При итоговой точности создаваемой модели более 95% возможен переход от «пороговых» правил в сторону поведенческого анализа потоков в промышленном использовании предлагаемого классификатора. Дальнейшее применение классификатора рассматривается в связке с IDS/IPS системами.

Литература

1. Колесниченко Д., “Машинное обучение на практике” [Электронный ресурс], 2018 - URL Режим доступа: https://xakep.ru/2018/08/01/rts-tender/ (дата обращения 15.03.2019).

2. Полякова Е.В., “ Исследование методов машинного обучения для анализа и принятия решений на основе данных Интернета вещей” [Электронный ресурс], 2018 - URL Режим доступа: https://publications.hse.ru/chapters/204754963 (дата обращения 15.03.2019).

3. C.-Y. Lee, P. W. Gallagher, and Z. Tu., «Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree», [Электронный ресурс], 2015 - URL Режим доступа: https://arxiv.org/abs/1509.08985 (дата обращения 15.03.2019).

4. Z. C. Lipton, J. Berkowitz, and C. Elkan, «A critical review of recurrent neural networks for sequence learning», Электронный ресурс], 2015 - URL Режим доступа: https://arxiv.org/abs/1506.00019 (дата обращения 15.03.2019).

5. Middlemiss M., Dick G., «Feature Selection of Intrusion detection data using a hybrid genetic of hybrid Intelligent systems», IOS Press Amsterdam, 2018.

6. F. Pierazzi, G.Apruzzese, M. Golajanni. A. Guido, «Scalable architecture for onlineprioritization of cyber threats», International Conference on Cyber Conflict, 2017.

К.Ф. Родичев, Ф.А. Дмитриев

РАЗРАБОТКА АЛГОРИТМА ШИФРОВАНИЯ С ИСПОЛЬЗОВАНИЕМ ТЕОРИИ ГРАФОВ

(Самарский университет)

Цель: создание алгоритма шифрования и исследование свойств шифра.

Задачи:

- Ввести основные понятия для описания алгоритма.
- Описать идею представления числа в виде графа.
- Разработать алгоритм шифрования на основе изложенной идеи.
- Написать программную реализацию алгоритма.
- Обосновать соответствие алгоритма изложенным требованиям.
- Проанализировать криптостойкость алгоритма.

Основные понятия, используемые в докладе

Специальные вершины – термин, использующийся в рамках данной работы – подмножество вершин графа, для каждой вершины которого определено множество смежных вершин. Эти множества не пересекаются и в объединении с множеством специальных вершин дают все множество вершин графа [1].

Введение

В современном мире, с ростом вычислительных мощностей, теория графов получила очень широкое применение. В большинстве случаев ее аппарат используется для решения оптимизационных задач. Однако, наряду с областями науки, в которых уже сегодня активно применяется теория графов, существуют области, в которых она еще не нашла широкого применения.

Одной из таких областей является криптография. Современные стандарты шифрования как блочные, так и поточные не используют теорию графов. Безусловно, существуют исследования, посвященные ее применению в криптографии [4,5]. Но в них не реализуются предложенные механизмы шифрования и не анализируют результаты их внедрения.

Актуальность данного исследования заключается не только в новизне предлагаемого алгоритма шифрования, но и в его качественном отличии – использовании теории графов. Практическая направленность состоит в программной реализации алгоритма для дальнейшего исследования и анализа.

Описание идеи представления числа с помощью графа

В основу алгоритма легла идея представления числа с помощью графа, возникшая при решении олимпиадной задачи по шифрованию. В ней для шифрования натурального числа N используется связный граф. Исходное число представляется в виде суммы натуральных чисел, которые «записываются» в вершины графа. Таким образом, у каждой вершины появляется значение. Затем к значениям вершин прибавляются значения из смежных вершин. В результате получается граф, в котором зашифровано исходное число. Получить это число можно, сложив значения специальных вершин.[2]

Разработка алгоритма шифрования на основе изложенной идеи

Изложенная идея не отвечает на следующие вопросы:

- Любое ли число можно зашифровать с помощью графа?
- Сколько слагаемых должно быть в представлении числа в виде суммы?
- Какие ограничения накладываются на эти слагаемые?
- Сколько специальных вершин необходимо выбирать и как это делать?
- По какому алгоритму представлять число в виде суммы слагаемых?
- Как строить граф, зная его «специальные» вершины?

