РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ПРЕЗЕНТАЦИЯ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2

дисциплина: Математические основы защиты информации и информационной безопасности

Студент: Пиняева Анна Андреевна

Группа: НПИмд-01-24

МОСКВА

2025

Цель работы

Изучение и реализация маршрутного шифрования, шифрования с помощью решеток и таблицы Виженера.

Ход работы

1. Маршрутное шифрование.

Рис. 1 Реализация маршрутного шифрования:

[1]:
```
function column(text, password)
    text = replace(uppercase(text), " " => "")
    text_chars = collect(text)
    n = length(collect(password))
    if length(text_chars) % n != 0
        append!(text_chars, fill('X', n - length(text_chars) % n))
    end
    m = length(text_chars) ÷ n
    table = [text_chars[i*n+1:(i+1)*n] for i in 0:m-1]
    password_chars = collect(password)
    order = sortperm(password_chars)
    cipher_chars = Char[]
    for col in order
        for row in 1:m
            push!(cipher_chars, table[row][col])
        end
    end
    return join(cipher_chars)
end
```

[1]:
```
column (generic function with 1 method)
```

Рис. 2 Тестирование маршрутного шифрования:

```
[2]: text1="нельзя недооценивать противника"
     password1="пароль"
     encrypted1=
     column(text1, password1)
     println(encrypted1)
```

```
ЕЕНПНЗОАТАЬОВОКННЕЬВЛДИРИЯЦТИХ
```

2. Шифрование с помощью решеток.

Рис. 3 Реализация шифрования с помощью решеток:

[4]:
```
function fleissner(k, text, password)
    grid = zeros(Int, 2k, 2k)
    num = 1
    temp_grid = reshape(1:(4*k^2), k, 4k)
    for rot in 0:3
        view = temp_grid[1:k, rot*k+1:(rot+1)*k]
        grid[1:k, 1:k] = view
        grid = rotl90(grid)
    end

    holes = collect(1:k^2)

    text = replace(uppercase(text), " " => "")
    text_chars = collect(text)

    total_cells = 4*k^2
    if length(text_chars) < total_cells
        append!(text_chars, fill('X', total_cells - length(text_chars)))
    elseif length(text_chars) > total_cells
        text_chars = text_chars[1:total_cells]
    end

    table = fill(' ', 2k, 2k)
    current_grid = copy(grid)
    idx = 1

    for rotation in 0:3
        for pos in 1:(2k)^2
            if current_grid[pos] in holes
                table[pos] = text_chars[idx]
                idx += 1
            end
        end
        current_grid = rotl90(current_grid)
    end
```

```
    password_chars = collect(password)
    order = sortperm(password_chars)

    result_chars = Char[]
    for col in order
        for row in 1:2k
            push!(result_chars, table[row, col])
        end
    end

    return join(result_chars)
end

function rotl90(matrix)
    return permutedims(reverse(matrix, dims=1))
end
```

Рис. 4 Тестирование шифрования с помощью решеток:

[8]:
```
k=2
text2="договор подписали"
password2="шифр"
encrypted2=fleissner(k,text2,password2)
println(encrypted2)
```

```
ГОЛИРППИВООДДОСА
```

3. Таблица Виженера.

[12]:
```
function vigener(text, password; encrypt=true)
    text = replace(uppercase(text), " " => "")
    text_chars = collect(text)
    password = uppercase(password)
    password_chars = collect(password)
    result_chars=Char[]
    for (i,char) in enumerate(text_chars)
        kchar=password_chars[(i-1)%length(password_chars)+1]
        shift=encrypt ? 1 : -1

        if 'А'<=char<='Я' && 'А'<=kchar<='Я'
            new_char = 'А' +(Int(char)-Int('А')+shift*(Int(kchar)-Int('А')))%32
            push!(result_chars,Char(new_char))
        else
            push!(result_chars,char)
        end
    end
     return join(result_chars)
end
```

[12]:
```
vigener (generic function with 1 method)
```

Рис. 5 Реализация таблицы Виженера:

Рис. 6 Тестирование таблицы Виженера:

[14]:
```
text3="криптография серьезная наука"
password3="математика"
encrypted3=vigener(text3,password3, encrypt=true)
println(encrypted3)
```

```
ЦРЪФЮОХШКФФЯГКЬЬЧПЧАЛНТШЦА
```

Вывод: В ходе данной работы мной были изучены и реализованы методы маршрутного шифрования, шифрования с помощью решеток и таблицы Виженера. Написан программный код на языке Julia и протестирован.