Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ **ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ**

КАФЕДРА **КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.01 Информатика и вычислительная техника**

МАГИСТЕРСКАЯ ПРОГРАММА **09.04.01/12 Интеллектуальный анализ больших данных в системах поддержки принятия решений.**

О Т Ч Е Т

по лабораторной работе № 4

Вариант № 9

Название: Электронно-цифровая подпись и примеры хеширования

Дисциплина: Информационная безопасность автоматизированных систем

Студент ИУ6-31М (Группа) _____ (Подпись, дата) И.С. Марчук (И.О. Фамилия)

Преподаватель _____ (Подпись, дата) Д.А. Миков (И.О. Фамилия)

Москва, 2024

Цель: овладеть практическими навыками закрытия информации электронно-цифровой подписью и приемами хеширования, рассмотрение хеширования методом контрольных сумм и методом наложения кодов — гаммированием.

Задание: составить программу шифрования методом контрольных сумм и методом хеширования с применением гаммирования.

Условия варианта:

Пусть a = 23, b = 19, c = MaxVal + 1 = 256, t0 = 235. Вычислить контрольные суммы для нескольких сообщений методом контрольных сумм (KSumm) и методом хеширования с применением гаммирования (SummKodBukvOtkr):

а) P = '0000123456', KSumm = ?, SummKodBukvOtkr – ?;

б) P = '6543210000', KSumm = ?, SummKodBukvOtkr – ?;

в) P = '10000001', KSumm = ?, SummKodBukvOtkr – ?;

г) P = '11000000', KSumm = ?, SummKodBukvOtkr – ?

Ход работы

Я реализовал программу на языке Kotlin при помощи стандартных компонентов библиотеки Swing. Интерфейс программы состоит из 1 окна, на котором пользователь может ввести исходную строку, выбрать использовать ли гаммирование, а также ввести параметры хеширования. Исходный код программы представлен в листинге 1. Результаты для варианта получились следующими:

а) P = '0000123456', KSumm = 245, SummKodBukvOtkr – 82;

б) P = '6543210000', KSumm = 245, SummKodBukvOtkr – 84;

в) P = '10000001', KSumm = 130, SummKodBukvOtkr – 236;

г) P = '11000000', KSumm = 130, SummKodBukvOtkr – 236

Листинг программы 1 – Программа генерации пароля

```
package org.example

import java.lang.Exception
```

```
import javax.swing.*
import javax.swing.text.AttributeSet
import javax.swing.text.PlainDocument
fun main() {
    // создание окна
    val frame = JFrame("Вычисление контрольных сумм")
    frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE)
    frame.setSize(800, 400)
    frame.setLocationRelativeTo(null)
    val italicBox = frame.add(JCheckBox("Гаммирование").apply {
        setBounds(450, 25, 250, 20)
        //check(false)
    }) as JCheckBox
    // ================== Ввод констант ==================
    frame.add(JLabel("Строка = ").apply {
        setBounds(25, 25, 250, 20)
    })
    val strInput = frame.add(JTextField().apply {
        setBounds(90, 25, 200, 20)
        text = "0000123456"
    }) as JTextField
    frame.add(JLabel("a = ").apply {
        setBounds(25, 50, 250, 20)
    })
    val aInput = frame.add(JTextField().apply {
        document = JTextFiledNumbersOnly(4)
        setBounds(90, 50, 100, 20)
        text = "23"
    }) as JTextField
    frame.add(JLabel("b = ").apply {
        setBounds(25, 75, 250, 20)
    })
    val bInput = frame.add(JTextField().apply {
        document = JTextFiledNumbersOnly(4)
        setBounds(90, 75, 100, 20)
```

```
    text = "19"
}) as JTextField
frame.add(JLabel("maxVal = ").apply {
    setBounds(25, 100, 250, 20)
})
val maxValInput = frame.add(JTextField().apply {
    document = JTextFiledNumbersOnly(4)
    setBounds(90, 100, 100, 20)
    text = "255"
}) as JTextField
frame.add(JLabel("t0 = ").apply {
    setBounds(25, 125, 250, 20)
})
val t0Input = frame.add(JTextField().apply {
    document = JTextFiledNumbersOnly(4)
    setBounds(90, 125, 100, 20)
    text = "235"
}) as JTextField
val output = frame.add(JLabel("Нажмите сгенерировать!").apply {
    setBounds(25, 175, 250, 20)
}) as JLabel
frame.add(JButton("Сгенерировать хеш!").apply {
    setBounds(25, 325, 740, 20)
    addActionListener {
        try {
            output.text = "Хэш = " + kSumm(
                input = strInput.text.toString(),
                a = aInput.text.toString().toInt(),
                b = bInput.text.toString().toInt(),
                maxVal = maxValInput.text.toString().toInt(),
                t0 = t0Input.text.toString().toInt(),
                useGammir = italicBox.isSelected
            ).toString()
        } catch (e: Exception) {
            output.text = "Ошибка ввода"
```

```
            }
        }
    })
    frame.layout = null
    // переотрисовка
    frame.isVisible = true
}
private fun kSumm(input: String, a: Int, b: Int, maxVal: Int, t0: Int, useGammir:
Boolean): Int {
    // коды символов T1, T2...
    print("T = [")
    var prevTi = 0
    val tCodes = Array(input.length) { pos ->
        val value =
            if (pos == 0) t0 else (a * prevTi + b) % (maxVal + 1) // ti
        // текущее значение  как предыдущее
        prevTi = value
        print("$value, ")
        value
    }
    println("]")
    // сумма
    var summ = 0
    input.indices.forEach {
        if (useGammir) {
            // расчет суммы
            val number = (input[it].code xor tCodes[it])
            summ += number
            print("[${input[it].code} xor ${tCodes[it]} > $number] + ")
        } else {
            // расчет суммы
            summ += input[it].code
            print("${input[it].code} + ")
        }
    }
    println(" = $summ")
```

```
    // получаем контрольную сумму
    val kSumm = summ % (maxVal + 1)
    println("kSumm = $kSumm")

    return kSumm
}
class JTextFiledNumbersOnly(
    private val limit: Int
) : PlainDocument() {
    override fun insertString(offs: Int, str: String?, a: AttributeSet?) {
        if (str == null)
            return
        // проверка на символы
        val newInput = StringBuilder()
        str.forEach {
            if (it.code in 48..58) newInput.append(it)
        }
        // если не достигли максимальной длины строки
        if (length + newInput.length <= limit)
            super.insertString(offs, newInput.toString(), a)
    }
}
```

