Лабораторная работа № 1

ИССЛЕДОВАНИЕ МЕТОДОВ ПОЛИАЛФАВИТНОЙ ПОДСТАНОВКИ

Цель работы

Изучение принципов построения моноалфавитных и полиалфавитных шифров замены. Исследование свойств подстановочных шифров.

Теоретическая часть

Моноалфавитные шифры замены имели существенный недостаток – они легко поддавались частотному криптоанализу. Возникла потребность в разработке более устойчивых методов шифрования. Так на смену моноалфавитным шифрам пришли шифры полиалфавитные.

Метод *Виженера* относится к числу полиалфавитных шифров замены. Берется небольшое целое число *m* и алфавит после каждой символьной подстановки сдвигается на *m* символов.

Например, если ключом будет слово *мышь* (смотри левую вертикальную колонку символов), тогда *m* = 4, при этом получаем следующую таблицу:

```
   абвгдеёжзийклмнопрстуфхцчшщъыьэюя
1  мнопрстуфхцчшщъыьэюяабвгдеёжзийкл
2  ыьэюяабвгдеёжзийклмнопрстуфхцчшщъ
3  шщъыьэюяабвгдеёжзийклмнопрстуфхцч
4  ьэюяабвгдеёжзийклмнопрстуфхцчшщъы
```

Исходный текст разбивается на группы по *m* символов (в рассмотренном случае – по 4). Для каждой группы первый символ заменяется соответствующей буквой из первого алфавита, второй – из второго и т.д. Например, фраза «от улыбки каждый день светлей» будет преобразована следующим образом:

```
отул ыбки кажд ыйде ньсв етле й
ынлз зьге чыяа зеьб ъчйю сндб ц
```

Алфавит не ограничивается только лишь буквами, в него можно добавить и другие символы – пробел, цифры, знаки препинания. Такая модификация позволит избежать двусмысленности при чтении текста после расшифровки на приёмной стороне (например, проблема простановки запятой во фразе «казнить нельзя помиловать»).

Ход работы

Получите индивидуальное задание у преподавателя. Реализуйте программный модуль в соответствии с полученным заданием. При реализации необходимо учесть следующие моменты:

1) предусмотреть возможность задания пользователем своего абсолютно произвольного алфавита, состоящего из любого набора символов, расположенных в любом порядке;

2) для удобства тестирования и взаимодействия с другими модулями реализовать файловый ввод исходных данных и файловый вывод результата криптографического преобразования.

После реализации программного модуля выполните статистический анализ текста до криптографического преобразования и после него.

Варианты заданий

1. Модуль для шифрования текста по алгоритму Виженера, ключ – слово или фраза.
2. Модуль для расшифровывания текста по алгоритму Виженера, ключ – слово или фраза.
3. Модуль для шифрования текста по алгоритму Виженера, ключ – числовая последовательность.
4. Модуль для расшифровывания текста по алгоритму Виженера, ключ – числовая последовательность.
5. Модуль для шифрования текста по алгоритму Гронсфельда, ключ – слово (до 10 символов).
6. Модуль для расшифровывания текста по алгоритму Гронсфельда, ключ – слово (до 10 символов).
7. Модуль для шифрования текста по алгоритму Гронсфельда, ключ – числовая последовательность.

8. Модуль для расшифровывания текста по алгоритму Гронсфельда, ключ – числовая последовательность.

Содержание отчёта

1. Цель работы.
2. Задание.
3. Анализ задания.
4. Алгоритм преобразования.
5. Программа на алгоритмическом языке.
6. Тестовые запуски и статистический анализ.
7. Выводы по работе.

Контрольные вопросы

1. Какой шифр будет реализовывать алгоритм Виженера при использовании ключа, состоящего из одного символа?
2. В чём состоит принципиальная разница моноалфавитных и полиалфавитных шифров замены?
3. Возможно ли применение статистических методов криптоанализа к полиалфавитным шифрам?

Лабораторная работа № 2

ШИФРОВАНИЕ МЕТОДОМ ПЕРЕСТАНОВКИ

Цель работы

Изучение принципов построения шифров перестановки. Исследование свойств перестановочных шифров.

Теоретическая часть

Метод заключается в том, что символы шифруемого текста переставляются по определенным правилам внутри шифруемого блока, при этом сами символы не изменяются.

Самая простая перестановка – написать исходный текст задом наперед и одновременно разбить шифрограмму на пятерки букв. Например:

исходный текст: пусть будет так, как мы хотели

подготовленный текст: пусть будет такка кмыхо тели

зашифрованный текст: илето хымка ккатт едубь тсуп

В последней группе (пятерке) не хватает одной буквы. Значит, прежде чем зашифровать исходное выражение, следует его дополнить незначащей буквой, например О, до числа, кратного пяти:

пусть будет такка кмыхо телио

Тогда шифрограмма будет выглядеть следующим образом:

оилет охымк аккат тедуб ьтсуп

Другой метод заключается в том, что исходный текст записывали в несколько строк, например по пятнадцать букв в каждой (с заполнением последней строки незначащими буквами):

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
п	у	с	т	ь	б	у	д	е	т	т	а	к	к	а
к	м	ы	х	о	т	е	л	и	к	л	м	н	о	п

После этого вертикальные столбцы по порядку пишутся в строку с разбивкой на пятерки букв:

пкумс ытхьо бтуед леитк тламк нкоап

Если строки укоротить, а количество строк увеличить, то получится *прямоугольник – решетка,* в которую записывается исходный текст.

Например:

1	2	3	4	5	6
п	у	с	т	ь	б
у	д	е	т	т	а
к	к	а	к	м	ы
х	о	т	е	л	и
а	б	в	г	д	е
м	л	к	и	з	ж

Если шифровать по диагоналям сверху вниз с левого верхнего угла, то получим:

п уу сдк текх ьтаоа бтктбм амевл ылгк иди ез ж

Третий вид данного шифра: перестановки с ключом. Необходимо знать ключ, например «радиатор». В соответствии с расположением букв в алфавите буква А получает номер 1, вторая буква А – 2, следующая по алфавиту буква Д – 3, потом И – 4, О – 5, первая буква Р – 6, вторая Р – 7 и буква Т – 8.

