bhv

Михаил Масленников

Практическая криптография

+ CD-ROM

- Некоторые идеи мэтра криптографии Клода Шеннона
- Основные криптологические процедуры
- Построение качественного генератора гаммы
- Хэш-функция и электронная подпись

МАСТЕР РЕШЕНИЙ

Михаил Масленников

Практическая криптография

Санкт-Петербург

«БХВ-Петербург»

2003

УДК 681.3.06
ББК 32.81
М31

Масленников М. Е.

М31 Практическая криптография. — СПб.: БХВ-Петербург, 2003. — 464 с.: ил.

ISBN 5-94157-201-8

Книга российского криптографа посвящена прикладным проблемам современной криптографии. Наряду с основными теоретическими положениями рассматривается: создание криптографического ядра, встраивание криптографических алгоритмов в Microsoft Outlook и Lotus Notes, создание автоматизированной системы документооборота, технология отпечатков пальцев. Все программное обеспечение, описываемое в книге, создано в Borland C++ Builder. На прилагаемом к книге компакт-диске находятся демонстрационные версии некоторых программ и документация.

Для широкого круга IT-специалистов и специалистов, отвечающих за безопасность систем

УДК 681.3.06
ББК 32.81

Группа подготовки издания:

Главный редактор	*Екатерина Кондукова*
Зам. главного редактора	*Анатолий Адаменко*
Зав. редакцией	*Анна Кузьмина*
Редактор	*Петр Науменко*
Компьютерная верстка	*Натальи Караваевой*
Корректор	*Виктория Пиотровская*
Оформление серии	*Via Design*
Дизайн обложки	*Игоря Цырульникова*
Зав. производством	*Николай Тверских*

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 25.11.02.
Формат 70×100 1/16. Печать офсетная. Усл. печ. л. 37,41.
Тираж 3000 экз. Заказ №
"БХВ-Петербург", 198005, 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 Академической типографии "Наука" РАН
199034, Санкт-Петербург, 9 линия, 12.

ISBN 5-94157-201-8

Содержание

Предисловие

Все взято в трубы, перекрыты краны,
Ночами только воют и скулят.
Что надо? Надо сыпать соль на раны.
Чтоб лучше помнить, пусть они болят.

Владимир Высоцкий. "Побег на рывок"

Россия, 1992 год. Переход к рынку. Динамика роста курса доллара:

01.07.92 1$ = 125 руб.

01.08.92 1$ = 161 руб. (рост за месяц почти на 29%)

01.09.92 1$ = 205 руб. (+27%)

01.10.92 1$ = 254 руб. (+24%)

01.11.92 1$ = 398 руб. (+57%)

01.12.92 1$ = 447 руб. (+12%)

Наш родной рубль в стремительном падении. И вдруг

02.12.92 1$ = 417 руб. (−7% за день!)

31.12.92 1$ = 415 руб.

Весь декабрь рубль оставался стабильным, несмотря на проходившие в то время бурные политические события: Съезд народных депутатов, на котором было отправлено в отставку правительство Гайдара. "Рубль аплодирует правительству Гайдара! " — заголовки газет того времени.

Конечно же, на курс рубля влияет огромное множество факторов. И все же, наверное, ЦБ что-то такое предпринял. Тем более, что экс-председатель Центрального банка России Виктор Владимирович Геращенко делал доклад на Съезде, отчитывался о мерах по стабилизации финансового рынка. Откроем этот доклад и прочитаем внимательно. И в одном абзаце найдем фразу о том, что с начала декабря 1992 года во всех расчетно-кассовых центрах ЦБ РФ стали применяться **криптографические** устройства для защиты от подделок почтовых и телеграфных авизо.

Любому серьезному проекту (а оснащение ЦБ РФ криптографическими методами защиты — это более чем серьезный проект!) всегда должна предшествовать длительная и кропотливая работа: НИР, ОКР, опытная эксплуатация,

приемка, доработка, обучение персонала, составление документации и так далее. Этот процесс может растянуться на годы. А если к этому добавить традиционно огромное число больших и прочих начальников, стремящихся увековечить свое участие в проекте, бюрократизм и стремление к минимуму ответственности, то естественно встает вопрос: а как же ЦБ удалось в 1992 году, в "смутный" период, оперативно (а фактически за два месяца!) решить проблему внедрения экзотических в то время криптографических устройств и спасти народные деньги от разграбления?

Многое познается только в кризисные времена. Например, с незапамятных времен в Советской Армии были переговорные таблицы в виде толстой книги. Об удобстве их использования особо никто не задумывался: солдату прикажут — он и "Капитал" Карла Маркса в качестве переговорной таблицы будет использовать. Но вот началась война в Афганистане. Под обстрелом — не до "Капитала", тут бы побыстрее с командиром связаться и поддержки попросить, а то накроют. И пошла связь открытым текстом, пошли из-за этого человеческие потери, и тогда наконец решили придумать что-то на замену этим таблицам. Придумали.

В городе Зеленограде, нашей Силиконовой долине, есть завод "Ангстрем", выпускающий советскую электронику, в том числе и бытовые программируемые калькуляторы. А не переделать ли такой калькулятор под специализированное устройство, предназначенное заменить в СА переговорные таблицы времен Царя Берендея? Легко сказать — переделать. Вернее, сравнительно легко оказалось придумать необходимые для такого устройства алгоритмы, т. е. всю "математику". А дальше — советская промышленность того (середина 80-х годов XX века) времени. Ей подавай дефицитные в то время материалы для изготовления специализированной микросхемы, включай в план, обхаживай и ублажай, в том числе, конечно, и деньгами. А деньги откуда?

И скрепя сердце решили начальники: изготовим таких калькуляторов побольше, часть из них пустим в открытую продажу (нарождавшимся тогда коммерсантам) и заломим за них умопомрачительную цену: в те времена практически никаких криптографических устройств на рынке не было, возьмут и по такой цене. А на вырученный от продажи доход оснастим Советскую Армию. Так в 1991 году на рынке появился портативный шифратор "Электроника — МК 85 С" (рис. П1).

Тут и смутные времена подоспели. Сменилась власть в нашем Государстве, армия перестала быть Советской, да и вообще то, что было плохо вчера, стало престижным сегодня. "Учитесь жить в условиях рынка, зарабатывать деньги!" — так стали наставлять своих подчиненных начальники. Стали учиться. Показывать свою продукцию, представлять ее на выставках, рекламировать. Кто пошустрее — те и частные фирмы создавать.

Переход к рынку, как мы теперь знаем, весьма труден, а ко всеобщему — тут и говорить нечего. И пошли в нашем Государстве фальшивые, или, как

их еще называли, "чеченские" авизо. Украдено по ним было, по приведенным в газете "Аргументы и Факты" оценкам МВД, чуть ли не треть национального бюджета. Как от них защититься? Увидев на одной из выставок "Электронику МК-85С", ЦБ РФ принял решение: это то, что надо.

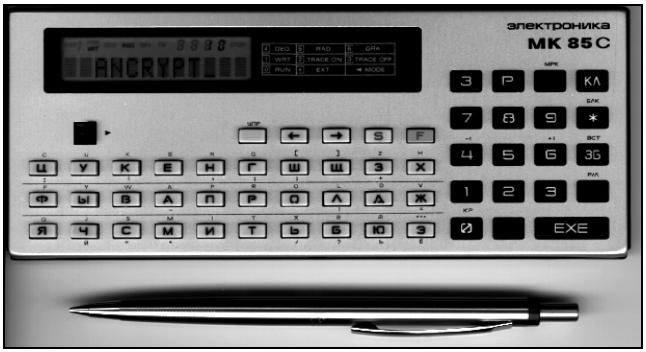

Рис. П1. Портативный шифратор "Электроника МК-85 С"

Истина — конкретна. Все в принципе может быть прекрасно, но дьявол всегда скрывается в деталях. Ведь это устройство предназначалось для шифрования информации, а ЦБ требовалась, фактически, цифровая проверочная комбинация, некое подобие цифровой подписи. А это совсем разные задачи. Кроме того, любое криптографическое устройство требует ключей, надо создать систему выработки и смены ключей, подготовить нормативные документы, обучить операционисток ЦБ, женщин, как правило ни разу в жизни ничего не слышавших ни о какой криптографии, работать с устройством и ключами к нему, подготовить программный аналог устройства на персональном компьютере. И все за два месяца. Ежу, как говорят математики, понятно, что в нашем Государстве такое может произойти только в одном случае: если число начальников, принимающих в этом участие, будет минимальным, а в идеале — стремиться к нулю. Попросту говоря, ЦБ напрямую обратился за помощью к специалистам.

Много воды утекло с той поры. В 1995 году был принят печально известный Указ № 334 Президента России, который я бы назвал криптографическим "Указом о трех колосках", откровенно ставящий целью отбить у любого нормального человека всякое желание связываться с криптографией. Хорошо это или плохо? Смотря с какой точки зрения подходить к этому вопросу. Будучи в 1992 году в самой гуще событий с ЦБ, невольно задаешься вопросом: а если некоторая аналогичная кризисная ситуация повторится, не будут ли интересы бюрократического аппарата, уютно чувствующего себя под этим Указом, ставиться превыше всего? Найдутся ли специалисты, к которым можно будет обратиться напрямую? И даже в некризисные времена в состоянии ли одно ведомство охватить все многообразие задач, в которых требуются криптографические приложения? Ведь как заметил один корреспондент

в беседе с вице-президентом Академии криптографии РФ Н. Н. Андреевым (журнал "Конфидент", N 23), "несколько смущает тот факт, что все выданные ФАПСИ сертификаты, по сути, выданы самим себе".

А между тем в мире идут как раз обратные процессы. Вот выдержка из Международного отчета "Криптография и свобода" о ситуации с криптографией, опубликованного Electronic Privacy Information Center, Вашингтон. Отчет по странам:

Соединенные Штаты

2000 ЖЕЛТЫЙ СЕКТОР/ЗЕЛЕНЫЙ СЕКТОР
1999 ЖЕЛТЫЙ СЕКТОР
1998 ЖЕЛТЫЙ/КРАСНЫЙ СЕКТОР

В течение долгого времени правительство США прилагало все усилия, чтобы ограничить разработку и распространение шифровальных технологий. За последние двадцать лет США неоднократно предпринимали такие попытки, манипулируя стандартами, рекомендуя принятие соответствующих законов и вводя экспортный контроль. Лишь недавно правительство признало, что электронный бизнес стал важным фактором американской экономики, и стало постепенно отказываться от прежней безрезультатной политики, вызывавшей негативные реакции в обществе. Наконец, в январе 2000 года американская администрация объявила о новых экспортных правилах и об ослаблении контроля над экспортом.

Франция

2000 ЖЕЛТЫЙ СЕКТОР/ЗЕЛЕНЫЙ СЕКТОР
1999 ЖЕЛТЫЙ СЕКТОР/ЗЕЛЕНЫЙ СЕКТОР
1998 КРАСНЫЙ СЕКТОР/ЖЕЛТЫЙ СЕКТОР

В 1999 г. политика в области крипто во Франции серьезно переменилась. 19 января 1999 года премьер-министр Лионель Жоспин (Lionel Jospin) объявил о коренном повороте в политике о криптографии. Эта политика отменяла существовавшую ранее сложную систему лицензирования импорта и внутреннего использования крипто, обязательную регистрацию ключей для частных пользователей и существование "доверенных" лиц государства.

