АВТОМАТИЗИРОВАННАЯ СИСТЕМА МОДЕЛИРОВАНИЯ БЛОЧНЫХ ШИФРОВ «СПЕКТР»

Е. В. Морозова (Санкт-Петербург)

Разнообразные автоматизированные средства проектирования и моделирования, созданные для различных областей науки и производства, существенно облегчают работу специалистов в соответствующих областях, одновременно повышая ее качество. В то же время такая важная область науки, как криптография, фактически до сих пор оставалась без универсальных средств автоматизации. Специалисты-криптографы, как разрабатывающие собственные шифры, так и исследующие чужие, вынуждены для каждого шифра для его тестирования писать новую программу. Это влечет за собой необходимость либо самим криптографам изучать программирование, либо приглашать программистов-профессионалов. Как первый, так и второй путь ведет к потерям времени и возможности возникновения ошибок на этапе реализации шифра.

В то же время, на сегодняшний день для построения блочных шифров используются достаточно хорошо изученные базовые примитивы. Криптографические свойства шифра во многом зависят именно от взаиморасположения составляющих его примитивов.

Автоматизированная система проектирования и исследования блочных шифров «Спектр» была задумана и разрабатывалась как средство, позволяющее специалистам в области криптографии, не имеющим специализированных знаний в области программирования, самостоятельно проводить весь цикл работ при разработке блочных шифров от создания схемы криптоалгоритма до проведения тестовых испытаний. Это позволяет сократить время и затраты на исследование блочных шифров при одновременном соблюдении высоких требований на качество получаемых результатов.

Структура АС «Спектр» состоит из графического редактора, подсистемы синтаксического контроля и подсистемы вычислений. Графический редактор представляет собой поле рисования и набор реализованных в системе криптографических примитивов в виде кнопок со схематическими изображениями соответствующих примитивов. Последние могут быть либо стрелками, предназначенными для передачи данных, либо компонентами-объектами, выполняющими преобразование данных. Кроме того, компоненты-объекты могут быть базовыми (например, сложение по модулю два, циклический сдвиг) и сложными (блок управляемых перестановок, произвольный блок). Сложные компоненты могут либо состоять из некоторого набора базовых компонентов (произвольный блок), либо хранить свое описание в отдельном файле (фиксированная перестановка, фиксированная подстановка). При вводе модели шифра в систему пользователь помещает на поле рисования необходимые компоненты, соединяет их стрелками и производит настройку свойств как стрелок, так и компонентов.

Каждое действие пользователя при работе в графическом редакторе анализируется подсистемой синтаксического контроля. Часть проверок происходит непосредственно во время построения и настойки криптосхемы, часть – проводится над уже построенной моделью целиком. В первом случае проверки всегда производятся автоматически, во втором – могут вызываться пользователем по нажатию на соответствующую кнопку или запускаться автоматически при попытке перейти к исследованию модели.

Подсистема синтаксического контроля обеспечивает синтаксические проверки трех типов:

- допустимость графических построений в рамках используемого графического языка (например, превышение в результате введения новой стрелки максимально допустимого для данного типа компонента количества входов/выходов);

- правильность настроек компонентов создаваемой схемы криптоалгоритма (например, соответствие размерностей входов и выходов объединяемых компонентов, отсутствие неоднозначности идентификаторов стрелок);
- правильность алгоритмических построений (например, контроль отсутствия контуров, висящих компонентов, наличие всех необходимых входов-выходов для каждого блока).

Если в результате синтаксической проверки были обнаружены ошибки, система выдает пользователю соответствующие диагностические сообщения, после чего в диалоговом режиме ошибки исправляются. Пока все ошибки исправлены не будут, пользователю доступна только работа в графическом редакторе. Переход к исследованию шифров возможен лишь при отсутствии синтаксических ошибок. Если же синтаксических ошибок в графическом представлении криптоалгоритма найдено не было, то его перевод в аналитическое представление пройдет корректно и полученный псевдокод будет вычислимым.

Текущая версия системы позволяет проводить над моделями шифров следующие эксперименты: статистические тесты, два теста из серии тестов известных ответов, дифференциальный анализ, линейный анализ и коэффициент ветвления. Статистические тесты предназначены для выявления зависимостей между входными и выходными данными, тесты известных ответов - для определения правильности реализации криптоалгоритма, а дифференциальный и линейный анализ позволяют выявить уязвимости криптоалгоритма к соответствующим атакам. Рассмотрим реализованные тесты подробнее.

Согласно рекомендациям NESSIE и NIST, статистические свойства блочных криптоалгоритмов следует оценивать по так называемым «тестам зависимостей» (dependence tests), которые в русской литературе чаще называются просто «статистические тесты». Данные тесты включают в себя оценку следующих параметров блочного шифра [3]:

среднее количество выходных бит, изменяющихся при изменении одного входного бита;

степень полноты преобразования;

степень лавинного эффекта;

степень соответствия строгому лавинному эффекту.

