Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра теоретических основ компьютерной безопасности и криптографии

ПРОЕКТИРОВАНИЕ ПОТОКОВЫХ ШИФРОВ НА ОСНОВЕ ГЕНЕРАТОРОВ ПСЕВДОСЛУЧАЙНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ

АВТОРЕФЕРАТ

дипломной работы

студента 6 курса 631 группы

специальности 090102.65 «Компьютерная безопасность»

факультета компьютерных наук и информационных технологий

Книжникова Игоря Александровича

Научный руководитель

профессор, д.ф.-м.н. В.А. Молчанов

Заведующий кафедрой

профессор, к.ф.-м.н. В.Н. Салий

Саратов 2016

ВВЕДЕНИЕ

Одной из ключевых проблем современной криптографии является построение псевдослучайных последовательностей с заданными криптографическими свойствами и обладающих хорошими статистическими показателями. Трудности, связанные с решением данной проблемы, создали предпосылки для развития алгебраических методов построения последовательностей с необходимыми параметрами, разработки алгоритмов тестирования таких последовательностей и оптимизации соответствующих методов вычислений.

Целью данной работы является исследование методов генерации применяемых в потоковых шифрах псевдослучайных последовательностей, основанных на сдвиговых регистрах с линейной обратной связью, и разработка методов улучшения некоторых наиболее важных криптографических свойств псевдослучайных последовательностей, используемых в бинарно-аддитивных синхронных потоковых шифрах, таких как линейная сложность и максимальная длинна единичных и нулевых подпоследовательностей.

В первом разделе данной работы описывается схема применения потоковых шифров, преимущества и недостатки их использования, приводится классификация потоковых шифров. Во втором разделе работы рассматриваются основные понятия алгебры последовательностей над конечным полем, приводятся описание и пример работы сдвигового регистра с линейной обратной связью. Третий раздел работы посвящен проблеме поиска примитивных многочленов над полем F_2. Здесь приводятся и оцениваются наиболее эффективные алгоритмы решения этой задачи. В четвертом разделе описываются линейные параметры последовательностей, такие как линейная сложность и линейный профиль. В пятом разделе работы предлагается алгоритм построения генератора последовательности с требуемой линейной сложностью. В шестом разделе исследуется проблема построения

последовательности для гаммирования, имеющей необходимую линейную сложность. Здесь также рассматривается проблема проявления дефектов последовательности в виде нулевых и единичных подпоследовательностей при шифровании, предлагается алгоритм исправления таких дефектов. В седьмом разделе описывается разработанное в ходе выполнения работы программное криптосредство QuickCrypt, реализующее алгоритм шифрования и безопасного удаления данных, основанный на ранее полученных результатах. В конце работы приводятся результаты тестирования этого алгоритма.

В приложении A работы приводится исходный код реализации алгоритма нахождения примитивных полиномов заданной степени. Некоторые из найденных с помощью данной программы полиномов приводятся в таблице в приложении B. В приложениях C и D приводятся соответственно программы для определения линейной сложности последовательности и генерации последовательности по заданной длине ключа. В приложении E приводится листинг исходного кода программы QuickCrypt.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел работы посвящен классификации и исследованию потоковых шифров, в частности бинарно-аддитивных шифров, и схемам их применения. В разделе также рассматриваются преимущества и недостатки таких шифров.

Потоковые шифры являются важным классом алгоритмов шифрования. Они шифруют индивидуальные символы (обычно двоичные цифры) открытого текста отдельно, используя шифрующее преобразование, которое может варьироваться от символа к символу. В противоположность потоковым шифрам, блоковые шифры одновременно шифруют группы символов открытого текста, используя фиксированное преобразование. В аппаратной реализации, потоковые шифры обычно демонстрируют большую скорость работы, нежели блоковые, и имеют более простую схему.

Потоковые шифры делятся три основных класса:

- одноразовый блокнот (шифр Вернама);
- синхронные потоковые шифры;
- самосинхронизирующиеся потоковые шифры.

Синхронным называется потоковый шифр, в котором ключевая последовательность генерируется вне зависимости от открытого текста или криптограммы.

Синхронные потоковые шифры требуют синхронизации – отправитель и получатель должны быть синхронизированы, то есть использовать один и тот же ключ и находиться в одном и том же состоянии, для корректной расшифровки. Если синхронизация была утеряна из-за потери или вставки одного или нескольких символов криптограммы, то для дальнейшей работы требуется ресинхронизация, которая использует такие методы как реинициализация, вставка синхронизирующих меток на определенных

интервалах последовательности, или, если открытый текст имеет достаточную избыточность, перебор всех смещений ключевой последовательности.

