Create a Strong Password and Evaluate Its Strength

Introductions:
In today's digital age, passwords serve as the first line of defense against unauthorized access to our personal and professional information. With the increasing number of cyber threats and data breaches, understanding what makes a password strong has become crucial for maintaining security. Weak or commonly used passwords can be easily cracked by attackers using techniques such as brute force or dictionary attacks, leaving sensitive data vulnerable.

The objective of this report is to evaluate the strength of various passwords using online password strength checkers and to analyze the factors that contribute to a password's security. By testing different combinations of characters, lengths, and complexity levels, we aim to identify best practices for creating secure passwords. This report also explores common password attacks and highlights the importance of using strong, unique credentials to protect against them.

Through this exercise, we seek to develop a practical understanding of password security and offer actionable tips for users to improve their password hygiene.

Objective: To understand what makes a password strong and analyze it using free online password strength tools.

Tool To be Used:
PasswordMeter: https://www.passwordmeter.com/

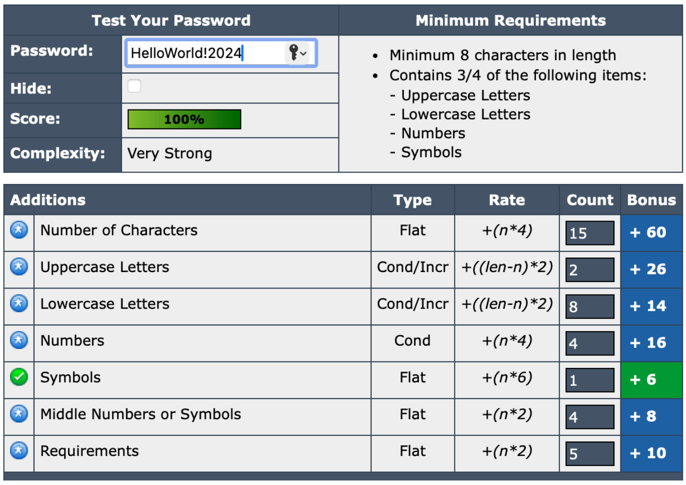
Create Sample Passwords as Follows

Table 1

Password	Length	Uppercase	Lowercase	Numbers	Symbols
hello123	8	No	Yes	Yes	No
HelloWorld!2024	14	Yes	Yes	Yes	Yes
123456	6	No	No	Yes	No
P@$$w0rd!	9	Yes	Yes	Yes	Yes
MyDogRexIs#1!	14	Yes	Yes	Yes	Yes

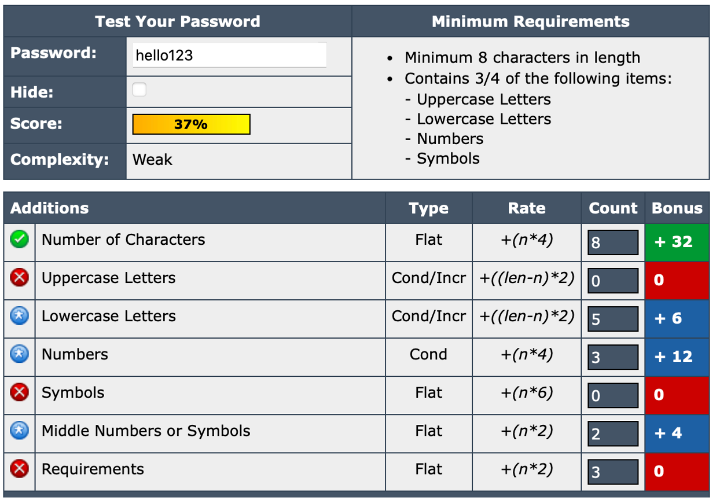
Test Passwords on Password Strength Tools

Using PasswordMeter.com tool to check the overall complexity of the Passwords in **table 1**



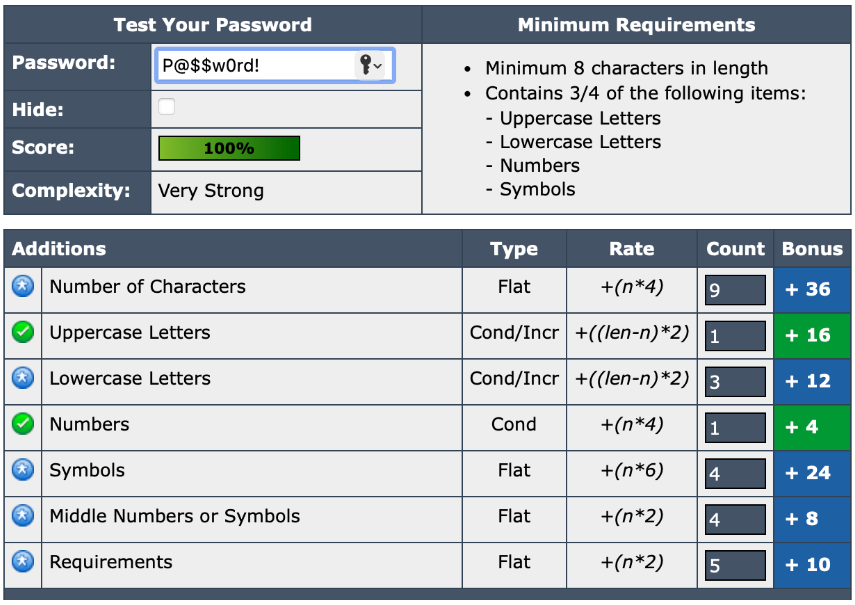
Test Your Password		Minimum Requirements
Password:	HelloWorld!2024	- Minimum 8 characters in length - Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	□	
Score:	100%	
Complexity:	Very Strong	

