Password Strength Analyzer and Wordlist Generator Report

Abstract

This project develops a Python-based tool to analyze password strength and generate custom wordlists for security testing. The tool uses the zxcvbn library to evaluate password strength and creates wordlists from user inputs, incorporating variations like leetspeak and appended years. The command-line interface ensures ease of use, with results exported to a text file for further use in security applications.

Introduction

Passwords are a critical component of cybersecurity, yet weak passwords remain a common vulnerability. This project aims to create a tool that assesses password strength and generates custom wordlists to simulate password-cracking scenarios. The tool helps users understand password vulnerabilities and supports ethical security testing by creating targeted wordlists based on user-provided inputs, such as names or dates.

Tools Used

The project leverages the following tools and libraries:

- **Python 3**: Core programming language for implementation.
- **zxcvbn**: Library for password strength analysis, providing entropy and crack time estimates.
- **argparse**: Python module for handling command-line arguments.
- **itertools**: For generating combinations of user inputs.

Steps Involved in Building the Project

1. **Setup**: Installed Python and required libraries (zxcvbn, argparse) using pip.
2. **Password Analysis**: Implemented a function using zxcvbn to analyze password strength, returning a score (0–4), estimated crack time, and improvement suggestions.
3. **Wordlist Generation**: Developed functions to generate wordlists from user inputs, including:
    - Combinations of inputs (e.g., name + pet).
    - Leetspeak variations (e.g., replacing 'a' with '@' or '4').
    - Appending years (2000–2025) to words.
4. **Output Handling**: Created a function to save the wordlist to a .txt file.
5. **Command-Line Interface**: Used argparse to allow users to input a password for analysis and/or inputs for wordlist generation via CLI.
6. **Testing**: Tested the tool with sample inputs (e.g., `python password_analyzer.py --password` to verify functionality.

Conclusion

The Password Strength Analyzer and Wordlist Generator successfully provides an accessible tool for evaluating password security and creating custom wordlists. The use of zxcvbn ensures accurate strength analysis, while the wordlist generator enhances its utility for ethical hacking scenarios. Future improvements could include a graphical interface using tkinter and support for additional variation patterns. This project demonstrates practical applications of Python in cybersecurity, offering valuable insights for secure password practices.