УДК 004.056.55

Система криптографической генерации идентичных данных на основе алгоритма Диффи-Хеллмана

Елисеев С.О.*, Крюков Д.А.**

Московский Технологический Университет,

проспект Вернадского, 78, Москва, 119454, Россия

**e-mail: ideawade@gmail.com*

*** e-mail: dm.bk@bk.ru*

Аннотация

В статье рассмотрены вопросы реализации и исследования системы криптографической генерации данных от источника сообщения к его получателю. Научная новизна исследования состоит в том, что представлен и обоснован алгоритм защищённого приёма-передачи данных явно не использующий ключей шифрования. Проведен анализ скорости генерации данных в зависимости от различных характеристик, определены пути повышения его эффективности, предложены методы предотвращения активных атак на алгоритм.

Ключевые слова: алгоритм Диффи-Хеллмана, асимметричное шифрование, безопасность данных, однонаправленные функции

Введение

Общеизвестно, что существует два классических подхода к конфиденциальному, обратимому кодированию данных в целях их передачи или

безопасного хранения. В основе использования каждого из них лежит абстракция, именуемая ключ шифрования. В подходах, основанных на алгоритмах симметричного шифрования, для процедуры шифрования и расшифрования применяется идентичный ключ в подходах ассиметричного шифрования используются открытые и закрытые ключи, связанные односторонним преобразованием. Подходы подробно изложены в [1-4]. В статье предлагается рассмотреть работу системы генерации идентичной информации в целях обеспечения её безопасного приёма и передачи свободную от использования такой абстракции как ключ шифрования, но имеющую криптографическую защиту. Генерация данных осуществляется конфиденциально, в основе лежит фундаментальный алгоритм Диффи-Хеллмана. Все преобразования, выполняемые в указанной системе, обратимы для формирования текста на стороне получателя. Поскольку явное использование ключа шифрования не предполагается, а системе присущи все характерные свойства алгоритма шифрования: конфиденциальность, целостность и доступность, то алгоритм, лежащий в её основе можно назвать алгоритмом квазишифрования данных, поскольку шифруются не данные, а сведения о данных (метаданные) или так называемым алгоритмом бесключевого шифрования [5]. Система генерации идентичной информации направлена на совершенствование походов применения алгоритмов асимметричной криптографии, и представляет собой решение, повышающее безопасность информационного обмена, конфиденциальность доступа и передачи данных с сохранением целостности и доступности. Новизна предлагаемой системы заключается в реализации полного

цикла защищённого обмена данными посредством операций с метаданными открытого текста без использования линейных и нелинейных преобразований открытого текста и ключей шифрования.

1. Математическое обоснование метода

Открытое распространение вспомогательных элементов, безопасно образующих общий секрет на основании неразрешимости задачи дискретного логарифмирования [6], позволяет паре различных пользователей системы выработать единую последовательность символов в незащищенном канале связи. Целесообразность данного подхода обуславливается тем, что криптостойкость такой системы основывается на высокой вычислительной сложности обращения показательной функции (1). Она вычисляется достаточно эффективно, в то время как даже самые современные алгоритмы решения задачи дискретного логарифмирования являются малоэффективными. При большом значении a функции (2) нахождение решения для такого уравнения потребует значительных временных ресурсов [7-8].

$$f(x) = a^x \quad (1)$$

$$A = g^a mod\ p \quad (2)$$

Следовательно, система, использующая подобный метод распределения общего секрета, будет обладать высокой криптостойкостью [9]. Данные соотношения, заданные односторонними функциями, лежат в основе ассиметричной криптографии. Вместе с тем, их особенности позволяют рассмотреть возможность защищенной передачи данных без использования ключей шифрования.

В основе работы системы, предложенной в статье, лежит фундаментальный алгоритм Диффи-Хеллмана, с помощью которого возможно образовать общий для участников информационного обмена «секрет» – целочисленное значение. Данное значение, идентичное на обеих сторонах информационного обмена, предлагается интерпретировать как последовательность символов, а не в качестве классического операнда алгебраических функций над конечными полями или эллиптическими кривыми или операции «XOR». Вместе с тем, стоит отметить, что ключевой особенностью данного алгоритма является отсутствие непосредственной передачи текста в любом его виде либо его фрагментов, обмен происходит только целочисленными значениями, необходимыми для формирования общей последовательности, а также вспомогательными параметрами, не имеющими определяющего значения для потенциального атакующего [10]. В связи с вышеизложенным полагается возможным ассоциировать процесс криптографической генерации (восстановления) идентичного текста на стороне его получателя с процессом квазишифрования данных.

