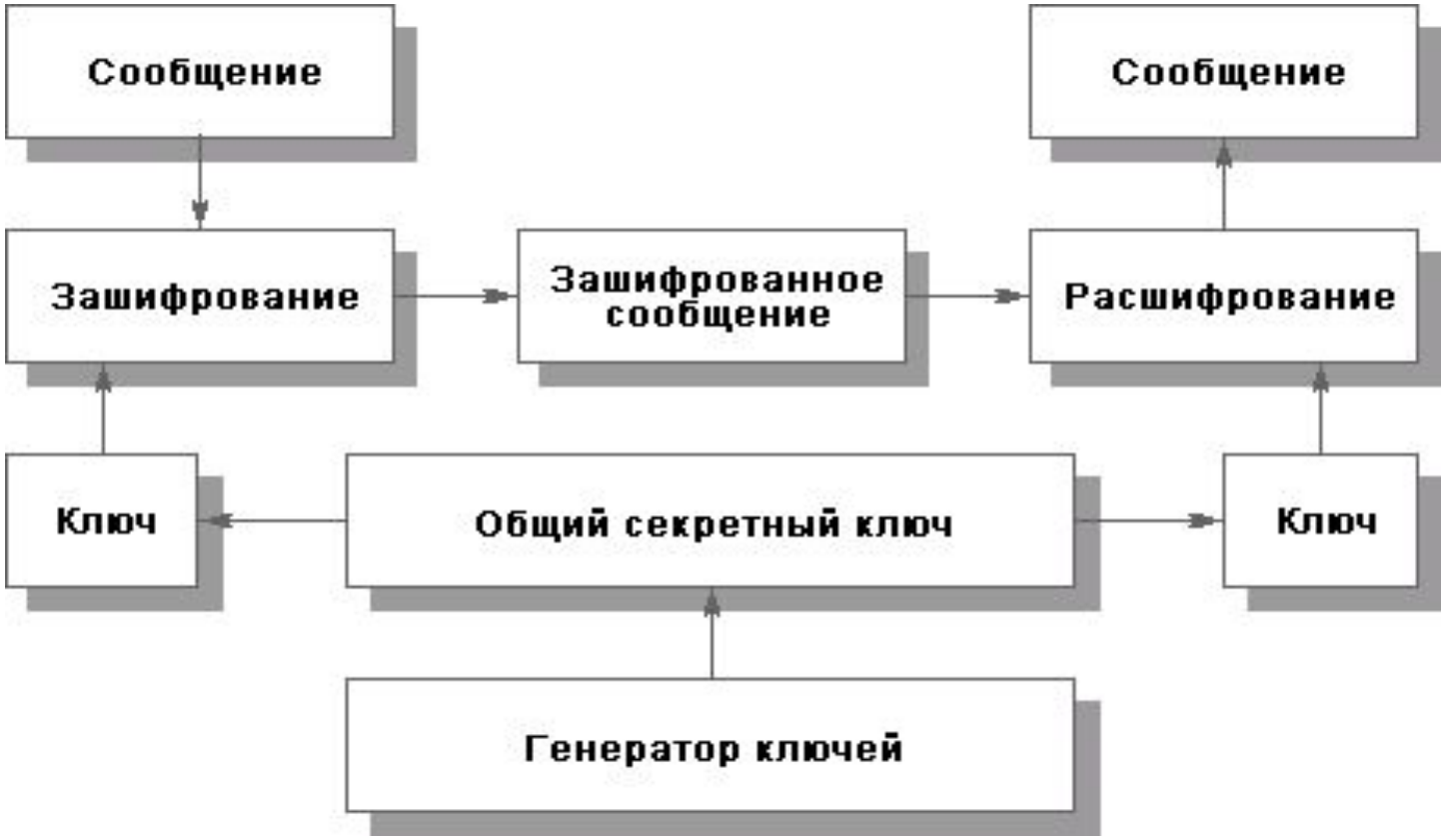
- используются два ключа. Один из них, несекретный (он может публиковаться вместе с другими открытыми сведениями о пользователе), применяется для шифрования, другой (секретный, известный только получателю) – для *расшифрования*
- Существенным недостатком асимметричных методов шифрования является их низкое быстродействие



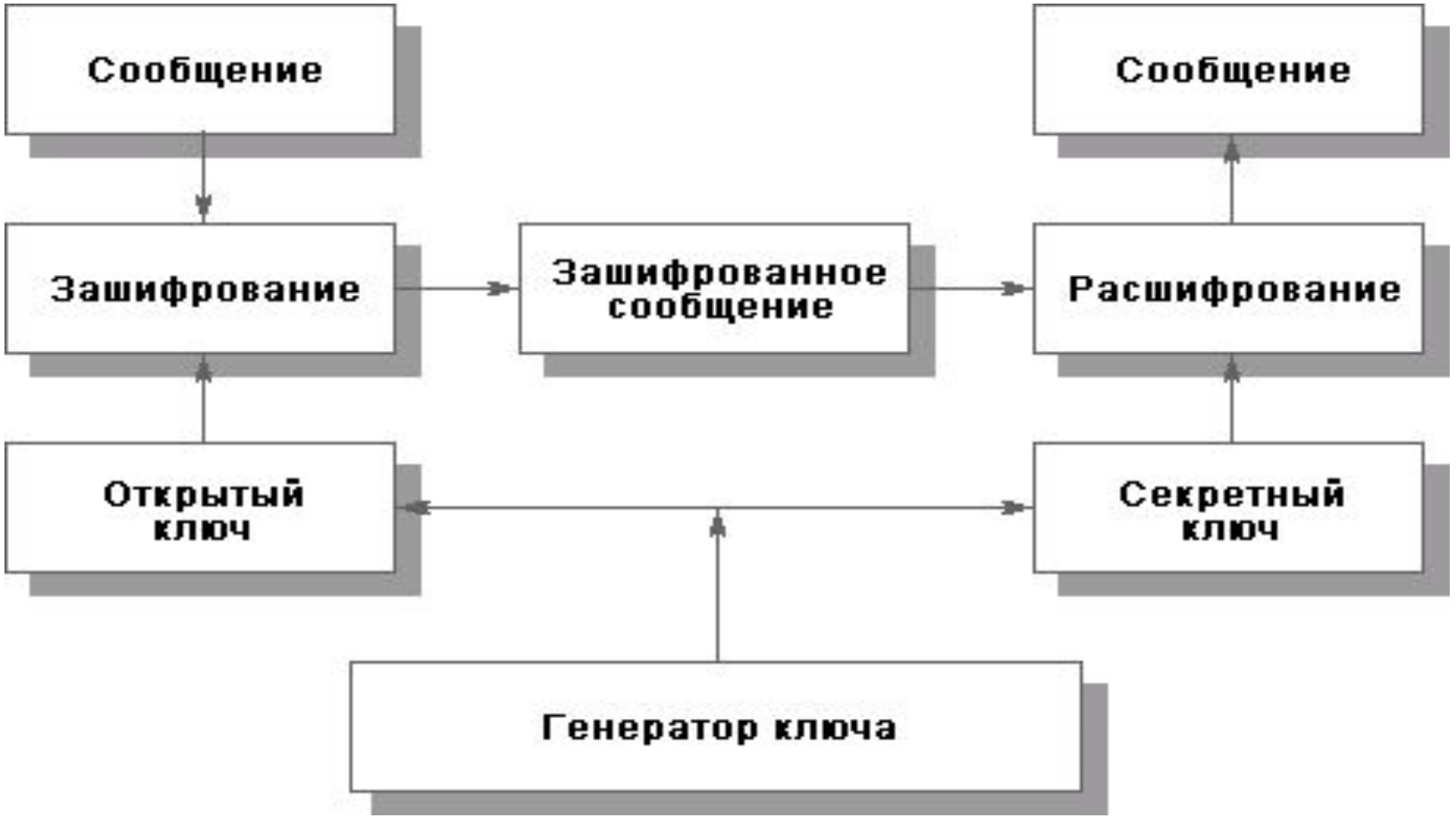


Составная криптосистема:

- *сообщение* сначала симметрично зашифровывают случайным ключом, затем этот ключ зашифровывают *открытым* асимметричным *ключом* получателя, после чего сообщение и ключ отправляются по сети



Использование симметричного метода шифрования



Использование асимметричного метода шифрования

Использование составного метода шифрования

Если противник узнал ключ не прибегая к криптоанализу, то говорят что ключ был скомпрометирован.

Попытка криптоанализа называется *атакой*.

Успешная криптоаналитическая атака называется *взломом* или *вскрытием*.

Известно 7 видов криптоаналитических атак:

1.Атака со знанием только шифротекста. В распоряжении криптоаналитика имеется несколько зашифрованных сообщений. Задача атаки состоит в нахождении открытого текста наибольшего числа перехваченных сообщений или ключей.

Дано: **C1=Ек1(P1), C2=Ек2(P2), ….. Ci=Екi(Pi)**

Найти **P1, P2,….Pi** или **К1,К2,…..Ki.**

2.Атака со знанием открытого текста. Криптоаналитик имеет доступ не только к шифрованным данным, но и к открытым текстам нескольких сообщений. От него требуется найти ключи, которые использовались при шифровании.

Дано: **P1, C1=Ек1(P1), P2,C2=Ек2(P2), ….. Pi,Ci=Екi(Pi)**

Найти: **К1,К2,…..Ki.**

***3.Атака с выбранным открытым текстом*.** Криптоаналитик не только знает шифрованные и открытые тексты нескольких сообщений, но и может определить содержание этих сообщений. Эта разновидность мощнее предыдущей, так как здесь криптоаналитик может по своему усмотрению выбирать открытый текст, подлежащий шифрованию и тем самым получать больше информации об используемых ключах.

Дано: **P1, C1=Eк1(P1), P2,C2=Eк2(P2), ….. Pi,Ci=Eкi(Pi)**, где

P1,P2,…Pi – выбранные криптоаналитиком открытые тексты

Найти: К1, К2, Кi.

4. ***Адаптивная атака с выбранным открытым текстом*.**

Это разновидность предыдущей атаки. Здесь криптоаналитик выбирает не только тексты посылаемых открытых сообщений, но и может менять свой выбор в зависимости от результатов шифрования.

***5.Атака с выбранным шифротекстом*.** Криптоаналитику предоставляется возможность выбора шифротекстов, подлежащих расшифрованию получателем. Имеется также доступ к соответствующим открытым текстам.

Дано: **C1, P1=Dк1(C1), C2,P2=Dк2(C2), ….. Ci,Pi=Dкi(Ci)**

Найти: **К1, К2,…..Ki.**

Атаки 3,4,5-го типов называется атаками с выбранным текстом.

***6.Атака с выбранным ключом*.**

Криптоаналитик обладает знаниями относительно правил, по которым выбираются ключи.

