КРИПТОАНАЛИЗ И КРИПТОГРАФИЯ: ИСТОРИЯ ПРОТИВОСТОЯНИЯ

С.М. Авдошин,
кандидат технических наук, доцент, руководитель отделения программной инженерии, заведующий кафедрой «Управление разработкой программного обеспечения» Государственного университета – Высшей школы экономики,
Адрес: 105118, Москва, Кирпичная 33/5, офис 532, Авдошин С.М.,
Тел: +7(495) 771-32-38 + 5151. E-mail: savdoshin@hse.ru
А.А. Савельева,
преподаватель кафедры «Управление разработкой программного обеспечения» Государственного университета – Высшей школы экономики,
E-mail: asavelieva@hse.ru

В статье рассмотрены основные этапы развития криптологии – науки, объединяющей криптографию и криптоанализ и ставшей в эпоху компьютеризации одной из наиболее активно развивающихся областей знаний. Показано, каким образом достижения в области взлома шифров влияют на прогресс в области криптографии, а успехи криптографов становятся катализатором для криптоаналитических исследований.

Ключевые слова: Криптоанализ, стойкость шифра, взлом, криптография, квантовые вычисления.

«Все любят разгадывать других, но никто не любит быть разгаданным», – эти слова французского писателя-моралиста Ларошфуко как нельзя лучше отражают сущность современной криптологии как соревнования методов криптографии и криптоанализа. Появление новых криптографических алгоритмов приводит к разработке методов их взлома. Результатом возникновения каждого нового метода криптоанализа является пересмотр оценок безопасности шифров, что, в свою очередь, влечет необходимость создания более стойких шифров. Таким образом, исторические этапы развития криптографии и криптоанализа неразрывно связаны.

Попытка криптоанализа называется *атакой*. Результаты криптоанализа могут варьироваться по степени практической применимости. Так, криптограф Л. Кнудсен предлагает следующую классификацию успешных исходов криптоанализа блочных шифров в зависимости от объёма и качества секретной информации, которую удалось получить:

- *полный взлом* – криптоаналитик извлекает секретный ключ;
- глобальная дедукция – криптоаналитик разрабатывает функциональный эквивалент исследуемого алгоритма, позволяющий зашифровывать и расшифровывать информацию без знания ключа;
- частичная дедукция – криптоаналитику удаётся расшифровать или зашифровать некоторые сообщения;
- информационная дедукция – криптоаналитик получает некоторую информацию об открытом тексте или ключе.

Однако взлом шифра совсем не обязательно подразумевает обнаружение способа, применимого на практике для восстановления открытого текста по перехваченному зашифрованному сообщению. В научной криптологии другие правила. Шифр считается взломанным, если в системе обнаружено слабое место, которое может быть использовано для более эффективного взлома, чем *метод полного перебора ключей* («brute-force approach»). Допустим, для дешифрования текста методом полного перебора требуется перебрать 2^{128} возможных ключей; тогда изобретение способа, требующего для дешифрования 2^{110} операций по подбору ключа, будет считаться взломом. Такие способы могут требовать нереалистично больших

объёмов подобранного открытого текста или памяти ЭВМ. Под взломом понимается лишь подтверждение наличия уязвимости криптоалгоритма, свидетельствующее о том, что свойства надёжности шифра не соответствуют заявленным характеристикам. Как правило, криптоанализ начинается с попыток взлома упрощённой модификации алгоритма, после чего результаты распространяются на полноценную версию: прежде чем браться за взлом, например, 16-раундовой версии DES, естественно для начала попытаться взломать шифр с меньшим количеством раундов, чем указано в его спецификации (например, 8-раундовую версию шифра).

Два последних десятилетия ознаменовались резким ростом числа открытых работ по криптологии, а криптоанализ становится одной из наиболее активно развивающихся областей исследований. Появился целый арсенал математических методов, представляющих интерес для криптоаналитика. Кроме того, повышение производительности вычислительной техники сделало возможными такие типы атак, которые раньше были неосуществимы.

