**PASSWORD STRENGTH ANALYZER WITH CUSTOM WORDLIST GENERATOR**

**Introduction:**

In today's cybersecurity world, protecting user credentials is crucial. Weak passwords are easy targets for attackers using brute-force or dictionary attacks. This project focuses on creating a simple tool that performs two key functions:

1. **Analyzing Password Strength**
2. **Generating Custom Wordlists using Crunch (in Kali Linux)**

This dual-purpose tool helps users create stronger passwords and assists penetration testers in crafting targeted password wordlists.

## Tools Used:

* **Kali Linux**
* **Python (for password strength analysis script)**
* **Crunch (for custom wordlist generation)**

## Steps Involved:

### **Part 1: Password Strength Analyzer**

* Checks password length
* Checks use of uppercase, lowercase, digits, special characters
* Estimates password entropy / strength level (Weak / Moderate / Strong / Very Strong)

**PYTHON SCRIPT:**

import re

def check\_password\_strength(password):

length\_error = len(password) < 8

digit\_error = re.search(r"\d", password) is None

uppercase\_error = re.search(r"[A-Z]", password) is None

lowercase\_error = re.search(r"[a-z]", password) is None

symbol\_error = re.search(r"[ !@#$%^&\*()\_+\-=\[\]{};':\"\\|,.<>\/?]", password) is None

errors = [length\_error, digit\_error, uppercase\_error, lowercase\_error, symbol\_error]

score = 5 - sum(errors)

if score == 5:

strength = "Very Strong"

elif score >= 4:

strength = "Strong"

elif score >= 3:

strength = "Moderate"

else:

strength = "Weak"

return strength

password = input("Enter Password to Analyze: ")

strength = check\_password\_strength(password)

print(f"Password Strength: {strength}")

### **Part 2: Custom Wordlist Generator using Crunch**

Crunch is a wordlist generation tool in Kali Linux. It allows you to generate password wordlists with specific patterns, length, and character sets.

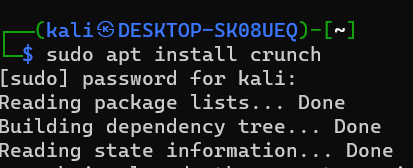
In order to crack a password, we have to try a lot of passwords to get the right one. When an attacker uses thousands or millions of words or character combinations to crack a password, there is no surety that any of those combinations will work. This collection of different combinations of characters is called a wordlist. And in order to crack a password or a hash, we need to have a good wordlist that could break the password. So to do so, we have a tool in kali Linux called crunch.

**STEPS TO CREATE CUSTOM WORDLISTS USING CRUNCH IN KALI LINUX**

**Step 1:**Installation

For Installation purposes run the following command on Terminal:-

sudo apt install crunch



**Step 2:**Creating Wordlist

crunch <min\_length> <max\_length> <characters> -o <output\_file>

1. **Generating a Simple Wordlist (Example: 4 to 6 characters, lowercase letters only):**

crunch 4 6 abcdefghijklmnopqrstuvwxyz -o custom\_wordlist.txt

1. **Generating a Complex Wordlist (Example: Including numbers and symbols):**

crunch 6 8 abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789!@# -o complex\_wordlist.txt

1. **Generating Wordlist with Known Prefix/Suffix Pattern:**

crunch 8 8 -t abc%%%%% -o prefix\_wordlist.txt

### **Final Combined Workflow:**

* User inputs password
* Python script analyzes strength
* If password is weak, suggest using Crunch to create stronger password ideas or generate wordlists for testing
* Run Crunch command based on user preference

Password Strength Analyzer Tools:

| **Tool** | **Type** | **Platform** | **Main Use** |
| --- | --- | --- | --- |
| John the Ripper | CLI | Linux/Win | Password cracking & audit |
| Hydra | CLI | Linux/Win | Brute-force login testing |
| Cracklib-check | CLI | Linux | Dictionary weakness check |
| pwstrength lib | API/Script | Linux | Script-based strength check |
| Zxcvbn | API/Script | All | Real-world strength estimator |
| Custom Python Tool | CLI | All | User-defined analysis |
| Online Checkers | Web-based | Any browser | Quick strength estimation |

## Conclusion:

By integrating a **Password Strength Analyzer (Python)** with a **Custom Wordlist Generator (Crunch)**, this tool provides both preventive (creating strong passwords) and offensive (for pentesting and security audits) capabilities.