В результате получается:

Р	А	Д	И	А	Т	О	Р
6	1	3	4	2	8	5	7
п	у	с	т	ь	б	у	д
е	т	т	а	к	к	а	к
м	ы	х	о	т	е	л	и
о							

Записывая столбики в соответствии с номерами букв ключа, получим:
уты ькт стх тао уал пемо дки бке

Модификацией последнего метода является использование двух ключей: одного – для перестановки столбцов, а другого – для перестановки строк. Такой метод называется двойной перестановкой.

Ещё одним методом перестановки является использование решёток Кардано (шифр «Поворотная решётка»).

Ход работы

Получите индивидуальное задание у преподавателя. Реализуйте программный модуль в соответствии с полученным заданием. При реализации необходимо учесть следующие моменты:

1) предусмотреть возможность задания пользователем параметров шифра;

2) предусмотреть визуализацию всех пользовательских настроек;

3) для удобства тестирования и взаимодействия с другими модулями реализовать файловый ввод исходных данных и файловый вывод результата криптографического преобразования.

После реализации программного модуля выполните статистический анализ текста по биграммам до криптографического преобразования и после него.

Варианты заданий

1. Модуль для шифрования текста с помощью решёток Кардано.
2. Модуль для расшифровывания текста с помощью решёток Кардано.
3. Модуль для шифрования текста по алгоритму двойной перестановки.
4. Модуль для расшифровывания текста по алгоритму двойной перестановки.

Содержание отчёта

1. Цель работы.
2. Задание.
3. Анализ задания.
4. Алгоритм преобразования.
5. Программа на алгоритмическом языке.
6. Тестовые запуски и статистический анализ.
7. Выводы по работе.

Контрольные вопросы

1. Оцените количество возможных простых перестановок текста, состоящего из пяти символов? Из десяти символов? Из n символов?
2. Чем принципиально отличаются шифры перестановки от шифров замены?
3. Возможно ли применение статистических методов криптоанализа к перестановочным шифрам?

Лабораторная работа № 3

ИССЛЕДОВАНИЕ ГАММИРОВАНИЯ ПРИ ШИФРОВАНИИ

Цель работы

Исследование методов генерации псевдослучайных последовательностей. Исследование гаммирования при шифровании данных.

Теоретическая часть

Гаммирование – это процесс наложения гаммы шифра на открытые данные по определенному закону.

Гамма шифра – псевдослучайная последовательность чисел, вырабатываемая по заданному алгоритму для зашифровывания открытых данных и расшифровывания зашифрованных данных.

Суть метода заключается в том, что символы шифруемого текста последовательно складываются с символами некоторой специальной последовательности, которая называется *гаммой*.

Процедуру наложения гаммы можно реализовать двумя способами.

1. Гаммирование по модулю *K*. Символы исходного текста и гаммы заменяются эквивалентными цифрами, которые затем складываются по модулю ***K***, где ***K*** – число символов в алфавите, т.е.:

$R_i = (S_i + G) \bmod (\boldsymbol{K} - 1)$,

где R_i, S_i, G – символы зашифрованного текста, исходного текста и гаммы соответственно.

2. Двоичное гаммирование. Символы исходного текста и гаммы представляются в виде двоичного кода, затем соответствующие разряды складываются по модулю 2. Вместо сложения по модулю 2 при гаммировании можно использовать другие логические функции, необходимым требованием к которым является свойство обратимости преобразования.

Пример шифрования двоичным гаммированием представлен в таблице:

Шифруемый текст	Б	У	Д	Ь	…
	010010	100000	110010	100100	
Знаки гаммы	7	1	8	2	…
	000111	000001	001000	000010	
Шифрованный текст	010101	100001	111010	100110	

Ход работы

Получите индивидуальное задание у преподавателя. Выберите метод получения гаммы шифра (псевдослучайной последовательности чисел). Реализуйте программный модуль в соответствии с полученным заданием. При реализации необходимо учесть следующие моменты:

1) предусмотреть возможность задания пользователем гаммы шифра;

2) предусмотреть визуализацию всех пользовательских настроек;

3) для удобства тестирования и взаимодействия с другими модулями реализовать файловый ввод исходных данных и файловый вывод результата криптографического преобразования.

После реализации программного модуля выполните статистический анализ текста до криптографического преобразования и после него.

Варианты заданий

1. Модуль для шифрования текста гаммированием по модулю.
2. Модуль для расшифровывания текста гаммированием по модулю.
3. Модуль для шифрования текста двоичным гаммированием.
4. Модуль для расшифровывания текста двоичным гаммированием.

Содержание отчёта

1. Цель работы.
2. Задание.
3. Анализ задания.
4. Алгоритм преобразования.
5. Программа на алгоритмическом языке.
6. Тестовые запуски и статистический анализ.
7. Выводы по работе.

Контрольные вопросы

1. Гаммирование: основные определения.
2. Алгоритм шифрования текста методом гаммирования
3. Двоичное гаммирование: основные особенности.

Лабораторная работа № 4

ШИФРОВАНИЕ С ПОМОЩЬЮ АНАЛИТИЧЕСКИХ ПРЕОБРАЗОВАНИЙ

Цель работы

Исследование шифров, основанных на аналитических преобразованиях.

Теоретическая часть

Достаточно надежное закрытие информации может быть обеспечено при использовании для шифрования некоторых аналитических преобразований. Например, умножение матрицы на вектор по правилу:

$$\overline{C} = A \times \overline{B}; \qquad \sum_{j=1}^{N} a_{ij} b_j .$$

Если матрицу $A = (a_{ij})$ использовать в качестве ключа, а вместо компонента вектора $B = (b_j)$ подставить символы текста, то компоненты вектора $C = (c_j)$ будут представлять собой символы зашифрованного текста.

Пример.