2. Реализация системы криптографической генерации идентичных данных

Как было упомянуто выше, в основе реализации метода криптографической генерации идентичных данных лежит работа У. Диффи и М. Хеллмана [11], посвященная функционированию односторонней функции с секретом. Суть работы состоит в том, что любой пользователь информационного обмена сможет вычислить функцию $f_C(x)$, но только обладатель секрета C сможет вычислить обратную

функцию $f_c^{-1}(x)$. В общем случае алгоритм Диффи-Хеллмана выглядит следующим образом (3):

$$C = B^a mod\ p \qquad (3)$$
$$C = A^b mod\ p$$

где, a – секрет пользователя 1, b – секрет пользователя 2, A - открытое число пользователя 1, B – открытое число пользователя 2, p – заранее выбранное и известное обоим пользователям просто число и C – результат формирования общего секрета.

В рамках рассматриваемой системы полученное число C предлагается не ассоциировать с ключом шифрования или аргументом для его получения, а рассматривать как последовательность байтов. Очевидно, что последовательность байтов, полученная подобным образом, эквивалентная для обеих сторон информационного обмена, не будет представлять собой осмысленный текст, но в этой последовательности могут находиться n случайных подпоследовательностей длиной S_1, S_2 ... S_n, которые нас интересуют с точки зрения алгоритма генерации текста. Таким образом, назовем данную последовательность C-последовательностью, а ее подпоследовательности S_i - S-фрагментами.

Итак, на стороне получателя (Боба) необходимо восстановить сообщение отправителя (Алисы) без непосредственной его передачи. Учитывая то обстоятельство, что у Алисы и Боба имеются идентичные последовательности, полагается целесообразным пошагово выполнять сравнение С-последовательности с текстом, который Алиса намерена воспроизвести у Боба. Сравнение производится целью, поиска общих S-фрагментов, входящих в С-последовательность и в

сообщение, которые можно использовать в информационном обмене неявно, то есть, информацию о которых можно передавать от Алисы к Бобу. Для повышения эффективности поиска S-фрагментов предлагается использовать алгоритм построения суффиксного массива. Используя суффиксный массив, можно перебирать входной текст неограниченное число раз, начиная с перебора по одному символу, смещаясь каждую итерацию на один символ, и заканчивая в тот момент, когда подстроки заданной длины будут отсутствовать. Это будет означать, что C-последовательность деградировала. В тот момент, когда совпадения будут найдены, необходимо установить индекс начала S-фрагмента в C-последовательности (его смещение) и длину самого S-фрагмента. Эти параметры будут использоваться для передачи собеседнику и последовательного восстановления фрагментов открытого текста. То есть при получении таких параметров от Алисы, Боб, зная смещение и длину S-фрагмента, сможет обнаружить в своей, общей с Алисой ключевой последовательности, необходимый S-фрагмент и использовать его в процедуре восстановления текста.

Далее, в тот момент, когда C-последовательность деградирует (будут отсутствовать совпадения текста и ключевой последовательности), а текст не до конца будет сгенерирован на стороне получателя, появится необходимость инициировать создание новой C-последовательности посредством алгоритма Диффи-Хеллмана. Таким образом, ключевая последовательность может меняться несколько раз за один сеанс связи, что повышает безопасность системы в целом подобно ротации сеансовых ключей в традиционных криптоалгоритмах.

Схема состояний системы криптографической генерации идентичных данных и их переходов показана на рисунке 1.

