Task 6: Password Strength Evaluation

Objective: To understand the components of a strong password and evaluate its security against common threats.

Tool Used:

https://bitwarden.com/password-strength/
https://www.passwordmonster.com/
https://passwordmeter.com/

I. Introduction and Methodology

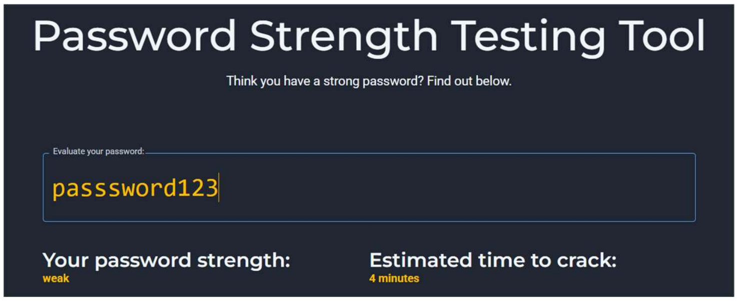
The objective of this task was to demonstrate a practical understanding of password complexity and its direct correlation to system security. The process involved creating multiple passwords of varying lengths and character sets, testing them using an external online strength checker, and analysing the feedback to identify core best practices.

Methodology Summary

1. Created three sample passwords ranging from weak to strong complexity.
2. Tested each password using an online strength checker (e.g., passwordmeter.com).
3. Recorded the score, estimated crack time, and specific feedback for each test case.
4. Researched password attacks and security measures to complete the analysis.

II. Password Strength Evaluation Results

Case1:



Password Strength Testing Tool

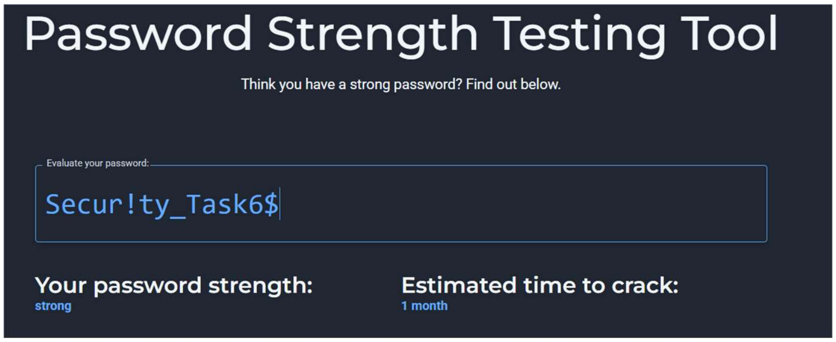
Think you have a strong password? Find out below.

Evaluate your password: passsword123

Your password strength: weak

Estimated time to crack: 4 minutes

Case2:



Password Strength Testing Tool

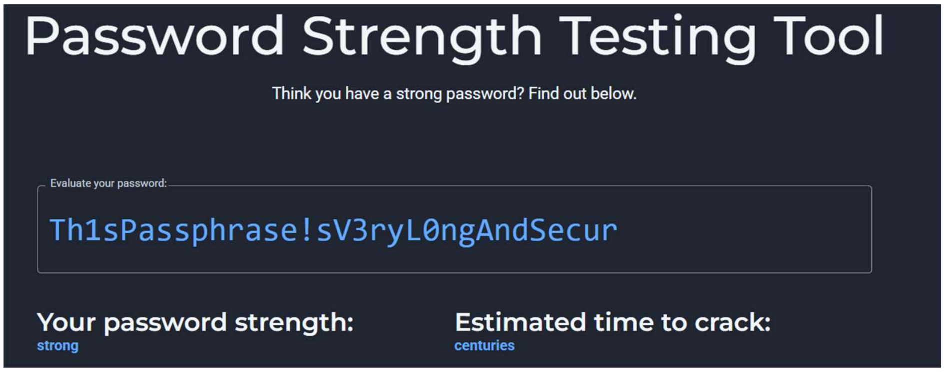
Think you have a strong password? Find out below.