Возьмем в качестве ключа квадратную матрицу третьего порядка

$$A = \begin{pmatrix} 14 & 8 & 3 \\ 8 & 5 & 2 \\ 3 & 2 & 1 \end{pmatrix}$$

Заменим буквы алфавита цифрами, соответствующими их порядковому номеру в алфавите: А–0, Б–1, В–2 и т.д. Тогда отрывку текста ВАТАЛА будет соответствовать 2, 0, 19, 0, 12, 0. По принятому алгоритму шифрования необходимо выполнить следующие действия:

$$\overline{C} = A \times \overline{B} = \begin{pmatrix} 14 & 8 & 3 \\ 8 & 5 & 2 \\ 3 & 2 & 1 \end{pmatrix} \times \begin{pmatrix} 2 \\ 0 \\ 19 \end{pmatrix} = \begin{pmatrix} 85 \\ 54 \\ 25 \end{pmatrix};$$

$$\overline{C} = A \times \overline{B} = \begin{pmatrix} 14 & 8 & 3 \\ 8 & 5 & 2 \\ 3 & 2 & 1 \end{pmatrix} \times \begin{pmatrix} 0 \\ 12 \\ 0 \end{pmatrix} = \begin{pmatrix} 96 \\ 60 \\ 24 \end{pmatrix}.$$

При этом шифрованный текст будет иметь вид: 85, 54, 25, 96, 60, 24.

Дешифрование осуществляется с использованием указанного правила умножения матрицы на вектор, только в качестве ключа берется матрица, обратная той, с помощью которой проводится зашифровывание, а в качестве вектора-сомножителя – соответствующие фрагменты символов закрытого текста; тогда значениями вектора-результата будут цифровые эквиваленты знаков открытого текста.

Матрицей, обратной данной $\boldsymbol{A}$, называется матрица $\boldsymbol{A}^{-1}$, получающаяся из присоединения матрицы делением всех ее элементов на определитель данной матрицы. Присоединенной называется матрица, составленная из алгебраических дополнений $\boldsymbol{A}_{ij}$, к элементам данной матрицы, которые вычисляются по формуле

$$A_{ij} = (-1)^{i+j} \Delta_{ij},$$

где Δ_{ij} – определитель матрицы, получаемой вычеркиванием *i*-й строки и *j*-го столбца исходной матрицы.

Определителем матрицы называется алгебраическая сумма *n!* членов (для определителя *n*-го порядка), составленная следующим образом: членами служат всевозможные произведения *n* элементов матрицы, взятых по одному в каждой строке и в каждом столбце; причем член суммы берется со знаком «+», если его индексы составляют четную подстановку, и со знаком «–» – в противоположном случае. Для матрицы третьего порядка определитель вычисляется следующим образом:

$$\Delta = a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{11}a_{23}a_{32} - a_{12}a_{21}a_{33} - a_{13}a_{22}a_{31}.$$

Процесс раскрытия выглядит так:

$$A^{-1} \times \overline{C} = \begin{pmatrix} 1 & -2 & 1 \\ -2 & 5 & -4 \\ 1 & -4 & 6 \end{pmatrix} \times \begin{pmatrix} 85 \\ 54 \\ 25 \end{pmatrix} = \begin{pmatrix} 1 \cdot 85 - 2 \cdot 54 + 1 \cdot 25 \\ -2 \cdot 85 + 5 \cdot 54 - 4 \cdot 25 \\ 1 \cdot 85 - 4 \cdot 54 + 6 \cdot 25 \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \\ 19 \end{pmatrix};$$

$$A^{-1} \times \overline{C} = \begin{pmatrix} 1 & -2 & 1 \\ -2 & 5 & -4 \\ 1 & -4 & 6 \end{pmatrix} \times \begin{pmatrix} 96 \\ 60 \\ 24 \end{pmatrix} = \begin{pmatrix} 1 \cdot 96 - 2 \cdot 60 + 1 \cdot 24 \\ -2 \cdot 96 + 5 \cdot 60 - 4 \cdot 24 \\ 1 \cdot 96 - 4 \cdot 60 + 6 \cdot 24 \end{pmatrix} = \begin{pmatrix} 0 \\ 12 \\ 0 \end{pmatrix}.$$

Таким образом, получена последовательность знаков раскрытого текста 2, 0, 19, 0, 12, 0, что соответствует исходному тексту.

Ход работы

Получите индивидуальное задание у преподавателя.

Реализуйте программный модуль в соответствии с полученным заданием.

После реализации программного модуля выполните статистический анализ текста до криптографического преобразования и после него.

Варианты заданий

1. Модуль для посимвольного шифрования текста.
2. Модуль для посимвольного расшифровывания текста.
3. Модуль для шифрования текста биграммами.
4. Модуль для расшифровывания текста биграммами.

Содержание отчёта

1. Цель работы.
2. Задание.
3. Анализ задания.
4. Алгоритм преобразования.
5. Программа на алгоритмическом языке.
6. Тестовые запуски и статистический анализ.
7. Выводы по работе.

Контрольные вопросы

1. В чем особенности метода аналитических преобразований.
2. Отличия метода посимвольного шифрования и шифрования текста биграммами.

3. Статистический анализ выполненного задания (минимум три примера).
4. Алгоритм выполнения поставленной задачи.

