МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Методические указания

по выполнению практических работ
по дисциплине
«Криптоанализ с помощью программных средств»
для студентов направления подготовки
10.04.01 Информационная безопасность

(ЭЛЕКТРОННЫЙ ДОКУМЕНТ)

1. ИССЛЕДОВАНИЕ ПРОЦЕССА ЗАШИФРОВАНИЯ С ПОМОЩЬЮ ПРОСТОЙ ЗАМЕНЫ И РЕШЕТКИ КАРДАНО

Цели работы:

1) углубить знания, полученные студентами на лекциях по основам одноалфавитного шифрования;
2) исследовать основные характеристики алгоритма шифрования.

Формируемые компетенции:

– Способен проводить инструментальный анализ защищенности компьютерной системы и осуществлять поиск уязвимостей программного обеспечения (ПК-3);

– Способен выполнять автоматизированную информационно-аналитическую поддержку процессов принятия решений в профессиональной деятельности (ПК-5));

Теоретическая часть

Одними из самых простых и известных шифров являются шифры простой однозначной замены. Данные шифры используют в качестве шифрующего алгоритма замены или подстановки, при которых осуществляется замена символов (слов) открытого текста соответствующими символами, принадлежащими алфавиту шифротекста.

Рассмотрим модель шифра подстановки (замены). Определим модель $\varepsilon_A = (X, K, Y, E, D)$ произвольного шифра замены. Будем считать, что открытые и шифрованные тексты являются словами в алфавитах A и B соответственно: $X \subset A^*$, $Y \subset B^*$, $|A| = n$, $|B| = m$. Здесь и далее C^* обозначает множество слов конечной длины в алфавите C.

Перед зашифрованием открытый текст предварительно представляется в виде последовательности подслов, называемых шифрве-личинами. При зашифровании шифрвеличины заменяются некоторыми их эквивалентами, которые назовем шифробозначениями. Как шифрвеличины, так и шифробозначения представляют собой слова из A^* и B^* соответственно.

Пусть $U = \{u_1,\ldots,u_N\}$ – множество возможных шифрвеличин, $V = \{v_1,\ldots,v_M\}$ – множество возможных шифробозначений. Эти множества должны быть такими, чтобы любые тексты $x \in X$, $y \in Y$ можно было представить словами из U^*, V^* соответственно. Требование однозначности расшифрования влечет неравенства $N \geq n$, $M \geq m$, $M \geq N$.

Для определения правила зашифрования $E_k(x)$ в общем случае нам понадобится ряд обозначений и понятие распределителя, который, по сути, и будет выбирать в каждом такте шифрования замену соответствующей шифрвеличине.

Поскольку $M \geq N$, множество V можно представить в виде объединения непересекающихся непустых подмножеств V^i. Семейство биекций, соответствующее этому семейству разбиений множества V, обозначим через $\{\varphi_\alpha\}$: $V \Rightarrow \{V_\alpha^{(0)},\ldots,V_\alpha^{(N)}\}$ и будем предполагать, что $\varphi_\alpha(u_i) = V_\alpha^{(i)}$, $i = 1,\ldots,N$.

Рассмотрим также произвольное отображение Φ: $K \times N \Rightarrow N^*$, где $N_{l(k)} = \{1,2,\ldots,r\}$, такое, что для любых $k \in K$, $l \in N$ выполнено $\Phi(k,l) = a_1^{(k)}\ldots a_l^{(k)}$, $a_j^{(k)} \in N$, $j = 1,\ldots,r$.

Назовем последовательность $\Phi(k,l)$ распределителем, отвечающим данным значениям $k \in K$, $l \in N$.

Теперь мы сможем определить правило зашифрования произвольного шифра замены. Пусть $x \in X$, $x = x_1\ldots x_l$, $x_i \in U$, $i = 1,\ldots,l$, $k \in K$ и $\Phi(k,l) = a_1^{(k)}\ldots a_l^{(k)}$. Тогда $E_k(x) = y$, где $y = y_1\ldots y_l$, $y \in \varphi_{aj}^{(k)}(x_j)$, $j = 1,\ldots,l$.

В качестве y_j можно выбрать любой элемент множества $\varphi_{aj}^{(k)}(x_j)$. Всякий раз при шифровании этот выбор можно производить случайно. Подчеркнем, что такая многозначность при зашифровании не препятствует расшифрованию, т. к. $V_\alpha^{(i)} \cap V_\alpha^{(j)} = \varnothing$ при $i \neq j$.

Рассмотрим пример реализации шифра одноалфавитной замены. Вскрытие одноалфавитного шифра осуществляется на учете частоты появления отдельных букв или сочетаний (биграмм, триграмм, и т. д.) в языке. Примером одноалфавитного шифра замены является шифр Цезаря. Шифрование осуществляется по таблице, представляющей собой матрицу, содержащую 2 строки и n столбцов, где n – число символов алфавита (для русского алфавита – 32). Первая строка содержит все символы алфавита. Вторая строка получается из предыдущей путем циклического сдвига вправо или влево на несколько символов (букв алфавита).

Выбирается циклический сдвиг шифра. После чего процесс зашифровывания осуществляется следующим образом:

1) строится матрица для осуществления зашифрования с установленным циклическим сдвигом;

2) каждая буква шифротекста находится на пересечении столбца таблицы, определяемого буквой открытого текста, и строки, определяемой буквой ключа.

Пример. Пусть надо зашифровать текст *А нам все равно*.

В качестве ключа используем циклический сдвиг влево на 11 букв русского алфавита (буква *А* заменяется на букву *К*).

Тогда процесс зашифрования можно представить в следующем виде. Таблица 1 для циклического сдвига на 11 символов русского алфавита представлена ниже.

Таблица 1.1

Шифр одноалфавитной замены

Исходный алфавит																
А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П	Р
Циклический сдвиг на 11 символов																
К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ
Исходный алфавит																
С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	_	
Циклический сдвиг на 11 символов																
Ы	Ь	Э	Ю	Я	_	А	Б	В	Г	Д	Е	Ж	З	И	Й	

В результате процедуры шифрования получаем текст, представленный в таблице 1.2.

Таблица 1.2

Пример шифрования текста с помощью шифра одноалфавитной замены

Исходный текст														
А	_	Н	А	М	_	В	С	Е	_	Р	А	В	Н	О
Зашифрованный текст														
К	Й	Ч	К	Ц	Й	М	Ы	П	Й	Ъ	К	М	Ч	Щ

Расшифровка осуществляется следующим образом. Во второй строке происходит поиск соответствующей буквы шифротекста. Находящаяся над ней в первой строке буква и будет соответствовать букве исходного текста.

Шифры перестановки, или транспозиции, изменяют только порядок следования символа или других элементов исходного текста. Пример – решетка Кардано, которая при наложении на лист бумаги оставляет открытыми лишь некоторые его части. При зашифровке буквы сообщения вписываются в эти отверстия. При расшифровке сообщение вписывается в диаграмму нужных размеров, затем накладывается решетка, после чего на виду оказываются только буквы открытого текста.

Решетки можно использовать двумя способами.

В первом случае зашифрованный текст состоит только из букв исходного сообщения. Решетка изготавливается таким образом, чтобы при ее последовательном использовании в различных положениях каждая клетка лежащего под ней листа бумаги оказалась занятой (это поворотная решетка).

1	2	3	4	5	1
5	1	2	3	1	2
4	3	1	1	2	3
3	2	1	1	3	4
2	1	3	2	1	5
1	5	4	3	2	1

Рис. 1.1. Пример решетки Кардано

Если решетку поворачивать на 90° последовательно, то при возврате к исходному состоянию все клетки будут заполнены. Причем в каждом окаймлении должна быть вырезана только одна цифра (внешнее обрамление имеет 5 дырок – отмечены серым цветом, во внутреннем – 1, в среднем – 3). Пример простейшей решетки Кардано приведен на рис. 1.1.

Второй – стеганографический – метод использования решетки позволяет скрыть факт секретного сообщения. В этом случае заполняется только часть листа, а в остальное место дописываются буквы или слова ложного текста.

Оборудование и материалы

Аппаратура. Для выполнения лабораторной работы необходим персональный компьютер.

Программное обеспечение. Для выполнения лабораторной работы необходима операционная система с поддержкой графическо-

го окружения, установленный офисный пакет приложений, векторный графический редактор, редактор диаграмм и блок-схем.

Задания

1. Изучить теоретический материал работы.
2. Провести исследование системы шифрования на основе алгоритма одноалфавитной замены.
3. Провести исследование системы шифрования на основе алгоритма Кардано.

Ход работы

Студенты делятся на две подгруппы. В первой подгруппе студенты выбирают ключевые слова, а также получают текст, выданный преподавателем. Затем они строят таблицу и осуществляют шифрование текста.

Студенты второй подгруппы, получив от студентов первой подгруппы зашифрованное сообщение и необходимый сдвиг, строят таблицу и осуществляют процесс дешифрования.

По окончанию расшифрования студенты второй подгруппы приступают к процедуре зашифрования с использованием нового ключевого слова. Студенты первой подгруппы, получив ключевое слово и зашифрованный текст, приступают к его расшифрованию.

Содержание отчета

Отчет по лабораторной работе оформляется в виде документа Word и должен включать:

1) название лабораторной работы;
2) цель лабораторной работы;
3) формулировку индивидуального задания и результат его выполнения;
4) краткие выводы по результатам выполнения лабораторной работы.

Контрольные вопросы для защиты работы

1. Дайте определение шифра. Какие виды шифров вы знаете?
2. Дайте определение шифра одноалфавитной замены.
3. Назовите основные достоинства и недостатки шифра одноалфавитной замены.

4. Дайте определение шифра простой подстановки замены.
5. Назовите основные достоинства и недостатки шифра Кардано.

Рекомендуемая литература

1. Рябко Б. Я., Фионов А. Н. Криптографические методы защиты информации. М.: Горячая линия-Телеком, 2012. 229 с.
2. Музыкантский А. И., Фурин В. В. Лекции по криптографии. М.: МЦНМО, 2011. 68 с.
3. Глухов М. М. Введение в теоретико-числовые методы криптографии / М. М. Глухов, И. А. Круглов, А. Б. Пичкур, А. В. Черемушкин. СПб.: Лань, 2011. 400 стр.
4. Торстейнсон П., Ганеш Г. А. Криптография и безопасность в технологии .NET. М.: Бином; Лаборатория знаний, 2013. 480 с.

2. ИССЛЕДОВАНИЕ ПРОЦЕССА ШИФРОВАНИЯ СООБЩЕНИЯ С ПОМОЩЬЮ ТАБЛИЦЫ ВИЖЕНЕРА

Цели работы:

1) углубить знания, полученные на лекциях по основам многоалфавитного шифрования;

2) исследовать основные характеристики алгоритма шифрования на основе таблицы Виженера.

Формируемые компетенции

– Способен проводить инструментальный анализ защищенности компьютерной системы и осуществлять поиск уязвимостей программного обеспечения (ПК-3);

– Способен выполнять автоматизированную информационно-аналитическую поддержку процессов принятия решений в профессиональной деятельности (ПК-5));

Теоретическая часть

Наиболее простыми являются шифры замены или подстановки, особенностью которых является замена символов (слов) открытого

текста соответствующими символами, принадлежащими алфавиту шифротекста. Различают: одноалфавитную, многоалфавитную замену.

Вскрытие одноалфавитного шифра осуществляется на учете частоты появления отдельных букв или сочетаний (биграмм, триграмм и т. д.) в языке. Примером многоалфавитного шифра замены является система Виженера. Шифрование осуществляется по таблице, представляющей собой квадратную матрицу размерности $n \times n$, где n – число символов алфавита (для русского алфавита – 32).

1. Первая строка содержит все символы алфавита.

2. Каждая последующая строка получается из предыдущей путем циклического сдвига вправо на один символ (или влево).

3. Выбирается ключ или ключевая фраза.

После чего процесс зашифровывания осуществляется следующим образом:

1) под каждой буквой исходного сообщения последовательно записываются буквы ключа (если ключ короче, его используют несколько раз);

2) каждая буква шифротекста находится на пересечении столбца таблицы, определяемого буквой открытого текста, и строки, определяемой буквой ключа.

Пример. Пусть надо зашифровать текст *А нам все равно*.

В качестве ключа используем слово *КОЛОКОЛА*.

Тогда процесс зашифрования можно представить в следующем виде. Таблица Виженера для ключевого слова КОЛОКОЛА примет вид табл.2.1. Буквы, выделенные в таблице, соответствуют символам шифротекста.

В результате процедуры шифрования получаем текст, представленный в табл. 2.2.

Расшифровка осуществляется следующим образом. Под буквами шифротекста последовательно записываются буквы ключа: в строке, соответствующей очередной букве ключа, происходит поиск соответствующей буквы шифротекста. Находящаяся над ней в первой строке буква и будет соответствовать букве исходного текста.

Таблица 2.1

Таблица Виженера с ключевым словом КОЛОКОЛА

А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	_
К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	_	А	Б	В	Г	Д	Е	Ж	З	И	Й
О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	_	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	**Н**
Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	**Ш**	Щ	Ъ	Ы	Ь	Э	Ю	Я	_	А	Б	В	Г	Д	Е	Ж	З	И	Й	К
О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	_	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н
К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	**Ц**	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	_	А	Б	В	Г	Д	Е	Ж	З	И	Й
О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	_	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	**Н**
Л	М	**Н**	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	_	А	Б	В	Г	Д	Е	Ж	З	И	Й	К
А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П	Р	**С**	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	_

Таблица 2.2

Пример шифрования текста с помощью таблицы Виженера

Исходный текст														
А	_	Н	А	М	_	В	С	Е	_	Р	А	В	Н	О
Ключевое слово														
К	О	Л	О	К	О	Л	А	К	О	Л	О	К	О	Л
Зашифрованный текст														
К	Й	Ч	К	Ц	Й	М	Ы	П	Й	Ъ	К	М	Ч	Щ

Пример расшифровки принятого зашифрованного сообщения с помощью алгоритма многоалфавитной замены показан в табл. 2.3.

