What is password cracking?

Password cracking is recovering a password from its stored (hashed) form, usually offline after an attacker steals a password database. Techniques range from brute force and dictionary attacks to GPU-accelerated, rule-based guessing. Weak, reused, or poorly hashed passwords fall in seconds. The defenses are long unique passwords, modern slow hashing with unique salts, and multi-factor authentication so a cracked password is not enough.

HOW IT WORKS

01 The main techniques

Attackers rarely guess blindly:

- Brute force tries every combination, feasible only for short passwords.
- Dictionary attacks try known words and common passwords first.
- Rule and mask attacks mutate dictionary words the way people do (Password becomes P@ssw0rd!), which is devastatingly effective.
- Hybrid attacks combine dictionaries with brute force on the ends.
- Rainbow tables use precomputed hashes, which is exactly why unique salts exist to defeat them.

Modern GPUs test billions of guesses per second against fast hashes, so "complex-looking" is not the same as "strong".

02 Why how you store passwords decides who wins

The single biggest factor in whether stolen passwords get cracked is how they were hashed:

- Fast, unsalted hashes (MD5, SHA-1, plain SHA-256) fall almost instantly on modern hardware.
- Slow, salted, purpose-built hashes (bcrypt, scrypt, Argon2) are designed to be expensive to compute, turning billions of guesses per second into thousands.
- A unique salt per password stops one cracked hash from revealing every user who chose the same password, and kills rainbow tables.

HOW TO DEFEND

- Long, unique passwords in a password manager, so length beats guessing and one leak does not open other accounts.
- Modern slow hashing with unique salts for any passwords you store.
- Multi-factor authentication, so even a cracked password is not enough to log in.
- Block known-breached passwords at sign-up and reset.
- Rate-limit and lock out online guessing, and monitor for attempts to dump password stores.

SOURCES

[1] Brute Force (Technique T1110)

[2] Digital Identity Guidelines (SP 800-63B)

[3] Password Storage Cheat Sheet

If you build software, this is the control that matters most. Use a modern slow algorithm with unique salts, and a stolen database buys the attacker far less.

Find out whether your passwords, and your hashes, would hold.

securelayer7.net/learn/threats/password-cracking

Open online