7.Атака с применением физической силы. Присутствуют подкуп, шантаж, пытки, чтобы получить сведения, необходимые для взлома криптосистемы.

Атаки со знанием открытого текста не так уж редко встречаются на практике: например, известны начало или конец сообщения.

Понятие стойкости алгоритма шифрования

Различные криптоалгоритмы обладают разной надежностью или стойкостью шифрования.

Стойкость зависит от того, насколько легко криптоаналитик может взломать шифр.

- Если при этом стоимость затрат на взлом превышает ценность получаемой информации, то владельцу шифра, возможно, беспокоиться не о чем.
- Если время, потраченное на взлом шифра, больше периода, в течение которого данные должны храниться в секрете, то данные, вероятно, - в безопасности.
- Если противник не накопил достаточного количества ваших сообщений, зашифрованных с помощью одного ключа, чтобы суметь определить этот ключ, то время его менять может быть не пришло.

Слова «возможно», «вероятно», «может быть» употреблены здесь не зря. Всегда существует вероятность, что в криптоанализе произойдут революционные изменения.

Под вскрытием (взломом) шифра обычно понимается решение одной из перечисленных ниже задач:

- *Полное вскрытие*. Криптоаналитик нашел ключ **К** такой, что **Dк (C)=P**
- *Глобальная дедукция*. Не зная ключа **К**, криптоаналитик отыскал альтернативный **Dк** алгоритм **A** такой, что **A(C)=P**.
- *Частичная дедукция*. Криптоаналитик получил неполную информацию о ключе или открытом тексте. Это может быть несколько битов ключа или дополнительные данные о структуре открытого текста.

Криптографический алгоритм называется безусловно стойким, если вне зависимости от того, каким объемом перехваченного шифротекста располагает криптоаналитик,

у него нет достаточной информации, чтобы восстановить исходный открытый текст.

Сложность криптоаналитической атаки на алгоритм шифрования можно охарактеризовать с помощью 3-х величин:

1) *Сложность по данным* - количество входных данных, необходимых для успешной криптоаналитической атаки на алгоритм шифрования.
2) *Вычислительная сложность* – время или количество операций, требуемое для успешной криптоаналитической атаки на алгоритм шифрования.
3) *Сложность по памяти* – объем памяти, который потребуется для успешной криптоаналитической атаки на алгоритм шифрования.

Часто под сложностью криптоаналитической атаки понимается максимальное значение из трех заданных величин. К примеру, если атака имеет сложность 2^128, то это означает, что для взлома шифра требуется выполнить 2^128 операций.

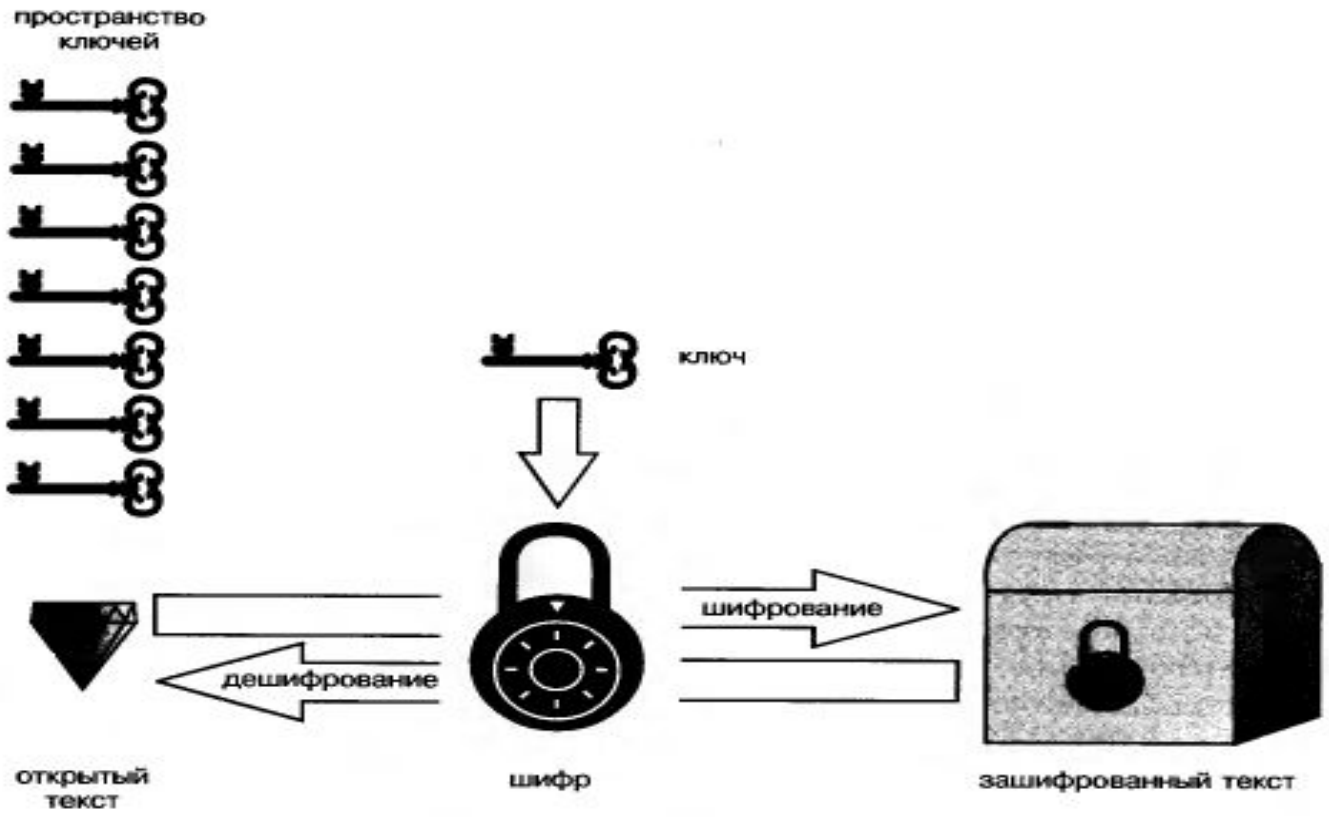


Рис. 2.1. Симметричное шифрование

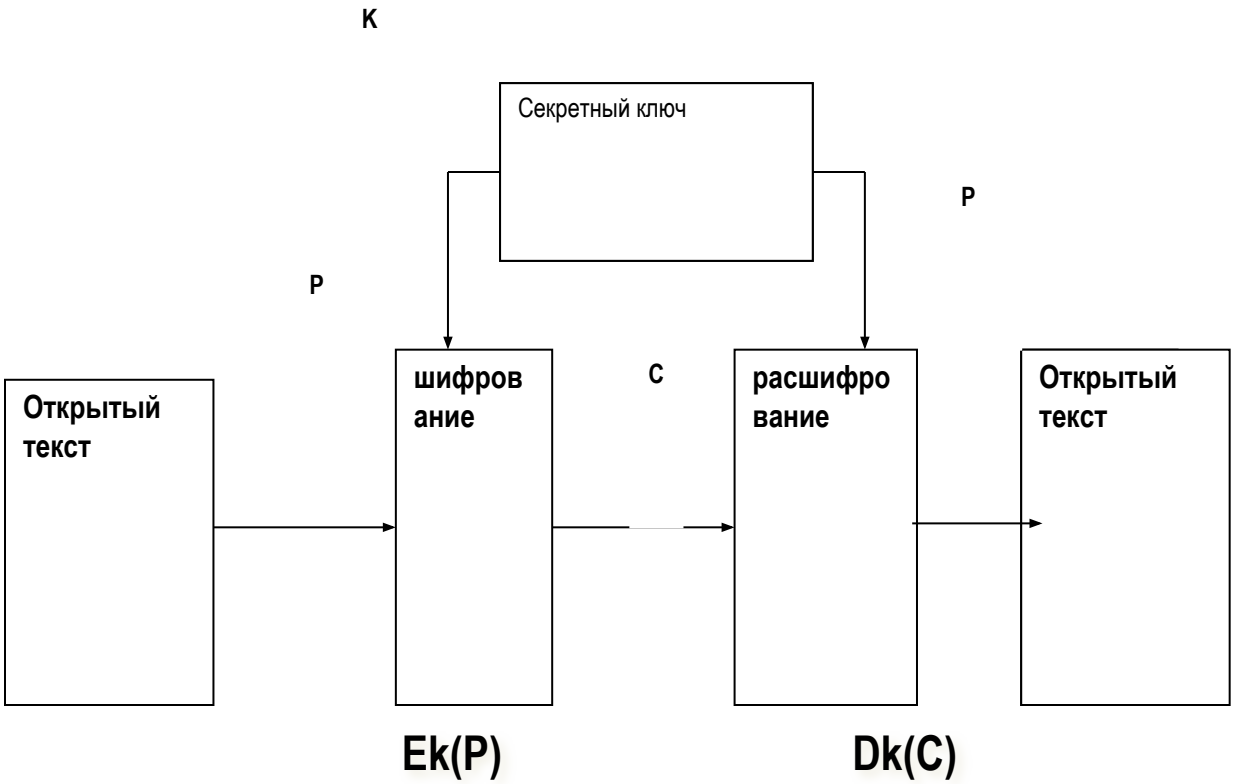


Рисунок Структурная схема симметричной криптографической системы

Классификация методов шифрования в симметричных криптографических системах

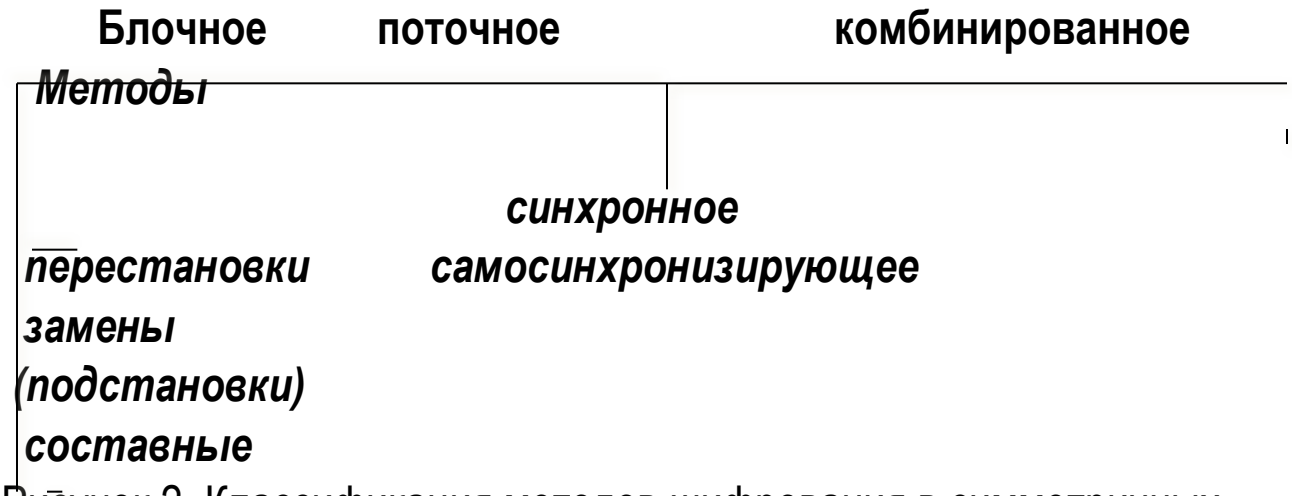


Рисунок 2 Классификация методов шифрования в симметричных криптографических системах

Если открытый текст разбивается на блоки, состоящие из нескольких бит (в современных алгоритмах - 64 бита),то алгоритм называется блочным.

