УДК *004.421*

Г.Н. Шурховецкий[1]

[1]Иркутский государственный университет путей сообщения, г. Иркутск, Российская Федерация

ЗАЩИТА ИНФОРМАЦИИ ВО ВНЕШНИХ ХРАНИЛИЩАХ ДАННЫХ МЕТОДОМ РАССЕЧЕНИЯ-РАЗНЕСЕНИЯ

Аннотация. *Сегодня ввиду большого распространения облачных технологий всё больше физических и юридических лиц пользуются их услугами. Общепринятой практикой в области облачных технологий является хранение информации (данных) у одного поставщика услуг в одном физическом и географическом местоположении. При этом можно задействовать внушительный набор инструментов по организационным и техническим мерам обеспечения защиты информации в «облаке», что, как правило, сводит на нет преимущества облачного хранилища. Поэтому ценная критически важная информация обязательно шифруется перед её отправкой в «облако» на стороне пользователя. Для этого можно использовать средства как простые (например, создание архивов, защищенных паролем), так и более продвинутые (ПО для создания шифрованных разделов). Для пользователя помимо обычного кодирования своей информации, существуют методы разделения информации, которые позволяют разделить их сообщение и отправить его по разным внешним хранилищам, что существенно лучше, чем просто отправить оригинал с резервными копиями по разным хранилищам. К таким методам можно отнести, например, метод рассечения-разнесения, оценке эффективности которого посвящена данная статья.*

Ключевые слова: *рассечение-разнесение, стеганография, криптоанализ, криптографическая атака, информационные технологии.*

G. N. Surowiecki[1]

[1]Irkutsk state University of railway transport, Irkutsk, Russian Federation

PROTECTION OF INFORMATION IN EXTERNAL DATA STORES USING THE DISSECTION-SEPARATION METHOD

Abstract. *Today, due to the large spread of cloud technologies, more and more individuals and businesses use their services. It is common practice in the field of cloud technologies to store in-formation (data) from one service provider in one physical and geographical location. At the same time, we must remember that the information passes through third parties whose intentions are not always known. Therefore, valuable critical information is usually encrypted before it is sent to the cloud on the user's side. To do this, you can use both simple tools (for example, creating password-protected archives) and more advanced tools (software for creating encrypted partitions). However, in addition to normal encryption, you can use the fact that there are many un-related repositories. Such methods include, for example, the dissection-separation method, which is discussed in this paper.*

Keywords: *dissection-separation, steganography, cryptanalysis, cryptographic attack, information technology.*

Введение

Общепринятой практикой в индустрии информационных технологий и связи является хранение информации (данных) у одного поставщика услуг, в одном физико-географическом месте. Часто поставщик услуг также обеспечивает избыточность и резервное хранение данных в удаленных местах, но это только для обеспечения производительности и доступности данных в случае аварии или других сбоев. Информация (данные) по-прежнему доверена этому единственному поставщику услуг, который должен реализовать множество мер безопасности для обеспечения защиты информации, а именно ее конфиденциальности, целостности и доступности. Эти меры могут носить превентивный, детективный или корректирующий характер и включать, помимо прочего, физический и логический контроль доступа, шифрование данных в состоянии покоя и при передаче, резервное копирование, мониторинг, оповещение, журнализацию, антивирус, брандмауэры, системы обнаружения и предотвращения вторжений, меры физической и экологической защиты, политики безопасности, процедуры и стан-

дарты безопасности, проверку данных, программы повышения осведомленности о безопасности, планы аудита и сертификации, реагирование на инциденты, уведомление о нарушениях, оценку рисков и т. д.

Важным элементом является шифрование. Однако оно может быть плохо реализовано, содержать бэкдоры, неизвестные пользователю или владельцу данных, а сами ключи могут быть переданы третьим лицам без ведома или согласия пользователя или владельца данных. Это делает актуальной проблему разработки методов дополнительных или даже альтернативных к шифрованию. Например, можно использовать факт существования множества не связанных между собой хранилищ. К таким методам, в частности, можно отнести, например, метод рассечения-разнесения, рассмотрим его эффективность и надёжность более детально.

Краткий обзор данной темы.