Еще одной важной чертой синхронных потоковых шифров является отсутствие распространения ошибки. Если символ криптограммы был изменен, остальные символы вне зависимости от этого будут расшифрованы корректно.

Большинство современных потоковых шифров являются *бинарно-аддитивными*, то есть синхронными потоковыми шифрами, в которых ключевая последовательность, открытый текст и криптограмма представлены в двоичном виде, а функция шифрования представляет собой функцию XOR

Во **втором разделе** работы рассматриваются основные понятия алгебры последовательностей над конечным полем, приводятся описание и пример работы сдвигового регистра с линейной обратной связью.

Рассмотрим примитивный элемент α поля F_{p^r}. Поскольку он является порождающим в мультипликативной группе поля F_{p^r}, все ненулевые элементы поля можно представить в виде степени элемента α, как это показано в [8], то есть

$$F_{p^r}\backslash\{0\} = \{1, \alpha, \alpha^2, \ldots, \alpha^{p^r-2}\};$$

Соответственно, при помощи примитивного многочлена, соответствующего элементу α, можно найти рекуррентное уравнение для всех ненулевых элементов поля F_{p^r}.

Таким образом, примитивные многочлены имеют максимальный период генерируемой последовательности элементов поля, так как генерируют все ненулевые элементы. На основе многочлена возможно реализовать генератор псевдослучайной последовательности, который можно использовать в бинарно-аддитивном шифре для генерации ключевой последовательности.

Сдвиговый регистр с линейной обратной связью состоит из двух основных компонент: регистра и функции обратной связи, заданной многочленом. Регистр представляет собой r ячеек памяти для хранения предыдущих r элементов последовательности.

Сдвиговый регистр работает в дискретном времени. В начальный момент регистр инициализирован некоторым состоянием, в качестве примера возьмем 01101 (слева направо). Далее функция обратной связи вычисляет при помощи рекуррентного соотношения $x^5 = x^2 + 1$ новый элемент по модулю 2. Так, в нашем случае новый вычисленный элемент будет равен нулю. Затем младший бит регистра, соответствующий x^0 и называющийся выходным, выводится в качестве очередного бита генерируемой последовательности, а старшие $r - 1$ бит смещаются вправо на одну позицию. В старшую ячейку записывается новый вычисленный элемент, и регистр переходит в новое состояние, завершив один такт [15]. Новым состоянием будет являться 00110.

Третий раздел работы посвящен проблеме поиска примитивных многочленов над полем F_2. Здесь приводятся и оцениваются наиболее эффективные алгоритмы решения этой задачи.

Как уже было отмечено ранее, примитивные многочлены над полем F_2 являются ключевыми элементами при построении сдвигового регистра с максимальным периодом, поэтому задача об их нахождении также является одной из основных. Определим вычислительную сложность этой задачи. Для этого воспользуемся оценкой количества примитивных многочленов степени r, приведенной в [8]:

$$\lambda(r) = \frac{\varphi(2^r - 1)}{r},$$

где $\lambda(r)$ – это количество примитивных многочленов над F_2 степени r, а φ – это функция Эйлера.

На практике, для нахождения всех примитивных полиномов небольших степеней применяются алгоритмы, перебирающие все возможные многочлены заданной степени над F_2 и проверяющие их на примитивность [19][20][21]. К таким алгоритмам относится следующий алгоритм:

Алгоритм 2.

1. Выбрать полином $f_i(x)$ степени r из проиндексированного множества $\{0, 2^r - 1\}$ и удалить его из множества

2. Построить LFSR на основе $f_i(x)$ и проверить его период. Если период не равен $2^r - 1$, вернуться на шаг 1
3. Вывести $f_i(x)$ как примитивный
4. Продолжать пока множество не пусто

В **четвертом разделе** описываются линейные параметры последовательностей, такие как линейная сложность и линейный профиль.

Линейной сложностью последовательности γ называется степень её минимального многочлена. Обозначим её $\Lambda(\gamma)$. Фактически линейная сложность представляет собой длину минимального сдвигового регистра, способного воссоздать заданную последовательность, и, следовательно, равна количеству фактически используемых бит ключа, которые придется перебирать злоумышленнику, поскольку не все аннулирующие многочлены будут являться минимальными для последовательности.

