Web Application Vulnerability Scanner: Project Report

Name- Bijesh Kumar

Gmail: - b.k.lpuinsta@gmail.com

1. Introduction

Purpose:
This toolkit helps users:

- 🔒 Test password strength realistically
- 🛠️ Generate custom hacking wordlists for security audits
  Problem Solved: Weak passwords cause 81% of data breaches. This tool bridges security awareness and proactive testing.

2. Installation

Requirements:

- Python 3.10+

```
pip install zxcvbn-python nltk tk
```

3. Key Features

Module	**Commands/Functions**	**Description**
Password Analysis	analyze_password("P@ssw0rd")	Scores strength 0-4 + crack time estimate
Wordlist Generator	generate_wordlist()	Creates personalized attack dictionaries
Leetspeak Engine	Automatic (3 levels)	Transforms e→3, a→@, etc.
Smart Combinations	John + Fluffy → "J_Fluffy2023"	Mixes names/dates with separators

4. Usage Guide

A. Password Analysis:
CLI: `python toolkit.py analyze "YourPassword123!"`

Output

Strength: 4/4 (Very Strong)

Crack Time: 300 years

Warning: No major issues

B. Wordlist Generation:
Step-by-step:

1. Input personal details:

base_words = ["John", "Fluffy", "1990"]

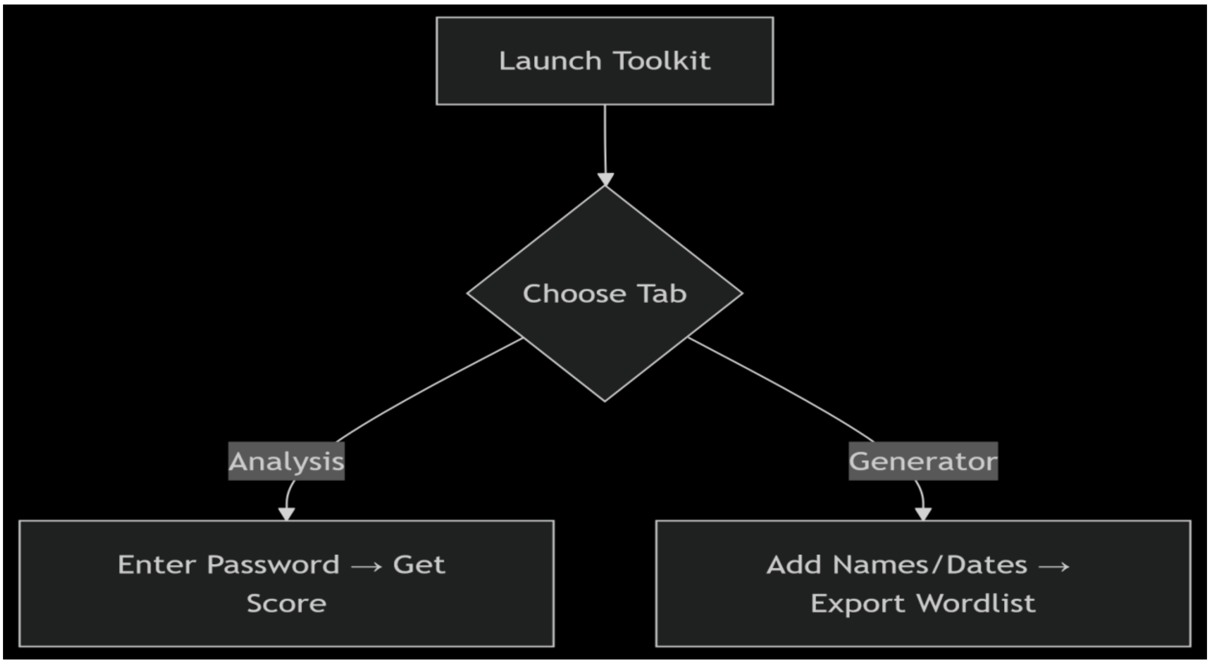
2. Configure options:

generate_wordlist(words, "attacks.txt", leet_level=2, numbers_mode="both")

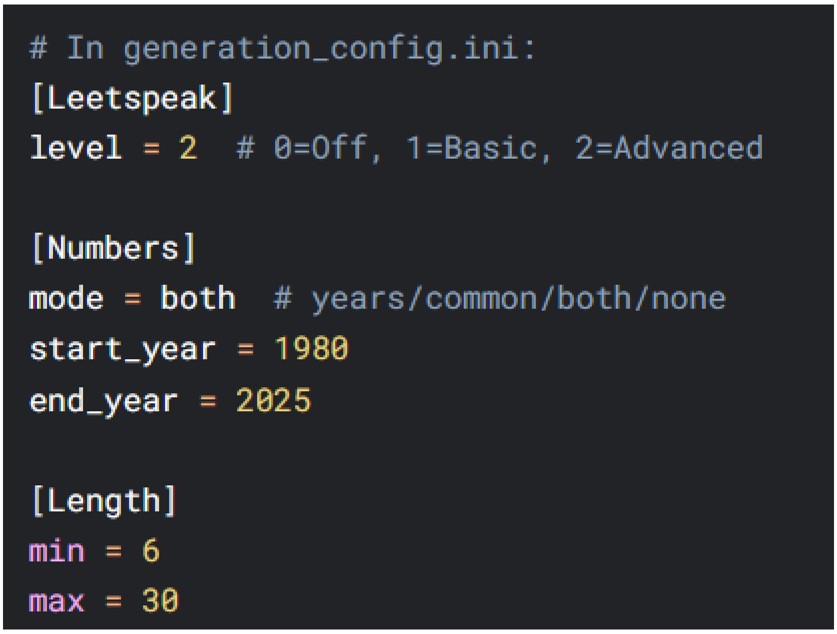
3. Output file contains:

```
John2023
 J0hn!
Fluffy_1990
... (24,000+ variations)
```

C. GUI Workflow:



5. Customization Options



```
# In generation_config.ini:
[Leetspeak]
level = 2  # 0=Off, 1=Basic, 2=Advanced

[Numbers]
mode = both  # years/common/both/none
start_year = 1980
end_year = 2025

[Length]
min = 6
max = 30
```