Таблица 2.3

Пример расшифрования текста

Зашифрованный текст														
К	Й	Ч	К	Ц	Й	М	Ы	П	Й	Ъ	К	М	Ч	Щ
Ключевое слово														
К	О	Л	О	К	О	Л	А	К	О	Л	О	К	О	Л
Расшифрованный текст														
А	_	Н	А	М	_	В	С	Е	_	Р	А	В	Н	О

Оборудование и материалы

Аппаратура. Для выполнения лабораторной работы необходим персональный компьютер.

Программное обеспечение. Для выполнения лабораторной работы необходима операционная система с поддержкой графического окружения, установленный офисный пакет приложений, векторный графический редактор, редактор диаграмм и блок-схем.

Задания

1. Изучить теоретический материал работы.
2. Провести исследование зашифрования с помощью системы многоалфавитной замены.
3. Провести исследование расшифрования с помощью системы многоалфавитной замены.

Ход работы

Студенты делятся на две подгруппы. В первой подгруппе студенты выбирают ключевые слова, а также получают текст, выдан-

ный преподавателем. Затем они строят таблицу и осуществляют шифрование текста.

Студенты второй подгруппы, получив от студентов первой подгруппы зашифрованное сообщение и необходимый сдвиг, строят таблицу и осуществляют процесс дешифрования.

По окончании расшифрования студенты второй подгруппы приступают к процедуре зашифрования с использованием нового ключевого слова. Студенты первой подгруппы, получив ключевое слово и зашифрованный текст, приступают к его расшифрованию.

Содержание отчета

Отчет по лабораторной работе оформляется в виде документа Word и должен включать:

1) название лабораторной работы;
2) цель лабораторной работы;
3) формулировку индивидуального задания и результат его выполнения;
4) краткие выводы по результатам выполнения лабораторной работы.

Вопросы для защиты работы

1. Дайте определение шифра многоалфавитной замены.
2. Приведите классификацию шифров замены.
3. Дайте определение шифра многоалфавитной замены.
4. Сравните шифры одноалфавитной замены и многоалфавитной замены.
5. Поясните алгоритм зашифрования с помощью таблицы Виженера.
6. Поясните алгоритм расашифрования с помощью таблицы Виженера
7. Назовите основные достоинства и недостатки шифра многоалфавитной замены.

Рекомендуемая литература

1. Рябко Б. Я., Фионов А. Н. Криптографические методы защиты информации. М.: Горячая линия-Телеком, 2012. 229 с.
2. Музыкантский А. И., Фурин В. В. Лекции по криптографии. М.: МЦНМО, 2011. 68 с.

3. Глухов М. М. Введение в теоретико-числовые методы криптографии / М. М. Глухов, И. А. Круглов, А. Б. Пичкур, А. В. Черемушкин. СПб.: Лань, 2011. 400 стр.

4. Алферов А. П., Зубов А. Ю. и др. Основы криптографии: учебное пособие, 2-е изд. М.: Гелиос АРВ, 2006. 480 с.

5. Баричев С. Г. Основы современной криптографии: учебный курс. М.: Телеком, 2007. 129 с.

3. ИССЛЕДОВАНИЕ ПРОЦЕССА ШИФРОВАНИЯ СООБЩЕНИЙ С ПОМОЩЬЮ УПРОЩЕННОГО S-DES

Цели работы:

1) углубить знания, полученные на лекциях по основам алгоритма блочного шифрования;

2) исследовать вопросы шифрования данных с использованием алгоритма S-DES.

Формируемые компетенции

– Способен проводить инструментальный анализ защищенности компьютерной системы и осуществлять поиск уязвимостей программного обеспечения (ПК-3);

– Способен выполнять автоматизированную информационно-аналитическую поддержку процессов принятия решений в профессиональной деятельности (ПК-5));

Теоретическая часть

Классические криптографические методы делятся на два основных типа: симметричные (шифрование секретным ключом) и асимметричные (шифрование открытым ключом).

В симметричных методах для шифрования и расшифровывания используется один и тот же секретный ключ. Наиболее известным стандартом на симметричное шифрование с закрытым ключом является стандарт для обработки информации в государственных учре-

ждениях США DES (Data Encryption Standard). Общая технология использования симметричного метода шифрования представлена на рис. 3.1.

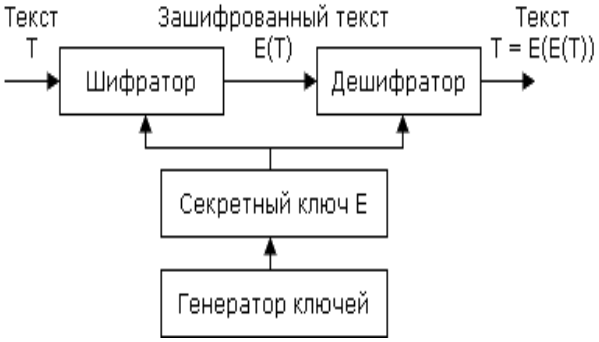

Рис. 3.1. Схема блочного шифра

Основной недостаток этого метода заключается в том, что ключ должен быть известен и отправителю, и получателю. Это существенно усложняет процедуру назначения и распределения ключей между пользователями, что может привести к компрометации ключа при его передаче.

Рассмотрим реализацию блочного шифрования с использованием упрощенного DES. На рисунке 3.2 показана общая структура упрощенного алгоритма DES, который мы в дальнейшем будем для краткости называть S-DES. закрытого текста.

Этот алгоритм получает на входе 8-битовый блок открытого текста (например, 10111101) и 10-битовый ключ, а на выходе генерируется 8-битовый блок закрытого текста.

Функция f_k, использует в качестве исходных данных не только шифруемый текст, но и 8-битовый ключ. Алгоритм можно построить так, чтобы он работал с 16-битовым ключом, состоящим из двух 8-битовых подключей, применяемых по отдельности каждый для своего вызова функции f_k .

Можно использовать и 8-битовый ключ, для чего просто следует ввести его дважды. Однако данный алгоритм не обеспечивает высокой криптостойкости.

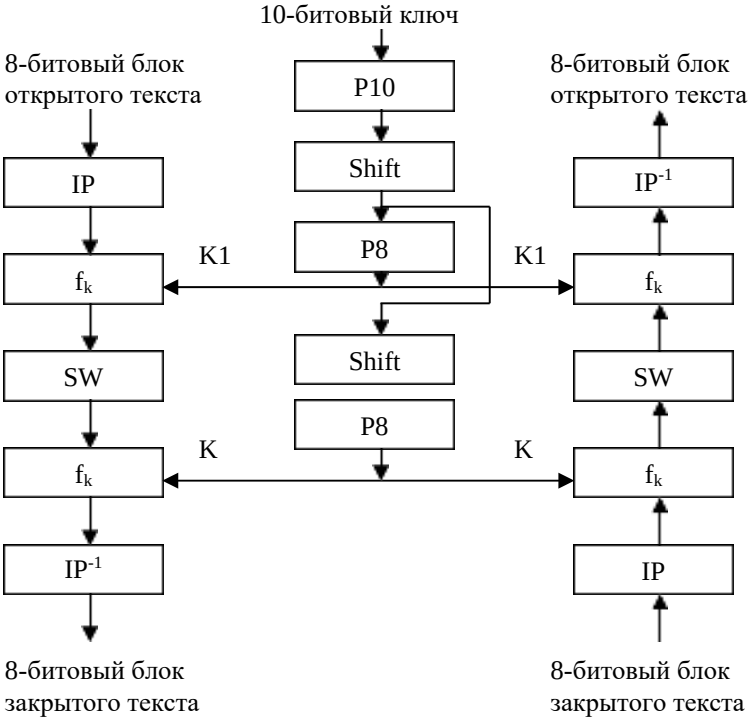

Рис. 3.2. Схема упрощенного алгоритма DES

Наконец, можно прибегнуть к комбинированному решению, когда требуется 10-битовый ключ, из которого генерируются два 8-битовых, как показано на рис. 3.1. В этом случае ключ сначала преобразуется путем перестановки (P10). После этого применяется операция сдвига, а полученные в ее результате данные поступают на вход перестановки (P8), которая генерирует первый 8-битовый ключ (K1). Те же полученные в результате операции сдвига данные поступают на вход другой операции сдвига и другой функции перестановки (P8), в результате чего генерируется второй подключ (K2). Рассмотрим алгоритм вычисления раундовых ключей в алго- ритме S-DES, когда используется 10-битовый ключ, который нахо- дится у отправителя и у получателя сообщения. Из этого ключа на определенных этапах шифрования и дешифрования генерируется два 8-битовых ключа. На рис. 3.3 показана схема процедуры создания подключей. Сначала выполняется перестановка битов ключа

следующим образом. Если 10-битовый ключ представить в виде $k_1 k_2 k_3 k_4 k_5 k_6 k_7 k_8 k_9 k_{10}$, то перестановку P10 можно задать формулой

$$P10\ (k_1\ k_2\ k_3\ k_4\ k_5\ k_6\ k_7\ k_8\ k_9\ k_{10}) = (k3\ k_5\ k_2\ k_7\ k_4\ k_{10}\ k_1\ k_9\ k_8\ k_6)$$

Можно также представить перестановку P10 в табличной форме:

P 10									
3	5	2	7	4	10	1	9	8	6

Эту таблицу следует читать слева направо. Каждый ее элемент идентифицирует позицию бита исходных данных в генерируемой выходной последовательности. Иными словами, первым битом в выходной последовательности будет третий бит исходной последовательности, вторым – пятый и т. д. Например, в соответствии с данной таблицей ключ *(1010000010)* будет преобразован к виду *(1000001100)*. После этого отдельно для первых пяти битов и отдельно для вторых выполняется циклический сдвиг влево (LS-1), который еще называют вращением. В нашем случае в результате будет получена последовательность *(00001 11000)*.

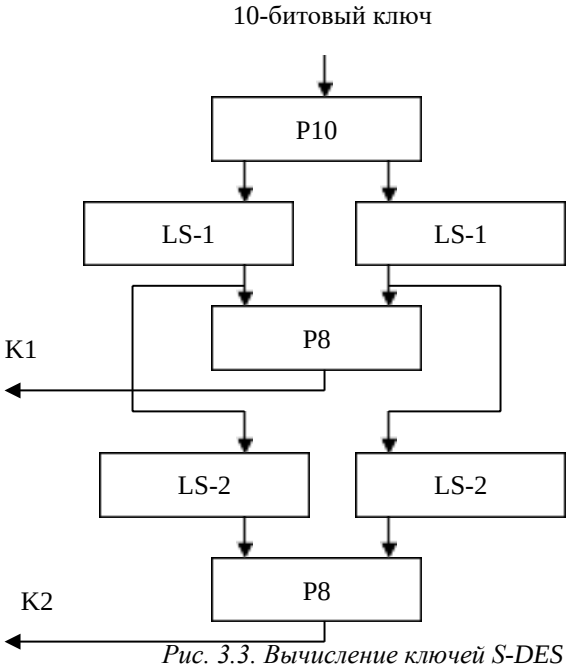

Рис. 3.3. Вычисление ключей S-DES

Затем применяется перестановка P8, в результате которой из 10-битового ключа сначала выбираются, а затем переставляются 8 битов по правилу:

P8							
6	3	7	4	8	5	10	9

В результате этой операции получается первый подключ (K1). В нашем примере он будет иметь вид *(10100100).*

Теперь нужно вернуться к двум 5-битовым строкам, полученным в результате применения функций LS-1**,** и выполнить с каждой из этих строк циклический сдвиг влево на две позиции (LS-2). В нашем конкретном случае значение *(00001 11000)* будет преобразовано к виду *(00100 00011).* Наконец, применив к полученной в результате последовательности перестановку P8, получим подключ (K2). Для нашего примера результатом будет *(01000011).*

Полученные раундовые ключи могут быть использованы при выполнении операции шифрования S-DES. Тогда данный алгоритм можно представить в виде композиции функций:

$$IP^{-1} \circ f_{k2} \circ SW \circ f_{k1} \circ IP, \quad (1)$$

или, иначе:

$$\textit{шифрованный текст} = IP^{-1}(f_{k2}(SW(f_{k1}(IP(\textit{открытый текст}))))), \quad (2)$$

где $K_1 = P8(\textit{сдвиг}(P10(\textit{ключ}))),$

$K_2 = P8(\textit{сдвиг}(\textit{сдвиг}(P10(\textit{ключ})))).$

Процесс дешифрования, также представленный на рисунке 3.1, по сути, является процессом, обратным процессу шифрования:

$$\textit{открытый текст} = IP^{-1}(f_{k1}(SW(f_{k2}(IP(\textit{шифрованный текст}))))). \quad (3)$$

На рис. 3.4 представлена более подробная схема алгоритма шифрования S-DES. Как уже упоминалось, процесс шифрования представляет собой последовательное выполнение пяти операций, которые мы рассмотрим здесь каждую в отдельности.