Лабораторная работа № 5

РАЗРАБОТКА И ИССЛЕДОВАНИЕ КРИПТОАЛГОРИТМА НА ОСНОВЕ ИСПОЛЬЗОВАНИЯ СКРЕМБЛЕРА

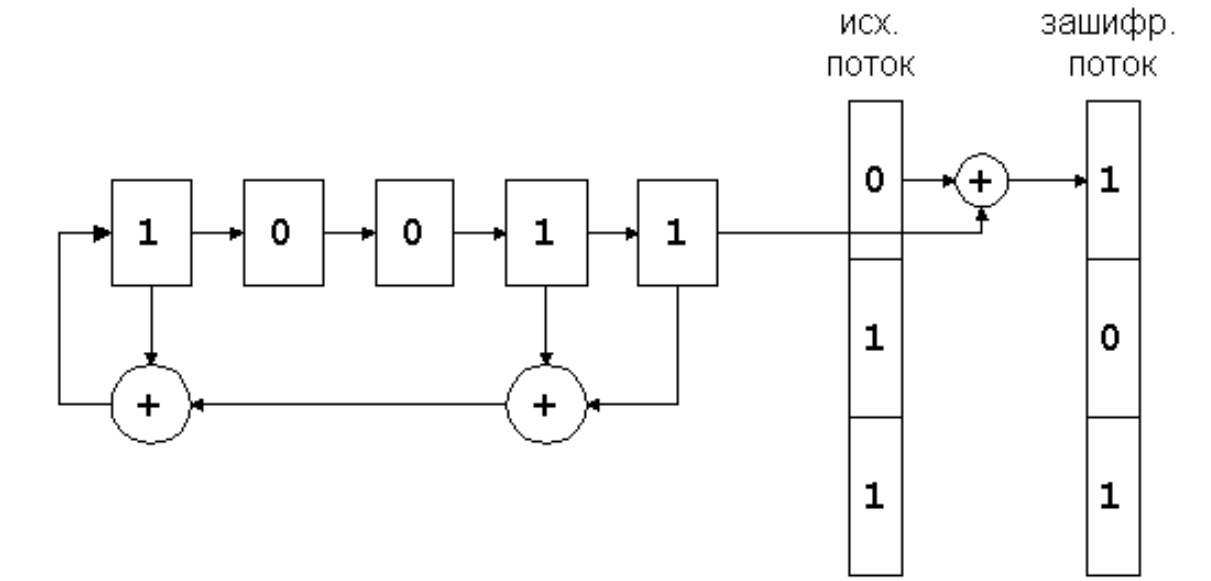
Цель работы

Познакомиться с простейшими методами потокового шифрования с использованием скремблеров.

Теоретическая часть

Суть скремблирования заключается в побитном изменении проходящих через вычислительную систему потока данных. Практически единственной операцией, используемой в скремблерах, является XOR – "побитное исключающее ИЛИ". Параллельно прохождению информационного потока в скремблере по определенному правилу генерируется поток бит – кодирующий поток. Как прямое, так и обратное шифрование осуществляется наложением кодирующей последовательности на исходную с использованием операции XOR.

Генерация кодирующей последовательности бит производится циклически из небольшого начального объема информации – ключа – по следующему алгоритму. Из текущего набора бит выбираются значения определенных разрядов и складываются с помощью операции XOR между собой. Все разряды сдвигаются на 1 бит, а только что полученное значение ("0" или "1") помещается в освободившийся самый младший разряд. Значение, находившееся в самом старшем разряде до сдвига, добавляется в кодирующую последовательность, становясь очередным ее битом (рисунок).



Ход работы

Получите индивидуальное задание у преподавателя. Реализуйте программный модуль в соответствии с полученным заданием. При реализации необходимо учесть следующие моменты:

1) кодирование строки осуществляется последовательным применением скремблера к каждому биту каждого ее символа;

2) определить период повторения последовательности, генерируемой заданным скремблером, для нескольких начальных значений ключа, выбранных случайным образом в диапазоне (0 – 255).

Варианты заданий

1. $x^8 + x^7 + x^6 + x^3 + x^2 + 1$
2. $x^9 + x^3 + 1$
3. $x^{10} + x^5 + x^4 + x^2 + 1$
4. $x^5 + x^4 + x^2 + 1$
5. $x^{11} + x^5 + x^2 + 1$
6. $x^{12} + x^7 + x^3 + x + 1$
7. $x^8 + x^6 + x^2 + 1$

Содержание отчёта

1. Цель работы.
2. Задание.
3. Анализ задания.
4. Алгоритм преобразования.
5. Программа на алгоритмическом языке.
6. Тестовые запуски и статистический анализ.
7. Выводы по работе.

Контрольные вопросы

1. Назовите преимущества и недостатки использования скремблера.
2. Укажите свойства, которыми должна обладать псевдослучайная последовательность, генерируемая скремблером.
3. Для каких целей используют скремблеры и дескремблеры?

Лабораторная работа 6. Исследование криптоалгоритма шифрования RSA

1. Цель работы.

Исследование структуры алгоритма и методики практической реализации криптосистемы шифрования RSA.

2. Основные теоретические положения

Как известно, алгоритмы симметричного шифрования используют ключи относительно небольшой длины и поэтому могут быстро шифровать большие объёмы данных.

При использовании алгоритма симметричного шифрования отправитель и получатель применяют для шифрования и расшифрования данных один и тот же секретный ключ. Таким образом, алгоритмы симметричного шифрования основываются на предположении о том, что зашифрованное сообщение не сможет прочитать никто, кроме того кто обладает ключом для его расшифрования. При этом если ключ не скомпрометирован, то при расшифровании автоматически выполняется аутентификация отправителя, т.к. только он имеет ключ, с помощью которого можно зашифровать сообщение. Таким образом, для симметричных криптосистем актуальна проблема безопасного распределения симметричных секретных ключей. В связи с этим без эффективной

организации защищённого распределения ключей использование обычной системы симметричного шифрования в вычислительных сетях практически невозможно.

Решением данной проблемы является использование асимметричных алгоритмов шифрования, называемых криптосистемами с открытым ключом. В них для зашифрования данных используется один ключ, называемый «открытым» а для расшифрования – другой называемый «закрытым или секретным». Следует иметь в виду, что ключ расшифрования не может быть определён из ключа зашифрования.

В асимметричных криптосистемах отрытый ключ и криптограмма могут быть отправлены по незащищённым каналам. Концепция таких систем основана на применении однонаправленных функций.

В качестве примера однонаправленной функции может служить целочисленное умножение. Прямая задача – вычисление произведения двух больших целых чисел p и q, $n = p*q$. Это относительно несложная задача для ЭВМ.

Обратная задача – факторизация или разложение на множители большого целого числа практически неразрешима при достаточно больших значениях n.

Например, если $p \approx q$, а их произведение $n \approx 2^{664}$, то для разложения этого числа на множители потребуется 2^{23} операций, что практически невозможно выполнить за приемлемое время на современных ЭВМ.