Шифры, в которых открытый текст обрабатывается побитно, называются потоковыми (поточными) шифрами.

Блочные алгоритмы симметричного шифрования

Шифры простой перестановки:

Шифр простой перестановки переупорядочивает текст регулярным образом в соответствии с выбранным ключом или правилом перестановки.

Пример 1. в шифре простой колонной перестановки исходный текст записывается построчно (число букв в строке фиксировано), а шифротекст получается считыванием букв по колонкам.

Зашифруем текст: «Юстас Алексу встречайте связного» в виде таблицы из 6 строк и 5 колонок.

Ю С Т А С
А Л Е К С
У В С Т Р
Е Ч А Й Т
Е С В Я З
Н О Г О Ъ

В тексте не используются пробелы. Оставшиеся пустые клетки заполним символом «Ъ». Шифротекст получится считыванием букв по колонкам:

Ю А У Е Е НС Л В Ч С О Т Е С А В Г А К Т Й Я О С С Р Т З Ъ

Для расшифрования такого текста нужно знать ключ – правило 6х5.

Иногда результат одной перестановки еще раз переставляется – сложная перестановка

Пример 2. Зашифруем тест « ЗАСЕДАНИЕ СОСТОИТСЯ ЗАВТРА ЮСТАС», используя блоки из 6 символов и ключ **245136**, Делим текст без пробелов на блоки по 6 символов в каждом.

З А С Е Д А| Н И Е С О С| Т О И Т С Я| З А В Т Р А| Ю С Т А С Ъ

Переставляем символы в блоке согласно ключу: на 1-ое место в блоке ставим 2-ой символ; на 2-ое место – 4-й символ; на 3-е место в блоке ставим 5-ый символ и т.д. Получаем зашифрованный текст:

А Е Д З С А| И С О Н Е С| О Т С Т И Я| А Т Р З В А| С А С Ю Т Ъ

Для того, чтобы расшифровать шифротекст, нужно еще раз выполнить перестановку в блоке. Используется тот же самый ключ и для шифрования и для расшифрования.

Шифры замены (подстановки)

Шифром замены называется алгоритм шифрования, который производит замены каждой буквы открытого текста, на какой-либо символ шифрованного текста. Получатель сообщения расшифровывает его путем обратной замены.

В классической криптографии различают четыре разновидности шифров замены:

1 Простая замена **(одноалфавитный шифр).** — **Каждая буква открытого текста заменяется на один и тот же символ шифротекста.**

2 Омофонная замена — **Замена, аналогичная простой замене с единственным отличием: каждой букве открытого текста ставится в соответствие несколько символов шифротекста А→ 5, 13, 25, 57. Б→ 7, 19, 31, 43. Ключом являются шифровальные таблицы**

3 Блочная замена — **шифрование открытого текста производится блоками. Например, блоку «АБА» соответствует блок «РНР», а блоку «АББ» – «СЛЛ» и т. д.**

4 Многоалфавитная замена — **состоит из нескольких шифров простой замены. Например, могут использоваться 5 шифров простой замены, а какой из них применяется для шифрования конкретного символа в открытом тексте зависит от его положения в тексте.**

Одноалфавитные шифры

К одноалфавитным шифрам относится и исторически известный шифр Цезаря, в котором каждая буква открытого текста заменялась третьей по счету буквой латинского алфавита (с учетом циклического сдвига).

Вскрытие такого шифра осуществлялось путем перебора возможных ключей, в качестве которых используется величина сдвига букв в сообщении до появления осмысленного текста.

Примером шифра простой замены может служить программа ROT13, которую обычно можно найти в ОС UNIX. С ее помощью буква «A» открытого текста на английском языке заменяется на букву «N», «B» – на «O» и т. д. ROT13 циклически сдвигает каждую букву английского алфавита на 13 позиций вправо. Чтобы получить исходный текст, надо применить функцию шифрования ROT13 еще раз:

P= ROT13(ROT13(P)).

Одноалфавитные шифры

При простой одноалфавитной подстановке каждый знак **Mi,** принадлежащий алфавиту **A**, заменяется соответствующим знаком **Hi**, принадлежащий к алфавиту шифротекста **B**.

Соответствие между знаками алфавитов **A** и **B** задается с помощью кодовой таблицы или выражения. Например, при использовании обобщенного «шифра Цезаря» выражение, устанавливающее связь между алфавитами **A** и **B**, имеет вид:

F(Hi)=(F(Mi)+p) mod K, где

K – число знаков в алфавите.

p – постоянная величина сдвига

F(Hi) и **F(Mi)** - это числа, соответствующие буквам алфавитов **A** и **B**, которые в рассматриваемом случае состоят из одних и тех же символов.

Переход к шифротексту осуществляется в результате суммирования с некоторой постоянной величиной **p**. Шифрование этим способом эквивалентно сдвигу алфавита на фиксированное число позиций.

Однако, несмотря на то, что число возможных перестановок букв алфавита равно **26!,** шифры одноалфавитной замены не являются высокостойкими. Все шифры простой замены легко взламываются с использованием современных компьютеров, поскольку замена недостаточно хорошо маскирует стандартные частоты встречаемости букв в открытом тексте.

ШИФР ЦЕЗАРЯ

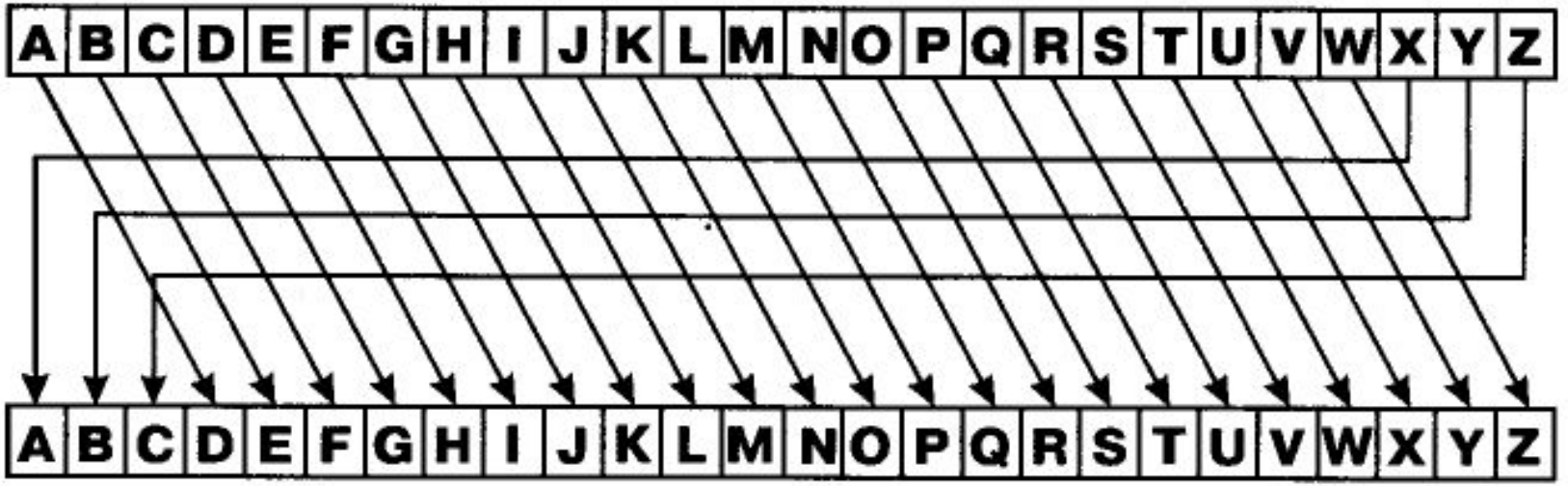


Рис. 2.3. Шифр Цезаря

Ключ: 3

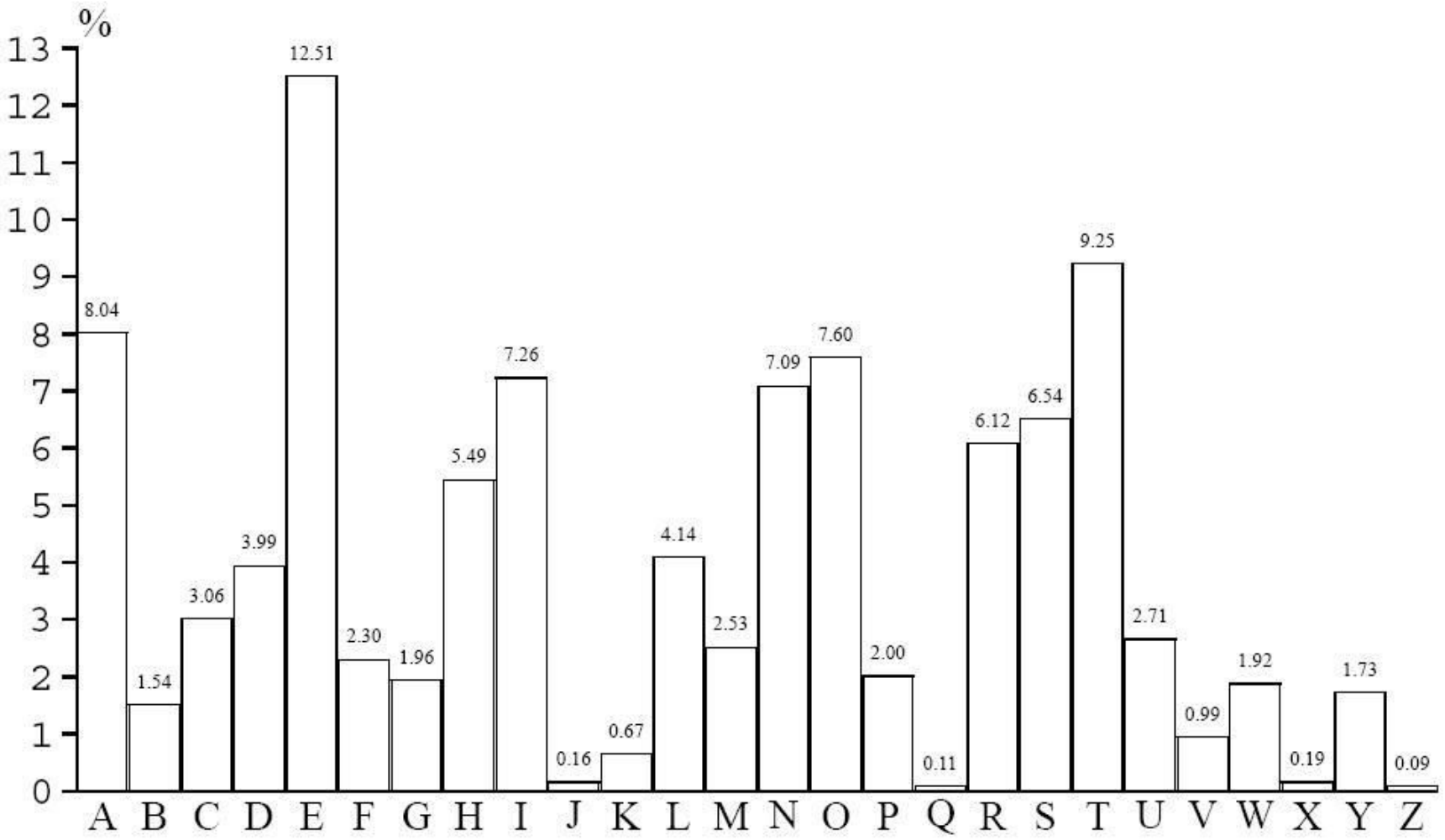
Открытый текст:

P = HELLO CAESAR CIPHER

Зашифрованный текст:

C = KHOOR FDHVDU FLSKHU

Частотный анализ для одноалфавитных шифров



Многоалфавитные шифры

В этих шифрах применяется несколько перемешанных алфавитов, поочередно используемых при замене символов исходного шифруемого сообщения.

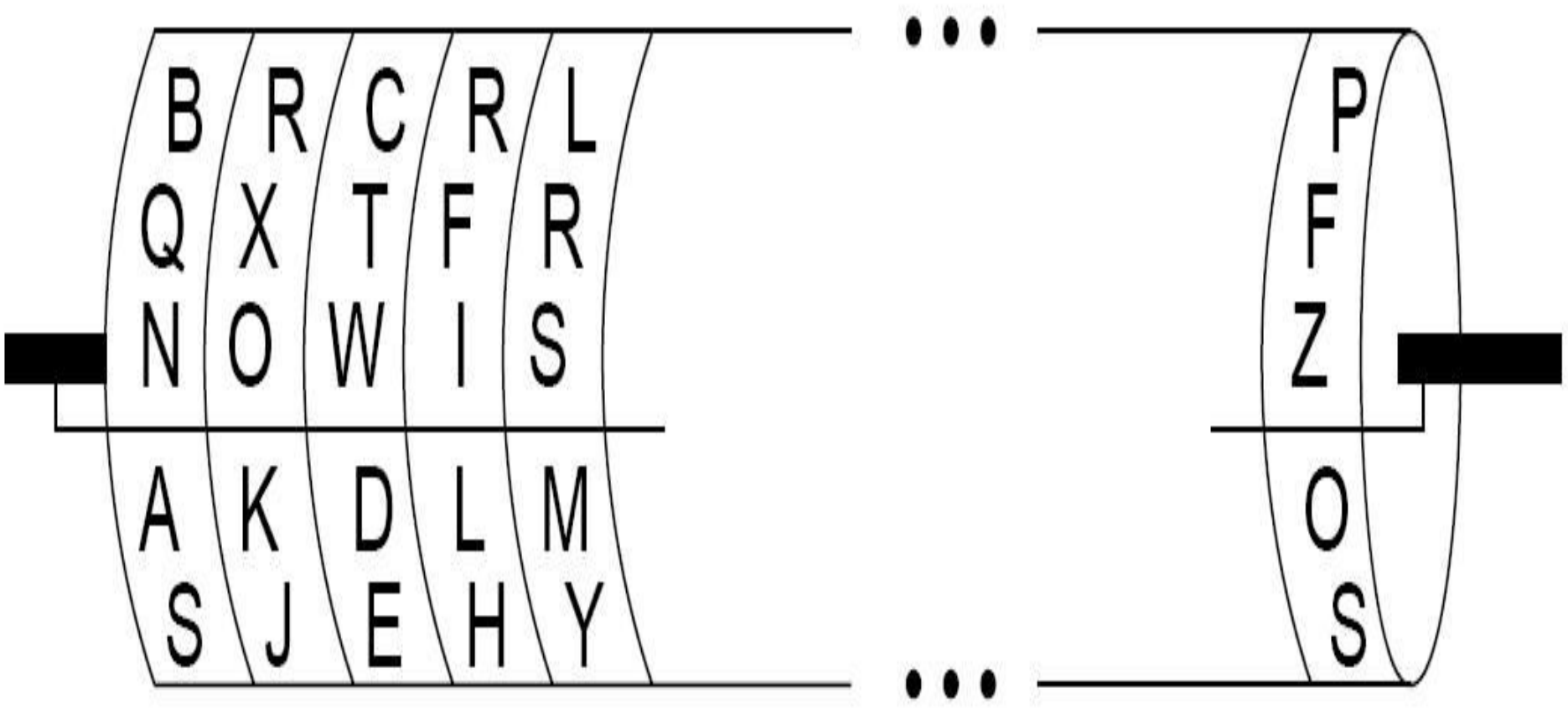
К таким шифрам относятся шифр Виженера, шифры получаемые при использовании немецкой шифровальной машины «Энигма» (буквы заменяются при помощи роторов), цилиндр Джефферсона и др.

При **n**-алфавитной подстановке

- знак $\mathbf{m_1}$ из исходного алфавита **A** заменяется знаком $\mathbf{h_1}$ алфавита **B1**, (знак $\mathbf{m_1 \in A}$, $\mathbf{h_1 \in B1}$)
- $\mathbf{m_2}$ заменяется знаком $\mathbf{h_2}$ из алфавита **B2** и т.д.,
- знак $\mathbf{m_{n+1}}$ снова заменяется символом из алфавита **B1**.

Эффект использования многоалфавитной подстановки состоит в том, что обеспечивается маскировка естественной частотной статистики исходного языка **A**, так как конкретный символ языка **A** может быть преобразован в несколько различных символов шифрованного алгоритма **B**.

Цилиндр Джефферсона для полиалфавитных шифров



Шифр ВИЖЕНЕРА

С изобретением телеграфа в середине 1800х годов интерес к криптографии стал расти, поскольку ненадежность моноалфавитных подстановочных шифров была уже хорошо известна.

Решение, найденное в ту эпоху, заключалось в использовании шифра Виженера, который, как это ни странно, к тому моменту был известен уже на протяжении почти 300 лет.

Этот шифр был известен во Франции, как «нераскрываемый шифр), и это был действительно выдающийся шифр своего времени.

Фактически, шифр Виженера оставался нераскрытым почти три столетия, с момента его изобретения в 1586 г. и до момента его взлома в 1854, когда Чарльз Бэббидж сумел, наконец, раскрыть его.

Шифр ВИЖЕНЕРА

Шифр Виженера представляет собой полиалфавитный (многоалфавитный) подстановочный шифр.

Это означает, что для подстановки используются многие алфавиты, благодаря чему частоты символов в зашифрованном тексте не соответствуют частотам символов в тексте открытом.

Следовательно, в отличие от моноалфавитных подстановочных шифров наподобие шифра Цезаря, шифр Виженера не поддается простому частотному анализу.

В сущности, шифр Виженера меняет соответствие между открытыми и зашифрованными символами для каждого очередного символа.

Он основывается на таблице, вид которой приведен на след. слайде. Каждая строка этой таблицы не что иное, как шифр Цезаря, сдвинутый на число позиций, соответствующее позиции в строке. Строка A сдвинута на 0 позиций, строка B - на 1, и так далее.

Открытый текст

Ключевое слово	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Рис. 2.6. Шифр Виженера

В шифре Виженера такая таблица используется в сочетании с ключевым словом, при помощи которого шифруется текст. Предположим, например, что нам требуется зашифровать фразу
WE NEED MORE SNOW FOR BETTER SKING

при помощи ключа SECURITY.

Для шифрования вы повторяете ключ столько раз, сколько необходимо для достижения длины открытоrо текста, просто записывая символы под символами открытого текста. Затем вы получаете поочередно каждый символ зашифрованноrо текста, беря столбец, определенный по символу открытого текста, и пересекая его со строкой, определенной по соответствующему символу ключа.

Многоалфавитные шифры

Пример 3. Зашифруем сообщение, используя восьмиалфавитный шифр подстановки.

Ключ SECURITY

WE NEED MORE SNOW FOR BETTER SKING

+

SECURITYSECURITYSECURITYSECURITYSE

Будем рассматривать алфавит как кольцо, состоящее из 27 символов (26 букв и пробел). Присваивая, соответственно значения 0 –пробелу, 1 – «A», 2 – «B»,…….26 – «Z», будем иметь восьмиалфавитный шифр подстановки (ключ SECURITY 8 букв).

Мы можем рассматривать первый алфавит как сдвигающий каждый знак, помещенный в кольцо на 19 (=S).

Второй алфавит как сдвигающий каждый знак на 5 (=E) и т.д.

Многоалфавитные шифры

Если мы используем сложение по модулю 27 в качестве средства преобразования секретной информации, получим зашифрованный текст:

W+S = (23+19) mod 27 =42 mod 27 =15 →O

E+E = (5+5) mod 27 =10 mod 27 =10 → J

пробел +C= (0+3) mod 27 =3 → C

N+U = (14 + 21) mod 27 = 35 mod 27 = 8 → H

E+R = (5+18) mod 27 = 23 mod 27 =23 → W

……..

OJCHWNXYETUZRAGMOEIIIIVCLYHLRADGASJ – полученный шифротекст.

Многоалфавитные шифры

Для расшифрования используется тот же ключ, только операция сложения заменена на вычитание:

15 – 19 = - 4, если значение меньше 0, то прибавляем 27: -4+27 = 23 →W

10 - 5 = 5 → E

3 – 3 = 0 → пробел

8 – 21 = -13 +27= 14 → N

……

Многоалфавитные шифры

Пример 4. В симметричных шифрах в качестве шифрующего преобразования очень часто применяется операция – сложение по модулю 2 (⊕).

0 ⊕ 0 = 0 0 ⊕1 =1 1 ⊕ 0 = 1 1 ⊕ 1 = 0

С помощью сложения по модулю 2 можно выполнить многоалфавитную замену, прибавляя к битам ключа соответствующие биты открытого текста.