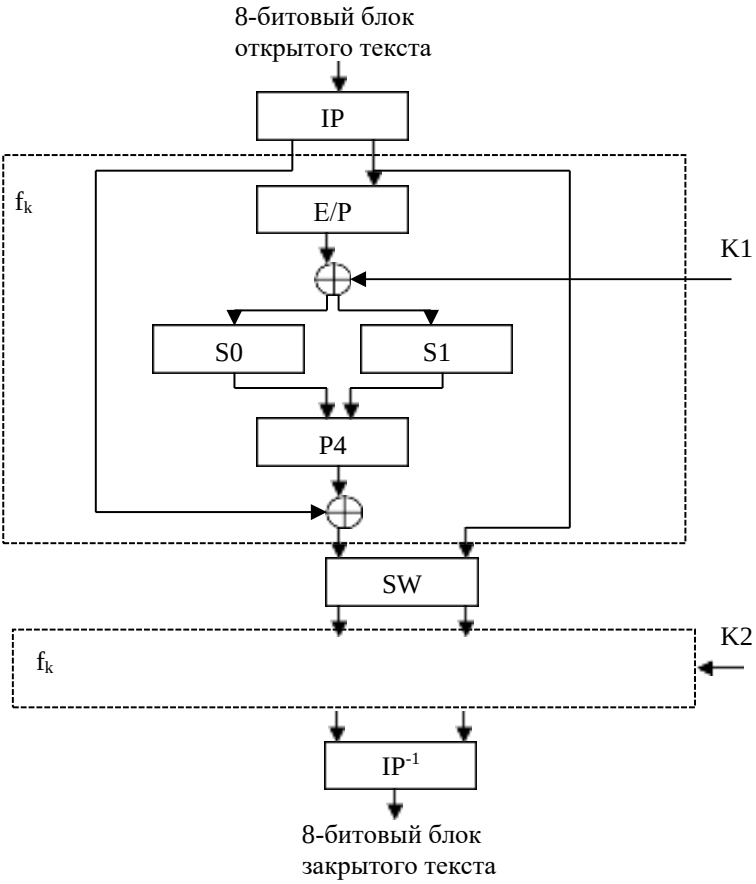

Рис. 3.4. Подробная схема шифрования S-DES

Начальная и завершающая перестановки. Первой операцией данного блочного алгоритма является начальная перестановка. В этом случае на вход алгоритма S-DES поступает 8-битовый блок открытого текста, к которому применяется начальная перестановка, заданная функцией IP.

IP							
2	6	3	1	4	8	5	7

Все 8 битов открытого текста сохраняют свои значения, но меняется порядок их следования. На завершающей стадии алгоритма выполняется обратная перестановка.

IP^{-1}							
4	1	3	5	7	2	8	6

Как легко убедиться с помощью простой проверки, вторая из приведенных выше перестановок действительно является обратной по отношению к первой, т. е. $IP^{-1}(IP(X)) = X$.

Функция f_k. Самым сложным компонентом S-DES является функция f_k, представляющая собой комбинацию перестановки и подстановки. Пусть *L* и *R* означают соответственно первые 4 бита и последние 4 бита 8-битовой последовательности, подаваемой на вход 4, и пусть *F* – некоторое отображение пространства 4-битовых строк в себя, не обязательно являющееся взаимно однозначным. Тогда

$$f_k(L,B)=(L+F(E, SK), R),$$

где *SK* обозначает подключ, а + – операцию XOR (побитовое исключающее ИЛИ). Например, если в результате применения функции IP (рис. 3.3) получено значение *(1011 1101)* и *F(1101, SK) = (1110)* для некоторого ключа *SK*, то $f_k(10111101) = (01011101)$, так как $(1011)+ (1110) = (0101)$.

Теперь опишем отображение *F*. На входе этого отображения имеем 4-битовое значение $(n_1\ n_2\ n_3\ n_4)$. Первой операцией является операция расширения / перестановки.

E/P							
4	1	2	3	2	3	4	1

Для дальнейшего рассмотрения удобнее представить результат в следующей форме:

$$n_4 \quad n_1 \qquad n_2 \quad n_3$$

$$n_2 \quad n_3 \qquad n_4 \quad n_1$$

К этому значению с помощью операции XOR добавляется 8-битовый подключ $K_1 = (k_{11},\ k_{12}\ k_{13}\ k_{14}\ k_{15}\ k_{16}\ k_{17}\ k_{18})$:

$$n_4 + k_{11} \quad n_1 + k_{12} \qquad n_2 + k_{13} \quad n_3 + k_{14}$$

$$n_2 + k_{15} \quad n_3 + k_{16} \qquad n_4 + k_{17} \quad n_1 + k_{18}$$

Переименуем 8 битов, полученных в результате, как показано ниже:

$$p_{00} \quad p_{01} \qquad p_{02} \quad p_{03}$$

$$p_{10} \quad p_{11} \qquad p_{12} \quad p_{13}$$

Первые четыре бита (первая строка приведенной выше матрицы) поступают на вход модуля *S0*, на выходе которого получается 2-битовая последовательность, а оставшиеся четыре бита (вторая строка матрицы) – на вход модуля *S1*, на выходе которого получается другая 2-битовая последовательность. Эти S-модули (матрицы кодирования) работают следующим образом. Первый и четвертый биты входной последовательности рассматриваются как 2-битовые числа, определяющие строку, а второй и третий – как числа, определяющие столбец *S*-матрицы. Модули *S0* и *S1* можно определить следующим образом:

$$S0=\begin{bmatrix} 1 & 0 & 3 & 2 \\ 3 & 2 & 1 & 0 \\ 0 & 2 & 1 & 3 \\ 3 & 1 & 3 & 1 \end{bmatrix} \qquad S1=\begin{bmatrix} 1 & 1 & 2 & 3 \\ 2 & 0 & 1 & 3 \\ 3 & 0 & 1 & 0 \\ 2 & 1 & 0 & 3 \end{bmatrix}$$

Элементы, находящиеся на пересечении соответствующей строки и столбца, задают 2-битовые выходные значения. Например, если $(P_{0,0}\ P_{0,3}) = (00)$ и $(p_{0,1}\ p_{02}) = (10)$, то выходные 2 бита задаются значением, которое находится на пересечении строки 0 и столбца 2 матрицы *S0* (оно равно *3* или *(11)* в двоичном представлении). Точно так же $(P_{1,0}\ P_{1,3})$ и $(p_{1,1}\ p_{12})$ служат для определения строки и столбца матрицы *S1*, на пересечении которых стоит значение, задающее вторые 2 бита.

Теперь 4 бита, полученные на выходе модулей *S0* и *S1*, преобразуются с помощью перестановки следующим образом

P4			
2	4	3	1

Результат применения перестановки P4 и является результатом функции F.

Оборудование и материалы

Аппаратура. Для выполнения лабораторной работы необходим персональный компьютер.

Программное обеспечение. Для выполнения лабораторной работы необходима операционная система с поддержкой графического окружения, установленный офисный пакет приложений, векторный графический редактор, редактор диаграмм и блок-схем.

Задания

1. Изучить теоретический материал работы.
2. Провести исследование процесса вычисления раундовых ключей алгоритма S-DES
3. Провести исследование процесса шифрования с помощью блочного алгоритма шифрования S-DES.

Ход работы

Каждый студент получает индивидуальное задание по выработке ключа для алгоритма упрощенного S-DES. Задание представлено в таблице 3.1.

Таблица 3.1

Задание на разработку ключей для S-DES

№ п/п	*Исходные данные*	*№ п/п*	*Исходные данные*
1.	1100110101	15.	1111110101
2.	1010100101	16.	1100100101
3.	0001010101	17.	0111010101
4.	1010111101	18.	1110111101
5.	1011101011	19.	1001101011
6.	1100111101	20.	1100100001
7.	0011010101	21.	0100010101
8.	1100001111	22.	1101101111
9.	0100110101	23.	0111110101
10.	1011010101	24.	1011100101
11.	0100001100	25.	1111101100
12.	0011001010	26.	1111001010
13.	1110101000	27.	0100101001
14.	1100011111	28.	0001100110

Каждый студент получает индивидуальное задание по осуществлению процесса шифрования. Открытый текст приведен в таблице 3.2. В качестве ключа для алгоритма S-DES использовать результаты предыдущего занятия.

Таблица 3.2

Открытый текст для зашифрования S-DES

№ п/п	*Исходные данные*	*№ п/п*	*Исходные данные*
1.	11111101	15.	00110101
2.	11000101	16.	10100101
3.	01110101	17.	00010101
4.	10111101	18.	10111101
5.	10001011	19.	10101011
6.	11100001	20.	11001101
7.	00010101	21.	00110101
8.	11011111	22.	11001111
9.	01110101	23.	00110101
10.	10110101	24.	11010101
11.	11111000	25.	00001100
12.	11110010	26.	00111010
13.	00101001	27.	11101000
14.	01100110	28.	00011111

Содержание отчета

Отчет по лабораторной работе оформляется в виде документа Word и должен включать:

1) название лабораторной работы;
2) цель лабораторной работы;
3) формулировку индивидуального задания и результат его выполнения.
4) краткие выводы по результатам выполнения лабораторной работы.

Вопросы для защиты работы

1. Дайте определение блочных шифров.
2. Приведите классификацию систем шифрования с секретным ключом.
3. Поясните, чем системы блочного шифрования отличаются от систем поточного шифрования.
4. Поясните работу петли Фейстеля.
5. Поясните особенности шифрования с использованием S-DES.
6. Какие виды шифров вы можете отнести к этой группе?
7. Какие виды ключей используются в S-DES?

8. Поясните алгоритм вычисления ключа в алгоритме блочного шифрования упрощенного S-DES.

9. Сравните систему получения ключей в S-DES и DES.

10. Поясните работу сети Фейстеля.

11. Поясните принцип суперпозиций, который используется при построении блочных шифров.

12. Приведите примеры применения двух основных криптографических примитива.

13. Поясните алгоритм шифрования упрощенного DES

14. В чем проявляются особенности процесса шифрования в S-DES?

15. Сколько раундов используется в S-DES?

16. Поясните работу S-блоков.

17. Основные операции шифрования с помощью алгоритма S-DES.

Рекомендуемая литература

1. Рябко Б. Я., Фионов А. Н. Криптографические методы защиты информации. М.: Горячая линия-Телеком, 2012. 229 с.

2. Музыкантский А. И., Фурин В. В. Лекции по криптографии. М.: МЦНМО, 2011. 68 с.

3. Глухов М. М. Введение в теоретико-числовые методы криптографии / Глухов М. М., И. А. Круглов, А. Б. Пичкур, А. В. Черемушкин. СПб.: Лань, 2011. 400 с.

4. Торстейнсон П., Ганеш Г. А. Криптография и безопасность в технологии .NET. М.: Бином; Лаборатория знаний, 2013. 480 с.

5. Петров А. А. Компьютерная безопасность. Криптографические методы защиты. М.: ДМК Пресс, 2008. 448 с.

6. Алферов А. П., Зубов А. Ю. и др. Основы криптографии: учебное пособие., 2-е изд. М.: Гелиос АРВ, 2006. 480 с.

4. ИССЛЕДОВАНИЕ ПРОЦЕССА РАСШИФРОВАНИЯ СООБЩЕНИЙ С ПОМОЩЬЮ УПРОЩЕННОГО S-DES

Цели работы:

1) углубить знания, полученные на лекциях по основам построения блочного алгоритма шифрования;

2) исследовать вопросы расшифрования принятого сообщения с помощью блочного алгоритма S-DES.

Формируемые компетенции

– Способен проводить инструментальный анализ защищенности компьютерной системы и осуществлять поиск уязвимостей программного обеспечения (ПК-3);

– Способен выполнять автоматизированную информационно-аналитическую поддержку процессов принятия решений в профессиональной деятельности (ПК-5));

Теоретическая часть

Основным классификационным признаком систем шифрования данных является способ их функционирования. По способу функционирования системы шифрования данных делят на два класса:

- системы «прозрачного» шифрования;
- системы, специально вызываемые для осуществления шифрования.

В системах «прозрачного» шифрования (шифрование «налету») криптографические преобразования осуществляются в режиме реального времени, незаметно для пользователя. Например, пользователь записывает подготовленный в текстовом редакторе документ на защищаемый диск, а система защиты в процессе записи выполняет его шифрование.

Системы второго класса обычно представляют собой утилиты (программы), которые необходимо специально вызывать для выполнения процедур шифрования.

Особое значение криптографические преобразования имеют при передаче данных по распределенным вычислительным сетям. Для защиты данных в распределенных сетях используются два подхода: канальное шифрование и оконечное (абонентское) шифрование.

В случае канального шифрования защищается вся информация, передаваемая по каналу связи, включая служебную. Этот способ шифрования обладает следующим достоинством – встраивание процедур шифрования на канальный уровень позволяет использовать аппаратные средства, что способствует повышению производительности системы.

Оконечное (абонентское) шифрование позволяет обеспечить конфиденциальность данных, передаваемых между двумя абонентами. В этом случае защищается только содержание сообщений, вся служебная информация остается открытой.

Алгоритм шифрования относится к блочным шифрам. На прошлой лабораторной работе было проведено исследование процедуры зашифрования блока открытого текста. На данном занятии будет проведено исследование процесса расшифрования с использованием S-DES.

Алгоритм дешифрования S-DES в качестве исходных данных использует 8-битовый блок шифрованного текста и тот же 10-битовый ключ, который применялся для шифрования, а в результате работы алгоритм дешифрования должен генерировать 8-битовый блок открытого текста.

Алгоритм дешифрования включает последовательное выполнение пяти операций; начальной перестановки IP, сложной функции, являющейся композицией операций перестановки и подстановки и зависящей от полученного ключа, перестановки SW, при которой две половинки последовательности данных просто меняются местами, еще раз функции f и, наконец, перестановки, обратной начальной (IP-1). Подробная схема дешифрования с использованием S-DES приведена на рис. 4.1.