Другим примером однонаправленной функции является модульная экспонента с фиксированным основанием и модулем.

Например, если $y = a^x$, то естественно можно записать, что $x = \log_a (y)$.

Задача дискретного логарифмирования формулируется следующим образом. Для известных целых a, n, y следует найти такое число x, при котором $a^x \pmod n = y$. Например, если $a = 2^{664}$ и $n=2^{664}$ нахождение показателя степени x для известного y потребует около 10^{26} операций, что также невозможно выполнить на современных ЭВМ .

В связи с тем, что в настоящее время не удалось доказать, что не существует эффективного алгоритма вычисления дискретного логарифма за приемлемое время, то модульная экспонента также условно отнесена к однонаправленным функциям.

Другим важным классом функций, используемых при построении криптосистем с открытым ключом являются, так называемые, однонаправленные функции с секретом. Функция относится к данному классу при условии, что она является однонаправленной и, кроме того, возможно эффективное вычисление обратной функции, если известен секрет.

В данной лабораторной работе исследуется криптосистема RSA, использующая модульную экспоненту с фиксированным модулем и показателем степени (т.е. однонаправленную функцию с секретом).

3. Методика выполнения работы

Задание на выполнение лабораторной работы выдаётся преподавателем после прохождения студентами собеседования по основам криптосистем с открытым ключом.

Порядок выполнения работы соответствует, приведённой ниже, криптосистеме шифрования данных по схеме RSA.

Схема алгоритма шифрования данных RSA

3.1. Определение открытого «e» и секретного «d» ключей

3.1.1.Выбор двух взаимно простых больших чисел p и q

3.1.2. Определение их произведения: $n = p * q$

3.1.3. Определение функции Эйлера: $\varphi(n) = (p-1)(q-1)$

3.1.4. Выбор открытого ключа e с учётом условий:

$$1 < e \leq \varphi(n), \quad НОД(e, \varphi(n)) = 1$$

3.1.5. Определение секретного ключа ***d***, удовлетворяющего условию

$$e * d \equiv 1 \pmod{\varphi(n)}, \quad \text{где} \quad d < n$$

3.2. Алгоритм шифрования сообщения M (действия отправителя)

3.2.1. Разбивает исходный текст сообщения на блоки $M_1, M_2, \ldots, M_n$

$$(M_i = 0, 1, 2, \ldots, n)$$

3.2.2. Шифрует текст сообщения в виде последовательности блоков:

$$C_i = M_i^e \pmod{n}$$

3.2.3. Отправляет получателю криптограмму : $C_1, C_2, \ldots C_n$

3.2.3. Получатель расшифровывает криптограмму с помощью секретного ключа ***d*** по формуле: $M_i = C_i^d \pmod{n}$

3.3. Процедуру шифрования данных рассмотрим на следующем примере (для простоты и удобства расчётов в данном примере использованы числа малой разрядности):

3.3.1. Выбираем два простых числа ***p*** и ***q***, $p = 3, \quad q = 11;$

3.3.2. Определяем их произведение (модуль) $n = p*q = 33;$

3.3.3. Вычисляем значение функции Эйлера $\varphi(n) = (p-1)(q-1)$

$$\varphi(n) = 2*10 = 20$$

3.3.4. Выбираем случайным образом открытый ключ с учётом выполнения условий $1 < e \leq \varphi(n)$ и $НОД(e, \varphi(n)) = 1, \ e = 7;$

3.3.5. Вычисляем значение секретного ключа d, удовлетворяющего условию

$$e*d \equiv 1 \pmod{\varphi(n)}, \quad 7*d \equiv 1 \pmod{20}; \ d = 3;$$

3.3.6. Отправляем получателю пару чисел $(n = 33, e = 7)$;

Представляем шифруемое сообщение ***M*** как последовательность целых чисел ***312.***

3.3.7. Разбиваем исходное сообщение на блоки $M_1 = 3, M_2 = 1, M_3 = 2;$

3.3.8. Шифруем текст сообщения, представленный в виде последовательности блоков: $C_i = M_i^e \pmod{n}$

$$C_1 = 3^7 \ (mod\ 33) = 2187\ (mod\ 33) = 9,$$

$$C_2 = 1^7 \ (mod\ 33) = 1\ (mod\ 33) = 1,$$

$$C_3 = 2^7 \ (mod\ 33) = 128\ (mod\ 33) = 29.$$

3.3.9. Отправляем криптограмму $C_1 = 9, C_2 = 1, C_3 = 29$.

3.3.10. Получатель расшифровывает криптограмму с помощью секретного ключа d по формуле: $M_i = C_i^d \ (mod\ n)$

$$M_1 = 9^3 \ (mod\ 33) = 729\ (mod\ 33) = 3$$

$$M_2 = 1^3 \ (mod\ 33) = 1\ (mod\ 33) = 1$$

$$M_3 = 29^3 \ (mod\ 33) = 24389\ (mod\ 33) = 2.$$

Полученная последовательность чисел ***312*** представляет собой исходное сообщение ***M***.

4. Содержание отчёта

4.3. Составить блок-схему и программу алгоритма шифрования RSA.

4.4. Листинг программы шифрования заданного сообщения ***M*** с использованием алгоритма RSA.

4.5. Выводы: преимущества и недостатки алгоритма шифрования RSA.

Литература: [1], [2],[3].

Лабораторная работа 7. Исследование электронной цифровой подписи (ЭЦП) RSA

1. Цель работы

Исследование структуры алгоритма и методики практической реализации (ЭЦП) RSA.

2.Основные теоретические положения

Технология применения системы ЭЦП предполагает наличие сети абонентов, обменивающихся подписанными электронными документами. При обмене электронными документами по сети значительно снижаются затраты, связанные с их обработкой, хранением и поиском.

Одновременно при этом возникает проблема, как аутентификации автора электронного документа, так и самого документа, т.е. установление подлинности автора и отсутствия изменений в полученном электронном сообщении.

В алгоритмах ЭЦП как и в асимметричных системах шифрования используются однонаправленные функции. ЭЦП используется для аутентификации текстов, передаваемых по телекоммуникационным каналам.

