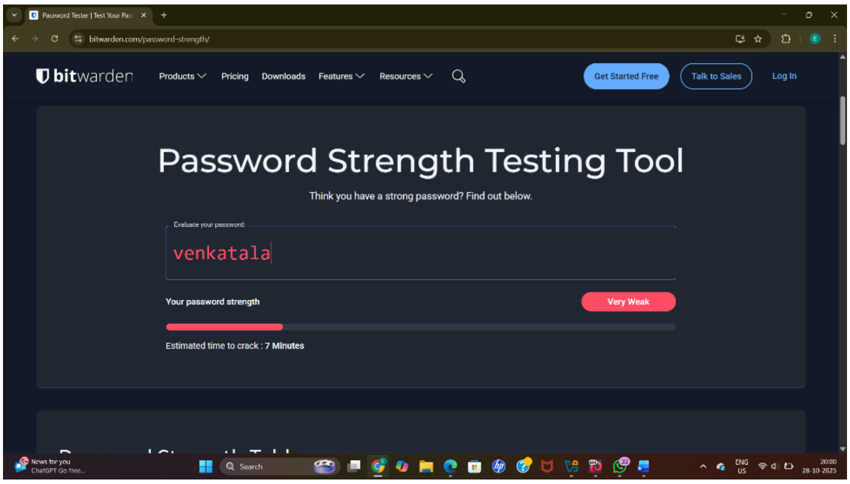
Task 6 : Create a Strong Password and Evaluate Its Strength

1. Create multiple passwords with varying complexity.
2. Use uppercase, lowercase, numbers, symbols, and length variations.
3. Test each password on password strength checker.
4. Note scores and feedback from the tool.
5. Identify best practices for creating strong passwords.
6. Write down tips learned from the evaluation.
7. Research common password attacks (brute force, dictionary).
8. Summarize how password complexity affects security.

- I have tried checking the strength of password's using website "bitwarden".
- Refer the following images where it tells the strength of each password.
- The website is analysing based on length of the password.



Password Strength Testing Tool

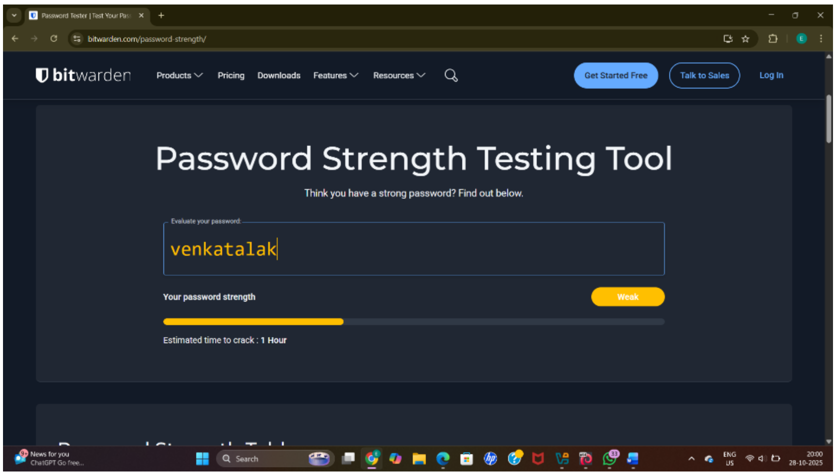
Think you have a strong password? Find out below.

Evaluate your password:

venkatala

Your password strength: Very Weak

Estimated time to crack : 7 Minutes



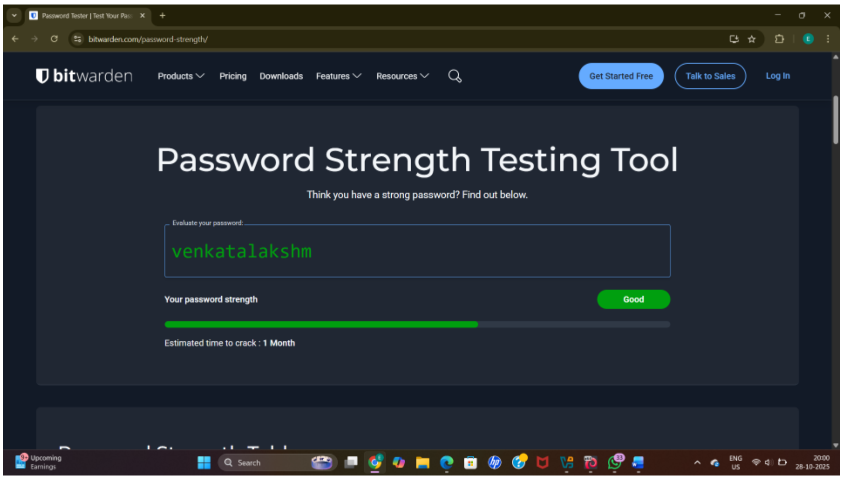
Password Strength Testing Tool

Think you have a strong password? Find out below.