Помимо данного метода рассечения-разнесения существуют такие "классические" методы шифрования как подстановка, перестановка и гаммирование [5, 7, 8]. Они по своей сути являются линейными в том смысле, что длина зашифрованного сообщения равна длине исходного текста. Возможно нелинейное преобразование типа подстановки вместо исходных символов (или целых слов, фраз, предложений) заранее выбранных комбинаций символов другой длины [6-8]. Также в последнее время становится популярной так называемая компьютерная стеганография (от греческих слов steganos – секрет, тайна и graphy – запись), представляющая собой сокрытие сообщения или файла в другом сообщении или файле [9, 10]. Например, можно спрятать зашифрованный аудио- или видеофайл в большом информационном или графическом файле.

Но в данной статье, прежде всего, рассматривается возможность использования разных мест хранения в «облаках» или любых других носителях. С той лишь разницей, что вышеперечисленные методы позволяют хранить изменённый исходный файл в одном месте хранения, а резервные копии в другом. И если злоумышленник сможет добыть этот файл и взломать его, то цель криптоатаки будет достигнута. Рассмотренный же в данной статье метод рассечения-разнесения позволяет разбить исходный файл на несколько частей и каждую хранить в разных местах хранения и если злоумышленник сможет добыть одну часть и успешно взломать её, то достичь цели криптоатаки всё равно не сможет. Это будет возможно только при благоприятных обстоятельствах, например, если он сможет добыть все части исходного файла или большую их часть.

Общий принцип работы и суть метода рассечения-разнесения

Рассечение (разнесение) или рассечение-разнесение информации – это криптографический метод защиты информации, который заключается в том, что массив защищенных данных делится на части, каждая из которых в отдельности не позволяет раскрыть содержание защищаемой информации. Эти фрагменты можно передавать по нескольким источникам, разносить по времени и по месту записи на дискете или любом другом запоминающем устройстве. Метод рассечение-разнесение по своей реализации подразделяется на смысловое и механическое [1, 4].

Рассмотрим одну из реализаций метода [1] на примере. Защищаемый текст, представляющий из себя блок первоначальной информации, разбивается на потоки как показано, например, в табл. 1 и 2. Способ разбиения данных - это ключ метода. Ключ состоит из натуральных чисел, нумерация которых начинается с 1. Вводится два ключа: ключ строк и ключ столбцов. Произведение этих ключей равно количеству потоков, на которые будет разбит текст (например, если потоков данных 8, то ключ столбцов состоит из цифр от 1 до 4, а ключ строк - от 1 до 2 (без повторений)) [1].

Текст исходного блока разбивается на потоки путем использования расчетной формулы (определяется номер потока для записи очередного символа):

$$K\varepsilon\psi = \nu*(\rho_\iota - 1) + \sigma_\varphi,$$

где $K\varepsilon\psi$ – номер потока, ν – количество столбцов, ρ_ι – значение ι позиции ключа строки, σ_φ – значение φ позиции ключа столбца [1].

Возьмём в качестве исходного сообщения текст:

«Знание_–_столь_драгоценная_вещь,_что_его_не_зазорно_добывать_из_любого_источника»

(Фома Аквинский) и на его примере попробуем оценить эффективность метода. Всего символов в сообщении 80.

Рассечём текст на 15 потоков для чего запишем открытый текст в следующем виде (табл. 1).

Для рассечения текста на 15 потоков выбраны 3 строки и 5 столбцов. Пусть столбцы σ_φ выбираются в последовательности {4, 1, 3, 5, 2}, а строки ρ_ι – в последовательности {2, 3, 1} [3]. Символы текста разносятся по потокам, которые идентифицируются номерами. Номер к потока Ф(к), куда записывается очередной символ открытого текста, определяется по формуле:

$$\kappa = (\rho_\iota - 1) * \nu + \sigma_\varphi,$$

где ν – число столбцов [4]. Тогда первый символ «З» запишется в поток с номером ($\rho_\iota = 2$, $\sigma_\varphi = 4$): $\kappa = (2 - 1) * 5 + 4 = 9$; второй символ «н» – в поток с номером ($\rho_\iota = 2$, $\sigma_\varphi = 1$): $\kappa = (2 - 1) * 5 + 1 = 6$; третий символ «а» - в поток с номером ($\rho_\iota = 2$, $\sigma_\varphi = 3$): $\kappa = (2 - 1) * 5 + 3 = 8$; и т.д. (табл. 2)

