Task 6 – Strong Password Creation and Evaluation

Date: 12/08/2025
Name : Abinash I
Tools Used:

- Kali Linux `cracklib-check` (offline strength test)
- passwordmeter.com (online strength test)

Objective

To understand the characteristics of a strong password by creating multiple passwords, testing them with both offline and online tools, and comparing the results.

Passwords Tested & Results

Password	Length	cracklib-check Result	Passwordmeter.com Score
apple123	8	it is based on a dictionary word	36% – Weak
Apple123	8	it is based on a dictionary word	52% – Medium
Apple@123	9	OK	72% – Strong
ApPlE@2025!	11	OK	86% – Very Strong
M!cr0S3cUr!Ty#4098	17	OK	100% – Excellent

Observations

- **Both tools** identified short and dictionary-based passwords as weak.
- **Passwordmeter.com** gives a percentage score, while `cracklib-check` gives a pass/fail with reasons.
- Adding symbols, numbers, and mixed case improves strength in both tools.
- Long random passwords (16+ chars) scored the highest.

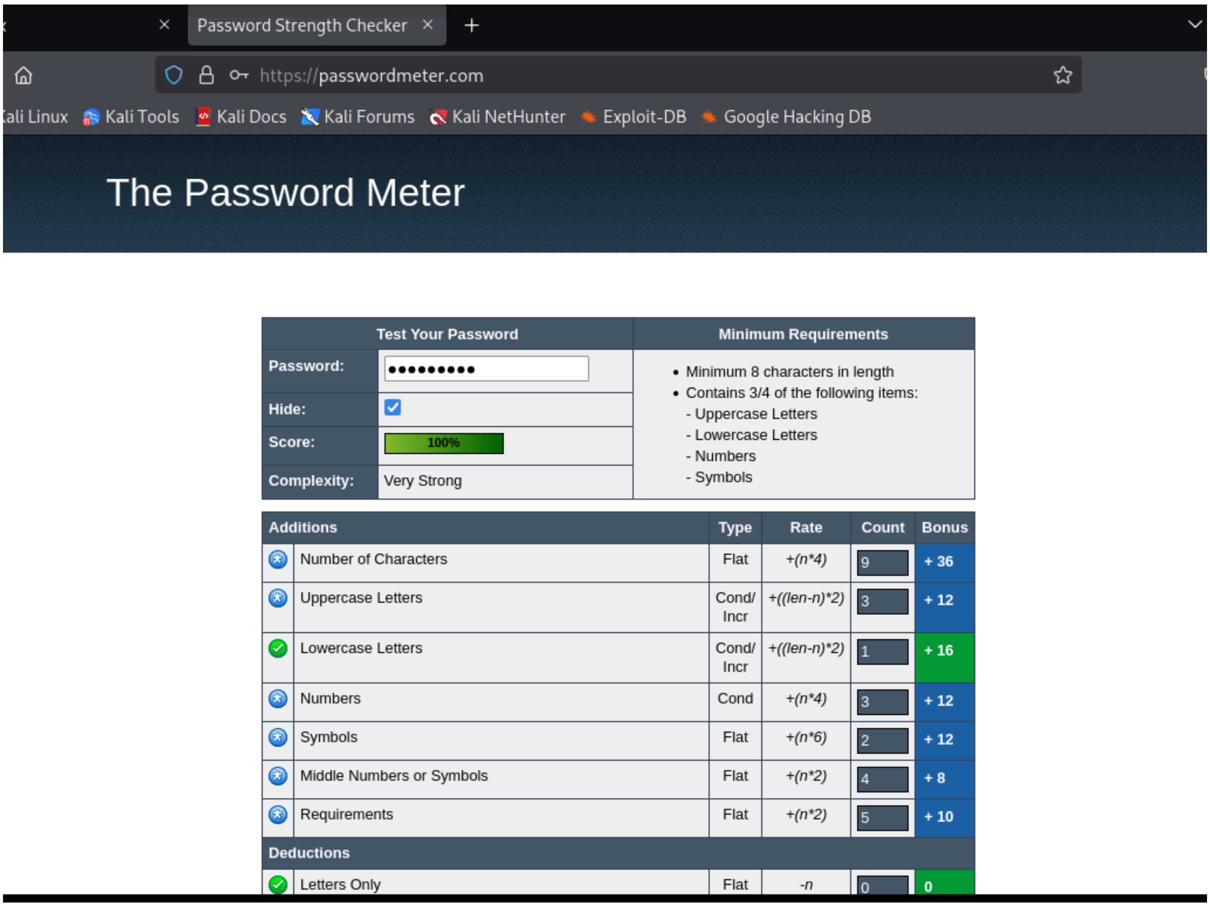
Best Practices Learned

1. Use at least **12–16 characters**.
2. Mix **uppercase, lowercase, numbers, and symbols**.
3. Avoid dictionary words, names, or predictable patterns.
4. Use **different passwords** for each account.
5. Consider **passphrases** for easier recall but strong complexity.

