

Future Interns Cyber Security Internship

Akshay P

CIN: FIT/FEB25/CS505

Task 2 – Password Analyzer Tool Report

Introduction

This report provides an overview of the Password Analyzer Tool, explaining its algorithm, functionality, and effectiveness in evaluating password strength. The tool is developed using Python with Tkinter for the graphical user interface (GUI) and hashlib for encryption.

Code

```
import tkinter as tk
from tkinter import messagebox
import re
import hashlib

# Function to check password strength
def check_password_strength(password):
    strength = 0
    criteria = [
        (r"[a-z]", "Lowercase letter"),
        (r"[A-Z]", "Uppercase letter"),
        (r"[0-9]", "Digit"),
        (r"[!@#$%^&*()_+\\-=\\[\\]{};'\":<>?/|]", "Special character"), # Fixed escaping
        (r".{8,}", "At least 8 characters")
    ]

    met_criteria = [desc for pattern, desc in criteria if re.search(pattern, password)]
    strength = len(met_criteria)

    if strength == 5:
        return "Strong", met_criteria
    elif strength >= 3:
        return "Moderate", met_criteria
    else:
        return "Weak", met_criteria

# Function to hash the password
def hash_password(password):
    return hashlib.sha256(password.encode()).hexdigest()

# Function to toggle password visibility
```

```

def toggle_password():
    if entry.cget('show') == '*':
        entry.config(show='')
        toggle_button.config(text="Hide")
    else:
        entry.config(show='*')
        toggle_button.config(text="Show")

# GUI application
def analyze_password():
    password = entry.get()
    if not password:
        messagebox.showwarning("Input Error", "Please enter a password")
        return

    strength, met_criteria = check_password_strength(password)
    hashed = hash_password(password)

    result_label.config(text=f"Strength: {strength}\nMet Criteria: {'', '.join(met_criteria)}\nHashed: {hashed}")

# GUI setup
root = tk.Tk()
root.title("Password Analyzer")
root.geometry("400x300")

label = tk.Label(root, text="Enter your password:")
label.pack(pady=10)

entry = tk.Entry(root, show="*", width=30)
entry.pack(pady=5)

toggle_button = tk.Button(root, text="Show", command=toggle_password)
toggle_button.pack(pady=5)

button = tk.Button(root, text="Analyze", command=analyze_password)
button.pack(pady=10)

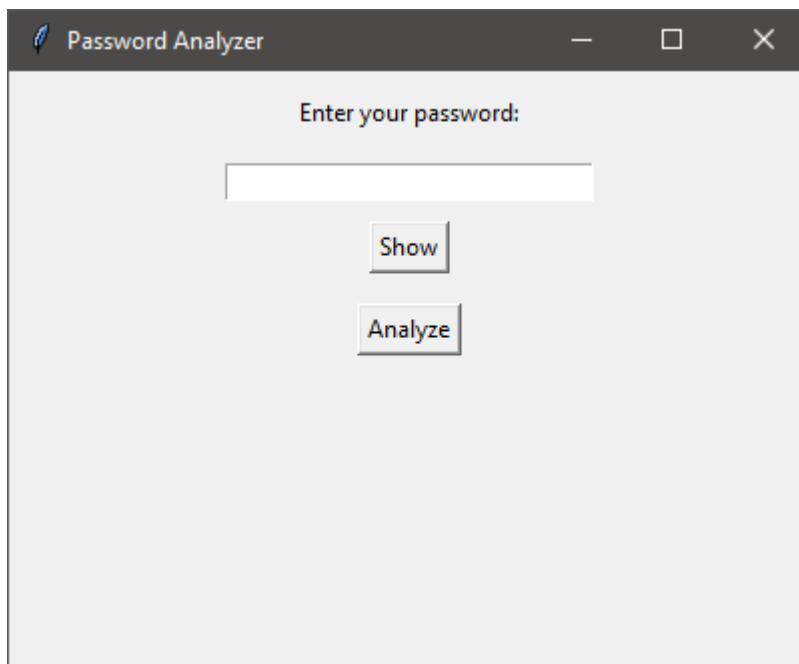
result_label = tk.Label(root, text="")
result_label.pack(pady=10)

root.mainloop()