Великобритания (Соединенное Королевство Великобритании и Сев.Ирландии)

2000 ЖЕЛТЫЙ СЕКТОР
1999 ЖЕЛТЫЙ СЕКТОР/ЗЕЛЕНЫЙ СЕКТОР
1998 ЗЕЛЕНЫЙ/ЖЕЛТЫЙ СЕКТОР

Сегодня в этой стране нет контроля ни за внутренним использованием крипто, ни за импортом криптографических средств. Есть лишь требование к радиолюбителям передавать свои данные в эфир в незашифрованном виде.

Великобритания всегда была самым верным соратником США в деле продвижения ограничений на шифрование. С 1996 года британское правительство (сначала консервативное, потом лейбористское) предлагало различные меры по развитию систем депонирования ключей. До сих пор эти попытки не имели успеха из-за противодействия со стороны общества и бизнеса.

Германия

2000 ЗЕЛЕНЫЙ СЕКТОР
1999 ЗЕЛЕНЫЙ СЕКТОР
1998 ЗЕЛЕНЫЙ СЕКТОР

Германия всегда была на переднем фронте в том, что касалось борьбы с ограничениями на криптографию. Она создавала противовес американским усилиям по внедрению систем депонирования ключей и международных ограничений на крипто. Германия сыграла заметную роль в создании в 1997 г. документа ЕС по криптографии и цифровым подписям. В 1999 г. благодаря усилиям Германии депонирование ключей не было включено в Вассенаарское соглашение.

Корея (Южная)

2000 ЗЕЛЕНЫЙ/ЖЕЛТЫЙ СЕКТОР
1999 ЗЕЛЕНЫЙ/ЖЕЛТЫЙ СЕКТОР
1998 ЖЕЛТЫЙ СЕКТОР

В своем отчете Организации по экономическому сотрудничеству и развитию Республика Корея указала, что в стране нет ограничений ни на импорт, ни на использование криптографии в частном секторе. Правительство рассматривает возможность введения общих правил для обращения с открытыми ключами.

Российская Федерация

2000 КРАСНЫЙ СЕКТОР
1999 КРАСНЫЙ СЕКТОР
1998 КРАСНЫЙ СЕКТОР

Импорт крипто и его использование регулируются в России следующими актами:

- Пунктом 5 Указа Президента N 334 от 3 апреля 1995 года, в котором запрещается импорт шифровальных средств без наличия лицензии.
- Пунктом 4 того же Указа, в котором запрещена любая деятельность по разработке, продаже и использованию шифровальных средств без лицензии, выдаваемой Федеральным агентством правительственной связи и информации (ФАПСИ), российским аналогом АНБ.
- Общим решением N 60 Гостехкомиссии и ФАПСИ от 24 июня 1997 года, определяющим процедуру получения лицензий на импорт и использование шифровальных средств.

По имеющимся данным, средства шифрования, не лицензированные ФАПСИ, широко доступны в России. Тем не менее, в феврале 2000 года Microsoft объявила,

что по соглашению с ФАПСИ выпускает русскую версию Windows 2000 без стойкого крипто. В декабре 1999 г. группа известных деятелей Интернет-сообщества подписала открытое письмо руководству страны, в котором призывала не устанавливать контроль над Internet, в частности, над крипто. Авторы письма заявили, что такой контроль снизит экспортные возможности российского бизнеса и уменьшит привлекательность электронного бизнеса в России для инвесторов.

Задав в поисковой системе Yandex слово "Криптография", я получил в ответ свыше 45000 ссылок. Де-факто, Россия уже пошла по пути других развитых стран и криптография перестала быть монополией спецслужб. Надеюсь, что вскоре это произойдет и де-юре. Число людей, желающих разобраться в криптографических хитросплетениях, стремительно растет. И неизбежно, вместо слепого поклонения бумаге с высочайшими подписями и печатями, криптографическим ГОСТам, отступление от которых выжигается каленым железом, у людей будут возникать вопросы:

- А какова оценка стойкости шифра в двоичных операциях?
- Надежность наилучшего метода определения ключа?
- Группа реализуемых преобразований?
- Является ли он наиболее оптимальным по скорости работы?
- Каков период вырабатываемой гаммы?
- Какова вероятность перекрытий гаммы?

И множество других подобных вопросов, ответы на которые, вместо общих фраз "стойкое нестойкое крипто", давали бы объективные характеристики того или иного криптографического решения.

Но этого мало. Сама по себе криптосхема, образующая криптографическое ядро, не гарантирует безопасности. Для нее нужна удобная система выработки, хранения и распределения ключей, система учета и контроля, система передачи ответственности, система управления и еще много других вспомогательных систем. Все это можно объединить одним словом — интерфейсная оболочка. И как показывает опыт, объем интерфейсной оболочки составляет 95—99% от общего объема законченной прикладной системы защиты компьютерной информации. Правильное построение интерфейсной оболочки, бережное и чрезвычайно деликатное обращение с криптографическими ключами (как с секретными, так и открытыми), удобство пользователя, автоматизация процедур обработки информации — все эти факторы напрямую влияют на информационную безопасность.

Я пытаюсь обобщить в этой книге весь свой 25-летний опыт работы в области криптографии. Постарался изложить его доступным языком. Надеюсь, что книга окажется полезной современному читателю.

Введение

Актуальна ли массовая популяризация теоретических основ криптографии? Этот вопрос был задан в 1998 году корреспондентом журнала "Компьютерра" заместителю заведующего лабораторией МГУ по математическим проблемам криптографии, кандидату физико-математических наук Валерию Владимировичу Ященко. "Да, вне всякого сомнения", — таков был ответ. Действительно, слово "криптография" достаточно часто можно встретить в Internet, различных газетных публикациях, статьях, специализированных журналах. Вышло несколько книг, целиком посвященных криптографии. Но все ли подобные публикации написаны квалифицированными специалистами в этой области? Вот опять же мнение В. В. Ященко: "В Internet масса сайтов, где встречается слово "crypto". Я на некоторые из них заглядывал, и у меня осталось впечатление, что это огромная мусорная свалка. И чтобы отсеять весь этот мусор от чего-то добротного, нужно иметь соответствующую квалификацию. Там масса неправильного, вредного, масса обманов. Так что, если говорить о публикации текстов, то нужно уметь четко отделить рекламный момент от содержательного. Это очень трудно. Ведь нет же никаких стандартов, нет способов проверки качества! Вот тебе дают продукт. Как ты можешь пытаться его взломать? Либо надо самому иметь высочайшую квалификацию, либо приходится довериться некоему коллективу специалистов. Потому что единственной гарантией стойкости шифра является то, что много квалифицированных криптоаналитиков пытались его вскрыть, но не смогли".

Практически всю свою сознательную жизнь — около 25 лет — я был так или иначе связан с криптографией. Сразу после школы — Технический факультет Высшей школы КГБ СССР, где готовили специалистов-криптографов. Сейчас он называется Институтом криптографии, связи и информатики, имеет свой сайт в Internet — **http://www.fssr.ru**. После его окончания — Теоретический отдел в 8 Главном управлении КГБ СССР, который занимался вопросами анализа и синтеза различных шифрсистем,

аспирантура, диссертация — все по тематике, связанной с криптографией. Мне приходилось работать в коллективе математиков-криптографов, квалификация которых без всяких сомнений может быть оценена по самым высоким мировым стандартам. Постоянные дискуссии и споры, вера только в ставшие очевидными доказательства, поиск нетривиальных решений, но вместе с тем достаточно демократичный и свободолюбивый дух общения, отсутствие слепого преклонения перед авторитетами, дружеские шутки и юмор — вот стиль работы Теоретического отдела в начале 80-х годов прошлого века. Советская криптографическая школа того периода на равных конкурировала с американцами.

Общий развал начала 90-х годов XX века не мог обойти и криптографию. Очень многие криптографы в этот период ушли в банки, коммерческие структуры, частный бизнес, стали заниматься вопросами, не относящимися к криптографии. Традиции оказались забытыми, в официальной криптографической политике воцарился чиновничий подход, далекий от криптографии как одного из разделов математики. А потребности в криптографии, наоборот, выросли. Появился Internet, электронная почта, разнообразные коммуникации, виртуальные банковские расчеты, покупки, электронный бизнес. И везде оказалась необходимой защита информации, осуществляемая с помощью криптографических методов. И не просто защита, а гарантированная защита, т. е. такая, которой не страшно было бы доверять финансовые тайны, являющиеся иногда в прямом смысле вопросами жизни и смерти.

Я попытался в этой книге сжато изложить основные идеи, необходимые для построения гарантированных криптографических систем. Книга разбита на 6 частей, в каждой части — несколько глав.

В первой части вкратце изложены некоторые сведения из истории криптографии и те основные понятия, которые наиболее часто встречаются в современной криптографической литературе.

В первой главе приведен ряд сведений из истории криптографии, в основном — Российской, за период с Первой мировой войны. Этот период, на мой взгляд, очень показателен тем, насколько важен профессиональный подход к криптографии, к каким негативным последствиям может привести пренебрежение вопросами криптографии или недостаточная квалификация криптографов.

Вторая глава посвящена результатам американского математика Клода Шеннона, который в конце 40-х годов XX века одним из первых осознал необходимость строгого математического подхода к криптографии и сумел обосновать понятие абсолютно стойкого шифра.

В третьей главе идет речь о том, осуществимы ли идеи Шеннона на практике. Здесь же даются и некоторые подробности того, как в 1992 году появилась система защиты телеграфных авизо для Центрального банка России.

Четвертая глава посвящена ключам. Что такое система с открытым распределением ключей, ее достоинства и недостатки по сравнению с традиционной ключевой системой.

В пятой главе объясняется, что такое блочный шифр, а также дается описание двух наиболее известных блочных шифров — алгоритмов DES и ГОСТ 28147-89. Я не могу скрыть своего негативного отношения к этим алгоритмам — они слишком медленны и не предназначены для программной реализации. Предлагаемый оригинальный алгоритм имеет предысторию не меньшую, чем, например, ГОСТ 28147-89, ориентирован на операции с байтами и по своей производительности значительно превосходит ГОСТ 28147-89. Вопрос о том, почему он не стал российским стандартом, с математикой и криптографией, на мой взгляд, связан очень слабо.

Шестая глава целиком посвящена одной из наиболее красивых криптографических идей — электронной подписи. Что такое хэш-функция, какие с ней связаны проблемы — обо всем этом также можно прочитать в этой главе.

Любая криптографическая идея плодотворна только тогда, когда удалось осуществить ее удобную реализацию. Проблемам реализации основных криптографических идей, изложенных в первой части книги, посвящены ее остальные части. Логика их построения — от простого к более сложному, от простых операций, связанных с шифрованием и подписью файлов к построению программных систем по защите информации, реализованных на базе Microsoft Outlook, Lotus Notes или на базе оригинальной системы электронного документооборота.

