

Разработка программы генерации паролей на языке программирования Python

Киселева Елизавета Александровна

Приамурский государственный университет им. Шолом-Алейхема

Студент

Аннотация

Вопрос информационной безопасности в наше время стоит на очень высоком месте. Практически каждый житель планеты имеет множества учетных записей почт, соцсетей и различных интернет-магазинов. К сожалению, многие люди не хотят придумывать новые пароли и используют один пароль для всего. Огромным недостатком этого то, что злоумышленник, получив пароль от одного сайта, получит доступ и к другим. Поэтому в ходе работы описана разработка программы для генерации паролей и продемонстрирован результат работы.

Ключевые слова: Python, Tkinter, Информационная безопасность.

Development of a password generation program in the Python programming language

Kiseleva Elizaveta Alexandrovna

Sholom-Aleichem Priamursky State University

Student

Abstract

The issue of information security nowadays stands in a very high place. Almost every inhabitant of the planet has many mail accounts, social networks and various online stores. Unfortunately, many people do not want to invent new passwords and use one password for everything. A huge disadvantage of this is that the attacker, having received the password from one site, will gain access to others. Therefore, in the course of the work, the development of a program for generating passwords is described and the result of the work is demonstrated.

Keywords: Python, Tkinter, Information Security.

Вопрос информационной безопасности стоит на очень высоком месте. Практически каждый житель планеты имеет множества учетных записей почт, соцсетей и различных интернет-магазинов. Утеря или кража пароля может принести большие убытки владельцу. К сожалению, многие люди не хотят придумывать новые пароли и используют один пароль для всего. Огромным недостатком этого то, что злоумышленник, получив пароль от одного сайта, получит доступ и к другим. Поэтому в ходе работы

планируется разработать программу для генерации паролей разного уровня сложности.

Цель исследования: разработка программы для генерации случайных паролей разного уровня сложности.

Многие ученые сталкивались с вопросами разработки. Е.А. Киселева [1, 2] в своих работах описала разработку программы для шифрования информации с помощью шифра виженера и провела обзор лучших GUI Фраемворков на языке Python. С.В. Радионов [3] в своей работе описал возможность разработки графических интерфейсов на языке Python3 с помощью встроенной библиотеки Tkinter. В.В. Найденев [4] в статье описал возможность тестирования производительности ORM в языках Python и C++. Ученый Н. Wang [5] продемонстрировал возможность графического интерфейса на языке программирования Python с помощью разработки игры.

Для данной реализации было решено использовать язык программирования Python. Python - высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой объём полезных функций.

К сожалению, одного питона в нашем случае не хватит, для этого было решено использовать встроенную библиотеку Tkinter. Tkinter - стандартный и надежный инструмент для работы с графическим интерфейсом на языке программирования Python. Благодаря простому использованию Tkinter является наиболее часто используемым каркасом GUI в Python.

В программе планируется возможность копирования полученного пароля. Для реализации данной функции было решено использовать библиотеку `urclip`.

Генерация пароля происходит путем нахождения случайных символов из ранее написанной строки. Каждый случайный найденный символ добавляется в пустую строчку. На рисунке 1 продемонстрирован код генерации пароля.

```
easy = "1234567890"
average = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz"
hurd = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789 !@#%&'()*"
password = ""

if var.get() == 1:
    for i in range(0, length):
        password = password + random.choice(easy)
    return password

elif var.get() == 0:
    for i in range(0, length):
        password = password + random.choice(average)
    return password

elif var.get() == 3:
    for i in range(0, length):
        password = password + random.choice(hurd)
    return password
```

Рисунок 1 – Код генерации пароля

Как видим, легкий пароль состоит только из цифр. Средний – из больших и маленьких латинских букв. Сложный – из больших, маленьких латинских букв, цифр и символов.