ЭЦП представляет собой относительно небольшой объём дополнительной цифровой информации, передаваемой вместе с подписанным текстом.

Концепция формирования ЭЦП основана на обратимости асимметричных шифров, а также на взаимосвязанности содержимого сообщения, самой подписи и пары ключей. Изменение хотя бы одного из этих элементов сделает невозможным подтверждение подлинности подписи, которая реализуется при помощи асимметричных алгоритмов шифрования и хэш-функций.

Система ЭЦП включает две процедуры:

- формирование цифровой подписи;
- проверку цифровой подписи.

В процедуре формирования подписи используется секретный ключ отправителя сообщения, в процедуре проверки подписи – открытый ключ отправителя.

Безопасность системы RSA определяется вычислительной трудностью разложения на множители больших целых чисел. Недостатком алгоритма цифровой подписи RSA является уязвимость её к мультипликативной атаке. Другими словами, алгоритм ЭЦП RSA позволяет хакеру без знания секретного ключа сформировать подписи под теми документами, в которых результат хэширования можно вычислить как произведение результата хэширования уже подписанных документов.

Обобщённая схема формирования и проверки электронной цифровой подписи приведена на рис.1.

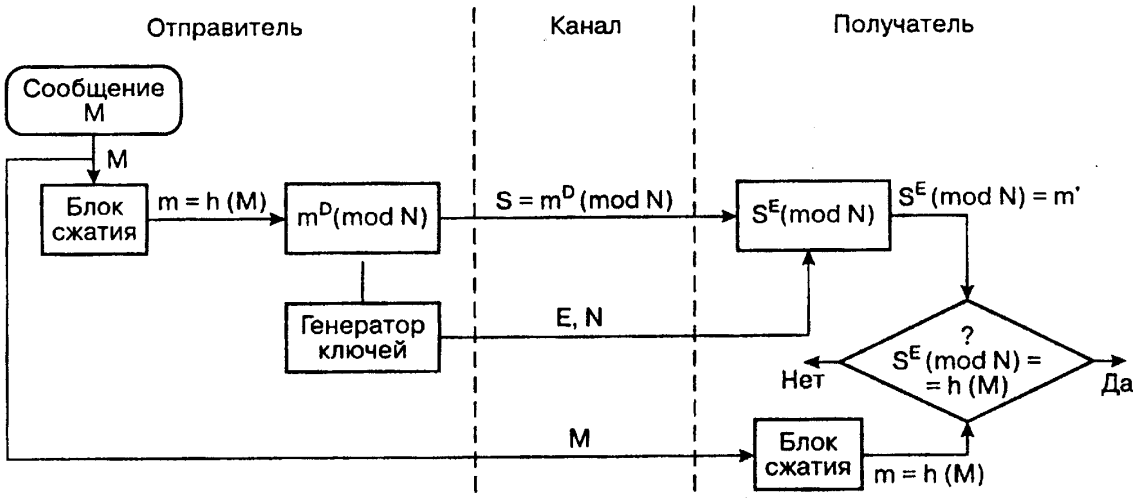


Рис. 1. Схема электронной цифровой подписи RSA

3. Методика выполнения работы

Алгоритм электронной цифровой подписи (ЭЦП) RSA

3.1.Определение открытого « *e*» и секретного « *d*» ключей (действия отправителя)

3.1.1. Выбор двух взаимно простых больших чисел p и q

3.1.2. Определение их произведения $n = p \cdot q$

3.1.3. Определение функции Эйлера: $\varphi(n) = (p-1)(q-1)$

3.1.4. Выбор секретного ключа d с учетом условий: $1 < d \leq \varphi(n)$, $НОД(d, \varphi(n)) = 1$

3.1.5. Определение значения открытого ключа e: $e < n$, $e \cdot d \equiv 1 (mod\ \varphi(n))$

3.2. Формирование ЭЦП

3.2.1. Вычисление хэш-значения сообщения M: $m = h(M)$

3.2.2. Для получения ЭЦП шифруем хэш-значение m с помощью секретного ключа d и отправляем получателю цифровую подпись $S = m^d (mod\ n)$ и открытый текст сообщения M

3.3.Аутентификация сообщения - проверка подлинности подписи

3.3.1. Расшифровка цифровой подписи S с помощью открытого ключа e и вычисление её хэш-значения $m' = S^e (mod\ n)$

3.3.2. Вычисление хэш-значения принятого открытого текста M $m = h(M)$

3.3.3. Сравнение хэш-значений m и m', если $m = m'$, то цифровая подпись S – достоверна.

Задание на выполнение лабораторной работы выдаётся преподавателем после прохождения студентами собеседования по основам аутентификации данных и концепции формирования электронной цифровой подписи.

Порядок выполнения работы соответствует, приведённому выше алгоритму формирования ЭЦП по схеме RSA.

Процедуру формирования ЭЦП сообщения ***M*** рассмотрим на следующем простом примере:

3.4. Вычисление хэш-значения сообщения *M:* $m = h(M)$.

Хешируемое сообщение ***M*** представим как последовательность целых чисел ***312***. В соответствии с приведённым выше алгоритмом формирования ЭЦП RSA выбираем два взаимно простых числа $p = 3$, $q = 11$, вычисляем значение $n = p*q = 3*11 = 33$, выбираем значение секретного ключа $d = 7$ и вычисляем значение открытого ключа $e = 3$. Вектор инициализации H_0 выбираем равным ***6*** (выбирается случайным образом).

