УДК 004.932.2+004.932.72'1

К.А. Ручкин, С.И. Сокур
Донецкий национальный технический университет, г. Донецк
Кафедра программного обеспечения интеллектуальных систем

АНАЛИЗ СТОЙКОСТИ УПРОЩЕННОЙ КРИПТОГРАФИЧЕСКОЙ СИСТЕМЫ AES

Аннотация

Ручкин К.А., Сокур С.И. Анализ стойкости упрощенной криптографической системы AES. *Выполнен сравнительный анализ современных алгоритмов шифрования. Подробно рассмотрен алгоритм шифрования AES. Произведено сравнение результатов алгоритмов шифрования с алгоритмом AES. Выбран оптимальный алгоритм исходя из всех тестов. Предложена модификация алгоритма AES.*

Ключевые слова: *шифрование, ключ, криптография, криптоанализ, шифр, дешифрование.*

Введение.

Задача использования криптографических методов в информационных системах стала в настоящий момент особо актуальна, так как расширилось использование компьютерных сетей, в частности Internet, по которым передаются большие объемы информации государственного, военного, коммерческого и частного характера, требующие обеспечения её безопасности, конфиденциальности, целостности, аутентичности и т.д.

В настоящий момент в шифровании данных используются системы с асимметричными и симметричными ключами. Асимметричные системы являются вычислительно медленными, но безопасными при инициализации своих ключей через незащищенные каналы связи. Поэтому они используются для формирования ключей к системам с симметричными ключами, которые используется уже для быстрой передачи больших объемов зашифрованной информации. Несмотря на довольно большое число различных асимметричных систем, наиболее популярна - криптосистема RSA, разработанная в 1977 году и получившая название в честь ее создателей: Рона Ривеста, Ади Шамира и Леонарда Эйдельмана. Алгоритм RSA основывается на том, что нахождение больших простых чисел и их умножение в вычислительном отношении осуществляется легко, но разложение на множители произведения двух таких чисел практически невыполнимо. Таким образом, раскрытие шифра RSA тесно связано с задачей разложения чисел на простые множители и нижняя оценка числа операций и, следовательно, время

раскрытия шифра RSA вычисляется исходя из производительности современных компьютеров и временной сложности наилучших из известных на текущий момент алгоритмов факторизации чисел. Именно такая возможность гарантированно оценить защищенность алгоритма RSA на основе современного состояния теории алгоритмов стала одной из причин её популярности и распространенности на фоне десятков других схем. Поэтому алгоритм RSA используется в банковских компьютерных сетях, особенно для работы с удаленными клиентами (обслуживание кредитных карточек).

Появление новых мощных компьютеров, технологий сетевых и нейронных вычислений сделало возможным дискредитацию криптографических систем еще недавно считавшихся практически не раскрываемыми. Таким образом, возникает необходимость в разработке и развитии новых алгоритмов шифрования, обладающих большей криптографической стойкостью. В настоящее время одним из таких алгоритмов является алгоритм Advanced Encryption Standard (AES). Исследованию этого алгоритма и посвящена данная работа.

Постановка задачи.

Целью данной работы является исследование и модификация алгоритма AES, а так же проведение ряда тестов для определения криптографической стойкости алгоритма модифицированного AES Advanced Encryption Standard.

Анализ литературы.

Проведем анализ современного состояния направления шифрование [1-7].

В работе [4] рассмотрено построение классификаций криптографических средств, криптоаналитических атак и злоумышленников, действия которых направлены на взлом криптосистем. Построена модель угроз безопасности в конкретной компьютерной системе, как композиция модели злоумышленника, модели атак и модели криптосистемы.

В работе [5-6] выполнен анализ показателей криптографической стойкости схемы разворачивания ключей шифра «Калина» к алгебраической атаке, строящейся на основе описания S-блоков с помощью переопределенной системы уравнений. Приведено обоснование стойкости СРК шифра «Калина» с точки зрения алгебраического анализа.

В работе [8] проведено исследование методов алгебраического криптоанализа. Получены системы уравнений для различных размеров таблиц нелинейных преобразований замены упрощенного алгоритма шифрования ГОСТ28147-89, а также выполнено решение одной из систем методом XL. В ходе работы программно реализован алгоритм генерации и решения системы уравнений для преобразований замены. Проведен анализ полученных нелинейных систем и выполнена оценка трудоемкости метода XL алгебраического криптоанализа для восьми блоков замены.

Упрощенная криптографическая система AES (Advanced Encryption Standard).