Табл. 1. Рассечения открытого текста

	*	4	1	3	5	2	4	1	3	5	2	4	1	3	5	2	4	1	3	5	2	4	1	3	5	2	4	1	3	5	2
**		1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
2	1	**З**	**н**	**а**	**н**	**и**	**д**	**р**	**а**	**г**	**о**	**ь**	**,**	**_**	**ч**	**т**	**а**	**з**	**о**	**р**	**н**	**_**	**и**	**з**	**_**	**л**	**ч**	**н**	**и**	**к**	**а**
3	2	**е**	**_**	**–**	**_**	**с**	**ц**	**е**	**н**	**н**	**а**	**о**	**_**	**е**	**г**	**о**	**о**	**_**	**д**	**о**	**б**	**ю**	**б**	**о**	**г**	**о**	-	-	-	-	-
1	3	**т**	**о**	**л**	**ь**	**_**	**я**	**_**	**в**	**е**	**щ**	**_**	**н**	**е**	**_**	**з**	**ы**	**в**	**а**	**т**	**ь**	**_**	**и**	**с**	**т**	**о**	-	-	-	-	-

* - столбцы σ_φ выбираются в последовательности {4, 1, 3, 5, 2}

** - строки ρ_ι – в последовательности {2, 3, 1}

Табл. 2. Разнесение открытого текста (который предварительно был рассечён)

	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	**9**	**6**	**8**	**10**	**7**	**9**	**6**	**8**	**10**	**7**	**9**	**6**	**8**	**10**	**7**
2	**14**	**11**	**13**	**15**	**12**	**14**	**11**	**13**	**15**	**12**	**14**	**11**	**13**	**15**	**12**
3	**4**	**1**	**3**	**5**	**2**	**4**	**1**	**3**	**5**	**2**	**4**	**1**	**3**	**5**	**2**
Продолжение табл. 2															
1	**9**	**6**	**8**	**10**	**7**	**9**	**6**	**8**	**10**	**7**	**9**	**6**	**8**	**10**	**7**
2	**14**	**11**	**13**	**15**	**12**	**14**	**11**	**13**	**15**	**12**	-	-	-	-	-
3	**4**	**1**	**3**	**5**	**2**	**4**	**1**	**3**	**5**	**2**	-	-	-	-	-

В результате потоки Ф(к), записанные в порядке номеров, будут содержать следующие символы: Ф1 = {**о_нви...**}, Ф2 = {**_щзьо...**}, Ф3 = {**лвеас**…}, Ф4 = {**тя_ы_...**}, Ф5 = = {**ье_тт**...}, Ф6 = {**нр,зин...**}, Ф7 = {**иотнла...**}, Ф8 = {**аа_ози...**}, Ф9 = {**Здьа_ч**…}, Ф10 = = {**нгчр_к**…}, Ф11 = {**_е_ _б**…}, Ф12 = {**саобо**…}, Ф13 = {**–недо**…}, Ф14 = {**ецоою**…}, Ф15 = {**_нгог**…}. Таким образом, один поток открытого текста заменяется пятнадцатью потоками, которые в сумме дают длину блока исходного текста. Дальше полученные потоки разносятся

в разные места хранения, причём всё это является обратимым процессом. Данная реализация метода рассечение-разнесение называется механическая.

Более просто суть данного метода и его принцип деления исходного блока на потоки можно изобразить следующим образом, в качестве примера возьмём текст:

«**По_выбранной_мужчиной_невесте_легко_судить,_каков_он_и_знает_ли_он_себе_цену.**» (Иоганн Вольфганг Гёте) и на его примере рассмотрим реализацию данного метода. Всего символов в сообщении 77. Рассечение исходного сообщения происходит по ключу – 10 столбцов, а разнесение по 10 потокам (эти же столбцы, но перемешанные) (см. табл. 3).

