Лабораторная работа №2

Шифры перестановки

Кубасов В.Ю.

Содержание

1 Цель работы

Реализовать более сложные шифры перестановок.

2 Задание

Реализовать: - Маршрутное шифрования - Шифрование с помощью решёток - Шифр Виженера

3 Теоретическое введение

Целью всех шифров перестановок является изменение порядка букв/слогов/слов в исходном тексте.

Чем “случайнее” будет сама перестановка, тем сложнее подобрать её при осознаном bruteforce ключа (например, по словарю).

Цель шифров замены - заменить буквы/слоги/слова и пр. на другие.

Со временем в шифры начали добавлять помимо необходимых символов различные “мусорные символы”, что делало шифр более устойчивее к частотному криптоанализу, но не к другим видам взлома. Так, например, Виженер [был взломан][1] засчёт уязвимости - повторения одного и того же ключа, примечателен тот факт, что о взломе Виженера не было открыто известно в момент его взлома. Скорее всего, это было связано с Королевскими делами и военными действиями, т.к. в то время Виженер был объективно сильным шифром, который считался “невзламываемым”. Он мог бы таким и остаться, например, при создании ключа длиной в длину сообщения со случайными символами. Однако, такой ключ проблематично запомнить, и, соответсвенно, использовать такой подход не будут.

Современные подходы к шифрованию используют подобный сценарий. Изменилось лишь количество шагов и сложность метода шифрования ввиду появления вычислительных приборов.

Расчёт всех шифров [заключается][2] в одном: мы считаем, что злоумышленник не может подобрать ключ в допустимое время и надеемся, что у него нет средств к уменьшению возможного спектра ключей.

4 Выполнение лабораторной работы

4.1 Маршрутное шифрования

Из предложенного маршрутного ключа реализовывать полный маршрут с m*n таблицей оказалось не обязательно.
Для начала получаем все необходимые вводные данные, корректируем их:

```
println("Введите пароль");
pass = lowercase(readline());
sortedPass = join(sort(collect(pass)));

numberOfColumn = [];
lengthOfPass = length(pass);

for i in 1:2:length(sortedPass) * 2
    push!(numberOfColumn, (Int64)((findfirst(sortedPass[i], pass) - 1) / 2 + 1));
end;

println("Строку к шифрованию (без пробелов и других символов кроме кириллицы)");
rawString = lowercase(readline());
```

Здесь же определяем порядок столбцов, которые необходимо загрузить в итоговую строку.

Далее объявляем строку, в которой будет храниться зашифрованное сообщение и дополняем строку до количества символов, кратного ключу:

```
encodedString = "";

numberOfRaws = ceil(length(rawString) / lengthOfPass);

while length(rawString) < (numberOfRaws * lengthOfPass)
    global rawString *= 'a'; # заглушка для количества символов
end;
```

Далее согласно порядку столбцов переписываем символы из начальной строки по индексам, где первый индекс в столбце - номер столбца, а все последующие - номер столбца + размер ключа, умноженный на 1, 2, 3 ...

```
for i in numberOfColumn
    current = i;
    while (current <= length(rawString))
        global encodedString *= rawString[2 * current - 1];
        current += lengthOfPass;
    end;
end;
```

Далее просто выведем на экран полученное сообщение

```
println(encodedString);
```

4.2 Шифр Виженера

Шифр Виженера - многоалфавитный шифр, в котором каждая буква смещается в зависимости от ключа.

```
const abcStart = codepoint('a');
const abcEnd = codepoint('я');
```

```
println("Введите пароль");
pass = lowercase(readline());

# Работает при условии только кириллицы, без других символов
println("Введите строку для шифрования");
rawString = lowercase(readline());
```

Здесь abcStart, abcEnd - начальный и конечный код для алфавита (коды букв А и Я).
Далее получаем пароль (который задаст смещение) и строку для шифрования.

```
encodedString = "";

while (length(pass) < length(rawString))
    global pass *= pass;
end;
```

Объявляем закодированную строку и дополняем размер ключа до размера строки к шифрованию.

```
for i in 1:2:2 * length(rawString)
    global encodedString *= (Char)(abcStart - 1 + mod(codepoint(rawString[i]) + codepo
end;
```

Далее для каждого символа исходной строки, с учётом смещения в зависимости от символа ключа получаем конечный символ (берём сумму кодов букв по модулю размера алфавита).
Выводим на экран.

```
println(encodedString);
```

5 Выводы

Реализовали: - Маршрутный шифр - Шифр Виженера

Список литературы

1. Бабаш А.В. и др. Расширение границ применения методов дешифрования шифра Виженера // Вопросы кибербезопасности. Акционерное общество «Научно-производственное объединение «Эшелон», 2019. № 5 (33). С. 42–50.
2. Rusetskaya I.A. Cryptographic meaning of the Voynich manuscript // ВЕСТНИК РГГУ. 2023. С. 93.