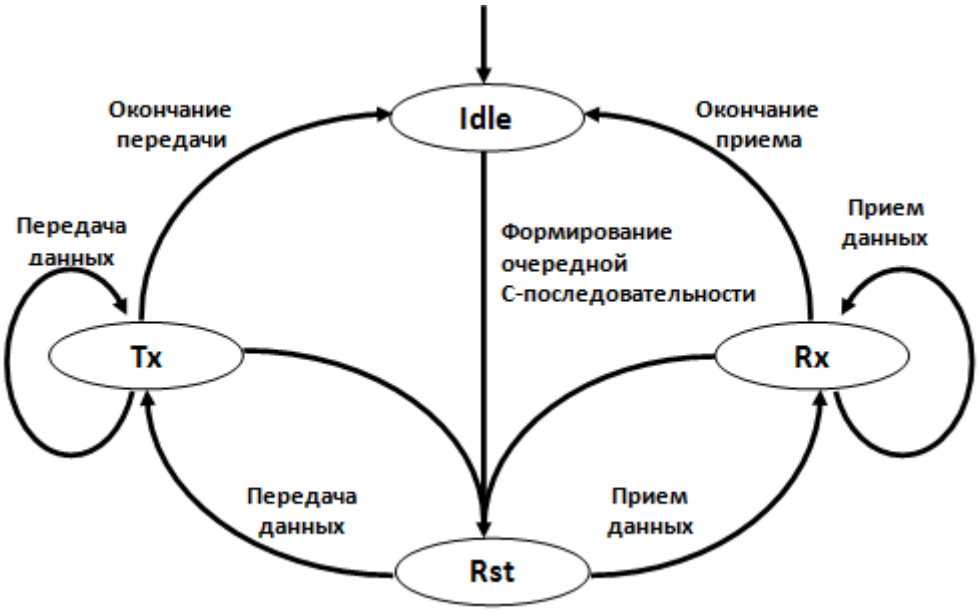

Рис. 1. Схема состояний системы

Множество состояний системы образовано следующими элементами:

- Idle – ожидание (начальное состояние);
- Tx – передача данных;
- Rx –прием данных;
- Rst – формирование очередной С-последовательности.

Условием перехода в состояние Rst является обстоятельство, свидетельствующее о необходимости формирования очередной С-последовательности, в канале связи. Этот процесс может быть реализован путем обмена значениями [11], необходимыми для работы алгоритма Диффи-Хеллмана.

Ситуации неоднозначности выбора состояния перехода не возникнет в связи с тем, что множество условий перехода в состояние Rst не пересекается с множеством

условий перехода в состояния передачи/приема данных и состояние Idle по причине различной длинны векторов элементов.

Блок схема работы алгоритма в части передачи информации [12] показана на рисунке 2.

