Password Strength Analyzer & Custom Wordlist Generator

Abstract:

This project evaluates password strength and generates a custom wordlist based on user-provided details such as name, pet name, and year. The analyzer checks password complexity factors including length, uppercase, lowercase, digits, and symbols, and calculates an entropy score. The wordlist generator simulates real-world dictionary attack techniques used by attackers, demonstrating why predictable passwords are vulnerable. This project helps understand password security and the importance of strong, complex passwords.

Introduction:

Password-based authentication is the most common security mechanism used across applications, yet weak passwords remain one of the biggest cybersecurity risks. Attackers often utilize dictionary attacks, where customized wordlists based on personal information are used to guess passwords. This project replicates that concept by analyzing user passwords and creating targeted wordlists. Through this, the importance of creating unpredictable, strong passwords becomes clear.

Objectives:

- Analyze password strength using character composition metrics.
- Calculate entropy to estimate password unpredictability.
- Generate custom wordlists using personal details.
- Demonstrate how dictionary attacks are formed.
- Educate users on password security best practices.

Tools Used:

- Python 3
- String operations
- Entropy scoring logic
- Basic transformation algorithms (leetspeak)

Steps Involved:

1. The user enters a password to analyze.
2. The program evaluates complexity: length, uppercase, lowercase, digits, and symbols.
3. An entropy score is generated based on character uniqueness and password length.
4. User inputs name, pet name, and year to generate personalized wordlist entries.
5. The tool creates variations and leetspeak mutations commonly used in cracking tools.
6. All generated words are exported to a file named *custom_wordlist.txt*.

Sample Output:

```
Password Strength: Medium
Entropy Score: 45
✔ Contains lowercase
✔ Contains digits
✘ No special characters
```

Conclusion:

This project effectively demonstrates password analysis and dictionary generation methods widely used in cybersecurity. By understanding how attackers construct targeted wordlists and evaluate password strength, users can improve their own password hygiene. The project highlights the importance of choosing strong, complex, and unpredictable passwords for secure authentication practices.