Заменяя символы текста цифровым двоичным эквивалентом и складывая их с двоичными символами некоторой специальной последовательности (ключа), называемой гаммой, получаем шифротекст

К Р О Н А

0001I 0010 I 0011 I 0100 I 0101

⊕

КЛЮЧ

1001

Многоалфавитные шифры

0001 0010 0011 0100 0101 исходный текст

⊕

1001 1001 1001 1001 1001

1000 1011 1010 1101 1100 зашифрованный текст

Поскольку двойное прибавление одной и той величины по модулю 2 восстанавливает исходное значение, шифрование и расшифрование выполняется одной и той же программой. Выполним обратное преобразование:

1000 1011 1010 1101 1100 зашифрованный текст

⊕

1001 1001 1001 1001 1001

0001 0010 0011 0100 0101 исходный текст

Многоалфавитные шифры

К сожалению, данный алгоритм обладает очень слабой стойкостью, тем не менее АНБ (Агентство национальной безопасности США) одобрило этот код для использования в мобильных телефонах американских производителей для засекречивания речевых переговоров. Данный шифр часто встречается в различных коммерческих программных продуктах.

Опытными криптоаналитиками взлом этого шифра производится следующим образом:

1. **Определяется длина ключа:**

- **шифротекст последовательно складывается по модулю 2 со своей копией, сдвинутой на различное число бит**
- **В полученном векторе подсчитывается число совпадающих компонент.**
- **Когда величина сдвига кратна длине ключа, то число совпадений превысит 6% от общей длины исследуемого шифротекста.**
- **Если величина сдвига не кратна длине ключа, то совпадений будет меньше (0,4%). Проанализировав полученные данные можно сделать выводы о длине ключа.**

2. **Затем складывается шифротекст по модулю 2 со своей копией, сдвинутой на величину длины ключа. Эта операция аннулирует ключ и оставит в наличии открытый текст.**

Составные шифры

По мнению Клода Шеннона, для получения стойких блочных шифров необходимо использовать два общих принципа: рассеивание и перемешивание.

- ***Рассеивание*** представляет собой распространение влияния одного знака открытого текста на много знаков шифротекста, что позволяет скрыть статистические свойства открытого текста. Развитием этого принципа является распространение влияния одного символа ключа на много символов шифрограммы, что позволяет исключить восстановление ключа по частям..
- ***Перемешивание*** предполагает использование таких шифрующих преобразований, которые усложняют восстановление взаимосвязи статистических свойств открытого и шифрованного текстов. Однако шифр должен не только затруднять раскрытие, но и обеспечивать легкость зашифрования и расшифрования при известном пользователю секретном ключе...

Составные шифры

Распространенный способ шифрования, при котором достигается хорошее рассеивание и перемешивание, состоит в использовании составного шифра, который реализован в виде последовательности простых шифров. При перестановке перемешивают символы открытого текста, при подстановке символ открытого текста заменяют другим символом из того же алфавита.

- В современных блочных шифрах блоки открытого и шифротекста представляют собой двоичные последовательности длиной 64 бита. Каждый блок может принимать 2^64 значений.
- При многократном чередовании простых перестановок и подстановок можно получить стойкий шифр с хорошим рассеиванием и перемешиванием.
- Одним из наглядных примеров криптоалгоритма, разработанного в соответствии с принципами рассеивания и перемешивания, может служить принятый в 1977 году национальным бюро стандартов США (АНБ) стандарт шифрования данных DES.

Составные шифры

DES предназначен для защиты от несанкционированного доступа к важной, но не секретной информации (коммерческие фирмы, электронные платежи).

Наиболее широко DES используется для шифрования при передаче данных между различными системами, почтовой связи, в электронных системах платежей и при электронном обмене коммерческой информацией. Первоначально методы шифрования лежащие в основе DES разработала для своих целей фирма IBM, и реализовало в системе «Люцифер».

Стандарт DES осуществляет шифрование 64 битовых блоков с помощью 64 битового ключа, в котором значащими являются 56 бит, используемыми непосредственно для алгоритма шифрования и 8 бит – для обнаружения ошибок.

Расшифрование в DES выполняется путем повторения операции шифрования в обратной последовательности.

Составные шифры

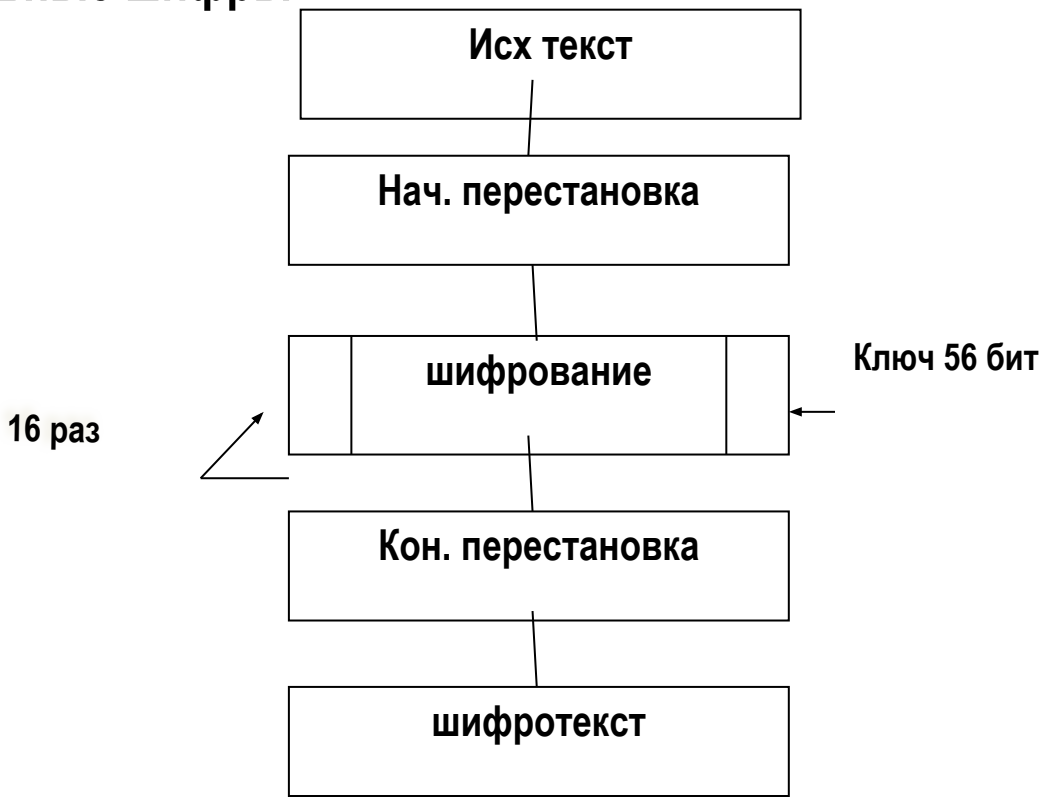


Рисунок 4 Обобщенный алгоритм шифрования DES

Составные шифры

Алгоритм DES предусматривает наличие у абонентов секретных ключей и соответственно налаженную систему генерации ключей, выпуска и распределения ключевой документации.

Криптоаналитик Циммерман считатал основными достоинствами алгоритма DES:

- использование только одного ключа длиной 56 бит;
- зашифровав сообщение с помощью одного пакета программ для его расшифровки можно использовать любой другой пакет, реализующий алгоритм DES;
- относительная простота алгоритма обеспечивает высокую скорость обработки;
- достаточно высокая стойкость алгоритма.

Составные шифры

Основные режимы работы алгоритма DES:

- **1. *Электронная кодовая книга (ECB*).** Файл разбивают на 64 битовые блоки. Каждый из этих блоков шифруют независимым образом и с использованием одного и того же ключа шифрования. Основное достоинство – простота. Недостаток – слабая устойчивость против квалифицированных криптоаналитиков.
- **2. *Сцепление блоков шифра (CBC).*** Исходный файл M, разбивается на 64 битовые блоки M1, M2, …, Mn. Первый блок M1 складывается по модулю 2 с 64 битовым начальным вектором. Начальный вектор изменяется ежедневно и держится в секрете. Полученная сумма затем шифруется с использованием ключа DES отправителя и получателя.
- **3. *Обратная связь по шифртексту (CFB).*** Размер блока может отличаться от 64 бит. Файл, подлежащий шифрованию, считывается последовательными блоками длиной **k** битов (k=1 до 64).
- **4.*Внешняя обратная связь (OFB***). Используется переменный размер блока и сдвиговый регистр, используемый также как и в режиме CFB (3). Отличие от CFB(3) состоит в методе обновления сдвигового регистра. Это осуществляется путем отбрасывания старших **k** битов и дописывания справа **Pi** , где **Pi** – старшие **k** битов операции DES **(Ci-1).**

Составные шифры

При принятии DES в качестве стандарта планировался его пересмотр каждые 5 лет. Поэтому данный стандарт пересматривался в 1983, 1988, 1993 и 1999 гг. Последний пересмотр происходил уже после старта конкурса AES, целью которого был выбор нового стандарта шифрования США вместо DES .

В 1999 г. уже невозможно было использовать DES для серьезной защиты из-за его короткого ключа, к этому времени уже были прецеденты вскрытия DES полным перебором ключа с использованием распределенных вычислений. Поэтому текущая версия стандарта устанавливает в качестве стандарта шифрования Triple DES, а собственно DES разрешалось использовать только в целях совместимости с действующими DES-шифраторами, причем везде, где это возможно, рекомендовалось осуществить переход на Triple DES.

Составные шифры

Advanced Encryption Standard (**AES**), также известный как **Rijndael** (произносится [rɛinda:l]) — симметричный алгоритм блочного шифрования (размер блока 128 бит, ключ 128/192/256 бит), принятый в качестве стандарта шифрования правительством США по результатам конкурса AES. Этот алгоритм хорошо проанализирован и сейчас широко используется, как это было с его предшественником DES.

Национальный институт стандартов и технологий США (англ. *National Institute of Standards and Technology*, NIST) опубликовал спецификацию AES 26 ноября 2001 года после пятилетнего периода, в ходе которого были созданы и оценены 15 кандидатур. 26 мая 2002 года AES был объявлен стандартом шифрования. По состоянию на 2009 год AES является одним из самых распространённых алгоритмов симметричного шифрования

Составные шифры

IDEA – международный алгоритм шифрования , запатентован швейцарской фирмой Ascom, применяется в общедоступном пакете конфиденциальной электронной почты PGP.

- Исходные блоки текста делятся на 4 группы по 16 бит.
- В IDEA применяется 52 субключа по 16 бит каждый.
- Субключи IDEA генерируются следующим образом: 128-битовый ключ IDEA определяет первые восемь ключей (128=16x8).
- Последующие 44 ключа определяются путем последовательности циклических сдвигов этого кода на 25 двоичных разрядов влево.

Стандарт шифрования Российской Федерации ГОСТ 28147-89

Описание стандарта шифрования Российской Федерации содержится в документе «Алгоритм криптографического преобразования данных ГОСТ 28147-89».