Advanced Encryption Standard (AES), также известный как Rijndael (произносится [rɛinda:l] (Рэндал)) — симметричный алгоритм блочного шифрования (размер блока 128 бит, ключ 128/192/256 бит), принятый в качестве стандарта шифрования правительством США по результатам конкурса AES. Этот алгоритм хорошо проанализирован и сейчас широко используется, как это было с его предшественником DES. Национальный институт стандартов и технологий США (англ. National Institute of Standards and Technology, NIST) опубликовал спецификацию AES 26 ноября 2001 года после пятилетнего периода, в ходе которого были созданы и оценены 15 кандидатур. 26 мая 2002 года AES был объявлен стандартом шифрования. По состоянию на 2009 год AES является одним из самых распространённых алгоритмов симметричного шифрования. Поддержка AES (и только его) введена фирмой Intel в семейство процессоров x86 начиная с Intel Core i7-980X Extreme Edition, а затем на процессорах Sandy Bridge.

Основные определения приведены в таблице 1.

Таблица 1 – Основные определения

Block	последовательность бит, из которых состоит input, output, State и Round Key. Также под Block можно понимать последовательность байт
Cipher Key	секретный, криптографический ключ, который используется Key Expansion процедурой, чтобы произвести набор ключей для раундов(Round Keys); может быть представлен как прямоугольный массив байтов, имеющий четыре строки и Nk колонок.
Ciphertext	выходные данные алгоритма шифрования
Key Expansion	процедура используемая для генерации Round Keys из Cipher Key
Round Key	Round Keys получаются из Cipher Key используя процедуру Key Expansion. Они применяются к State при шифровании и дешифровании
State	промежуточный результат шифрования, который может быть представлен как прямоугольный массив байтов имеющий 4 строки и Nb колонок
S-box	нелинейная таблица замен, использующаяся в нескольких трансформациях замены байт и в процедуре Key Expansion для взаимно-однозначной замены значения байта. Предварительно рассчитанный S-box можно увидеть ниже.
Nb	число столбцов(32-х битных слов), составляющих State. Для

	AES, Nb = 4
Nk	число 32-х битных слов, составляющих шифроключ. Для AES, Nk = 4, 6, или 8
Nr	число раундов, которое является функцией Nk и Nb. Для AES, Nr = 10, 12, 14
Rcon[]	массив, который состоит из битов 32-х разрядного слова и является постоянным для данного раунда. Предварительно рассчитанный Rcon[] можно увидеть ниже.

Процесс шифрования выглядит следующим образом. AES является стандартом, основанным на алгоритме Rijndael. Для AES длина input(блока входных данных) и State(состояния) постоянна и равна 128 бит, а длина шифроключа K составляет 128, 192, или 256 бит. При этом, исходный алгоритм Rijndael допускает длину ключа и размер блока от 128 до 256 бит с шагом в 32 бита. Для обозначения выбранных длин input, State и Cipher Key в 32-битных словах используется нотация Nb = 4 для input и State, Nk = 4, 6, 8 для Cipher Key соответственно для разных длин ключей.

В начале шифрования input копируется в массив State по правилу $\text{state}[r,c] = \text{input}[r+4c]$, для $0 \le r < 4$ и $0 \le c < Nb$. После этого к State применяется процедура AddRoundKey() и затем State проходит через процедуру трансформации (раунд) 10, 12, или 14 раз (в зависимости от длины ключа), при этом надо учесть, что последний раунд несколько отличается от предыдущих. В итоге, после завершения последнего раунда трансформации, State копируется в output по правилу $\text{output}[r+4c] = \text{state}[r,c]$, для $0 \le r < 4$ и $0 \le c < Nb$.

Отдельные трансформации SubBytes(), ShiftRows(), MixColumns(), и AddRoundKey() — обрабатывают State. Массив w[] — содержит key schedule.

Стоит так же отметить тот факт, что AES очень быстро и надежно шифрует большие объемы информации.

Исследование алгоритмов шифрования.

Проведем исследование алгоритмов шифрования. Результаты исследования приведены в таблице 2.

Таблица 2– Сравнение актуальных алгоритмов шифрования

Алгоритмы	**Авторы или организация**	**Основные результаты анализа**
CAST-256	Entrust Technologies, Inc.	В алгоритме не обнаружено уязвимостей. Однако, скорость шифрования у данного алгоритма невысока; кроме того, у него высокие требования к оперативной и энергонезависимой памяти.

