Н. А. Молдовян, А. А. Молдовян, М. А. Еремеев

Криптография

От примитивов к синтезу алгоритмов

НАУЧНОЕ ИЗДАНИЕ

bhv®

Н.А. Молдовян, А.А. Молдовян, М.А. Еремеев

Криптография

От примитивов к синтезу алгоритмов

Санкт-Петербург
«БХВ-Петербург»
2004

УДК 681.3
ББК 32.81
М75

***Рецензенты*:**
доктор технических наук, профессор В.И. Коржик
доктор технических наук, профессор А.М. Чуднов

Молдовян Н. А., Молдовян А. А., Еремеев М. А.
М75 Криптография: от примитивов к синтезу алгоритмов. — СПб.: БХВ-Петербург, 2004. — 448 с.: ил.

ISBN 5-94157-524-6

В книге приводятся элементы математических основ криптографии. Раскрывается содержание симметричных и асимметричных шифров, систем цифровой электронной подписи и хэш-функций и основные требования к ним. Излагаются новые результаты в направлении проектирования скоростных шифров на основе управляемых преобразований.

Представлена классификация управляемых примитивов, на основе которых синтезируются новые классы операций, зависящих от преобразуемых данных. Анализируются основные свойства управляемых примитивов. Дается описание ряда новых криптографических примитивов и алгоритмов с оценкой их стойкости к дифференциальному, линейному и другим методам криптоанализа.

Для специалистов в области безопасности информации, криптографии, прикладной математики, информатики и электроники, а также для преподавателей, студентов и аспирантов инженерно-технических вузов.

УДК 681.3
ББК 32.81

Группа подготовки издания:

Главный редактор	*Екатерина Кондукова*
Зам. главного редактора	*Игорь Шишигин*
Зав. редакцией	*Григорий Добин*
Компьютерная верстка	*Сергея Матвеева*
Корректор	*Евгений Камский*
Дизайн обложки	*Игоря Цырульникова*
Зав. производством	*Николай Тверских*

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 31.05.04.
Формат 70×100$^{1}/_{16}$. Печать офсетная. Усл. печ. л. 36,12.
Тираж 2000 экз. Заказ №
"БХВ-Петербург", 190005, Санкт-Петербург, Измайловский пр., 29.

Гигиеническое заключение на продукцию, товар № 77.99.02.953.Д.001537.03.02 от 13.03.2002 г. выдано Департаментом ГСЭН Минздрава России.

Отпечатано с готовых диапозитивов
в ГУП "Типография "Наука"
199034, Санкт-Петербург, 9 линия, 12.

ISBN 5-94157-524-6

Содержание

Введение **7**

Глоссарий **10**

Обозначения **17**

Сокращения **19**

Глава 1. Вопросы одноключевой криптографии **21**

1.1. Варианты реализации шифров 21
1.2. Повышение стойкости шифрования при ограничении длины секретного ключа. 24
1.3. Криптосистемы с гибким алгоритмом и требования к алгоритму предвычислений 27
1.4. Управляемые операции как криптографический примитив 28
1.5. Аппаратная реализация шифров на основе битовых перестановок, зависящих от преобразуемых данных 30
1.6. Особенности проектирования блочных шифров на основе управляемых операций 33
1.6.1. Управляемые операции и отображения 33
1.6.2. Расписание использования ключа 34
1.6.3. Варианты криптосхем 37
1.6.4. Этапы проектирования шифров 41
1.7. Класс алгоритмов с выборкой подключей, зависящей от преобразуемых данных 43
1.7.1. Формальное описание шифрующих процедур и свойство равномерности выборки 44
1.7.2. Алгоритмическая реализация 48
1.7.3. Гибкие шифры с доказуемой неэквивалентностью всех модификаций криптоалгоритма 50

Глава 2. Математические основы криптографии **61**

2.1. Введение в конечные поля 61
2.2. Элементы теории чисел 66
2.2.1. Некоторые определения и утверждения 66

2.2.2. Функция Эйлера....69
2.2.3. Алгоритм Евклида....71
2.2.4. Расширенный алгоритм Евклида....72
2.2.5. Показатели и первообразные корни....73
2.3. Булевы функции и свойства криптографических примитивов....77
2.3.1. Преобразование Уолша-Адамара....78
2.3.2. Сбалансированность БФ....80
2.3.3. Корреляционные свойства БФ....82
2.3.4. Критерии распространения изменений для БФ....84
2.3.5. Исследование нелинейности БФ....86
2.3.6. БФ, достигающие максимальной нелинейности....89
2.3.7. Обобщение показателей качества подстановочных преобразований...91

Глава 3. Двухключевые криптосистемы....93

3.1. Сравнительная характеристика одноключевых и двухключевых шифров....93
3.2. От открытого распределения ключей до электронной цифровой подписи ..95
3.2.1 Система распределение ключей Диффи-Хеллмана....95
3.2.2 Открытый шифр Эль-Гамаля....96
3.3. Системы ЭЦП на основе задачи дискретного логарифмирования....97
3.3.1. Общие положения....97
3.3.2. Сокращение длины подписи....103
3.3.3. Примеры анализа слабых ЭЦП....106
3.3.4. Системы ЭЦП с дополнительными свойствами....109
3.3.5. Слепая подпись....113
3.3.6. Проблема бесключевого шифрования....115
3.4. ЭЦП на эллиптических кривых....121
3.4.1. Основные свойства эллиптических кривых....121
3.4.2. Групповой закон сложения точек на ЭК....122
3.4.3. Групповой закон сложения точек ЭК над конечными полями с различной характеристикой *p*....125
3.4.4. Способы повышение быстродействия вычислений в циклической группе точек ЭК....126
3.4.5. Исследование стойкости алгоритмов защиты информации, использующих эллиптические криптографические конструкции 130
3.4.6. Алгоритмы выбора ЭК....133
3.4.7. Оптимизация параметров эллиптических криптографических конструкций....146
3.4.8. Протоколы защищенного информационного обмена на основе свойств эллиптических кривых....147
3.5. Инфраструктура открытых ключей....156
3.5.1. Компоненты ИОК и их функции....156
3.5.2. Верификация цепочки сертификатов....157
3.5.3. Использование ИОК в приложениях....158
3.5.4. Стандарты в области ИОК и основанные на ИОК....159

Глава 4. Подстановочно-перестановочные сети с минимальным управляемым элементом 161

4.1. Управляемые битовые перестановки как криптографический примитив ... 161
4.2. Блочный шифр на основе переменных перестановок 165
4.3. Расширение класса управляемых операций с использованием управляемых элементарных инволюций 173
4.4. Полная классификация нелинейных элементов $F_{2/1}$ 182
4.5. Синтез управляемых операционных подстановок на основе элементов $F_{2/1}$ 191
4.5.1. Принципы построения управляемых операционных подстановок 191
4.5.2. Исследование основных свойств и оптимизация УОП 193
4.5.3. Вероятностные характеристики УОП 202
4.5.4. Оценка схемотехнической сложности реализации УОП 208

Глава 5. Класс управляемых элементов $F_{2/2}$ 209

5.1. Варианты представления и критерии отбора управляемых элементов $F_{2/2}$ 209
5.2. Классификация основных типов управляемых элементов $F_{2/2}$ по нелинейным и дифференциальным свойствам 214
5.3. Вопросы построения управляемых операций 219

Глава 6. Класс управляемых элементов $F_{3/1}$ 225

6.1. Варианты представления и критерии отбора управляемых элементов $F_{3/1}$ 225
6.2. Классификация УЭ $F_{3/1}$ по нелинейным и дифференциальным свойствам 228
6.3. Построение управляемых операционных блоков 241
6.4. Особенности использования УОБ на основе УЭ $F_{3/1}$ в синтезе криптографических функций 244
6.5. Сравнительная характеристика УЭ $F_{2/2}$ и $F_{3/1}$ 249

Глава 7. Переключаемые управляемые операции 251

7.1. Построение управляемых подстановочно-перестановочных сетей различного порядка 251
7.2. Проблемы построения блочных шифров с простым расписанием использования ключа 263
7.3. Понятие переключаемой операции 265
7.4. Управляемые операционные подстановки как класс попарно взаимно-обратных модификаций 266
7.5. Переключаемые управляемые операционные подстановки с симметричной топологической структурой 274
7.6. Переключаемые УППС различных порядков 278
7.7. Упрощение аппаратной реализации ПУОП 280

7.8. Переключаемые УППС с управляемыми элементами, включающими попарно взаимно-обратные модификации 282
7.8.1. Переключаемые УППС на основе элементов типа $F_{2/1}$ 283
7.8.2. Переключаемые УППС на основе элементов типа $F_{2/2}$ 284
7.8.3. Переключаемые УОП на основе элементов типа $F_{3/1}$ 286
7.9. Расширение свойства переключаемости УОП 294

Глава 8. Скоростные шифры с простым расписанием ключа 301

8.1. Криптосхемы и шифры на основе управляемых и переключаемых операций 301
8.2. Криптосхема COBRA–H64 318
8.3. Блочный шифр COBRA–H128 326
8.4. Блочные шифры на основе управляемых подстановочно-перестановочных сетей 331
8.5. Анализ стойкости и статистическое тестирование шифров на основе управляемых и переключаемых операций 337

Глава 9. Хэш-функции и их построение на основе управляемых операций 369

9.1. Защита от модифицирования данных 369
9.2. Хэш-функции 371
9.3. Построение хэш-функций на основе блочных преобразований 374
9.4. Нахождение коллизий в общем случае 379
9.5. Атака «встреча посередине» 384
9.6. О построении хэш-функций на основе управляемых операций 387
9.7. Хэш-функции на основе выборок значений из таблицы в зависимости от преобразуемых данных 392
9.8. Алгоритмы формирования расширенного ключа на основе УППС 396
9.8.1. Принципы построения алгоритмов формирования расширенного ключа 396
9.8.2. Исследование свойств АФРК 399

Глава 10. Криптографический практикум 405

10.1. Задания для практических занятий 405
10.1.1. Открытое шифрование 406
10.1.2. Системы цифровой подписи 408
10.1.3. Генерация простых чисел 415
10.2. Задачи для решения на практических занятиях. 420
10.2.1. Примитивы и схемы одноключевых криптосистем 406
10.2.2. Булевы функции 408
10.2.3. Двухключевые криптосистемы 415
10.3. Как запатентовать алгоритм 423

Заключение 435

Список литературы 439

Введение

В последние годы интерес к криптографии пробудился у широких кругов специалистов, работающих в самых разнообразных областях информационных технологий и их приложений. Особый интерес к этому предмету проявляют специалисты, имеющие отношение к разработке систем и средств защиты информации, а также практическому использованию последних. Откликом на такую потребность явилось появление на книжном рынке большого числа отечественных и переводных книг, посвященных криптографии и ее приложениям. Несмотря на это, многие современные направления в области шифрования остаются слабо освещенными. Одним из таких направлений прикладной криптографии является имеющий большое практическое значение новый подход к построению скоростных шифров, который заключается в использовании так называемых управляемых операций в качестве криптографических примитивов (концепция управляемых преобразований). Впервые в цельном виде это направление прикладной криптографии было представлено в книге «Криптография: скоростные шифры» [14], изданной в 2002 г. Первый тираж этой книги быстро разошелся, свидетельствуя, что даже эта относительно узкая область прикладной криптографии вызывает интерес достаточно широкого круга читателей. За прошедший год у авторов настоящей книги, которые проводят исследования по развитию концепции управляемых преобразований, появились новые интересные результаты. Эти результаты позволяют более полно представить возможности применения управляемых операций и дают более полное понимание их места среди других криптографических примитивов. В частности, они позволяют построить достаточно прозрачные для оценки основных свойств шифрующие преобразования, дают возможность применить новые типы обобщенных криптосхем, обладающих свойством универсальности и являются удобным учебным объектом. Значительно расширился класс управляемых операций.

Новые результаты явились основным содержанием настоящей книги. Однако авторы сочли целесообразным включить три главы, отражающие общую картину современной криптографии. Это позволит читателю более широко ознакомиться с проблематикой криптографии и ее идеями, кроме того, это отражает место концепции управляемых преобразований. Такая практика была апробирована в книге [14] и нашла понимание читателей, поскольку позволяет в рамках одной книги ознакомиться с более широким кругом задач, решаемых криптографией, и более разнообразными ее методами и идеями. Поскольку настоящая книга задумана как продолжение монографии [14], то первые три главы написаны на основе материала, дополняющего и существенно расширяющего вводную главу из [14]. Материал, изложенный в главах 1-3 из настоящей книги и главах 1 и 2 из [14], представляет собой достаточно полное

введение в современную криптографию и ее проблематику. Аналогично этому основное содержание этих книг изложено независимо, но в то же время обе книги отражают разные грани единого подхода – концепции управляемых преобразований в синтезе блочных шифров – и могут рассматриваться как два тома, посвященных одному вопросу.