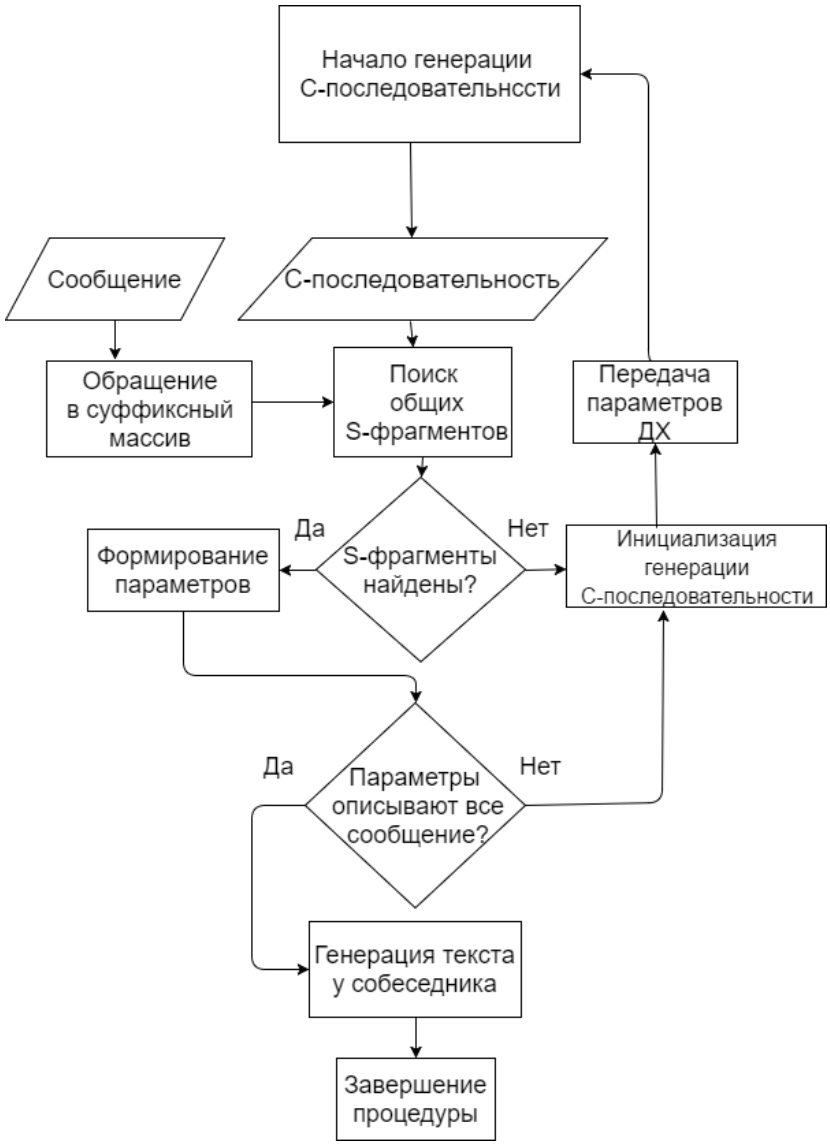

Рис. 2. Блок-схема передачи информации

3. Исследование работы системы

В целях исследования производительности работы предложенного алгоритма подготовлен макет программного обеспечения в форме сетевого приложения.

Работа системы строится по принципу взаимодействия точка-точка (P2P). Каждый экземпляр данного приложения, реализует одновременную функциональность в роли приемника и передатчика.

Время генерации информации на стороне получателя в зависимости от объема сообщения представлено в таблице 1, измерения проводились для С-последовательности длиной 128 бит. Измерения в количестве 10 итераций проводились для случайного сообщения длиной 10, 140, 1000, 2000, 3000 байтов.

Таблица 1

Время работы (мс.) алгоритма для С-последовательности длиной 128 бит

Сообщение из 10 байтов	**Сообщение из 140 байтов**	**Сообщение из 1000 байтов**	**Сообщение из 2000 байтов**	**Сообщение из 3000 байтов**
407	560	816	213	256
82	232	217	461	405
432	409	300	737	424
507	350	97	803	977
578	599	867	340	674
978	290	449	138	924
431	19	367	283	76
111	545	290	422	75
640	985	937	986	949
843	276	896	59	298

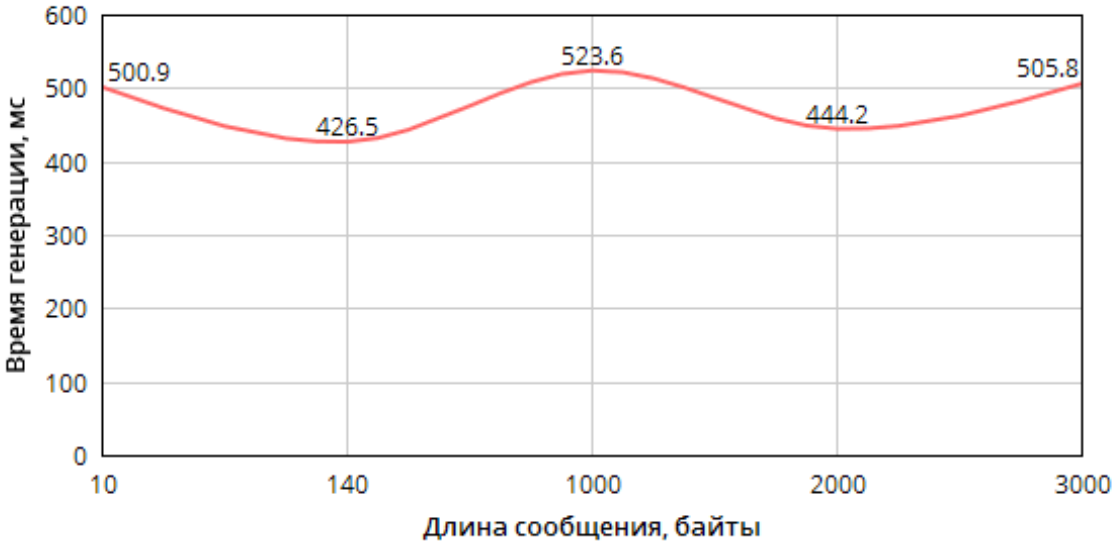

Рис. 3. График среднего времени работы системы для сообщений разной длины

Усредненные результаты анализа сведены в график на рисунке 3. Ниже представлены зависимости времени генерации сообщений для С-последовательностей длиной 96, 196, 384 и 512 бит.

