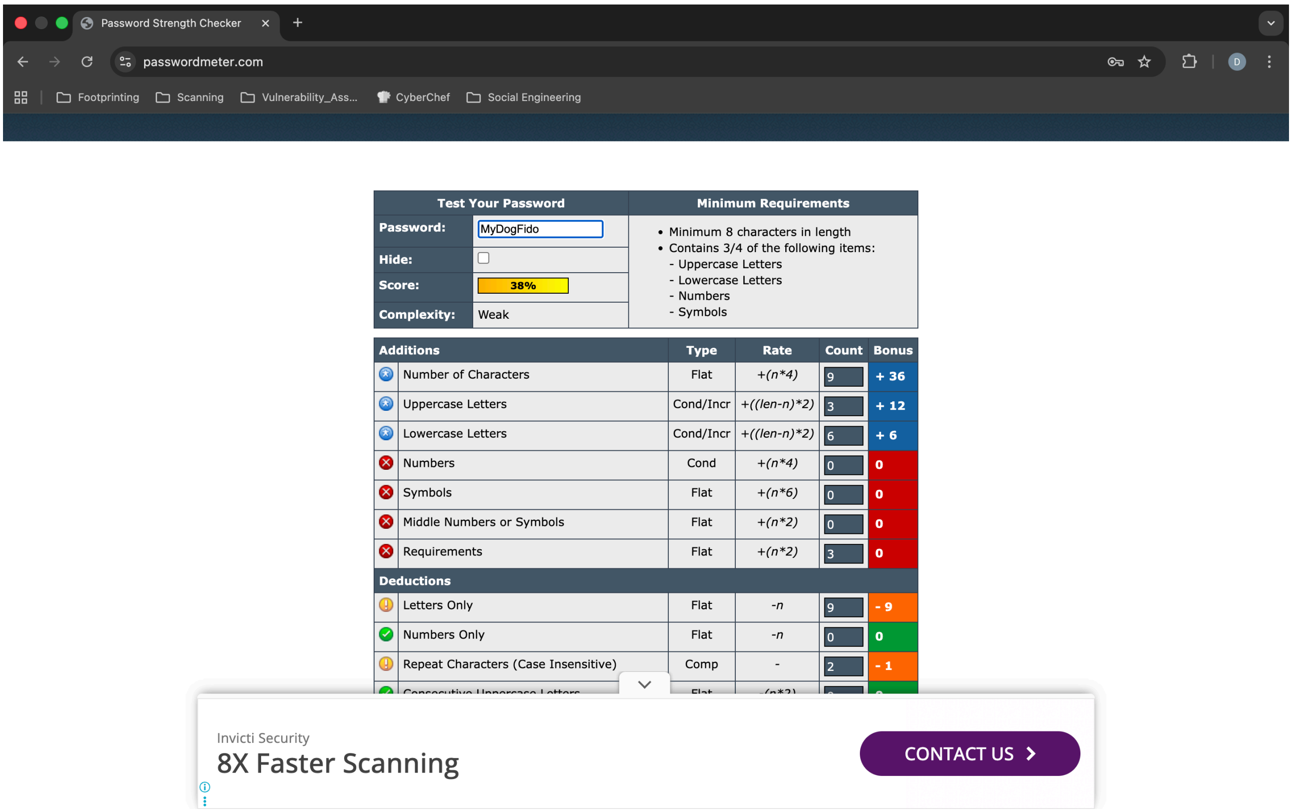
Password Security Analysis and Best Practices Report

Date: September 30, 2025

This report outlines the process of analyzing various password complexities, simulating their strength scores, and summarizing established cybersecurity best practices to ensure robust digital security.