Чтобы ответить на эти вопросы необходимо определить, к какому типу будет относиться шифр и каким требованиям он должен соответствовать.

Будем использовать симметричное блочное шифрование вида SP-сеть. Такой выбор обусловлен тем, что графы представимы в виде матриц – блоков. SP-сеть была выбрана, так как при ее использовании проводить матричные операции эффективнее (с временной точки зрения), чем в сети Фейстеля.

Основными требованиями к алгоритму стали: наличие лавинного эффекта [3] и отсутствие линейности преобразований.

Таким образом, ответив на поставленные вопросы, и, учтя требования, мы разработали алгоритм, в котором число шифруется на графе из 8 вершин, а в качестве ключа выступают их значения.

Разработанный алгоритм:

1. Получить числовое представление символа в соответствии с кодировкой.
2. Разбить полученное число N на сумму из 8 слагаемых по алгоритму:
   - Диапазон значений для k-ой вершины – d_k, значение вершины – v_k.
   - Определить диапазон значений для текущего элемента по формуле (1).

$$d_k = [1, (N - (n - k) - \sum_{1}^{k-1} v_i))] \quad (1)$$

   - Выбрать случайным образом значение v_k из диапазона d_k.
   - Повторять предыдущие пункты N раз.
3. Сформировать первичный ключ из значений этих слагаемых.
4. Составить матрицу смежности для графа.
   a. Найти число специальных вершин из диапазона.
   b. Определить, какие из вершин графа будут являться специальными.
   c. Для специальных вершин сформировать списки смежности.
   d. Составить матрицу смежности графа и дополнить ее отношениями между множествами смежности специальных вершин.
5. Зашифровать число с помощью матрицы и первичного ключа.
   a. Получить крипто-ключ. Для этого для каждой из 8-и вершин:
      i. Найти сумму значений всех смежных с ней вершин.
      ii. Полученную сумму сложить со значением самой вершины и записать в крипто-ключ.
   b. Сформировать 12 ключей для раундов, используя S-box.
   c. В цикле 12 раз произвести раунд шифрования:
      i. XOR матрицы и ключа.
      ii. Закольцованный сдвиг значений *i*-й строки и столбца матрицы на значение, равное *i*-му значению ключа.
      iii. Циклический сдвиг строк на n влево, где n – номер строки.

Для изложенного алгоритма была разработана программная реализация.

Соответствие алгоритма изложенным требованиям и его анализ

Проверка на соответствие требованиям и анализ проводились с помощью программной реализации алгоритма.

Для проверки на соответствие требованию лавинного эффекта был многократно зашифрован один символ, и найдено количество совпавших бит в шифроблоках путем их попарного сравнения. Анализ был проведен с помощью математической статистики. В качестве случайной величины было взято количество совпадающих бит. В результате, для выборки из 1000 шифроблоков был получен полигон частот в виде нормального закона распределения (рис. 1). Также было получено, что случайная величина в 99% случаев попадает в диапазон $[\bar{x} - \sigma; \bar{x} + \sigma]$, что говорит о низкой дисперсности.

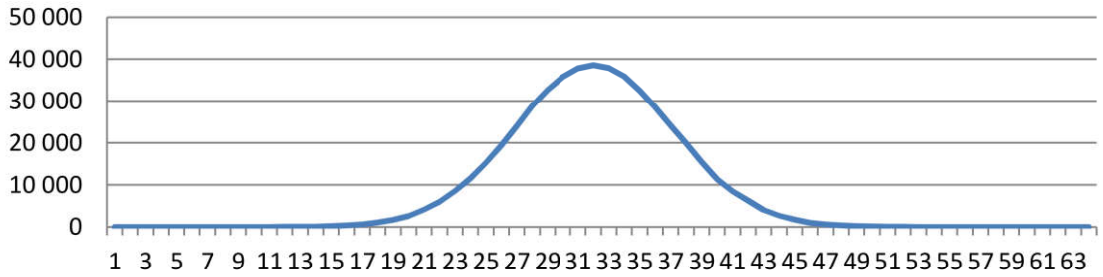

Рис. 1. Полигон частот для случая со 1000-ми шифроблоками

На основании вышеизложенного можно заключить, что в большинстве случаев для одного и того же числа мы получим в половину отличающиеся шифроблоки, что говорит о наличии лавинного эффекта.