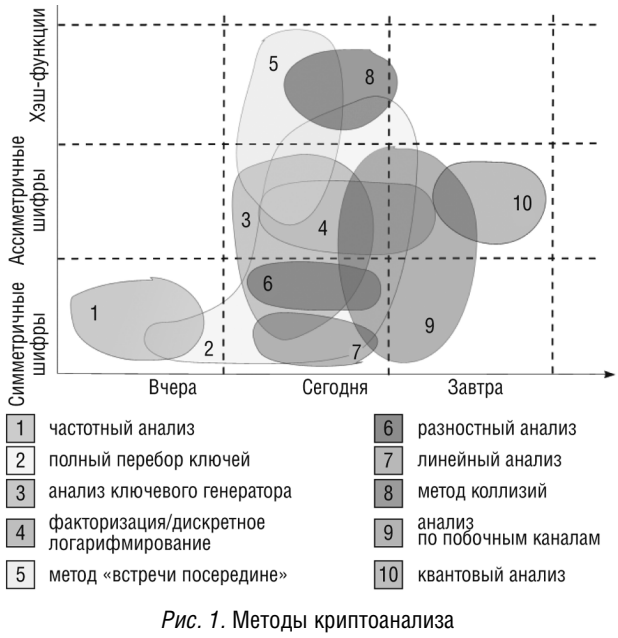

Рис. 1. Методы криптоанализа

На *рис.1* методы криптоанализа систематизированы по хронологии их появления и применимости для взлома различных категорий криптосистем. Горизонтальная ось разделена на временные промежутки: в область «вчера» попали атаки, которые успешно применялись для взлома шифров в прошлом; «сегодня» – методы криптоанализа, представляющие угрозу для широко используемых в настоящее время криптосистем; «завтра» – эффективно применяемые уже сегодня методы, значение которых в будущем может возрасти, а также методы, которые пока не оказали серьезного влияния на криптологию, однако со временем могут привести прорывам во взломе шифров. На вертикальной оси обозначены области применения методов криптоанализа: для взлома криптосистем с секретным ключом, открытым ключом или хеш-функций.

Частотный анализ

На протяжении веков дешифрованию криптограмм помогает частотный анализ появления отдельных символов и их сочетаний. Частотный криптоанализ использует статистические и лингвистические методы для получения дополнительной информации о ключе, а аналитические методы предполагают математическое изучение алгоритма шифрования. Вероятности появления отдельных букв в тексте сильно различаются. Для русского языка, например, буква «о» появляется в 45 раз чаще буквы «ф» и в 30 раз чаще буквы «э». Анализируя достаточно длинный текст, зашифрованный методом замены, можно по частотам появления символов произвести обратную замену и восстановить исходный текст. Кроме того, порядок букв в словах и фразах естественного языка подчиняется определенным статистическим закономерностям. Частотный анализ учитывает частоту появления различных буквосочетаний: например, пара стоящих рядом букв «ся» в русском языке более вероятна, чем «цы», а «оь» не встречается никогда. Для большинства естественных языков такая статистика документирована. Сегодня эти принципы широко применяются программах по подбору паролей. Если программа перебора вначале подбирает наиболее вероятные пароли, а менее вероятные оставляет на потом, то перебор сокращается в десятки и сотни раз.

Как было отмечено выше, каждый новый метод криптоанализа добавляет новые требования к алгоритмам шифрования. Так, частотный метод, в котором по распределению символов в шифртексте выдвигаются гипотезы о ключе шифрования, породил требование равномерного распределения символов в шифртексте.

Метод полного перебора

С появлением высокопроизводительной вычислительной техники у криптоаналитиков появилась возможность вскрывать шифры методом перебора ключей. В процессе криптоанализа приходится

перебирать миллиард ключей со скоростью тысяча ключей в секунду.

При осуществлении попытки атаки на основе только шифртекста криптоаналитику требуется анализировать выходные данные алгоритма и проверять их «осмысленность». В случае, когда в качестве объекта шифрования выступает графический файл или программа, задача определения «осмысленности» выходных данных становится очень трудной. Если известно, что открытый текст представляет собой предложение на естественном языке, проанализировать результат и опознать успешный исход дешифрования сравнительно несложно, тем более что криптоаналитик зачастую располагает некоторой априорной информацией о содержании сообщения. Задачу выделения осмысленного текста, т.е. определения факта правильной дешифрации, решают при помощи ЭВМ с использованием теоретических положений, разработанных в конце XIX века петербургским математиком А.А. Марковым, – *цепей Маркова*.

