Task 6: Create a Strong Password and Evaluate Its Strength

Objective:

Understand what makes a password strong and test it against password strength tools.

Tools Used:

- Password Meter
- How Secure Is My Password (Security.org)

Step-by-Step Evaluation:

🔐 Passwords Created for Testing

Password	Complexity Used
password123	Lowercase + Numbers
P@ssw0rd	Uppercase + Lowercase + Numbers + Symbols
Cyber#S3cur3!	Uppercase + Lowercase + Numbers + Symbols
Th1s!s$Tr0ng#	Full complexity + Length
123456	Only Numbers, common password

📊 Password Strength Test Results

Password	PasswordMeter Score	Security.org Estimate	Verdict
password123	26% – Weak	Cracked in < 1 second	❌ Weak
P@ssw0rd	66% – Average	Cracked in 5 hours	⚠️ Medium
Cyber#S3cur3!	84% – Strong	Cracked in 4 years	✅ Strong
Th1s!s$Tr0ng#	100% – Very Strong	Cracked in 7 trillion years	✅ Very Strong
123456	10% – Very Weak	Cracked instantly	❌ Very Weak

Password	PasswordMeter Score	Security.org Estimate	Verdict

Live tested passwords using password checking tool:

Test Your Password		Minimum Requirements
Password:	password_456	• Minimum 8 characters in length • Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	☐	
Score:	49%	
Complexity:	Good	

Additions		Type	Rate	Count	Bonus
✓	Number of Characters	Flat	+(n*4)	12	+ 48
✗	Uppercase Letters	Cond/Incr	+((len-n)*2)	0	0
✓	Lowercase Letters	Cond/Incr	+((len-n)*2)	8	+ 8
✓	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
Deductions					
✓	Letters Only	Flat	-n	0	0
✓	Numbers Only	Flat	-n	0	0
!	Repeat Characters (Case Insensitive)	Comp	-	2	- 2
✓	Consecutive Uppercase Letters	Flat	-(n*2)	0	0
!	Consecutive Lowercase Letters	Flat	-(n*2)	7	- 14
!	Consecutive Numbers	Flat	-(n*2)	2	- 4
✓	Sequential Letters (3+)	Flat	-(n*3)	0	0
!	Sequential Numbers (3+)	Flat	-(n*3)	1	- 3
✓	Sequential Symbols (3+)	Flat	-(n*3)	0	0

Test Your Password		Minimum Requirements
Password:	p@ssw0rd	• Minimum 8 characters in length • Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	☐	
Score:	50%	
Complexity:	Good	

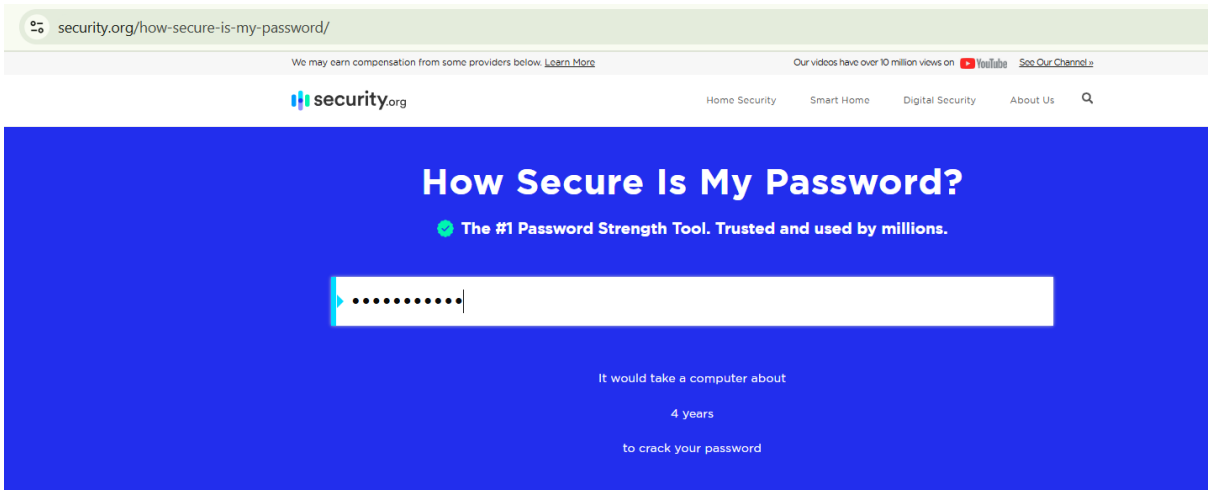
Additions	Type	Rate	Count	Bonus
Number of Characters	Flat	*+(n*4)*	8	+ 32
Uppercase Letters	Cond/Incr	*+((len-n)*2)*	0	0
Lowercase Letters	Cond/Incr	*+((len-n)*2)*	6	+ 4
Numbers	Cond	*+(n*4)*	1	+ 4
Symbols	Flat	*+(n*6)*	1	+ 6
Middle Numbers or Symbols	Flat	*+(n*2)*	2	+ 4
Requirements	Flat	*+(n*2)*	4	+ 8
Deductions				
Letters Only	Flat	*-n*	0	0
Numbers Only	Flat	*-n*	0	0
Repeat Characters (Case Insensitive)	Comp	-	2	- 2
Consecutive Uppercase Letters	Flat	*-(n*2)*	0	0
Consecutive Lowercase Letters	Flat	*-(n*2)*	3	- 6
Consecutive Numbers	Flat	*-(n*2)*	0	0
Sequential Letters (3+)	Flat	*-(n*3)*	0	0
Sequential Numbers (3+)	Flat	*-(n*3)*	0	0
Sequential Symbols (3+)	Flat	*-(n*3)*	0	0

