Password Strength Analyzer with Custom Wordlist Generator

Introduction

Password security is a critical component of cybersecurity. Weak or predictable passwords make systems vulnerable to brute-force and dictionary attacks. To combat this, I developed a Password Toolkit that helps users evaluate the strength of their passwords and generate custom wordlists for testing or educational purposes. The project integrates both password strength analysis and wordlist generation into a single graphical interface.

Abstract

This project presents a GUI-based Python application that integrates two key functionalities: a Password Strength Analyzer and a Custom Wordlist Generator. The analyzer evaluates password robustness using the `zxcvbn` algorithm, offering actionable feedback. The generator allows users to create targeted password lists by combining base words, leetspeak variants, and number patterns. The application is built using `Tkinter` and includes usability features such as theme switching, show/hide password, and clear/reset functionality. The final output can be exported in `.txt` format for ethical testing use.

Tools Used

- **Python**: Core programming language used for logic and UI.
- **Tkinter**: Used for designing the graphical user interface.
- **zxcvbn**: A password strength estimator from Dropbox used for analyzing password complexity.
- **itertools**: Python module used to generate leetspeak variations efficiently.
- **GitHub**: Version control and project hosting.
- **VS Code** – Code editor and development environment

Steps Involved in Building the Project

1. Researched password strength metrics and leetspeak patterns.
2. Planning & Design:
    - Defined the two main modules: Analyzer and Generator
    - Designed GUI layout using Tkinter Notebook tabs
3. Password Strength Analyzer:
    - Integrated `zxcvbn` to evaluate passwords.
    - Displayed score (0-4) and suggestions using labels.
4. Wordlist Generator:
    - Accepted user-defined base words and number patterns.

- Generated variants using leetspeak and case changes.
- Saved wordlist to `custom_wordlist.txt`

5. Usability Enhancements:
    - Built a user-friendly GUI with Tkinter including:
        - Tabbed interface for analyzer and generator.
        - Light/Dark theme toggle.
        - Show/Hide password toggle.
        - Clear output buttons.
        - Generation status indicator in wordlist generator.
    - Used modular code and error handling.
    - Verified results through local testing.
6. Deployment:
    - Uploaded project to GitHub with organized structure.
    - Created README, screenshots, and documentation.

Conclusion

The Password Toolkit provides a dual-purpose solution for strengthening password practices and supporting ethical cybersecurity testing. It serves as both a learning aid and a security utility, helping users understand password strength and prepare for real-world scenarios in ethical hacking or awareness campaigns. The combination of functionality, usability, and ethical application makes this project a comprehensive and educational tool.

Note:

This project is intended strictly for ethical, educational, and awareness purposes.

Project Author:

Name: Sahiti M

Project Title: Password Toolkit- Strength Analyzer + Wordlist Generator

GitHub: password_analyzer_generator

Date: June 2025