Таблица 3. Рассечение и разнесение открытого текста

*	7	4	8	3	9	10	1	5	2	6	7	4	8	3	9	10	1	5	2	6	7	4	8	3	9	10	1	5	2	6
	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
	П	**о**	**_**	**в**	**ы**	**б**	**р**	**а**	**н**	**н**	**о**	**й**	**_**	**м**	**у**	**ж**	**ч**	**и**	**н**	**о**	**й**	**_**	**н**	**е**	**в**	**е**	**с**	**т**	**е**	**_**
Продолжение табл.3																														
*	7	4	8	3	9	10	1	5	2	6	7	4	8	3	9	10	1	5	2	6	7	4	8	3	9	10	1	5	2	6
	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
	л	**е**	**г**	**к**	**о**	**_**	**с**	**у**	**д**	**и**	**т**	**ь**	**,**	**_**	**к**	**а**	**к**	**о**	**в**	**_**	**о**	**н**	**_**	**и**	**_**	**з**	**н**	**а**	**е**	**т**
Продолжение табл.3																														
*	7	4	8	3	9	10	1	5	2	6	7	4	8	3	9	10	1	5	2	6										
	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10										
	_	**л**	**и**	**_**	**о**	**н**	**_**	**с**	**е**	**б**	**е**	**_**	**ц**	**е**	**н**	**у**	**.**	-	-	-										

* - номер потока, куда попадёт символ из исходного сообщения, выбираются в последовательности {7,4,8,3,9,10,1,5,2,6}. Всего 10 потоков.

Получаем: 1**) рчсскн_.** 2) **ннедвее** 3) **вмек_и_е** 4) **ой_еьнл_** 5) **аитуоас** 6) **но_и_тб** 7) **Пойлто_е** 8) **_ _нг,_иц** 9) **ыувок_он** 10) **бже_азну** таким образом, один блок открытого текста заменяется десятью потоками, которые в сумме дают длину блока исходного текста. Дальше полученные потоки разносятся в разные места хранения, причём всё это является обратимым процессом.

Данная реализация метода рассечение-разнесение называется механическая (заметим, что аналогичным образом можно подойти и к первому примеру, в этом случае первый и второй подход дадут одинаковый результат, а потому изложенное ниже относится и к первому случаю). Рассмотрим её стойкость.

Оценка криптографической стойкости.

Криптоанализ метода рассечение-разнесение проведём для данного случая методом грубой силы по 3 направлениям, когда злоумышленник:

1. Знает только 1 поток;
2. Знает способ реализации метода рассечение-разнесение и 1 поток из 10 исходного блока (при этом не длина блока, ни количество потоков ему неизвестны);
3. Когда злоумышленник знает все 10 потоков исходного текста, которые ему нужно собрать, а соответственно и всю длину исходного блока.

1-й способ. В случае если злоумышленнику известен только 1 поток, то ему необходимо для восстановления первоначального текста знать:

1) всю длину исходного блока;
2) в каком месте блока стоит конкретный символ;
3) подставлять символы в оставшиеся (пустые) места, чтобы восстановить сообщение, т.е. комбинировать.

Обозначим длину исходного сообщения за *N*, длину одного потока за *n* (причем, как правило, потоки имеют равную длину, расхождения могут быть незначительны), а их количество за *l* (*l=N/n*) (тогда *l-1* – количество неизвестных потоков). Обозначим наш алфавит, куда входят все варианты символов за *m*, для русского языка это будет 83 символа (буквы русского алфавита в верхнем и нижнем регистре, кроме «ё», т.к. она идентична «е», а также кроме «Ъ», «Ь», «Ы», также учитываются все цифры, знаки препинания и пробел). Допустим, злоумышленнику известен второй поток из десяти – это **ннедвее**, но он не знает, сколько всего потоков, а, следовательно, и длину всего сообщения. Поэтому вся подборка комбинаций сведётся сначала к предположению о том, что всего два потока, потом возможно три и т.д. Всего число угадываний расположения символов для известного нам второго потока (важно отметить, что злоумышленник не знает номер этого потока из общего их числа, а также, сколько их всего) (он просто имеет какой-то из потоков) в исходном сообщении в случае, если было два потока, будет равняться двум. Если три, то трём и т.д., каждый раз повышаясь на единицу с каждым новым потоком. См. на схеме 1. Это связано с тем, что рассечение происходит равномерно. Если посмотреть на табл. 3, то можно видеть, что блок исходного сообщения последовательно рассекается на 10 частей, которые разносятся на 10 потоков, 9-й символ «н» идёт во 2-й поток по ключу {7,4,8,3,9,10,1,5,2,6} и так через каждые 9 символов, каждый 10 идёт во 2-й поток. Поэтому исходя из этой периодичности рассечения блока на части и его разнесения на потоки, в предполагаемом нами исходном сообщении символы известного нам потока будут располагаться через определённый период (если всего 2 потока, то через единицу (имеется в виду в исходном блоке сообщения), если 3 потока, то через две позиции и т.д.).