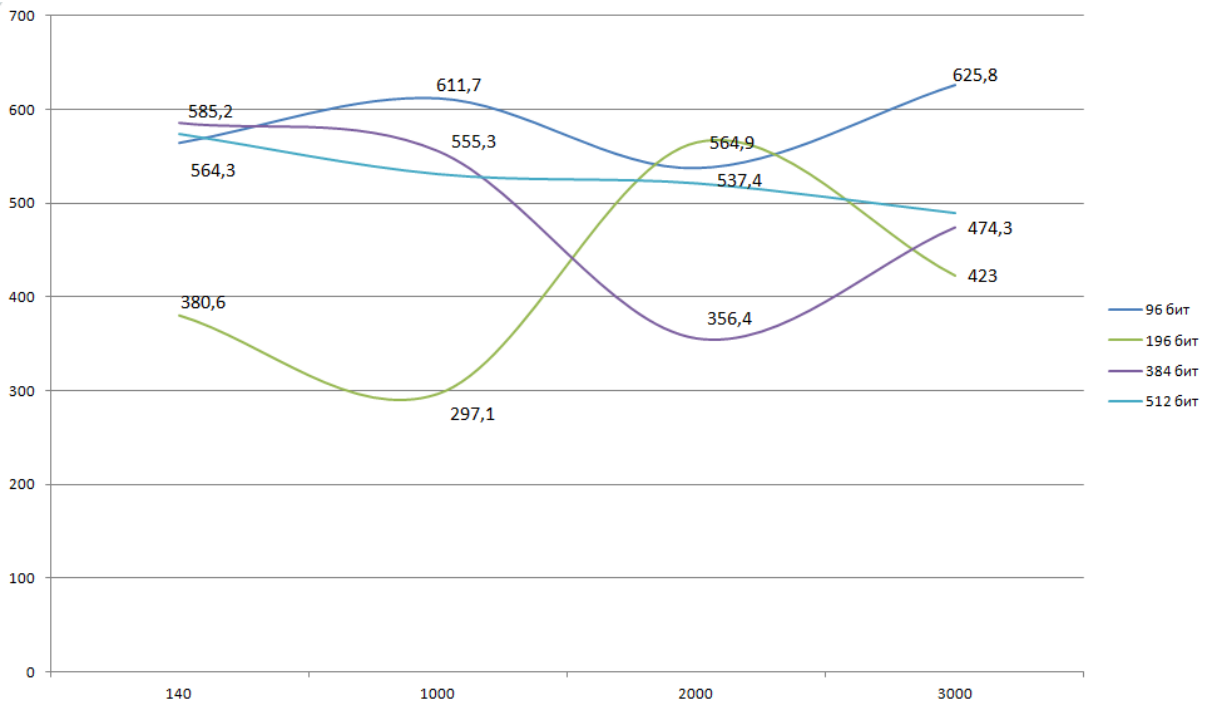

Рис. 4. Графики среднего времени генерации для сообщений разной длины

Как видно, время генерации сообщения не зависит от его длины и не превышает секунды. Данное обстоятельство объясняется тем, что любой алфавит

конечен, и, соответственно, время генерации сообщения будет ограничено временем обработки всех комбинаций допустимых символов алфавита того языка, на котором мы передаем сообщение.

Далее проведен анализ частоты деградации С-последовательности для генерации случайного сообщения. Результаты анализа представлены в таблице 2.

Таблица 2

Частота деградаций С-последовательности длиной 128 бит за один сеанс передачи сообщения

Сообщение из 10 байт	**Сообщение из 140 байт**	**Сообщение из 1000 байт**	**Сообщение из 2000 байт**	**Сообщение из 3000 байт**
5	6	5	9	7
6	5	6	3	7
2	5	5	7	6
8	6	5	4	10
3	5	6	5	4
2	4	6	11	8
5	6	5	9	7
2	7	12	7	12
3	10	9	6	5
8	5	7	5	8

Усредненные результаты анализа сведены в график на рисунке 5. Как видно, зависимость имеет экспоненциальный характер.