Если обозначить через x вектор $x=(x_1,...x_n)\in(\mathrm{GF}(2))^n$,а через $x^{(i)}\in(\mathrm{GF}(2))^n$ вектор, полученный из вектора x путем инвертирования i-го бита (i=1,...,n), то функция f: $(\mathrm{GF}(2))^n \to (\mathrm{GF}(2))^m$ называется полной, если каждый выходной бит зависит от каждого входного бита, т.е. $\forall i=1,...n\ \forall j=1,...m\ \exists x\in(\mathrm{GF}(2))^n$, $(f(x^{(i)}))_j\neq(f(x))_j$. Бинарный вектор $z^{(i)}=f(x^{(i)})\oplus \mathrm{f}(x)$ называется лавинным вектором по компоненту i.

Говорят, что функция f: $(\mathrm{GF}(2))^n \to (\mathrm{GF}(2))^m$ имеет лавинный эффект, если при инвертировании одного входного бита в среднем изменяется половина выходных битов, т.е. $\frac{1}{2^n}\sum_{x\in(GF(2))^n} w(f(x)^i - f(x)) = \frac{m}{2}$ для всех i=1,...,n.

Говорят, что функция f: $(\mathrm{GF}(2))^n \to (\mathrm{GF}(2))^m$ удовлетворяет строгому лавинному критерию, если каждый бит выхода изменяется с вероятностью ½ при изменении одного входного бита, т.е. $\forall i=1,...n\ \forall j=1,...m\ \mathrm{P}(f(x^{(i)}))_j\neq(f(x))_j = 1/2$.

Матрица зависимостей функции f: $(\mathrm{GF}(2))^n \to (\mathrm{GF}(2))^m$ – это матрица $n\times m$, обозначаемая как A, у которой (i,j) элемент a_{ij} показывает количество входных векторов, для которых инвертирование i-го входного бита приводит к изменению j-го выходного бита, т.е. $a_{ij} = \#\{x\in(\mathrm{GF}(2))^n | (f(x^{(i)}))_j\neq(f(x))_j\}$ для i=1,...,n и j=1,...,m.

Матрица расстояний функции f: $(GF(2))^n \to (GF(2))^m$ – это матрица $n\times(m+1)$, обозначаемая как B, у которой (i,j) элемент b_{ij} показывает количество входных векторов, для которых инвертирование i-го входного бита приводит к изменению j выходных бит, т.е. $b_{ij} = \#\{x\in(GF(2))^n \mid w(f(x^{(i)}) - f(x)=j\}$ для $i=1,...,n$ и $j=1,...,m$. В данном определении w – это вес Хэмминга, обозначающий число ненулевых компонентов вектора x.

Безусловно, невозможно рассматривать все возможные входные векторы. Поэтому на практике берут вместо множества всех возможных двоичных векторов размерности n $(GF(2))^n$ некоторое случайно выбранное подмножество этого множества. Так, NESSIE считает, что мощность N этого подмножества должна быть равна 10000.

После вычисления матрицы зависимостей и матрицы расстояний можно получить результаты по искомым четырем критериям.

Среднее число битов выхода, изменяющихся при изменении одного бита входа (критерий №1), оценивается по формуле:

$$d_1 = \frac{1}{n}\sum_{i=1}^{n}\frac{\sum_{j=1}^{m} jb_{ij}}{N}. \qquad (1)$$

Степень полноты преобразования (критерий №2) оценивается по формуле

$$d_c = 1 - \frac{\#\{(i,j)\mid a_{ij}=0\}}{nm}. \qquad (2)$$

Степень лавинного эффекта (критерий №3) – вычисляется по формуле

$$d_a = 1 - \frac{\sum_{i=1}^{n}\left|\frac{1}{N}\sum_{j=1}^{m} 2jb_{ij} - m\right|}{nm}. \qquad (3)$$

Степень соответствия строгому лавинному критерию (№4) – по формуле

$$d_{sa} = 1 - \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}\left|\frac{2a_{ij}}{N} - 1\right|}{nm}. \qquad (4)$$

Для того, чтобы функция f соответствовала указанным критериям, необходимо, чтобы $d_c=1$, $d_a\approx 1$, $d_{sa}\approx 1$.

Одним из наиболее мощных видов криптоанализа на сегодняшний день является линейный и дифференциальный анализ. При этом проведение полного криптоанализа данных видов является очень ресурсоемким, поэтому в данной системе реализованы элементы этих видов криптоанализа. Например, при выборе дифференциального криптоанализа будет проведен поиск всех возможных дифференциальных характеристик для рассматриваемого шифра и найдена максимальная из их вероятностей. При выборе линейного криптоанализа проводится поиск линейных характеристик криптоалгоритма и также ищется их максимальная вероятность.

Одной из важных характеристик стойкости шифра к дифференциальному или линейному криптоанализу может служить значение соответствующего коэффициента ветвления. В настоящей версии системы реализован поиск линейного коэффициента ветвления.