Помимо нескольких тесно связанных между собой процедур шифрования, в документе описан один, построенный на общих принципах с ними, алгоритм выработки *имитовставки*.

Имитовставка является криптографической контрольной комбинацией, то есть кодом, вырабатываемым из исходных данных с использованием секретного ключа с целью защиты данных от внесения в них несанкционированных изменений.

На различных шагах алгоритмов ГОСТа данные, которыми они оперируют, интерпретируются и используются различным образом:

- как массивы независимых битов,
- как целое число без знака,
- как имеющий структуру сложный элемент, состоящий из нескольких более простых элементов.

Элементы данных обозначаются заглавными латинскими буквами с наклонным начертанием (например, X).

Через $|X|$ обозначается размер элемента данных X в битах.

Таким образом, если интерпретировать элемент данных ***X*** как целое неотрицательное число, можно записать следующее неравенство:

$$0 \le X < 2^{|X|}.$$

Если элемент данных состоит из нескольких элементов меньшего размера, то этот факт обозначается следующим образом:

$$X = (X0, X1, \ldots, Xn-1) = X0 \,||\, X1 \,||\ldots||\, Xn-1.$$

Процедура объединения нескольких элементов данных в один называется ***конкатенацией*** данных и обозначается символом «||». Естественно, для размеров элементов данных должно выполняться следующее соотношение: $|X| = |X0| + |X1| + \ldots + |Xn\text{-}1|$.

Документ, задающий ГОСТ 28147–89, содержит описание алгоритмов нескольких уровней. На самом верхнем находятся практические алгоритмы, предназначенные для шифрования массивов данных и выработки для них имитовставки.

Все они опираются на три алгоритма низшего уровня, называемые в тексте ГОСТа ***циклами***. Эти фундаментальные алгоритмы упоминаются в дальнейшем как ***базовые циклы***, чтобы отличать их от всех прочих циклов:

- **цикл зашифрования (32-З);**
- **цикл расшифрования (32-Р);**
- **цикл выработки имитовставки (16-З).**

В свою очередь, каждый из базовых циклов представляет собой многократное повторение одной единственной процедуры, называемой для определенности далее ***основным шагом криптопреобразования***.

Таким образом, чтобы разобраться в ГОСТе, надо понять три следующие вещи:

- а) что такое ***основной шаг*** криптопреобразования;
- б) как из ***основных шагов*** складываются ***базовые циклы***;
- в) как из трех ***базовых циклов*** складываются все практические алгоритмы ГОСТа.

В соответствии с принципом Кирхгофа, которому удовлетворяют все современные известные широкой общественности шифры, именно секретность ключевой информации обеспечивает секретность зашифрованного сообщения.

Ключевая информация

В ГОСТе ключевая информация состоит из двух структур данных:

1. Ключ является массивом из восьми 32-битовых элементов кода, далее он обозначается символом ***K*: *K* ={*Ki*}** **$0 \le i \le 7$.**

В ГОСТе элементы ключа используются как 32-разрядные целые числа без знака: **$0 \le Ki \le 2^{32}$.**

Размер ключа составляет **32*·8 = 256 бит или 32 байта.**

2. *Таблица замен* может быть представлена в виде матрицы размера 8 x16, содержащей 4-битовые элементы, которые можно представить в виде целых чисел от 0 до 15.

Строки *таблицы замен* называются *узлами замен*, они должны содержать различные значения, то есть каждый *узел замен* должен содержать 16 различных чисел от 0 до 15 в произвольном порядке.

Таблица замен обозначается символом ***H*: H = {Hi,j}**

$$0 \le i \le 7, \qquad 0 \le j \le 15$$

$$0 \le Hi,j \le 15.$$

Таким образом, общий объем таблицы замен равен:

8 узлов x 16 элементов/узел x 4 бита/элемент = 512 бит или 64 байта.

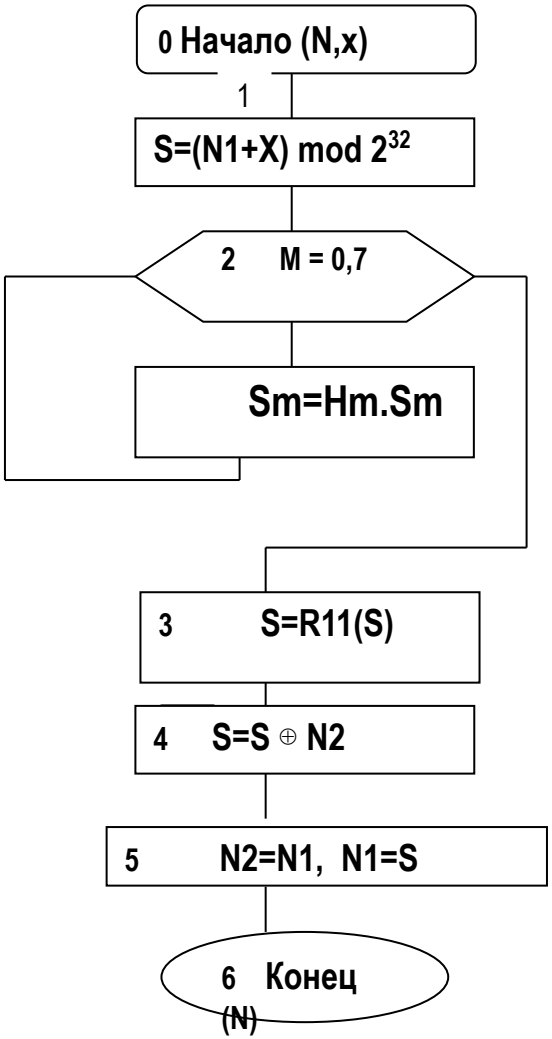


рисунок 5 Схема основного шага криптопреобразования алгоритма ГОСТ 28147-89.

Основной шаг криптопреобразования является оператором, определяющим преобразование 64-битового блока данных. Дополнительным параметром этого оператора является 32-битовый блок, в качестве которого используется какой-либо элемент ключа.

Схема алгоритма основного шага приведена на рисунке 5. Ниже даны пояснения к алгоритму основного шага:

- **Шаг 0.** Определяет исходные данные для основного шага криптопреобразования: *N* – преобразуемый 64-битовый блок данных, в ходе выполнения шага его младшая (*N*1) и старшая (*N*2) части обрабатываются как отдельные 32-битовые целые числа без знака. Таким образом, можно записать *N*=(*N*1,*N*2) ;
- ***X*** – 32-битовый элемент ключа;
- **Шаг 1.** Сложение с ключом. Младшая половина преобразуемого блока складывается по модулю 2^32 с используемым на шаге элементом ключа, результат передается на следующий шаг;

- **Шаг 2.** Поблочная замена. 32-битовое значение, полученное на предыдущем шаге, интерпретируется как массив из восьми
- 4-битовых блоков кода: $S = (S0, S1, S2, S3, S4, S5, S6, S7)$.
- Далее значение каждого из восьми блоков заменяется новым, которое выбирается по таблице замен следующим образом: значение блока *Si* меняется на *Si*-тый по порядку элемент (нумерация с нуля) *i*-того узла замен (т.е. *i*-той строки таблицы замен, нумерация также с нуля). В качестве замены для значения блока выбирается элемент из таблицы замен с номером строки, равным номеру заменяемого блока, и номером столбца, равным значению заменяемого блока как 4-битового целого неотрицательного числа.
- **Шаг 3.** Циклический сдвиг на 11 бит влево. Результат предыдущего шага сдвигается циклически на 11 бит в сторону старших разрядов и передается на следующий шаг.
- **Шаг 4.** Побитовое сложение: значение, полученное на шаге 3, побитно складывается по модулю 2 со старшей половиной преобразуемого блока.
- **Шаг 5.** Сдвиг по цепочке: младшая часть преобразуемого блока сдвигается на место старшей, а на ее место помещается результат выполнения предыдущего шага.
- **Шаг 6.** Полученное значение преобразуемого блока возвращается как результат выполнения алгоритма основного шага криптопреобразования.

Базовые циклы криптографических преобразований.

Алгоритмы зашифрования, расшифрования и «учета» в контрольной комбинации одного блока называются *базовыми циклами* ГОСТа, что подчеркивает их фундаментальное значение для построения этого шифра.

Базовые циклы построены из *основных шагов* криптографического преобразования. В процессе выполнения основного шага используется только один элемент ключа, в то время как ключ ГОСТ содержит восемь таких элементов. Базовые циклы заключаются в многократном выполнении *основного шага* с использованием разных элементов ключа и отличаются друг от друга только числом повторения шага и порядком использования ключевых элементов. Ниже приведен этот порядок для различных циклов.

1. Цикл зашифрования **32-З**:

K0,***K***1,***K***2,***K***3,***K***4,***K***5,***K***6,***K***7,***K***0,***K***1,***K***2,***K***3,***K***4,***K***5,***K***6,***K***7,***K***0,***K***1,***K***2,***K***3, ***K***4, ***K***5,***K***6,***K***7,***K***7,***K***6,***K***5,***K***4,***K***3,***K***2,***K***1,***K***0.

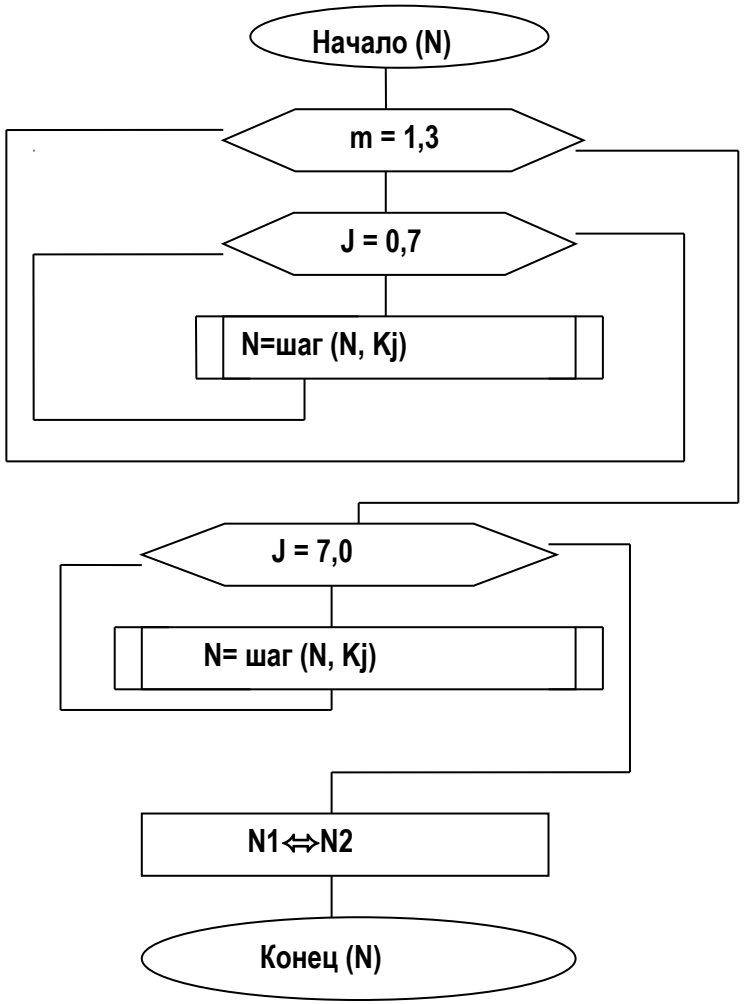


Рис. 6. Схема цикла зашифрования 32-З.