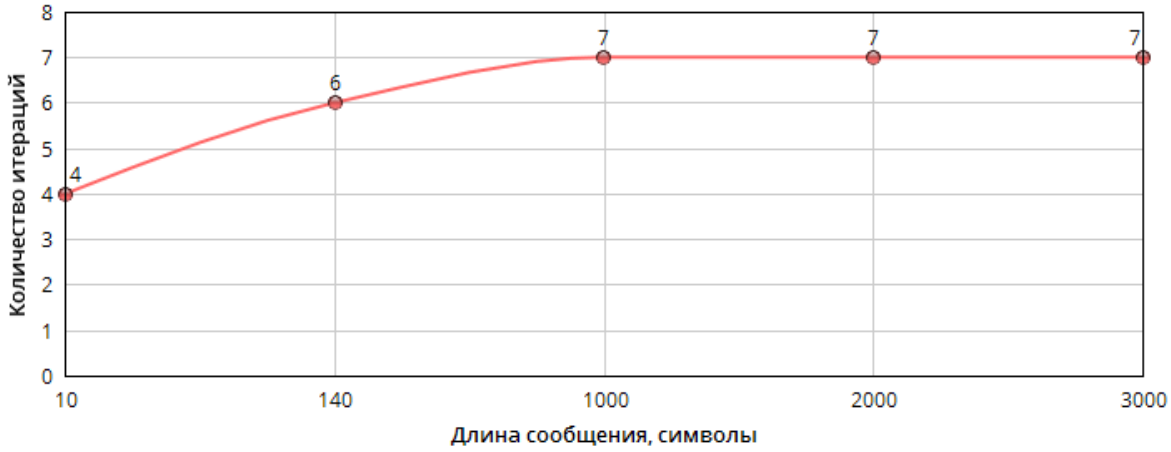

Рис. 5. График среднего числа итераций алгоритма для сообщений разной длины

С целью установить насколько длина выбранной С-последовательности влияет на скорость генерации сообщения, проведен анализ частоты итераций С-последовательности для сообщения длиной 140 символов и С-последовательности длиной 96, 128, 192, 384 и 512 битов. Результаты усредненных вычислений сведены в график 6. Таким образом, прослеживается гиперболическая зависимость количества итераций С-последовательности в зависимости от её длины.

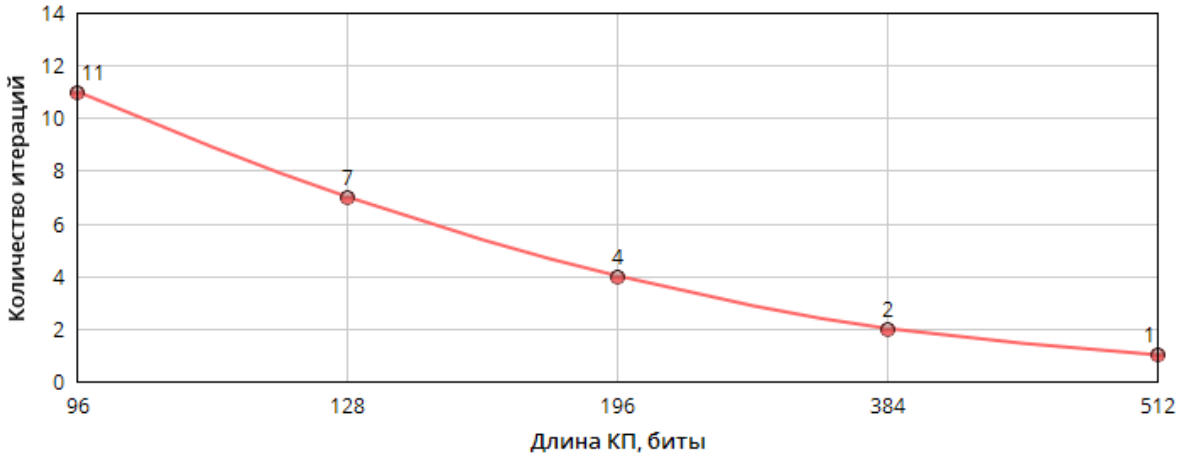

Рис. 6. График среднего количества итераций С-последовательности для генерации сообщения длиной 140 байт

Очевидно, что обмен информацией об S-фрагментах, состоящих из символа, может превратить алгоритм генерации данных в небезопасный шифр перестановки

[13], где каждому значению (в пределах одной С-последовательности) будет поставлено в соответствие пара значений. Поэтому одним из направлений повышения безопасности алгоритма является требование по использованию более длинных S-фрагментов. Для того чтобы исследовать то, насколько изменение длины S-фрагмента влияет на скорость генерации в целом, рассмотрена зависимость времени генерации (секунды) от минимальной длины S-фрагмента (байты). Полученный график представлен на рисунке 7. В исследовании использована С-последовательность длиной 1536 бит.

