Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники»

Факультет информационных технологий и управления
Кафедра интеллектуальных информационных технологий
Дисциплина «Средства и методы защиты информации в интеллектуальных системах»

ОТЧЁТ
к лабораторной работе №1
на тему
«ГЕНЕРАЦИЯ ПАРОЛЕЙ»

БГУИР 6-05-0611-03 130

Выполнил студент группы 321701
СЕМЕНЯКО Владимир Дмитриевич

_____
(дата, подпись студента)

Проверил
САЛЬНИКОВ Даниил Андреевич

_____
(дата, подпись преподавателя)

Минск 2025

1 ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

а) Разработать программу, реализующую следующие функции:

– генерация строки с заданной пользователем длиной, состоящей из символов алфавита в соответствии с вариантом задания (использовать функции rand( ), srand( ) и инициализацию от таймера);

– поверка равномерности распределения символов путем визуализации частотного распределения;

– вычисление среднего времени подбора пароля, выбираемого из сгенерированной строки.

б) Построить график зависимости среднего времени подбора пароля от его длины.

в) Дать практические рекомендации по выбору пароля исходя из предположений об алфавите пароля; ценности информации, доступ к которой защищается с помощью этого пароля; производительности вычислительного средства атакующего и времени атаки.

Вариант алфавита для генерации пароля:

2) Латиница строчные и прописные.

2 ВЫПОЛНЕНИЕ РАБОТЫ

Программа реализована на языке Python и выполняет следующие функции: генерация множества паролей заданной длины, подсчёт частоты появления символов и построение гистограммы, расчёт среднего времени подбора паролей для разных длин, визуализация зависимости времени подбора от длины пароля.

Листинг 1 – Код программы

```
import random
import string
import time
import matplotlib.pyplot as plt

def generate_passwords(num_passwords: int, length: int) -> list[str]:
    alphabet = string.ascii_letters
    random.seed(time.time())
    return [
        ''.join(random.choice(alphabet) for _ in range(length))
        for _ in range(num_passwords)
    ]

def plot_frequency_distribution(passwords: list[str]):
    alphabet = string.ascii_letters
```

```
    freq = {ch: 0 for ch in alphabet}

    for pwd in passwords:
        for ch in pwd:
            freq[ch] += 1

    plt.figure(figsize=(12, 6))
    plt.bar(freq.keys(), freq.values())
    plt.titleЧастотное(” распределение символов по( множеству паролей)”)
    plt.xlabelСимвол(””)
    plt.ylabelЧастота(””)
    plt.show()

def average_bruteforce_time(length: int, rate: float = 1e9) -> float:
    N = len(string.ascii_letters)
    total = N ** length
    avg_time = total / (2 * rate)
    return avg_time

def plot_bruteforce_times(max_length: int, rate: float = 1e9):
    lengths = list(range(1, max_length + 1))
    times = [average_bruteforce_time(L, rate) for L in lengths]

    plt.figure(figsize=(10, 5))
    plt.plot(lengths, times, marker='o')
    plt.yscale(”log”)
    plt.titleСреднее(” время подбора пароля от его длины”)
    plt.xlabelДлина(” пароля”)
    plt.ylabelСреднее(” время сек()”)
    plt.grid(True, which=”both”, linestyle=”--”, linewidth=0.5)
    plt.show()

if __name__ == ”__main__”:
    num_passwords = 10000
    length = 8

    passwords = generate_passwords(num_passwords, length)
    print(fСгенерировано” {num_passwords} паролей длиной {length} символов”)

    plot_frequency_distribution(passwords)

    plot_bruteforce_times(12, rate=1e9)
```

При переборе в среднем пароль находится за половину пространства:

$$T = \frac{N^L}{2R}$$

Рисунок 1 – Формула для вычисления среднего времени подбора пароля

В ходе эксперимента было сгенерировано 10 000 паролей длиной 8 символов. На гистограмме видно, что распределение символов приближается к равномерному (каждая буква встречается примерно одинаковое количество раз).