Главы с четвертой по шестую посвящены вопросам построения управляемых подстановочно-перестановочных сетей (УППС), основанных на элементарных управляемых подстановках трех различных типов: 2×2 с двумя состояниями, 2×2 с четырьмя состояниями и 3×3 с двумя состояниями. Интерес к использованию таких управляемых сетей возрожден предложением и обоснованием использования управляемых перестановочных сетей для построения криптосхем, основанных на переменных перестановках, реализуемых в форме битовых перестановок, зависящих от шифруемых данных. Поскольку УППС являются более общим вариантом управляемых сетей и включают перестановочные сети как частный случай, то в общем случае можно ожидать, что переменные подстановки, выполняемые с помощью УППС, являются более эффективным криптографическим примитивом. Результаты указанных глав подтверждают это предположение. С увеличением размера управляемого элемента резко возрастает число различных вариантов их реализации и появляется больше возможностей при выборе стойких криптосхем. Однако, при этом существенно возрастает сложность аппаратной реализации, поэтому основное внимание было уделено анализу управляемых элементов, которые легко реализуются в заказных и программируемых интегральных схемах.

Седьмая глава посвящена построению переключаемых операционных блоков, основанных на УППС. Переключаемые управляемые операции представлены как новый широкий класс криптографических примитивов. Рассматриваются различные варианты топологии переключаемых УППС. Вводится понятие переключаемой управляемой операционной подстановки (УОП) и обсуждаются различные варианты их экономичной реализации. Показана перспективность использования переключаемых УОП для построения блочных шифров с простым расписанием использования ключа свободных от слабых ключей и гомогенности шифрующих преобразований.

В восьмой главе описывается ряд шифров, построенных на основе управляемых операций, включая переключаемые переменные подстановки и перестановки. Достоинством этих криптосистем являются высокие значения стойкости и скорости шифрования при относительно низкой сложности аппаратной реализации. Для ряда шифров с переменными операционными подстановками приводится дифференциальный анализ и рассматриваются некоторые другие типы криптоаналитических атак.

В девятой главе рассматриваются вопросы построения алгоритмов вычисления криптографических контрольных сумм, в том числе хэш-функций, которые применяются для осуществления контроля целостности данных и в системах цифровой электронной подписи. Данные построения также основаны на применении новых примитивов – операционных подстановок, зависящих от преобразуемых данных.

Десятая глава представляет собой криптографический практикум, который включает методический материал к практическим занятиям и список задач по различным разделам криптографии. В целом весь материал настоящей книги и моно-

графии [14] очень подходит для формирования заданий по курсовому проектированию по криптографии. На его основе можно предложить достаточное число вариантов курсовых заданий, относящихся к проектированию, тестированию и выполнению дифференциального криптоанализа блочных шифров, построенных на основе новых криптографических примитивов – операций преобразования, зависящих от преобразуемых данных. Варианты проведения практических занятий относятся к основным разделам двухключевой криптографии. Приведенный перечень задач включает задачи различного уровня и различной тематики, существенная доля из которых относится к управляемым операциям, что позволяет более глубоко освоить вопросы проектирования блочных шифров с их использованием. В криптографический практикум включены также вопросы патентования изобретений в области криптографии, в частности обсуждается каким образом можно запатентовать изобретенный алгоритм.

Данная книга рассчитана на широкий круг читателей: студентов, преподавателей, инженеров, аспирантов, научных работников и профессионалов в области защиты информации и криптографии. Авторы надеются, что данная книга позволит каждому читателю найти для себя новые и интересные вопросы, относящиеся к современной криптографии, которые представлены в доступной форме изложения и с ориентацией на практическое применение.

Главы 1, 2, 3 написаны совместно Молдовяном Н.А. и Еремеевым М.А.

Главы 4, 5, 7 и Заключение написаны Молдовяном А.А.

Главы 6, 8 и Введение написаны совместно Молдовяном Н.А., Молдовяном А.А. и Еремеевым М.А.

Главы 9 и 10 написаны совместно Молдовяном Н.А. и Еремеевым М.А.

Авторы приносят искреннюю благодарность своим коллегам – Бодрову А.В., Гуцу Н.Д., Изотову Б.В., Молдовяну П.А. и Морозовой Е.В за участие в совместных работах, Матвееву С.А. за техническую помощь и В.П. Чернолесу за неоценимые консультации при патентовании новых способов криптографического преобразования. Особая благодарность директору СЦПС «Спектр» Долгиреву В.А. создавшему своим подчиненным все условия для творческой работы над книгой.

Авторы очень признательны Беляеву Е.А., Буренину Н.И., Гаскарову Д.В., Зиме В.М., Макарову В.Ф., Пальчуну Б.П., Пожарскому В.Н., Советову Б.Я., Стрельцову А.А., Хиже Г.С., Чижову В.А., Юсупову Р.М. и другим ученым и специалистам, существенно содействовавшим получению новых результатов в затронутом направлении исследований.

Глоссарий

Алгебраическая степень нелинейности — количество сомножителей в крайнем правом элементе алгебраической нормальной формы булевой функции.

Алгоритм защитного контрольного суммирования — алгоритм вычисления некоторого двоичного вектора сравнительно малого размера, зависящего от каждого бита сообщения произвольной длины. Важнейшим типом алгоритмов защитного контрольного суммирования являются хэш-функции.

Аппаратный шифр — шифр, ориентированный на реализацию в виде электронного устройства.

Аутентификация — процедура установления подлинности пользователя (абонента сети, отправителя сообщения), программы, устройства или данных (информации, получаемого сообщения, ключа). Частным вариантом аутентификации является установление принадлежности сообщения конкретному автору.

Блочный шифр — шифр, осуществляющий преобразование блоков данных фиксированного размера.

Вероятностное шифрование — процесс шифрования с использованием случайных параметров.

Вес двоичного вектора — число ненулевых битов двоичного вектора.

Вес Хемминга — вес двоичного вектора.

Гибкий шифр — шифр, описываемый как набор криптоалгоритмов, выбираемых в зависимости от секретного ключа.

Гибридная криптосистема — криптосистема, в которой распределение ключей осуществляется с помощью двухключевых криптоалгоритмов, а процесс шифрования информации – с помощью одноключевых. Гибридные криптосистемы сочетают в себе удобство распределения секретных ключей и высокую скорость шифрования.

Двухключевая криптография — направление исследований в области криптографии и разработки криптосистем, основанных на использовании двух ключей – открытого и закрытого. Открытый (публичный, общедоступный) ключ предполагается известным всем пользователям криптосистемы и потенциальному противнику. Закрытый ключ предполагается известным только одному пользователю.

Дифференциальный криптоанализ — метод анализа шифров, заключающийся в поиске наиболее вероятных разностей, получаемых на выходе шифра при заданной входной разности.

Доверительный центр — организация, осуществляющая регистрацию, хранение и распространение открытых ключей в двухключевых криптосистемах. Основным назначением доверительного центра является аутентификация открытых ключей пользователей. Для распространения открытых ключей используются 1) электронные справочники открытых ключей и 2) цифровые сертификаты. Справочники и сертификаты подписываются доверительным центром.

Зашифрование — процесс преобразования множества открытых сообщений во множество шифрованных сообщений с использованием ключа в целях защиты от несанкционированного доступа. Стойкость зашифрования (или уровень защищенности) определяется тем, что при зашифровании используется один или несколько секретных параметров, которые предполагаются известными только законным (легальным) пользователям.

Имитозащита — способ защиты от навязывания ложных сообщений путем передачи вместе с сообщением небольшой дополнительной информации, называемой имитовставкой.

Имитовставка — криптографическая контрольная сумма, зависящая от каждого бита сообщения и вычисляемая с использованием криптографических алгоритмов (например, алгоритмов шифрования) и ключа.

Инволюция — операция преобразования (отображение некоторого множества на себя), являющаяся обратной самой себе.

Ключ — параметр шифра, определяющий выбор конкретного варианта преобразования для зашифрования или расшифрования из множества преобразований, составляющих шифр.

Конфиденциальность — свойство информации, характеризующее ее защищенность от ознакомления с ее содержанием со стороны лиц, не имеющих права доступа к ней.

Корреляционная иммунность порядка k — свойство булевой функции, заключающееся в сбалансированности всех частных функций, полученных из исходной функции фиксированием любых ее k или менее переменных. Данное свойство позволяет обеспечить стойкость криптографических преобразований к статистическим атакам при фиксированных значениях битов на входе преобразования. Максимальный порядок корреляционной иммунности булевой функции от n переменных не превышает значения n–1.

Криптоанализ — процесс (алгоритм) получения исходного текста по зашифрованному без знания ключа или процесс вычисления ключа по исходному и шифрованному тексту. Криптоанализ выполняется противником с целью получения возможности осуществления несанкционированного доступа или разработчиком с целью оценивания стойкости шифра.

Криптографическое преобразование — процедура специального преобразования информации для решения одной из следующих криптографических задач: шифрования данных, формирования цифровой электронной подписи, вычисления специальных криптографических контрольных сумм и имитовставки.

Криптографический примитив — в широком смысле это операция или процедура, используемая в качестве элемента шифра, в узком смысле это операции и проце-

дуры, определяющие требуемые свойства криптосистемы (стойкость, возможность зашифрования и расшифрования с использованием различных ключей и т. п.).

Криптографический протокол — протокол, предусматривающий взаимодействие двух и более сторон с использованием криптографических алгоритмов.

Криптосистема — система обеспечения безопасности защищенной сети, использующая криптографические средства. В качестве подсистем может включать системы шифрования, идентификации, цифровой подписи и др., а также систему распределения ключей.

Криптостойкость (*стойкость шифра*) — способность криптосистемы противостоять атакам противника, направленным на получение ключа, открытого сообщения или навязывание ложного сообщения. Количественно выражается числом операций некоторого типа, которые необходимо выполнить для решения задачи криптоанализа. При этом указывается тип атаки на криптосистему, т. е. исходные данные для криптоанализа. Если исходные данные не указываются, то подразумевается стойкость к лучшему известному алгоритму криптоанализа (для атаки на основе специально подобранных текстов).

Лавинный эффект — свойство распространения влияния одного входного бита на многие выходные биты (свойство размножения ошибок).

Линейный криптоанализ — метод анализа шифров, заключающийся в поиске наилучших линейных аппроксимаций для отображений, выполняемых шифром.

Лобовая атака (силовая атака) — криптоанализ путем исчерпывающего перебора всех возможных ключей, т.е. методом подбора ключа. Для криптосистем с конечным ключом этот метод является универсальным, т. е. он применим к любому шифру такого типа. Однако, вероятность раскрытия ключа с помощью метода опробования всех возможных ключей является весьма низкой. Для задания низкой вероятности успеха такой атаки для симметричных криптосистем требуется использовать ключи размером не менее 80 бит. В настоящее время считается, что гарантированную стойкость к лобовому нападению обеспечивают равновероятно генерируемые случайные ключи размером 128 бит.

Модификация управляемой операции — преобразование, осуществляемое управляемой операцией при заданном фиксированном значении управляющего вектора.

Нелинейность — свойство булевой функции, показывающее степень удаленности булевой функции от множества аффинных или линейных булевых функций. Определяется значением минимального расстояния Хэмминга между заданной функцией и множеством аффинных функций. Характеризует стойкость к линейному методу криптоанализа.

Операционная подстановка — преобразование, осуществляемое с помощью операционного блока, представляющего собой подстановочно-перестановочную сеть.

Переключаемая операция — операция, выполняемая в зависимости от дополнительного однобитового управляющего вектора e, причем модификации, соответствующие значениям $e = 0$ и $e = 1$, являются взаимно-обратными. Частным случаем переключаемых операций являются переключаемые управляемые операции.

Перестановочная сеть — управляемая перестановочная сеть, состоящая из некоторого числа активных каскадов (активных слоев), между которыми расположены узлы фиксированной коммутации (фиксированные перестановки), причем каждый активный каскад состоит из совокупности управляемых элементарных переключателей, управляемых некоторым управляющим битом и выполняющих тождественное преобразование или перестановку двух битов в зависимости от значения управляющего бита.

Подстановочно-перестановочная сеть — узел преобразования, состоящий из некоторого числа каскадов (слоев), между которыми расположены узлы фиксированной коммутации (перестановки), причем каждый каскад состоит из совокупности блоков подстановки (обычно одного размера).