Линейным профилем (*профилем линейной сложности*) последовательности $\gamma = \{\gamma_x\}$, $x = 1, 2, \ldots, q$ называется последовательность вида $\{\Lambda_k(\gamma)\}$, $k = 1, 2, \ldots, q$ где $\Lambda_k(\gamma)$ это линейная сложность подпоследовательности последовательности γ, имеющей вид $\{\gamma_1, \gamma_2, \ldots, \gamma_k\}$.

Линейный профиль обнаруживает больше недостатков псевдослучайной последовательности, нежели просто линейная сложность. Так, для последовательности из вида $(a, a, \ldots, a, b)$ линейный профиль будет иметь вид (при $a \neq 0$, a и b не взаимно простые) : $(1, 1, \ldots, 1, q - 1)$.

Утверждение. Пусть дано поле F_p и линейные рекуррентные последовательности $\alpha_k \in F_p$, k = 1, 2, …, m с соответствующими им линейными сложностями n_k и минимальными многочленами $f_k(\lambda)$ соответственно. Докажем, что их сумма α имеет линейную сложность $n_1 + n_2 + \ldots + n_m$, если $(f_1(\lambda), f_2(\lambda), \ldots, f_m(\lambda)) = 1$.

В **пятом разделе** работы предлагается алгоритм построения генератора последовательности с требуемой линейной сложностью.

Опираясь на утверждение из предыдущего раздела, можно предложить следующий алгоритм генерации LFSR с гарантированной линейной сложностью, используемый в реализации генератора псевдослучайной последовательности по заданной линейной сложности в приложении D.

Алгоритм предполагает наличие списка минимальных многочленов $f_1(\lambda)$, $f_2(\lambda)$, …, $f_N(\lambda)$ где $\deg(f_i(\lambda)) = i$, то есть списка минимальных многочленов всех степеней от 1 до N, которые могут быть найдены при текущих ограничениях в вычислительных возможностях, либо получены из некоторого источника и получает на вход положительное число n – требуемую линейную сложность. При значениях n от 1 до N, алгоритм строит регистр соответствующий n-ому многочлену из списка. В случае если значение n выходит за пределы списка, алгоритм, если это возможно, находит разложение числа в виде суммы:

$$n = \sum_{i=0}^{k} n_i$$

где $0 < k$, $n_i < N + 1$, и $\left(f_{n_1}(\lambda), f_{n_2}(\lambda), \ldots, f_{n_k}(\lambda)\right) = 1$, то есть раскладывает число n в сумму степеней взаимно простых многочленов из списка. В случае если такого разложения не существует – задача неразрешима в связи с недостаточной длиной списка. Если же разложение найдено, алгоритм находит произведение многочленов соответствующих разложению и генерирует LFSR, основанный на этом произведении.

В **шестом разделе** исследуется проблема построения последовательности имеющей линейную сложность, достаточную для использования в бинарно-аддитивном шифре. Здесь также рассматривается проблема проявления дефектов последовательности в виде нулевых и единичных подпоследовательностей при шифровании, предлагается алгоритм исправления таких дефектов.

Несмотря на то, что предложенный в разделе генератор псевдослучайной последовательности имеет хорошие статистические показатели, как это будет продемонстрировано в описании реализации, его практическое применение

может привести к плохим результатам. В частности, так как 256 ячеек генератора могут принимать значения из подмножества множества всех возможных ненулевых, существует вероятность генерации значительных по длине (до 255 бит) участков последовательности, состоящих только из нулей или единиц (далее нулевые и единичные *S*-граммы соответственно, где *S* – это длина *S*-граммы). Так как многочлен $G(x)$, получаемый в результате построения в алгоритме, не является примитивным, он принимает не все возможные ненулевые значения в рамках своего периода, а лишь их часть. Поэтому вероятность появления, к примеру, состояния с нулевой *S*-граммой для некоторого *S*, если такое состояние входит в период $G(x)$, может быть значительно увеличена. В разделе предлагаются алгоритм построения последовательности с линейной сложностью 256 и алгоритм исправления единичных *S*-грамм.

Седьмой раздел посвящен описанию и тестированию разработанного в ходе выполнения работы криптосредства QuickCrypt, предназначенного для шифрования и безопасного удаления данных и реализованного на языке C\C++ с использованием библиотек QT версии 5.5.0 а также функций WinAPI.