Crypton	Future Systems, Inc.	Алгоритм без явных недостатков. Однако, Crypton уступает по всем характеристикам похожему на него алгоритму Rijndael. Эксперты конкурса сочли, что в финале конкурса Crypton однозначно проиграет Rijndael, поэтому не выбрали его во второй раунд конкурса.
DEAL	Richard Outerbridge, Lars Knudsen	Множество недостатков: наличие подмножеств слабых ключей, подверженность дифференциальному криптоанализу[1], отсутствие усиления при использовании 192-битных ключей по сравнению с 128-битными.
DFC	CNRS — Centre National pour la Recherche Scientifique — Ecole Normale Superieure	Достоинство алгоритма: очень высокая скорость шифрования на 64-битных платформах. При этом алгоритм весьма медленно шифрует на остальных платформах, а также имеет классы слабых ключей.
E2	NTT — Nippon Telegraph and Telephone Corporation	Аналогично алгоритму CAST-256, в E2 не найдено уязвимостей, но требования к энергонезависимой памяти слишком высоки.
FROG	TecApro Internacional S.A.	Алгоритм с наиболее оригинальной структурой и с наибольшим количеством недостатков: наличие уязвимостей, низкая скорость шифрования и высокие требования к ресурсам.
HPC	Richard Schroeppel	Аналогично алгоритму DFC, HPC очень быстро шифрует на 64-битных платформах, но весьма медленно на остальных. Кроме того, сложная и медленная процедура расширения ключа ограничивает возможные применения алгоритма, а наиболее сложная из всех представленных на конкурс алгоритмов структура раунда усложняет анализ алгоритма и делает недоказуемым отсутствие закладок.
LOKI97	Lawrie Brown, Josef Pieprzyk, Jennifer	Низкая скорость шифрования, высокие требования к ресурсам, наличие

	Seberry	уязвимостей.
Magenta	Deutsche Telekom AG	Алгоритм подвержен атакам методами линейного[2] и дифференциального криптоанализа; медленная скорость шифрования.
MARS	IBM	У алгоритма отсутствуют серьезные недостатки, за исключением низкой скорости шифрования на ряде платформ, не имеющих аппаратной поддержки требуемых операций и некоторых других незначительных недостатков. Алгоритм был незначительно модифицирован в течение первого раунда; измененный вариант вышел в финал конкурса.
RC6	RSA Laboratories	RC6 имеет весьма похожие на MARS проблемы с реализацией на некоторых платформах. Эксперты посчитали это незначительным недостатком — алгоритм вышел в финал конкурса.
Rijndael	Joan Daemen, Vincent Rijmen	Каких-либо недостатков у алгоритма не обнаружено; алгоритм вышел в финал конкурса.
SAFER+	Cylink Corporation	Алгоритм подвержен ряду атак и имеет низкую скорость выполнения.
Serpent	Ross Anderson, Eli Biham, Lars Knudsen	Выявлены некоторые незначительные недостатки, алгоритм вышел в финал конкурса.
Twofish	Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, Niels Ferguson	Из недостатков эксперты отметили сложность алгоритма, затрудняющую его анализ. Алгоритм вышел в финал конкурса.

В данной таблице как видно только пять алгоритмов удовлетворяют современным требованиям к безопасности: MARS, RC6, Rijndael, Serpent и Twofish.

А вот сравнительный анализ скорости шифрования, данный тест конечно же относительный и в разных условиях он может отличаться, но не на много. Данный тест дает более менее полную картину по скорости шифрования.

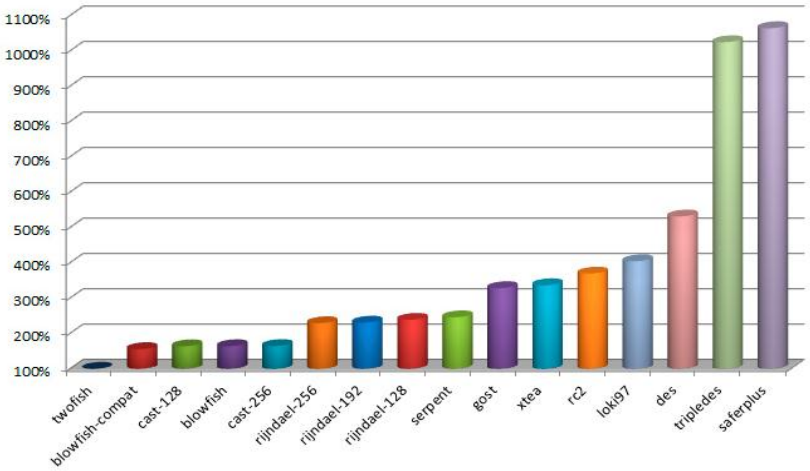

Рисунок 1 – Скорость алгоритмов шифрования

Как видно из результатов лидирует пятерка:Twofish, Blowfish-compact, Cast-128,Blowfish, Cast-256 и немного Rijndael.