Длина пароля зависит от выставленных настроек, которые задаются и считывают количество кругов в цикле и записывая по одному символу за каждый круг. Затем после написания логики, с помощью библиотеки Tkinter был создан графический интерфейс.

При выставлении легкого пароля и выбранной длины программа генерирует случайный легкий пароль, состоящий только из цифр. На рисунке 2 демонстрация легкого пароля.

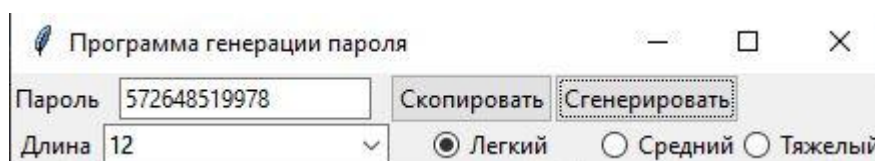


Рисунок 2 – Демонстрация лёгкого пароля

При выставлении среднего пароля и выбранной длины программа генерирует случайный средний пароль, состоящий из латинских заглавных и малых букв. На рисунке 3 демонстрация среднего пароля.

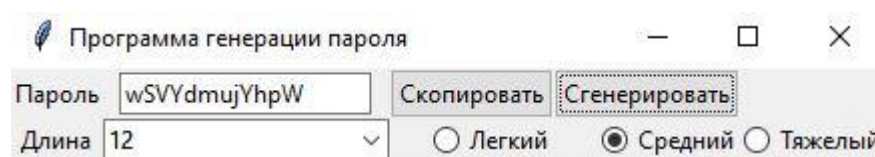


Рисунок 3 – Демонстрация среднего пароля

При выставлении сложного пароля и выбранной длины программа генерирует случайный сложный пароль, состоящий из заглавных и маленьких латинских букв, цифр и символов. На рисунке 4 демонстрация сложного пароля.

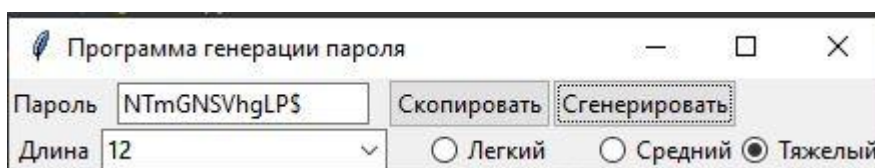


Рисунок 4 – Демонстрация сложного пароля

Во всех случаях, при нажатии на кнопку «Скопировать», сгенерированный пароль помещается в буфер обмена, с помощью чего его можно без труда вставить в требуемое место.

Заключение. Таким образом, в ходе работы была разработана программа на языке программирования Python в связке для графического отображения с библиотекой Tkinter. Программа генерирует пароли трех уровней сложности и имеет функцию выставления длины пароля и

копирования в буфер обмена. Кроме того, была успешно протестирована в работе и продемонстрирован результат в конце статьи.

Библиографический список

1. Киселева Е. А. Разработки криптографической программы на основе шифра Виженера с помощью языка программирования Python и Tkinter //Постулат. 2020. №. 1.
2. Киселева Е. А. Обзор GUI Framework на языке программирования Python //Постулат. 2020. №. 1.
3. Радионов С. В. Создание интерфейса для программы python3 с помощью библиотеки tkinter //Постулат. 2019. №. 1.
4. Найденов В. В. Тестирование производительности ORM в языках Python и C++ //RSDN Magazine. 2014. №. 1. С. 05-08.
5. Wang H. Teaching CS1 with Python GUI game programming //AIP Conference Proceedings. – American Institute of Physics, 2016. Т. 1247. №. 1. С. 253-260.
6. Python. URL: <https://www.python.org/> (дата обращения 30.01.2020)
7. Tkinter. URL: <https://docs.python.org/3/library/tkinter.html> (дата обращения 30.01.2020)
8. Puperclip. URL: <https://pypi.org/project/puperclip/> (дата обращения 30.01.2020)