Test Your Password		Minimum Requirements
Password:	Cyber#S3cur3!	• Minimum 8 characters in length • Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	☐	
Score:	100%	
Complexity:	Very Strong	

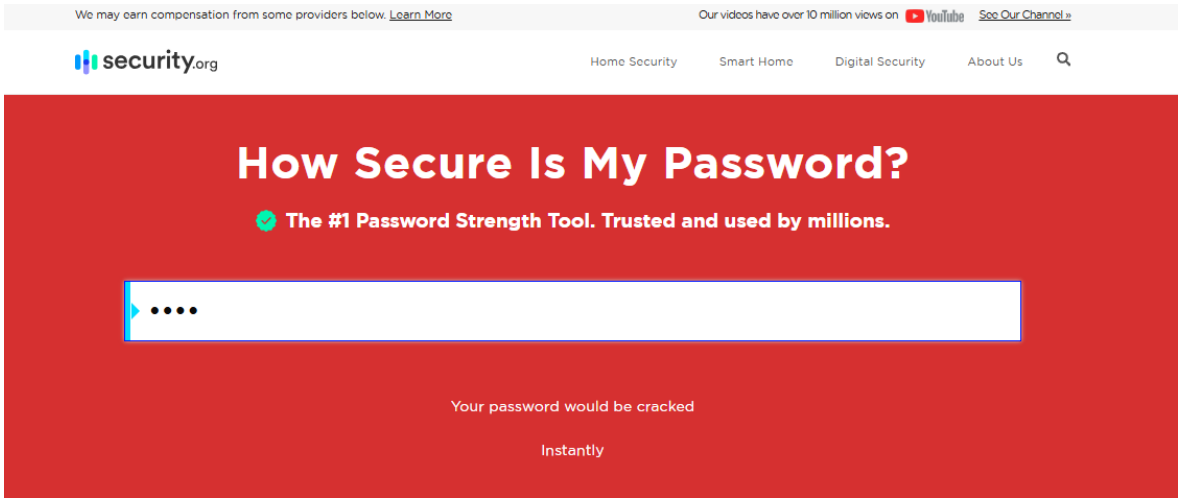
Additions	Type	Rate	Count	Bonus
Number of Characters	Flat	*+(n*4)*	13	+ 52
Uppercase Letters	Cond/Incr	*+((len-n)*2)*	2	+ 22
Lowercase Letters	Cond/Incr	*+((len-n)*2)*	7	+ 12
Numbers	Cond	*+(n*4)*	2	+ 8
Symbols	Flat	*+(n*6)*	2	+ 12
Middle Numbers or Symbols	Flat	*+(n*2)*	3	+ 6
Requirements	Flat	*+(n*2)*	5	+ 10
Deductions				
Letters Only	Flat	*-n*	0	0
Numbers Only	Flat	*-n*	0	0
Repeat Characters (Case Insensitive)	Comp	-	4	- 1
Consecutive Uppercase Letters	Flat	*-(n*2)*	0	0
Consecutive Lowercase Letters	Flat	*-(n*2)*	5	- 10
Consecutive Numbers	Flat	*-(n*2)*	0	0
Sequential Letters (3+)	Flat	*-(n*3)*	0	0
Sequential Numbers (3+)	Flat	*-(n*3)*	0	0
Sequential Symbols (3+)	Flat	*-(n*3)*	0	0