Схема 1. Расположение символов известного нам потока в блоке исходного сообщения, в зависимости от их (потоков) предполагаемого числа.

н		**н**		**е**		**д**		**в**		**е**		**е**	

	н		**н**		**е**		**д**		**в**		**е**		**е**

н			**н**			**е**			**д**			**в**			**е**			**е**		

	н			**н**			**е**			**д**			**в**			**е**			**е**	

		н			**н**			**е**			**д**			**в**			**е**			**е**

И т.д.

В неизвестных позициях можно рассчитать по формуле размещение с повторением:

$\bar{A}_m^n = m^n$, где *m* – это все варианты выборки словаря символов (*m* = 83), *n* – длина одного потока, символы которого злоумышленнику могут быть либо известны, либо нет (*n* = 7) [2]. Берём именно эту формулу, потому что нам в неизвестные места предполагаемого исходного блока сообщения нужно расположить на каждую позицию 1 из 83 элементов нашего словаря и так нужно сделать в каждой позиции, причём символы могут повторяться. В случае, если в потоке нам известны все символы, то число комбинаций размещения символов этого потока в исходном блоке сообщения определяется как 2, 3 и более (как было указано выше в пояснении к схеме 1), в противном случае по формуле размещения с повторениями. Общее же число комбинаций будет рассчитываться по правилу произведения из комбинаторики: $2 \cdot 83^7 \approx 5 \cdot 10^{13}$. Если предположить, что потоков три, тогда получим $3 \cdot 83^{7 \cdot 2} \approx 2 \cdot 10^{27}$ и так с каждым новым неизвестным потоком. Прирост новых комбинаций будет увеличиваться с каждым новым неизвестным потоком в среднем на 10^{13} и это ещё без учёта того, что потоки могут быть разной длины (но, как правило, потоки имеют равную длину, расхождения могут быть незначи-

тельны), что ещё более усложнит попытку реализации криптоатаки. Данное направление поиска уязвимости можно считать безперспективным на данном этапе исследования. Но в случае если исходный блок сообщения был поделён только на 2 потока с символами не более 8, или 3-4 потока, при этом число символов также 3-4, то взломать можно за допустимое время для хорошего компьютера, число комбинаций примерно от тысячи миллиардов до ста тысяч миллиардов. Например, если нам известен 1 поток с 4 символами, то все возможные комбинации до 3 потоков (12 символов) можно рассчитать за допустимое время, примерно: $3 \cdot 83^8 \approx 7 \cdot 10^{15}$.

2-й способ. Когда злоумышленник знает способ реализации метода рассечение-разнесение и 1 поток из 10 исходного блока (при этом не длина блока, ни количество потоков ему неизвестны).

Теперь он знает, как происходило разделение и может восстановить в исходном сообщении известный поток, понять, сколько всего потоков и сделать предположение об общей длине сообщения и некоторых неизвестных потоков. Допустим, ему известен второй поток из 10 - **ннедвее.** В табл. 4 он их (символы) восстановит в исходном сообщении.

Таблица 4. Рассечение и разнесение открытого текста, известные злоумышленнику

*	7	4	8	3	9	10	1	5	2	6	7	4	8	3	9	10	1	5	2	6	7	4	8	3	9	10	1	5	2	6
	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
									н										**н**										**е**	

Продолжение табл.3

*	7	4	8	3	9	10	1	5	2	6	7	4	8	3	9	10	1	5	2	6	7	4	8	3	9	10	1	5	2	6
	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
									д										**в**										**е**	

Продолжение табл.3

*	7	4	8	3	9	10	1	5	2	6	7	4	8	3	9	10	1	5	2	6
	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
									е										-	

* - номер потока, куда попадёт символ из исходного сообщения, выбираются в последовательности {7,4,8,3,9,10,1,5,2,6}. Из этого ключа можно сделать выводы, что всего 10 потоков.

Известный поток 7 символов, но некоторые потоки могут быть 8 символов или 6.