2. Цикл расшифрования 32-Р:

***K*0,*K*1,*K*2,*K*3,*K*4,*K*5,*K*6,*K*7,*K*7,*K*6,*K*5,*K*4,*K*3,*K*2,*K*1,*K*0,*K*7,*K*6,*K*5,*K*4,*K*3,*K*2,*K*1,*K*0,*K*7,*K*6,*K*5,*K*4,*K*3,*K*2,*K*1,*K*0.**

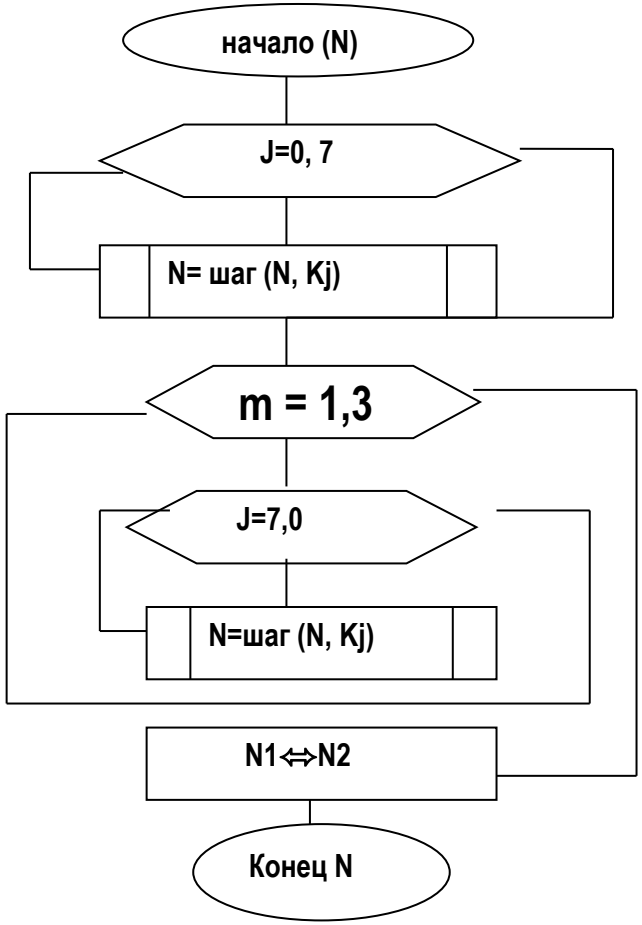


Рис. 7 Схема цикла расшифрования 32-Р.

3 Цикл выработки имитовставки 16-3:

***K*0,*K*1,*K*2,*K*3,*K*4,*K*5,*K*6,*K*7,*K*0,*K*1,*K*2,*K*3,*K*4,*K*5,*K*6,*K*7.**

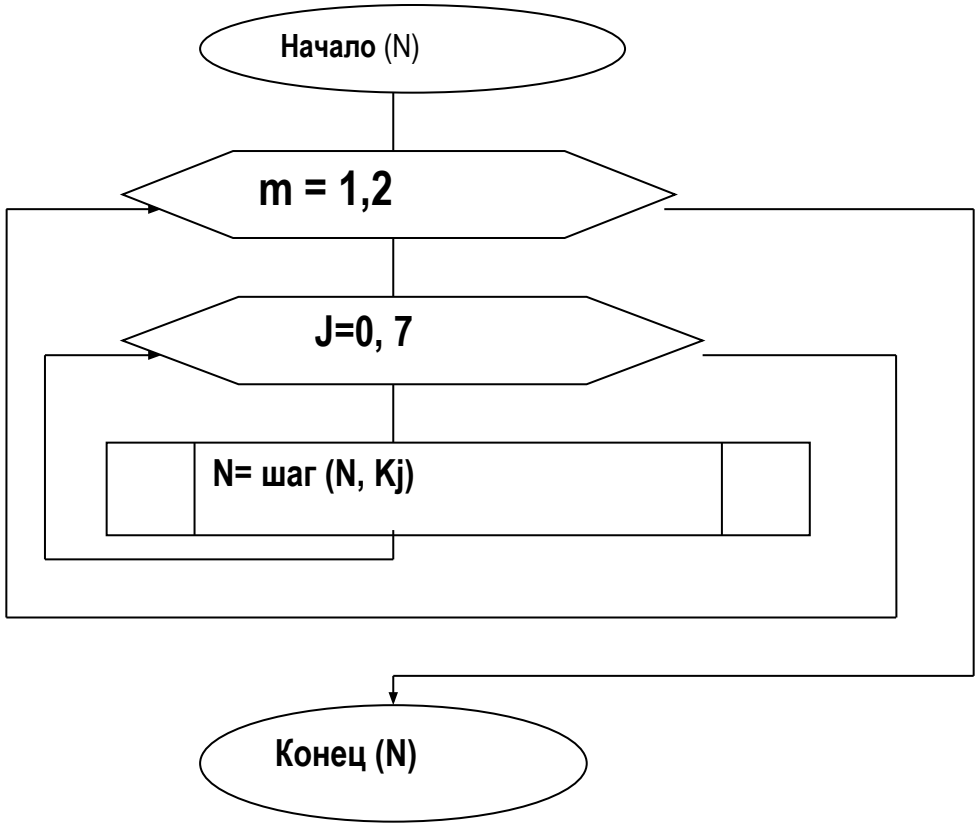


рис. 8 Схема цикла выработки имитовставки

Стандарт шифрования Российской Федерации ГОСТ 28147-89

- Цикл расшифрования должен быть обратным циклу зашифрования, то есть последовательное применение этих двух циклов к произвольному блоку должно дать в итоге исходный блок. Для выполнения этого условия для алгоритмов, подобных ГОСТу, необходимо и достаточно, чтобы порядок использования ключевых элементов соответствующими циклами был взаимно обратным.
- Каждый из базовых циклов принимает в качестве аргумента и возвращает в качестве результата 64-битовый блок данных, обозначенный на схемах ***N***.
- Символ Шаг(***N***,***X***) обозначает выполнение основного шага криптопреобразования для блока ***N*** с использованием ключевого элемента ***X***.
- Цикл выработки имитовставки вдвое короче циклов шифрования, порядок использования ключевых элементов в нем такой же, как в первых 16 шагах цикла зашифрования, поэтому этот порядок в обозначении цикла кодируется той же самой буквой «З».

Основные режимы шифрования.

ГОСТ 28147-89 предусматривает три следующих режима шифрования данных:

- простая замена,
- гаммирование,
- гаммирование с обратной связью,
- дополнительный режим выработки имитовставки.

В любом из этих режимов данные обрабатываются блоками по 64 бита, на которые разбивается массив, подвергаемый криптографическому преобразованию, именно поэтому ГОСТ относится к блочным шифрам. Однако в двух режимах гаммирования есть возможность обработки неполного блока данных размером меньше 8 байт, что существенно при шифровании массивов данных с произвольным размером, который может быть не кратным 8 байтам.

Простая замена.

Зашифрование в данном режиме заключается в применении цикла **32-З** к блокам открытых данных, расшифрование – цикла **32-Р** к блокам зашифрованных данных., 64-битовые блоки данных обрабатываются в нем независимо друг от друга.

Режим шифрования простой заменой имеет следующие особенности:

- 1. Так как блоки данных шифруются независимо друг от друга и от их позиции в массиве, при зашифровании двух одинаковых блоков открытого текста получаются одинаковые блоки шифротекста и наоборот. Отмеченное свойство позволит криптоаналитику сделать заключение о тождественности блоков исходных данных, если в массиве зашифрованных данных ему встретились идентичные блоки, что является недопустимым для серьезного шифра.

- 2. Если длина шифруемого массива данных не кратна 8 байтам или 64 битам, возникает проблема, чем и как дополнять последний неполный блок данных массива до полных 64 бит.

На первый взгляд, перечисленные выше особенности делают практически невозможным использование режима простой замены, ведь он может применяться только для шифрования массивов данных с размером кратным 64 битам, не содержащим повторяющихся 64-битовых блоков.

Напомним, что размер ключа составляет 32 байта, а размер таблицы замен – 64 байта.

ГОСТ предписывает использовать режим простой замены исключительно для шифрования ключевых данных.

2. Гаммирование.

Гаммирование – это наложение на открытые (зашифрованные - для гаммирования с обратной связью) данные криптографической гаммы, то есть последовательности элементов данных, вырабатываемых с помощью некоторого криптографического алгоритма, для получения зашифрованных (открытых) данных. Для наложения гаммы при зашифровании и ее снятия при расшифровании должны использоваться взаимно обратные бинарные операции, например, сложение и вычитание по модулю 2^64 для 64-битовых блоков данных.

В ГОСТе для этой цели используется операция побитного сложения по модулю 2, поскольку она является обратной самой себе и, к тому же, наиболее просто реализуется аппаратно.

Гаммирование решает обе упомянутые проблемы:

1) все элементы гаммы различны для реальных шифруемых массивов и, следовательно, результат зашифрования даже двух одинаковых блоков в одном массиве данных будет различным.

2) хотя элементы гаммы и вырабатываются одинаковыми порциями в 64 бита, использоваться может и часть такого блока с размером, равным размеру шифруемого блока.

Требования к качеству ключевой информации и источники ключей.

Не все ключи и таблицы замен обеспечивают максимальную стойкость шифра. Для каждого алгоритма шифрования существуют свои критерии оценки ключевой информации. Так, для алгоритма DES известно существование так называемых «***слабых ключей***», при использовании которых связь между открытыми и зашифрованными данными не маскируется достаточным образом, и шифр сравнительно просто вскрывается.

Очевидно, нулевой ключ и тривиальная таблица замен, по которой любое значение заменяется но него самого, являются слабыми, при использовании хотя бы одного из них шифр достаточно просто взламывается, каков бы ни был второй ключевой элемент.