Соответствие требованию нелинейности преобразований обусловлено использованием таблицы подстановки (S-box) для генерации ключей раундов из исходного ключа.

Таким образом, алгоритм удовлетворяет изложенным требованиям.

Дальнейший статистический анализ был произведен с целью определения криптостойкости алгоритма, его результаты приведены в выводе.

Вывод

1. Алгоритм соответствует требованиям наличия лавинного эффекта и нелинейности преобразований, что говорит о его общей надежности.

2. Алгоритмом не соблюдаются такие требования абсолютной устойчивости как уникальность генерируемого ключа, его статистическая надежность, а так же избыточность информации в открытом тексте. При этом, как можно заключить из криптоанализа, полученные несоответствия нельзя назвать критическими. Несоответствие требованию уникальности ключа не критично, так как вероятность появления одинаковых ключей мала (≈6,1E-21). Наблюдаемая статистическая ненадежность ключа (рис. 2.) прослеживается в незначительном отклонении от равномерного распределения вероятности появления «1» в 7-м и 8-м бите каждого байта. Максимальное отклонение составляет 0.03.

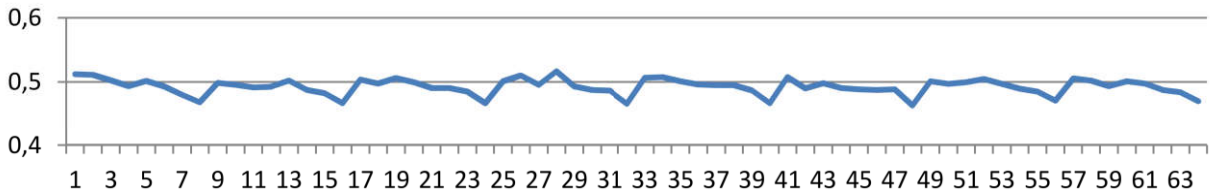

Рис. 2. Гистограмма для расширенной выборки

Отсутствие избыточности информации, может быть решено за счет введения битов четности для строк и столбцов матриц.

3. Сложность взлома прямым перебором оценивается комбинациями ключа. Зная, что современные суперкомпьютеры подбирают около 2^{44} ключей, обеспечивая ими все население планеты - 2^{33}человек, нам потребуется $2^{128-44-33-25}= 2^{26}$лет. Однако, беря в учет закон Мура и прогнозируемые Intel качественные улучшения процессоров, получим, что с помощью прямого перебора понадобится 26*1,5 = 39 лет на взлом.

Из вышесказанного можно заключить, что разработанный алгоритм относится к достаточно стойким. Но таковым он будет до тех пор, пока не будет найдена эффективная атака с использованием эвристик. Реализация такой атаки может стать следующим шагом в исследовательской работе по этой теме.

Литература

1. Емеличев, В.А. Лекции по теории графов [Текст]: учеб. пособие / [В.А. Емеличев, О.И. Мельников и др.] —М.: Наука, 1990 - 384 с.

2. Межрегиональная олимпиада школьников по математике и криптографии [Электронный ресурс]: Олимпиадная задача 2010/2011. URL: http://v-olymp.ru/cryptolymp/archive_task/469/3463/

3. Katz, J. Introduction to modern cryptography / J. Katz, Y. Lindell — CRC Press. —2008. — P. 166—167.

4. Ustimenko, V.A. On graph-based cryptography and symbolic computations / V.A. Ustimenko // Serdica.Journal of Computing. – 2007. – P. 131-156.

5. Yamuna, M. Encryption usinggraph theory and linear algebra/ M. Yamuna, M. Gogia, A. Sikka, Md. Jazib Hayat Khan //International Journal of Computer Application. — 2012 — ISSN:2250-1797