QuickCrypt предоставляет пользователю возможность генерировать ключевые файлы, шифровать, расшифровывать, и безвозвратно удалять данные с жесткого диска.

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной работы рассмотрены основные проблемы проектирования потоковых шифров, связанные с построением псевдослучайных последовательностей на основе сдвиговых регистров с линейной обратной связью, такие как поиск примитивных полиномов, обеспечение необходимой линейной сложности последовательности, исправление дефектов последовательности в виде излишне длинных нулевых или единичных подпоследовательностей. Рассмотрены основные алгоритмы поиска примитивных полиномов, один из которых был реализован в приложении А. Также были найдены все примитивные полиномы над полем F_2 до двадцатой степени включительно, часть из которых была представлена в приложении В, были исследованы основные линейные характеристики последовательностей, реализован алгоритм их определения. Были предложены, реализованы и протестированы алгоритм построения сдвиговых регистров с фиксированной линейной сложностью и алгоритм исправления дефектов последовательности. На основе этих алгоритмов было разработано и протестировано программное средство QuickCrypt, реализующее потоковый шифр и механизм безопасного удаления данных.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Яковлев, А.В. Криптографическая защита информации : учебное пособие / А.В. Яковлев, А.А. Безбогов, В.В. Родин, В.Н. Шамкин. – Тамбов : Изд-во Тамб. гос. техн. ун-та, 2006. – 140 с.

2. Кафедра Алгебры и Компьютерной топологии.Методические пособия [Электронный ресурс] / С. В. Яблонский, Введение в дискретную математику. – URL: http://topology.math.csu.ru/posob.htm (дата обращения: 13.10.2015).

3. Centre For Applied Cryptographic Research: The University of Waterloo [Электронный ресурс] / Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. Handbook of Applied Cryptography. – URL: http://cacr.uwaterloo.ca/hac/ (дата обращения: 13.10.2015).

4. Department of Computer Science and Engineering, HKUST [Электронный ресурс] / Cunsheng Ding, Wenpei Si. Binary Additive Counter Stream Ciphers. – URL: http://www.cse.ust.hk/faculty/cding/CONFERENCE/countercipher.pdf (дата обращения: 13.10.2015).

5. Rick Wash [Электронный ресурс] / Lecture notes on stream ciphers and RC4. – URL: http://rickwash.com (дата обращения: 13.10.2015).

6. Нечаев В.И. Элементы криптографии (Основы теории защиты информации): Учеб. пособие для ун-тов и пед. вузов/ Под ред. В.А. Садовничего – М.: Высш. шк.,1999.

7. Лидл Р., Пильц Г. Прикладная абстрактная алгебра: Учеб. пособие / Пер. с англ. – Екатеринбург: Изд-во Урал. ун-та, 1996.

8. Акритас А. Основы компьютерной алгебры с приложениями: Пер. с англ. — М., Мир, 1994. — 544 с.

9. Computer-Aided Engineering | Computing in the College of Engineering [Электронный ресурс] / Kewal K. Saluja. Linear Feedback Shift Register Theory and Applications. – URL: http://homepages.cae.wisc.edu/~ece553/handouts/LFSR-notes.PDF (дата обращения: 13.10.2015).

10. Penn Engeneering [Электронный ресурс] / LFSR and Encryption. – URL: http://www.cis.upenn.edu/~cis110/13sp/hw/hw04/lfsr.shtml (дата обращения: 13.10.2015).
11. It Should Work... [Электронный ресурс] / Breaking LFSR-based pseudo-random number generators. – URL: http://vierito.es/wordpress/2011/01/22/breaking-lfsr-based-pseudo-random-number-generators/ (дата обращения: 13.10.2015).
12. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы и исходные тексты на языке Си. –М.: Издательство ТРИУМФ, 2013 – 816 с.: ил.
13. EECS Instructional Support Group Home Page [Электронный ресурс] / Components and Design Techniques for Digital Systems. – URL: https://inst.eecs.berkeley.edu/~cs150/sp03/handouts/15/LectureA/lec27-2up.pdf (дата обращения: 13.10.2015).
14. Cryptography [Электронный ресурс] / Linear feedback shift register (LFSR) sequence commands. – URL: http://doc.sagemath.org/html/en/reference/cryptography/sage/crypto/lfsr.html (дата обращения: 13.10.2015).
15. The Easy Way [Электронный ресурс] / Understanding Linear Feedback Shift Registers. – URL: http://www.yikes.com/~ptolemy/lfsr_web/ (дата обращения: 13.10.2015).
16. Department of Electrical & Computer Engineering - University of Alberta [Электронный ресурс] / Linear Feedback Shift Register. – URL: http://www.ece.ualberta.ca/~elliott/ee552/studentAppNotes/1999f/Drivers_Ed/lfsr.html (дата обращения: 13.10.2015).
17. World Academy of Science, Engineering and Technology [Электронный ресурс] / Deepthi P.P., P.S. Sathidevi . Hardware Stream Cipher Based on LFSR and Modular Division Circuit – URL: http://waset.org/publications/550/hardware-stream-cipher-based-on-lfsr-and-modular-division-circuit (дата обращения: 13.10.2015).