Атаки с использованием известного или подобранного открытого текста встречаются чаще, чем можно подумать. В среде криптоаналитиков нельзя назвать неслыханными факты добычи открытого текста шифрованного сообщения или подкупа лица, которое должно будет зашифровать избранное сообщение. Предположим, злоумышленнику известна одна или несколько пар (x, y). Пусть для простоты для любой пары (x, y) существует единственный ключ k, удовлетворяющий соотношению $E_k(x) = y$. Примем проверку одного варианта ключа $k \in K$ за одну операцию. Тогда полный перебор ключей потребует $|K|$ операций, где $|K|$ – число элементов в множестве. Если в качестве оценки трудоёмкости метода взять математическое ожидание

Можно подумать, что с ростом мощности компьютеров разрядность ключа, достаточная для обеспечения безопасности информации против атаки методом полного перебора, будет неограниченно расти. Однако это не так. Существуют фундаментальные ограничения вычислительной мощности, наложенные структурой вселенной: например, скорость передачи любого сигнала не может превышать скорость распространения света в вакууме, а количество атомов во Вселенной (из которых, в конечном счете, состоят компьютеры) огромно, но конечно. Так, например, в [5] описаны два фундаментальных ограничения:

1. *Предел, основанный на выделяемой Солнцем энергии.* Все вычисления потребляют энергию. Согласно принципам классической термодинамики и статистической механики, потребление энергии при осуществлении необратимого преобразования (вычисления) имеет порядок $k \cdot T$, где T – температура окружающей среды (по абсолютной шкале Кельвина), а k – постоянная Больцмана (равная $1{,}4 \cdot 10^{-23}$ Дж/°К). Мощность излучения Солнца составляет приблизительно $3{,}86 \cdot 10^{26}$ Вт; таким образом, за весь свой предполагаемый период существования – 10 млрд. лет, или $3 \cdot 10^{17}$ секунд – Солнце выделит около 10^{44} Дж). Предположим, температура окружающей среды $T = 10^{-6}$ градусов, тогда энергозатраты на одну операцию составляют $1{,}4 \cdot 10^{-29}$ Дж. Значит, количество вычислительных операций, которые можно осуществить с использованием всей выделяемой солнцем энергии, равно выделяемой мощности, поделенной на количество энергии, необходимой для осуществления одной операции, т.е. всего 10^{73} операций. Такое количество операций потребовалось бы на взлом ключа из 73 десятичных цифр (или около 250 бит) методом прямого перебора при грубом предположении, что для проверки одного значения ключа необходима всего одна операция (на самом деле – сотни операций). Для справки, количество атомов солнечной системы – около 10^{60}.

2. Предел, основанный на массе Земли. Масса Земли составляет порядка $6 \cdot 10^{24}$ кг. Масса протона – $1{,}6 \cdot 10^{-27}$ кг, т.е. Земля содержит приблизительно $4 \cdot 10^{51}$ протонов. Сопоставим каждому протону отдельный компьютер и примем за скорость выполнения операции на таком компьютере время, за которое луч света проходит расстояние, равное диаметру этого протона

$$\frac{10^{-15}\ \text{м}}{3 \cdot 10^{10}\ \text{м/с}}\,\text{с} = \frac{1}{3} \cdot 10^{-25}.$$

Таким образом, каждый компьютер может выполнять $3 \cdot 10^{25}$ операций в секунду. Если все эти компьютеры будут работать параллельно, их суммарное быстродействие составит $4 \cdot 10^{51} \cdot 3 \cdot 10^{25}$ операций в секунду, т.е. 10^{77} операций в секунду. Возраст Вселенной приблизительно 10 млрд. лет, т.е. $3 \cdot 10^{17}$ секунд. За весь период существования Вселенной такие гипотетические компьютеры размером с протон смогли бы выполнить $3 \cdot 10^{94}$ операций. При описанных в п. 1 предположений относительно количества операций, необходимых на проверку значения ключа, такое количество операций позволит взломать ключ длиной 95 десятичных цифр, или 320 бит.

Таким образом, минимальный размер ключа, необходимого для защиты информации от атак злоумышленника, будет расти по мере повышения быстродействия компьютеров; тем не менее, приведённые выше вычисления показывают, что существует возможность выбрать такую длину ключа, что атаку методом полного перебора будет осуществить в принципе невозможно, вне зависимости от повышения вычислительной мощности компьютеров или успехов в области классической теории алгоритмов.

случайной величины, соответствующей числу опробований до момента обнаружения использованного ключа, то мы получим $|K|/2$ операций. Кроме того, алгоритм полного перебора допускает распараллеливание, что позволяет значительно ускорить нахождение ключа.

