

## **Project Title: Password Strength Analyzer with Custom Wordlist Generator**

**Your Name:** Abhyas Kathuria

**Internship:** Elevate Labs Cybersecurity Internship

**Project Number:** 4

**Introduction** This project involves building a Python-based tool that evaluates password strength and generates custom wordlists based on user-provided personal information. The objective is to demonstrate common password vulnerabilities, highlight the risks of weak and predictable passwords, and promote awareness of secure password practices. The tool is developed purely for educational purposes and aligns with ethical cybersecurity learning.

### **Abstract**

**The application performs two main functions:**

1. **Password Strength Analysis** – Uses the zxcvbn library (used by Dropbox) to provide realistic strength scoring (0–4), estimated crack time, entropy calculation, warnings, and improvement suggestions.
2. **Custom Wordlist Generation** – Takes personal inputs (name, birth year, pet name, city, etc.), applies transformations such as leetspeak, capitalization, year/number appending, reversals, and pairwise combinations, and optionally includes common English words from NLTK. The generated wordlist is exported as a .txt file. The tool supports both Command Line Interface (CLI) and a user-friendly Tkinter Graphical User Interface (GUI).

### **Tools Used**

- Python 3.x
- zxcvbn (for realistic strength scoring)
- NLTK (for common English words integration)
- Tkinter (for GUI)
- argparse, datetime, math (built-in)

### **Steps Involved in Building the Project**

1. Set up Python environment and installed required libraries (zxcvbn, NLTK).
2. Implemented password analysis: Used zxcvbn for score/crack time/feedback + custom entropy formula.
3. Built wordlist generator: Added leetspeak variations, year/number appending, capitalization, reversals, and pairwise combinations.
4. Integrated optional NLTK common words for more realistic lists.
5. Added file export functionality and CLI with argparse.

6. Created Tkinter GUI for easy interaction (analysis + generation with file save dialog).
7. Tested thoroughly with weak/strong passwords and various inputs; verified outputs.

## Conclusion

This project provided hands-on experience with real-world password security concepts, including dictionary attacks, pattern-based cracking, and entropy measurement. I gained deep understanding of why weak passwords are easily compromised and the importance of unpredictability. The tool successfully meets all deliverables: it evaluates password strength accurately and generates attack-specific wordlists. Limitations: Designed only for educational use and awareness – not for actual malicious activities. Future enhancements could include integration with larger public datasets or hashing demonstration. This project has significantly boosted my ability to explain password security topics confidently in technical interviews.

```
>>> nltk.download('words')
[nltk_data] Downloading package words to
[nltk_data] C:\Users\kathu\AppData\Roaming\nltk_data...
[nltk_data] Package words is already up-to-date!
True
>>> python password_analyzer.py --analyze "password123"
File "stdin", line 1
python password_analyzer.py --analyze "password123"
SyntaxError: invalid syntax
>>> exit()
(env) PS C:\Users\kathu\Desktop\projects\PasswordAnalyzer> python password_analyzer.py --analyze "password123"

=== PASSWORD ANALYSIS ===
Password      : password123
Strength Score: 0/4
Crack Time    : less than a second
Entropy       : 56.87 bits
Warning       : This is a very common password.
Suggestions   : Add another word or two. Uncommon words are better.
(env) PS C:\Users\kathu\Desktop\projects\PasswordAnalyzer> python password_analyzer.py --analyze "Tr0ub4dor&3xp10r3r2025!"

=== PASSWORD ANALYSIS ===
Password      : Tr0ub4dor&3xp10r3r2025!
Strength Score: 4/4
Crack Time    : centuries
Entropy       : 150.76 bits
Warning       : No warning.
Suggestions   : Strong password! No suggestions.
(env) PS C:\Users\kathu\Desktop\projects\PasswordAnalyzer> python password_analyzer.py --generate Abhyas 2007 dog delhi --use_nltk

=== GENERATING WORDLIST ===

Wordlist exported to: C:\Users\kathu\Desktop\projects\PasswordAnalyzer\custom_wordlist.txt (1281 entries)
(env) PS C:\Users\kathu\Desktop\projects\PasswordAnalyzer> python password_analyzer.py

Wordlist exported to: C:\Users\kathu\Desktop\projects\PasswordAnalyzer\custom_wordlist2.txt (1281 entries)
(env) PS C:\Users\kathu\Desktop\projects\PasswordAnalyzer>
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\kathu\Desktop\projects\PasswordAnalyzer> python -m venv env
PS C:\Users\kathu\Desktop\projects\PasswordAnalyzer> env\Scripts\activate
(env) PS C:\Users\kathu\Desktop\projects\PasswordAnalyzer> pip install zxcvbn
Collecting zxcvbn
  Using cached zxcvbn-4.5.0-py2.py3-none-any.whl.metadata (5.9 kB)
  Using cached zxcvbn-4.5.0-py2.py3-none-any.whl (489 kB)
Installing collected packages: zxcvbn
Successfully installed zxcvbn-4.5.0

[notice] A new release of pip is available: 24.2 -> 25.3
[notice] To update, run: python.exe -m pip install --upgrade pip
(env) PS C:\Users\kathu\Desktop\projects\PasswordAnalyzer> pip install nltk
Collecting nltk
  Downloading nltk-3.9.2-py3-none-any.whl.metadata (3.2 kB)
Collecting click (from nltk)
  Downloading click-8.3.1-py3-none-any.whl.metadata (2.6 kB)
Collecting joblib (from nltk)
  Downloading joblib-1.5.3-py3-none-any.whl.metadata (5.5 kB)
Collecting regex-2021.8.3 (from nltk)
  Downloading regex-2025.11.3-cp312-cp312-win_and64.whl.metadata (41 kB)
Collecting tqdm (from nltk)
  Using cached tqdm-4.67.1-py3-none-any.whl.metadata (57 kB)
Collecting colorama (from click->nltk)
  Using cached colorama-0.4.6-py2.py3-none-any.whl.metadata (17 kB)
  Downloading nltk-3.9.2-py3-none-any.whl (1.5 MB)
  Downloading regex-2025.11.3-cp312-cp312-win_and64.whl (277 kB)
  Downloading click-8.3.1-py3-none-any.whl (108 kB)
  Downloading joblib-1.5.3-py3-none-any.whl (309 kB)
  Using cached tqdm-4.67.1-py3-none-any.whl (78 kB)
  Using cached colorama-0.4.6-py2.py3-none-any.whl (25 kB)
Installing collected packages: regex, joblib, colorama, tqdm, click, nltk
Successfully installed click-8.3.1 colorama-0.4.6 joblib-1.5.3 nltk-3.9.2 regex-2025.11.3 tqdm-4.67.1

[notice] A new release of pip is available: 24.2 -> 25.3
[notice] To update, run: python.exe -m pip install --upgrade pip
(env) PS C:\Users\kathu\Desktop\projects\PasswordAnalyzer> pip install argparse
Collecting argparse
```