Предположим, что все потоки по 7 символов. Каждый неизвестный поток определяется по формуле размещения с повторениями, а общее число комбинаций определяется по правилу произведения из комбинаторики, т.к. потоки должны согласовываться по смыслу, а для этого надо все возможные комбинации для первого потока сопоставить со всеми возможными комбинациями второго потока, потом третьего и т.д. Получаем: $n \cdot m^{n \cdot l} = 7 \cdot 83^{7 \cdot 9} \approx 6 \cdot 10^{121}$. ($n$ = 7 – это длина нашего известного потока, $m^n = 83^7$ – это длина нашего неизвестного потока, состоящего из 7 символов, и таких потоков l = 9) Т.е. близко к абсолютно стойкому шифру. Безусловно, осмысленных вариантов исходного блока значительно меньше, чем $6 \cdot 10^{121}$, но их поиск потребует дополнительных вычислений. И в этом может быть уязвимость.

3-й способ. Когда злоумышленник знает все 10 потоков исходного текста, которые ему нужно собрать, а соответственно и всю длину исходного блока.

Можно предположить, что он пойдёт по трём сценариям криптоатаки:

1-й. Самый сложный. Нужно собрать осмысленный текст из 77 символов, среди которых есть одинаковые, а число возможных комбинаций определяется числом размещений с повторениями. $\overline{P}(N_1, N_{2,\dots,} N_k) = \frac{N!}{N_{1,!} N_{2!,\dots,} N_{k!}}$, где N – в числителе общее

число символов, а в знаменателе тоже общее число символов по типам k (например, буква «а» – это тип, «а» встречается в сообщении 5 раз, это её общее число) [2].

$$\bar{P}_{77}(1,3,2,3,1,1,8,1,1,4,2,3,2,1,8,7,1,3,3,3,1,1,1,1,13,1,1) =$$

$$= \frac{77!}{1!*3!*2!*3!*1!*1!*8!*1!*1!*4!*2!*3!*2!*1!*8!*7!*1!*3!*3!*3!*1!*1!*1!*1!*13!*1!*1!} \approx$$

$$\approx 3 \cdot 10^{83}.$$

Предварительно произведя в тексте подсчёт количества символов, получим: 1-П, 3-а, 2-б, 3-в, 1-г, 1-д, 8-е, 1-ж, 1-з, 4-и, 2-й, 3-к, 2-л, 1-м, 8-н, 7-о, 1-р, 3-с, 3-т, 3-у, 1-ц, 1-ч, 1-ы, 1-ь, 13- пробелов «_», 1-запятая «,», 1-точка «.».

2-й. Расположить потоки друг под другом, можно уверенно предположить, что первые символы 10 потоков образуют первую часть исходного сообщения, вторые - вторую часть и т.д. Тогда можно воспользоваться формулой перестановки без повторений: $P_c=c!$, где c – количество символов на 1-й позиции всех потоков, затем на 2-й и т.д. [3] и перемножим полученные варианты, т.к. все варианты текста для первого потока нужно сопоставить с каждым вариантом второго, затем третьего и т.д., чтобы выстроить осмысленное сообщение. Получим 10!·10!·…·7! (т.к. на 8-й позиции символы есть только у 7 потоков) или $(10!)^7 \cdot 7! \approx 4 \cdot 10^{49}$. Это несопоставимо меньше, чем $83^{63} \approx 6*10^{121}$ для абсолютно стойкого шифра. Это уже относительная уязвимость. Поэтому знание всех потоков может считаться уязвимостью данного метода.

1**) рчсскн_.**
2) **ннедвее**
3) **вмек_и_е**
4) **ой_еьнл_**
5) **аитуоас**
6) **но_и_тб**
7) **Пойлто_е**
8) **_ _нг,_иц**
9) **ыувок_он**
10) **бже_азну**

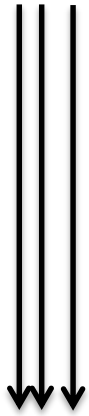
3-й. Если злоумышленник имеет представление о том, что рассечение и разнесение исходного блока сообщения на потоки происходит через определённый период, то можно предположить, что те потоки, которые он получил, воспользовавшись также формулой перестановки без повторений, т.к. у нас есть все потоки их нужно просто скомбинировать таким образом, чтобы, расположив их друг под другом, можно было бы вертикально последовательно прочесть всё исходное сообщение. Т.е. наши потоки сопоставились бы с их исходным расположением в табл. 3, тогда получим:

{7, 4, 8, 3, 9, 10, 1, 5, 2, 6}
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

Дальше можно увидеть, что они при вертикальном прочтении дают злоумышленнику исходное сообщение (см. ниже):

«**По_выбранной_мужчиной_невесте_легко_судить,_каков_он_и_знает_ли_он_себе_цену.**» (Иоганн Вольфганг Гёте) Количество всех вариантов при этом будет равно: 10! = 3 628 800 – это приемлемый результат, за приемлемое время. А если обратить внимание что три из десяти потоков имеют не 8 символов, а

лишь 7, то они однозначно должны идти в конце (на позиции 8, 9 и 10 из всех потоков). А 7 потоков должны идти в начале. Тогда получим: 7! · 3! = 30 240. Из этого можно сделать вывод, что знание всех потоков является уязвимостью данного метода.



7) **П о й л т о _ е**
4) **о й _ е ь н л _**
8) **_ _ н г , _ и ц**
3) **в м е к _ и _ е**
9) **ы у в о к _ о н**
10)**б ж е _ а з н у**
1**) р ч с с к н _ .**
5) **а и т у о а с**
2) **н н е д в е е**
6) **н о _ и _ т б**

Заключение

Из материалов работы можно сделать вывод, что данный криптографический метод защиты информации во внешних хранилищах данных очень интересен, отвечает «духу времени», но сам метод рассечение-разнесение в отдельных случаях относительно уязвим, особенно если злоумышленник владеет всеми частями исходного текста, и текст имеет относительно небольшие размеры, а потому он нуждается в дальнейшем изучении и развитии. Причём речь идёт о данной реализации метода.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Безбогов А.А., Яковлев А.В., Шамкин В.Н. Методы и средства защиты компьютерной информации: Учебное пособие. - Тамбов: Издательство ТГТУ, 2006. - 196.

2. Виленкин Н.Я. Комбинаторика. М.: Наука. Гл. ред. физ.-мат. лит., 1969.— 323 с.

3. Партыка Т. Л., Попов И. И. Информационная безопасность. Учебное пособие для студентов учреждений среднего профессионального образования. — М.: ФОРУМ: ИНФРА-М, 2002. - 368 с.: ил. - (Серия «Профессиональное образование»). ISBN 5-8199-0060-X (ФОРУМ), ISBN 5-16-001155-2 (ИНФРА-М)

4. Тайнопись. Практическое пособие по ручному шифрованию. – 4-е изд., доп. – Саратов: издательство «Новый ветер», 2010. – 206 с. ISBN 978-5-98116-120-9

5. Ravi Gupta, Rahul Singh. (Eds.). SSCC - An Improved Substitution Method for Data Encryption Using DNA Sequence and CDMB. Communications in Computer and Information Science, Pages: 197-206. | 2015. DOI http://dx.doi.org/10.1007/978-3-319-22915-7_19

6. Ekhlas Abbas Albahrani, Tayseer Karam Alshekly. (Eds.). New Chaotic Substation and Permutation Method for Image Encryption. International Journal of Applied Information Systems, Issue: 4, Volume: 12, Pages: 33-39. | Jul 6, 2017. DOI http://dx.doi.org/10.5120/ijais2017451698

7. Fei Huo, Guang Gong. (Eds.). XOR Encryption Versus Phase Encryption, an In-Depth Analysis. IEEE Transactions on Electromagnetic Compatibility, Issue: 4, Volume: 57, Pages: 903-911. | Jan 30, 2015. DOI http://dx.doi.org/10.1109/temc.2015.2390229

8. Muhammed Jassem Al-Muhammed. (Ed.). Light but Effective Encryption Technique based on Dynamic Substitution and Effective Masking. International Journal of Advanced Computer Science and Applications, Issue: 9, Volume: 9, | 2018. DOI https://dx.doi.org/10.14569/IJACSA.2018.090909

9. Sefa Tuncer. (Ed.). Information Encryption and Hiding into an Image By Steganography Methods to Improve Data Security. journal of new results in science, Issue: 12, Volume: 5, Pages: 170-177. | Nov 7, 2016. Source https://www.lens.org/083-031-697-623-665

10. Gotfried C Prasetyadi, Achmad Benny Mutiara, Rina Refianti. (Eds.). File encryption and hiding application based on advanced encryption standard (AES) and append insertion ste-ganography method. 2017 Second International Conference on Informatics and Computing (ICIC), | Nov 1, 2017. DOI

http://dx.doi.org/10.1109/iac.2017.8280584

REFERENCES

1. Bezbokov A. A., Yakovlev A.V., Simkin V. N. Methods and means of computer information protection: a Training manual. - Tambov: TSTU publishing house, 2006. - 196.