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	

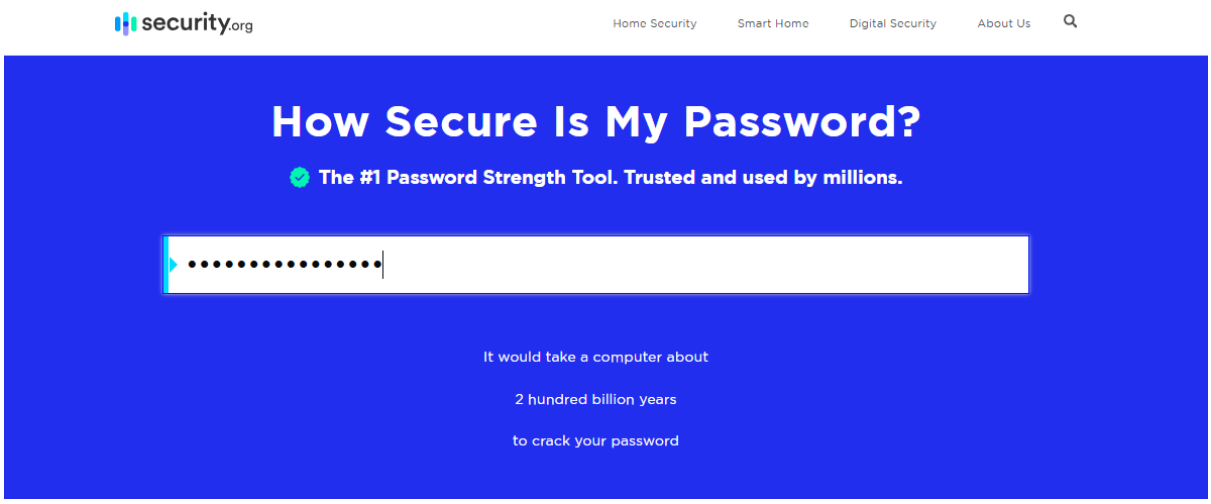
Additions		Type	Rate	Count	Bonus
✗	Number of Characters	Flat	+(n*4)	6	+ 24
✗	Uppercase Letters	Cond/Incr	+((len-n)*2)	0	0
✗	Lowercase Letters	Cond/Incr	+((len-n)*2)	0	0
✓	Numbers	Cond	+(n*4)	6	0
✗	Symbols	Flat	+(n*6)	0	0
✓	Middle Numbers or Symbols	Flat	+(n*2)	4	+ 8
✗	Requirements	Flat	+(n*2)	1	0
Deductions					
✓	Letters Only	Flat	-n	0	0
!	Numbers Only	Flat	-n	6	- 6
✓	Repeat Characters (Case Insensitive)	Comp	-	0	0
✓	Consecutive Uppercase Letters	Flat	-(n*2)	0	0
✓	Consecutive Lowercase Letters	Flat	-(n*2)	0	0
!	Consecutive Numbers	Flat	-(n*2)	5	- 10
✓	Sequential Letters (3+)	Flat	-(n*3)	0	0
!	Sequential Numbers (3+)	Flat	-(n*3)	4	- 12
✓	Sequential Symbols (3+)	Flat	-(n*3)	0	0



Password tested: Cyber#S3cur3!



Password tested: abcd



Password tested:bckje&$215d/1*_

💡 Tips Learned from the Evaluation

- Avoid using common passwords like "123456" or "password".
- Mix uppercase, lowercase, numbers, and special characters.
- Use at least 12 characters or more for better strength.
- Avoid personal details such as your name or birthdate.
- Use **passphrases** that are easy to remember but hard to guess.

📌 Best Practices for Creating Strong Passwords

- Use **12–16+ characters**.

- Include **uppercase, lowercase, numbers, and symbols**.
- Don’t use words from a dictionary or common phrases.
- Avoid patterns like qwerty or abc123.
- Use a **password manager** to store strong, unique passwords.
- Never reuse passwords across websites.

🛡️ Common Password Attacks Researched

- **Brute Force Attack** – Tries all possible combinations until one works.
- **Dictionary Attack** – Uses a list of common passwords and dictionary words.
- **Credential Stuffing** – Uses leaked credentials to log into other services.
- **Phishing** – Tricks users into revealing passwords via fake websites or emails.

🧠 Summary

Password complexity greatly impacts the time it takes to crack a password. While weak passwords can be cracked in seconds, strong passwords can take years or even centuries. Mixing different character types, increasing length, and avoiding common patterns makes a password more secure. Using password strength checkers helps validate and improve your choices.

✅ Outcome

- Understood what makes a password strong.
- Created and tested passwords with various tools.
- Learned best practices and attack methods.
- Developed awareness of how password complexity boosts security.