На вход алгоритма поступает 8-битовый блок закрытого текста, к которому применяется начальная перестановка, заданная функцией IP.

IP							
2	6	3	1	4	8	5	7

Все 8 битов закрытого текста сохраняют свои значения, но меняется порядок их следования. На завершающей стадии алгоритма выполняется обратная перестановка.

IP^{-1}							
4	1	3	5	7	2	8	6

Как легко убедиться с помощью простой проверки, вторая из приведенных выше перестановок действительно является обратной по отношению к первой, т.е. $IP^{-1}(IP(X)) = X$.

Криптографически сложная функция расшифрования f_k. Самым сложным компонентом S-DES является функция f_k, представляющая собой комбинацию перестановки и подстановки. Пусть

L и *R* означают соответственно первые 4 бита и последние 4 бита 8-битовой последовательности, подаваемой на вход 4, и пусть *F* – некоторое отображение пространства 4-битовых строк в себя, не обязательно являющееся взаимно однозначным. Тогда

$$f_k(L,B)=(L+F(E, SK), R),$$

где *SK* обозначает подключ, а + – операцию XOR (побитовое исключающее ИЛИ). Например, если в результате применения функции IP (рисунок 3) получено значение *(1011 1101)* и *F(1101, SK) = (1110)* для некоторого ключа *SK*, то $f_k(10111101) = (01011101)$, так как *(1011)+ (1110) = (0101).*

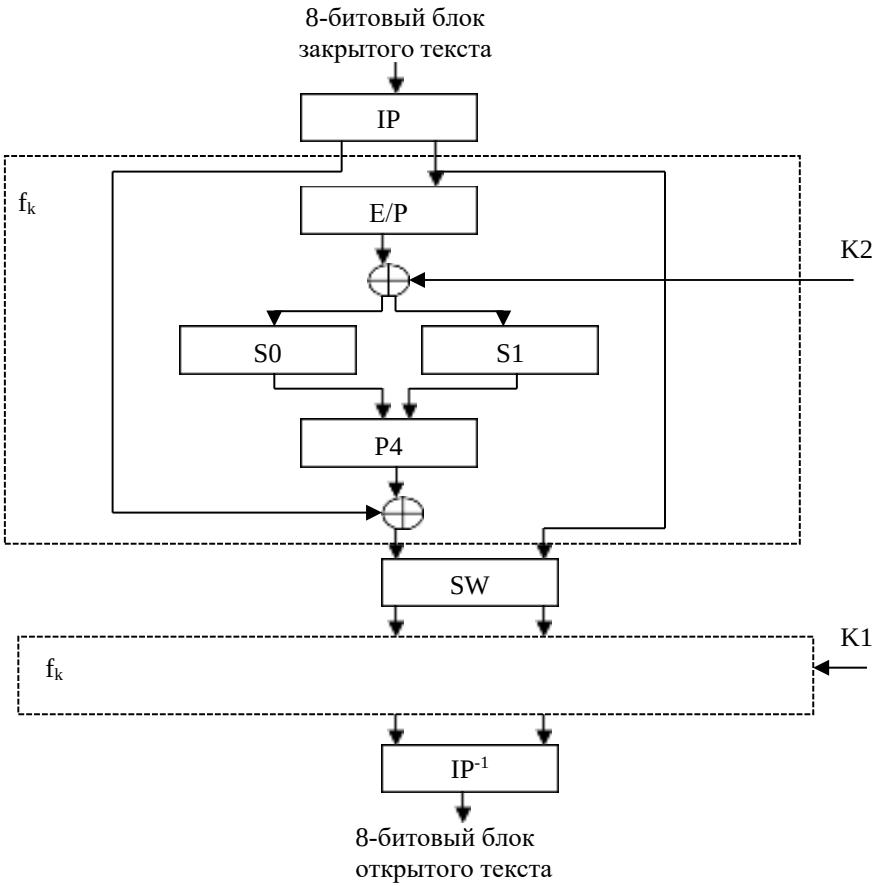

Рис. 4.1. Подробная схема расшифрования S-DES

Теперь опишем отображение *F*. На входе этого отображения имеем 4-битовое значение $(n_1\ n_2\ n_3\ n_4)$. Первой операцией является операция расширения/перестановки.

E/P							
4	1	2	3	2	3	4	1

Для дальнейшего рассмотрения удобнее представить результат в следующей форме:

$$n_4 \quad n_1 \qquad n_2 \quad n_3$$

$$n_2 \quad n_3 \qquad n_4 \quad n_1$$

К этому значению с помощью операции XOR добавляется 8-битовый подключ

$$K_1 = (k_{11}, \quad k_{12} \quad k_{13} \quad k_{14} \quad k_{15} \quad k_{16} \quad k_{17} \quad k_{18}):$$

$$n_4 + k_{11} \quad n_1 + k_{12} \qquad n_2 + k_{13} \quad n_3 + k_{14}$$

$$n_2 + k_{15} \quad n_3 + k_{16} \qquad n_4 + k_{17} \quad n_1 + k_{18}$$

Переименуем полученные в результате 8 битов, как показано ниже:

$$p_{00} \quad p_{01} \qquad p_{02} \quad p_{03}$$

$$p_{10} \quad p_{11} \qquad p_{12} \quad p_{13}$$

Первые четыре бита (первая строка приведенной выше матрицы) поступают на вход модуля *S0*, на выходе которого получается 2-битовая последовательность, а оставшиеся четыре бита (вторая строка матрицы) – на вход модуля *S1*, на выходе которого получается другая 2-битовая последовательность. Эти S-модули (матрицы кодирования) работают следующим образом. Первый и четвертый биты входной последовательности рассматриваются как 2-битовые числа, определяющие строку, а второй и третий – как числа, определяющие столбец *S*-матрицы. Модули *S0* и*S1* можно определить следующим образом:

$$S0 = \begin{bmatrix} 1 & 0 & 3 & 2 \\ 3 & 2 & 1 & 0 \\ 0 & 2 & 1 & 3 \\ 3 & 1 & 3 & 1 \end{bmatrix} \qquad S1 = \begin{bmatrix} 1 & 1 & 2 & 3 \\ 2 & 0 & 1 & 3 \\ 3 & 0 & 1 & 0 \\ 2 & 1 & 0 & 3 \end{bmatrix}$$

Элементы, находящиеся на пересечении соответствующей строки и столбца, задают 2-битовые выходные значения. Например, если $(P_{0,0}\ P_{0,3}) = (00)$ и $(p_{0,1}\ p_{02}) = (10)$, то выходные 2 бита задаются

значением, которое находится на пересечении строки 0 и столбца 2 матрицы *S0* (оно равно *3* или *(11)* в двоичном представлении). Точно так же *($P_{1,0}$ $P_{1,3}$)* и *($p_{1,1}$ p_{12})* служат для определения строки и столбца матрицы *S1*, на пересечении которых стоит значение, задающее вторые 2 бита.

Теперь 4 бита, полученные на выходе модулей *S0* и *S1*, преобразуются с помощью перестановки следующим образом:

P4			
2	4	3	1

Результат применения перестановки P4 и является результатом сложной функции F.

Оборудование и материалы

Аппаратура. Для выполнения лабораторной работы необходим персональный компьютер.

Программное обеспечение. Для выполнения лабораторной работы необходима операционная система с поддержкой графического окружения, установленный офисный пакет приложений, векторный графический редактор, редактор диаграмм и блок-схем.

Задания

1. Изучить теоретический материал работы.
2. Провести исследование процесса расшифрования с использованием алгорима S-DES.

Ход работы

Каждый студент получает индивидуальное задание по осуществлению дешифрования. В качестве зашифрованного текста, а также в качестве ключа для упрощенного S-DES использовать результаты предыдущих занятий

Содержание отчета

Отчет по лабораторной работе оформляется в виде документа Word и должен включать:

1) название лабораторной работы;
2) цель лабораторной работы;
3) формулировку индивидуального задания и результат его выполнения;

4) краткие выводы по результатам выполнения лабораторной работы.

Вопросы для защиты работы

1. Поясните особенности процесса дешифрования в S-DES.
2. Сколько раундов в S-DES для осуществления дешифрования?
3. Проведите сравнение алгоритма DES и S-DES по этапу расшифрования передаваемого сообщения.
4. Поясните, почему алгоритмы шифрования DES и S-DES имеют четное количество раундов.
5. Поясните назначение операции «рассеивания».
6. Поясните назначение операции «размешивание».
7. В чем достоинства схемы Фейстеля?
8. Чем определяется нелинейность алгоритма блочного шифрования DES и S-DES?
9. Какие требования предъявляются к S-блокам?

Рекомендуемая литература

1. Рябко Б. Я., Фионов А. Н. Криптографические методы защиты информации. М.: Горячая линия-Телеком, 2012. 229 с.
2. Музыкантский А. И., Фурин В. В. Лекции по криптографии. М.: МЦНМО, 2011. 68 с.
3. Глухов М. М. Введение в теоретико-числовые методы криптографии / М. М. Глухов, И. А. Круглов, А. Б. Пичкур, А. В. Черемушкин. СПб.: Лань, 2011. 400 стр.
4. Молдовян А. А. и др. Криптография. СПб.: Лань, 2007. 224 с.
5. Введение в криптографию / под ред. В. В.Ященко. М.: МЦНМО, 2006. 288 с.

5. ИССЛЕДОВАНИЕ ПОТОЧНОГО ШИФРОВАНИЯ СООБЩЕНИЙ В СИНХРОНИЗИРУЮЩИХСЯ СИСТЕМАХ, ПОСТРОЕННЫХ НА ОСНОВЕ МНОГОТАКТОВЫХ КОДОВЫХ ФИЛЬТРОВ

Цели работы:

1) углубить знания, полученные на лекциях по основам поточного шифрования;

2) исследовать вопросы получения синхронной псевдослучайной последовательности.

Формируемые компетенции:

– Способен проводить инструментальный анализ защищенности компьютерной системы и осуществлять поиск уязвимостей программного обеспечения (ПК-3);

– Способен выполнять автоматизированную информационно-аналитическую поддержку процессов принятия решений в профессиональной деятельности (ПК-5));

Теоретическая часть

Построить эффективный криптоалгоритм можно, лишь отказавшись от абсолютной стойкости. Возникает задача разработки такого теоретически нестойкого шифра, для вскрытия которого противнику потребовалось бы выполнить такое число операций, которое неосуществимо на современных и ожидаемых в ближайшей перспективе вычислительных средствах за разумное время. В первую очередь представляет интерес схема, использующая ключ небольшой разрядности, который в дальнейшем выполняет функцию «зародыша», порождающего значительно более длинную ключевую последовательность.

Гаммированием называют процедуру наложения на входную информационную последовательность гаммы шифра, т. е. последовательности с выходов генератора псевдослучайных кодов.

Последовательность называется псевдослучайной, если по своим статистическим свойствам она неотличима от истинно случайной последовательности, но в отличие от последней является детерминированной, т. е. знание алгоритма ее формирования дает возможность ее повторения необходимое число раз.

Если символы входной информационной последовательности и гаммы представлены в двоичном виде, наложение чаще всего реализуется с помощью операции поразрядного сложения по модулю 2. Надежность шифрования методом гаммирования определяется качеством генератора гаммы.

Простейшие устройства поточного синхронного и самосинхронизирующегося шифрования с использованием ГПК, реализованного на основе N-разрядного регистра сдвига с линейной обратной связью – LFSR (Linear Feedback Shift Register), называются *скремблерами*, а сам процесс преобразования – скремблированием.

Используя приложение выбираем производящий полином *P(x)* такой, чтобы его степень была равна трем, вес полинома – 3.

В качестве образующего полинома используем неприводимый многочлен $P(x) = x^4 + x + 1$.

Количество ячеек памяти должно равняться степени порождающего полинома (т. е. четырем), количество сумматоров по модулю 2 определяется числом ненулевых коэффициентов перед степенью *«x»* (т. е. один). Тогда схема кодирующего устройства примет следующий вид

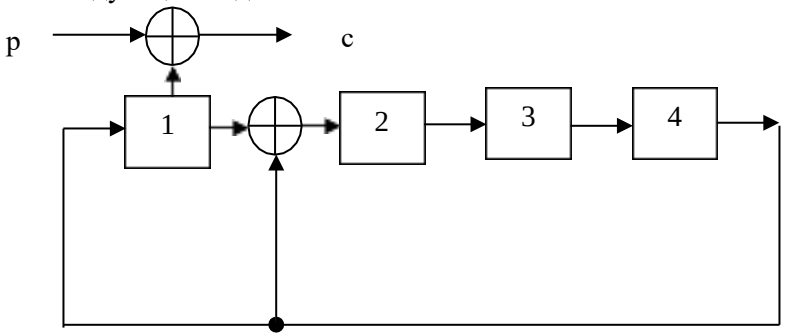

Рис. 5.1. Шифрующее устройство поточного кода

Для иллюстрации работы устройства воспользуемся соотношениями, показывающими процесс образования символов в ячейках:

Символ в 1 яч. = яч. 4^*

Символ в 2 яч. = яч. 4^* + яч.1^*

Символ в 3 яч. = яч. 2^*

Символ в 4 яч. = яч. 3^*

В таблице 5.1 приведены результаты шифрования открытого кода в течение первых 7 тактов.

