What is DLL hijacking?

DLL hijacking is a Windows technique where an attacker abuses the DLL search order to make a program load a malicious library instead of the intended one. If a privileged program (a service or an elevated application) looks for a DLL in a writable directory or fails to specify a full path, an attacker places a malicious DLL of the right name there and their code runs with the program's privileges, often SYSTEM. The fix is loading DLLs from fixed, protected paths.

HOW IT WORKS

01 The abuse and payload

The attacker finds a privileged program that loads a DLL from a writable or missing location, then plants one:

- Identify the missing or hijackable DLL (procmon-style analysis shows "NAME NOT FOUND" DLL lookups in writable paths).
- Build a malicious DLL whose `DllMain` runs the payload: `msfvenom -p windows/x64/exec CMD="cmd.exe" -f dll -o hijack.dll`
- Place it where the program searches first and trigger a load (restart the service or app).
- The code runs as the program's account, often SYSTEM.

Documented technique shown for defenders.

> TWO INGREDIENTS
>
> *DLL hijacking needs a privileged program that loads a DLL from a writable or missing path. Remove either, fixed paths or no write access, and it fails.*

HOW TO DEFEND

- Have applications load DLLs from fixed, protected paths and use safe loading APIs (full paths, `LoadLibraryEx` flags).
- Remove write access from application and service directories for non-admin users.
- Keep software patched, since vendors fix hijackable load behaviour.
- Use application control (allow-listing) to block unexpected DLLs.
- Monitor for DLLs appearing in program directories and unusual module loads.

SOURCES

[1] Microsoft: Service Security and Access Rights

[2] MITRE ATT&CK: Privilege Escalation (TA0004)

[3] NIST SP 800-115 Technical Guide to Security Testing

Find the privilege-escalation paths before an attacker does.

securelayer7.net/learn/privilege-escalation/what-is-dll-hijacking

Open online