- **Password Strength Analyzer with Custom Wordlist Generator**
- **Prepared by: Dushyant Chaudhary**
- **Cybersecurity Intern**
- **Date: 10-09-2025**

Password Strength Analyzer with Custom Wordlist Generator

Introduction

Passwords remain the first line of defense for personal and professional digital security. Weak or easily guessable passwords are a critical vulnerability, making strong password creation and auditing essential for every user and organization.

Abstract

This project delivers a Python-based tool that analyzes the strength of user-supplied passwords and generates custom password wordlists using hints such as names, dates, or pet names. Security testers and individuals can use this tool to both evaluate existing passwords and generate targeted wordlists for password audits or penetration testing exercises. The aim is to raise awareness about password robustness and provide practical tools for security evaluation.

Tools Used

- **Python 3.x**
- **zxcvbn-python** (for realistic password strength evaluation)
- **Standard Python libraries** (for file handling and string manipulation)

Steps Involved in Building the Project

1. **Project Setup:**
    - Organized code and outputs in designated folders for clarity and reproducibility (e.g., `wordlists/` for generated files).
2. **Password Analysis:**

- Used the `zxcvbn-python` library to compute a detailed strength score (0–4) and provide feedback on user-input passwords, including estimated crack times and suggestions for improvement.

3. **Custom Wordlist Generation:**
    - Collected hints from the user (common personal data like names, dates, or keywords).
    - Generated common password variants such as lower/upper case, leetspeak substitutions, and numeric suffixes based on these hints.
    - Wrote all wordlist entries to a text file inside `wordlists/`, for use in security testing tools.
4. **Command-Line Interface:**
    - Implemented an interactive CLI script (`analyzer.py`) that guides users through password analysis and wordlist creation in a single session.

Conclusion

This project equips users with an efficient way to measure password strength and craft custom wordlists for penetration testing or security audits. The tool emphasizes ease of use, rapid feedback, and practical output, making it valuable for anyone who wants to better understand or improve password security. The extensible codebase allows for further enhancements such as more sophisticated wordlist rules or integration with GUI frameworks.

Note:

- The `analyzer.py` script and supporting wordlists are included in the project repository.
- The tool was developed and tested on Windows. It can be adapted to other platforms with minimal changes.