Хэш-код сообщения $M = 312$ формируется следующим образом:

$$H_1 = (M_1 + H_0)^2 \pmod n = (3 + 6)^2 \pmod{33} = 81 \pmod{33} = 15;$$

$$H_2 = (M_2 + H_1)^2 \pmod n = (1 + 15)^2 \pmod{33} = 256 \pmod{33} = 25;$$

$$H_3 = (M_3 + H_2)^2 \pmod n = (2 + 25)^2 \pmod{33} = 729 \pmod{33} = 3;\ m = 3$$

3.4.1. Для получения ЭЦП шифруем хэш-значение m с помощью секретного ключа ***d*** и отправляем получателю цифровую подпись

$S = m^d \pmod n$ и открытый текст сообщения ***M***

$S = 3^7 \pmod{33} = 2187 \pmod{33} = 9$

3.4.2. Проверка подлинности ЭЦП

Расшифровка ***S*** (т. е. вычисление её хэш-значения m') производится с помощью открытого ключа e.

$$m' = S^e \pmod n = 9^3 \pmod{33} = 729 \pmod{33} = 3$$

3.4.3. Если сравнение хэш-значений m' и m показывает их равенство, т.е. $m = m'$, то подпись достоверна.

4.Содержание отчета

4.1. Составить блок-схему алгоритма и программу формирования ЭЦП RSA.
4.2. Листинг программы расчета ЭЦП RSA в соответствии с заданием
4.3.Выводы преимущества и недостатки ЭЦП RSA.

Литература: [2, 3, 4]

Лабораторная работа 8. Исследование криптоалгоритма шифрования Эль -Гамаля

1. Цель работы

Исследование структуры алгоритма и методики практической реализации криптосистемы шифрования Эль Гамаля.

2. Основные теоретические положения

Схема шифрования Эль Гамаля может быть использована как для формирования цифровых подписей, так и шифрования данных.

Безопасность схемы Эль Гамаля обусловлена сложностью вычисления дискретных логарифмов в конечном поле.

В настоящее время наиболее перспективными системами криптографической защиты являются системы с открытым ключом. В таких системах для шифрования сообщения используется закрытый ключ, а для расшифрования – открытый.

Открытый ключ не является секретным и может быть опубликован для использования всеми пользователями системы, которые зашифровывают данные. Расшифровывание данных с помощью открытого ключа невозможно.

Для расшифрования данных получатель зашифрованной информации использует секретный ключ, который не может быть определён из открытого ключа.

При использовании алгоритма шифрования Эль Гамаля длина шифротекста вдвое больше длины исходного открытого текста ***M***.

В реальных схемах шифрования необходимо использовать в качестве модуля n большое простое число, имеющее в двоичном представлении длину ***512… 1024 бит***.

Следует отметить, что формирование каждой подписи по данному методу требует нового значения ***k***, причём это значение должно выбираться случайным образом. Если нарушитель раскроет значение ***k***, повторно используемое отправителем, то может раскрыть и секретный ключ ***x*** отправителя.

Алгоритм шифрования данных по схеме Эль Гамаля приведён в разделе 3.

3. Методика выполнения работы

Задание на выполнение лабораторной работы выдаётся преподавателем после прохождения студентами собеседования по основам криптографической защиты информации.

Порядок выполнения работы соответствует приведённой ниже криптосистеме шифрования данных по схеме Эль Гамаля.

Схема алгоритма шифрования данных Эль Гамаля

3.1. Определение открытого "*y*" и секретного "*x*" ключей

3.1.1. Выбор двух взаимно простых больших чисел p и q, $q < p$

3.1.2. Выбор значения секретного ключа x, $x < p$

3.1.3. Определение значения открытого ключа y из выражения:

$$y = q^x \ (mod\ p)$$

3.2. Алгоритм шифрования сообщения M

3.2.1. Выбор случайного числа k, удовлетворяющего условию:

$$0 \le k < p-1 \text{ и } НОД\ (k,p-1) = 1$$

3.2.2. Определение значения a из выражения: $a = q^k (mod\ p)$

3.2.3. Определение значения b из выражения: $b = y^k M\ (mod\ p)$

3.2.4. Криптограмма C, состоящая из a и b, отправляется получателю

3.2.5. Получатель расшифровывает криптограмму с помощью выражения: $M a^x = b\ (mod\ p)$

3.3. Процедуру шифрования данных рассмотрим на следующем примере

(для удобства расчётов в данном примере использованы числа малой разрядности):

3.3.1. Выбираем два взаимно простых числа $p = 11$ и $q = 2$;

3.3.2. Выбираем значение секретного ключа x, $(x < p)$, $x = 8$;

3.3.3. Вычисляем значение открытого ключа у из выражения

$$y = q^x \ (mod\ p) = 2^8 \ (mod\ 11) = 256 \ (mod\ 11) = 3$$

3.3.4.Выбираем значение открытого сообщения $M = 5$;

3.3.5. Выбираем случайное число $k = 9$; $НОД\ (9, 10) = 1$;

3.3.6. Определяем значение а из выражения:

$$a = q^k \ (mod\ p) = 2^9 \ (mod\ 11) = 512 \ (mod\ 11) = 6;$$

3.3.7. Определяем значение b из выражения:

$$b = y^k M \pmod p = 3^9 * 5 \pmod{11} = 98415 \pmod{11} = 9.$$

Таким образом, получаем зашифрованное сообщение как $(a, b) = (6, 9)$ и отправляем получателю.

3.3.8. Получатель расшифровает данный шифротекст, используя секретный ключ x и решая следующее сравнение:

$$M * a^x \equiv b \pmod p = 5 * 6^8 \equiv 9 \pmod{11} = 8398080 \equiv 9 \pmod{11}$$

Вычисленное значение сообщения $M = 5$ представляет собой заданное исходное сообщение.

4. Содержание отчёта

4.1. Составить блок-схему и программу алгоритма шифрования Эль Гамаля.

4.2. Листинг программы шифрования заданного сообщения с использованием алгоритма Эль Гамаля.

4.3. Выводы.

Литература: [2],[3],[4].

Лабораторная работа 9. Исследование электронной цифровой подписи (ЭЦП) Эль Гамаля

1. Цель работы

Исследование структуры алгоритма и методики практической реализации (ЭЦП) Эль Гамаля.

2. Основные теоретические положения

Общепризнанные приёмы установления подлинности физической подписи под документом абсолютно не пригодны при обработке документов в электронной форме. Решением данного вопроса является алгоритм, так называемой, системы электронного подписывания документов. Для гарантии подлинности авторства и целостности информационного сообщения необходимо зашифровать его содержимое. При использовании цифровой подписи информация не шифруется и остаётся доступной любому пользователю, имеющему к ней доступ.