Здесь возникает много специфических и, я бы сказал, эксплуатационных проблем: как вырабатывать секретные ключи, как хранить открытые ключи, как проводить смену ключей, как распознавать зашифрованный текст, как встроить криптографические процедуры в основное меню, как автоматизировать процессы криптографической подготовки и пересылки электронных документов и т. п. Все эти эксплуатационные проблемы напрямую связаны, с одной стороны, с созданием необходимых удобств пользователям, а с другой — с информационной безопасностью. И следует отметить, что ради достижения гарантированной информационной безопасности иногда приходится идти на ограничение удобств пользователей. Если, к примеру, программная система, отвечающая за информационную безопасность, может "помочь" пользователю в случае, если он забыл пароль или потерял дискету с ключом, то точно так же она сможет оказать помощь и потенциальному злоумышленнику, взявшемуся за ее взлом. Гарантированная система доставляет и некоторые другие неудобства пользователям, например, необходимость периодической полной смены секретных и соответствующих им открытых ключей. Есть предел в компромиссах между удобством и информационной безопасностью. Задача криптографа — ясно и четко рассказать обо всем этом заказчику, а он сам принимает решение, что важнее: удобства или гарантированная безопасность. А еще одна задача криптографа —

не создавать при этом лишних, надуманных неудобств, как это иногда бывает, если к криптографии прикладывают руку чиновники от криптографии.

Нужна ли широкая публикация исходных текстов криптографических программ? Ответ отрицательный, и даже не только потому, что при этом могут быть затронуты права интеллектуальной собственности их разработчика. Дело в том, что современные профессионально подготовленные криптографические системы, как правило, практически не поддаются взлому с помощью чисто криптографических, математических методов. И потенциальный злоумышленник начинает искать обходные пути — пытается похитить секретный ключ или открытый текст. Создает так называемые криптовирусы, задача которых — обычное воровство ключа, только с помощью изощренных программных методов. И наличие исходного текста сильно помогает такому взломщику при написании криптовируса.

Но, с другой стороны, потребитель криптографической системы должен иметь некоторое углубленное представление об этой системе и особенностях ее программной реализации с тем, чтобы принять решение о том, какие секреты можно ей доверить. Разработчик вынужден немного приоткрыть тайны своих решений, но не настежь. Примерно такими соображениями я руководствовался при написании этой книги. В них достаточно много различных листингов, но они, как правило, не являются законченными криптографическими процедурами, в некоторые их параметры я сознательно внес коррективы. В основном они предназначены для разработчиков интерфейса к криптографическим системам. По собственному опыту знаю, насколько более ценным при программировании оказывается листинг с примерами, чем длинное и витиеватое описание.

Последняя, шестая часть книги затрагивает вопросы, имеющие к криптографии косвенное отношение. Это идентификация пользователя по отпечатку его пальца. Но криптографические идеи могут быть плодотворно использованы и здесь. Отпечаток пальца — это некоторая случайная черно-белая матрица, источник, по которому можно построить индивидуальный секретный ключ пользователя. А этот ключ можно использовать как в криптографии, так и для иных целей: для доступа в помещение или к каким-то ресурсам, в криминалистике — для опознания человека, для создания специализированных баз данных и т. п. Такой ключ нельзя оставить без присмотра, потерять, изменить. Потенциальный злоумышленник, помимо чисто технических трудностей, имеет в этом случае дело с задачей перебора, для которой можно дать более-менее точные математические оценки ее сложности.

В книге используется общепринятая криптографическая терминология: секретный и открытый ключ, система с открытым распределением ключей, открытый и шифрованный текст, гамма наложения, блочный шифр и т. п. Эти термины по ходу изложения поясняются, так что читатель, взявший в руки эту книгу, вполне может быть человеком, весьма далеким от проблем криптографии. Примеры листингов программ подготовлены с помощью

языка программирования Borland C++ Builder 3, Client/Server Suite. Я с ним сильно подружился за последние годы. Его русскоязычное описание можно найти, например, в книге Рейсдорф К., Хендерсон К. Освой самостоятельно Borland C++ Builder. – М.:Издательство БИНОМ, 1998, или в книге Шамис В. А. Borland C++ Builder 3. Техника визуального программирования. – М.: Издательство "Нолидж", 1998.

Я постарался изложить криптографические идеи доступным и понятным языком, не перегружая текст излишними формулами, непонятными терминами. Иногда с анекдотами и юмором. Но нас так учили в Высшей школе КГБ: "Через 15—20 минут рассказа — отвлеките внимание аудитории, расскажите ей какую-нибудь смешную историю или анекдот". После таких слов лектора вся аудитория сразу же начала отсчет времени, ибо эта была последняя лекция перед обедом. Через 20 минут с полсотни голодных глоток дружно заскандировали: "А-нек-дот! А-нек-дот!" Лектор вынужден был отпустить нас на обед.

Время пошло!

Часть I

Основы криптографии

Как я представляю себе своего читателя? Это молодой человек, лет 20—30, может побольше, интеллигентный, эрудированный, с чувством юмора. Работает программистом в банке. А может и не программистом и не в банке. Но с компьютерами тесно связан, даже, бывает, иногда пишет свои программы. Или, по крайней мере, разбирается в программировании.

И вот вызывает его однажды к себе начальник и говорит:

— Давай-ка ты, парень, напиши какую-нибудь защиту для нашей очень ценной базы данных.

А может и не для базы данных, может для системы "Банк-клиент", "Банковский операционный день" или еще для какой-то другой системы. Мало ли в банке всяких баз данных и программ, с деньгами работающих. А то, что с деньгами работает, лучше все-таки защитить от посторонних глаз.

Так ведь есть уже много готовых систем защиты! Но начальника все они почему-то не устраивают. То ли очень сложные, громоздкие, то ли слишком дорогие, то ли операционная система не та, то ли вообще начальник не с той ноги сегодня встал или по пути на работу ему черная кошка дорогу перебежала. Сделай свою, оригинальную, и все тут! Отвечает мой читатель:

— Так ведь я ничего и не слышал про криптографию, никогда раньше никаких дел с ней не имел.

— Вот иди и разберись со всеми этими шифрами, ключами и электронными подписями!

И грозно сверкнул глазами.

Не бойся, читатель! Разберемся. И начальник твой тоже разберется.

Глава 1

История криптографии

Всем читателям, интересующимся историей криптографии, я настоятельно рекомендую прочитать давно ставшую классикой книгу Дэвида Кана "Криптографы". Об этой книге я впервые узнал, будучи слушателем Технического факультета Высшей школы КГБ в середине 70-х годов. Опытные и заставшие еще сталинские времена преподаватели, рассказывая нам, молодым 20-летним парням, об истории криптографии, упоминали о том, что в спецбиблиотеке есть таинственная книга Дэвида Кана про историю криптографии, но круг допущенных к ней читателей строго ограничен "по идеологическим соображениям". Запретный плод сладок. Будучи уже офицером КГБ, получил все-таки доступ к этой таинственной книге и сразу же стал искать в ней раздел про Россию. О послереволюционной России Дэвид Кан писал: "Одной из задач тайной полиции являлась защита диктатуры пролетариата от самих пролетариев, которые не обрели обещанного счастья при новоявленных диктаторах". Да, в 70-х годах это было страшной тайной.

Давайте немного проследим историю криптографии по книге Дэвида Кана, которую в 2000 году я свободно купил на книжном рынке у Кузнецкого моста (Кан Д. Взломщики кодов. — М.: Центрополиграф, 2000), да и по многим другим источникам, появившимся в последнее время в Internet.

Криптография зародилась в глубокой древности, однако ее назначение подчас было в "усилении тайны и, следовательно, колдовской силы поминальных текстов". Как пишет Дэвид Кан, "секретность проистекала из понятного желания древних египтян заставить прохожего прочитать их эпитафии и тем самым выразить умершим благословения, которые содержались в надгробных надписях. В Древнем Египте, с характерной для него верой в загробную жизнь, количество надписей быстро выросло до такой степени, что внимание к ним прохожих пошатнулось. Чтобы возродить их интерес, писцы нарочно делали надписи несколько туманными. Они ввели криптографические знаки, дабы привлечь внимание читателя, заставить его задуматься и вызвать у него желание разгадать их смысл".

Начало криптографии, в современном понимании этого слова, связывают с Юлием Цезарем. Вот что пишет об этом в своем цикле статей "Криптография. Ее истоки и место в современном обществе" Андрей Винокуров: "Первые действительно достоверные сведения с описанием метода шифрования относятся к периоду смены старой и новой эры и описывают шифр Цезаря — способ, которым Юлий Цезарь прятал свои записи от излишне любопытных глаз. С высоты достижений современной криптографии шифр Цезаря предельно примитивен: в нем каждая буква сообщения заменялась на следующую за ней по алфавиту. Однако для того времени, когда умение читать и писать было редким исключением, его криптостойкости вполне хватало. Использование шифра решало проблему секретности передаваемого сообщения, а проблема его подлинности решалась практически сама собой".

Известен также пример другого древнего шифра — шифра Сциталла. Он положил начало так называемым шифрам перестановки. Вот цитата из курса "История Криптографии", Московский государственный университет.

"Одним из первых приборов, реализующих шифр перестановки, является так называемый прибор СЦИТАЛЛА. Он был изобретен в древней "варварской" Спарте во времена Ликурга; Рим быстро воспользовался этим прибором. Для зашифровывания текста использовался цилиндр определенного диаметра. На цилиндр наматывался тонкий ремень из пергамента, и текст выписывался построчно по образующей цилиндра (вдоль его оси). Затем ремень сматывался и отправлялся — получателю сообщения. Последний наматывал его на цилиндр того же диаметра и читал текст по оси цилиндра. В этом примере ключом такого шифра являлся диаметр цилиндра и его длина, которые, по существу, порождают двухстрочную запись, указанную выше.

Интересно, что изобретение дешифровального устройства "АНТИСЦИТАЛЛА" приписывается великому Аристотелю. Он предложил для этого использовать конусообразное "копье", на которое наматывался перехваченный ремень, который передвигался по оси до того положения, пока не появлялся осмысленный текст".

В России криптография впервые появилась при Петре I, когда он начал "прорубать окно в Европу". Как пишет Дэвид Кан, "в XVIII веке Россия переняла у Запада одно из его полезных нововведений — "черные кабинеты". Так же, как в Англии и Австрии, у русских они размещались в почтовых отделениях. В число сотрудников "черных кабинетов" входили специалисты по вскрытию конвертов и подделке печатей, переводчики и дешифровальщики".

Но наибольшую значимость криптография всегда приобретала во время войн. Так, во время первой мировой войны разгром русских армий Самсонова и Ренненкампфа в Восточной Пруссии Дэвид Кан напрямую связывает со слабостью шифровальных средств, использовавшихся в тот период в русской армии. Вот несколько цитат из его книги.

"Хотя в начале войны Россия испытывала большие трудности в обеспечении своих войск всем необходимым, в том числе и средствами связи, уже в первой половине сентября 1914 года ей удалось полностью снабдить их шифровальными средствами. 14 сентября российская ставка верховного главнокомандования отдала распоряжение о том, что все военные приказы подлежат зашифровыванию.