Таблица 5.1

Номер такта	Открытый текст	Ячейка № 1	Ячейка № 2	Ячейка № 3	Ячейка № 3	Закрытый текст
1	1	1	1	0	0	1
2	1	1	0	1	1	1
3	0	1	0	0	1	0
4	1	1	1	0	0	0
5	0	0	0	1	0	0
6	1	0	0	0	1	1
7	1	1	0	0	0	0

В синхронных поточных шифрах гамма формируется независимо от входной последовательности, каждый элемент (бит, символ, байт и т. п.) которой таким образом шифруется независимо от других элементов. В синхронных поточных шифрах отсутствует эффект размножения ошибок, т. е. число искаженных элементов в расшифрованной последовательности равно числу искаженных элементов зашифрованной последовательности, пришедшей из канала связи.

Вставка или выпадение элемента зашифрованной последовательности недопустимы, так как из-за нарушения синхронизации это приведет к неправильному расшифрованию всех последующих элементов.

В таблице 5.2 показан пример поточного шифрования и расшифрования двоичной последовательности *11100101010110* с использованием гаммы формируемой 4-разрядным *LFSR* при начальном состоянии **1001**. Зашифрованная последовательность имеет вид ***01001010010010.***

Таблица 5.2

Пример поточного шифрования и расшифрования двоичной последовательности при отсутствии ошибок в принятой комбинации

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования при отсутствии ошибок в принятой комбинации*					
Передающая сторона							*Приемная сторона*					
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*			
0	*1*	1	1	0	0		***0***	*1*	1	1	0	0
1	*1*	0	1	1	0		***1***	*1*	0	1	1	0
0	*1*	1	0	1	1		***0***	*1*	1	0	1	1

Продолжение таблицы 5.2

0	*0*	0	1	0	1
1	*0*	1	0	1	0
0	*1*	1	1	0	1
1	*0*	1	1	1	0
0	*1*	1	1	1	1
0	*0*	0	1	1	1
1	*1*	0	0	1	1
0	*0*	0	0	0	1
0	*1*	1	0	0	0
1	*1*	0	1	0	0
0	*0*	0	0	1	0

0	*0*	0	1	0	1
1	*0*	1	0	1	0
0	*1*	1	1	0	1
1	*0*	1	1	1	0
0	*1*	1	1	1	1
0	*0*	0	1	1	1
1	*1*	0	0	1	1
0	*0*	0	0	0	1
0	*1*	1	0	0	0
1	*1*	0	1	0	0
0	*0*	0	0	1	0

При отсутствии искажений в канале связи после расшифрования с использованием той же гаммы получается исходная последовательность.

На рис. 5.2 приведена структура дешифратора поточного шифра, использующего ПСП, реализованного на основе многотактового кодового фильтра. Количество ячеек памяти должно равняться степени порождающего полинома (т. е. четырем), количество сумматоров по модулю два определяется числом ненулевых коэффициентов перед степенью *«x»* (т. е. один). Тогда схема кодирующего устройства примет такой вид (рис. 5.2.).

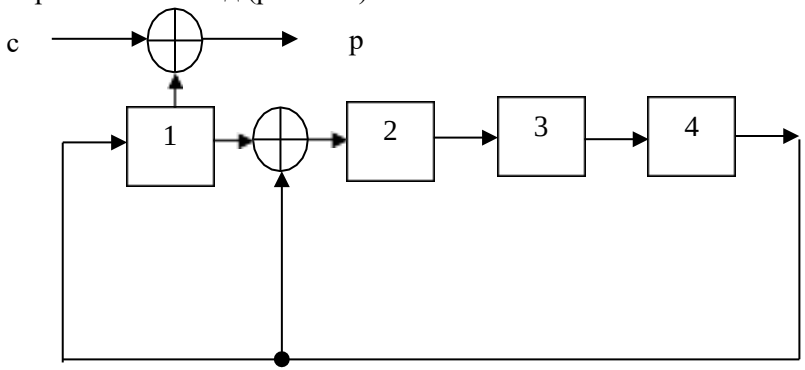

Рис. 5.2. Дешифратор поточного кода

В таблице 5.3 рассмотрена ситуация, когда при передаче зашифрованной последовательности был потерян четвертый бит, и вместо правильной последовательности к получателю пришла последовательность **0101010010010.**

Таблица 5.3

Пример поточного шифрования и расшифрования двоичной последовательности, когда при передаче был потерян четвертый бит

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования, когда при передаче был потерян четвертый бит*						
Передающая сторона							*Приемная сторона*						
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*				
0	*1*	1	1	0	0		***0***	*1*	1	1	0	0	
1	*1*	0	1	1	0		***1***	*1*	0	1	1	0	
0	*1*	1	0	1	1		***0***	*1*	1	0	1	1	
0	*0*	0	1	0	1		***1***	*1*	0	1	0	1	
1	*0*	1	0	1	0		***0***	*1*	1	0	1	0	
0	*1*	1	1	0	1		***1***	*0*	1	1	0	1	
1	*0*	1	1	1	0		***0***	*1*	1	1	1	0	
0	*1*	1	1	1	1		***0***	*1*	1	1	1	1	
0	*0*	0	1	1	1		***1***	*1*	0	1	1	1	
1	*1*	0	0	1	1		***0***	*0*	0	0	1	1	
0	*0*	0	0	0	1		***0***	*0*	0	0	0	1	
0	*1*	1	0	0	0		***1***	*0*	1	0	0	0	
1	*1*	0	1	0	0		***0***	*0*	0	1	0	0	
0	*0*	0	0	1	0								

Видно, что после расшифрования всех битов, следующих после выпавшего, происходит искажение информации. В результате вместо битовой строки **0101010110** будет получена строка **1101110000.**

В таблице 5.4 рассмотрена ситуация, когда при передаче зашифрованной последовательности произошло искажение пятого (1–0) и восьмого (0–1) битов и вместо правильной последовательности к получателю пришла последовательность **01000011010010.** Видно, что после расшифрования вместо правильной строки будет получена строка **11101100010110** с искаженным пятым (0–1) и восьмым (1–0) битами.

Таблица 5.4

Пример поточного шифрования и расшифрования двоичной последовательности, когда при передаче произошло искажение битов

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования, когда при передаче произошло искажение битов*						
Передающая сторона							*Приемная сторона*						
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*				
0	*1*	1	1	0	0		***0***	*1*	1	1	0	0	
1	*1*	0	1	1	0		***1***	*1*	0	1	1	0	
0	*1*	1	0	1	1		***0***	*1*	1	0	1	1	
0	*0*	0	1	0	1		***0***	*0*	0	1	0	1	
0	*0*	1	0	1	0		***0***	*1*	1	0	1	0	
0	*1*	1	1	0	1		***0***	*1*	1	1	0	1	
1	*0*	1	1	1	0		***1***	*0*	1	1	1	0	
1	*1*	1	1	1	1		***1***	*0*	1	1	1	1	
0	*0*	0	1	1	1		***0***	*0*	0	1	1	1	
1	*1*	0	0	1	1		***1***	*1*	0	0	1	1	
0	*0*	0	0	0	1		***0***	*0*	0	0	0	1	
0	*1*	1	0	0	0		***0***	*1*	1	0	0	0	
1	*1*	0	1	0	0		***1***	*1*	0	1	0	0	
0	*0*	0	0	1	0		***0***	*0*	0	0	1	0	

Оборудование и материалы

Аппаратура. Для выполнения лабораторной работы необходим персональный компьютер.

Программное обеспечение. Для выполнения лабораторной работы необходима операционная система с поддержкой графического окружения, установленный офисный пакет приложений, векторный графический редактор, редактор диаграмм и блок-схем.

Задания

1. Изучить теоретический материал работы.
2. Провести исследование системы поточного шифрования, которая использует синхронную ПСП.

Ход работы

Студенты получают задание, согласно номеру в журнале. Используя данные, представленные в таблице 5.5, необходимо произвести разработку генератора ПСП. Провести исследование проце-

дуры получения синхронной ПСП, зашифрования открытого текста (длина 15 двоичных разрядов). Разработать структуру дешифратора. Провести исследование процедуры расшифрования при отсутствии ошибки, при наличии двукратной ошибки, при наличии ошибки – выпадение символа.

В таблице 5.5 порождающие неприводимые полиномы представлены в 16-ричной системе счисления. Необходимо провести перевод в двоичную систему счисления, а затем перейти к полиномиальной форме представления.

Пример представления в полиномиальной форме заданного в шестнадцатеричной системе счисления.

$$8B = 10001011 = x^7 + x^3 + x + 1.$$

Таблица 5.5

Порождающие полиномы

№ п/п	*Исходные данные*	*№ п/п*	*Исходные данные*
1.	25	15.	91
2.	29	16.	9D
3.	2F	17.	A7
4.	37	18.	AB
5.	3D	19.	B9
6.	43	20.	BF
7.	49	21.	C1
8.	57	22.	CB
9.	5B	23.	D3
10.	61	24.	E5
11.	6D	25.	EF
12.	73	26.	F1
13.	83	27.	FD
14.	87	28.	11D

Содержание отчета

Отчет по лабораторной работе оформляется в виде документа Word и должен включать:

1) название лабораторной работы;
2) цель лабораторной работы;
3) формулировку индивидуального задания студенту и результат его выполнения;
4) краткие выводы по результатам выполнения лабораторной работы.

Вопросы для защиты работы

1. Проведите сравнительную оценку блочных и поточных шифров.
2. В чем проявляются достоинства поточных шифров?
3. Дайте классификацию систем поточного шифрования.
4. Определение синхронного поточного шифрования.
5. Свойства синхронного поточного шифрования.
6. Достоинства синхронных ПСП.
7. Недостатки синхронных ПСП.
8. Поясните алгоритм построения генератора ПСП
9. Основные способы построения М-последовательностей.
10. Дайте определение периода М-последовательностей.
11. По заданной степени порождающего полинома определите основные характеристики М-последовательностей.

Рекомендуемая литература

1. Рябко Б. Я., Фионов А. Н. Криптографические методы защиты информации. М.: Горячая линия-Телеком, 2012. 229 с.
2. Музыкантский А. И., Фурин В. В. Лекции по криптографии. М.: МЦНМО, 2011. 68 с.
3. Глухов М. М. Введение в теоретико-числовые методы криптографии / М. М. Глухов, И. А. Круглов, А. Б. Пичкур, А. В. Черемушкин. СПб.: Лань, 2011. 400 стр.
4. Алферов А. П., Зубов А. Ю. и др. Основы криптографии: учебное пособие. 2-е изд. М.: Гелиос АРВ, 2006. 480 с.
5. Баричев С. Г. Основы современной криптографии: учебный курс. М .: Телеком, 2007. 129 с.
6. Молдовян А. А. и др. Криптография. СПб.: Лань, 2007. 224 с.

6. ИССЛЕДОВАНИЕ ПОТОЧНОГО ШИФРОВАНИЯ СООБЩЕНИЙ В САМОСИНХРОНИЗИРУЮЩИХСЯ СИСТЕМАХ НА ОСНОВЕ МНОГОТАКТОВЫХ КОДОВЫХ ФИЛЬТРОВ

Цели работы:

1) углубить знания, которые были получены студентами на лекциях, по основам поточного шифрования;

2) исследовать вопросы получения самосинхронизирующейся псевдослучайных последовательностей на основе многотактовых кодовых фильтров.

Формируемые компетенции

– Способен проводить инструментальный анализ защищенности компьютерной системы и осуществлять поиск уязвимостей программного обеспечения (ПК-3);

– Способен выполнять автоматизированную информационно-аналитическую поддержку процессов принятия решений в профессиональной деятельности (ПК-5));

Теоретическая часть

В настоящее время различают алгоритмы поточного шифрования (гаммирование) с конечной и бесконечной гаммами. В первом случае источником гаммы является аппаратный или программный ГПК. Примером бесконечной гаммы может служить последовательность цифр в десятичной записи числа 3,1415926...

В том случае, если множеством используемых для шифрования знаков является алфавит, отличный от бинарного (Z_2 = {0,1}), например алфавит Z_{33} – русские буквы и пробел, его символы и символы гаммы заменяются цифровыми эквивалентами, которые затем суммируются по модулю *N:*

$$c_i = (\tilde{p}_i + \gamma_i) mod\, N, i = 1,2,..m$$

где $c_i, \tilde{p}_i, \gamma_i$ – очередной i-й знак соответственно исходного сообщения, гаммы и шифротекста; N – число символов в алфавите; m – число знаков открытого текста.

В самосинхронизирующихся поточных шифрах элементы входной последовательности зашифровываются с учетом N предшествующих элементов (рис. 6.1), которые принимают участие в формировании ключевой последовательности. В самосинхронизирующихся шифрах имеет место эффект размножения ошибок, в то же время, в отличие от синхронных, восстановление синхронизации происходит автоматически через *N* элементов зашифрованной последовательности.

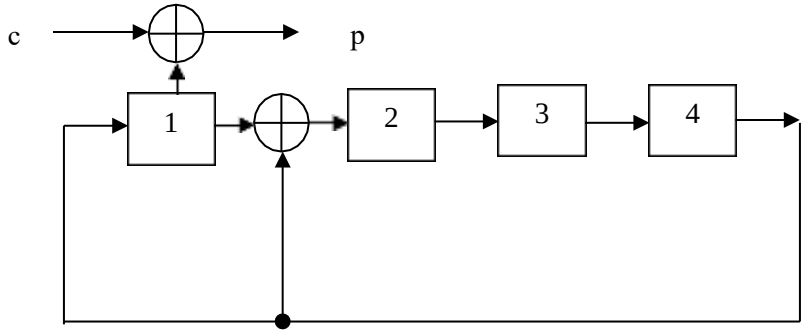

Рис. 6.1. Схема устройства зашифрования с ПСП

В таблице 6.1 показан пример шифрования и расшифрования двоичной последовательности **11100111010100** с использованием 4-разрядного *LFSR* при начальном состоянии, равном 1001. Зашифрованная последовательность имеет вид **01011001111100**. При отсутствии искажений в канале после расшифрования получается исходная последовательность.

