# Password Strength Analyzer with Custom Wordlist Generator

---

## Title:

**Password Strength Analyzer with Custom Wordlist Generator**

---

## Introduction:

Passwords are the first line of defense in cybersecurity, yet weak passwords continue to be a leading cause of security breaches. This project aims to help users assess the strength of their passwords and generate personalized wordlists using common patterns. These wordlists can be used for penetration testing or user education, demonstrating how easily personal information can be leveraged in attacks.

---

## Abstract:

The password analyzer uses the `zxcvbn` library, developed by Dropbox, to estimate how secure a password is based on known patterns and entropy. In addition, a wordlist generator was built that takes user-specific information (such as name, birth year, and pet's name) and transforms it using leetspeak, appended years, and reversed combinations. This approach demonstrates how attackers generate custom dictionaries for password attacks. The project is entirely command-line based and is written in Python.

---

## Tools Used:

- Python 3
- `zxcvbn-python` library
- `argparse` (for command-line input handling)
- `itertools` (for combinations)
- Linux Terminal
- Optional: `venv` for isolated environment

---

### Steps Involved in Building the Project:

1. **Environment Setup:**

   - Created a virtual environment using `venv`
   - Installed dependencies (`zxcvbn-python`, `argparse`)

2. **Password Strength Evaluation:**
   - User enters a password as input.
   - The tool analyzes it using zxcvbn and provides:
     - Score (0–4)
     - Number of guesses
     - Crack time estimates
     - Feedback for improvement
3. **Custom Wordlist Generation:**
   - Accepts user inputs (e.g., name, DOB, pet).
   - Generates combinations with:
     - Appended and prepended years
     - Leetspeak substitutions (e.g., “a” → “@”, “e” → “3”)
     - Reversed strings and merged variants
4. **Export Feature:**
   - Wordlist saved as `.txt` (compatible with password cracking tools)
   - Sample file includes 100–200 variants based on inputs
5. **Testing and Validation:**
   - Ran with known weak and strong passwords
   - Verified output of wordlist and analysis feedback

---

## Conclusion:

This tool provides users and ethical hackers with an easy way to analyze password strength and simulate realistic password attacks using custom wordlists. It helps raise awareness of password reuse and predictability and shows the importance of using unique, complex, and unrelated passwords. The project is beginner-friendly, easy to extend, and aligns with real-world cybersecurity practices.