Управляемая подстановочно-перестановочная сеть — управляемая подстановочно-перестановочная сеть, состоящая из некоторого числа активных каскадов (слоев управляемых подстановок), между которыми расположены узлы фиксированной коммутации (фиксированные перестановки), причем каждый активный каскад состоит из совокупности управляемых блоков подстановок, управляемых некоторым управляющим двоичным вектором (или битом), задающим выбор текущей реализуемой подстановки.

Порядок управляемой перестановки — максимальное количество входных битов, которые могут быть переставлены в произвольные разряды выходного двоичного вектора при некотором значении управляющего вектора.

Поточный шифр — шифр, преобразующий последовательно отдельные биты или знаки исходного сообщения.

Преобразование Уолша-Адамара булевой функции — разновидность дискретного преобразования Фурье, заключающееся в линейном преобразовании векторного n-мерного пространства над двоичным полем в значение действительного числа и имеющее следующий вид: $U_{\alpha}\left(f(X)\right) = \sum_{X \in \mathrm{GF}(2)^n} f(X)(-1)^{\langle \alpha, X \rangle}$, где знак "$\langle\ \rangle$" обозначает скалярное произведение двух векторов.

Противник — субъект (оснащенная группа специалистов или физическое лицо), пытающийся преобразовать шифртекст в открытый текст без знания ключа или вычислить ключ по известным исходным и шифрованным текстам. Синонимами являются термины – атакующий, нападающий, криптоаналитик, криптоаналитик противника.

Предвычисления — процедуры преобразований или вычисления, выполняемые предварительно и используемые в дальнейшем при многократном выполнении некоторого алгоритма.

Программный шифр — шифр, ориентированный на реализацию в виде программы.

Простое расписание использования ключа — расписание использования ключа, при котором подключи, входящие в шифрующие преобразования, представляют собой непрерывные битовые цепочки, являющиеся частью секретного ключа. Представляет собой вариант отказа от сложных процедур преобразования сек-

ретного ключа. Применяется с целью уменьшения сложности схемотехнической реализации шифров и увеличения производительности криптосистем.

Протокол рукопожатия — протокол, позволяющий двум удаленным сторонам, владеющим некоторым общим секретом, осуществить взаимную проверку подлинности без раскрытия секрета.

Протокол тайного электронного голосования — криптографический протокол, обеспечивающий тайну голосования для каждого голосующего при возможности проверки со стороны последнего, как учтен его голос («за» или «против»). При этом никто, включая избирательный комитет, не сможет установить, как проголосовал избиратель.

Расшифрование — процесс преобразования множества шифрованных сообщений во множество открытых сообщений с использованием ключа. (Отметим, что вместо термина «расшифрование» иногда используется термин «дешифрование». В российской специальной литературе под термином «дешифрование» обычно понимают процесс восстановления открытого текста без знания ключа или вычисление ключа по открытым текстам и шифртекстам.)

Раскрытие шифра (криптосистемы, криптоалгоритма) — нахождение способа решения задачи криптоанализа за разумное время при использовании современных вычислительных средств. В качестве синонима используется термин «взлом шифра».

Регулярное отображение — такое отображение $Y=\varphi(X):GF(2)^n \to GF(2)^m$ при $n \geq m$, при котором Y ровно 2^{n-m} раза принимает все 2^m различных значений из $GF(2)^m$, в то время как X проходит 2^n значений из $GF(2)^n$. При $n=m$ каждое регулярное отображение является биективным. Необходимым условием регулярности отображения при $n \geq m$ является сбалансированность всех линейных комбинаций булевых функций, выполняющих данное отображение.

Режим шифрования — вариант осуществления криптографического преобразования информации (зашифрование и расшифрование) или вариант использования шифра. Например, блочные шифры могут быть использованы в режиме электронной кодовой книги, режиме сцепления блоков шифра, режиме выработки ключевой гаммы.

Резилентная (совершенная) булева функция — это корреляционно-иммунная степени k булева функция, которая является сбалансированной функцией.

Сбалансированная булева функция — булева функция, таблица истинности которой содержит одинаковое количество нулей и единиц. Данное понятие эквивалентно условию, что значение преобразования Уолша-Адамара на нулевом векторе равно нулю.

Симметричная криптосистема — криптосистема с секретным ключом. Симметричность означает, что ключи, используемые при зашифровании и в процессе расшифрования одинаковы, или могут быть получены один из другого с небольшой трудоемкостью.

Слепая подпись — протокол формирования цифровой электронной подписи, позволяющий сформировать правильную ЭЦП, соответствующую некоторому цифро-

вому сообщению, которое подписывающая сторона получает в зашифрованном (т. е. недоступном для нее) виде.

Совершенно нелинейная булева функция (*бент-функция*) — это булева функция от n переменных, для которой расстояние Хэмминга до множества всех аффинных функций над векторным n-мерным пространством максимально и имеет постоянную величину.

Статистические тесты — экспериментальные тесты, включающие статистическую обработку двоичных последовательностей или множеств двоичных векторов с целью проверки гипотез о свойствах последовательностей или алгоритмов, формирующих двоичные последовательности и вектора. Применяются для оценки качества алгоритмов шифрования или формирования ключей.

Стеганография — способ скрытной передачи информации или способ формирования скрытного канала. Часто используется для передачи шифртекстов. Широкое применение компьютерных технологий дало новый сильный импульс в развитии стеганографических методов. В настоящее время наблюдается тенденция использования криптографических идей в современной стеганографии, например, секретных ключей. Актуальным направлением стеганографии является создание цифровых водяных знаков.

Удостоверяющий центр — доверительный центр.

Управление ключами — совокупность мероприятий и процедур, обеспечивающих защищенную генерацию, распределение, хранение и уничтожение ключей.

Управляемые операции — операции, описываемые как множество некоторых более простых операций, выбираемых в зависимости от значения некоторого управляющего кода.

Управляемые инволюции — управляемые операции, каждая модификация которых является инволюцией.

Управляемые перестановочные инволюции — управляемые перестановки, каждая модификация которых является инволюцией.

Управляемый элемент — элементарный управляемый блок (узел) управляемой подстановочно-перестановочной или перестановочной сети; обозначается как блок $\mathbf{F}_{t/w}$, реализующий преобразование t-битовых двоичных векторов в зависимости от w-битового управляющего вектора; используется в качестве типового элемента при построении управляемых подстановочно-перестановочных сетей. Типичными значениями параметров w и t являются $w = 1, 2$ и $t = 2, 3, 4$.

Управляющий вектор — двоичный вектор, задающий выбор текущей модификации управляемой операции. Синонимом является термин «управляющий код».

Уравнение вычисления подписи — уравнение, задающее определенное соотношение между секретным ключом, хэш-функцией документа и цифровой подписью. Служит для формирования цифровой электронной подписи, подлинность которой может быть проверена с использованием открытого ключа.

Уравнение проверки подписи — уравнение, задающее определенное соотношение между открытым ключом, хэш-функцией документа и цифровой подписью. Служит для проверки подлинности подписи.

Хэширование — процесс вычисления значения хэш-функции.

Хэш-функция — криптографическая функция (процедура, алгоритм), аргументом которой являются произвольные сообщения (тексты, документы), представленные в виде последовательности битов, и значения которой лежат в области от 0 до $2^m - 1$, где m – размер хэш-кода (выходного значения хэш-функции). Хэш-функции представляют собой специальный класс криптографических контрольных сумм, вычисляемых обычно без использования секретных параметров и обеспечивающих сложную зависимость выходного значения от каждого бита входного сообщения.

Целостность информации — характеристика соответствия информации ее эталонному состоянию. Нарушение целостности информации есть ее несанкционированное модифицирование (умышленное или неумышленное).

Цифровой сертификат — электронный документ, содержащий информацию о владельце сертификата (ФИО, должность, организация, адрес, срок действия, открытый ключ и др.) и подписанный доверительным центром.

Цифровая подпись — электронная цифровая подпись.

Шифр с открытым ключом — двухключевая криптосистема. Особенностью данных криптосхем является наличие связанных между собой ключей для зашифрования и расшифрования, каждый из которых не может быть получен из другого с приемлемыми вычислительными и временными затратами. Синонимами являются термины «криптосистема с открытым ключом», «асимметричная криптосистема».

Шифр — семейство обратимых преобразований множества открытых сообщений в множество шифрованных сообщений и обратно, каждое из которых определяется некоторым параметром, называемым ключом.

Шифрование — процесс преобразования информации с использованием некоторой дополнительной информации, управляющей этим процессом и называемой ключом. Реализуется в виде процедуры расшифрования или зашифрования. Часто под шифрованием понимается зашифрование.

Шифратор — электронное устройство или программа, реализующая алгоритмы шифрования.

Электронная цифровая подпись — некоторая дополнительная информация, соответствующая данному электронному документу (сообщению), которая могла быть сформирована только владельцем некоторого секрета — закрытого ключа и которая позволяет с использованием специального алгоритма установить факт соответствия подписи закрытому ключу подписывающего. Под электронной цифровой подписью (ЭЦП) понимается также криптографическая система (совокупность алгоритмов и правил), позволяющая подписывать цифровые сообщения и проверять правильность формируемых цифровых подписей.

Обозначения

$\mathbf{Z}$ – множество целых чисел;

$E_Q(M)$ – функция преобразования сообщения M в зависимости от ключа Q;

$GF(q)$ – конечное поле или поле Галуа, $q = p^m$, где p – простое число, называемое характеристикой конечного поля, m – натуральное число, называемое степенью поля, q – порядок поля (количество элементов поля);

$GF(2)$ – конечное двоичное поле;

$GF(2)^n$ – n–мерное векторное пространство над двоичным полем;

$\mathbf{F}, \mathbf{F}^{-1}$ – прямая и обратная управляемые операции;

$\mathbf{F}_1 \bullet \mathbf{F}_2$ – суперпозиция операций $\mathbf{F}_1$ и $\mathbf{F}_2$;

$\oplus$ – операция поразрядного суммирования по модулю два;

$>>>k$ – операция циклического сдвига на k бит;

$\|$ – операция конкатенации двух двоичных векторов;

$a \leftarrow b$ – операция присваивания значения b переменной a;

$\#\, M$ *или* $\#\, \{M\}$ – мощность множества M;

$НОД(a, b)$ – наибольший общий делитель чисел a и b;

$\mathbf{Z}_p = \mathbf{Z}/p$ – множество классов вычетов целых чисел по модулю p (кольцо вычетов по модулю p);

$\phi(n)$ – функция Эйлера;

$GF(2)^n \rightarrow GF(2)^m$ – отображение пространства $GF(2)^n$ n-мерных векторов над полем $GF(2)$ в другое пространство $GF(2)^m$ m-мерных двоичных векторов;

$\langle a, b \rangle$ – скалярное произведение векторов a и b;

$S(f)$ – таблица истинности булевой функции $f : GF(2)^n \rightarrow GF(2)$;

$\boldsymbol{wt}(\alpha)$ – вес Хэмминга двоичного вектора α (количество единиц);

$d(f, g)$ – расстояние Хемминга между двумя двоичными векторами f и g (таблицами истинности булевых функций)

$r_\beta(f)$ – значение автокорреляционной функции булевой функции f относительно двоичного вектора β;

$U_\alpha(f)$ – значение преобразования Уолша-Адамара булевой функции f относительно вектора α;

$\lfloor n \rfloor$ – наибольшее целое число, меньшее либо равное значению аргумента n;

$\lceil n \rceil$ – наименьшее целое число, большее либо равное значению аргумента n;

deg(f) – алгебраическая степень нелинейности булевой функции (наибольшее количество сомножителей в алгебраической нормальной форме булевой функции);

NL(f) – нелинейность булевой функции f (значение минимального расстояния Хэмминга между функцией f и множеством всех аффинных функций);

$\mathbf{P}_{2/1}$ – элементарный блок управляемых перестановок, выполняющий или не выполняющий перестановки двух входных битов в зависимости от одного управляющего бита;

$\mathbf{P}_{m/n}(X, V)$ – блок управляемых перестановок, осуществляющий перестановку битов в n-разрядном входном векторе $X \in \mathrm{GF}(2)^n$ в зависимости от m-разрядного управляющего вектора $V \in \mathrm{GF}(2)^m$;

$\mathbf{F}^{(v)}$ – конкретная модификация управляемой операции, соответствующая управляющему вектору v;

$\mathbf{F}_{2/1}$ – элементарный управляемый элемент, выполняющий подстановку над двумя входными битами в зависимости от одного управляющего бита;

$\mathbf{F}_{n/m}(X, V)$ – управляемая подстановочно-перестановочная сеть (управляемая операционная подстановка), осуществляющая подстановочное преобразование n-разрядного входного вектора $X \in \mathrm{GF}(2)^n$ в зависимости от m-разрядного управляющего вектора $V \in \mathrm{GF}(2)^m$.

