PROJECT

Password Strength Checker

REPORT

Date: 3 December 2024

Created by Bhargav Raj Dutta

(International Student at Murdoch University Dubai)

3rd Year Bachelor of Information Technology in Cyber Security and Digital Forensics.

Project Overview

The Password Strength Checker is a Python program designed to assess the strength of a password that the user sets and it is based on various criteria such as length, character diversity (uppercase, lowercase, special characters, digits), and whether the password is commonly used or not. It provides feedback and suggestions on improving a weak password and helps users choose stronger passwords for better security.

Features

Password Strength Evaluation: The program checks password strength based on several factors: length, uppercase letters, lowercase letters, digits, and special characters.

Common Password Check: It verifies if the entered password is one of the most common passwords, based on a list from a common-password.txt file.

User Feedback: Provides detailed feedback on how to improve the password based on the evaluation and data prompted by the user.

Functionalities

1. **Check Common Passwords**: The program checks if the entered password matches any of the commonly known passwords from a common-password.txt file.

2. **Evaluate Password Strength**: The strength of the password is determined by:

Length of the password: Passwords longer than 8 characters are given higher scores.

Character diversity: The presence of uppercase letters, lowercase letters, digits, and special characters (e.g., @, #, $).

3. **Strength Categories:**

- **Very Weak**: Score <= 2

- **Weak**: Score = 3

- **Moderate**: Score = 4

- **Strong**: Score = 5-6

- **Very Strong**: Score >= 7

4. **Feedback & Suggestions**: For weaker passwords, the program suggests how to improve the password, such as adding numbers, special characters, and increasing length.

High-Level Algorithm for Password Strength Checker

1. **Start**: The program starts by welcoming the user and explaining its functionality.

2. **Check Password Length**:

    - If the password is shorter than 8 characters, it's considered weak.

    - Points are awarded for passwords longer than 8, 12, 15, and 20 characters.

3. **Check for Character Types**:

    - The password is analyzed for uppercase letters, lowercase letters, digits, and special characters.

    - Each valid character type adds to the score.

4. **Check for Common Passwords**:

    - The program checks if the password matches any common passwords by reading the file common-password.txt.

5. **Calculate Score**: The password's strength is determined based on its length and diversity of characters, with a maximum score of 7.

6. **Categorize Password Strength**:

    - Based on the score, the password is categorized as "Very Weak," "Weak," "Moderate," "Strong," or "Very Strong."

7. **Provide Feedback**:

If the password is weak, the program provides suggestions to improve

8. **End**: The program ends when the user chooses to exit.

Requirements

Python 3.x: The program is written in Python 3.

common-password.txt file: A text file containing a list of common passwords. This file should be in the same directory as the script for proper functionality

Setup Instructions

1. **Install Python 3.x**

Visit Python's official website (https://www.python.org/downloads/) and download the latest version of Python 3 for your operating system.

2. Create or download the common-password.txt File:

You can create a simple common-password.txt file containing common passwords or download an existing list. The passwords should be separated by line breaks. Some example passwords **include 123456, password, qwerty, etc.**

3. Place the Script and File in the Same Directory: Save the **Python script (e.g, password_strength_checker.py)** and the common-password.txt file in the same folder on your system.

4. Run the Program:

Open a terminal or command prompt and navigate to the directory where the script and file are located.

Run the following command to start the program: **python password_strength_checker.py**

Usage Instructions

1. Upon running the script, you will be presented with a welcome message and options:

Option 1: Check password strength.

Option 2: Exit the program.