Пример работы программы

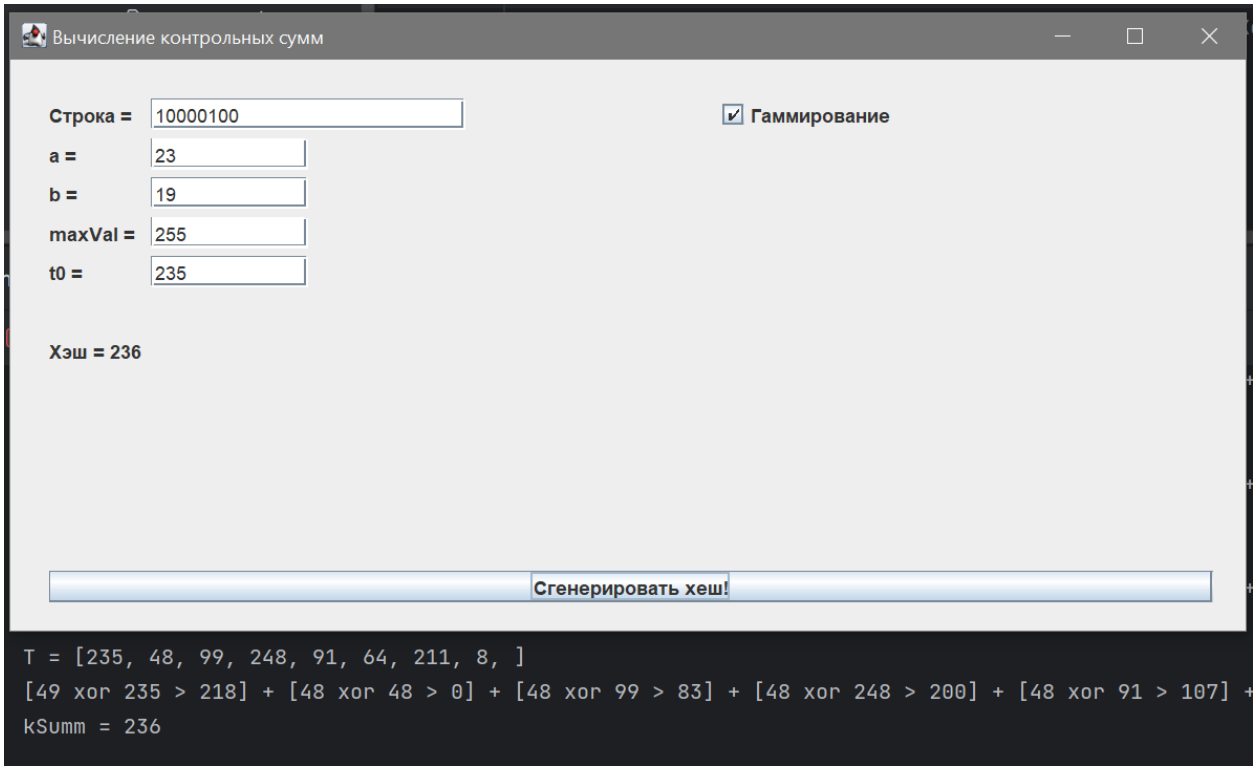

Рисунок 1 – Окно генерации хеш кода

Контрольные вопросы

1. Назвать три функции ЭЦП.

