Лабораторная работа №3

Шифрование гаммированием

Тазаева Анастасия Анатольевна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Ознакомиться с шифрованием гаммированием. Реализовать его.

2 Задание

Реализовать на языке программирования Julia:

1. Шифрование гаммированием.

3 Теоретическое введение

Гаммирование - это метод шифрования, при котором символы исходного текста складываются (или иным образом комбинируются) с символами некоторой случайной последовательности, называемой гаммой. *Криптографическая стойкость* данного метода обеспечивается за счёт уникальной, истинно случайной гаммы, длина которой не меньше длины шифруемого сообщения, при этом каждый символ исходного текста преобразуется с помощью соответствующего символа гаммы. Принцип обратимости процесса позволяет восстановить исходный текст путём вычитания той же самой гаммы из зашифрованного сообщения, что делает гаммирование одним из наиболее надежных методов симметричного шифрования при соблюдении всех требований к генерации и использованию гаммы. Основное преимущество метода заключается в том, что при правильном применении (одноразовость гаммы и её истинная случайность) он обеспечивает абсолютную криптостойкость, доказанную математически.

4 Выполнение лабораторной работы

4.1 Шифрование гаммированием

Написан программный код на языке Julia [1], реализующий маршрутное шифрование:

```
function gamma_encryption(text, gamma_code)
    # massiv ASCII-kodov kirillicy + simvolov
    alphabet = vcat(1040:1045, 1025, 1046:1071, 32:33, 44, 46, 63,
    # println(alphabet)
    # filtryem text, izvestnye simvoly ostautsya
    filtr = filter(x -> findfirst(isequal(Int(only(x))), alphabet)
    #razbivaem text na simvoly
    edited_text = Int.(only.(split(filtr,"")))
    n = length(edited_text)
    #massiv xranit poryadkovyi nomer kajdogo simvola v alfavite
    por_num = [findfirst(isequal(edited_text[i]), alphabet) for i i
    # propisnye -> zaglavnye
    for i in 1:n
        if por_num[i] > 38
            por_num[i] -= 38
        end
    end
```

```
    #rejem na kyso4ki gamma_code, preobrazyija ego v simvolnyi forma
    edited_code = [findfirst(isequal(i), alphabet) for i in Int.(on
    m = length(edited_code)
    temp = [alphabet[mod(por_num[i]+edited_code[mod(i-1,m)+1]-1,38)+
    result=""
    result*=join(Char.(temp))
    return result
end
end
```

Получен следующий результат выполнения программного кода (рис. 4.1).

```
gamma_encryption("ПРИКАЗ", "ГАММА")
```

```
"УСЦШБЛ"
```

Рисунок 4.1: Шифрование гаммированием. Пример отработки

5 Выводы

В ходе лабораторной работы были изучено и реализовано шифрование гаммированием.

Список литературы

[1] *Julia 1.10 Documentation*. Англ. 2024. URL: https://docs.julialang.org/en/v1/base/strings/.