Атака по ключам

Одной из причин ненадёжности криптосистем является использование слабых ключей. Фундаментальное допущение криптоанализа, впервые сформулированное О. Кирхгоффом, состоит в том, что секретность сообщения всецело зависит от ключа, т.е. весь механизм шифрования, кроме значения ключа, известен противнику (секретность алгоритма не является большим препятствием: для определения типа программно реализованного криптографического алгоритма требуется лишь несколько дней инженерного анализа исполняемого кода). Слабый ключ – это ключ, не обеспечивающий достаточного уровня защиты или использующий в шифровании закономерности, которые могут быть взломаны. Обычно считается, что алгоритм шифрования должен по возможности не иметь слабых ключей. Если это невозможно, то количество слабых ключей должно быть минимальным, чтобы уменьшить вероятность случайного выбора одного из них. Тем не менее, все слабые ключи должны быть заранее известны, чтобы их можно было отбраковать в процессе создания ключа.

Генераторы случайных чисел – ещё одно место, в котором часто ломаются криптографические системы. Это означает, что, если для генерации ключей используется криптографический слабый алгоритм, независимо от используемого шифра вся система будет нестойкой. Качественный ключ, предназначенный для использования в рамках симметричной криптосистемы, представляет собой случайный двоичный набор. Если требуется ключ разрядностью n, в процессе его генерации с одинаковой вероятностью должен получаться любой из 2^n возможных вариантов. Генерация ключей для асимметричных криптосистем – процедура более сложная, т.к. ключи, применяемые в таких системах,

Известно два направления в организации параллельного вычисления ключа методом полного перебора [2].

1. *Построение конвейера.* Предположим, цель злоумышленника заключается в осуществлении атаки на основе открытого текста. Тогда ему необходимо последовательно проверять истинность соотношения $E_k(x) = y$ для всевозможных значений k при известной паре (x, y). Один шаг алгоритма можно представить в виде детерминированной цепочки простейших операций: $O_1, O_2, ..., O_N$.

Возьмём N процессоров $A_1, A_2, ..., A_N$ и положим, что i-й процессор выполняет три одинаковые по времени операции:

1) приём данных от $(i–1)$-го процессора;

2) выполнение операции O_i;

3) передача данных следующему $(i+1)$-му процессору.

Тогда конвейер из N последовательно соединённых, параллельно и синхронно работающих процессоров работает со скоростью $v/3$, где v – скорость выполнения одной операции процессором.

2. *Распределенный поиск.* Множество ключей K разбивается на непересекающиеся подмножества $K_1, K_2, ..., K_Q$. Система из Q машин перебирает ключи так, что i-я машина осуществляет перебор ключей из множества K_i, $i = \overline{1,Q}$. Как только одна из машин находит ключ, система прекращает работу. Сложность в изложенном подходе – организация деления ключевого множества. Если организовать поиск ключа, чтобы при очередном опробовании каждый из процессоров стартовал со случайной точки, время опробования увеличится, но схема упростится. Среднее число шагов опробования N процессорами ключей из множества K составит $|K|/N$.

Реализация такого параллелизма допускает различные решения. Самое очевидное – создание компьютерного вируса для распространения программы-взломщика в глобальной сети. Программа подключается к серверу, получает от него набор ключей для перебора и после окончания работы возвращает результат. Вирус должен использовать периоды простоя компьютера (по данным исследований, компьютер простаивает 70–90% времени) для осуществления перебора по множеству ключей, создание вируса, незаметно для пользователя устанавливающего на подключённый к сети компьютер программу, способную осуществлять дешифрование сообщения путём перебора ключей. С развитием сетей (в частности, Интернета), стало возможным эффективно использовать этот метод. Подтверждение этому – вскрытие RC5-64 (блочного шифра компании RSA, использующего 64-битный ключ). Стартовавший в 1997 г. на сайте www.distributed.net проект «распределённого взлома» (в нём на добровольной основе приняли участие более 300 тысяч пользователей), успешно завершён за пять лет (1757 дней). За это время было перебрано 85% всего пространства ключей. Такой подход применим не только для взлома шифров, но и, например, для подбора двух текстов, имеющих одинаковое значение хеш-функции.

должны обладать определёнными математическими свойствами. Например, в случае системы RSA модуль шифрования представляет собой произведение двух больших простых чисел.

Исследования компании Counterpane, президентом которой является известный криптограф Б. Шнайер, показали, что определённые генераторы случайных чисел могут быть надёжными при использовании с одной целью, но ненадёжными для другой; обобщение анализа надёжности опасно.

Метод «встречи посередине»

Другой популярный метод криптоанализа – алгоритм «встречи посередине» – поддается эффективному распараллеливанию. Например, для логарифмирования в группе порядка p при параллельной работе n процессоров, где $n \times p$, время работы алгоритма уменьшается в n раз.