Тесты известных ответов могут быть использованы для определения правильности реализации криптоалгоритмов. Данные тесты были обязательными при подаче блочных шифров-кандидатов на американский конкурс AES. В текущей версии реализовано два теста из этой серии – тест известных ответов с переменным ключом и тест

известных ответов с переменным открытым текстом. В ходе первого теста на вход модели шифра подается открытый текст, состоящий из нулевых бит. Каждый ключ, используемый для шифрования блока нулевого открытого текста, представляется в виде вектора, состоящего из «1» в первой позиции и «0» во всех других позициях. Открытый текст обрабатывается системой в соответствии с аналитическим представлением модели шифра для получения шифртекста. Каждый возможный ключ тестируется аналогично, путем сдвига «1»на одну позицию за раз. Для m-битового ключа будет m результатов, которые записываются в файл *_vk.rez.

В ходе теста известных ответов с переменным открытым текстом каждый входной блок данных представляет собой n-битовый вектор, состоящий из «1» в первой позиции и «0» в остальных позициях. Входной блок обрабатывается системой и полученный результат записывается в файл *_vt.rez. Аналогично тестируется каждый возможный открытый текст, получаемый из предыдущего сдвигом»1» на одну позицию за раз. В результате для n-битового открытого текста будет получено n результатов.

Полученные результаты системы записывает в текстовый файл. Для некоторых тестов пользователь может настроить степень подробности вывода. Например, при дифференциальном анализе можно сохранять как все находимые дифференциальные характеристики, так и их максимальные вероятности. Таким же образом настраивается подробность вывода результатов при линейном анализе. Это связано с большими объемами информации, которые должен проанализировать криптограф при этих видах криптоанализа. Так, применение дифференциального криптоанализа к шифру с входным блоком 16 бит даст результирующую таблицу (при полном выводе результатов) 65536×65536 (2^{16}). В текущей версии АС «Спектр» допускается поэтапный счет для линейного и дифференциального криптоанализа, т.е. пользователь может в любой момент прервать вычисления, а потом начать их с любого места или с места перерыва, или подвести итоги имеющихся на данный момент результатов. Кроме того, т.к. файл с результатами является текстовым и может быть изменен или просмотрен в любом текстовом редакторе, то пользователь может закомментировать какие-то строки этого файла, чтобы при окончательном подведении итогов система их не принимала во внимание.

Если пользователь проводил статистические эксперименты, поиск коэффициента ветвления или тесты известных ответов, то вывод результатов настройке не подлежит. В первом случае будут выведены полученные результаты по всем четырем статистическим критериям, а также матрицы зависимостей и расстояний. Во втором и в третьем случае в файл выводятся результаты каждого прохождения теста и исходные данные, по которым эти результаты получаются. Выводимые переменные и результаты комментируются, чтобы пользователь мог сразу увидеть, что представляет собой та или иная запись.

Аналогично система выводит результаты и при построении контрольных векторов. Кроме того, помимо вывода в текстовый файл, АС «Спектр» выводит полученные результаты и непосредственно на экран, в отдельное окно. В том же окне показывается псевдокод, по которому осуществлялись вычисления. Таким образом пользователь может одновременно посмотреть результаты вычислений и убедиться, что эти результаты основаны на правильном алгоритмическом представлении шифра.

Таким образом, преимуществами автоматизированной системы проектирования и исследования блочных шифров по сравнению с проведением экспериментов вручную являются:

- наглядность;

- возможность быстрого внесения изменений в имеющуюся модель и получение сравнительных результатов;
- точность и полнота вычислений заданных характеристик;
- быстрота ввода схемы шифрования;
- непосредственное участие разработчика алгоритма в проведении экспериментов безотносительно к его опыту программирования (исчезает проблема недопонимания между постановщиком задачи и ее исполнителем).

Автоматизированная система оценки блочных шифров внедрена в НФ ФГУП «НИИ «Вектор» СЦПС «Спектр», где используется для разработки блочных шифров и новых примитивов, и в Санкт-Петербургском государственном университете водных коммуникаций (СПГУВК), где используется для организации учебного процесса.

АС «Спектр» разработана в НФ ФГУП «НИИ «Вектор» СЦПС «Спектр». Для ее создания использовалась инструментальная среда Delphi 6, и ее общий объем составляет более 15000 строк кода. В настоящее время ведутся работы по дальнейшему совершенствованию данной системы.

Литература

1. **Молдовян А.А., Молдовян Н.А., Гуц Н.Д., Изотов Б.В.** Криптография: скоростные шифры. –СПб: БХВ-Петербург, 2002. – 496 с.
2. **Морозова Е.В.** Использование системы визуального моделирования криптоалгоритмов в учебных целях//Труды всеармейской научно-практической конференции "-Инновационная деятельность в Вооруженных Силах Российской Федерации". –СПб: ВУС, 2003. –С. 115-117.
3. Comments by the NESSIE Project on the AES Finalists/Preneel B., Bosselaers A., Rijmen V., Van Rompay B., et al.//http://www.nist.gav/aes