– функция авторизации — подтверждение того, что подписавшийся действительно является тем, за кого мы его принимаем;

– обеспечение того, что подписавшийся не может отказаться от документа, который он подписал;

– подтверждение того, что отправитель подписал именно тот документ, который отправил, а не какой-либо иной.

2. Перечислить этапы формирования ЭЦП.

1) Корреспондент X по специальному алгоритму обрабатывает документ, предназначенный для отправки адресату Y. В результате применения этого алгоритма, вырабатывается некоторый параметр, характеризующий документ в целом.

2) Затем X с помощью секретной части ключа шифрует полученный параметр. Полученный таким образом шифр является ЭЦП корреспондента X.

3) Корреспондент X отправляет адресату Y документ и свою электронную цифровую подпись.

4) Адресат Y реализует на полученном документе тот же алгоритм, которым пользовался корреспондент X.

5) Затем Y дешифрует электронную цифровую подпись, полученную от X, пользуясь открытой частью ключа, предоставленной ему корреспондентом X.

6) Окончательно адресат Y сравнивает значение параметра, полученного на четвертом этапе, с расшифрованным значением ЭЦП

3. Что шифруется при применении ЭЦП?

   Применение ЭЦП не предполагает обязательного засекречивания (шифрования) самого передаваемого документа. Шифруется только некоторая интегральная характеристика этого документа.

4. Что называется хеш-значением документа?

   Значение интегрального параметра называют хеш-значением документа

5. Что называется хеш-функцией?

   способ (алгоритм) получения хеш-значения

6. Что называется сворачиванием (хешированием)документа?

   Получение хеш-значения с помощью хеш-функции называют сворачиванием (хешированием) текста документа в более короткий текст (интегральный параметр)

7. В чем заключается метод контрольных сумм?

   Под контрольной суммой понимается некоторое значение, рассчитанное путем сложения всех чисел (кодов символов), соответствующих данному тексту. Если сумма всех таких чисел K превышает максимально допустимое значение (MaxVal), заданное

заранее, то величина контрольной суммы полагается равной остатку от деления полученной суммы на максимально возможное значение контрольной суммы, увеличенное на единицу. Таким образом, контрольную сумму можно записать в следующем виде:

$$K\text{Summ} = \begin{cases} \bar{K}, & \text{при } K \le \text{Max}\,Val; \\ K \bmod (\text{Max}\,Val + 1), & \text{при } K > \text{Max}\,Val. \end{cases}$$

8. Перечислить этапы метода хеширования с применением гаммирования.
   - Пусть каждому символу документа (открытого текста) соответствует восьмибитовое двоичное слово Xi. Таким образом, исходный документ представляется в виде последовательности восьмибитовых двоичных слов: X1, X2 , …, Xp.
   - Затем выработаем последовательность псевдослучайных чисел ti по рекуррентной формуле ti+1 = (a ∗ ti + b) mod c, где i = 0, 1, …, p – 1; a, b, t0 — заданные числа; p — количество символов в тексте.
   - При c = 2n. Если взять n = 8, то двоичные представления чисел ti не будут превышать восьми двоичных знаков.
   - Далее каждое число ti представим в виде восьмибитового двоичного слова. Получаем последовательность двоичных слов: T1, T2, ..., Tp.
   - Двоичные числа Xi и Ti, сложим поразрядно по модулю 2. Получим новую последовательность двоичных слов: Y1 = X1 ⊕ T1, Y2 = X2 ⊕ T2, ..., Yp = Xp ⊕ Tp.
   - Каждое двоичное слово, рассматриваемое как двоичное число, переведем в десятичную систему, при этом получим последовательность чисел: y1, y2, …, yp.

- Полученная последовательность целых чисел суммируется по модулю MaxVal + 1 (если п = 8, то MaxVal = 255).

9. Недостаток метода контрольных сумм.

   Недостаток метода контрольных сумм (в обоих вариантах) заключается в том, что хотя несовпадение значений этих сумм служит верным признаком того, что документ подвергся изменению, но равенство значений еще не дает гарантии, что информация осталась неизменной. Можно произвольным образом изменить порядок следования букв, цифр или слов и фраз в документе, при этом контрольная сумма сохранит прежнее значение.

Вывод

Я разработал программу шифрования методом контрольных сумм и методом хеширования с применением гаммирования. А также, овладел практическими навыками закрытия информации электронно-цифровой подписью и приемами хеширования, и рассмотрел хеширование методом контрольных сумм и методом наложения кодов — гаммированием.