Принятая шифрсистема основывалась на многоалфавитном шифре цифровой замены, в котором допускалось зашифровывание нескольких букв подряд по одному алфавиту. Этот шифр представлял собой таблицу, в верхней части которой в строку были выписаны буквы русского алфавита. Сама таблица состояла из восьми строк двузначных цифровых групп, выписанных в произвольном порядке. Строки отличались друг от друга порядком расположения в них этих групп. Слева они были бессистемно пронумерованы. При зашифровывании эти строки использовались поочередно: сначала под номером один, потом два и так далее. Каждая из строк применялась для зашифровывания нескольких знаков открытого текста. Количество знаков, подлежащих шифрованию данной строкой, определялось самим шифровальщиком. Для того чтобы адресат мог расшифровать полученное сообщение, в его заголовке пять раз проставлялась цифра, соответствующая количеству знаков, которые были зашифрованы каждой из строк. Когда в процессе шифрования оператор хотел изменить это число, он вставлял в текст шифровки пятизначную группу, элементами которой была одна и та же цифра, соответствующая новому числу знаков, шифруемых одной и той же строкой. Таким образом, шифртелеграммы русской армии состояли из групп букв, зашифрованных одним и тем же алфавитом. Длина каждой группы букв определялась однозначно по пятизначной цифровой группе, состоявшей из одной и той же цифры".

Обратите внимание на описание этого шифра. Мне пришлось перечитать этот абзац несколько раз, чтобы хотя бы приблизительно представить себе шифровальную систему Русской армии 1914 года. Подчас с помощью запутанного описания разработчик шифровальной системы пытается самоуспокоиться, скрыть грубые криптографические просчеты. Так было и в этом случае.

"Уже к 19 сентября молодой одаренный начальник русского отделения дешифровальной службы Австро-Венгрии капитан Герман Покорный вскрыл эту систему и полностью восстановил все ее строки. Дело в том, что такие шифрсистемы не представляли непреодолимых преград для криптоаналитиков, поскольку в шифртексте зачастую сохранялась структура наиболее часто встречавшихся в открытом тексте слов, таких как "атака" и "дивизия", которые полностью шифровались одной строкой таблицы".

Здесь мне хотелось бы немного прокомментировать этот абзац. В каждом открытом тексте — будь то разговорный текст, телефонное сообщение, компьютерный файл — обязательно присутствует преобладание встречаемости

одних символов перед другими. Например, в русском разговорном языке наиболее часто встречающиеся буквы — это СЕНОВАЛИТР. Если усидчивый читатель не поленится и возьмется подсчитать статистику встречаемости различных букв, например, на этой странице, то наибольшее преобладание в статистике будут иметь как раз эти буквы. Предположим теперь, что мы взяли и заменили во всем тексте, например, букву С — на К, Е — на Р, Н — на И, О — на П, В — на Т, А — на О, Л — на Г, И — на Р, Т — на А, Р — на Ф. Текст получится визуально нечитаемым, но наиболее часто встречающиеся буквы будут КРИПТОГРАФ. Статистика встречаемости букв от этого не изменится. Можно даже заменить буквы разными значками, палочками и крючочками, "пляшущими человечками", как это описано у Конан Дойля в одноименном рассказе, — это не меняет статистики текста. Преобладание встречаемости одних знаков перед другими сохраняется. Почему возникает подобный эффект — об этом поподробнее мы поговорим в следующей главе.

Такой шифр в криптографии принято называть шифром простой замены. Он вскрывается достаточно просто: зная наиболее часто встречающиеся буквы языка, собираем статистику шифрованного текста и наиболее часто встречающиеся буквы шифртекста заменяем на наиболее часто встречающиеся буквы открытого текста. Для помощи используем, например, "протяжку вероятного слова", как это делал Шерлок Холмс в рассказе про пляшущих человечков, принятые в предполагаемом открытом тексте стандарты (деловая переписка, как правило, начинается со слов Уважаемый г-н, в EXE-файле компьютерной программы в начале присутствует заголовок, начинающийся с MZP, и т. п.). Единственное, что нужно для вскрытия простой замены, — знать статистику языка, на котором подготовлено открытое сообщение, и сравнительно немного шифрованного текста для набора его статистики. Так типовое сообщение на русском языке при шифровании его с помощью шифра простой замены вскрывается уже при 24 символах.

Итак, получается, что в 1914 году гибель русских армий Самсонова и Ренненкампфа поизошла во многом по той причине, что действующие в то время шифры не смогли скрыть статистику открытого текста.

С этими азами криптографии, к сожалению, нередко приходится сталкиваться и сейчас. Типичное искушение, которое испытывает программист, создающий парольную систему защиты: взять пароль, сложить его со знаками открытого текста. Пароль короткий, а знаков много. Ничего, пароль можно повторить несколько раз, пока не кончится текст. Такая, или подобная ей, система "защиты" фактически является простой заменой, только не всех знаков подряд, а находящихся друг от друга на расстоянии длины пароля, и без труда вскрывается мало-мальски грамотным хакером. Протянув вероятное слово (например, серию пробелов в начале строки), легко вычислить и сам пароль.

Продолжим читать Дэвида Кана дальше. "Примерно в это же время (25 сентября) русские впервые сменили шифр. Сами строки остались без изменений, переменился порядок выбора строк для шифрования. Новый шифр был вскрыт Покорным в течение нескольких минут: все трудности отпали, когда одна из русских радиостанций передала зашифрованную новым шифром телеграмму, переданную еще до смены шифра".

Смена ключей. До боли знакомая проблема, ведь это весьма хлопотная и нервная процедура. Представьте: вы наладили закрытый документооборот, все неурядицы улеглись, система работает без сбоев, а тут вдруг надо взять и своими собственными руками ее разломать. Отменить старые ключи. Выработать и разослать новые. Получить подтверждения их получения. Проконтролировать, чтобы все одновременно в "час X" перешли со старых на новые ключи. И наверняка в большой сети найдется разгильдяй, который что-нибудь напутает. Криптографический дух Русской армии 1914 года прямо витает над современными банками!

И все-таки смена ключей необходима. Конечно же, правильно построенная современная криптографическая система не содержит таких грубых "дыр", какие были в 1914 году, уж что-что, а статистика шифртекста в ней не имеет преобладаний. И вообще, вскрыть современный шифр чисто умозрительными методами, с карандашом и бумагой, подобно Герману Покорному, невозможно. Даже для анализа шифра современному хакеру нужен мощный компьютер и, как правило, INTERNET-поддержка. Но в жизни всякое может случиться. Операционистка в банке, работающая с шифрсистемой и недовольная своей зарплатой, может уволиться. Дискета с секретным ключом, оставленная на столе без присмотра, может попасть в чужие руки. В общем, возможна компрометация ключей. Тогда смена ключей позволяет локализовать негативные последствия, вызванные такой компрометацией.

И еще одна старая песня. Нельзя ли при смене ключей новые ключи шифровать с помощью старых? Это голос управления безопасности, стремящегося таким образом облегчить себе жизнь. А зачем тогда вообще менять ключи? Если старый ключ скомпрометирован, то новый, зашифрованный на старом, также автоматически оказывается скомпрометированным. Читайте Дэвида Кана, господа!

Вторая мировая война. "Советская шифровальная служба в основном учла плачевный опыт своей российской предшественницы времен первой мировой войны. Об этом свидетельствует следующий полный драматизма обмен радиограммами между советскими воинскими частями 22 июня 1941 г. Сразу же после внезапного нападения Германии на Советский Союз один из передовых постов Красной Армии передал по радио открытым текстом: "Нас обстреливают. Что нам делать?" На что последовал следующий ответ: "Вы с ума сошли! Почему Ваше сообщение не зашифровано?"

Во время второй мировой войны шифровальная служба Красной Армии использовала в основном коды с перешифровкой".

Прервемся в этом месте. Коды с перешифровкой — что это такое? В армии принято отдавать короткие четкие команды: "Батарея — огонь!", "Прицел 120", "Взвод — в атаку" и т. п. Для таких коротких команд изготавливают специальную кодовую книгу, где каждому слову в команде ставят в соответствие некоторую цифровую комбинацию, наиболее частым словам — покороче, редким — подлиннее. Такой код называется разнозначным. Вместо слов "атака", "дивизия" по рации можно передать короткий код, например, 032, 1458. Но если кодовую книгу использовать напрямую, без дополнительной защиты, то мы попадемся в ту же ловушку, что и Русская армия в 1914 году. Кодовая книга не скрывает статистики текста, часто повторяющиеся команды выделяются и вскрываются. Поэтому их еще и перешифровывают, т. е. накладывают на них некоторую гамму наложения. И тут опять тянется ниточка в наше время. Помните разрабатывавшееся для Советской Армии портативное шифровальное устройство "Электроника МК 85 С"? Так вот, одним из его назначений как раз и была перешифровка кодовых книг. Что это значит? А то, что в "Электронике МК 85 С" был предусмотрен режим, когда как открытый, так и шифрованный текст являются только цифрами. И кто бы мог предполагать при разработке "математики" для "Электроники МК 85 С", что именно этот режим и спасет в 1992 году Центральный банк России от фальшивых авизо! Неисповедимы пути Господни!

Вернемся к Кану. "Немецкая радиоразведка против Советского Союза была малоэффективной. В стратегическом отношении она вообще не имела ни одного сколько-нибудь заметного успеха. Немцы оказались не в состоянии вскрыть шифрсистемы, применявшиеся для засекречивания переписки высшего советского военного командования". "Явная неспособность немецких криптографов вскрыть советские стратегические шифрсистемы, с помощью которых засекречивалась самая важная информация, вынудила одного немецкого криптографа признать, что, хотя Россия и проиграла первую мировую войну в эфире, во время второй мировой войны она сумела взять реванш за свое поражение". "Шифрпереписка советских разведчиков не поддавалась дешифрованию. Большинство из них использовало стандартную для советской агентуры того времени шифрсистему, которая была триумфом шифровальной техники. Она представляла собой доведенную до совершенства старую систему, применявшуюся русскими революционерами, и объединяла в себе шифр равнозначной замены с одноразовой гаммой. В Москве обоснованно считали ее абсолютно стойкой".

Какой ценой строилась абсолютно стойкая советская криптография — об этом можно узнать из романа Александра Солженицына "В круге первом". А вот выдержка из воспоминаний Леонида Александровича Кузьмина о развитии криптографии в первый послевоенный период, опубликованных в журнале "Конфидент", N 22.

"По окончании Великой Отечественной войны наступило время, потребовавшее не меньших усилий от советских криптографов, — период "холодной войны". Между тем многие научные работники военной криптографической службы были уже демобилизованы...

В этих условиях немногочисленные теперь квалифицированные криптографы военного призыва нашли в себе мужество обратиться через голову своего могущественного и страшного шефа — Лаврентия Берия — к самому "отцу народов". Вопреки ожиданиям, обращение было услышано и советская криптография была принята под крыло самого мощного органа Советского Союза — ЦК ВКП(б). Осенью 1949 года Политбюро ЦК приняло ряд важнейших для советской криптографии решений, суть которых сводилась к следующему:

- на базе разрозненных подразделений создавалось Главное управление специальной службы (ГУСС) при ЦК ВКП(б) с одновременным выделением больших средств для его становления и развития;
- были приняты меры к привлечению наиболее сильных ученых как для выполнения оперативных задач криптографической службы, так и в роли преподавателей для подготовки новых высококвалифицированных кадров;
- для выполнения последней цели создавались Высшая школа криптографов и закрытое отделение механико-математического факультета МГУ.

Реализация этих решений за каких-то три с небольшим года позволила коренным образом изменить лицо советской криптографии.