Данный метод криптоанализа основан на *«парадоксе дней рождения»*. Пусть нам нужно найти ключ k по известному открытому тексту x и криптограмме y. Если множество ключей криптоалгоритма замкнуто относительно композиции, т.е. для любых ключей k' и k'' найдется ключ k такой, что результат шифрования любого текста последовательно на k' и k'' равен результату шифрования этого же текста на k, т.е. $E_{k''}(E_{k'}, x) = E_k(x)$, то можно воспользоваться этим свойством. Поиск ключа k сведем к поиску эквивалентной ему пару ключей k', k''. Для текста x построим базу данных, содержащую случайное множество ключей k' и соответствующих криптограмм $w = E_k(x)$, и упорядочим её по криптограммам w. Объём базы данных выбираем

$$O(\sqrt{|\{k'\}|}),$$

где $|\{k'\}|$ – мощность множества ключей k'.

Затем подбираем случайным образом ключи k'' для расшифровки текстов y и результат расшифрования $v = E_{k''}(y)$ сравниваем с базой данных. Если текст v окажется равным одной из криптограмм w, то ключ $k'k''$ эквивалентен искомому ключу k.

Обозначим $\alpha = |\{k\}|$ – общее количество возможных ключей k. Временная сложность метода составляет

$$O(\sqrt{\alpha}\log\alpha).$$

Множитель $\log\alpha$ учитывает сложность сортировки. Требуемая память равна

$$O(\sqrt{\alpha}\log\alpha) \text{ бит, или } O(\sqrt{\alpha}) \text{ блоков}$$

(предполагается, что длина блока и длина ключа различаются на ограниченную константу).

Алгоритм является вероятностным. Однако существуют и детерминированный аналог этого алгоритма «giant step – baby step» с такой же сложностью, предложенный американским математиком Д. Шенксом.

Криптоанализ симметричных шифров

Наибольший прогресс в разработке методов раскрытия блочных шифров был достигнут в самом конце XX века и в основном связан с появлением в начале 90-х годов двух методов – *разностного (дифференциального) криптоанализа и линейного криптоанализа*.

Метод разностного анализа сочетает в себе обобщение идеи общей линейной структуры с применением вероятностно-статистических методов исследования. Этот метод относится к атакам по выбранному открытому тексту. Хотя Д. Копперсмит утверждает, что разностный криптоанализ был известен команде разработчиков DES алгоритма еще в начале 70-х годов, официальной датой появления этого метода считается 1990 г., а первенство в разработке признано за израильскими математиками Э. Бихамом и А. Шамиром. Разностный анализ основан на использовании неравновероятности в распределении значений разности двух шифртекстов, полученных из пары открытых текстов, имеющих некоторую фиксированную разность. Отметим, что разностный анализ применим и для взлома хеш-функций.

Подобно разностному анализу, линейный криптоанализ является комбинированным методом, сочетающим в себе поиск линейных статаналогов для уравнений шифрования, статистический анализ имеющихся открытых и шифрованных текстов, использующий также методы согласования и перебора. Этот метод исследует статистические линейные соотношения между отдельными координатами векторов открытого текста, соответствующего шифртекста и ключа, и использует эти соотношения для определения статистическими методами отдельных координат ключевого вектора.

На сегодняшний день метод линейного криптоанализа позволил получить наиболее сильные результаты по раскрытию ряда итерационных систем блочного шифрования, в том числе и системы DES. Метод линейного криптоанализа в неявном виде

появился еще в работе С. Мёрфи в 1990 г., где он успешно применялся при анализе системы блочного шифрования FEAL. В 1992 г. М. Мацуи формализовал этот подход, а позже успешно применил его к анализу криптоалгоритма DES.

В 2001 г. на смену DES и Triple DES пришёл стандарт AES (Advanced Encryption Standard), действующий и по сей день. Шифр AES основан на алгоритме Rijndael, разработанном бельгийцами Д. Дейменом и В. Райменом.