Таблица 6.1

Пример поточного шифрования и расшифрования двоичной последовательности, когда отсутствуют ошибки в принятой комбинации

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования, когда отсутствуют ошибки в принятой комбинации*						
Передающая сторона							*Приемная сторона*						
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*				
0	*1*	0	1	0	0		***0***	*1*	0	1	0	0	
1	*1*	1	0	1	0		***1***	*1*	1	0	1	0	
0	*1*	0	1	0	1		***0***	*1*	0	1	0	1	
1	*0*	1	0	1	0		***1***	*0*	1	0	1	0	
1	*0*	1	1	0	1		***1***	*0*	1	1	0	1	

Продолжение таблицы 6.1

0	*1*	0	1	1	0			***0***	*1*	0	1	1	0
0	*1*	0	0	1	1			***0***	*1*	0	0	1	1
1	*1*	1	0	0	1			***1***	*1*	1	0	0	1
1	*0*	1	1	0	0			***1***	*0*	1	1	0	0
1	*1*	1	1	1	0			***1***	*1*	1	1	1	0
1	*0*	1	1	1	1			***1***	*0*	1	1	1	1
1	*1*	1	1	1	1			***1***	*1*	1	1	1	1
0	*0*	0	1	1	1			***0***	*0*	0	1	1	1
0	*0*	0	0	1	1			***0***	*0*	0	0	1	1

В таблице 6.2 рассмотрена ситуация, когда при передаче зашифрованной последовательности был потерян третий, равный нулю бит и вместо правильной последовательности к получателю пришла последовательность **01111001111100.**

Видно, что после расшифрования может произойти искажение не более 4 бит (в общем случае не более *N),* следующих после выпавшего символа. В рассмотренном примере вместо 4-битовой строки 0011 будет получена строка 0010. Все остальные биты будут приняты без искажений.

Таблица 6.2

Пример поточного шифрования и расшифрования двоичной последовательности, когда при передаче был потерян третий бит

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования, когда при передаче был потерян третий бит*						
Передающая сторона							*Приемная сторона*						
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*				
0	*1*	0	1	0	0		***0***	*1*	0	1	0	0	
1	*1*	1	0	1	0		***1***	*1*	1	0	1	0	
0	*1*	0	1	0	1		***1***	*0*	0	1	0	1	
1	*0*	1	0	1	0		***1***	*0*	1	0	1	0	
1	*0*	1	1	0	1		***0***	*1*	1	1	0	1	
0	*1*	0	1	1	0		***0***	*0*	0	1	1	0	
0	*1*	0	0	1	1		***1***	*1*	0	0	1	1	
1	*1*	1	0	0	1		***1***	*0*	1	0	0	1	
1	*0*	1	1	0	0		***1***	*1*	1	1	0	0	
1	*1*	1	1	1	0		***1***	*0*	1	1	1	0	
1	*0*	1	1	1	1		***1***	*1*	1	1	1	1	
1	*1*	1	1	1	1		***0***	*0*	1	1	1	1	
0	*0*	0	1	1	1		***0***	*0*	0	1	1	1	
0	*0*	0	0	1	1								

В таблице 6.3 рассмотрена ситуация, когда при передаче зашифрованной последовательности произошло искажение первого (0–1) бита и вместо правильной последовательности пришла последовательность **11011001111100.** Видно, что после расшифрования, помимо неправильно принятого бита, могут исказиться еще не более 4 последующих. В примере будет неправильно принят первый бит и вместо правильной 4-битовой строки 1100 будет получено 1111.

Таблица 6.3

Пример поточного шифрования и расшифрования, двоичной последовательности, когда при передаче произошло искажение битов

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования, когда при передаче произошло искажение битов*						
Передающая сторона							*Приемная сторона*						
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*				
1	*1*	0	1	0	0		***1***	*0*	0	1	0	0	
1	*1*	1	0	1	0		***1***	*1*	1	0	1	0	
0	*1*	0	1	0	1		***0***	*1*	0	1	0	1	
1	*0*	1	0	1	0		***1***	*1*	1	0	1	0	
1	*0*	1	1	0	1		***1***	*1*	1	1	0	1	
0	*1*	0	1	1	0		***0***	*1*	0	1	1	0	
0	*1*	0	0	1	1		***0***	*1*	0	0	1	1	
1	*1*	1	0	0	1		***1***	*1*	1	0	0	1	
1	*0*	1	1	0	0		***1***	*0*	1	1	0	0	
1	*1*	1	1	1	0		***1***	*1*	1	1	1	0	
1	*0*	1	1	1	1		***1***	*0*	1	1	1	1	
1	*1*	1	1	1	1		***1***	*1*	1	1	1	1	
0	*0*	0	1	1	1		***0***	*0*	0	1	1	1	
0	*0*	0	0	1	1		***0***	*0*	0	0	1	1	

Оборудование и материалы

Аппаратура. Для выполнения лабораторной работы необходим персональный компьютер.

Программное обеспечение. Для выполнения лабораторной работы необходима операционная система с поддержкой графического окружения, установленный офисный пакет приложений, векторный графический редактор, редактор диаграмм и блок-схем.

Задания

1. Изучить теоретический материал работы.

2. Провести исследование системы поточного шифрования, которая использует самосинхронизирующую ПСП.

Ход работы

Студенты получают задание, согласно номеру в журнале. Используя данные, представленные в таблице 6.4, произвести разработку генератора ПСП. Провести исследование процедуры получения самосинхронизирующейся ПСП, зашифрования открытого текста (длина 15 двоичных разрядов). Разработать структуру дешифратора. Провести исследование процедуры расшифрования при отсутствии ошибки, при наличии двукратной ошибки, при наличии ошибки – выпадение символа.

В таблице 6.4 порождающие неприводимые полиномы представлены в шестнадцатиричной системе счисления. Необходимо провести перевод в двоичную систему счисления, а затем перейти к полиномиальной форме представления.

Таблица 6.4

Порождающие полиномы

№ п/п	*Исходные данные*	*№ п/п*	*Исходные данные*
1.	25	29.	91
2.	29	30.	9D
3.	2F	31.	A7
4.	37	32.	AB
5.	3D	33.	B9
6.	43	34.	BF
7.	49	35.	C1
8.	57	36.	CB
9.	5B	37.	D3
10.	61	38.	E5
11.	6D	39.	EF
12.	73	40.	F1
13.	83	41.	FD
14.	87	42.	11D

Содержание отчета

Отчет по лабораторной работе оформляется в виде документа Word и должен включать:

1) название лабораторной работы;
2) цель лабораторной работы;

3) формулировку индивидуального задания и результат его выполнения;

4) краткие выводы по результатам выполнения лабораторной работы.

Вопросы для защиты работы

1. В чем проявляются достоинства поточных шифров?
2. Дайте классификацию систем поточного шифрования.
3. Проведите сравнительную оценку блочных и поточных шифров.
4. Определение самосинхронизирующегося поточного шифрования.
5. Свойства самосинхронизирующегося поточного шифрования.
6. Сравните два вида ПСП.
7. Достоинства самосинхронизирующегося поточного шифрования.
8. Недостатки самосинхронизирующегося поточного шифрования.
9. Поясните алгоритм построения генератора ПСП, используемого при зашифровании текста.
10. Основные способы построения М-последовательностей.
11. Поясните алгоритм построения генератора ПСП, используемого для расшифрования закрытого текста.

Рекомендуемая литература

1. Рябко Б. Я., Фионов А. Н. Криптографические методы защиты информации. М.: Горячая линия-Телеком, 2012. 229 с.
2. Музыкантский А. И., Фурин В. В. Лекции по криптографии. М.: МЦНМО, 2011. 68 с.
3. Глухов М. М. Введение в теоретико-числовые методы криптографии / М. М. Глухов, И. А. Круглов, А. Б. Пичкур, А. В. Черемушкин. СПб.: Лань, 2011. 400 с.
4. Торстейнсон П., Ганеш Г.А. Криптография и безопасность в технологии .NET. М.: Бином; Лаборатория знаний, 2013. 480 с.
5. Петров А. А. Компьютерная безопасность. Криптографические методы защиты. М.: ДМК Пресс, 2008. 448 с.
6. Алферов А. П., Зубов А. Ю. и др. Основы криптографии: учебное пособие. 2-е изд. М.: Гелиос АРВ, 2006. 480 с.
7. Баричев С. Г. Основы современной криптографии: учебный курс. М.: Телеком, 2007. 129 с.
8. Молдовян А. А. и др. Криптография. СПб.: Лань, 2007. 224 с.
9. Введение в криптографию / под ред. В. В. Ященко. М.: МЦНМО, 2006. 288 с.

7. ИССЛЕДОВАНИЕ ПОТОЧНОГО ШИФРОВАНИЯ СООБЩЕНИЙ В СИНХРОНИЗИРУЮЩИХСЯ СИСТЕМАХ, ПОСТРОЕННЫХ НА ОСНОВЕ ГЕНЕРАТОРОВ ФИББОНАЧИ

Цели работы:

1) углубить знания, которые были получены студентами на лекциях, по основам поточного шифрования;

2) исследовать вопросы получения синхронного ПСП с использованием генератора типа Фиббоначчи.

Формируемые компетенции

– Способен проводить инструментальный анализ защищенности компьютерной системы и осуществлять поиск уязвимостей программного обеспечения (ПК-3);

– Способен выполнять автоматизированную информационно-аналитическую поддержку процессов принятия решений в профессиональной деятельности (ПК-5));

Теоретическая часть

Шифр Вернама можно считать исторически первым поточным шифром. Так как поточные шифры, в отличие от блочных, осуществляют поэлементное шифрование потока данных без задержки в криптосистеме, их важнейшим достоинством является высокая скорость преобразования, соизмеримая со скоростью поступления входной информации. Таким образом, обеспечивается шифрование практически в реальном масштабе времени вне зависимости от объема и разрядности потока преобразуемых данных.

Простейшие устройства синхронного и самосинхронизирующегося шифрования с использованием ГПК, реализованного на основе *N*-разрядного *регистра сдвига с линейной обратной связью – LFSR* (Linear Feedback Shift Register), называются *скремблерами,* а сам процесс преобразования – скремблированием.

В синхронных поточных шифрах гамма формируется независимо от входной последовательности, каждый элемент (бит, символ, байт и т. п.) которой таким образом шифруется независимо от других элементов. В синхронных поточных шифрах отсутствует эффект размноже-

ния ошибок, т. е. число искаженных элементов в расшифрованной последовательности равно числу искаженных элементов зашифрованной последовательности, пришедшей из канала связи.

Вставка или выпадение элемента зашифрованной последовательности недопустимы, так как из-за нарушения синхронизации это приведет к неправильному расшифрованию всех последующих элементов. Синхронное поточное шифрование с использованием LFSR показано на рис. 7.1.

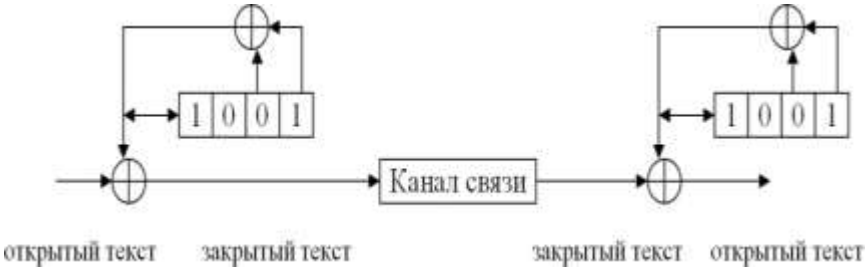

Рис. 7.1 Синхронное поточное шифрование

В таблице 7.1 показан пример поточного шифрования и расшифрования двоичной последовательности *11100101010110* с использованием гаммы формируемой четырехразрядным *LFSR* при начальном состоянии **1001**. Зашифрованная последовательность имеет вид ***01001010010010.***

При отсутствии искажений в канале связи после расшифрования с использованием той же гаммы получается исходная последовательность

Таблица 7.1

Пример поточного шифрования и расшифрования двоичной последовательности, когда отсутствуют ошибки в принятой комбинации

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования, когда отсутствуют ошибки в принятой комбинации*						
Передающая сторона							*Приемная сторона*						
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*				
0	*1*	1	1	0	0		***0***	*1*	1	1	0	0	
1	*1*	0	1	1	0		***1***	*1*	0	1	1	0	
0	*1*	1	0	1	1		***0***	*1*	1	0	1	1	
0	*0*	0	1	0	1		***0***	*0*	0	1	0	1	

Продолжение таблицы 7.1

1	*0*	1	0	1	0
0	*1*	1	1	0	1
1	*0*	1	1	1	0
0	*1*	1	1	1	1
0	*0*	0	1	1	1
1	*1*	0	0	1	1
0	*0*	0	0	0	1
0	*1*	1	0	0	0
1	*1*	0	1	0	0
0	*0*	0	0	1	0

1	*0*	1	0	1	0
0	*1*	1	1	0	1
1	*0*	1	1	1	0
0	*1*	1	1	1	1
0	*0*	0	1	1	1
1	*1*	0	0	1	1
0	*0*	0	0	0	1
0	*1*	1	0	0	0
1	*1*	0	1	0	0
0	*0*	0	0	1	0

В таблице 7.2 рассмотрена ситуация, когда при передаче зашифрованной последовательности был потерян четвертый бит, и вместо правильной последовательности к получателю пришла последовательность **0101010010010.**

Таблица 7.2

Пример поточного шифрования и расшифрования двоичной последовательности, когда при передаче был потерян четвертый бит

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования, когда при передаче был потерян четвертый бит*						
Передающая сторона							*Приемная сторона*						
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*				
0	*1*	1	1	0	0		***0***	*1*	1	1	0	0	
1	*1*	0	1	1	0		***1***	*1*	0	1	1	0	
0	*1*	1	0	1	1		***0***	*1*	1	0	1	1	
0	*0*	0	1	0	1		***1***	*1*	0	1	0	1	
1	*0*	1	0	1	0		***0***	*1*	1	0	1	0	
0	*1*	1	1	0	1		***1***	*0*	1	1	0	1	
1	*0*	1	1	1	0		***0***	*1*	1	1	1	0	
0	*1*	1	1	1	1		***0***	*1*	1	1	1	1	
0	*0*	0	1	1	1		***1***	*1*	0	1	1	1	
1	*1*	0	0	1	1		***0***	*0*	0	0	1	1	
0	*0*	0	0	0	1		***0***	*0*	0	0	0	1	
0	*1*	1	0	0	0		***1***	*0*	1	0	0	0	
1	*1*	0	1	0	0		***0***	*0*	0	1	0	0	
0	*0*	0	0	1	0								

Видно, что после расшифрования всех битов, следующих после выпавшего, происходят искажения информации. В результате вместо битовой строки **0101010110** будет получена строка **1101110000.**

В таблице 7.3 рассмотрена ситуация, когда при передаче зашифрованной последовательности произошло искажение пятого (1–0) и восьмого (0–1) битов и вместо правильной последовательности к получателю пришла последовательность **01000011010010.** Видно, что после расшифрования вместо правильной строки будет получена строка **11101100010110** с искаженным пятым (0–1) и восьмым (1–0) битами.