	Additions	Type	Rate	Count	Bonus
★	Number of Characters	Flat	*+(n*4)*	15	+ 60
★	Uppercase Letters	Cond/Incr	*+((len-n)*2)*	2	+ 26
★	Lowercase Letters	Cond/Incr	*+((len-n)*2)*	8	+ 14
★	Numbers	Cond	*+(n*4)*	4	+ 16
✓	Symbols	Flat	*+(n*6)*	1	+ 6
★	Middle Numbers or Symbols	Flat	*+(n*2)*	4	+ 8
★	Requirements	Flat	*+(n*2)*	5	+ 10



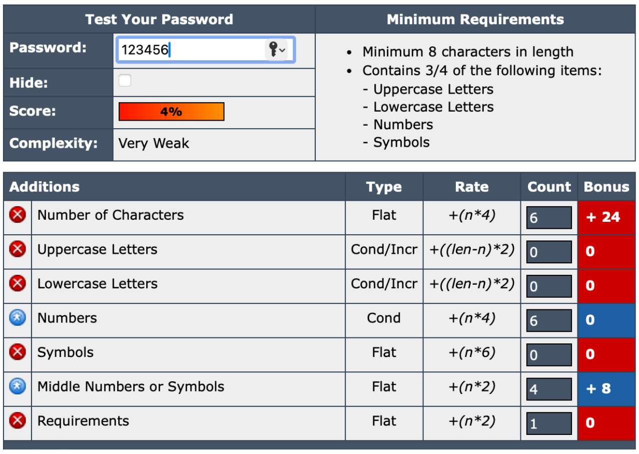
Test Your Password		Minimum Requirements
Password:	hello123	- Minimum 8 characters in length - Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	□	
Score:	37%	
Complexity:	Weak	

	Additions	Type	Rate	Count	Bonus
✓	Number of Characters	Flat	*+(n*4)*	8	+ 32
✗	Uppercase Letters	Cond/Incr	*+((len-n)*2)*	0	0
★	Lowercase Letters	Cond/Incr	*+((len-n)*2)*	5	+ 6
★	Numbers	Cond	*+(n*4)*	3	+ 12
✗	Symbols	Flat	*+(n*6)*	0	0
★	Middle Numbers or Symbols	Flat	*+(n*2)*	2	+ 4
✗	Requirements	Flat	*+(n*2)*	3	0



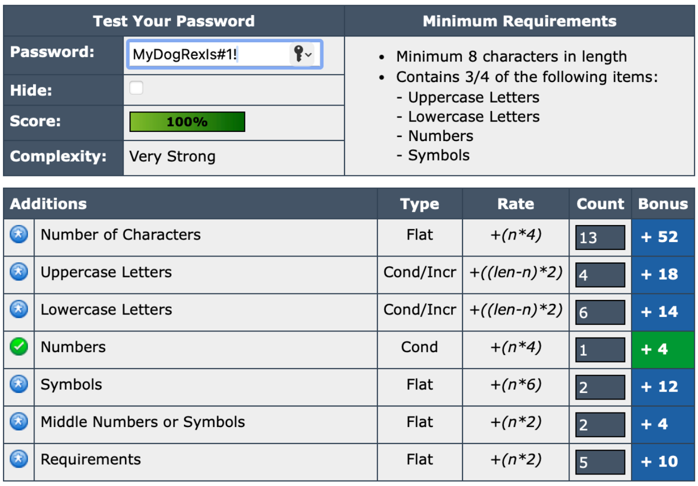
Test Your Password		Minimum Requirements
Password:	P@$$w0rd!	- Minimum 8 characters in length - Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	☐	
Score:	100%	
Complexity:	Very Strong	

	Additions	Type	Rate	Count	Bonus
★	Number of Characters	Flat	+(n*4)	9	+ 36
✓	Uppercase Letters	Cond/Incr	+((len-n)*2)	1	+ 16
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	3	+ 12
✓	Numbers	Cond	+(n*4)	1	+ 4
★	Symbols	Flat	+(n*6)	4	+ 24
★	Middle Numbers or Symbols	Flat	+(n*2)	4	+ 8
★	Requirements	Flat	+(n*2)	5	+ 10



Test Your Password		Minimum Requirements
Password:	123456	- Minimum 8 characters in length - Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	☐	
Score:	4%	
Complexity:	Very Weak	

