Password Strength Analyzer & Wordlist Generator

Introduction

With the increasing reliance on digital services, password-based authentication remains a critical line of defense for user data security. However, many users still opt for weak or predictable passwords, making their accounts vulnerable to brute force or dictionary attacks. This project addresses this concern by providing a dual-purpose tool: one that evaluates the strength of user-defined passwords and another that generates personalized wordlists to simulate common attack scenarios.

Abstract

This project aims to develop a Python-based application capable of analyzing password strength and generating attack-specific wordlists. The tool supports both command-line and graphical user interfaces to make it accessible to both technical and non-technical users. For password analysis, the tool integrates the zxcvbn library to estimate crack time and provide real-time feedback. For wordlist generation, it collects user-specific inputs such as names, dates, or personal preferences, and generates permutations using common password patterns like leetspeak and year suffixes. The generated wordlists can be exported as .txt files for further use in penetration testing or auditing. The GUI, developed using Tkinter, enhances usability and provides an intuitive workflow.

Tools Used

- **Python 3.11**: Core language for scripting and logic development.
- **Tkinter**: Used to build the graphical user interface.
- **argparse**: For implementing command-line interaction.
- **zxcvbn (Python port)**: For password strength estimation and feedback.
- **UUID & datetime**: For uniquely naming output files with timestamps.
- **OS & Subprocess**: To handle file saving, opening, and directory management.

Steps Involved in Building the Project

1. Planning & Requirement Analysis: The initial phase involved identifying key functionalities: password analysis and custom wordlist generation. User requirements such as optional inputs, export format, and usability were gathered.
2. CLI Implementation: A command-line interface was first created using argparse to allow users to either analyze a password (--password) or generate a wordlist using inputs like name, birthdate, or pet name.

3. Password Analysis: Integrated zxcvbn to estimate password strength, provide a security score (0–4), and give textual feedback regarding common vulnerabilities or improvement suggestions.
4. Wordlist Generation Logic: A custom module was developed to permute user inputs with patterns like leetspeak, reversed strings, and numeric suffixes. The resulting list could be exported to a .txt file inside the wordlists/ directory.
5. GUI Development: Using Tkinter, a user-friendly tabbed interface was built with separate sections for password analysis and wordlist generation. Features like password preview toggle, overwrite prompt, and auto-open file after generation were added.
6. Output Handling: If the user doesn't provide a filename, the tool automatically creates one using the current date, time, and a short UUID string to ensure uniqueness.

Conclusion

The Password Strength Analyzer & Wordlist Generator provides a practical and versatile tool for both cybersecurity enthusiasts and professionals. By combining password analysis with customizable wordlist creation, it serves both defensive (user education) and offensive (red teaming, testing) purposes. With support for both GUI and CLI, the tool adapts well to different use cases, offering a comprehensive solution for password security evaluation. Future enhancements could include hash format exports, advanced entropy modeling, or integration with real-time breach databases.