Таблица 7.3

Пример поточного шифрования и расшифрования двоичной последовательности, когда при передаче произошло искажение битов

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования, когда при передаче произошло искажение битов*						
Передающая сторона							*Приемная сторона*						
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*				
0	*1*	1	1	0	0		***0***	*1*	1	1	0	0	
1	*1*	0	1	1	0		***1***	*1*	0	1	1	0	
0	*1*	1	0	1	1		***0***	*1*	1	0	1	1	
0	*0*	0	1	0	1		***0***	*0*	0	1	0	1	
0	*0*	1	0	1	0		***0***	*1*	1	0	1	0	
0	*1*	1	1	0	1		***0***	*1*	1	1	0	1	
1	*0*	1	1	1	0		***1***	*0*	1	1	1	0	
1	*1*	1	1	1	1		***1***	*0*	1	1	1	1	
0	*0*	0	1	1	1		***0***	*0*	0	1	1	1	
1	*1*	0	0	1	1		***1***	*1*	0	0	1	1	
0	*0*	0	0	0	1		***0***	*0*	0	0	0	1	
0	*1*	1	0	0	0		***0***	*1*	1	0	0	0	
1	*1*	0	1	0	0		***1***	*1*	0	1	0	0	
0	*0*	0	0	1	0		***0***	*0*	0	0	1	0	

Оборудование и материалы

Аппаратура. Для выполнения лабораторной работы необходим персональный компьютер.

Программное обеспечение. Для выполнения лабораторной работы необходима операционная система с поддержкой графического окружения, установленный офисный пакет приложений, векторный графический редактор, редактор диаграмм и блок-схем.

Задания

1. Изучить теоретический материал работы.

2. Провести исследование системы поточного шифрования, которая использует синхронную ПСП, полученная с помощью генератора типа Фибоначчи.

Ход работы

Студенты получают задание, согласно номеру в журнале. Используя данные, представленные в таблице 7.4, произвести разработку генератора ПСП. Провести исследование процедуры получения синхронной ПСП, зашифрования открытого текста (длина 15 двоичных разрядов).

Разработать структуру дешифратора.

Провести исследование процедуры расшифрования при отсутствии ошибки, при наличии двукратной ошибки, при наличии ошибки типа выпадение символа.

В таблице 7.4 порождающие неприводимые полиномы представлены в шестнадцатиричной системе счисления. Необходимо провести перевод в двоичную систему счисления, а затем перейти к полиномиальной форме представления.

Таблица 7.4

Порождающие полиномы

№ п/п	*Исходные данные*	*№ п/п*	*Исходные данные*
1	25	16	49
2	29	17	9D
3	2F	18	A7
4	37	19	AB
5	3D	20	B9
6	43	21	BF
7	49	22	C1
8	57	23	CB
9	5B	24	D3
10	61	25	91
11	6D	26	EF
12	73	27	F1
13	83	28	FD
14	87	29	11D
15	91	30	E5

Содержание отчета

Отчет по лабораторной работе оформляется в виде документа Word и должен включать:

1) название лабораторной работы;
2) цель лабораторной работы;
3) формулировку индивидуального задания для студента и результат его выполнения;
4) краткие выводы по результатам выполнения лабораторной работы.

Вопросы для защиты работы

1. Проведите сравнительную оценку блочных и поточных шифров.
2. В чем проявляются достоинства поточных шифров?
3. Дайте классификацию систем поточного шифрования.
4. Определение синхронного поточного шифрования.
5. Свойства синхронного поточного шифрования.
6. Достоинства синхронных ПСП.
7. Недостатки синхронных ПСП.
8. Поясните алгоритм построения генератора ПСП
9. Основные способы построения М-последовательностей.
10. Как определяется длина двоичной М-последовательности?
11. Пусть задан неприводимый полином 8 степени. Определите длину цикла вырабатываемой ПСП.
12. По заданному порождающему полиному построить структуру генератора ПСП.
13. По заданному порождающему полиному определить основные характеристики полученной ПСП.
14. Можно ли снимать ПСП с выходов разных триггеров генератора ПСП?

Рекомендуемая литература

1. Рябко Б. Я., Фионов А. Н. Криптографические методы защиты информации. М.: Горячая линия-Телеком, 2012. 229 с.
2. Музыкантский А. И., Фурин В. В. Лекции по криптографии. М.: МЦНМО, 2011. 68 с.
3. Глухов М. М. Введение в теоретико-числовые методы криптографии / М. М. Глухов, И. А. Круглов, А. Б. Пичкур, А. В. Черемушкин. СПб.: Лань, 2011. 400 стр.

4. Алферов А. П., Зубов А. Ю. и др. Основы криптографии: учебное пособие. 2-е изд. М.: Гелиос АРВ, 2006. 480 с.

5. Баричев С. Г. Основы современной криптографии: учебный курс. М .: Телеком, 2007. 129 с.

6. Молдовян А. А. и др. Криптография. СПб.: Лань, 2007. 224 с.

8. ИССЛЕДОВАНИЕ ПОТОЧНОГО ШИФРОВАНИЯ СООБЩЕНИЙ В САМОСИНХРОНИЗИРУЮЩИХСЯ СИСТЕМАХ НА ОСНОВЕ ГЕНЕРАТОРОВ ТИПА ФИБОНАЧЧИ

Цели работы:

1) углубить знания, полученные на лекциях по основам поточного шифрования;

2) исследовать вопросы получения самосинхронизирующейся ПСП, используя генератор типа Фиббоначчи.

Формируемые компетенции

– Способен проводить инструментальный анализ защищенности компьютерной системы и осуществлять поиск уязвимостей программного обеспечения (ПК-3);

– Способен выполнять автоматизированную информационно-аналитическую поддержку процессов принятия решений в профессиональной деятельности (ПК-5));

Теоретическая часть

В самосинхронизирующихся поточных шифрах элементы входной последовательности зашифровываются с учетом N предшествующих элементов (рисунок 8.1), которые принимают участие в формировании ключевой последовательности. В самосинхронизирующихся шифрах имеет место эффект размножения ошибок, в то же время в отличие от синхронных, восстановление синхронизации происходит автоматически через *N* элементов зашифрованной последовательности.

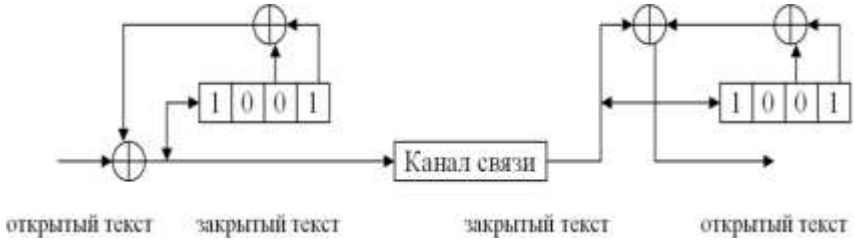

Рис. 8.1. Схема устройства зашифрования с ПСП

Таблица 8.1

Пример поточного шифрования и расшифрования двоичной последовательности, когда отсутствуют ошибки в принятой комбинации

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования, когда отсутствуют ошибки в принятой комбинации*						
Передающая сторона							*Приемная сторона*						
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*				
0	*1*	0	1	0	0		***0***	*1*	0	1	0	0	
1	*1*	1	0	1	0		***1***	*1*	1	0	1	0	
0	*1*	0	1	0	1		***0***	*1*	0	1	0	1	
1	*0*	1	0	1	0		***1***	*0*	1	0	1	0	
1	*0*	1	1	0	1		***1***	*0*	1	1	0	1	
0	*1*	0	1	1	0		***0***	*1*	0	1	1	0	
0	*1*	0	0	1	1		***0***	*1*	0	0	1	1	
1	*1*	1	0	0	1		***1***	*1*	1	0	0	1	
1	*0*	1	1	0	0		***1***	*0*	1	1	0	0	
1	*1*	1	1	1	0		***1***	*1*	1	1	1	0	
1	*0*	1	1	1	1		***1***	*0*	1	1	1	1	
1	*1*	1	1	1	1		***1***	*1*	1	1	1	1	
0	*0*	0	1	1	1		***0***	*0*	0	1	1	1	
0	*0*	0	0	1	1		***0***	*0*	0	0	1	1	

В таблице 8.1 показан пример шифрования и расшифрования двоичной последовательности **11100111010100** с использованием 4-разрядного *LFSR* при начальном состоянии, равном 1001. Зашифрованная последовательность имеет вид **01011001111100.** При отсутствии искажений в канале после расшифрования получается исходная последовательность. В таблице 8.2 рассмотрена ситуация, когда при передаче зашифрованной последовательности был потерян тре-

тий, равный нулю бит и вместо правильной последовательности к получателю пришла последовательность **01111001111100.**

Видно, что после расшифрования может произойти искажение не более 4 бит (в общем случае не более *N),* следующих после выпавшего символа. В рассмотренном примере вместо 4-битовой строки 0011 будет получена строка 0010. Все остальные биты будут приняты без искажений

Таблица 8.2

Пример поточного шифрования и расшифрования двоичной последовательности, когда при передаче был потерян третий бит

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования, когда при передаче был потерян третий бит*					
Передающая сторона							*Приемная сторона*					
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*			
0	*1*	0	1	0	0		***0***	*1*	0	1	0	0
1	*1*	1	0	1	0		***1***	*1*	1	0	1	0
0	*1*	0	1	0	1		***1***	*0*	0	1	0	1
1	*0*	1	0	1	0		***1***	*0*	1	0	1	0
1	*0*	1	1	0	1		***0***	*1*	1	1	0	1
0	*1*	0	1	1	0		***0***	*0*	0	1	1	0
0	*1*	0	0	1	1		***1***	*1*	0	0	1	1
1	*1*	1	0	0	1		***1***	*0*	1	0	0	1
1	*0*	1	1	0	0		***1***	*1*	1	1	0	0
1	*1*	1	1	1	0		***1***	*0*	1	1	1	0
1	*0*	1	1	1	1		***1***	*1*	1	1	1	1
1	*1*	1	1	1	1		***0***	*0*	1	1	1	1
0	*0*	0	1	1	1		***0***	*0*	0	1	1	1
0	*0*	0	0	1	1							

В таблице 8.3 рассмотрена ситуация, когда при передаче зашифрованной последовательности произошло искажение первого (0–1) бита и вместо правильной последовательности пришла последовательность **11011001111100**. Видно, что после расшифрования, помимо неправильно принятого бита, могут исказиться еще не более 4 последующих. В примере будет неправильно принят первый бит и вместо правильной 4-битовой строки 1100 будет получено 1111.

Таблица 8.3

Пример поточного шифрования и расшифрования, двоичной последовательности, когда при передаче произошло искажение битов

Процедура поточного шифрования двоичной последовательности							*Процедура поточного расшифрования, когда при передаче произошло искажение битов*						
Передающая сторона							*Приемная сторона*						
c	*p*	*Генератор ПСП*					*c*	*p*	*Генератор ПСП*				
1	*1*	0	1	0	0		***1***	*0*	0	1	0	0	
1	*1*	1	0	1	0		***1***	*1*	1	0	1	0	
0	*1*	0	1	0	1		***0***	*1*	0	1	0	1	
1	*0*	1	0	1	0		***1***	*1*	1	0	1	0	
1	*0*	1	1	0	1		***1***	*1*	1	1	0	1	
0	*1*	0	1	1	0		***0***	*1*	0	1	1	0	
0	*1*	0	0	1	1		***0***	*1*	0	0	1	1	
1	*1*	1	0	0	1		***1***	*1*	1	0	0	1	
1	*0*	1	1	0	0		***1***	*0*	1	1	0	0	
1	*1*	1	1	1	0		***1***	*1*	1	1	1	0	
1	*0*	1	1	1	1		***1***	*0*	1	1	1	1	
1	*1*	1	1	1	1		***1***	*1*	1	1	1	1	
0	*0*	0	1	1	1		***0***	*0*	0	1	1	1	
0	*0*	0	0	1	1		***0***	*0*	0	0	1	1	

Оборудование и материалы

Аппаратура. Для выполнения лабораторной работы необходим персональный компьютер.