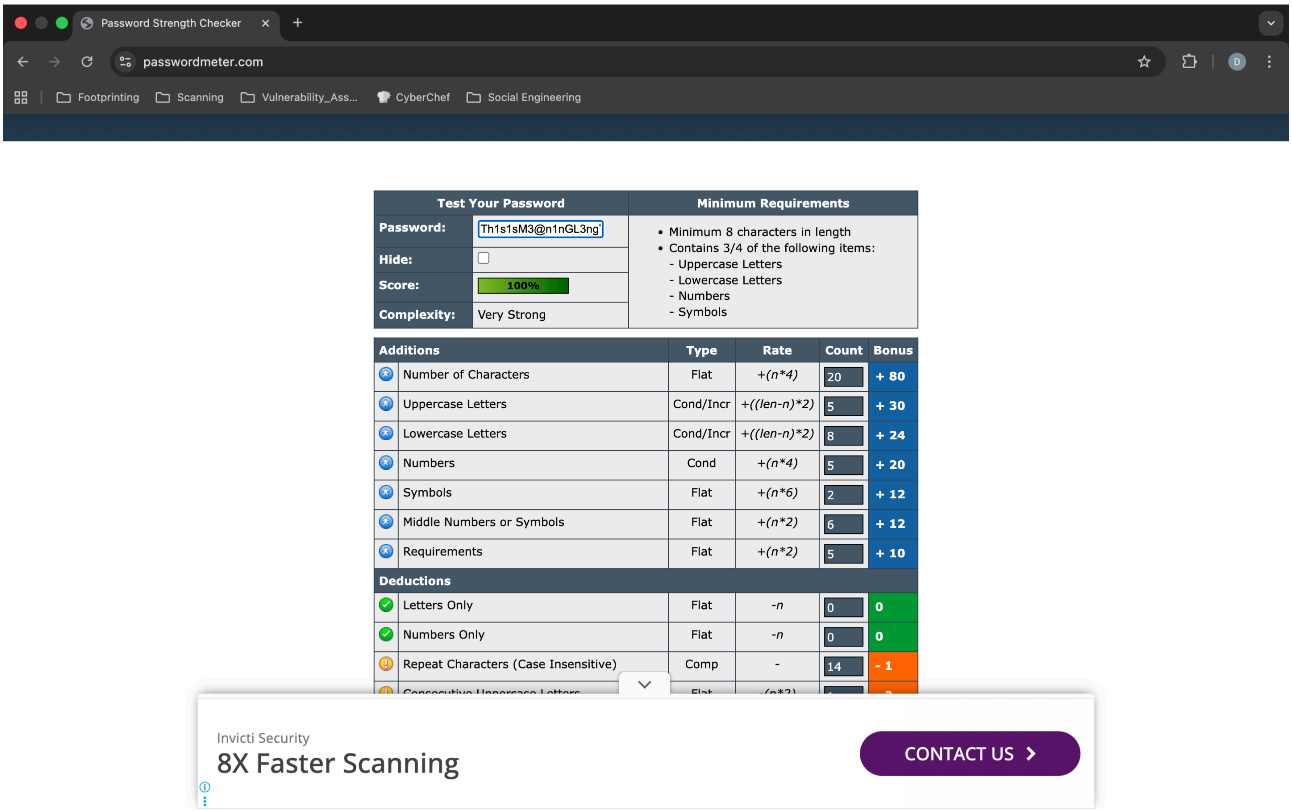
I. Password Testing and Simulated Evaluation

Five example passwords were created with varying levels of length and character set usage. These were assessed using a simulated password strength analysis to understand the direct correlation between complexity and security score.



Test Your Password		Minimum Requirements
Password:	MyDogFido	• Minimum 8 characters in length • Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	□	
Score:	38%	
Complexity:	Weak	

	Additions	Type	Rate	Count	Bonus
★	Number of Characters	Flat	+(n*4)	9	+ 36
★	Uppercase Letters	Cond/Incr	+((len-n)*2)	3	+ 12
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	6	+ 6
☒	Numbers	Cond	+(n*4)	0	0
☒	Symbols	Flat	+(n*6)	0	0
☒	Middle Numbers or Symbols	Flat	+(n*2)	0	0
☒	Requirements	Flat	+(n*2)	3	0
	Deductions				
!	Letters Only	Flat	-n	9	- 9
✓	Numbers Only	Flat	-n	0	0
!	Repeat Characters (Case Insensitive)	Comp	-	2	- 1
✓	Consecutive Uppercase Letters	Flat	-(n*2)		