Сокращения

АКФ – автокорреляционная функция;
АНФ – алгебраическая нормальная форма;
АФРК – алгоритм формирования расширенного ключа;
БУОП – блок управляемой операционной подстановки;
БУП – блок управляемых перестановок;
БФ – булева функция;
ВБФ – векторная булева функция;
ВКФ – взаимная корреляционная функция;
ЗКС – защитная контрольная сумма;
ИА – итеративная архитектура;
ИОК – инфраструктура открытых ключей;
КА – конвейерная архитектура;
КВК – коэффициент взаимной корреляции;
КИ – корреляционная иммунность;
КР – критерий распространения;
КСЛЭ – критерий строгого лавинного эффекта;
ЛХ – линейная характеристика;
НОД – наибольший общий делитель;
ОПУА – обратное преобразование Уолша-Адамара;
ПЛИС – программируемая логическая интегральная схема;
ППС – подстановочно-перестановочная сеть;
ПРИК – простое расписание использования ключа;
ПРК – простое расписание ключа;
ПУА – преобразование Уолша-Адамара;
ПУО – переключаемая управляемая операция;
ПУОП – переключаемая управляемая операционная подстановка;
ПУСК – процедура усложнения секретного ключа;

СБИС – сверхбольшая интегральная схема;
СОС – список отозванных сертификатов;
Спектр УА – спектр Уолша-Адамара;
УОБ – управляемый операционный блок;
УОП – управляемая операционная подстановка;
УП – управляемая перестановка;
УППС – управляемая подстановочно-перестановочная сеть;
УПС – управляемая перестановочная сеть;
УЦ – удостоверяющий центр;
УЭ – управляемый элемент;
ЦР – центр регистрации;
ЦС – центр сертификации;
ЭВМ – электронная вычислительная машина;
ЭК – эллиптическая кривая;
ЭЦП – электронная цифровая подпись.

ГЛАВА 1

Вопросы одноключевой криптографии

1.1. Варианты реализации шифров

В настоящее время алгоритмы шифрования, представляющие практический интерес, являются для ручного шифрования достаточно сложными. Также в историю вошли алгоритмы, лежавшие в основе механических и электромеханических шифраторов. Современные алгоритмы шифрования обычно реализуются в виде некоторого электронного устройства, основной частью которого является крипточип – интегральная схема, реализующая алгоритм шифрования, и в виде программ для ЭВМ. Алгоритмы, разрабатываемые как стандарты для широкого использования, должны обеспечивать высокую производительность как при программной реализации, так и при недорогой аппаратной реализации. Однако для многих технологических применений предполагается использование либо устройств шифрования, либо программ шифрования. Очевидно, что в таких случаях разумно ориентироваться только на один из указанных вариантов реализации. Это позволит повысить производительность шифрующих программ, а при аппаратной реализации — существенно уменьшить стоимость криптографических устройств. Способы построения аппаратно-ориентированных шифров рассмотрены в книге [14].

К программным шифрам относятся криптосистемы, которые обеспечивают высокую производительность шифрования данных при их реализации в виде компьютерных программ. Система команд микропроцессоров является достаточно ограниченной, однако она позволяет реализовать достаточно производительные алгоритмы шифрования. В значительной степени это связано с тем, что в системе команд всегда присутствует команда пересылки (чтения) содержимого некоторой ячейки оперативной памяти в один из регистров процессора. Данная операция, реализующая выборку из некоторого массива в памяти, представляет собой не что иное, как нахождение значения некоторой функции, заданной табличным способом, по некоторому значению аргумента, заданному как адрес ячейки памяти. Табличным способом могут быть заданы произвольные функции, в том числе и операции подстановок, которые являются базовым криптографическим примитивом во многих современных криптосистемах.

В конечном счете, любой блочный шифр представляет собой чрезвычайно большое множество подстановок большого размера (число возможных входных значений 2^{64} для 64-битового шифра или 2^{128} для 128-битового), выбираемых в зависимости от секретного ключа. Однако такое непосредственное задание шифра практически нереализуемо, поскольку требует неимоверно большого объема памяти. Но такие под-

становки можно генерировать. Соответствующий генератор и представляет собой алгоритм шифрования. Если бы мы могли выбирать непосредственным образом некоторую секретную подстановку, то в принципе мы могли бы выбрать ее случайным образом. При использовании генератора формируемые подстановки не являются случайными, даже если мы выберем случайный секретный ключ, который задает выбор конкретной подстановки. Проблема разработки алгоритмов шифрования – это проблема задания такого множества подстановок, каждая из которых являлась бы псевдослучайной (практически неотличимой от случайной). Иными словами, разработка стойкого шифра связана с построением генератора псевдослучайных подстановок, управляемого секретным ключом.

Что касается подстановок малого размера (4×4, 8×8 и даже 16×16), то они легко реализуются программным и аппаратным способом и используются как базовые операции при проектировании шифров. При этом учитывается, что при увеличении размера подстановки резко возрастает ресурс, необходимый для их реализации. Наиболее эффективными для аппаратной реализации представляются подстановки размера 4×4, а для программной – 4×4 и 8×8. Подстановки размера 8×8 также имеют приемлемую сложность схемотехнической реализации. Именно эти варианты подстановок и представляют собой криптографический примитив, который позволяет сочетать эффективность программной и аппаратной реализации разрабатываемого шифра. Другой операцией, органически дополняющей операцию подстановки, является перестановка битов преобразуемых данных. Произвольная перестановка аппаратным путем реализуется практически без затрат ресурсов, а при программной реализации с использованием современных процессоров широкого назначения она вносит существенную задержку в процесс шифрования. Этот недостаток может быть полностью устранен путем расширения системы команд универсального процессора командой управляемой битовой перестановки, что и было предложено в работах [3, 20].

Выбор размера и конкретных таблиц подстановок при построении шифров является одной из главных задач, которые должны быть решены, но сама возможность эффективного осуществления операций подстановок при программной реализации обеспечивается стандартной системой элементарных команд процессора. Вместо перестановок произвольного типа в программных шифрах используются частные виды перестановок, например, операция циклического сдвига. Наряду с базовой операцией подстановки могут быть использованы другие элементарные команды процессора:

- операции циклического сдвига на фиксированное число двоичных разрядов;
- операции циклического сдвига на фиксированное число двоичных разрядов, зависящее от ключа;
- операции циклического сдвига на фиксированное число двоичных разрядов, зависящее от преобразуемых данных;
- поразрядное суммирование по модулю два;
- суммирование и/или вычитание по модулю 2^{32};
- операции поразрядного логического умножения и/или сложения двух n-битовых двоичных векторов и др.

В целом разработка программных шифров связана с учетом специфики обработки данных в компьютерных системах, что позволяет получить высокие скорости шифрования при использовании микропроцессоров широкого применения. Практическая потребность решения проблемы защиты электронной информации в массовом масштабе обусловливает актуальность разработки программных шифров и перспективы их широкого применения.

При выборе операций подстановок можно использовать следующие возможности, связанные с программной реализацией:

- можно использовать таблицы подстановок, зависящие от секретного ключа (в этом случае таблицы подстановок формируются по ключу на этапе предвычислений, выполняемых при инициализации криптосистемы);
- можно использовать таблицы подстановок достаточно большого размера;
- можно использовать табличные подстановки, зависящие от преобразуемых данных (данный тип подстановок реализуется с помощью некоторого множества пронумерованных таблиц подстановок).

Операции подстановок являются только частным случаем табличного задания функций отображения. Подстановки являются биективными отображениями, т. е. позволяют однозначно выполнить обратное преобразование. Это свойство необходимо для многих схем построения алгоритмов шифрования. Но оно не является необходимым условием для обеспечения возможности осуществления расшифрования закрытого сообщения. Например, в криптосхеме Фейстеля при построении раундовой функции могут быть использованы произвольные операции преобразования. Таким образом, представляют интерес не только подстановки, но и операции отображения более общего типа. Последние также могут быть эффективно реализованы программными средствами. Вместо операций подстановок или дополнительно к ним в программных шифрах можно использовать операции табличного отображения (фиксированные, зависящие от ключа и/или от преобразуемых данных). Одним из важных вариантов реализации операций табличного отображения является механизм выборки подключей в зависимости от преобразуемых данных, который успешно использован при разработке ряда скоростных программных шифров [93-96].

С программной реализацией криптосистем связана возможность применения достаточно сложных процедур предвычислений, реализуемых как этап инициализации криптосистемы, осуществляемый после ввода секретного ключа. В частности, инициализация может включать процедуру настройки алгоритма шифрования по секретному ключу [40]. При аппаратной реализации использование сложных предвычислений приводит к значительному повышению стоимости устройств шифрования и снижению их производительности при частой смене ключей. Шифры, в которых алгоритм шифрования формируется в зависимости от секретного ключа, называются гибкими или недетерминированными. В гибких шифрах каждому ключу соответствует уникальная модификация алгоритма шифрования. Поскольку множество ключей ограничено, то это означает, что гибкий шифр представляет собой множест-

во алгоритмов шифрования, описываемых с помощью некоторого алгоритма, который задает правило формирования алгоритма шифрования в зависимости от секретного ключа. Формирование секретных таблиц подстановок, таблиц операции отображения и настройка алгоритма шифрования предполагают использование этапа настройки шифра, выполняемой однократно при введении секретного ключа. После настройки криптосистема многократно выполняет процедуры шифрования и расшифрования данных. При сравнительно редкой смене ключей (например, ключ сменяется каждые 10 секунд) наличие этапа настройки практически не изменяет среднюю скорость шифрования.

В случае ограничения длины k секретного ключа (например, $k < 40$ бит) возможность значительного усложнения этапа предвычислений, используемого в программных шифрах, может быть использована для повышения стоимости (сложности) раскрытия ключа. Это позволит уменьшить вероятность раскрытия ключа со стороны большого числа потенциальных нарушителей. Еще более надежную страховку может обеспечить использование алгоритмов, настраиваемых по некоторому дополнительному параметру, который должен быть известен узкому кругу пользователей и иметь достаточный размер r (например, $r = 80$ бит). В этом случае для внешнего нарушителя переборная сложность задачи раскрытия криптосистемы может быть оценена как 2^{k+r} шифрований, что существенно превышает сложность задачи криптоанализа (2^k шифрований) для внутренних пользователей.

1.2. Повышение стойкости шифрования при ограничении длины секретного ключа

В случае коротких ключей наиболее результативным способом нападения на шифр может стать силовая атака, которая заключается в переборе всех возможных вариантов секретного ключа (или пароля, если последний используется в качестве ключа). Наличие этапа предвычислений служит определенным барьером против силовой атаки. Наличие предвычислений, которые необходимо выполнить для каждого испытываемого варианта ключа, существенно затрудняет такую атаку. Время выполнения процедур настройки t может быть задано достаточно большим путем усложнения алгоритма настройки или путем многократного его использования. Трудоемкость силовой атаки можно оценить по формуле:

$$W \approx 2^k W_t/2,$$

где W_t – трудоемкость алгоритма предвычислений, k – длина секретного ключа в битах (предполагается, что ключ является случайным и равновероятным по множеству всех ключей длины k). Мы полагаем, что атакующий имеет некоторый известный исходный текст и соответствующий ему шифртекст, причем процедура его зашифрования имеет сложность намного меньше значения W_t (т. е. время шифрования известного текста пренебрежимо по сравнению с t и составляет, например, одну миллисекунду). В случае пароля, состоящего из букв естественного языка или выбираемого из словаря, эта формула также может быть применена, если в качестве значения k использовать некоторую эффективную длину $k_e < k$.

Требуемое время для выполнения силовой атаки можно оценить по формуле:

$$T \approx 2^{k-1}t,$$

где t – время, затрачиваемое на выполнение алгоритма предвычислений. Для многих приложений значение t равное от 0.5 до 1 секунды, является вполне приемлемым, поскольку для законного пользователя эта задержка относится только к однократно выполняемой инициализации криптосистемы. Однако, для нарушителя при длине ключа, составляющей $k = 32$ бит, в среднем время силового взлома составит более 50 лет работы однопроцессорной ЭВМ широкого применения (для которой $t = 0.5$ с). Очевидно, что это делает стоимость раскрытия ключа для многих потенциальных нарушителей (например, хакеров) неприемлемой и заставит их отказаться от осуществления атаки. Для проведения атаки за разумное время потребуется использование очень большого числа ЭВМ. Например, для того чтобы раскрыть один ключ за один месяц, потребуется непрерывная работа более 500 компьютеров широкого применения либо применение специализированных многопроцессорных ЭВМ, которые есть в наличии только у весьма ограниченного числа организаций и являются весьма дорогостоящими, равно как и их эксплуатация.