Программное обеспечение. Для выполнения лабораторной работы необходима операционная система с поддержкой графического окружения, установленный офисный пакет приложений, векторный графический редактор, редактор диаграмм и блок-схем.

Задания

1. Изучить теоретический материал работы.

2. Провести исследование системы поточного шифрования, которая использует самосинхронизирующуюся ПСП, которая получена с помощью генератора типа Фибоначчи.

Ход работы

Студенты получают задание, согласно номеру в журнале. Используя данные, представленные в таблице 8.4, произвести разработку ге-

нератора ПСП. Провести исследование процедуры получения синхронной ПСП, зашифрования открытого текста (длина 15 двоичных разрядов). Разработать структуру дешифратора. Провести исследование процедуры расшифрования при отсутствии ошибки, при наличии двукратной ошибки, при наличии ошибки – выпадение символа.

В таблице 8.4 порождающие неприводимые полиномы представлены в шестнадцатиричной системе счисления. Необходимо провести перевод в двоичную систему счисления, а затем перейти к полиномиальной форме предсталвения.

Таблица 8.4

Порождающие полиномы

№ п/п	Исходные данные	№ п/п	Исходные данные
1.	25	16	49
2.	29	17	9D
3.	2F	18	A7
4.	37	19	AB
5.	3D	20	B9
6.	43	21	BF
7.	49	22	C1
8.	57	23	CB
9.	5B	24	D3
10.	61	25	91
11.	6D	26	EF
12.	73	27	F1
13.	83	28	FD
14.	87	29	11D
15.	91	30	E5

Содержание отчета

Отчет по лабораторной работе оформляется в виде документа Word и должен включать:

1) название лабораторной работы;
2) цель лабораторной работы;
3) формулировку индивидуального задания и результат его выполнения;
4) краткие выводы по результатам выполнения лабораторной работы.

Вопросы для защиты работы

1. В чем проявляются достоинства поточных шифров?
2. Дайте классификацию систем поточного шифрования.
3. Проведите сравнительную оценку блочных и поточных шифров.
4. Определение самосинхронизирующегося поточного шифрования.
5. Свойства самосинхронизирующегося поточного шифрования.
6. Сравните два вида ПСП.
7. Достоинства самосинхронизирующегося поточного шифрования.
8. Недостатки самосинхронизирующегося поточного шифрования.
9. Поясните алгоритм построения генератора ПСП, используемого при зашифрования текста.
10. Основные способы построения М-последовательностей.
11. Поясните алгоритм построения генератора ПСП, используемого для расшифрования закрытого текста

Рекомендуемая литература

1. Рябко Б.Я. Фионов А.Н. Криптографические методы защиты информации. М.: Горячая линия-Телеком, 2012. 229 с.
2. Музыкантский А. И., Фурин В. В. Лекции по криптографии. М.: МЦНМО, 2011. 68 с.
3. Глухов М. М. Введение в теоретико-числовые методы криптографии / М. М. Глухов, И. А. Круглов, А. Б. Пичкур, А. В. Черемушкин. СПб.: Лань, 2011. 400 стр.
4. Алферов А. П., Зубов А. Ю. и др. Основы криптографии: учебное пособие. 2-е изд. М.: Гелиос АРВ, 2006. 480 с.
5. Баричев С. Г. Основы современной криптографии: учебный курс. М .: Телеком, 2007. 129 с.
6. Молдовян А. А. и др. Криптография. СПб.: Лань, 2007. 224 с.

9. ИССЛЕДОВАНИЕ ПРОЦЕССА АССИМЕТРИЧНОГО ШИФРОВАНИЯ БЕЗ ПЕРЕДАЧИ КЛЮЧА

Цели работы:

1) углубить знания, полученные на лекциях по основам шифрования без передачи ключа;

2) исследовать основные характеристики алгоритма шифрования без передачи ключа.

Формируемые компетенции

– Способен проводить инструментальный анализ защищенности компьютерной системы и осуществлять поиск уязвимостей программного обеспечения (ПК-3);

– Способен выполнять автоматизированную информационно-аналитическую поддержку процессов принятия решений в профессиональной деятельности (ПК-5));

Теоретическая часть

Эффективными системами криптографической защиты данных являются асимметричные криптосистемы, называемые также криптосистемами с открытым ключом. В таких системах для зашифрования данных используется один ключ, а для расшифрования – другой ключ (отсюда и название – асимметричные системы шифрования).

Первый ключ является открытым и может быть опубликован для использования всеми пользователями системы, которые зашифровывают данные. Процедура расшифрования данных с помощью открытого ключа совсем невозможна из-за особенностей построения алгоритма.

Для расшифрования данных получатель зашифрованной информации использует второй ключ, который является секретным. Разумеется, ключ расшифрования не может быть определен из ключа зашифрования.

Обобщенная схема асимметричной криптосистемы с открытым ключом показана на рис. 9.1. В этой криптосистеме применяют два различных ключа:

$K_в$ – открытый ключ отправителя А;

$k_в$ – секретный ключ получателя В.

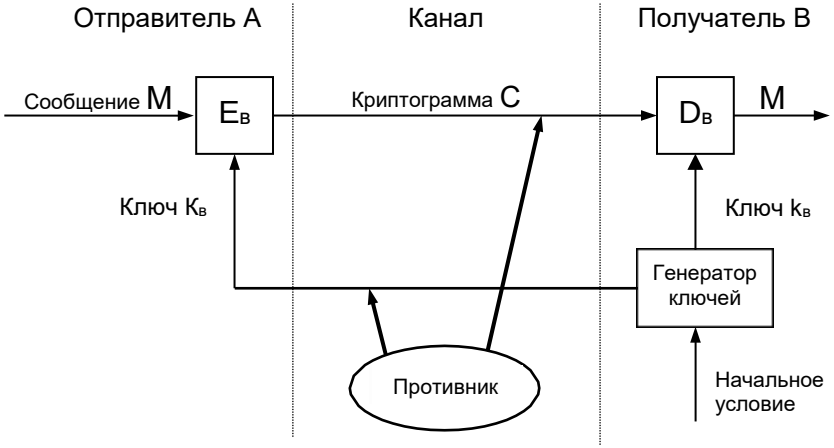

Рис. 9.1. Обобщенная схема асимметричной криптосистемы с открытым ключом

Генератор ключей целесообразно располагать на стороне получателя B (чтобы не пересылать секретный ключ $k_в$ при передаче по незащищенному каналу). Значения ключей $K_в$ и $k_в$ зависят от начального состояния генератора ключей.

Раскрытие секретного ключа $k_в$ по известному открытому ключу $K_в$ должно быть вычислительно неразрешимой задачей.

Характерные особенности асимметричных криптосистем:

1. Открытый ключ $K_в$ и криптограмма C могут быть отправлены по незащищенным каналам, т. е. противнику известны $K_в$ и C.

2. Алгоритмы шифрования и расшифрования

$$E_в : M \to C,$$
$$D_в : C \to M$$

являются открытыми.

Защита информации в асимметричной криптосистеме основана на секретности ключа $k_в$.

У. Диффи и М. Хеллман сформулировали требования, выполнение которых обеспечивает безопасность асимметричной криптосистемы:

1. Вычисление пары ключей ($K_в$, $k_в$) получателем B на основе начального условия должно быть простым.

2. Отправитель A, зная открытый ключ Кв и сообщение M, может легко вычислить криптограмму

$$C = E_{K_в}(M) = E_в(M). \quad (1)$$

2. Получатель B, используя секретный ключ $k_в$ и криптограмму C, может легко восстановить исходное сообщение

$$M = D_{k_в}(C) = D_в(C) = D_в[E_в(M)]. \quad (2)$$

4. Противник, зная открытый ключ $К_в$, при попытке вычислить секретный ключ $k_в$ наталкивается на непреодолимую вычислительную проблему.

5. Противник, зная пару ($К_в$, C), при попытке вычислить исходное сообщение M наталкивается на непреодолимую вычислительную проблему, которая имеет высокую криптостойкость.

Рассмотрим более подробно один из первых алгоритмов асимметричного шифрования данных. Этот алгоритм носит название – без передачи секретных ключей. Алгоритм работы такой асимметричной системы шифрования состоит в следующем.

Пусть стороны *A* и *B* решили организовать секретную передачу информации между собой. Для этого выбирается большое число *p*, такое, что *p-1* хорошо разлагается на простые множители.

Далее абоненты независимо друг от друга осуществляют выбор ключей *a* и *b*, которые являются первыми секретными, согласно следующим правилам:

$$1 < a < p - 2;\ 1 < b < p - 2.$$

Затем стороны производят вычисления вторых ключей, которые также являются секретными, согласно условию:

$$A : a \to \alpha \quad \alpha \cdot a \equiv 1 mod\ \varphi(p)$$

$$B : b \to \beta \quad \beta \cdot b \equiv 1 mod\ \varphi(p),$$

где $\varphi(p)$ – функция Эйлера.

3. Пусть абонент *A* решает передать сообщение *n*. Тогда сторона *A* зашифровывает это сообщение своим первым закрытым ключом:

$$n_1 = n^a\ mod\ p$$

Полученное сообщение отправляется на сторону B.

4. Тогда сторона *B* зашифровывает принятое сообщение своим первым закрытым ключом:

$$n_2 = n_1^b\ mod\ p$$

Полученное сообщение отправляется на сторону *B*.

5. Тогда сторона *A* зашифровывает принятое сообщение своим вторым закрытым ключом:

$$n_3 = n_2^{\alpha} \bmod p$$

Полученное сообщение отправляется на сторону *B*.

6. Тогда сторона *B* зашифровывает принятое сообщение своим первым закрытым ключом:

$$n = n_3^{\beta} \bmod p$$

Полученное сообщение отправлялось на сторону *B* от абонента *A*.

Основным недостатком рассмотренной системы криптозащиты является неоднократная передача информации от одного пользователя к другому, что в конечном итоге приводит к низкой скорости передачи информации. Кроме того, довольно сложно подобрать число *p*, чтобы обеспечить необходимую надёжность защиты.

Оборудование и материалы

Аппаратура. Для выполнения лабораторной работы необходим персональный компьютер.

Программное обеспечение. Для выполнения лабораторной работы необходима операционная система с поддержкой графического окружения, установленный офисный пакет приложений, векторный графический редактор, редактор диаграмм и блок-схем.

Задания

1. Изучить теоретический материал работы.
2. Провести исследование системы асимметричного шифрования без передачи ключей.

Ход работы

Передаваемое сообщение m и кодирующее число p представлены в таблице 9.1.

Таблица 9.1

Задание для исследования системы шифрования

Вариант	*Кодирующее число p*	*Сообщение m*
1	23	17
2	29	18
3	31	19
4	37	20

5	41	21

Продолжение таблицы 9.1

6	43	22
7	47	23
8	53	24
9	23	16
10	29	15
11	31	14
12	37	13
13	41	28
14	43	29
15	47	30
16	53	31
17	23	14
18	29	15
19	31	16
20	37	17
21	41	18
22	43	22
23	47	23
24	53	24
25	23	16
26	29	19
27	31	20
28	37	21
29	41	22
30	43	23
31	47	24
32	53	25
33	37	26
34	41	27

Студенты самостоятельно выбирают значение ключей для обеих сторон (A и B) и исследуют процесс зашифрования и расшифрования сообщения.

Содержание отчета

Отчет по лабораторной работе оформляется в виде документа Word и должен включать:

1) название лабораторной работы;
2) цель лабораторной работы;
3) формулировку индивидуального задания и результат его выполнения;

4) краткие выводы по результатам выполнения лабораторной работы.

Вопросы для защиты работы

1. Принципы построения алгоритма шифрования без передачи ключей.
2. Поясните, чем обоснован выбор разработки асимметричных систем шифрования.
3. Поясните алгоритм работы асимметричных систем шифрования.
4. Как осуществляется выбор первого секретного ключа?
5. Как осуществляется выбор второго секретного ключа?
6. Для заданного модуля *p* нейдите пару секретных ключей для выполнения алгоритма без передачи ключей.
7. Основные характеристики алгоритма шифрования.
8. Для заданных ключей осуществить первые два этапа шифрования данных.
9. Достоинства шифрования без передачи ключа.
10. Недостатки шифрования без передачи ключа.

Рекомендуемая литература

1. Рябко Б. Я., Фионов А. Н. Криптографические методы защиты информации. М.: Горячая линия-Телеком, 2012. 229 с.
2. Музыкантский А. И., Фурин В. В. Лекции по криптографии. М.: МЦНМО, 2011. 68 с.
3. Глухов М. М. Введение в теоретико-числовые методы криптографии / М. М. Глухов, И. А. Круглов, А. Б. Пичкур, А. В. Черемушкин. СПб.: Лань, 2011. 400 с.
4. Торстейнсон П., Ганеш Г.А. Криптография и безопасность в технологии .NET. М.: Бином; Лаборатория знаний, 2013. 480 с.
5. Петров А. А. Компьютерная безопасность. Криптографические методы защиты. М.: ДМК Пресс, 2008. 448 с.
6. Алферов А. П., Зубов А. Ю. и др. Основы криптографии: учебное пособие. 2-е изд. М.: Гелиос АРВ, 2006. 480 с.
7. Баричев С. Г. Основы современной криптографии: учебный курс. М .: Телеком, 2007. 129 с.
8. Молдовян А. А. и др. Криптография. СПб.: Лань, 2007. 224 с.
9. Введение в криптографию / под ред В. В. Ященко. М.: МЦНМО, 2006. 288 с.