В [3] рассматриваются вопросы устойчивости алгоритмов ГОСТ 28147-89 и AES к линейному и разностному методам криптоанализа. Дать оценку устойчивости алгоритма ГОСТ к конкретным видам криптоанализа невозможно без спецификации узлов замен, которые не зафиксированы в стандарте, но существенно влияют на качество шифра. Исследования близких по архитектуре шифров с заданными таблицами подстановок показали, что криптоанализ шифра с 16 раундами требует очень большого числа исходных данных, хотя в принципе осуществим, а при 20–24 раундах становится теоретически бесполезным. Предусмотренных ГОСТом 32-х раундов шифрования хватает с запасом, чтобы успешно противостоять указанным видам криптоанализа. Шифр Rijndael, по оценкам разработчиков, уже на четырёх раундах шифрования приобретает достаточную устойчивость к линейному и разностному методам. Согласно спецификации, в шифре предусмотрено 10–14 раундов, а теоретической границей, за которой эти виды криптоанализа теряют смысл, является рубеж в 6–8 раундов в зависимости от размера блока. Таким образом, Rijndael устойчив к указанным видам криптоанализа с определенным запасом.

Криптоанализ асимметричных шифров

Практически все используемые алгоритмы асимметричной криптографии основаны на задачах факторизации (например, известная криптосистема RSA) и дискретного логарифмирования в различных алгебраических структурах (схема электронно-цифровой подписи Эль-Гамаля). Несмотря на то, что принадлежность этих задач к классу *NP*-полных задач не доказана, на сегодняшний день не найден полиномиальный алгоритм их решения. Для криптоанализа асимметричных криптосистем можно применять универсальные методы – например, метод «встречи посередине». Однако есть и другие методы, учитывающие специфику систем с открытым ключом. Они заключаются в решении математической задачи, положенной в основу асимметричного шифра. С того момента, как У. Диффи и М. Хеллман в 1976 г. предложили концепцию криптографии с открытым ключом, задачи факторизации целых чисел и дискретного логарифмирования стали объектом пристального изучения для математиков всего мира. За последние годы в этой области наблюдался значительный прогресс. Подтверждением тому может служить следующий казус: в 1977 г. Р. Ривест заявил, что разложение на множители 125-разрядного числа потребует 40 квадриллионов лет, однако уже в 1994 г. было факторизовано число, состоящее из 129 двоичных разрядов!

Задача дискретного логарифмирования считается более сложной, чем задача факторизации. Если будет найден полиномиальный алгоритм её решения, станет возможным и разложение на множители (обратное не доказано).

Последние достижения теории вычислительной сложности показали, что общая проблема логарифмирования в конечных полях не может считаться достаточно прочным фундаментом. Наиболее эффективные на сегодняшний день алгоритмы дискретного логарифмирования имеют уже не экспоненциальную, а субэкспоненциальную временную сложность. Это алгоритмы «index-calculus», использующие факторную базу. Первый такой алгоритм был предложен Л. Адлеманом и имеет временную сложность

$$L_p\left[\frac{1}{2}; c\right].$$

При вычислении дискретного логарифма в простом поле $\mathbf{Z}_p$

$$L_N[\gamma; c] = e^{(c+o(1))(\log N)^{\gamma}(\log\log N)^{1-\gamma}},$$

где $0 < y < 1$, $c = \text{const}$, $c > 0$.

На практике алгоритм Адлемана оказался недостаточно эффективным; Д. Копперсмит, А. Одлыжко и Р. Шреппель предложили алгоритм дискретного логарифмирования COS с эвристической оценкой сложности

$$L_p\left[\frac{1}{2}; 1\right] \text{ операций.}$$

Алгоритм решета числового поля, предложенный О. Широкауэром, при работает эффективнее различных модификаций метода COS; его временная сложность составляет порядка

$L_p\left[\frac{1}{3}; (64/9)^{1/3}\right]$ арифметических операций.

Ряд успешных атак на системы, основанные на сложности дискретного логарифмирования в конечных полях, привёл к тому, что стандарты ЭЦП России и США, которые были приняты в 1994 г. и базировались на схеме Эль-Гамаля, в 2001 г. были обновлены: переведены на эллиптические кривые. Схемы ЭЦП при этом остались прежними, но в качестве чисел, которыми они оперируют, теперь используются не элементы конечного поля $GF(2'')$ или $GF(p)$, а эллиптические числа – решения уравнения эллиптических кривых над указанными конечными полями. Роль операции возведения числа в степень в конечном поле в обновленных стандартах выполняет операция взятия кратной точки эллиптической кривой – «умножение» точки на целое число. Надлежащий выбор типа эллиптической кривой позволяет многократно усложнить задачу взлома схемы ЭЦП и уменьшить рабочий размер блоков данных. Старый российский стандарт ЭЦП оперирует 1024-битовыми блоками, а новый, основанный на эллиптических кривых, – 256-битовыми, и при этом обладает большей стойкостью.