18. International Journal of Computer Applications IJCA [Электронный ресурс] / An Faheem Masoodi, Shadab Alam. Analysis of Linear Feedback Shift Registers in Stream Ciphers. – URL: http://research.ijcaonline.org/volume46/number17/pxc3879714.pdf (дата обращения: 06.10.2015).
19. The Object Server Home Page [Электронный ресурс] / Information on Primitive and Irreducible Polynomials – URL: : http://theory.cs.uvic.ca/inf/neck/PolyInfo.html (дата обращения: 13.10.2015).
20. Sean Erik O'Connor - Home Page and Free Mathematical Software. [Электронный ресурс] / Computing Primitive Polynomials - Theory And Algorithm. – URL: http://www.seanerikoconnor.freeservers.com/Mathematics/AbstractAlgebra/PrimitivePolynomials/theory.html (дата обращения: 13.10.2015).
21. Home page Matematickog fakulteta [Электронный ресурс] / A Table of Primitive Binary Polynomials. – URL: http://poincare.matf.bg.ac.rs/~ezivkovm/publications/primpol1.pdf (дата обращения: 13.10.2015).
22. Technical Papers from Stanford CRC [Электронный ресурс] / Saxena and McCluskey. Primitive polynomial generation algorithms implementation and performance analysis. – URL: http://crc.stanford.edu/crc_papers/CRC-TR-04-03.pdf (дата обращения: 13.10.2015).
23. Mathematics at Ghent University [Электронный ресурс] / Machine-readable Cunningham tables. – URL: http://cage.ugent.be/~jdemeyer/cunningham (дата обращения: 13.10.2015).
24. UMD Department of Computer Science [Электронный ресурс] / Hu Qi. Stream Ciphers and Linear Complexity. – URL: https://www.cs.umd.edu/~huqi/MasterThesisR.pdf (дата обращения: 13.10.2015).
25. Signal and Information Processing Laboratory (ISI) [Электронный ресурс] / James L. Massey, Shirlei Serconek. Linear Complexity of Periodic Sequences: A

General Theory. – URL: http://www.isiweb.ee.ethz.ch/archive/massey_pub/pdf/BI334.pdf (дата обращения: 13.10.2015).
26. Computer Science Department - University of Kentucky [Электронный ресурс] / Andrew Klapper. Linear Complexity of Finite Field Sequences over Different Fields. – URL: https://www.cs.uky.edu/~klapper/pdf/lcdiff.pdf (дата обращения: 13.10.2015).
27. School of Mathematics [Электронный ресурс] / Erin Casey. Berlekamp-Massey algorithm.–URL: http://www.math.umn.edu/~garrett/students/reu/MB_algorithm.pdf (дата обращения: 06.10.2015).
28. Фомичев В. М. Дискретная математика и криптология. Курс лекций / Под общ. ред. д-ра. физ.–мат. н. Н. Д. Подуфалова. – М.:ДИАЛОГ- МИФИ, 2003 – 400 с.
29. NIST.gov - Computer Security Division - Computer Security Resource Center [Электронный ресурс] / Andrew Rukhin, Juan Soto. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications . – URL: http://csrc.nist.gov/publications/nistpubs/800-22-rev1a/SP800-22rev1a.pdf (дата обращения: 06.10.2015).
30. School of Mathematics [Электронный ресурс] / Dan Biebighauser. Testing random numbers generators – URL: http://www.math.umn.edu/~garrett/students/reu/pRNGs.pdf (дата обращения: 06.10.2015).