Аналогичные оценки для секретных ключей, имеющих длину от 8 до 10 байт (т.е. от 64 до 80 бит), показывают, что силовой взлом шифров с предвычислениями легко сделать практически неосуществимым даже для нарушителя, обладающего очень мощными вычислительными ресурсами. Несомненно, противодействие силовой атаке путем увеличения длины ключа является наиболее эффективным общим приемом для всех криптосистем, допускающих произвольный выбор длины секретного ключа, однако при ограничении его длины этот прием не может быть использован в полной мере.

С пользовательской точки зрения, проблема выбора пароля (секретного ключа) и повышения стойкости к атакам на основе подбора пароля заслуживает внимания, поскольку для всех широко используемых систем защиты требуется выбирать хорошие (случайные) пароли, которые трудно запомнить. В средствах защиты можно использовать активный контроллер паролей, т.е. систему, блокирующую выбор плохих паролей. Другое возможное решение проблемы совмещения удобства пользователей с высоким уровнем секретности состоит в использовании паролей-фраз, т.е. не отдельных слов, а целых фраз, которые удобны для запоминания, но трудны для подбора из-за большой длины. Целесообразно построение таких механизмов парольного доступа к ресурсам ЭВМ, которые содержат встроенный механизм противодействия атакам на основе перебора возможных паролей (или парольных фраз). Такие механизмы могут состоять, например, в использовании достаточно сложных процедур вычисления однонаправленной функции от пароля, требующих времени около 1с для процессоров широкого применения. В постоянной памяти ЭВМ будет храниться таблица значений этой однонаправленной функции от паролей всех законных пользователей (таблица образов паролей), а проверка подлинности текущего пользователя будет осуществляться как вычисление значения этой функции от значения текущего пароля и сравнение полученного значения с соответствующим значением из таблицы образов паролей.

Применение долговременных ключевых элементов при ограничении длины секретного ключа

Наиболее результативным способом повышения стойкости криптосистем в условиях применения коротких секретных ключей (например, имеющих длину $k = 32$ бит) является использование долговременных ключей. Примером криптосистемы с долговременным ключом является российский стандарт шифрования ГОСТ 28147–89 [8], в котором таблицы подстановок являются секретными. В соответствии с общепризнанным принципом Керхкоффа при оценке стойкости алгоритм шифрования предполагается известным атакующему. В более общей трактовке этот принцип можно выразить так: *все долговременные элементы механизмов защиты информации необходимо считать известными потенциальному нарушителю*. Это связано с тем, что обычно долговременные элементы известны многим участникам разработки шифра.

Идея использовать долговременные ключи большого размера или долговременные секретные элементы алгоритма шифрования для повышения стойкости является отступлением от принципа Керхкоффа, но в случае ограничения длины секретного ключа этот прием является обоснованным тем, что этот элемент усиления не имеет цель обеспечить гарантированную стойкость. Преследуемая цель состоит только в существенном повышении сложности раскрытия короткого ключа, например, со стороны внешних нарушителей. Ограниченная длина ключа предполагает, что предусматривается защита от потенциального нарушителя с весьма ограниченными вычислительными ресурсами. При наличии у нарушителя больших вычислительных ресурсов подбор ключа реализуем за приемлемое время. Использование долговременного ключа для обоих типов нарушителей потребует определения долговременного ключа, а это может быть сделано либо путем более сложного криптоанализа, либо получением этого ключа «некриптографическими» методами. Долговременный ключ может представлять собой:

- секретные константы;
- секретные таблицы подстановок (отображений);
- секретный алгоритм предвычислений (настройки);
- секретный алгоритм шифрования.

При правильном построении криптосистемы короткий ключ вычислить практически невозможно (при использовании сколь угодно больших реально существующих вычислительных ресурсов) без знания перечисленных секретных элементов (если таковые используются). Это сделает невозможной задачу раскрытия короткого секретного ключа при атаке на криптосистему, осуществляемую субъектами, не владеющими долговременным ключом.

С точки зрения общей оценки криптографической стойкости, использование долговременных ключей не приводит к повышению секретности, поскольку долговременный ключ предполагается известным фиксированному кругу пользователей,

внутри которого реально действующим является только короткий ключ. Примером систем с долговременным ключом является стандарт шифрования ГОСТ 28147–89. Используемое в нем «заполнение таблиц блока подстановки», которое «является долговременным ключевым элементом, общим для сети ЭВМ», должно рассматриваться известным атакующей стороне в некоторых возможных ситуациях. Согласно описанию этого стандарта «заполнение таблиц блока подстановки является секретным элементом и поставляется в установленном порядке». Однако роль такого секретного параметра должна быть оценена с учетом упомянутых выше замечаний.

1.3. Криптосистемы с гибким алгоритмом и требования к алгоритму предвычислений

Использование долговременных (хотя даже и сменяемых) секретных частей шифрующих систем в общем случае не приводит к существенному повышению стойкости. Однако сама идея использования не только секретных параметров, но и секретных элементов алгоритма шифрования заслуживает внимания. Если секретные элементы алгоритма сделать легко сменяемыми, то в этом случае можно говорить о гибких криптосистемах или шифрах с гибким алгоритмом. Такие конструкции могут быть легко реализованы в программных шифрах. С примерами можно ознакомиться в работах [14, 42, 94]. В гибких шифрах долговременным элементом являются процедуры настройки алгоритма шифрования, а конкретная его модификация и ключ шифрования являются сменными элементами криптосистемы, которые автоматически заменяются одновременно со сменой паролей (ключей) и являются уникальными для каждого пользователя (или каждой пары абонентов защищенной сети связи).

Хранение описания секретной модификации алгоритма шифрования приводит к определенным неудобствам для пользователей. Кроме того, при очень большом числе таких модификаций возникают проблемы их хранения. Устранение этих проблем связано с использованием генератора модификаций алгоритма шифрования. Формируется алгоритм, генерирующий конкретную модификацию по конкретному вводимому секретному параметру. Этим параметром может быть дополнительный или основной секретный ключ. В последнем случае только длина секретного ключа будет определять переборную стойкость криптосистемы. Тем не менее, секретность конкретной модификации алгоритма приводит к существенному повышению стойкости к другим типам атак, которые являются наиболее опасными. Действительно, переборную атаку легко устранить простым удлинением ключа, тогда как противодействие некоторым другим типам атак требует тщательной проработки всех элементов криптосистемы.

Процедура формирования алгоритма шифрования по секретному ключу является существенно более сложной по сравнению с непосредственным шифрованием данных. Очевидно, что настройку алгоритма шифрования разумно выполнить как часть предвычислений. Алгоритм предвычислений является частью криптосистемы, поэтому он также вносит свой вклад в задание общей секретности (стойкости) шифра. Эта часть не является столь критичной как сам алгоритм непосредственного шифрования. Кроме того, для реализации предвычислений могут быть использованы значи-

тельные вычислительные ресурсы, что упрощает задачу построения необходимых процедур предвычислений. В общем случае секретная модификация алгоритма непосредственного шифрования и некоторые сгенерированные в зависимости от секретного ключа параметры или массивы данных могут быть рассмотрены как расширенный ключ. Рассмотрим общие требования к алгоритму предвычислений. Одним из требований к алгоритму формирования ключа шифрования является следующее: *количество возможных выходных последовательностей не должно быть существенно меньше числа возможных секретных ключей*. Желательно, чтобы число возможных расширенных ключей было равно числу различных значений секретного ключа. Это требование связано с тем, что число различных расширенных ключей может быть только равным или меньшим. Действительно, длина расширенного выходного ключа превышает длину секретного ключа, но для определенных процедур предвычислений может оказаться, что различным секретным ключам будут соответствовать одинаковые расширенные ключи.

В случае применения односторонних преобразований на этапе формирования ключей шифрования значительное сужение пространства ключей шифрования является маловероятным, однако желательно получение гарантии того, что мощность множества различных ключей шифрования равна мощности множества секретных ключей длиной $l < L$, где L – длина расширенного ключа в байтах. Последнее условие достигается, если используемые процедуры предвычислений на каждом шаге преобразований задают подстановку L-байтового блока данных M, полученного как первые L байт периодического ряда, представляющего собой многократное повторение секретного ключа. В качестве составной части алгоритма предвычислений можно использовать некоторый известный блочный шифр, используемый для преобразования сообщения M в режиме сцепления блоков шифра. При этом в качестве ключа можно использовать некоторое специфицированное значение Q. Криптограмма $C = \mathrm{E}_Q(M)$ может служить в качестве расширенного ключа. При этом выполняется также требование *псевдослучайности расширенного ключа*.

Рассмотренные выше два требования к процедурам формирования расширенного ключа представляются достаточными. При желании без труда можно предложить алгоритм настройки, удовлетворяющий некоторым другим (дополнительным) требованиям. Можно принять требование вычислительной сложности определения секретного ключа по расширенному ключу и известным процедурам предвычислений. Приемлемые алгоритмы построения расширенного ключа описаны в работах [14, 41]. Смысл использования сложных процедур настройки состоит в том, чтобы заставить нападающего отказаться от их рассмотрения и принять предположение о случайности ключа шифрования.

1.4. Управляемые операции как криптографический примитив

Управляемые операции давно привлекают внимание разработчиков алгоритмов шифрования. Одной из наиболее ранних работ, посвященных проектированию криптосистем на основе управляемых операций, является статья [78], в которой рассмот-

рено использование управляемой подстановочно-перестановочной сети. Другие попытки [105, 123] были связаны с применением управляемых перестановочных сетей в качестве криптографического примитива. Однако в предложенных схемах выбор конкретной модификации реализуемой операции осуществлялся в зависимости от секретного ключа. В таком применении управляемых операций фиксируется конкретная их модификация, которая не изменяется при шифровании большого числа блоков данных. В случае управляемых битовых перестановок мы имеем некоторую фиксированную перестановку, которая является линейной операцией, хотя и неизвестной атакующему. Детальные исследования стойкости различных вариантов криптосхем на основе управляемых операций, зависящих от секретного ключа, показало, что они не могут конкурировать с другими шифрами по производительности.

Другим типом управляемых операций являются операции, зависящие от преобразуемых данных. Их особенностью является изменчивость реализуемых модификаций, что позволяет использовать термин «переменные операции». Наиболее известными алгоритмами, использующими переменные операции в качестве базового криптографического примитива, являются итеративные блочные шифры DES [89, 113], RC5 [109], RC6 [103, 110].

Первый алгоритм использует управляемые табличные подстановки размера 4×4, реализованные как блоки подстановок размера 6×4, обеспечивающие выбор одной из четырех возможных подстановок 4×4. Однако размерность векторов, которые преобразуются этими блоками подстановок, невелика, а при увеличении размера табличных подстановок существенно возрастет как сложность выбора оптимальных табличных подстановок, так и сложность их реализации. В алгоритмах RC5 и RC6 в качестве управляемых операций применяются операции циклического сдвига на число битов, выполняемые в зависимости от преобразуемых данных. Несмотря на то, что упомянутые типы переменных операций обладают сравнительно малым числом различных реализуемых модификаций, они являются эффективным криптографическим примитивом. Таким образом, переменные операции с малым числом реализуемых модификаций оказываются более эффективными по сравнению с операциями, зависящими от ключа, хотя последние и обладают очень большим числом модификаций.

Это сопоставление приводит к идее применения операций с большим числом модификаций в качестве переменных операций [92, 97]. Наиболее детально эта идея проработана по отношению к управляемым битовым перестановкам, выполняемым над подблоками данных размером 32 и 64 бит [62, 72, 75, 82, 98]. Переход к произвольным перестановкам дал возможность существенно увеличить число различных модификаций, реализуемых переменной операцией (до $n!$, где n – длина преобразуемого вектора). Для обеспечения возможности выполнения расшифровывающего преобразования шифруемый блок данных разбивают на два подблока – управляемый и преобразуемый, которые обычно имеют одинаковый размер. В этом случае число изменяющихся модификаций ограничивается размером управляющего подблока данных и равно 2^n, где n – размер последнего. Очевидно, что перестановка, зависящая от преобразуемых данных, описывается как операция подстановки частного вида, выполняемая над всем преобразуемым блоком данных и оставляющая управляющий подблок без изменения. Эффективность переменной перестановки можно объ-

яснить тем, что это подстановка, выполняемая над всем преобразуемым блоком данных. Некоторые ее слабости (линейность суммы выходов, сохранение веса Хемминга) связаны именно с тем, что эта подстановка относится к специальному типу.