```

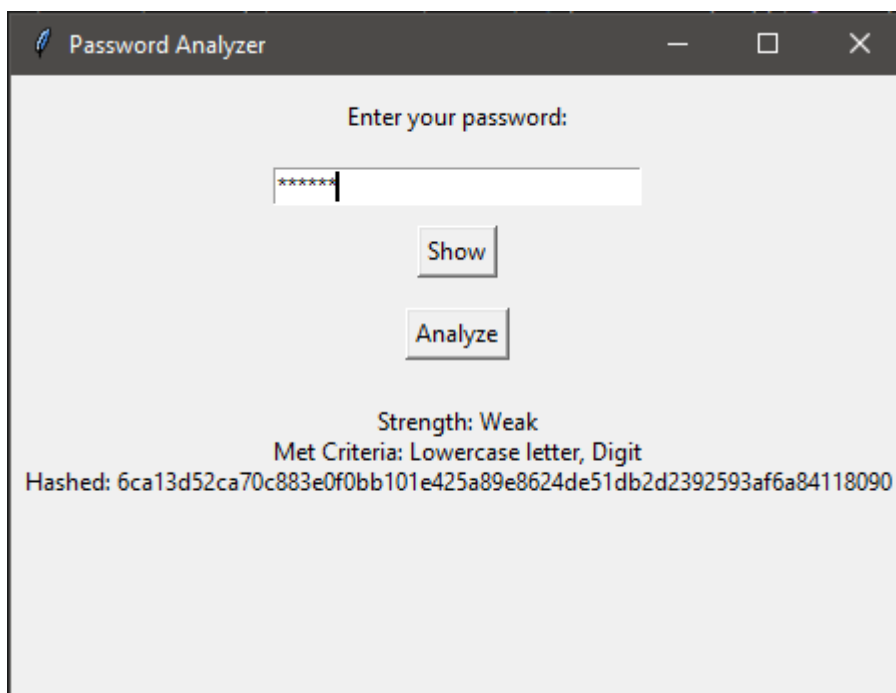
Output

Below is the output of the Password Analyzer tool.



A screenshot of a web application window titled "Password Analyzer". The window has a dark header bar with the title and standard window controls (minimize, maximize, close). The main content area is light gray and contains the text "Enter your password:" above a white text input field. Below the input field are two buttons: "Show" and "Analyze".

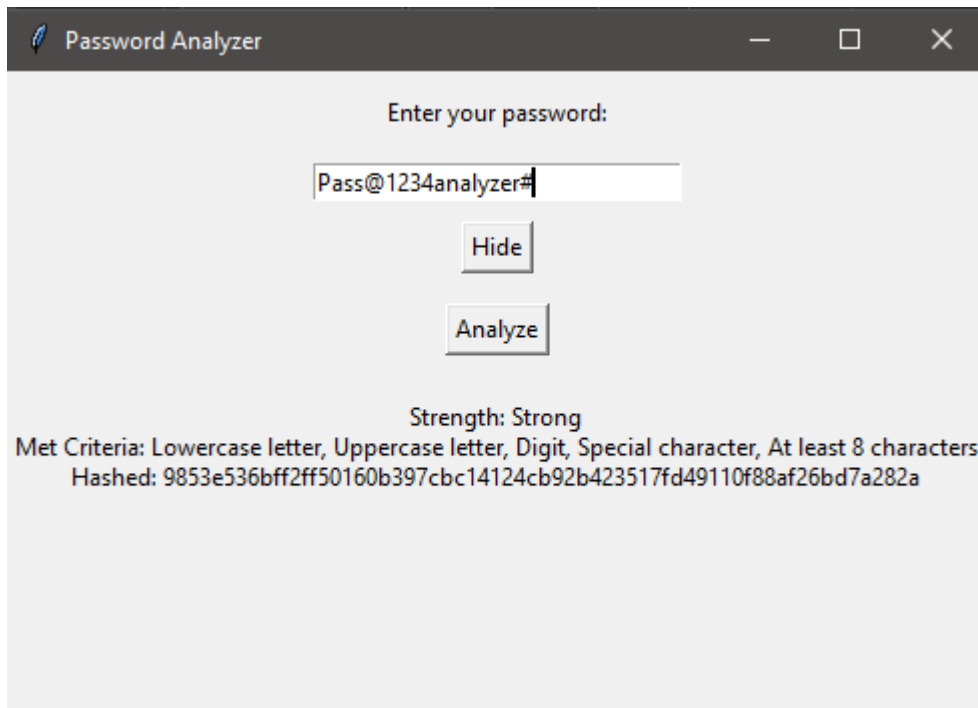
Below is a Screenshot wherein a weak password such as 'abc123' is used.