В начале 1949 года я закончил Московский авиационный институт и начал работу в КБ знаменитого С. В. Ильюшина. В декабре того же года в ленинградском райкоме ВКП(б) я в числе других комсомольцев завода проходил один из этапов традиционной процедуры вступления в партию — собеседование со старыми большевиками. По окончании мероприятия меня неожиданно пригласили в другую комнату, где состоялся уже нетрадиционный разговор с неизвестным мне человеком средних лет. Эта беседа, резко изменившая мою судьбу, началась с обычных вопросов о семье, биографических данных, а затем перешла в другое русло:

— Нравится ли вам нынешняя работа?

— Да, нравится!

— Были ли во время учебы в МАИ трудности с математическими дисциплинами?

— Не было...

— Это очень хорошо! Вот анкета — заполните её сейчас же, а вот направление на медкомиссию, пройдете её завтра!

Продолжение этой истории последовало только через год, когда мне передали приказ от самого С. В. Ильюшина: явиться завтра в райком партии за новым назначением! Там последовало новое распоряжение: "Явитесь с паспортом в дом 21 по улице Дзержинского и позвоните по внутреннему телефону, там вас встретят".

Встретил меня тот же человек, который давал мне анкету в райкоме партии. Им оказался один из руководителей отдела кадров Главного управления

специальной службы при ЦК ВКП(б) майор М. Н. Хитров. Разговор снова короткий:

— Мы отобрали вас для учебы — в течение двух лет будете слушателем нашего вуза со стипендией 1800 рублей в месяц. Вскоре вам будет присвоено офицерское звание.

— Чему я должен учиться?

— Об этом узнаете позже. Вы, кажется, конструктор? До начала обучения остается полтора месяца, так что пока мы вас направим в наше конструкторское бюро. Там явитесь к генерал-майору Ивану Павловичу Волоску, который и введет вас в курс наших задач.

Через час я стою в небольшом кабинете перед тучным генералом, на груди которого сияет значок лауреата Сталинской премии. Генерал сразу перепоручает меня тут же вызванному заместителю — подполковнику Евстигнею Дмитриевичу Щукину, с которым мы, пройдя по коридору, заходим в одну из комнат, где стоят 4—5 кульманов и десяток столов. Он сажает меня на стул рядом с собой и тихим голосом, чтобы не мешать работающим людям, начинает разговор:

— Так вы авиаконструктор? Эта работа ценится нашим народом. Товарищ Сталин вам поставил задачу: "Летать дальше всех, выше всех, быстрее всех!" Наша же работа не афишируется, мы — криптографы, работаем в области секретной связи, с шифрами. Но и нам товарищ Сталин поставил задачу: "Читать всех, но наши переговоры и переписку читать никто не должен!" Я и мои товарищи создаём для этого специальную технику засекречивания телеграфной связи. Но об этом не говорят! Ни ваша мать, ни жена, ни один из друзей не должны знать, что теперь и вы причастны к этой работе!"

В заключение этой главы не могу не привести еще одной цитаты из Дэвида Кана. "Представляет интерес поразмышлять об успехах русской криптографии. Россия сама по себе остается загадкой, овеянной тайной из тайн. То же самое касается и ее средств связи. Одноразовые шифрблокноты обеспечивают надежную защиту для сообщений российских разведчиков, военных, дипломатов и работников тайной политической полиции. Грамотно сконструированные шифры навечно сохраняют в секрете от врагов России ее наиболее важную дипломатическую, агентурную и военную переписку. В период "холодной войны" русские сумели вскрыть шифры американского посольства в Москве. Такие подвиги свидетельствуют об их осведомленности, базирующейся на глубоком понимании шифровального дела и криптоанализа. Исходят ли эти знания из врожденной способности русских к естественным наукам, что позволило им первыми запустить искусственные спутники Земли, или же из большого опыта в области криптологии, которая исправно служила коммунистическим диктаторам в России в их борьбе за власть, или же из привычки, которая впиталась в кровь всякому жителю тоталитарного общества, на каждом шагу видеть и разгадывать секреты, или из врожденной любви славян ко всему таинственному в природе, но так или иначе русские вознесли достижения своей страны в криптологии до высоты полета ее космических спутников".

Глава 2

Стойкость шифра

Интуитивно ясно, что стойкость шифра — это его способность противостоять попыткам взлома. Что такое попытка взлома? Это попытка потенциального злоумышленника получить некоторую дополнительную информацию об открытом тексте, зашифрованном с помощью этого шифра. А что такое "информация об открытом тексте"? Как дать этому, в общем-то довольно туманному, понятию строгое математическое определение? Ответ на эти вопросы был получен американским математиком Клодом Шенноном в конце 1940-х годов. К точным математическим формулам мы перейдем чуть позже, а пока, для неискушенного в вопросах математики читателя, я позволю себе привести цитату, посвященную Клоду Шеннону из книги Дэвида Кана.

"Клод Шеннон родился в городе Петоски в штате Мичиган 30 апреля 1916 года. Поступив в Мичиганский университет, Шеннон занялся серьезным изучением электротехники и математики. Именно там у него впервые проявился интерес к теории связи и криптографии.

В Массачусетском технологическом институте Шеннон написал диссертацию, в которой содержалось множество новаторских идей, связанных с разработкой телефонных систем. Получив степень доктора математических наук, Шеннон поступил на службу в лабораторию компании "Белл", которая была заинтересована в реализации этих идей на практике.

"Во время второй мировой войны, — рассказывал Шеннон, — компания "Белл" работала над засекречиванием информации. Я тогда занимался системами связи и был назначен в несколько комиссий, изучавших криптоаналитические методы. Начиная примерно с 1941 г. исследования в области математической теории связи и теории шифров велись мной одновременно. Я трудился в обеих областях сразу, и кое-какие идеи в одной из них возникали у меня, когда я работал в другой. Я не хочу сказать, что одна из этих областей доминирует над другой. Просто они настолько тесно связаны, что их невозможно разделить". Хотя разработка обеих теорий была в основном завершена примерно к 1944 г., Шеннон продолжал уточнять полученные

результаты до 1948—1949 гг., когда они были опубликованы в виде двух отдельных статей в солидном теоретическом журнале "Bell system technical journal".

В обеих статьях Шеннона — "Математическая теория связи" и "Теория связи в секретных системах" — идеи излагаются в краткой, математической форме. Точный и выразительный стиль изложения Шеннона вдохнул в них жизнь. В результате его первая статья породила теорию информации, а вторая — теорию шифров.

Главной в работах Шеннона является концепция избыточной информации. Избыточность, по Шеннону, означает, что в сообщении содержится больше символов, чем в действительности требуется для передачи информации. Избыточность связана с излишком правил, обременяющих все языки. Второй источник языковой избыточности происходит из человеческой лености, которая заставляет людей выбирать легко выговариваемые и узнаваемые звуки. Процесс корректорской правки текста сродни криптоанализу, ибо при вскрытии шифров криптоаналитики также используют свое знание правил фонетики, грамматики, идиом, слов-функций и фонетических склонностей, которые в совокупности и придают языку избыточность.

Тут я позволю себе немного дополнить признанного мэтра криптографии. Появление компьютеров и специфической компьютерной информации (текста, набранного в текстовом редакторе, файла, содержащего рисунок, exe-файла и т. п.) еще больше обострило проблему избыточности. К упоминаемым Шенноном правилам фонетики и грамматики добавились стандартные служебные символы, добавляемые компьютерной программой, пробелы, возврат каретки и перевод строки, различные заголовки, информация о версии, Copyright, информация о пользователе и многое, многое другое. Возьмите текстовой файл, набранный в Microsoft Word, и посмотрите, сколько места в нем занимает сам текст, а сколько — служебные символы. В совокупности избыточность компьютерной информации оказалась во много раз выше избыточности простой разговорной речи.

Продолжим рассказ Дэвида Кана о Шенноне. "С чего начинается криптоанализ? При исправлении ошибки все избыточные элементы, используемые для правки, лежат в готовом виде на поверхности. В криптограмме все наоборот — они незаметны. Криптоаналитик производит подсчет частот букв криптограммы и соотносит полученные результаты с известными частотами букв предполагаемого языка, на котором записан открытый текст. Откуда у криптоаналитика уверенность в том, что частоты букв открытого текста данной криптограммы примерно совпадают с частотами эталонного открытого текста? Разве не может это соответствие нарушиться из-за различий в словарном запасе корреспондентов и в темах их переписки? Нет, не может, ибо избыточные элементы языка превалируют над остальными. Сила ума Шеннона, его огромный вклад в теорию шифровального дела выразились в открытии избыточности как основы криптоанализа. Шеннон первым сумел объяснить постоянство частот встречаемости букв, а тем самым и такое зависящее от него явление, как криптоанализ, дав возможность глубоко понять процесс аналитического вскрытия шифров".

Применение идей Шеннона на практике мы уже видели в прошлой главе, на примере Русской армии 1914 года.

А теперь — немного математики, взятой из учебника "Лекции по теории информации", МГУ, раздел "Количество информации в дискретном сообщении. Энтропия".

"Предположим, что источник сообщений может в каждый момент времени случайным образом принять одно из конечного множества возможных состояний. Такой источник называют дискретным источником сообщений. При этом принято говорить, что различные состояния реализуются вследствие выбора их источника. Каждому состоянию источника U ставится в соответствие условное обозначение в виде знака. Совокупность знаков u_1, u_2, ..., u_i, ..., u_N, соответствующих всем N возможным состояниям источника, называют его алфавитом, а количество состояний N — объемом алфавита. Формирование таким источником сообщений сводится к выбору им некоторого состояния u_i и выдачи соответствующего знака. Таким образом, под элементарным дискретным сообщением будем понимать символ u_i, выдаваемый источником, при этом в течение некоторого времени T источник может выдать дискретное сообщение в виде последовательности элементарных дискретных сообщений, представляющей собой набор символов u_i (например, u_5, u_1, u_3), каждый из которых имеет длительность t_i секунд. В общем случае необязательно одинаковую для различных i. Такая модель источника сообщений соответствует реальной ситуации, имеющей место в телеграфии ($t_i \neq \text{const}$) и передаче данных ($t_i = \text{const}$). Отдельные состояния источника могут выбираться им чаще, другие реже. Поэтому в общем случае он хранится дискретным ансамблем U, т. е. полной совокупностью состояний с вероятностями их появления, составляющими в сумме 1.

$$U = \begin{pmatrix} u_1 & u_2 & \dots & u_i & \dots & u_N \\ P(u_1) & P(u_2) & \dots & P(u_i) & \dots & P(u_N) \end{pmatrix}, \qquad \sum_{i=1}^{N} P(u_i) = 1, \tag{1.1}$$

где $P(u_i)$ — это вероятность выбора состояния u_i источником сообщений. При выдаче источником сообщений в виде последовательности элементарных дискретных сообщений, полным вероятностным описанием является вероятность совместного появления набора различных символов u_i в момент t_1, t_2, ..., t_n, где n — длина последовательности

$$P(u_i^{t_1}, u_j^{t_2}, \dots, u_k^{t_i}, \dots, u_l^{t_n}).$$[1]

[1] В нашем случае эта модель описывает множество всевозможных открытых текстов, которые могут быть затем подвергнуты зашифровыванию.

