What is Hashcat?

Hashcat is an open-source, GPU-accelerated password-cracking tool that recovers cleartext passwords from hashes by trying candidates and comparing the result. It supports hundreds of hash types via mode numbers (1000 = NT hash, 1800 = SHA-512crypt, 5600 = NetNTLMv2, 13100 = Kerberoast). Attackers feed it the hashes they dumped, then use wordlists, rules, masks, and brute force to crack weak ones in seconds to hours. It is why a dumped hash of a weak password is as good as the password.

HOW IT WORKS

01 How it is used and payload

After dumping hashes, the attacker picks the mode and an attack:

- NT hashes from the SAM/NTDS: `hashcat -m 1000 nt.txt rockyou.txt -r best64.rule`
- NetNTLMv2 from LLMNR poisoning: `hashcat -m 5600 net.txt rockyou.txt`
- Kerberoast tickets: `hashcat -m 13100 spn.txt wordlist`
- Mask brute force for short passwords: `hashcat -m 1000 nt.txt -a 3 ?u?l?l?l?l?d?d`

Weak and reused passwords fall quickly. Documented for defensive context.

> SPEED IS THE POINT
>
> *On a GPU, Hashcat tries billions of guesses per second against fast, unsalted hashes like NT. Length, not complexity, is what defeats it: long passphrases push cracking out of reach.*

HOW TO DEFEND

- Enforce long passwords/passphrases (length beats complexity against cracking).
- Block common and breached passwords so wordlist attacks fail.
- Use slow, salted hashing for any passwords you store (bcrypt/argon2 for apps; for Windows, reduce NTLM exposure).
- Protect the hashes in the first place (limit local admin, Credential Guard) so there is nothing to crack.
- Detect the dumping that precedes cracking, since cracking itself is offline and invisible.

SOURCES

[1] MITRE ATT&CK: Credential Access (TA0006)

[2] NIST SP 800-63B Digital Identity Guidelines

[3] MITRE ATT&CK: OS Credential Dumping (T1003)

Find the exposed credentials before an attacker does.

securelayer7.net/learn/credential-access/what-is-hashcat

Open online