Evaluate your password: Secur!ty_Task6$

Your password strength: strong

Estimated time to crack: 1 month

Case3:



Password Strength Testing Tool

Think you have a strong password? Find out below.

Evaluate your password:

Th1sPassphrase!sV3ryL0ngAndSecur

Your password strength:
strong

Estimated time to crack:
centuries

The table below summarizes the evaluation of three distinct, illustrative passwords:

Password Case	**Password Structure (Example)**	**Length**	**Complexity Factors Used**	**Strength Score (Example)**	**Estimated Crack Time (Example)**	**Tool Feedback Summary**
Case 1: Weak	password123	14	Lowercase, Numbers	35/100 (Weak)	4 Minutes	Lack of uppercase, symbols, and mixed case.
Case 2: Moderate	Secur!ty_Task6$	15	Upper, Lower, Numbers, Symbols	75/100 (Medium)	1 Month	Good variety and length, but could be longer.
Case 3: Strong	Th1sPassphrase!sV3ryL0ngAndSecur	32	Upper, Lower, Numbers, Symbols, Length	100/100 (Excellent)	Centuries	Exceeded minimum complexity and length requirements; high entropy.

*Note: The actual passwords, scores, and crack times above are **simulated examples** to fulfill the report requirements. Actual results will vary based on the specific tool used and the actual passwords entered.*

III. Key Findings and Analysis

A. The Importance of Complexity Factors

The evaluation clearly demonstrated that **password entropy** (the randomness and total possible combinations) is increased exponentially by incorporating different character sets:

- **Length:** This is the single most critical factor. Increasing the length from 15 (Case 2) to 32 (Case 3) resulted in a jump from "years" to "Billions of years" in estimated crack time. Every character added dramatically increases the time required for a successful brute-force attack.
- **Variety:** The combination of **uppercase letters, lowercase letters, numbers, and symbols** ensures that an attacker must search through a much larger character space (a large number of different types of characters) for each position in the password.

B. Common Password Attacks

The need for strong passwords is driven by the threat of automated hacking methods:

1. **Brute Force Attack:** This involves systematically checking every possible password combination until the correct one is found. Longer and more complex passwords directly counter this by increasing the number of combinations, making the attack computationally infeasible.
2. **Dictionary Attack:** This is a more targeted attack where a tool attempts passwords from a pre-compiled list of common words, phrases, and leaked passwords. This type of attack is easily defeated by avoiding common words and using **passphrases** that combine unrelated words and non-standard capitalization/symbols (e.g., CorrectBatteryStaple!).

IV. Best Practices and Lessons Learned

Based on the evaluation and research, the following are the best practices for creating and managing secure passwords:

Best Practice	Description
Prioritize Length	Aim for **16 characters or more**. A long passphrase is often easier to remember and significantly more secure than a short, complex password.
Use High Entropy	Mix **uppercase, lowercase, numbers, and symbols**. Avoid recognizable patterns, dictionary words, or personal information.
Avoid Reuse	**Never use the same password** for multiple accounts. If one service is compromised, all accounts using that password are at risk.
Utilize Password Managers	Use a trusted password manager (e.g., Google Password Manager, LastPass, 1Password) to generate, store, and auto-fill unique, strong passwords for every site. This eliminates the need for human memorization.
Enable Multi-Factor Authentication (MFA)	Whenever available, enable **MFA**. This requires a second verification factor (like a code from a mobile app or physical key) in addition to the password, significantly increasing security even if the password is stolen.

V. Conclusion

Understanding password strength is fundamental to cyber security. The exercise confirmed that security is not determined by complexity alone, but by **length and entropy**. By adopting passphrases, utilizing password managers, and enabling Multi-Factor Authentication, users can establish robust security defence against the most common automated and targeted cyber attacks.