При обмене электронными документами по сети значительно снижаются затраты, связанные с их обработкой, хранением и поиском.

Одновременно при этом возникает проблема, как аутентификации автора электронного документа, так и самого документа, т.е. установление подлинности автора и отсутствия изменений в полученном электронном сообщении.

ЭЦП используется для аутентификации текстов, передаваемых по телекоммуникационным каналам. Функционально она аналогична обычной рукописной подписи и обладает основными её свойствами:

- удостоверяет, что подписанный текст исходит от лица, поставившего подпись;

- не даёт этому самому лицу возможности отказаться от обязательств, связанных с подписанным текстом;
- гарантирует целостность подписанного текста.

ЭЦП представляет собой относительно небольшой объём дополнительной цифровой информации, передаваемой вместе с подписанным текстом.

Концепция формирования ЭЦП по схеме Эль Гамаля также основана на обратимости асимметричных шифров и на взаимосвязанности содержимого сообщения, самой подписи и пары ключей.

Идея алгоритма цифровой подписи Эль Гамаля основана на том, что для обоснования практической невозможности фальсификации цифровой подписи в ней использована более сложная вычислительная задача дискретного логарифмирования, чем разложение на множители большого целого числа. Основным достоинством такой схемы цифровой подписи является возможность выработки ЭЦП для большого числа сообщений с использованием одного секретного ключа.

Безопасность схемы Эль Гамаля обусловлена сложностью вычисления дискретных логарифмов в конечном поле.

Описание схемы формирования ЭЦП Эль Гамаля представлено в разделе 3.

3. Методика выполнения работы

Задание на выполнение лабораторной работы выдаётся преподавателем после прохождения студентами собеседования по основам аутентификации данных и концепции формирования электронной цифровой подписи по схеме Эль Гамаля.

Схема формирования ЭЦП Эль Гамаля

3.1. Определение открытого “*y*” и секретного “*x*” ключей (действия отправителя)

3.1.1. Выбор двух взаимно простых больших чисел p и q, $q<p$

3.1.2. Выбор значения секретного ключа x, $x < p$

3.1.3. Определение значения открытого ключа y из выражения:

$$y = q^x \pmod p$$

3.2.Формирование ЭЦП

3.2.1. Вычисление хэш-значения сообщения M: $m = h(M)$

3.2.2. Выбор случайного числа k, $0 < k < p-1$ и $НОД\ (k, p-1) = 1$

3.2.3. Определение значения a из выражения: $a = q^k \pmod p$

3.2.4. Определение значения b из выражения:

$$m = (xa + kb) \pmod{(p-1)}$$

3.2.5. Цифровая подпись $S = (a, b)$ и открытый текст сообщения M отправляются получателю.

3.3. Аутентификация сообщения – проверка подлинности подписи (действия получателя)

3.3.1. Вычисление хэш-значения принятого открытого текста сообщения M

$$m' = h(M)$$

3.3.2. Подпись считается достоверной, если $a < p$, $m = m'$ и выполняется условие $y^a a^b \pmod p = q^{m'} \pmod p$

3.4. В качестве процедуры формирования ЭЦП рассмотрим следующий пример (для удобства расчётов в данном примере использованы числа малой разрядности):

3.4.1. Выбираем простое число ***p*** и два случайных числа ***q*** и ***x (q и x < p),***

p = 11, q = 2 и секретный ключ ***x = 8;***

3.4.2. Вычисляем значение открытого ключа **y**

$$y = q^{x} \pmod{p} = 2^{8} \pmod{11} = 3;$$

3.4.3. Определяем хэш-значение исходного сообщения ***M, (312)***

m = h (M), в данном примере принимаем ***m = 3*** (методика определения хэш-значения сообщения ***M*** приведена в работе 2).

3.4.4. Выбираем случайное целое число ***k,*** взаимно простое с ***p-1.***

Принимаем ***k = 9, НОД (9, 10) = 1.***

3.4.5. Для формирования ЭЦП вычисляем элементы подписи ***a и b***

$$a = q^{k} \pmod{p} = 2^{9} \pmod{11} = 6.$$

Элемент ***b*** определяем с помощью расширенного алгоритма Евклида из следующего соотношения:

$$m = (xa + kb) \pmod{(p-1)} ; 3 = (8*6 + 9*b) \pmod{10} = 9*b = -45 \pmod{10}$$

$$b = 5.$$

В данном примере цифровой подписью является пара чисел ***a = 6, b = 5.***

Цифровая подпись ***S = (a,b)*** и открытый текст сообщения ***M*** отправляются получателю. Для контроля целостности сообщения и достоверности ЭЦП получатель вычисляет хэш-значение m′ принятого открытого текста сообщения ***M***. При этом отправитель и получатель использует одну и ту же хэш-функцию ***h (·).***

Получив подписанное сообщение и открытый ключ ***y = 3***, получатель для проверки подлинности подписи проверяет выполнение условия

$$y^{a} a^{b} \pmod{p} = q^{m'} \pmod{p}$$

$$3^{6} * 6^{5} \pmod{11} = 2^{3} \pmod{11}$$

$$5668704 \pmod{11} = 8 \pmod{11}$$

$$8 \pmod{11} = 8 \pmod{11},$$

так как условие выполняется, то принятое получателем сообщение признаётся подлинным.

Таким образом, процедура установления подлинности принятого сообщения состоит в проверке соответствия аутентификатора сообщения.

Следует иметь ввиду, что каждая подпись по схеме Эль Гамаля требует нового значения $\boldsymbol{k}$. Случайное значение $\boldsymbol{k}$ должно храниться в секрете.

4. Содержание отчета

4.1. Составить блок-схему алгоритма и программу формирования ЭЦП Эль Гамаля на любом удобном для студента языке.

4.2. Листинг программы расчёта ЭЦП Эль Гамаля в соответствии с заданием.

4.3. Выводы: преимущества и недостатки ЭЦП Эль Гамаля.

Литература: [1],[2],[3]