1. ***Ключ*** должен являться массивом статистически независимых битов, принимающих с равной вероятностью значения 0 и 1. Ключи, выработанные с помощью некоторого датчика истинно случайных чисел, будут качественными с вероятностью, отличающейся от единицы на ничтожно малую величину. Если же ключи вырабатываются с помощью генератора псевдослучайных чисел, то используемый генератор должен обеспечивать указанные выше статистические характеристики, и, кроме того, обладать высокой криптостойкостью, не меньшей,

2. ***Таблица замен*** является долговременным ключевым элементом, то есть действует в течение гораздо более длительного срока, чем отдельный ключ. Предполагается, что она является общей для всех узлов шифрования в рамках одной системы криптографической защиты. Даже при нарушении конфиденциальности таблицы замен стойкость шифра остается чрезвычайно высокой и не снижается ниже допустимого предела.

К качеству отдельных узлов замен можно предъявить приведенное ниже требование.

Каждый узел замен может быть описан четверкой логических функций, каждая из которых имеет четыре логических аргумента. Необходимо, чтобы эти функции были достаточно сложными. На практике бывает достаточно получить узлы замен как независимые случайные перестановки чисел от 0 до 15, это может быть практически реализовано, например, с помощью перемешивания колоды из шестнадцати карт, за каждой из которых закреплено одно из значений указанного диапазона.

Для обратимости циклов шифрования «32-З» и «32-Р» не требуется, чтобы узлы замен были перестановками чисел от 0 до 15. Все работает даже в том случае, если в узле замен есть повторяющиеся элементы, и замена, определяемая таким узлом, необратима, однако в этом случае снижается стойкость шифра.

Если вы разрабатываете программы, использующие криптографические алгоритмы, вам необходимо позаботиться об утилитах, вырабатывающих ключевую информацию, а для таких утилит необходим источник случайных чисел (СЧ) высокого статистического качества и криптостойкости.

Наилучшим подходом здесь было бы использование аппаратных датчиков СЧ, однако это не всегда приемлемо по экономическим соображениям. В качестве разумной альтернативы возможно (и очень широко распространено) использование различных программных датчиков СЧ. При генерации небольшого по объему массива ключевой информации широко применяется метод «электронной рулетки», когда очередная получаемая с такого датчика порция случайных битов зависит от момента времени нажатия оператором некоторой клавиши на клавиатуре компьютера.

Поточные шифры в отличие от блочных осуществляют поэлементное шифрование потока данных без задержки в криптосистеме. В общем случае каждый символ (бит)открытого текста шифруется, передается и расшифровывается независимо от других символов. Шифрующее преобразование элемента открытого текста меняется от одного элемента к другому, в то время как для блочных шифров оно остается неизменным.

Достоинства:

1. Высокая скорость преобразования данных (практически в реальном масштабе).
2. Высокая криптостойкость, так как вскрытие такой системы предлагает точное определение структуры генератора ключевой последовательности (ГКП) и его начальной фазы.

Поточные шифры основываются на использовании ключевой последовательности с заданными свойствами случайности и двоичным представлением информационных сообщений.

Шифрование и расшифрование осуществляется с использованием операции сложения по модулю 2 (открытого) исходного текста и псевдослучайной ключевой последовательности. Ключ состоит из сгенерированной определенным образом последовательности символов с заданными свойствами случайности (непредсказуемости) получения определенного символа

Шифр Вернама

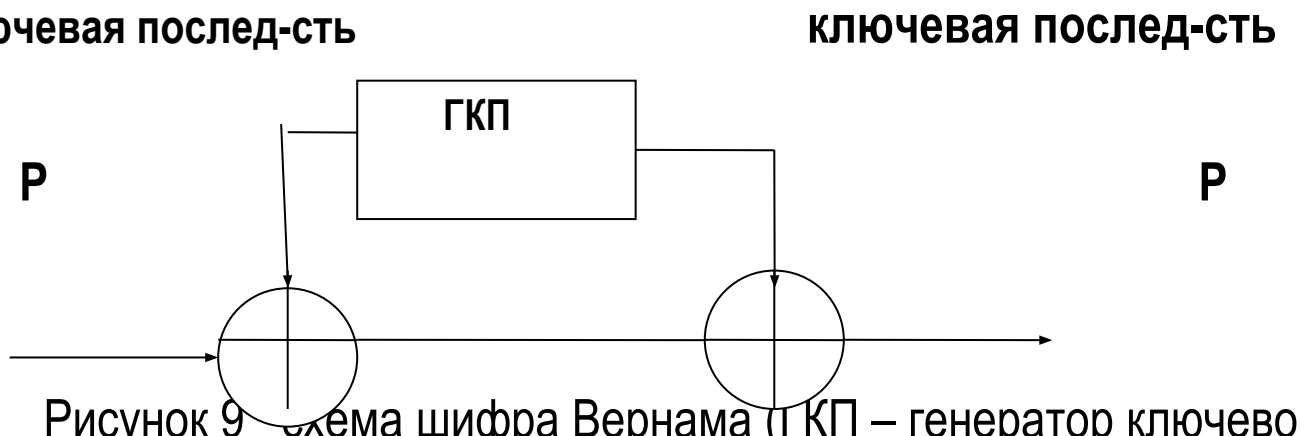


Рисунок 9 – Схема шифра Вернама (ГКП – генератор ключевой последовательности)

В шифре Вернама длина ключевой последовательности равна длине открытого текста. Недостаток – неудобно хранить сверхдлинные ключевые последовательности.

Синхронные поточные шифры

В синхронных поточных шифрах КП (гамма) формируется ГКП (генератором псевдослучайной последовательности) независимо от последовательности символов открытого текста и каждый символ шифруется независимо от других символов. Ключом является начальная установка **K** генератора псевдослучайной последовательности (ПСП).

В общем случае:

- **Yi= E (Xi, Fi (k))** – шифрование,
- **Xi=D(Yi , Fi (k))** – расшифрование,

где

- **E** – функция шифрования;
- **D** – функция расшифрования;
- **Xi** – двоичный символ открытого текста;
- **Yi** – двоичный символ зашифрованного текста
- **Fi(k)** – **i**-ый символ ПСП , выработанные генератором с функцией обратной связи **F** и начальным состоянием **k**.

Классификация синхронных поточных шифров:

- *по способам построения*
  - комбинирование ПСП;
  - метод функциональных отображений;
- *по соотношению размера открытого текста и периода ПСП*
  - с конечной ПСП;
  - с бесконечной ПСП – период ПСП больше размера текста;
- по способам технической реализации генератора ПСП
  - с нелинейной внешней логикой;
  - с нелинейной внутренней логикой.

Синхронный поточный шифр может быть реализован в виде блочного шифра, работающего в режиме обратной связи по выходу OFB (пример реализации алгоритм RC4).

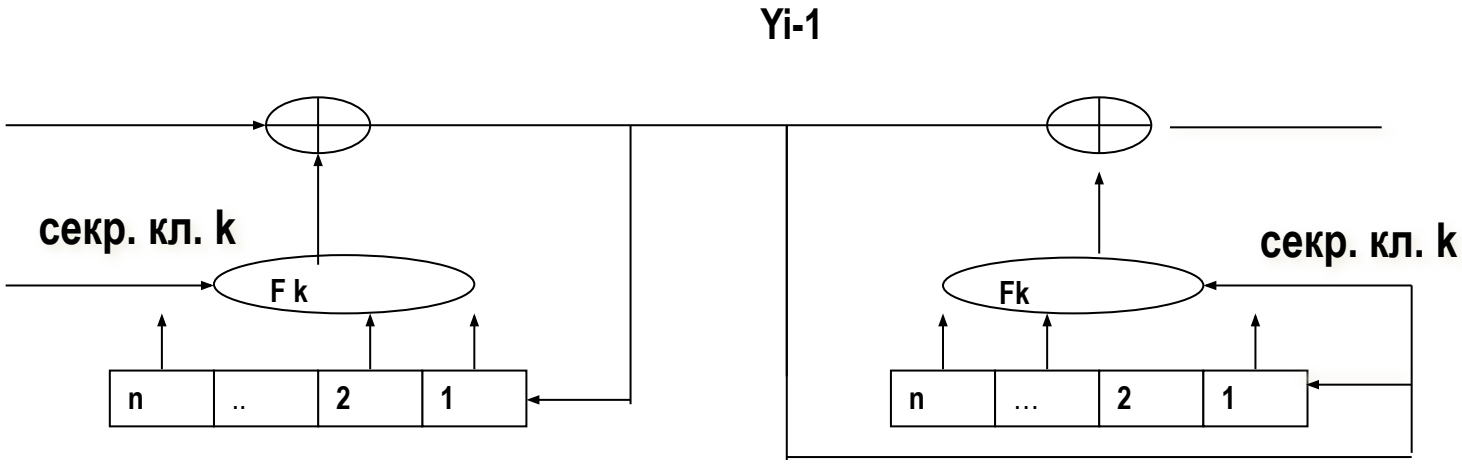


Рисунок 10 Схема поточного самосинхронизирующегося шифра

Символы открытого текста шифруются с учетом ограниченного числа предшествующих **n** символов шифротекста. При этом секретным ключом **k** является функция обратной связи генератора ПСП.

- Yi=E(Xi , Fk(Yi-1,Yi-2,…Yi-n)) – шифрование,
- Xi=D(Yi , Fk(Yi-1,Yi-2,…Yi-n)) - расшифрование

Самосинхронизирующийся поточный шифр может быть реализован в виде блочного шифра, работающего в режиме обратной связи по шифротексту CFB.

Системы поточного шифрования близки по своим параметрам к криптосистемам с одноразовым блокнотом для шифрования, в которых размер ключа равен размеру открытого текста.

Хотя подавляющее большинство существующих шифров с секретным ключом с определенностью могут быть отнесены или к поточным или к блочным шифрам, теоретически граница между этими классами остается довольно размытой. Так, например, допускается использование алгоритмов блочного шифрования в режиме поточного шифрования (например, режимы CBF и OFB для алгоритма DES или режим гаммирования для алгоритма ГОСТ 28147-89).

Поточные шифры почти всегда работают быстрее и обычно требуют для своей реализации гораздо меньше программного кода, чем блочные шифры. Наиболее известный поточный шифр был разработан Р. Ривестом; это шифр RC4, который характеризуется переменным размером ключа и байт-ориентированными операциями. На один байт

требуется от 8 до 16 действий, программная реализация шифра выполняется очень быстро. Независимые аналитики исследовали шифр, и он считается защищенным. RC4 используется для шифрования файлов в таких изделиях, как RSA SecurPC. Он также применяется для защиты коммуникаций, например, для шифрования потока данных в Интернет-соединениях, использующих протокол SSL.

Алгоритмы поточного шифрования

Поточное шифрование является наиболее перспективным.

Примеры поточных шифраторов: SEC – 15, SEC – 17, SDE – 100, скорость шифрования от 256 бит/сек до 2304 кбит/сек, ключ состоит из 72 шестнадцатиричных цифр.

Комбинированные шифры

В таких шифрах реализуются принципы как блочных, так и поточных шифров. Примерами комбинированных шифров являются шифры ГОСТ 28147-89 и DES.