2. If you select Option 1, you will be prompted to **enter your password.**

3. The program will:

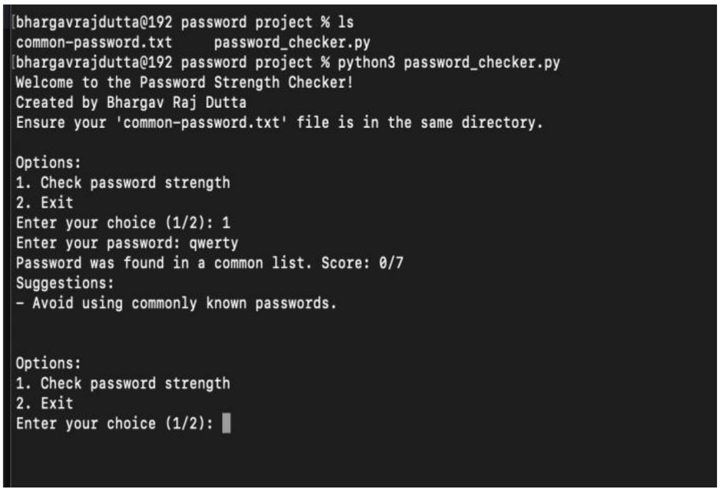
Check if your password is on the common password list.

Evaluate the password's strength based on length and character diversity.

Provide feedback and suggestions on how to improve the password if it is weak.

4. If you select Option 2, the program will exit.

Example Output



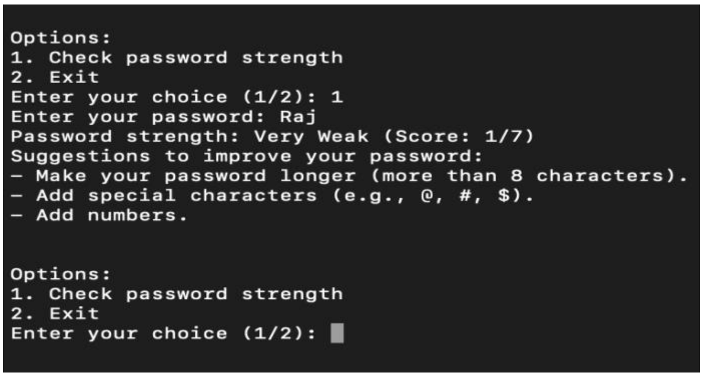
```
bhargavrajdutta@192 password project % ls
common-password.txt     password_checker.py
bhargavrajdutta@192 password project % python3 password_checker.py
Welcome to the Password Strength Checker!
Created by Bhargav Raj Dutta
Ensure your 'common-password.txt' file is in the same directory.

Options:
1. Check password strength
2. Exit
Enter your choice (1/2): 1
Enter your password: qwerty
Password was found in a common list. Score: 0/7
Suggestions:
- Avoid using commonly known passwords.


Options:
1. Check password strength
2. Exit
Enter your choice (1/2): 
```

Case 1: Common Password

Password was found in a common list. Score: 0/7

Suggestions:

- Avoid using commonly known passwords.



```
Options:
1. Check password strength
2. Exit
Enter your choice (1/2): 1
Enter your password: Raj
Password strength: Very Weak (Score: 1/7)
Suggestions to improve your password:
- Make your password longer (more than 8 characters).
- Add special characters (e.g., @, #, $).
- Add numbers.


Options:
1. Check password strength
2. Exit
Enter your choice (1/2): 
```

Case 2: Weak Password

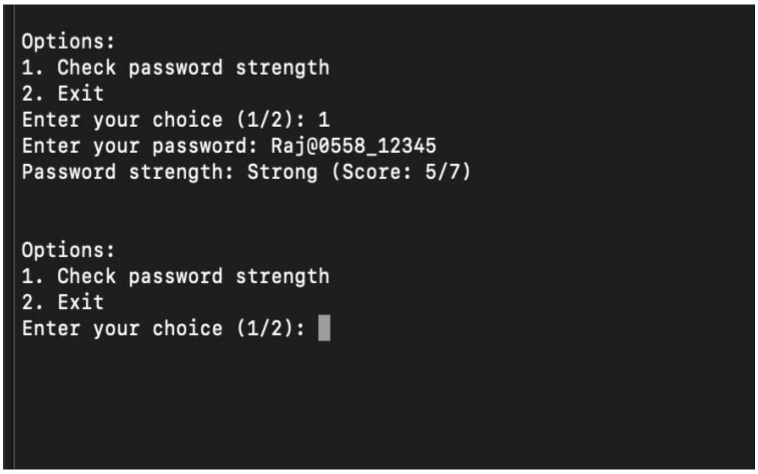
Password strength: Weak (Score: 3/7)

Suggestions to improve your password:

- Make your password longer (more than 8 characters).