В монографии [14] приведены описание и анализ ряда скоростных блочных шифров, основанных на битовых перестановках, зависящих от преобразуемых данных. Несмотря на то, что переменные перестановки являются линейным криптографическим примитивом, их комбинирование с операциями, имеющими «небольшую» нелинейность, эффективно нейтрализуют линейный криптоанализ [62, 73, 82]. Это объясняется тем, что единственной линейной комбинацией выходов переменной перестановки является сумма всех выходных битов, а при наличии некоторой дополнительной нелинейной операции трудоемкость линейной атаки становится достаточно высокой.

Другим интересным типом управляемых операций, обладающих большим числом модификаций, являются управляемые сумматоры [9, 12], представляющие собой частный случай управляемых двухместных операций [27, 74]. Данные операции также могут быть применены в качестве переменных операций. Они представляют как самостоятельный интерес, так и для комбинирования с переменными перестановками в единой криптосистеме. В *главе 4* будет дано обобщение управляемых перестановок и построен класс управляемых операционных подстановок, реализуемых на основе подстановочно-перестановочных сетей с использованием управляемых элементов минимального размера. В *главах 5* и *6* будут рассмотрены управляемые операционные подстановки, реализуемые с использованием других типов управляемых элементов. Следует отметить, что в настоящее время имеется достаточно большое число различных типов управляемых операций, обладающих большим числом реализуемых модификаций и перспективных для использования в качестве переменных операций.

Эффективность аппаратной реализации шифров на основе переменных битовых перестановок показана в работах [38, 84]. При использовании сравнительно малых аппаратных ресурсов обеспечивается высокая производительность при реализации как в заказных, так и в программируемых СБИС [119]. Вопросы построения операционных блоков управляемых перестановок (БУП), обладающих заданными свойствами, рассмотрены в работах [10, 11]. На самом деле БУП представляют собой перестановочные сети, которые ранее были широко исследованы в области параллельных вычислений и телефонии [53, 54, 102, 124], однако криптографическое их применение требует рассмотрения некоторых других свойств таких сетей, например, линейных характеристик [2].

1.5. Аппаратная реализация шифров на основе битовых перестановок, зависящих от преобразуемых данных

Представляют интерес два следующих основных варианта аппаратной реализации алгоритмов шифрования:

- в заказных сверхбольших интегральных схемах (СБИС);

- в программируемых логических матрицах – программируемых логических интегральных схемах (ПЛИС).

Первый вариант используется для серийного производства шифраторов, поскольку в этом случае он обеспечивает более низкую стоимость единицы изделия. Другим преимуществом является более высокое быстродействие изготавливаемых специализированных СБИС. Второй вариант является предпочтительным при штучном производстве устройств шифрования. Кроме того, при разработке технологии производства заказных СБИС он применяется как предварительный для изготовления и тестирования экспериментальных устройств. Достоинствами использования ПЛИС являются малый срок разработки и возможность многократного перепрограммирования под другие алгоритмы или при модифицировании алгоритмов. Другими важными направлениями использования ПЛИС являются выполнение вычислительных экспериментов большого объема и решение криптоаналитических задач.

Аппаратную реализацию итеративных шифров осуществляют в соответствии с двумя основными типами архитектур:

- итеративной;
- конвейерной.

В первом случае схемно реализуется только один раунд шифрования. Раундовая функция шифрования используется многократно для выполнения всех раундов шифрования. Дополнительные схемные компоненты осуществляют смену раундовых ключей и другие функции, необходимые для правильного осуществления вычислений, предписываемых алгоритмом. Важнейшим достоинством итеративной архитектуры является обеспечение практически одинаковой производительности в режиме электронной кодовой книги (независимое шифрование блоков данных) и в режиме сцепления блоков шифра.

Конвейерная архитектура предполагает схемную реализацию всех раундов шифрования и возможность одновременного преобразования многих блоков данных. Обычно на выходе каждого раунда устанавливается регистр для хранения промежуточного значения преобразуемого блока данных и количество одновременно преобразуемых блоков данных равно числу раундов шифрования R. Раундовые ключи хранятся в регистре ключей постоянно при выполнении шифрования сообщения. Если схема одного раунда имеет достаточно малое время задержки, то за один такт работы устройства осуществляется преобразование сразу R блоков данных, причем один из них проходит последний раунд шифрования. Таким образом, в конвейерной архитектуре за время одного такта осуществляется преобразование одного блока данных и производительность устройства оказывается примерно в R раз более высокой по сравнению с итеративной архитектурой. Недостатками конвейерной архитектуры являются существенно более высокая стоимость реализации и невозможность сохранения высокой производительности в режиме сцепления блоков шифра.

В таблице 1.1 приводятся параметры реализации следующих шифров, основанных на переменных перестановках: CIKS-1 [92] и SPECTR-H64 [72], DDP-64 (*см. гл. 4*), COBRA-H64 [99] и COBRA-H128 (*см. гл. 8*). В таблице 1.2 дается сопоставление характеристик реализации шифров SPECTR-H64 и COBRA-H128 с широко из-

вестными криптосистемами DES [79], AES и IDEA для случая использования ПЛИС. Шифр SPECTR-H64 обеспечивает более высокую скорость при меньших затратах аппаратных ресурсов по сравнению с криптосистемами AES и IDEA. Стоимость реализации SPECTR-H64 несколько превышает стоимость реализации алгоритма DES, однако, первый обеспечивает многократное превышение по производительности. Сравнение 128-битовых шифров COBRA-H128 и AES показывает, что при примерно одинаковых аппаратных затратах первый из них обеспечивает существенно более высокую производительность.

Таблица 1.1

Характеристика аппаратной реализации шифров, основанных на переменных битовых перестановках, с использованием программируемых и заказных СБИС

Шифр	Арх-ра	ПЛИС (Xilinx Vitrex)			Заказные СБИС (0.33 мкм)		
		К-во блоков CLB*	Частота МГц	Скорость шифров. Гбит/с	Площадь, sqmil**	Частота, МГц	Скорость шифров. Гбит/с
CIKS-1	Итер.	907	81	0.648	3456	93	0.744
	Конв.	6346	81	5.184	21036	95	5.824
SPECTR-H64	Итер.	713	83	0.443	3194	91	0.485
	Конв.	7021	83	5.312	32123	94	6.016
DDP-64	Итер.	615	85	0.544	2620	92	0.589
	Конв.	3440	95	6.1	14050	101	6.5
COBRA-H64	Итер.	615	82	0.525	2694	100	0.640
	Конв.	3020	85	5.5	14640	110	7.1
COBRA-H128	Итер.	2364	86	0.917	6364	90	1.00
	Конв.	12080	90	11.5	48252	95	12.1

* CLB (Configurable Logic Blocks) – конфигурируемые логические блоки, являющиеся типовыми логическими элементами ПЛИС.

** Площадь используемой поверхности кремниевого кристалла указана в единицах sqmil; 1 sqmil = 7.45 10^{-4} мм^2

Таблица 1.2

Сравнение результатов аппаратной реализации различных шифров с использованием программируемых СБИС

Шифр	Размер входа, бит	Арх-ра	ПЛИС (Xilinx Vitrex)		
			К-во блоков CLB*	Частота, Мбит/с	Скорость шифрования Гбит/с
COBRA-H128	128	Итер.	2364	86	0.917
		Конв.	12080	90	11.5

Шифр	**Размер входа, бит**	**Арх-ра**	**ПЛИС (Xilinx Vitrex)**		
			К-во блоков CLB*	Частота, Мбит/с	Скорость шифрования Гбит/с
SPECTR-H64 [119]	64	Итер.	713	83	0.443
		Конв.	7021	83	5.312
AES [117]	128	Итер.	2358	22	0.259
		Конв.	17314	28.5	3.650
IDEA [66]	128	Итер.	2878	150	0.600
DES [79]	64	Итер.	722	11	0.181

Приведенные характеристики различных вариантов реализаций шифров, основанных на переменных перестановках, показывают, что использование операций, зависящих от преобразуемых данных, обеспечивает возможность создания высокоскоростных аппаратных шифраторов, которые могут быть изготовлены с минимальным потреблением аппаратных ресурсов.

1.6. Особенности проектирования блочных шифров на основе управляемых операций

1.6.1. Управляемые операции и отображения

Любую операцию, используемую при построении блочных шифров, можно представить как отображение векторного пространства h-мерных двоичных векторов $W = (w_1, w_2, \ldots, w_h)$ в векторное пространство n-мерных двоичных векторов $Y = (y_1, y_2, \ldots, y_n)$, где для всех $j \in \{1, \ldots, h\}$ и $i \in \{1, \ldots, n\}$ имеем $w_j, y_i \in \mathrm{GF}(2)$. Такое отображение можно записать в виде $\mathrm{GF}(2)^h \to \mathrm{GF}(2)^n$. В вероятностных шифрах используются отображения, относящиеся к случаю $h < n$, в котором выходное значение зависит от некоторого случайного значения. В таких случаях по выходному значению однозначно определяется входной двоичный вектор. При $h > n$ в общем случае нельзя определить однозначно входной вектор по выходному. Операции такого типа могут применяться при построении шифров на основе криптосхемы Фейстеля, которая для произвольной раундовой функции задает корректное построение блочного шифра, т. е. обеспечивает возможность правильного расшифрования. Операции, соответствующие случаю $h = n$ и устанавливающие взаимно-однозначное соответствие между входными и выходными векторами, задают преобразование исходного векторного пространства. Такие операции задают некоторую подстановку. Подстановками иногда называют также операции, соответствующие случаю $h > n$ (см., например, подстановки типа 6×4 в алгоритме DES).

При $h > n$ отображения можно интерпретировать как управляемые операции с размером управляющего входа равным $m = h - n$. Однако часто более удобным, наглядным и полезным для анализа оказывается рассмотрение операции как управляе-

мой, хотя всегда надо иметь в виду, что наиболее общим является описание в виде указанного отображения. Мы можем конструировать управляемые операции, исходя из тех или иных соображений и механизмов, оставаясь в каком-то частном классе отображений. При малых значениях m и n отображение можно задать табличным способом, который является наиболее общим. Но при больших значениях m и n (например, $m = 64$ и $n = 32$) табличный способ оказывается неприменимым. В этом случае можно строить некоторый генератор, который будет формировать отображения такого типа и называться операцией преобразования. В некоторых частных вариантах такой генератор можно назвать управляемой операцией.

Обычно в управляемой операции можно выделить информационный вход (или просто вход) и управляющий вход. Отображаемый вектор W длины h представляется в виде конкатенации (X, V) преобразуемого вектора X длины n и управляющего вектора длины $m = h - n$. Такие управляемые операции можно назвать одноместными, поскольку на информационный вход поступает только один операнд. Операция при фиксированном V называется модификацией управляемой операции. Если при каждом фиксированном V реализуется биективное (взаимно-однозначное) отображение пространства входных n-битовых векторов в выходное пространство n–битовых векторов, то можно говорить об управляемой операции преобразования или об управляемом преобразовании. При $2n < h$ можно говорить о двухместных управляемых операциях с размером управляющего входа $m = h - 2n$, в которых на информационный $2n$-разрядный вход поступают два n-битовых вектора. Целесообразным является синтез таких управляемых операций, множество модификаций которых можно было бы отнести естественным способом к некоторому классу. Характерным примером являются управляемые битовые перестановки.

Подход к разработке блочных шифров, опирающийся на использование переменных операций, связан с использованием управляемых операций с очень большим числом возможных модификаций, когда значение m в два и более раза превышает значение n. Однако в алгоритмах шифрования предполагается разбиение преобразуемого блока данных на равные подблоки. Обычно разбиение осуществляют на два подблока. Очевидно, что один из подблоков подлежит преобразованию (ведь мы хотим его зашифровать), а другой может быть использован для формирования управляющего вектора. Простейшим вариантом такого формирования является случай использования каждого бита управляющего подблока данных для задания нескольких битов управляющего вектора. При аппаратной реализации это реализуется простым разветвлением проводников и практически не требует затрат схемотехнических ресурсов. Блок такого разветвления будем называть блоком расширения E.

1.6.2. Расписание использования ключа

Ниже при рассмотрении нескольких типовых итеративных схем синтеза блочных шифров на основе управляемых операций мы остановимся на построениях, которые обеспечивают возможность зашифрования и расшифрования с помощью одной и той же электронной схемы. Смена режима шифрования в таких криптосистемах осуществляется изменением расписания использования ключа или простым обращением очередности использования раундовых ключей. В блочных шифрах использование

секретного ключа является важной частью криптосистемы в целом. Говорят о расписании использования ключа или просто о расписании ключа (key scheduling). Под этим термином понимаются все части механизма, определяющего вхождение ключевых элементов в шифрующие процедуры. В качестве ключевых элементов могут использоваться непосредственно некоторые части секретного ключа (подключи) или некоторые псевдослучайные значения, вырабатываемые в зависимости от секретного ключа по некоторым достаточно сложным процедурам. Данные процедуры называются процедурами усложнения секретного ключа (ПУСК), а полная совокупность вырабатываемых ими ключевых элементов – расширенным ключом. Если алгоритм формирования расширенного ключа выполняется до осуществления непосредственного шифрования данных, то говорят, что расширенный ключ формируется на этапе предвычислений.