Располагая такими сведениями об источнике, можно вычислить вероятность любого отрезка сообщения длиной меньше n.[1]

В каждом элементарном сообщении содержится для его получателя определенная информация: совокупность сведений о состоянии дискретного источника сообщения. Определяя количественную меру этой информации, мы совершенно не будем учитывать ее смыслового содержания, также ее значения для конкретного получателя. Очевидно, что при отсутствии сведений о состоянии источника имеется неопределенность относительно того, какое сообщение u_i из числа возможных им выбрано, а при наличии этих сведений данная неопределенность полностью исчезает. Естественно количество информации, содержащейся в дискретном сообщении, измерять величиной исчезнувшей неопределенности. Введем меру этой неопределенности, которую можно рассматривать и как меру количественной информации. Мера должна удовлетворять ряду естественных условий, одним из них является необходимость ее монотонного возрастания с увеличением возможности выбора, т. е. объема алфавита источника N. Кроме того, желательно, чтобы вводимая мера обладала *свойством аддитивности*, заключающемся в следующем: если 2 независимых источника с объемами алфавита N и M рассматривать как один источник, одновременно реализующий пары состояний n_i и m_i, то в соответствии с принципом аддитивности полагают, что неопределенность объединенного источника равна сумме неопределенностей исходных источников. Поскольку объем алфавита объединенного источника равен NM, то искомая функция при равной вероятности состояний источников должна удовлетворять условию $f(NM) = f(N) + f(M)$. Можно математически строго показать, что единственной функцией, при перемножении аргументов которой значение функций складываются, является логарифмическая функция. Поэтому перечисленные требования выполняются, если в качестве меры неопределенности источника с равновероятными состояниями и характеризующего его ансамбля U принять логарифм объема алфавита источника

$$H(u) = \log N.$$

Легко видеть, что:

- с ростом N величина $H(U)$ монотонно возрастает;
- в случае если объем алфавита источника N равен 1, т. е. когда неопределенность отсутствует,

$$H(u) = \log 1 = 0;$$

- величина $H(U)$ обладает свойством аддитивности, поскольку

$$\log(NM) = \log(N) + \log(M).$$

[1] То есть вероятность любого открытого текста длины, не превосходящей n.

Впервые данная мера была предложена Хартли в 1928 г. Основание логарифма не имеет принципиального значения и определяет только масштаб или единицу количества информации. Чаще всего в качестве основания используют число 2, при этом единица количества информации называется двоичной единицей или битом и представляет собой информацию, содержащуюся в одном дискретном сообщении источника равновероятных сообщений с объемом алфавита равным двум. При выборе основания логарифма равным 10 получаем десятичную единицу, называемую дитом. Иногда используют натуральную единицу количества информации, называемую натом, при этом основание логарифма равно $e \approx 2{,}7$. Рассматриваемая мера количества информации может иметь лишь ограниченное применение, поскольку предполагает равную вероятность выбора источником любого из возможных его состояний.[1]

В более общем случае, когда вероятности различных состояний источника не одинаковы, степень неопределенности конкретного состояния зависит не только от объема алфавита источника, но и от вероятности этого состояния. В такой ситуации количество информации, содержащееся в одном дискретном сообщении u_k, целесообразно определить как функцию вероятности появления этого сообщения $P(u_k)$ и характеризовать величиной

$$i(u_k) = -\log P(u_k) = \log \frac{1}{P(u_k)} \qquad (1.2)$$

Теперь количество информации, содержащееся в дискретном сообщении, зависит от степени неожиданности этого сообщения, характеризуемой вероятностью его появления. Количество информации в сообщении тем больше, чем оно более неожиданно. Если источник выдает последовательность зависимых между собой элементарных сообщений, то наличие предшествующих сообщений может изменить вероятность последующего а, следовательно, и количество информации в нем.[2]

Оно должно определяться по условной вероятности $P(u_k/u_{k-1}, u_{k-2}, ...)$ выдачи сообщений u_k при известных предшествующих сообщениях $u_{k-1}, u_{k-2}, ...$, тогда количество информации

$$i(u_k/u_{k-1}, u_{k-2}, ...) = -\log P(u_k/u_{k-1}, u_{k-2}, ...) \qquad (1.3)$$

Определения (1.2) и (1.3) количества информации являются случайной величиной, поскольку сами сообщения являются случайными. Его распределение вероятностей, определяемое распределением вероятностей сообщений

---

[1] В нашем случае открытые тексты выбираются, как правило, неравновероятно.

[2] Именно так и обстоит дело, к примеру, в русском, да и в любом другом разговорном языке. Например, за буквой 'ы' не может следовать 'ь'. Наличие таких запретов помогает криптоаналитику вскрыть шифр.

в данном ансамбле для цифровой характеристики всего ансамбля или источника сообщения, используется для математического ожидания количества информации в отдельных сообщениях, называемого *энтропией*.

$$H(U) = M\left\{\log\frac{1}{P(u_i)}\right\} = \sum_{i=1}^{N} P(u_i)\cdot\log\left(\frac{1}{P(u_i)}\right) \qquad (1.4)$$

Чем больше энтропия источника, тем больше степень неожиданности выдаваемых им сообщений в среднем, т. е. тем более неопределенным является ожидание сообщений. Впервые мера (1.4) была преложена Клодом Шенноном в его фундаментальной работе "Математические основы теории связи", опубликованной в 1948 г., в которой были заложены основы современной теории информации".

Итак, Шеннон впервые вывел точную формулу для меры неопределенности открытого текста. И с точки зрения криптографии сделал фундаментальный вывод: обосновал абсолютно стойкий шифр, т. е. такой шифр, который никто и никогда не сможет вскрыть. А именно:

Определение 2.1

Шифр является абсолютно стойким, если энтропия открытого текста при условии известного шифртекста равна безусловной энтропии открытого текста.

Попросту говоря, если наличие шифртекста не дает криптоаналитику никакой новой информации об открытом тексте. Конечно же, этому условию не удовлетворяет простая замена: в ней статистика шифртекста — это просто переставленная статистика открытого текста. По шифртексту криптоаналитик в этом случае сразу же определяет места появления наиболее часто встречающихся символов в открытом тексте, т. е. сразу же получает огромную дополнительную информацию об открытом тексте.

Теперь осталось сделать последний шаг — построить пример абсолютно стойкого шифра. И это, имея формулы Шеннона, описывающие энтропию, оказалось возможно.

Определение 2.2

Если шифр получается путем наложения на открытый текст случайной и равновероятной гаммы, то такой шифр является абсолютно стойким.

Из теории вероятностей известно, что при сложении двух случайных величин, если одна из них является случайной и равновероятной, то сумма, независимо от распределения другой величины, также будет случайной и равновероятной. Если при гаммировании произвольный открытый текст (неравновероятная случайная величина) складывается со случайной и равновероятной гаммой, то шифртекст будет случайным и равновероятным.

Подставляя в формулу (1.4) значение $P(u_i) = 1/N$ для всех i, получаем максимальное значение $H(U) = \log(N)$.

Помните слова Дэвида Кана из прошлой главы о том, что "одноразовые шифрблокноты обеспечивают надежную защиту для сообщений российских разведчиков, военных, дипломатов и работников тайной политической полиции"? Это и есть пример абсолютно стойких шифров. В них пишется случайная и равновероятная гамма, и вскрыть их, не имея второго экземпляра блокнота, теоретически невозможно.

Однако в большинстве практических случаев одноразовые шифрблокноты, т. е. таблички со случайной гаммой, сразу после наложения которой они уничтожаются, неудобны. Запас гаммы ограничен, для предотвращения компрометации нужно соблюдать очень строгие правила при хранении неиспользованных блокнотов. При защите компьютерной информации применение шифрблокнотов возможно только в уж очень экзотических ситуациях, например, при смене секретных ключей для рассылки новых ключей по электронной почте. В повседневной, реальной жизни используются программные генераторы гаммы. А их уже необходимо анализировать и не только на предмет того, насколько случайную и равновероятную гамму они вырабатывают, но и с точки зрения возможности определения начальных параметров генератора по какому-то отрезку гаммы или, крайний случай, по всей гамме. Такие начальные параметры и являются секретным ключом системы шифрования.

Насколько правомерна такая задача? В каких случаях потенциальному злоумышленнику может быть известен отрезок гаммы? Давайте не забывать, что в открытом тексте могут быть достаточно длинные стандарты. Криптоаналитиков часто выручают выражения вроде: "В ответ на Вашу телеграмму от", "На Ваш исходящий № от ", "Уважаемый господин ", "С уважением" и т. п. Они позволяют определить некоторые отрезки гаммы, и если по этой информации оказывается возможным вычислить начальные параметры генератора гаммы, то сразу же вскрывается и весь остальной шифрованный текст. Например, если в качестве генератора гаммы используется линейная рекуррентная последовательность вида

$$x_{i+n} = x_i + x_{i+1} + \ldots + x_{i+n-1},$$

а секретным ключом являются первые n значений, то если хотя бы в одном месте криптоаналитику будет известно n знаков гаммы, решив систему линейных уравнений он без труда вычислит секретный ключ и весь открытый текст.

Более подробно о требованиях, предъявляемых к генераторам гаммы, мы поговорим в следующей главе, а сейчас, раз мы говорим о стойкости шифра и уже определили понятие абсолютной стойкости, подумаем о том, что следует понимать под гарантированной стойкостью.

Любой программный генератор гаммы в принципе может быть вскрыт. Число начальных параметров ограничено, поэтому всегда можно предполагать, что потенциальный злоумышленник попробует их все перебрать. Но представим себе, что он попробует сделать это, как ранее в приведенном примере, когда каждый элемент ключа — байт, а $n = 17$. Количество всевозможных вариантов перебора составит

$$(2^8)^{17} = 2^{136} = 10^{40}.$$

(Здесь и дальше мы будем пользоваться распространенным у криптографов округлением $2^{10} \approx 10^3$.)

Если ключ (17 начальных значений) выбирался случайно и равновероятно, то по теории вероятности в среднем половину из этого (10^{40}) числа вариантов надо будет опробовать, пока не попадется истинный вариант. Говорить о половине при таких значениях несерьезно, это 5×10^{39}, т. е. почти те же порядки, при округлении мы огрубили эту величину гораздо больше. Как можно наглядно представить себе эту величину?

Производительность современного компьютера примем за 10^{10} простейших операций в секунду. Тогда за 1 час = 3600 сек компьютер выполнит $3{,}6 \times 10^{13}$ простейших операций, за сутки — $24 \times 3{,}6 \times 10^{13} \approx 10^{15}$, за год — $3{,}6 \times 10^{17}$, округляем до 10^{18}. Для перебора 10^{40} значений такому компьютеру потребуется 10^{22} лет. Ясно, что это нереальная задача. Предположим, что для перебора используется не один, а несколько (например, 10^6 — миллион) таких компьютеров. Им всем, работая в параллельном режиме, для перебора потребуется 10^{16} лет. Даже с учетом постоянно растущей производительности компьютеров, ясно, что задача перебора всевозможных значений имеет разумные пределы. Отодвинем ее до неразумных (с учетом постоянно возникающих сообщений о возможности появления сверхпроводниковых, квантовых, биологических и прочих новейших технологий) и примем величину в 10^{100} как гарантированную оценку стойкости.

Сложнее с другой стороной дела. В приведенном выше примере мы видели, что, несмотря на высокую оценку тотального перебора, шифр элементарно вскрывался за счет плохо выбранного генератора гаммы. А искусство выбора и всестороннего изучения генератора гаммы и есть искусство математика-криптографа. Об этом — в следующей главе.