Evaluate your password: venkatalak

Your password strength: Weak

Estimated time to crack : **1 Hour**



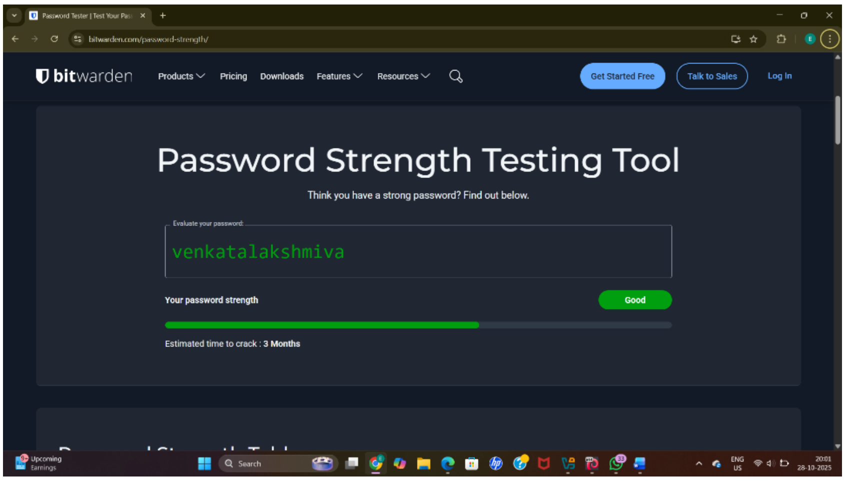
Password Strength Testing Tool

Think you have a strong password? Find out below.

Evaluate your password: venkatalakshm

Your password strength: Good

Estimated time to crack : **1 Month**



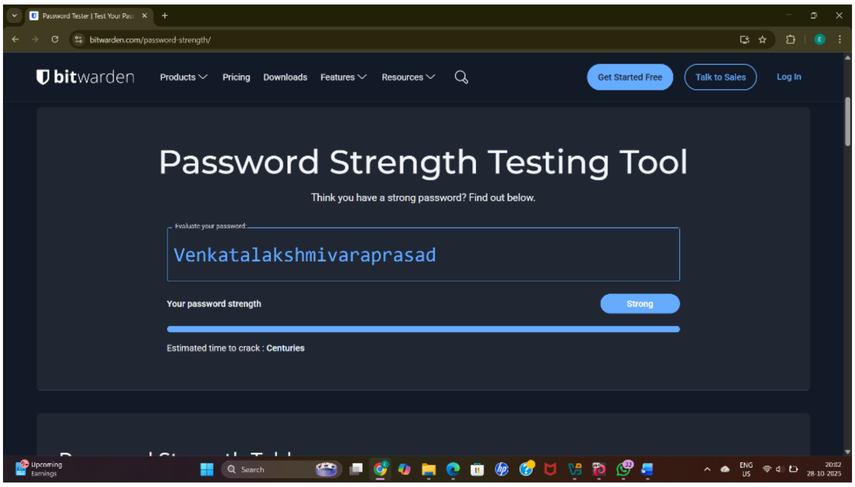
Password Strength Testing Tool

Think you have a strong password? Find out below.

Evaluate your password: venkatalakshmiva

Your password strength: Good

Estimated time to crack : **3 Months**



Password Strength Testing Tool

Think you have a strong password? Find out below.

Evaluate your password: Venkatalakshmivaraprasad

Your password strength: Strong

Estimated time to crack : **Centuries**

Research common password attacks (brute force, dictionary):

- **Brute-force attack:** trying every possible combination of characters until the correct password is found. This is simple but costly in time unless the password is short or the attacker has massive compute.
- **Dictionary attack**: tries passwords from a precompiled list of likely passwords (words, leaked passwords, common variations). Much faster than pure brute force because it tests only likely candidates. Modern dictionaries include huge collections of leaked passwords and common substitutions.
- **Password spraying**: attacker tries a small set of common passwords across many accounts to avoid triggering per-account lockouts.
- **Credential stuffing**: attacker uses username/password pairs obtained from other breaches, they replay those credentials on your site. This relies on password reuse.

Best practices for creating strong passwords:

- Use 12–16 characters or more to strengthen resistance against attacks.
- Include: Uppercase, Lowercase, Numbers, Special characters
- Avoid these: Don’t use personal info (name, DOB), Avoid common words like password, 123456, or qwerty.
- Create a Passphrase:

- Use a meaningful phrase that’s easy to remember but hard to guess. Example:IH@teDr!nkingcoffee!
- Don’t reuse passwords
- Each account should have its own unique password.
- Update When Needed
- Change passwords if compromised or after a data breach.
- Use a Password Manager
- Let secure tools like Bitwarden, 1Password, or LastPass store and generate strong passwords.
- Turn On Multi-Factor Authentication (MFA)
- Add extra protection using OTPs, biometrics, or security keys.

Summarize how password complexity affects security:

- **Increases possible combinations**: Each added character type exponentially increases the total number of guesses an attacker must try.
- **Harder for dictionary attacks**: Complex passwords aren’t found in common wordlists or leaked-password databases.
- **Slows brute-force attacks**: Attackers need far more attempts to find the right match.