Ну и самое точное это результаты от TrueCrypt:

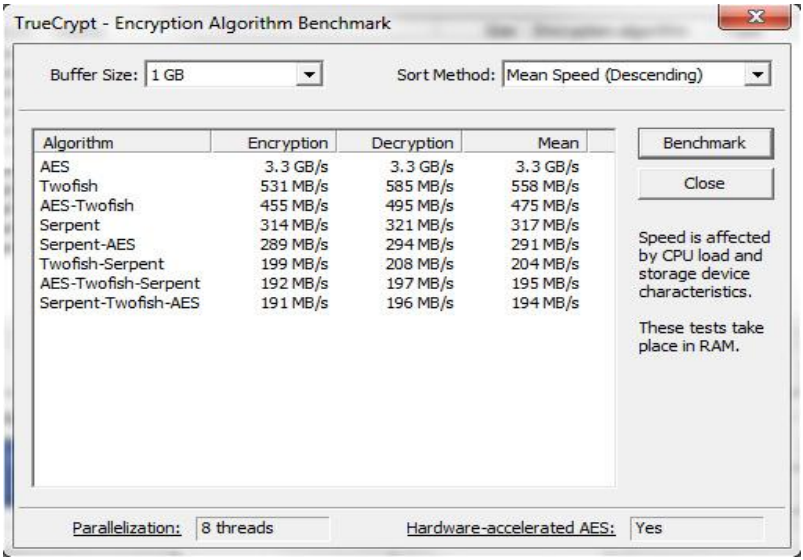

Рисунок 2 – Тесты от TrueCrypt

Из всех выше перечисленных алгоритмов лучше всех себя показали Rijndael и Twofish, т.к. они одни из самых быстрых и самых защищенных на текущий момент. Выбор Rijndael обоснован тем, что он менее увесист в коде и имеет много исходных кодов и библиотек. На данный момент Apple использует AES (256 бит) и RSA для передачи ключа.

Для того что бы модифицировать алгоритм AES, предполагается упростить его и применить к нему методы защиты других алгоритмов шифрования (получения гибрида алгоритмов). В частности соединение алгоритмов AES и RSA даст большую криптографическую стойкость. После усовершенствования алгоритмов будет проведен тест NIST.

Выводы.

В данной работе были рассмотрены некоторые алгоритмы шифрования, такие как криптосистема RSA (Rivest, Shamir & Adleman), симметричный алгоритм блочного шифрования AES (Advanced Encryption Standard), 3-WAY, симметричный алгоритм шифрования DES (Data Encryption Standard), симметричный поточный алгоритм SEAL (Software-optimized Encryption Algorithm), twofish.

Следует заметить, что поскольку размер шифруемого блока в RSA ограничен размером ключа (например, 2048бит, что есть 256 байт), и процедура такого шифрования занимает гораздо больше времени чем шифрование блока этих же данных симметричным алгоритмом (хотя, казалось бы, за раз зашифровать можно гораздо больше), напрямую для шифрования данных RSA не используется. Вместо этого генерируется случайный ключ для симметричного алгоритма, и передаваемые данные шифруются им.

Наиболее простой критерий такой эффективности - вероятность раскрытия ключа или мощность множества ключей (M). По сути это то же самое, что и криптографическая стойкость. Для ее численной оценки можно использовать также и сложность раскрытия шифра путем перебора всех ключей.

Однако, этот критерий не учитывает других важных требований к криптосистемам:

- невозможность раскрытия или осмысленной модификации информации на основе анализа ее структуры;
- совершенство используемых протоколов защиты;
- минимальный объем используемой ключевой информации;
- минимальная сложность реализации (в количестве машинных операций), ее стоимость;
- высокая оперативность.

Желательно конечно использование некоторых интегральных показателей, учитывающих указанные факторы. Для учета стоимости, трудоемкости и объема ключевой информации можно использовать удельные показатели - отношение указанных параметров к мощности множества ключей шифра.

Список литературы

1. Григорьев В.А., Григорьев С.В. Передача сообщений. / Под ред. В.А. Григорьева. – СПб.: ВУС, 2002. – 460с.

2. Оков И.Н. Криптографические системы защиты информации. – СПб.: ВУС, 2001. – 236с.

3. Т. В. Кузьминов. Криптографические методы защиты информации. Наука, Сибирское предприятие РАН, Новосибирск, 1998.

4. Harry Nyquist. Certain factors affecting telegraph speed. – Bell System Technical Journal, 3, 1924. С.324–346.

5. С.М. Авдошин, А.А. Савельева, Проблемы оценки криптозащищённости информационных систем, Бизнес информатика №2(04)–2008 г., с.1-15.

6. А.В. КАЗИМИРОВ, Р.В. ОЛЕЙНИКОВ. Харьковский национальный университет радиоэлектроники, Украина Алгебраические свойства схемы разворачивания ключей блочного симметричного шифра «Калина»

7. State space cryptanalysis of the MICKEY cipher. T Helleseth, CJA Jansen, O Kazymyrov, A Kholosha. Information Theory and Applications Workshop (ITA), 2013, 1-10

8. Е.А. Маро. Алгебраический анализ стойкости криптографических систем защиты информации. Инженерный вестник Дона. 2013. - 4.