Алгоритмов, выполняющих дискретной логарифмирование на эллиптических кривых в общем случае хотя бы с субэкспоненциальной сложностью, на сегодняшний день не существует. Тем не менее, известны работы И.А. Семаева, в одной из которых рассматривается метод, идейно близкий методам логарифмирования в конечном поле Адлемана. В другой работе для эллиптических кривых специального вида (накладываются некоторые условия на модуль арифметики и на мощность группы точек) Семаев указал способ сведения с полиномиальной сложностью задачи логарифмирования в группе точек эллиптической кривой к задаче логарифмирования в некотором расширении простого поля. При этом используется так называемое спаривание Вейля, после чего можно применять известные субэкспоненциальные методы. Аналогичные результаты опубликованы за рубежом.

Криптоанализ хеш-функций

Основная атака на хеш – это метод коллизий [2]. Пусть M и M' – сообщения, H – хеш-функция, а ЭЦП представляет собой некоторую функцию S от хеша сообщения: $C = S[H(M)]$. Законный обладатель пары «открытый ключ – секретный ключ» готов подписать сообщение M, но злоумышленник заинтересован в получении подписи под сообщением M'. Если M' выбрано так, что $H(M) = H(M')$, злоумышленник может предъявить пару (M', C): атака удалась. Реализовать подбор такого сообщения можно методом, основанном на упомянутом

Известно, что если считать, что дни рождения распределены равномерно, то в группе из 23 человек с вероятностью 0,5 у двух человек дни рождения совпадут. В общем виде этот парадокс формулируется так: если $a\sqrt{b}$ предметов выбираются с возвращением из некоторой совокупности размером b, то вероятность того, что два из них совпадут, равна

$$1 - e^{-\frac{a^2}{2}}$$

(в описанном частном случае b = 365 – количество дней в году, $a\sqrt{b} = 23$, т.е. $a \approx 1{,}204$).

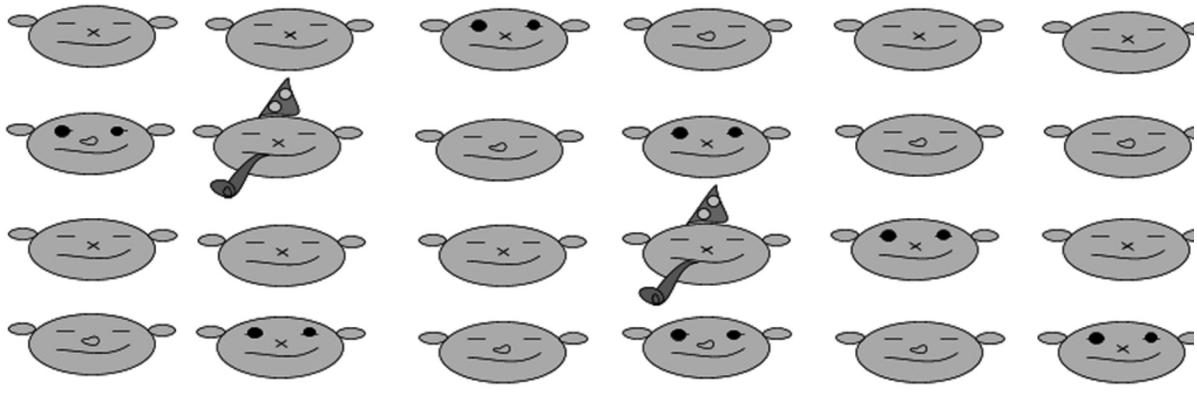

Рис. 2. Парадокс дней рождения

выше «парадоксе дней рождения». Варьируя интервалы, шрифты, формат и т.п., злоумышленник получает n пар вариантов M и M' без изменения их смысла. Сообщения $M_1,...,M_n$ отличаются слабо, а их хеш-функции – значительно, т.е. можно считать, что значения хеш-функций выбираются случайно, равновероятно и независимо друг от друга. Тогда при $n = t\sqrt{N}$ ($t > 0$ – некоторая константа, N – мощность множества всевозможных хеш-функций) вероятность того, что имеется пара сообщений M и M', для которых $H(M) = H(M')$, вычисляется по формуле

$$1 - e^{-\frac{t^2}{2}}.$$

Этот метод криптоанализа породил требования устойчивости к коллизиям для хеш-функций.