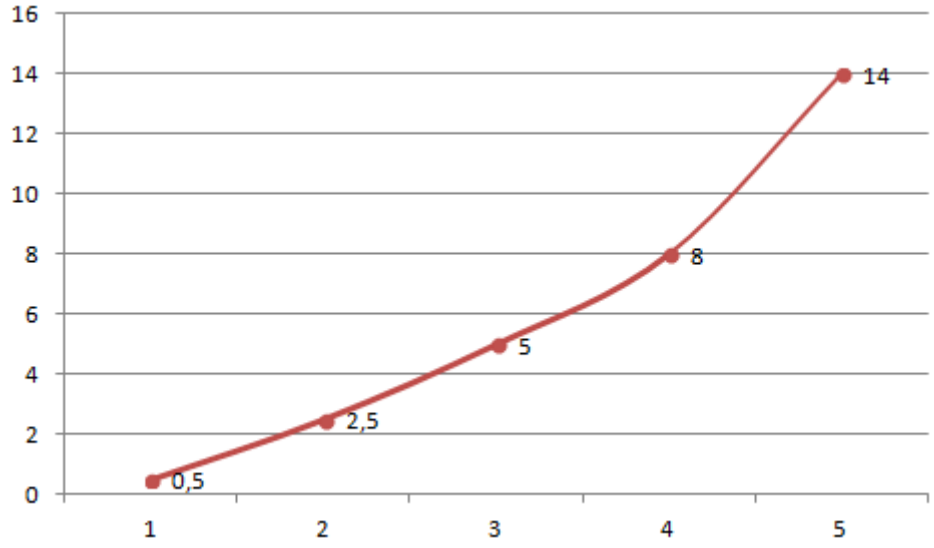

Рис. 7. Зависимость времени генерации от длины S-фрагмента

Исходя из графика, можно подтвердить заключение, что увеличение длины S-фрагмента не только обеспечит системе большую криптостойкость, но и замедлит скорость генерации. Повысить скорость генерации возможно посредством увеличения длины С-последовательности. Как следствие увеличится вариативность потенциальных S-фрагментов. Для решения данной задачи целесообразно действовать в двух направлениях: увеличивать разрядность вновь генерируемой С-последовательности или применять методы эволюционных вычислений [14-16] при

выработке новых С-последовательностей каждой следующей итерации алгоритма, например, основные биологические операторы: селекция, скрещивание, мутация [17]. Кроме того, возможно видоизменять С-последовательности или производить идентичные операции над ними на обеих сторонах информационного обмена. В общем случае варианты не являются взаимоисключающими. Второе направление решения задачи может стать развитием данной работы.

В рамках исследования осуществлялась генерация одного вида данных. Для наглядности в качестве объекта генерации использованы текстовые сообщения, однако следует отметить, что алгоритм сохранит функциональность для любых категорий данных. Рекомендуется использовать алгоритмы сжатия данных перед непосредственной генерацией для исключения фрагментов текста, характерных для различных алфавитов, кодировок, форматов файлов, а так же сокращения времени восстановления данных.

4. Методы защиты от активного воздействия

Как было сказано ранее, алгоритм Диффи-Хеллмана, используемый в системе для получения С-последовательности реализованный по требованиям самих У. Диффи и М. Хеллмана, а также, если в соответствии с требованиями NIST [18] используются эллиптические кривые, устойчив к атакам прямого перебора и подслушивания, но не может обеспечить защиту от активного прослушивания канала связи, то есть возможности внесения изменений (подлога) в информационный обмен сторон (т.н. атаки «человек-посередине» [19]). Для

обеспечения безопасности пользователей от атаки «человек посередине» предлагается два варианта. Традиционный подход связан с подписыванием компонентов образования общей С-последовательности сертификатами открытого ключа стандарта X.509 [1,4,20]. При этом следует принять во внимание необходимость существования центра сертификации [1,4,20], которому будут «доверять» обе стороны. Второй вариант заключается в использовании дополнительного канала связи между собеседниками и сервера авторизации. Во время получения С-последовательности оба собеседника отправляют на сервер сокращенный результат применения криптографической хеш-функции [4,21] от С-последовательности, который может быть однозначно преобразован в пару символов или слов.

Эти последовательности будут направлены абонентам потенциального информационного обмена с целью взаимной проверки отсутствия гипотетического атакующего субъекта посредством дополнительного канала связи (например, голосового). Если хеш-значения, полученные сервером различны, то существует вероятность того, что канал прослушивается и в механизм образования С-последовательности были внесены подложные значения прослушивающим субъектом.

Заключение

В статье рассмотрен подход к реализации системы криптографической генерации идентичных данных. Разработано программное обеспечения, в форме сетевого приложения, реализующего функциональность приемника и передатчика.