Глава 3

Генератор гаммы

Поскольку качественный генератор гаммы является неотъемлемой частью шифрсистемы, то давайте поподробнее рассмотрим, каким требованиям он должен удовлетворять.

Шеннон писал: "С криптографической точки зрения секретная система почти тождественна системе связи при наличии шума". В теории связи термин "шум" имеет особое значение. Под шумом подразумевается любая помеха, создающая ошибки при передаче по каналу связи. Шеннон исходит из того, что шум схож с наложением гаммы. "Основное различие между ними заключается, во-первых, в том, что преобразование при помощи шифра имеет обычно более сложный характер, чем возникающее за счет шума в канале; во-вторых, в том, что ключ в секретной системе выбирается из конечного множества, тогда как шум обычно вносится в канал постоянно и выбирается из бесконечного множества".

Понятие "конечное — бесконечное множество" весьма относительно. Конечное множество, состоящее из 10^{100} элементов, с точки зрения практических задач организации его перебора, вполне можно считать бесконечным. И задача построения качественного генератора гаммы сводится как раз к тому, чтобы накладываемая на открытый текст гамма по своим характеристикам напоминала бы "белый шум", т. е. шум, который полностью заглушает все осмысленные параметры открытого текста. А принципиальное отличие от традиционного шума в том, что генератор гаммы должен уметь повторять с точностью до единого знака этот шум, с тем чтобы была возможность расшифровать зашифрованный ранее текст. А это, как правило, невозможно осуществить без использования такого традиционного математического аппарата, как рекуррентные последовательности, т. е. такие, в которых каждый последующий знак вычисляется как некоторая постоянная функция от n предыдущих. В примере из прошлой главы в роли такой функции выступала сумма n предыдущих знаков. На этом примере мы смогли убедиться, что не любая рекуррентная последовательность пригодна для

использования в качестве генератора гаммы. Давайте рассмотрим этот и некоторые другие примеры несколько подробнее.

Поскольку в этой книге речь, в основном, идет о защите компьютерной информации, то в качестве алфавита открытого и шифрованного текста у нас часто (но не всегда) будет выступать вся таблица ASCII-символов, т. е. множество всевозможных целых неотрицательных чисел от 0 до 255 с определенными на нем операциями сложения и вычитания по модулю 256. Такую конструкцию в математике еще называют кольцом вычетов по модулю 256 и обозначают как $Z/256$.[1]

$$x_{i+n} = x_i + x_{i+1} + \ldots + x_{i+n-1}$$

Предположим, что в качестве начальных значений этого рекуррентного соотношения выбраны все нули. Тогда совершенно очевидно, что вся вырабатываемая гамма также будет состоять из одних нулей, т. е. получается, что у такого генератора гаммы есть потенциально опасный ключ. Такую точку в теории рекуррентных последовательностей называют изолированной, а соответствующий ключ можно назвать критическим.

Предположим теперь, что все начальные значения — нечетные числа и n — нечетно. Тогда, очевидно, все значения гаммы наложения также будут нечетными. После наложения такой гаммы на открытый текст все четные знаки открытого текста станут нечетными, а нечетные — наоборот, четными. По четности знаков шифртекста мы сразу же определяем четность знаков открытого текста, и выработанная гамма заведомо не удовлетворяет условиям Шеннона. Внимательному читателю предлагаю самому представить, что будет в том случае, когда все начальные значения рекуррентного соотношения — четные (независимо от четности n).

Аналогично, в случае, если все начальные значения гаммы кратны 3, 5, 7 либо любому другому простому числу, то все знаки вырабатываемой гаммы также будут кратны этому числу.

Таким образом, среди всевозможных ключей в таком генераторе заведомо есть криптографически плохие, т. е. вырабатываемая ими гамма не удовлетворяет условиям Шеннона.

Предположим теперь, что $n = 5$ и начальные значения x_0, x_1, x_2, x_3, x_4 равны:

$x_0 = 32$;

$x_1 = 32 + 128$;

---

[1] Замечание "не всегда" связано с тем, что иногда приходится сталкиваться с проблемой отображения информации на экран, например, при работе с текстовым редактором. Там среди всевозможных ASCII-символов приходится отбирать только те, которые можно отобразить на экране и которые не вызовут каких-то конфликтов при работе программного обеспечения. Я предпочитаю в качестве таких символов использовать латинские буквы и цифры.

$x_2 = 32;$

$x_3 = 32 + 128;$

$x_4 = 32.$

Тогда

$x_5 = 32 + 128;$

$x_6 = 32;$

$x_7 = 32 + 128;$

$x_8 = 32;$

и т. д.

Здесь мы столкнулись с таким криптографически важным понятием, как повторяемость, и частным случаем повторяемости — периодичностью знаков гаммы. Этот пример сродни тому случаю, когда программист, строя собственную систему защиты, использует периодически повторяющийся пароль для наложения его на открытый текст, а это, как правило, первое, что приходит в голову человеку, неискушенному в вопросах криптографии. О повторяемости гаммы мы поговорим позже, сейчас же отметим, что при выборе генератора гаммы один из первых вопросов, который должен задать себе разработчик, — как избежать описанных ранее и подобных им казусов и обосновать какие-то свойства генератора гаммы? Такое обоснование — это весьма трудоемкий и сложный процесс, требующий знаний из алгебры, теории полей, теории подстановок, теории линейных рекуррентных последовательностей, а подчас просто интуиции и опыта криптографа. Описать все возможности даже в первом приближении в рамках этой книги не представляется возможным, да я и не ставил себе такой задачи. Но некоторые вещи можно обсудить.

1. Если посмотреть на пример, то легко увидеть, что одной из его особенностей является неправильный выбор ключевых параметров. В качестве секретного ключа мы выбрали начальное заполнение, от которого затем раскручиваются все остальные знаки. В процессе выработки очередного знака в нем принимают участие только n предыдущих. Поэтому если, в силу наличия стандарта в тексте, мы сумеем определить n подряд идущих знаков гаммы, то все следующие за ними будут вычисляться автоматически. В этом примере ключевые параметры хотя и принимают участие в выработке каждого знака гаммы, но это участие зависит от предыдущих знаков выработанной гаммы, что и привело в конечном счете к возможности вычисления последующих знаков гаммы по предыдущим. Поэтому ключевые параметры должны принимать участие в выработке каждого знака гаммы *независимо* от выработанных предыдущих знаков.
2. Криптоаналитику значительно облегчило работу то обстоятельство, что выбранная рекуррентная последовательность оказалась *линейной*. Легко

составляется и решается система уравнений, уравнения дают хорошие (с точки зрения криптоанализа) результаты при сложении или вычитании различных знаков гаммы, например, такие:

$$x_{i+n} - x_{i+n-1} = x_i + x_{i+1} + \ldots + x_{i+n-2}$$

$$x_{i+n+1} - x_{i+n} = x_{i+1} + x_{i+2} + \ldots + x_{i+n-1} =$$

$$= x_{i+n} - x_{i+n-1} - x_i + x_{i+n-1} = x_{i+n} - x_i.$$

Линейность надо убрать. Но представим, например, что, убрав линейность, мы выбрали функцию

$$x_{i+n} = x_i \times x_{i+1} \times x_{i+n-1}.$$

Это означает, что любое начальное заполнение, в котором есть хотя бы один ноль, будет давать полностью нулевую гамму. И не только это. Если, например, $n = 8$, то начальное заполнение, в котором все значения равны 2, будет давать тоже полностью нулевую гамму, так как мы выполняем все операции по модулю 256.

В алгебре есть понятие *подстановки*. В случае модуля 256 это просто таблица со всеми значениями от 0 до 255, но как-то перемешанными. Всего можно построить 256! всевозможных подстановок по модулю 256. Множество всевозможных таких подстановок принято называть *симметрической группой* S_{256}. Давайте посмотрим, как изменится линейное уравнение, если к нему добавить некоторую подстановку π

$$x_{i+n} = \pi(x_i + x_{i+1} + \ldots + x_{i+n-1}).$$

Нетрудно видеть, что вычитание различных знаков гаммы при этом уже не будет давать такого простого результата, как в случае, когда функция была линейной. Несомненно, что использование подстановки значительно усложнит задачу криптоаналитику. В то же время скорость реализации такого преобразования практически не уменьшится, поскольку замена значения по подстановке — это одна операция обращения к памяти.

3. А что если в приведенном выше примере сделать ключевым параметром подстановку? На предварительном этапе, один раз перед сеансом выработки гаммы, мы вычисляем случайную подстановку в зависимости от секретного ключа. На скорости реализации гаммы это практически не скажется. Зато задача анализа такого генератора, при неизвестной подстановке, намного усложнится.

4. Этот генератор работает достаточно быстро. А что если использовать не все знаки подряд, а через несколько? Это еще больше усложнит работу криптоаналитика.

Это — основные идеи. Конкретная реализация зависит от фантазии разработчика, его вдумчивости, опыта и целей. Однако могу заверить, что таким

путем можно построить криптографически качественный и высокоскоростной генератор гаммы, значительно превосходящий по скорости выработки гаммы алгоритмы типа DES и ГОСТ.

Теперь — о другой страшной опасности: перекрытиях гаммы.

Что такое наложение гаммы на открытый текст? Это, обычно, процесс познакового сложения открытого текста и гаммы. Например, если открытый текст обозначить как a_1, a_2, ..., a_n, гамму — x_1, x_2, ..., x_n, то шифртекстом будет последовательность $s_1 = a_1 + x_1$, $s_2 = a_2 + x_2$, ..., $s_n = a_n + x_n$. Предположим, что с некоторого места k гамма начала повторяться, т. е. $x_1 = x_k$, $x_2 = x_{k+1}$. Отсюда следует, что

$$s_1 - s_k = a_1 - a_k,$$

$$s_2 - s_{k+1} = a_2 - a_{k+1},$$

...

Таким образом, разность знаков шифрованного текста равна разности знаков открытого текста. Появляется возможность *бесключевого чтения*: предполагая некоторое вероятное слово или фразу в открытом тексте, определяем открытый текст через k знаков и по его читаемости принимаем решение, верно наше предположение или нет. Критериев читаемости или нечитаемости текста может быть множество, например, если речь идет о русском языке и в вычисленном тексте встретилась запретная для русского языка пара ЫЬ, то такой текст признается ложным.

Теперь нетрудно заметить, что точно такой же эффект будет и в том случае, если, использовав один раз генератор гаммы и осуществив с его помощью зашифровывание открытого текста, мы попытаемся точно такой же гаммой зашифровать другой открытый текст. Начинают срабатывать те же методы бесключевого чтения. Сам генератор гаммы может быть сколь угодно хорошим, вырабатываемая им гамма удовлетворять всем условиям Шеннона, но для определения открытого текста по шифрованному нам не потребуется знание гаммы. Предполагая наличие некоторого стандарта в одном тексте, проверяем читаемость вычисленного другого открытого текста и принимаем решение об истинности или ложности сделанного предположения.

Отсюда следует один из наиболее фундаментальных принципов построения качественного генератора гаммы.

> ***Определение 3.1***
>
> Всякий раз гамма, вырабатываемая генератором гаммы, должна быть новой, отличной от всех предыдущих гамм, выработанных этим генератором.