Tool – 1:

passwordmeter.com (online strength test)

- **Open browser** in Kali Linux → go to https://passwordmeter.com/.
- In the text box, type one password at a time:
  - `apple123`
  - `Apple123`
  - `Apple@123`
  - `ApPlE@2025!`
  - `M!cr0S3cUr!Ty#4098`
- For each password, note down:
  - **Score (%)** shown at the top.
  - **Complexity rating** (Weak, Strong, etc.).
  - **Additions & Deductions** feedback from the tool.



The Password Meter

Test Your Password		Minimum Requirements
Password:	●●●●●●●●●	• 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)	3	+ 12
Lowercase Letters	Cond/ Incr	+((len-n)*2)	1	+ 16
Numbers	Cond	+(n*4)	3	+ 12
Symbols	Flat	+(n*6)	2	+ 12
Middle Numbers or Symbols	Flat	+(n*2)	4	+ 8
Requirements	Flat	+(n*2)	5	+ 10
Deductions				
Letters Only	Flat	-n	0	0

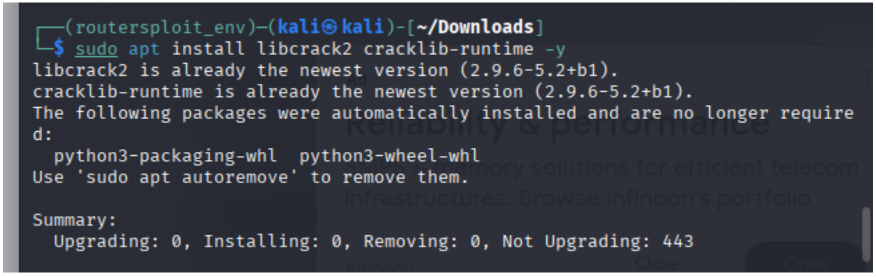
Tool – 2:

Kali Linux `cracklib-check` (offline strength test)

- Opened Kali Linux terminal.

- Installed cracklib:

```
sudo apt update
sudo apt install libcrack2 cracklib-runtime -y
```

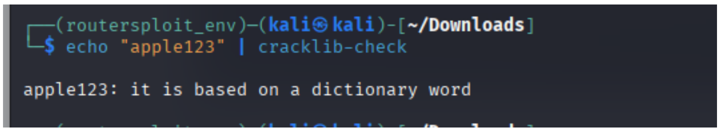


```
┌──(routersploit_env)-(kali㉿kali)-[~/Downloads]
└─$ sudo apt install libcrack2 cracklib-runtime -y
libcrack2 is already the newest version (2.9.6-5.2+b1).
cracklib-runtime is already the newest version (2.9.6-5.2+b1).
The following packages were automatically installed and are no longer require
d:
  python3-packaging-whl  python3-wheel-whl
Use 'sudo apt autoremove' to remove them.

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 443
```

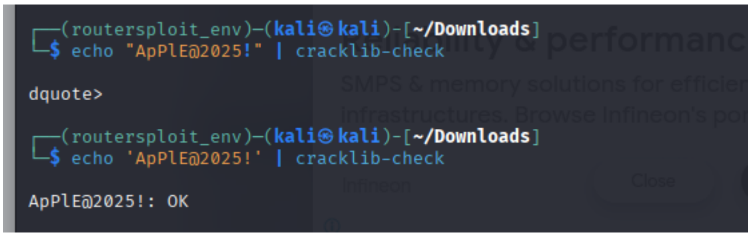
Tested each password:

```
echo "apple123" | cracklib-check
echo "Apple123" | cracklib-check
echo "Apple@123" | cracklib-check
echo "ApPlE@2025!" | cracklib-check
echo "M!cr0S3cUr!Ty#4098" | cracklib-check
```



```
┌──(routersploit_env)-(kali㉿kali)-[~/Downloads]
└─$ echo "apple123" | cracklib-check

apple123: it is based on a dictionary word
```



```
┌──(routersploit_env)-(kali㉿kali)-[~/Downloads]
└─$ echo "ApPlE@2025!" | cracklib-check

dquote>

┌──(routersploit_env)-(kali㉿kali)-[~/Downloads]
└─$ echo 'ApPlE@2025!' | cracklib-check

ApPlE@2025!: OK
```

- Passwords containing dictionary words like **"apple"** are flagged as weak, even if numbers are added.
- Adding symbols and increasing length improves the result to **OK**.
- Very long and complex passwords pass without warnings.

Common Password Attacks

- **Brute Force:** Tests all possible combinations until correct.
- **Dictionary Attack:** Uses common words/password lists.
- **Credential Stuffing:** Reuses leaked passwords from other breaches.

Conclusion

Using both offline (`cracklib-check`) and online (passwordmeter.com) tools shows that **password length and complexity are the most important factors** for security. Weak passwords can be cracked quickly; strong ones resist common attacks.