Под расширенным ключом иногда понимают и совокупность частей секретного ключа, используемых во всех раундах или на всех шагах вхождения ключевых элементов в те или иные операции преобразования. Поскольку в этих случаях не выполняется преобразование секретного ключа или его частей, то говорят о простом расписании использования ключа. Как правило, в современных блочных шифрах используется процедура усложнения ключа, хотя российский стандарт шифрования ГОСТ 28147–89 обходится без этого, показывая пример стойкого шифра с простым расписанием ключа (ПРК). Разработка шифров с ПРК представляет интерес, поскольку при аппаратной реализации в этом случае экономятся ресурсы, которые бы потребовались для воплощения алгоритма усложнения ключа. Кроме того, при частой смене ключей ПУСК может приводить к снижению производительности шифратора. Последнее можно избежать, если ПУСК выполнять параллельно с осуществлением шифрующих процедур.

Это может быть сделано следующим образом. На первом раунде шифрования используется непосредственно какая-то часть секретного ключа. Пока выполняется процедура шифрования вычисляется ключевой элемент (раундовый ключ), используемый на втором раунде. В течение времени выполнения второго раунда вычисляется раундовый ключ для третьего раунда. При этом ПУСК должна быть такой, чтобы при зашифровании раундовый ключ, используемый в i-ом раунде, формировался таким образом, что при выполнении расшифровывающих преобразований он был бы сформирован к началу выполнения $(R - i + 1)$-го раунда. Если разрабатываемый блочный шифр предполагается использовать для построения итеративных хэш-функций, то следует учитывать следующее обстоятельство. Наличие сложной ПУСК приводит к увеличению стоимости аппаратной реализации и снижению производительности хэш-функций.

В итеративных шифрах, как правило, используется достаточно большое число раундов шифрования, при этом в одном раунде используется подключ (или раундовый ключ) длины от 32 до 256 бит. Если учесть, что длина секретного ключа обычно равна от 64 до 256 бит, то возникает проблема формирования расширенного ключа, представляющего собой совокупность всех раундовых ключей, а также ключей начального и конечного преобразования, если таковые используются в данном конкретном шифре. Различные варианты построения расписания использования секретного ключа имеют свои достоинства и недостатки. Дадим краткую характеристику применяемым подходам.

Использование предвычислений для формирования расширенного ключа позволяет обеспечить сложную зависимость раундовых ключей от секретного ключа. При этом расширенный ключ представляет собой псевдослучайную последовательность. Как правило, при использовании хорошего (криптографически сильного) алгоритма расширения ключа криптоанализ осуществляется в предположении независимости раундовых ключей. Недостатком этого подхода является снижение скорости шифрования в приложениях, требующих частой смены ключей. Кроме того, при аппаратной реализации потребляются существенные дополнительные схемотехнические ресурсы (часто превосходящие ресурсы, необходимые для реализации алгоритма шифрования).

Непосредственное использование секретного ключа заключается в использовании частей (размером 32 или 64 бита) секретного ключа в качестве раундовых ключей. Примером шифров, в которых используется такой подход, является российский стандарт ГОСТ 28147–89. Недостатком такого подхода к формированию раундовых ключей является то, что раундовые ключи являются явно зависимыми, что может быть использовано при криптоанализе. Кроме того, оценка стойкости шифра, выполняемая при его проектировании существенно усложняется необходимостью учета данного обстоятельства. Недостатком представляется также наличие большого числа слабых ключей, т. е. таких ключей, для которых процедура зашифрования совпадает с процедурой расшифрования. Несмотря на то, что доля слабых ключей чрезвычайно мала, разработчики блочных криптосистем стараются не допустить их наличия. Достоинством непосредственного использования частей секретного ключа в качестве раундовых ключей является то, что обеспечивается сохранение высокой скорости шифрования в режиме частой смены ключей и не требуется использования аппаратных ресурсов для реализации алгоритма расширения ключа.

Формирование раундовых подключей в процессе шифрования блока данных. В этом подходе при аппаратной реализации в качестве первого раундового ключа используется часть секретного ключа, а при выполнении первого раунда шифрования осуществляется формирование второго раундового подключа. При выполнении второго раунда шифрования вычисляется третий раундовый ключ и т. д. Такой ход формирования раундовых ключей имеет место как при выполнении зашифрования, так и при выполнении расшифрования. Учитывая связь между очередностью использования раундовых ключей в этих двух режимах, легко увидеть важность обеспечения формирования одинаковых раундовых ключей на i-том раунде расшифрования и $(R - i + 1)$-ом раунде зашифрования. Несмотря на то, что этот подход также требует дополнительных аппаратных ресурсов, он обеспечивает высокую производительность криптосистемы при частой смене ключей, что является важным в ряде сетевых приложений.

Преобразование подключей в зависимости от преобразуемых данных заключается в том, что части секретного ключа используются непосредственно, но перед их наложением на подблоки данных они преобразуются с помощью операций, зависящих от текущего значения одного из подблоков данных. Такое преобразование (механизм внутреннего усложнения ключа) может быть выполнено одновременно с преобразованием другого подблока данных, поэтому оно не приводит к снижению скорости шифрования, хотя обеспечивает существенное улучшение характеристик

раундового преобразования. Для этого подхода также имеет место проблема устранения слабых ключей. Данная проблема может быть решена путем построения универсальных криптосхем, в которых один раунд шифрования не является инволюцией и включает так называемую переключаемую операцию, которая является разновидностью управляемых операций. Переключаемой операцией называется управляемая операция, включающая две взаимообратные модификации и управляемая битом режима преобразования e ($e = 0$ соответствует зашифрованию, $e = 1$ – расшифрованию). Использование переключаемых операций в сочетании с механизмом внутреннего усложнения ключа представляется весьма перспективным для построения шифров с простым расписанием использования секретного ключа и являющихся свободными от наличия слабых ключей. Удачные решения в данном направлении имеют большое практическое значение, поскольку позволяют существенно упростить аппаратную реализацию и обеспечить высокое быстродействие при частой смене ключей.

1.6.3. Варианты криптосхем

В рассматриваемых ниже криптосхемах мы предполагаем, что используется ПРК. Они описывают некоторые типы общей структуры итеративного преобразования с использованием управляемых операций, условно обозначенных операционными блоками $\mathbf{F}$, $\mathbf{F}^{-1}$, $\mathbf{F_i}$ и $\mathbf{S}$. Блоки $\mathbf{F}$ и $\mathbf{F}^{-1}$ выполняют взаимно-обратные управляемые операции, т. е. при одном и том же значении управляющего вектора реализуемые ими модификации являются взаимно-обратными операциями. Блок $\mathbf{F_i}$ представляет собой управляемую инволюцию, т. е. для каждого из возможных значений управляющего вектора этот блок реализует операцию, являющуюся инволюцией. Таким образом, для используемых гипотетических управляемых операций для всех V и X выполняются соотношения

$$X = \mathrm{F}^{-1(V)}(\mathrm{F}^{(V)}(X)), \quad X = \mathrm{F}^{(V)}(\mathrm{F}^{-1(V)}(X)) \quad \text{и} \quad X = \mathbf{F_i}^{(V)}(\mathbf{F_i}^{(V)}(X)).$$

Криптосхема, представленная на рис. 1.1а, описывает один раунд шифрования, который можно представить в следующем виде:

$$(A_j, B_j) = \mathrm{Crypt}(A_{j-1}, B_{j-1}, K_j, Q_j),$$

где (A_{j-1}, B_{j-1}) и (A_j, B_j) – входной и преобразованный блоки данных, представленные в виде конкатенации подблоков одного размера, (K_j, Q_j) – j-ый раундовый ключ. Если выбрать эту схему, то дальнейшее проектирование шифра будет сводиться к разработке конкретной пары управляемых операций $\mathbf{F}$ и $\mathbf{F}^{-1}$, выбору числа раундов шифрования R и формированию расписания ключа. Данное раундовое преобразование не является инволюцией, но оно легко обращается простой перестановкой раундовых подключей K_j и Q_j:

$$\mathrm{Crypt}(A_j, B_j, Q_j, K_j) = (A'_j, B'_j);$$

$$\mathrm{Crypt}^{-1}(A'_j, B'_j, K_j, Q_j) = (A_j, B_j).$$

Легко показать, что при использовании ПРК, для которого выполняется условие $(K_j, Q_j) = (Q_{R-j+1}, K_{R-j+1})'$, где штрихом обозначен $(R - j + 1)$-ый раундовый ключ

процедуры расшифрования, шифрование является корректным, т. е. процедура расшифрования будет обратной процедуре зашифрования.

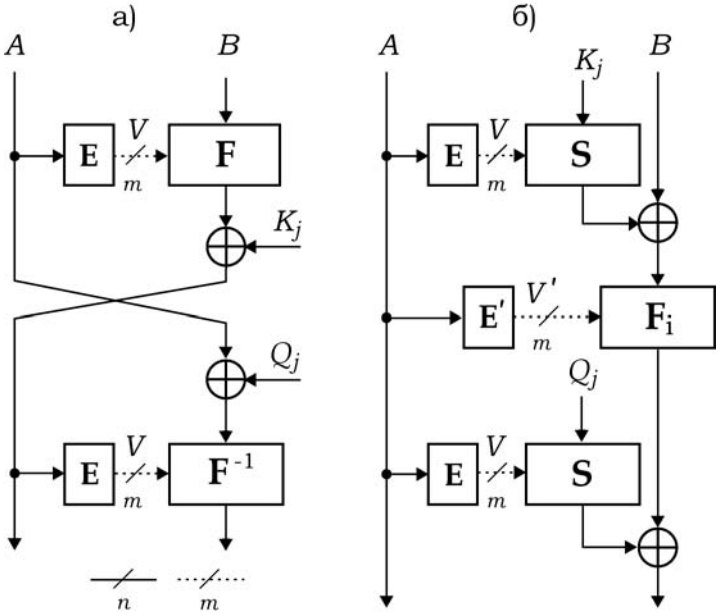

Рис. 1.1.

Один раунд шифрования, представленный на рис. 1.1б, также не является инволюцией, но легко обращается транспозицией раундовых подключей K_j и Q_j. Благодаря использованию управляемой инволюции нет необходимости выполнения двух взаимно-обратных операций при поочередном преобразовании подблоков данных, что упрощает структуру раундового преобразования. Особенностью данной криптосхемы является использование преобразования раундовых ключей с помощью управляемой операции **S**. Преобразование раундовых подключей в зависимости от преобразуемых данных можно назвать внутренним развертыванием ключа (internal key scheduling). Достоинством этой криптосхемы является возможность параллельного осуществления преобразования правого подблока с помощью операции $\mathbf{F_i}$ и подключа Q_j с помощью операции **S**. Таким образом, одна из двух операций преобразования подключей не вносит никакой временной задержки. Расшифрование выполняется таким изменением расписания ключа, при котором выполняется условие $(Q_{R-j+1}, K_{R-j+1})' = (K_j, Q_j)$. Укрупненная схема итеративного шифрования с использованием раундовых преобразований, показанных на рис. 1.1а и 1.1б, представлена на рис. 1.2а.

Возможно построение итеративных алгоритмов шифрования, в которых один отдельный раунд шифрования не является инволюцией и не может быть обращен перестановкой используемых в нем подключей. В таких случаях возможность легкой смены режима зашифрования на режим расшифрования обеспечивается использованием конечного преобразования, которое вносит необходимую симметрию в общую

процедуру шифрования. Благодаря этому становится возможным осуществить смену режима шифрования путем простого изменения расписания ключа.

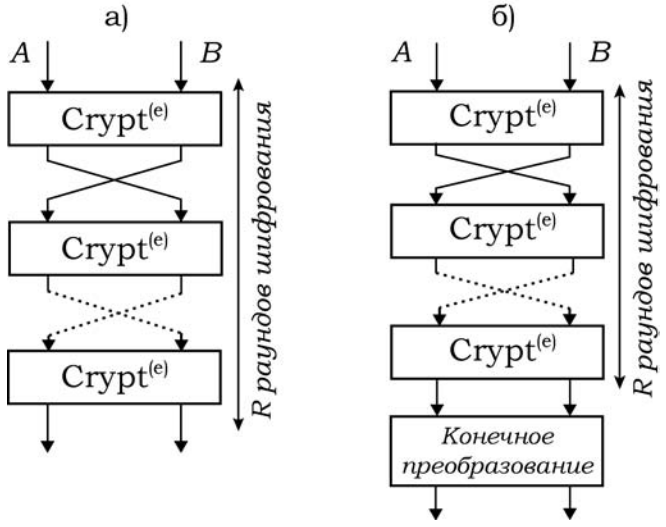

Рис. 1.2.