Test Your Password		Minimum Requirements
Password:	Th1s1sM3@n1nGL3ng	• 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)	20	+ 80
★	Uppercase Letters	Cond/Incr	+((len-n)*2)	5	+ 30
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	8	+ 24
★	Numbers	Cond	+(n*4)	5	+ 20
★	Symbols	Flat	+(n*6)	2	+ 12
★	Middle Numbers or Symbols	Flat	+(n*2)	6	+ 12
★	Requirements	Flat	+(n*2)	5	+ 10
	Deductions				
✓	Letters Only	Flat	-n	0	0
✓	Numbers Only	Flat	-n	0	0
!	Repeat Characters (Case Insensitive)	Comp	-	14	- 1
!	Consecutive Uppercase Letters	Flat	-(n*2)		

Password	Length	Components Used	Simulated Score (Out of 100)	Simulated Feedback
password123	12	Lowercase, Numbers	43	**Extremely Weak:** Easily cracked; present in common dictionaries.
MyDogFido	9	Uppercase, Lowercase	38	**Very Weak:** Simple, common capitalization pattern.
Gr3at!W0rld!	12	Uppercase, Lowercase, Numbers, Symbols	100	**Strong:** Good mix of character types; moderate complexity.
P@w0rd!	9	Uppercase, Lowercase, Numbers, Symbols	66	**Weak:** Uses predictable leetspeak substitutions (@ for a, $ for s).
Th1s1sM3@n1nGL3ngTh!	22	Uppercase, Lowercase, Numbers, Symbols	100	**Excellent:** High entropy due to exceptional length and random character mix.

II. Best Practices and Evaluation Findings

The evaluation confirms that **password length** is the single most important factor contributing to high security scores, followed closely by the inclusion of diverse character sets (entropy).

A. Core Best Practices

- **Prioritize Length:** A minimum of **14 characters** is the modern standard, with **16 or more** being strongly recommended.
- **Maximize Complexity:** A strong password must combine all character sets:
    - **Uppercase Letters** (A-Z)
    - **Lowercase Letters** (a-z)
    - **Numbers** (0-9)
    - **Symbols** ($\text{!@\#\$%\text{&}\text{*}}$)
- **Ensure Randomness:** Avoid dictionary words, sequential patterns, keyboard patterns (qwerty), and personal information.

B. Tips Learned from Evaluation

- **Use Passphrases:** Create a long, complex, and memorable phrase instead of a short, difficult-to-remember password (e.g., MyB1rdSingsInTHeMorn1ng!).
- **Utilize a Password Manager:** For ultimate security, use a reputable password manager (e.g., 1Password, Bitwarden) to generate and store unique, high-entropy passwords for every account.
- **Avoid Reuse:** Never use the same password for more than one service.

III. Password Attacks and Security Conclusion

Understanding common attack vectors reinforces the necessity of adopting highly complex passwords.

A. Common Password Attacks

1. **Brute-Force Attack:**
   - ◦ *Method:* Systematically trying every possible character combination until the correct password is found.
   - ◦ *Defense:* **Length** is the key defense, as it exponentially increases the number of combinations, rendering the attack computationally infeasible.
2. **Dictionary Attack:**
   - ◦ *Method:* Using lists of common words, phrases, and previously exposed passwords to attempt login.
   - ◦ *Defense:* **Randomness and Complexity** are the key defenses, ensuring the password contains no recognizable words or easy substitutions.

B. Conclusion: Complexity and Security

Password complexity directly determines a password's **entropy** (randomness). High entropy increases the number of attempts required to crack the password, significantly reducing the probability of a successful compromise.

Factor	Effect on Security
High Entropy	Protects against both dictionary and brute-force attacks.
Low Entropy	Highly susceptible to automated cracking and guessing algorithms.