	Additions	Type	Rate	Count	Bonus
✗	Number of Characters	Flat	+(n*4)	6	+ 24
✗	Uppercase Letters	Cond/Incr	+((len-n)*2)	0	0
✗	Lowercase Letters	Cond/Incr	+((len-n)*2)	0	0
★	Numbers	Cond	+(n*4)	6	0
✗	Symbols	Flat	+(n*6)	0	0
★	Middle Numbers or Symbols	Flat	+(n*2)	4	+ 8
✗	Requirements	Flat	+(n*2)	1	0



Test Your Password		Minimum Requirements
Password:	MyDogRexIs#1!	- Minimum 8 characters in length - Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	☐	
Score:	100%	
Complexity:	Very Strong	

	Additions	Type	Rate	Count	Bonus
★	Number of Characters	Flat	+(n*4)	13	+ 52
★	Uppercase Letters	Cond/Incr	+((len-n)*2)	4	+ 18
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	6	+ 14
✓	Numbers	Cond	+(n*4)	1	+ 4
★	Symbols	Flat	+(n*6)	2	+ 12
★	Middle Numbers or Symbols	Flat	+(n*2)	2	+ 4
★	Requirements	Flat	+(n*2)	5	+ 10

Tips & Best Practices for Strong Passwords

What Makes a Password Strong

1. **Length Matters**
    - Passwords with 12 or more characters are significantly stronger.
    - Longer passwords increase the time and effort required for brute-force attacks.
2. **Character Variety Enhances Strength**
    - Strong passwords include a mix of:
        - **Uppercase letters (A–Z)**
        - **Lowercase letters (a–z)**
        - **Numbers (0–9)**
        - **Symbols (@, #, $, %, etc.)**
3. **Unpredictability Is Key**
    - Avoid using common words, phrases, or personal information (e.g., name, birth year).
    - Use random combinations or passphrases (e.g., Sun$et-Road_42!Tree).
4. **Avoid Repetition and Patterns**
    - Passwords like abc123, qwerty, or password1 are predictable and easily cracked.
    - Tools often flag these as very weak due to their prevalence in breach databases.
5. **Substitutions Alone Aren't Enough**
    - Replacing letters with symbols (e.g., P@ssw0rd) slightly improves strength but is still vulnerable if the pattern is common.

Common Weaknesses Identified

- Short passwords (under 8 characters)
- Use of dictionary words or personal info
- Lack of symbols or mixed case
- Reused passwords across accounts

Common Password Attacks

Brute Force Attack

- Tries every possible combination.
- Short or simple passwords are easy targets.
- A 6-character password (only lowercase) can be cracked in seconds.

Cons:

- Takes a lot time to crack the password as guessing the password will not give any proper idea of what the password is.

Dictionary Attack

- Uses known words or phrases from a dictionary.
- Common for passwords like password, admin, welcome123.
- Faster than Brute Force

Cons:

- Weak Passwords can easily be cracked

Rainbow Attack

- When a password is stored securely, it is usually **hashed** using algorithms like MD5, SHA-1, or SHA-256.
- A rainbow table contains a **list of precomputed hash values** for many possible password combinations.
- The attacker compares the **hashed password** (stolen from a database breach) to entries in the rainbow table.
- If a match is found, the corresponding **original password** is revealed.

Cons: if password is salted then the rainbow table is useless

Why Password Complexity Matters ?

Password complexity plays a vital role in protecting digital accounts and sensitive information from unauthorized access. In cybersecurity, the strength of a password is directly related to how difficult it is for an attacker to guess or crack it using automated tools. Simple or predictable passwords are highly vulnerable to cyberattacks, while complex ones offer significantly better protection.

Weak Passwords Are Easily Cracked

Hackers often use automated techniques like **brute-force attacks** and **dictionary attacks** to break passwords:

- In a **brute-force attack**, every possible combination of characters is tried until the correct one is found. Short and simple passwords can be cracked in seconds.
- In a **dictionary attack**, common passwords and words from dictionaries are tested. If a password uses real words or patterns like "password123" or "welcome2024," it can be cracked almost instantly.

Complexity Increases Security

Adding complexity to a password increases the number of possible combinations exponentially. For example:

- A 6-character password with only lowercase letters has **308 million** combinations.
- A 10-character password with uppercase, lowercase, numbers, and symbols has over **60 trillion** combinations.

Complexity Components That Matter

1. **Length** – The longer the password, the more difficult it is to crack.
2. **Character Variety** – Use of uppercase, lowercase, numbers, and special characters improves unpredictability.
3. **Avoiding Common Patterns** – Random combinations are far stronger than predictable words or sequences.
4. **Uniqueness** – Using the same password across accounts increases risk. If one account is compromised, others become vulnerable too.

Conclusion

Password complexity is not just a best practice—it’s a critical defense mechanism against increasingly sophisticated cyber threats. By using long, random, and diverse passwords, users can significantly reduce their chances of falling victim to password-based attacks.