Атаки по сторонним, или побочным, каналам используют информацию, которая может быть получена с устройства шифрования и не является при этом ни открытым текстом, ни шифртекстом. Такие атаки основаны на корреляции между значениями физических параметров, измеряемых в разные моменты во время вычислений, и внутренним состоянием вычислительного устройства, имеющим отношение к секретному ключу. Этот подход менее обобщённый, но зачастую более мощный, чем классический криптоанализ.

В последние годы количество криптографических атак, использующих слабости в реализации и размещении механизмов криптоалгоритма, резко возросло. Противник может замерять время, затрачиваемое на выполнение криптографической операции, анализировать поведение криптографического устройства при возникновении ошибок вычисления. Другой подход предполагает отслеживание энергии, потребляемой смарт-картой в процессе выполнения операций с секретным ключом (например, расшифрования или генерации подписи). Побочную информацию собрать несложно – сегодня выделено более десяти побочных каналов, в т.ч. электромагнитное излучение, ошибки в канале связи, кэш-память и световое излучение. Подробное описание перечисленных типов атак можно найти в материалах доклада А.Е. Жукова на конференции РусКрипто-2006 [4], использованных при подготовке данного раздела.

Нанотехнологии в криптоанализе

С помощью квантового компьютера можно проводить вычисления, не реализуемые на классических компьютерах. В 1994 г. П. Шор открыл так называемый «ограниченно-вероятностный» алгоритм факторизации, позволяющий разложить на множители число N за полиномиальное от размерности задачи время $O[(\log N)^3]$. Алгоритм Шора разложения чисел на множители – главное достижение в области квантовых вычислительных алгоритмов. Это не только крупный успех математики. Именно с этого момента началось усиленное финансирование работ по созданию квантовых компьютеров.

Важно отметить, что алгоритм Шора чрезвычайно прост и довольствуется гораздо более скромным аппаратным обеспечением, чем нужное для универсального квантового компьютера. Квантовое устройство для разложения на множители будет построено, вероятно, задолго до того, как весь диапазон квантовых вычислений станет технологически осуществимым. На сегодня есть конкретные результаты. Так, IBM продемонстрировала использование созданного в лабораториях компании семикубитового квантового компьютера для факторизации чисел по алгоритму Шора. Хотя решённая им задача вряд ли способна поразить воображение (компьютер верно определил, что делителями числа 15 являются числа 5 и 3), это самое сложное вычисление за всю историю квантовых компьютеров.

Заключение

Возникает вопрос: если прогресс в области разработки новых методов взлома велик, почему мы продолжаем использовать криптосистемы, чья стойкость постоянно снижается? Ещё во времена Второй Мировой войны основоположник современной криптографии Клод Шеннон доказал существование принципиально не раскрываемых шифров – совершенно секретных систем, в которых ключ, «накладываемый» на текст, не может использоваться повторно, а его размер больше либо равен объёму текста.

Дело в том, что использование способа шифрования, получившего название *«одноразовых блокнотов»*, в большинстве случаев оказывается слишком дорогим и неоправданным. Нет смысла бороться за устойчивость системы защиты информации к взлому ниже некоторой «фоновой» вероятности, когда на другой чаше весов оказываются такие характеристики криптосистемы, как стоимость, сложность реализации и скорость доступа к зашифрованному тексту. Выбор необходимой степени защиты – это

поиск компромисса между уровнем безопасности и расходами на ее обеспечение. Таким образом, разработка новых методов криптоанализа с последующей публикацией и открытым обсуждением результатов является основным двигателем современной криптографии. ■

Литература

1. Ростовцев А.Г., Михайлова Н.В. Методы криптоанализа классических шифров // [Электронный ресурс] Сайт Санкт-Петербургского Государственного Политехнического Университета, 1998. URL: http://www.ssl.stu.neva.ru/psw/crypto/rostovtsev/cryptoanalysis.html (дата обращения: 14.04.09).
2. Грушо А.А., Применко Э.А., Тимонина Е.Е. Анализ и синтез криптоалгоритмов. Курс лекций. – Йошкар-Ола: Мар. фил. Моск. открытого соц. ун-та, 2000. – 110 с.
3. Материалы ассоциации Рускрипто: архивы научно-исследовательского семинара «Защита информации: аспекты теории и вопросы практических приложений» под рук. А.Е.Жукова и ежегодных конференций Рускрипто // [Электронный ресурс] Сайт ассоциации Рускрипто. URL: http://ruscrypto.ru/sources/ (дата обращения: 14.04.09).
4. Dam K.W., Lin H. S. Cryptography's Role in Securing the Information Society. National Academy Press. Washington, D.C. 1996.