Title Page

Title:

Password Strength Checker and Brute-Force Estimation Tool

Submitted by:

Srushti Mhatre & Aisha Shaikh

Cybersecurity Internship – 2025

Digisuraksha Parhari Foundation

Infinisec Technologies Pvt. Ltd.

Date:

12th May 2025

Abstract

Passwords are the first line of defense for digital systems, yet many users continue to choose weak and easily guessable passwords. This project presents a lightweight Python-based tool that evaluates password strength using basic character heuristics and estimates brute-force cracking time based on complexity. The tool checks for length, character variety, and presence in a list of commonly used passwords. Additionally, it demonstrates how long a password might take to be cracked using brute-force attacks at an assumed rate of one million guesses per second. The tool aims to increase user awareness about strong password practices and the real-world implications of weak credentials. Designed to be simple, educational, and entirely offline, this tool is suitable for cybersecurity awareness training, classrooms, or personal use.

Problem Statement & Objective

Weak passwords are a major cause of security breaches, accounting for a significant percentage of successful attacks. Users often select passwords that are short, predictable, or reused across platforms. The goal of this project is to build a basic tool that educates users about password strength by analyzing common patterns and estimating brute-force risk.

Literature Review

[1] Verizon Data Breach Report (2023): Over 80% of breaches involved weak or stolen credentials.

[2] OWASP Top 10: Password misuse remains a persistent security concern.

[3] NIST Password Guidelines (2022): Emphasizes use of long, complex, and unique passwords.

[4] “Cracking Passwords Using Brute-Force” – SANS Institute Whitepaper.

[5] GitHub Security Blog – Credential stuffing and dictionary attacks.

[6] Microsoft Report (2023): The average user reuses passwords across 5+ services.

[7] "Improving Password Strength Metering" – Egelman et al., IEEE 2017.

[8] Keepass, LastPass, Bitwarden – Popular password managers suggest complexity requirements.

[9] Kali Linux tools – Password cracking utilities (Hydra, John the Ripper).

[10] Real-world password dumps (e.g., RockYou.txt) used in credential stuffing.

Research Methodology

This project was developed using the Python programming language. The tool uses string analysis to evaluate password strength based on:

Length

Character types (uppercase, lowercase, digits, symbols)

Brute-force time estimation is based on calculating the total number of character combinations and dividing it by a fixed guessing rate (1 million guesses per second).

Tool Implementation

Language: Python 3.x

No external libraries required

Two modules:

1. check_strength() — evaluates character properties and flags weak passwords
2. estimate_brute_force_time() — calculates estimated crack time using complexity

Supports real-time input through command line

Results & Observations

Most weak passwords were under 8 characters and used common patterns.

Use of symbols and length significantly increased estimated crack time.

Ethical Impact & Market Relevance

This tool is intended for ethical use only, such as personal awareness, training environments, and cybersecurity workshops. It highlights the importance of using strong and unique passwords and provides a non-threatening way to visualize brute-force vulnerability.

Future Scope

Add GUI interface for broader user access

Integrate with password managers

Expand logic to detect keyboard patterns (e.g., "qwerty", "asdf")

Add dictionary attack simulation mode

References

(See the 10 references listed in Literature Review)