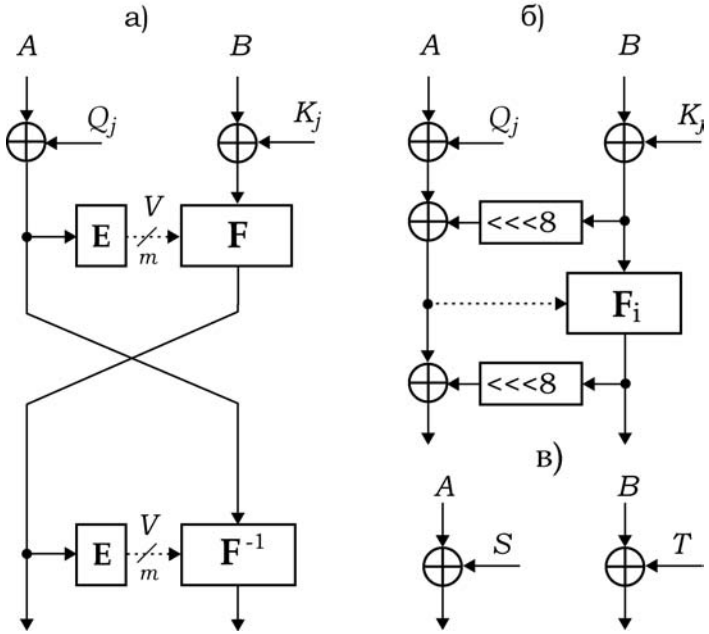

Рис. 1.3.

На рис. 1.3а и 1.3б показаны структуры раундового преобразования, которые требуют применения симметрирующего конечного преобразования. Первая схема характеризуется тем, что правый подблок преобразуется путем наложения на него подключей K_j, за которым следует выполнение прямой управляемой операции **F**, а левый – путем наложения на него подключей Q_j, за которым следует выполнение обратной управляемой операции $\mathbf{F}^{-1}$. После выполнения заданного числа раундов шифрования выполняется конечное преобразование (рис.1б.) с использованием подключей S и T:

$$(A', B') = (A_R \oplus S,\ B_R \oplus T).$$

Корректность шифрования обеспечивается таким изменением расписания ключа, при котором используемые при расшифровании подключи равны:

$$K'_1 = T;\quad K'_j = Q_{R-j+2} \text{ для } j = 2, 3, \ldots, R;\quad S' = Q_1;$$

$$Q'_1 = S;\quad Q'_j = K_{R-j+2} \text{ для } j = 2, 3, \ldots, R;\quad T' = K_1.$$

Раунд шифрования, показанный на рис. 1.3б, отличается более простым строением, что достигнуто использованием управляемой инволюции в качестве переменной операции. Корректность шифрования обеспечивается использованием указанных выше соотношений между подключами зашифрования и расшифрования:

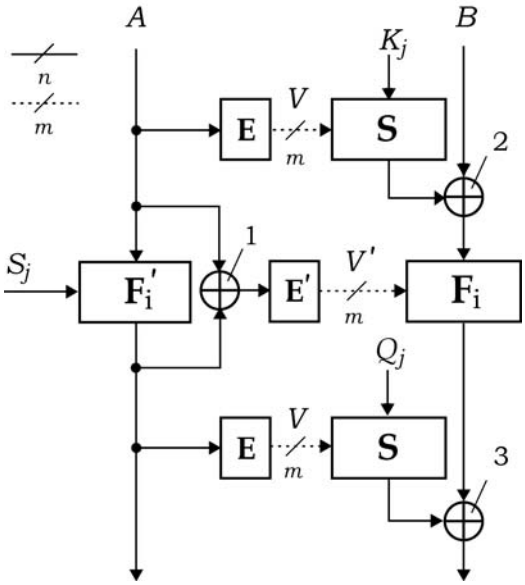

Рис. 1.4.

Представляет интерес криптосхема, показанная на рис. 1.4, где левый и правый подблоки данных преобразуются путем выполнения над ними управляемых инволю-

ций $\mathbf{F_i}'$ и $\mathbf{F_i}$, соответственно. Если временные задержки, вносимые управляемыми операциями $\mathbf{S}$, $\mathbf{F_i}'$ и $\mathbf{F_i}$, одинаковы, то при аппаратной реализации преобразование левого подблока и подключа K_j разумно выполнить одновременно, затем одновременно выполнить первую и вторую операции поразрядного суммирования по модулю два ($\oplus$), после чего выполнить параллельно преобразование правого подблока и подключа Q_j. Раундовое преобразование завершается выполнением третьей операции XOR. В результате оказываются преобразованными оба подблока данных, однако более сильному преобразованию подвергается правый подблок. Управляемая операция $\mathbf{F_i}'$, выполняемая над левым подблоком, зависит от подключа S_j. Вместо этой операции можно использовать каскад из 8 блоков подстановок размера 4×4 или из 4 блоков подстановок типа 8×8, в котором каждая подстановка является нелинейной инволюцией. Это устраняет необходимость использовать подключ S_j. Особенностью криптосхемы является суммирование входного и выходного значений операции $\mathbf{F_i}'$, что приводит к формированию одинаковых управляющих векторов операции $\mathbf{F_i}$ на соответствующих шагах процедур зашифрования и расшифрования. Для обеспечения корректности процедуры шифрования следует использовать ПРК, описываемое следующими формулами:

$$K'_j = Q_{R-j+1}; \quad Q'_j = K_{R-j+1}; \quad S'_j = S_{R-j+1}, \text{ где } j = 1, 2, \ldots, R.$$

Большое число других схем построения итеративных шифров на основе управляемых операций читатель может найти в монографии [14].

1.6.4. Этапы проектирования шифров

При проектировании шифров первостепенным вопросом является вопрос обеспечения стойкости. Этот вопрос одновременно является и наиболее сложным. Оценка стойкости является одним из наиболее длительных, трудоемких и разноплановых этапов. При этом важным является творческий подход, поскольку в настоящее время нет законченной теории блочных шифров, которая бы позволила выработать законченную методику оценивания стойкости. Однако уже предложены некоторые общие требования к качеству шифрующих преобразований. Если шифр удовлетворяет таким требованиям, то говорят о доказуемой стойкости (к известным методам криптоанализа) или о достижении гарантированных свойств шифрующих преобразований.

Необходимо иметь в виду, что достижение высокой стойкости не является самоцелью. Для криптографа, имеющего определенный опыт, не составляет труда разработать практически нераскрываемый шифр достаточно быстро как для аппаратной, так и для программной реализации. Важным является обеспечить параметры шифров (производительность, сложность реализации и др.), необходимые для конкретного применения. Эксплуатационные требования приводят к тому, что разработка подходящего шифра потребует большой искушенности и труда. Конкретные условия применения могут существенно влиять на выбор общей схемы построения шифра и на этап оценивания. Процесс синтеза блочных шифров связан со следующими этапами и соответствующими им вопросами:

1. ***Изучение области применения***. На этом этапе осуществляется выбор типа криптосистемы и формулирование требований к ее основным параметрам. Поточные шифры позволяют получить более высокие скорости шифрования и обеспечивают возможность независимого преобразования отдельных байтов и битов, что позволяет снизить влияние ошибок, возникающих при передаче зашифрованных сообщений по каналам с шумом. Однако для поточных шифров типичным является последовательная обработка знаков или битов, что затрудняет произвольный доступ к зашифрованным данным. Этот недостаток может быть устранен с использованием поточных шифров, элемент ключевой гаммы в которых вырабатывается в зависимости от секретного ключа и номера этого элемента. В таких шифрах уже имеются признаки блочного шифрования. Блочные шифры в настоящее время нашли более широкое применение. Они обеспечивают высокую стойкость в режиме независимого шифрования отдельных блоков, позволяя осуществлять произвольный доступ к зашифрованным массивам данных. В ряде средств защиты информации от несанкционированного доступа используется скоростное шифрование файлов в режиме on-line с использованием шифра комбинированного типа, имеющего признаки поточного и блочного шифрования. В подобных шифрах осуществляется независимое преобразование отдельных байтов, но они шифруются в зависимости от своего номера в файле и от специальной метки файла, вырабатываемой по случайному закону при создании файла. При выборе блочного варианта шифрования требуется выполнить обоснование длины используемого блока. С учетом областей применения определяются значения варианта реализации (программный, аппаратный, универсальный), сложности аппаратной реализации и скорости шифрования. Одновременно с выполнением шифрующих преобразований на криптографическое устройство может быть возложена задача вычисления хэш-функций от документов и сообщений. Реализация будет более экономичной, если хэш-функцию построить на основе алгоритма шифрования. Это вносит некоторые новые специфические вопросы, которые также должны быть учтены.
2. ***Выбор длины секретного ключа***. Нужно учитывать, что для любой криптосистемы с конечным ключом всегда существует возможность найти ключ методом полного перебора возможных ключей. В настоящее время для универсального случая рекомендуется длина ключа не менее 128 бит, но в отдельных случаях может применяться длина ключа 64 и даже 56 бит, если информация не представляет значительной ценности. В ситуациях, когда использование и разработка шифров с ограниченной длиной ключа (например, менее 40 бит) не ограничивается законодательно, может быть поставлена задача повышения стойкости таких криптосистем (см. раздел 1.2).
3. ***Выбор расписания использования ключа***. На этом этапе рекомендуется воспользоваться разъяснениями, приведенными в параграфе 1.6.2.
4. ***Выбор базовых криптографических примитивов и разработка криптосхемы***. Выполнение этого пункта требует знаний об основных подходах к построению шифров, типах блочных криптосистем, а также о конкретных шифрах и свойствах используемых операций, стоимости их аппаратной реализации и вносимом времени задержки.

5. ***Оценивание потребляемых ресурсов для реализации алгоритма шифрования***. Рассматриваются варианты программной и аппаратной реализации. Осуществляется реализация экспериментальных шифраторов и их программных моделей.
6. ***Оценивание производительности шифра***. Определяется значение скорости шифрования для различных вариантов программной и аппаратной реализации. Если оценки, полученные на шагах 5 и 6, не удовлетворяют значениям определенным на шаге 1, то повторяется этап 4 с учетом результатов, полученных на текущем этапе.
7. ***Рассмотрение стойкости к возможным типам криптоаналитических атак***. Рассматриваются основные варианты криптоанализа (на основе известного и специально подобранного текста), а также другие возможные варианты атак с использованием особенностей применения (инженерный анализ). Выявление наиболее эффективной атаки, трудоемкость которой и будет определять значение стойкости шифра. Разработчику полезно использовать критерий неотличимости блочного шифра от случайного преобразования. Достаточно определить трудоемкость для проведения атаки, позволяющей показать отличие шифра от случайного преобразования. Полученное значение не превышает трудоемкости вычисления ключа. Рассматривается объем необходимой памяти для выполнения различных вариантов атак и их трудоемкость.
8. ***Модифицирование алгоритма с учетом предварительного анализа***. С учетом результатов, полученных на этапе 7, осуществляется оптимизация основных узлов криптосхемы с целью повышения трудоемкости наиболее эффективной атаки. Затем выполняется предварительная оценка стойкости модифицированного алгоритма. Если необходимо, повторить этот пункт несколько раз до получения приемлемого предварительного значения стойкости.
9. ***Выполнение детального анализа модифицированного шифра***. Если детальный анализ выявил существенные слабости в модифицированной схеме, то требуется вернуться к этапу 8 или даже к этапу 4.
10. ***Выполнение статистических тестов и специальных экспериментов***. На этом шаге осуществляется программирование алгоритма шифрования или его реализация в ПЛИС и проводятся стандартные статистические тесты (см. [14]) и специальные эксперименты, планируемые с учетом результатов анализа для проверки полноты и адекватности теоретического анализа. Кроме того, спроектированный шифр (возможно уже вовлеченный в практическое использование) должен быть рассмотрен в плане возможных новых атак, основанных на последних криптоаналитических идеях. Этап анализа должен продолжаться и в процессе использования шифра.

1.7. Класс алгоритмов с выборкой подключей, зависящей от преобразуемых данных

В монографии [14] описан ряд скоростных программных алгоритмов шифрования, основанных на механизме выборки подключей (из расширенного ключа) в зави-