How DNS Works (and How Hackers Poison It)

Quick reference cheat sheet · Episode 002 · Net+ / Sec+

1. Caches	Browser checks its own cache, then the OS cache. A hit ends the lookup instantly; only a double miss goes to the network.	browser cache, OS cache
2. Resolver	A recursive resolver (your ISP's, or 8.8.8.8) agrees to do the whole lookup and return one final answer.	recursive resolver, 8.8.8.8
3. Hierarchy	Resolver walks root -> TLD (.net) -> authoritative. Root points to TLD, TLD points to authoritative, authoritative holds the real record.	root, TLD, authoritative, A record
4. Caching / TTL	The answer is cached for a TTL set by the domain owner, so the next lookup skips the walk. A poisoned entry also lasts the whole TTL.	TTL, cache
5. Trust flaw	Classic DNS rides UDP and matches a reply by a single 16-bit transaction ID. It trusts the FIRST reply with the matching ID; it never checks who sent it.	UDP, transaction ID (16-bit)
6. Cache poisoning	A race: flood forged replies guessing the ID, beat the real server, and the resolver caches the lie. Kaminsky 2008 (CVE-2008-1447) used random subdomains for unlimited fresh races.	cache poisoning, Kaminsky, CVE-2008-1447
7. Defenses	Source-port randomization = guess ID AND port (65k -> billions), kills the easy attack. DNSSEC signs records so forgeries fail (the only full fix), but adoption is low.	source port, DNSSEC

Acronym key

DNS name to IP system

TTL how long an answer is cached

UDP fast, no-handshake transport

TLD top-level domain server (.net)

resolver does the full lookup for you

authoritative holds the real record

transaction ID 16-bit reply matcher

cache poisoning forged answer in the cache

Kaminsky 2008 reliable poisoning

CVE-2008-1447 the Kaminsky bug

source port extra entropy vs guessing

DNSSEC signed records, only full fix