Что это означает на практике? Всякий раз, перед началом выработки гаммы наложения, необходимо в параметры, влияющие на выработку гаммы, добавить некоторый дополнительный параметр, называемый в разной литературе

по-разному: разовым ключом, маркантом, ключом на телеграмму и т. п. Мне больше нравится слово *маркант*, поскольку этот параметр, как таковой, чаще всего *ключом не является*. Вы должны сообщить его тому, кто будет расшифровывать текст, чтобы он смог повторить, имея секретный ключ, точно такую же гамму. Маркант иногда просто дописывается *в открытом виде* в шифрованное сообщение и пересылается вместе с шифртекстом адресату. А алгоритм выработки гаммы надо строить с тем учетом, что маркант известен потенциальному злоумышленнику. Маркант чаще всего строится по текущему моменту времени, в который осуществляется зашифровывание, хотя возможны и другие варианты, например, когда сам пользователь, нажимая на клавиши, вырабатывает случайный маркант. Единственное требование: маркант практически никогда не должен повторяться. Повторение марканта равносильно повторению гаммы и бесключевому чтению.

Вот такие основные правила должен соблюдать разработчик, создавая собственный генератор гаммы. Но и это еще не все.

Предположим, что мы зашифровываем банковскую информацию. Например, сумму в 1 000 000 рублей в банковском платежном поручении. И в отличие от предыдущих рассуждений, будем предполагать, что открытый и шифрованный тексты состоят только из цифр, а сложение осуществляется по модулю 10.

Пусть, например, для зашифровывания этой суммы наш генератор гаммы выработал последовательность 5017329. Тогда шифртекст — сумма открытого текста и гаммы — будет 6017329 и этот шифртекст был послан в канал связи, причем такой, к которому потенциальный злоумышленник имеет доступ. Ключа к шифру злоумышленник не знает и при правильно построенном генераторе гаммы вычислить не может. Но это ему в некоторых случаях и не нужно. Предположим, что злоумышленник имеет возможность подменять шифртекст в канале связи (например, при пересылке платежного поручения по телеграфу, почти как по Пушкину: "и в суму его пустую суют грамоту другую"). Воспользовавшись этой возможностью, злоумышленник подменит шифртекст 6017329 на 7017329.

Что произойдет на приемном конце? Получив платежное поручение с зашифрованной суммой — 7017329 — операционистка, обладая ключом к выработке гаммы, выработает ее: 5017329, и произведя расшифрование, т. е. вычитание гаммы из шифртекста, получит сумму 2 000 000, которую и зачислит на счет получателя. Нетрудно предположить, что получатель и будет тем злоумышленником, который, подменив шифртекст, не зная ключа к шифру, тем не менее сумел получить таким образом лишний миллион рублей.

Следовательно, при использовании шифра гаммирования подмена некоторых знаков в шифртексте может быть не обнаружена при расшифровании. В криптографии свойство шифра обнаруживать подмену некоторых знаков

в шифрованном тексте называют *имитостойкостью*. И разработчик должен помнить, что шифр гаммирования не обладает имитостойкостью.

Начало сентября 1992 года. Старое здание ЦБ на Неглинке. Комната-пенал, два стола вдоль стены, никаких излишеств.

— Нам нужна Ваша помощь.

Это говорит, вставая из-за стола у окна, высокий стройный человек с седой и густой шевелюрой. Спокойный, уравновешенный, без начальственных привычек. Наверное, раньше был инженером, технарем. А в ЦБ я, действующий офицер ФАПСИ, 35-летний подполковник, без ведома большого начальства. В Конторе какие-то очень уж непонятные времена начались. Сразу после августовских событий 91-го года, как только Ельцин победил, буквально на следующий день нашего начальника Главка завалили рапортами об увольнении. В моем отделении едва ли не треть ребят подали рапорта. И все — молодые, не закостенелые, кому до смерти надоели эти постоянные разговоры ни о чем в курилках да игры на компьютере. Те, кто хочет еще как-то двигаться, а не просто ждать пенсии. Визировал их, как зам. начальника отделения, без сожаления. Сам тоже твердо решил: протяну как-нибудь до 1994-го года, там будет 20 лет выслуги, пенсия, и — сразу на свободу. Начальники тоже как-то притихли. В январе 1992 года, на ежегодном отчете нашего отделения, прямо сказали: "Учитесь зарабатывать деньги, развивайте коммерцию". Позаключали разных договоров. И вот в начале 1992 года — новая метла, новый самый большой начальник. Прозвали его "папой". В мае — очередная крутая смена курса. "Всякую коммерческую деятельность запретить, все договоры разорвать!" Ну, разорвать так разорвать, офицеру что прикажут — то и сделает. Но на душе как-то муторно. А когда будет очередной прогиб Генеральной линии? Через полгода, год, два? И в какую сторону?

— Надо защитить народные деньги.

Ну, это по моей части! Система электронной подписи к тому времени уже готова, и даже более того, готова комплексная программа "Криптоцентр", в которой наворочено все, что душе угодно: подпись, шифрование на индивидуальном секретном ключе, шифрование с использованием открытых ключей, журнал учета, помощь. С собой экзотический в те времена Notebook. Ставлю его на стол, показываю "Криптоцентр". Все в диковинку, смотрят внимательно, но что-то не особенно заинтересованно.

— Понимаете, это хорошо, но нам сейчас нужно не то. У нас около 2000 РКЦ, компьютеров почти нигде нет. Связь, в основном, по почте и телеграфу. А в Сибири есть и такие, до которых 3 дня на лодке надо плыть.

"Широка страна моя родная", что и говорить. Да, наверное они правы: "Криптоцентр" тут не всем подойдет.

— Нам бы как-нибудь ваш калькулятор приспособить и полгодика продержаться. А там что-нибудь придумаем.

Калькулятор — это "Электроника МК 85 С". В его "математике" не ГОСТ-монстр, а человеческий регистр сдвига. С ГОСТом он бы помер еще до рождения — слишком медленный этот брат-близнец американского DESa. Сколько людей написали диссертации на регистрах сдвига! Вот только реализация — советская. На каких трех китах держался мир социализма? На русской экономии и бережливости, на польском трудолюбии и на монгольской электронике.

— Авизовка — это платежное сообщение, которое затем передают по почте или обычному телеграфу. Нам бы туда добавить каких-нибудь проверочных знаков, которые будут на вашем калькуляторе вычисляться, но не больше 10, больше наши технологии не позволяют.

Чуть не падаю со стула. Не больше 10! Да там один маркант — 10 знаков! Сказали тоже — проверочная комбинация. Шифр гаммирования не является имитостойким, это нам еще на 4-м курсе Высшей школы твердо втолковали.

— У нас каждая операционистка в день может обрабатывать до трехсот авизовок.

Однако! Почище, чем в армии. Там, дай бог, десяток команд, да и то не каждый день. В общем, все ясно! Глухо дело. Не подойдет для них калькулятор.

— Ну, Вы подумаете, как можно нам помочь?

Вежливо прощаемся. Чисто символически, обещаю подумать. Все ясно — чего там думать! Не подойдет им калькулятор. 10 знаков! Тут электронная подпись нужна. Хотя в 10 знаков никакая электронная подпись не влезет — для нормальной подписи около 100 знаков надо. Нечего и голову ломать, напрасно терять время — пусть пишут ТЗ, открывают НИР, ОКР, года через 2 может чего от "папы" и получат. А я человек маленький, мне какое дело до их проблем!

"Бакс" уже за 200. Интересно, а его рост — это ограниченная функция? Если да, то какой величиной? За пару месяцев — почти вдвое! Если за каждую пару месяцев — вдвое, то к Новому году будет за 500. Это еще недавно была моя месячная зарплата в Конторе.

В "АиФ" статья Дунаева, замминистра МВД. Примерно треть бюджета украдена по фальшивым авизо и чекам "Россия". Воруют!

"Какой-то червяк начинает глодать". Ведь это же ЦБ, огромная сеть, в прямом смысле вся страна. Вот было бы внедрение диссертации на всю страну! При защите справку о внедрении, естественно, из пальца пришлось высасывать. Новое направление. Результаты. Это вам не DES в стандарты протаскивать, бумажки писать. Тут одни логарифмические подстановки чего стоили — построен предельный случай. Эту красоту бы в ЦБ! Но шифр гаммирования не является имитостойким, может те, кто делает фальшивые авизовки, этого и не знают, но в Конторе-то уж — наверняка! И там не простят ошибки!

А может взять какую-нибудь кодовую книгу, как в армии, что-то по ней закодировать, потом перешифровать на калькуляторе. Вот смеху-то будет. Назад, в Мазурские болота! ЦБ взяло на вооружение шифр русской армии 14-го года! Бред какой-то. А видно, им это очень надо. И срочно. Иначе пошли бы к "папе".

А что вообще-то есть там, в этом калькуляторе? Шифр гаммирования. Вводишь открытый текст, 10 раз жмешь на кнопку-генератор марканта, получаешь шифртекст. В цифровых пятизначных группах. Да если на каждую авизовку девочка будет по 10 раз нажимать кнопку марканта, 300 авизовок в день, то это 3000 нажатий на одну советскую кнопку в день! Этот день уж точно будет последним для этой кнопки.

А сейчас там, в этом ЦБ, есть что-нибудь? В смысле защиты? А как же! Еще со времен не то Николая II, не то Иосифа Грозного. Маскировка. Примерно такая же, что слово "доллар" словом "бакс" замаскировать. Никто ни в жизни не догадается! Вась, что мне всю ночь лимон с ножками снился? А, так вот кто вчера мою канарейку в чай выжал. Вот взять и перешифровать на калькуляторе эти "лимоны" и "канарейки". Круто! Что я знаю о шифрах? (Это так можно число π запомнить: по количеству букв каждого слова в этой фразе — 3,1416.) В них нежелательны стандарты. Предполагаешь некоторое слово в открытом тексте, определяешь по нему гамму, под эту гамму подставляешь, например, вместо лимона арбуз — и пошла авизовка, что шифровали, что нет, все едино. Засмеют, по крайней мере. А то и побьют. Шифр гаммирования не является имитостойким.

Ну ладно. Рассуждаем логически. Нужно использовать классические способы электронной подписи. Вычисляем хэш-функцию. Легко сказать — вычисляем хэш-функцию! А как? Руками? Бабушка из сибирской Панкрушихи руками вычисляет хэш-функцию. Заточила поострее карандашик, взяла листочек бумаги, обматерила того, кто все это придумал, и — пошла писать губерния! Через полчаса может что и вычислит. С ошибками. И все платежи враз встанут, как наш бронепоезд, в тупик на запасном пути. Калькулятор-то ведь не приспособлен ни для какой хэш-функции. А потом, что с ней делать, с этой хэш-функцией? Шифровать ее? Так ведь супостат быстрее бабушки сможет точно такую же функцию вычислить. И подставить свою вместо бабушкиной под гамму. Все. Нет никакой защиты. Ну нет имитостойкости у шифра гаммирования!

Совершенно тупиковая задача. И если б можно было ее на компьютере сделать — нет проблем! Придумать алгоритм выработки кода, подтверждающего подлинность авизовки, — милое дело. В хэш-функцию ввел ключевой параметр, хэш-функция вся просчитана, запрограммировал — готово! Но в Панкрушихе и Ребрихе нет компьютеров и долго еще не будет. Да и свет там, бывает, отключают. И обучить бабушку работе на компьютере Да, калькулятор им больше подойдет. И наштамповали их целую кучу для армии — девать теперь некуда. Будь она неладна, эта имитостойкость!