2. Vilenkin N. Ya. Combinatorika. M.: Nauka. GL. ed. Fiz. - Mat. lit., 1969. - 323 p.

3. Partyka T. L., Popov I. I. (Eds.). Information security. Training a textbook for students of institutions of secondary vocational education. - Moscow: FORUM: INFRA-M, 2002. - 368 p.: ill. - (Series "Professional education"). ISBN 5-8199-0060-X (FO-RUM), ISBN 5-16-001155-2 (INFRA-M).

4. Cryptogram. A practical guide to manual encryp-tion. - 4th ed., additional – Saratov: Novy Veter publishing house , 2010. – 206 p. ISBN 978-5-98116-120-9

5. Ravi Gupta, Rahul Singh. (Eds.). SSCC - An Improved Substitution Method for Data Encryption Using DNA Sequence and CDMB. Communications in Computer and Information Science, Pages: 197-206. | 2015. DOI http://dx.doi.org/10.1007/978-3-319-22915-7_19

6. Ekhlas Abbas Albahrani, Tayseer Karam Alshekly. (Eds.). New Chaotic Substation and Permutation Method for Image Encryption. International Journal of Applied Information Systems, Issue: 4, Volume: 12, Pages: 33-39. | Jul 6, 2017. DOI http://dx.doi.org/10.5120/ijais2017451698

7. Fei Huo, Guang Gong. (Eds.). XOR Encryption Versus Phase Encryption, an In-Depth Analysis. IEEE Transactions on Electromagnetic Compatibility, Issue: 4, Volume: 57, Pages: 903-911. | Jan 30, 2015. DOI http://dx.doi.org/10.1109/temc.2015.2390229

8. Muhammed Jassem Al-Muhammed. (Ed.). Light but Effective Encryption Technique based on Dynamic Substitution and Effective Masking. International Journal of Advanced Computer Science and Applications, Issue: 9, Volume: 9, | 2018. DOI https://dx.doi.org/10.14569/IJACSA.2018.090909

9. Sefa Tuncer. (Ed.). Information Encryption and Hiding into an Image By Steganography Methods to Improve Data Security. journal of new results in science, Issue: 12, Volume: 5, Pages: 170-177. | Nov 7, 2016. Source https://www.lens.org/083-031-697-623-665

10. Gotfried C Prasetyadi, Achmad Benny Mutiara, Rina Refianti. (Eds.). File encryption and hiding application based on advanced encryption standard (AES) and append insertion ste-ganography method. 2017 Second International Conference on Informatics and Computing (ICIC), | Nov 1, 2017. DOI http://dx.doi.org/10.1109/iac.2017.8280584

Информация об авторах

Шурховецкий Георгий Николаевич – аспирант кафедры «Информационные системы и защита информации», Иркутский государственный университет путей сообщения, г. Иркутск, e-mail: gshn5@yandex.ru

Authors

Shurkhovetsky George Nikolaevich - Post-graduate student of the Department of Information Systems and Information Security, Irkutsk State University of Railway Transport, Irkutsk, e-mail: gshn5@yandex.ru

Для цитирования

Шурховецкий Г.Н. Защита информации во внешних хранилищах данных методом рассечения-разнесения [Электронный ресурс] / Г.Н. Шурховецкий // Молодая наука Сибири: электрон. науч. журн. – 2020. – №3(9). – Режим доступа: http://mnv.irgups.ru/toma/39-2020, свободный. – Загл. с экрана. – Яз. рус., англ. (дата обращения: 03.10.2020)

For citation

Shurkhovetsky G.N. Protection of information in external data stores using the dissection-separation method. *The electronic scientific journal "Young science of Siberia"*, 2020, no. 3(9). [Accessed 03/10/20] (in Russian)