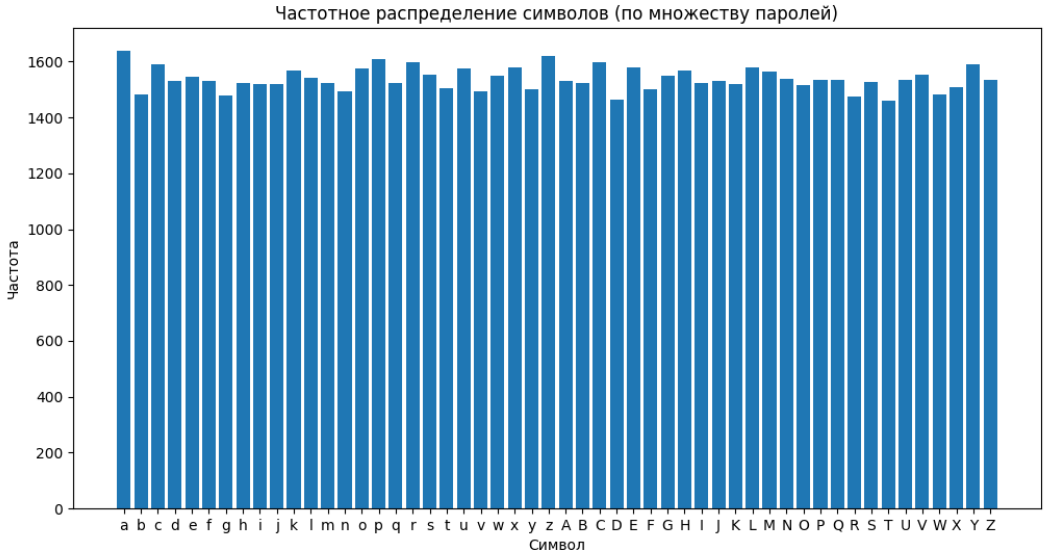

Рисунок 2 – Частотное распределение символов

Также был построен график, отображающий среднее время подбора пароля от его длины. График построен в логарифмической шкале, что позволяет наглядно увидеть экспоненциальный рост времени перебора при увеличении длины пароля.

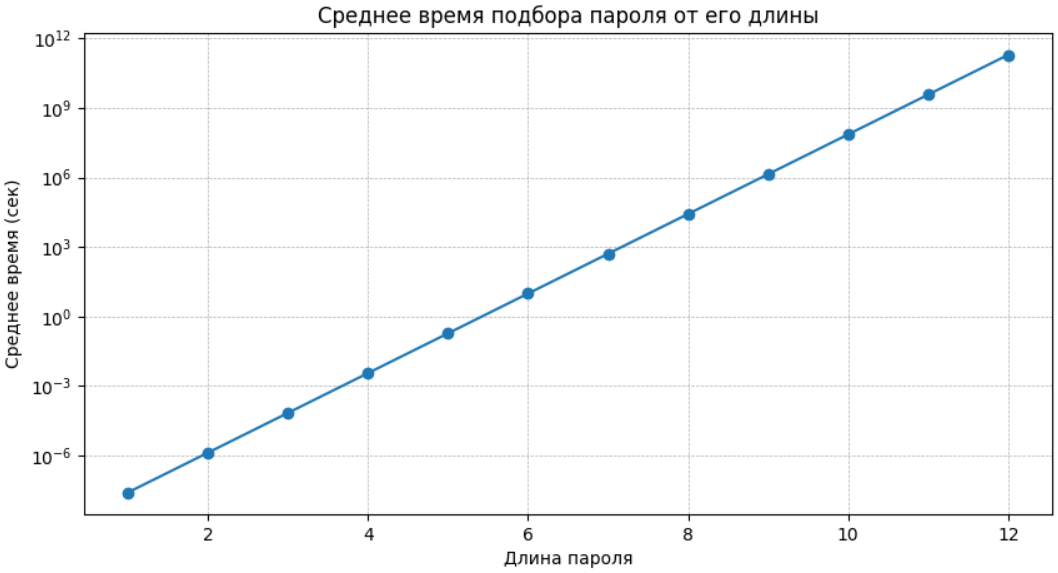

Рисунок 3 – Среднее время подбора пароля от его длины

При выборе пароля необходимо учитывать алфавит, длину пароля, ценность защищаемой информации и возможности потенциального атакующего. Использование только строчных или только прописных букв существенно ограничивает количество возможных комбинаций, что облегчает подбор пароля методом полного перебора. Комбинация строчных и прописных букв увеличивает мощность алфавита до 52 символов, а добавление цифр и специальных символов ещё больше повышает стойкость пароля.

Длина пароля играет ключевую роль. Короткие пароли (6–8 символов) подбираются за считанные секунды или минуты при современных вычислительных мощностях, в то время как пароли длиной 10 символов требуют уже порядка лет при скорости проверки 10^9 паролей в секунду. Пароли длиной 12 и более символов обеспечивают надёжную защиту даже при использовании современных GPU, способных проверять до 10^{11} комбинаций в секунду.

ВЫВОД

В работе было реализовано программное средство для генерации паролей и анализа их стойкости. Показано, что время перебора пароля экспоненциально растёт с увеличением длины. Практические рекомендации подтверждают необходимость использования длинных и сложных паролей для защиты информации.