A screenshot of the same "Password Analyzer" web application window. The text input field now contains "*****" followed by a cursor. Below the input field are the "Show" and "Analyze" buttons. The "Analyze" button is highlighted with a blue border. Below the buttons, the following text is displayed:

Strength: Weak
Met Criteria: Lowercase letter, Digit
Hashed: 6ca13d52ca70c883e0f0bb101e425a89e8624de51db2d2392593af6a84118090

Below we see when a password 'Pass@1234analyzer#' is given.



The screenshot shows a window titled "Password Analyzer". Inside, there is a label "Enter your password:" above a text input field containing "Pass@1234analyzer#". Below the input field are two buttons: "Hide" and "Analyze". At the bottom of the window, the analysis results are displayed: "Strength: Strong", "Met Criteria: Lowercase letter, Uppercase letter, Digit, Special character, At least 8 characters", and "Hashed: 9853e536bff2ff50160b397cbc14124cb92b423517fd49110f88af26bd7a282a".

Algorithm

The Password Analyzer Tool employs a multi-step approach to evaluate password strength. The core algorithm includes the following steps:

1. User Input Capture: The tool takes a password as input from the user through the GUI.
2. Regular Expression (Regex) Validation: The password is checked against predefined patterns to ensure it contains a mix of uppercase letters, lowercase letters, digits, and special characters.
3. Entropy Calculation: The tool computes the entropy of the password to estimate its randomness and predictability.
4. Common Password Detection: The password is compared against a list of commonly used passwords to flag weak choices.
5. Hashing for Secure Storage: The tool uses hashlib to generate a secure hash of the password to demonstrate encryption basics.
6. Feedback to User: The tool provides an assessment of the password's strength, along with recommendations to improve it if necessary.

Effectiveness

The effectiveness of the Password Analyzer Tool is based on its ability to detect weak passwords and provide meaningful feedback. The following aspects contribute to its effectiveness:

1. **Comprehensive Pattern Matching:** The use of regex ensures that passwords meet basic security requirements.
2. **Entropy-Based Evaluation:** By calculating entropy, the tool gives a more data-driven measure of password strength.
3. **Common Password Detection:** Preventing the use of frequently used passwords enhances security.
4. **User-Friendly Feedback:** Clear recommendations help users improve their password choices.
5. **Hashing for Security Awareness:** Although not intended for secure storage, demonstrating hashing mechanisms helps users understand password security.

Limitations and Future inclusions

While the Password Analyzer Tool is effective, it has some limitations:

1. **Does Not Enforce Policies:** The tool evaluates passwords but does not enforce strong password creation.
2. **Limited Dictionary Checking:** The common password list can be expanded to detect more weak passwords.
3. **No Real-Time Breach Detection:** Future improvements could include checking passwords against real-world leaked databases.

Conclusion

The Password Analyzer Tool is an effective solution for assessing password strength based on regex validation, entropy calculation, and hashing. It serves as an educational tool to help users understand password security principles and adopt stronger password practices. Future enhancements can further improve its accuracy and usability.