Работа системы строится по принципу взаимодействия точка-точка в соответствии с предложенной схемой перехода состояний. Проведено исследование различных характеристик работы системы. Установлено, что система обеспечивает возможность конфиденциальной и целостной передачи данных при выполнении минимально достаточных требований по длине S-фрагментов. По сравнению с аналогами разработанная криптосистема обладает относительно малой скоростью при передаче больших массивов информации, обладающих высокой энтропией распределения байтов, но при этом остается относительно устойчивой благодаря применению однонаправленных функций, реализованных посредством алгоритма Диффи-Хеллмана, и использованию метаданных фрагментов открытого текста. В качестве направлений дальнейшего исследования системы криптографической генерации идентичных данных целесообразно выделить задачи повышения вариативности комбинаций С-последовательности и поиска оптимального значения длины S-фрагментов по отношению к времени генерации и устойчивости системы к атакам.

Библиографический список

1. Schneier B. Applied Cryptography: Protocols, Algorithms and Source Code in C, John Wiley & Sons, 2015, 784 p.

2. Menezes A., Van Oorscot P., Vanstone S. Handbook of Applied Cryptography. Boca Raton, CRC Press, 1996, 780 p.

3. Глебов О.И. Специализированная система электронного документооборота // Труды МАИ. 2005. № 18. URL: http://trudymai.ru/published.php?ID=34190

4. Мао В. Современная криптография: Теория и практика. – М.: Вильямс, 2005. – 768 с.

5. Муравьев А.В., Березин А.Н., Молдовян Д.Н. Протокол стойкого шифрования сообщений с использованием коротких ключей // Известия высших учебных заведений. Приборостроение. 2014. № 57. С. 68 - 72.

6. Buchmann J., Jacobson M., Teske E. On some computational problems in finite abelian groups // Mathematics of Computation, 1997, vol. 66, no. 220, pp. 1663 - 1687.

7. Гречников Е.А. Двусторонние оценки числа неподвижных точек дискретного логарифма // Вестник Московского университета. Математика. Механика. 2012. № 3. С. 3 – 8.

8. Boppana R.B., Lagarias J.C. One-way functions and circuit complexity // Information and Computation, 1987, vol. 74, no. 3, pp. 226 - 240, doi:10.1016/0890-5401(87)90022-8.

9. Sipser M. Introduction to the Theory of Computation, Thomson Course Technology, 2006, 431 p.

10. Елисеев С.О., Крюков Д.А. Об одном подходе к реализации бесключевого шифрования данных в информационных системах // XVI научно-практическая конференция «Современные информационные технологии в управлении и образовании» (Москва, 20 апреля 2017): Сборник научных трудов. - М.: ФГУП НИИ «Восход», 2017. С. 156-163.

11. Diffie W., Hellman M. New Directions in Cryptography // IEEE Transactions on Information Theory, 1976, vol. 22, pp. 644 - 654.

12. Елисеев С.О., Крюков Д.А. Перспективы использования бесключевого шифрования данных в информационных системах // XV научно-практическая конференция «Современные информационные технологии в управлении и образовании». (Москва, 21 апреля 2016). Сборник научных трудов. - М.: НИИ «Восход», 2016. С. 76 - 82.

13. Бабаш А.В., Шанкин Г.П. Криптография. – М.: Солон-пресс, 2007. – 512 с.

14. Holland J.N. Adaptation in Natural and Artificial Systems. An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence, Cambridge, MIT Press, 1992, 232 p.

15. Goldberg D. Genetic Algorithms in Search, Optimization and Machine learning, Boston, Addison-Wesley, 1989, 432 p.

16. Mitchell M. An introduction to Genetic Algorithm, Cambridge, MIT Press, 1999, 158 p.

17. Метлицкая Д.В. Генетические алгоритмы поиска оптимального управления непрерывными детерминированными системами // Труды МАИ. 2011. № 45. URL: http://trudymai.ru/published.php?ID=25544

18. Barker E., Chen L., Roginsky A., Vassilev A., Davis R. Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, National Institute of Standards and Technology, 2013, pp. 21-22, doi:10.6028/nist.sp.800-56ar3.

19. Козлов В.А., Рындюк В.А., Воробьев Г.А., Чернышев А.Б. Модели и методы защиты от атак «man in the middle» (MITM) // Современные фундаментальные и прикладные исследования. 2017. № 24. С. 27 - 35.

20. Gielesberger M. Alternatives to X.509, Münich, Technical University of Münich, 2013, pp. 51 - 52.

21. Лапонина О.Р. Криптографические основы безопасности. - М.: Интуит, 2016. – 242 р.