- Include at least one uppercase letter.

- Add special characters (e.g., @, #, $).



```
Options:
1. Check password strength
2. Exit
Enter your choice (1/2): 1
Enter your password: Raj@0558_12345
Password strength: Strong (Score: 5/7)


Options:
1. Check password strength
2. Exit
Enter your choice (1/2): 
```

Case 3: Strong Password, Strength: Strong (Score: 5/7)

Code Explanation

Main Program Loop: The program provides a menu for the user to either check the password strength or exit the program. The program will loop until the user chooses to exit.

Functions:

1. **check_common_password(password):** Opens the common-password.txt file and checks if the entered password is present in the list of common passwords.

2**. password strength(password):** Evaluates the password's strength based on length and character

diversity, assigning points to each criterion and determining the overall strength category.

3. **feedback(password):** Provides feedback on the password strength, including suggestions for improvement if necessary.

Error Handling

File Not Found: If the common-password.txt file is not found, an error message is displayed.

Empty Password: If the user enters an empty or whitespace-only password, the program will prompt for a valid input.

Created by Bhargav Raj Dutta

Password Strength Checker - Test Cases

Test Case ID	Input Password	Expected Strength	Suggestions
TC01	12345	Very Weak	Make it longer, add letters, special characters
TC02	password	Weak	Include uppercase letters, numbers, and special characters.
TC03	Password123	Moderate	Add special characters for more security.
TC04	P@ssw0rd123	Strong	Consider making it longer for enhanced strength.
TC05	Admin2024!	Strong	Consider making it longer for enhanced strength.
TC06	admin	Very Weak	Include uppercase letters, numbers, and special characters.
TC07	aBc!12345	Strong	Consider making it longer for enhanced strength.
TC08	abcdefghijklmno	Weak	Add uppercase letters, numbers, and special characters.
TC09	P@ssword2023#	Very Strong	No suggestions. The password is strong.
TC10	User1	Very Weak	Make it longer, add letters, special characters, and numbers.
TC11	Simple123!	Moderate	Consider adding more length and diversity.
TC12	Qwerty2024@	Strong	Consider making it longer for enhanced strength.
TC13	letmein	Weak	Include uppercase letters, numbers, and special characters.
TC14	SuperSafe2022*	Strong	Consider making it longer for enhanced strength.
TC15	h@ckMeNot987!	Very Strong	No suggestions. The password is strong.
TC16	P@ssw0rd	Moderate	Consider making it longer and adding numbers.
TC17	welcome123	Weak	Add uppercase letters, numbers, and special characters.
TC18	St@ySafe#2023	Very Strong	No suggestions. The password is strong.
TC19	securepass!	Moderate	Add numbers and uppercase letters for more security.
TC20	T0pS3cret!	Very Strong	No suggestions. The password is strong.

Future Improvements

Strengthening Evaluation: The program can be enhanced to use more advanced algorithms for evaluating password strength (e.g., entropy calculations).

Enhanced Common Password List: The program can be extended to pull the list of common passwords from an online database or API for updated checks.

Graphical User Interface (GUI): A GUI can be built for users who are more comfortable with graphical interfaces rather than the command line.

Conclusion

This Password Strength Checker is a simple yet effective tool for helping users assess the strength of their passwords and improve them by following basic security practices. It encourages the use of strong, unique passwords to enhance security and reduce the likelihood of account compromise.