Detecting Electron Application Abuse for Proxy Execution, Detection Strategy DET0025

Archived: 2026-04-05 18:03:57 UTC

AN0071

Abuse of trusted Electron apps (Teams, Slack, Chrome) to spawn child processes or execute payloads via malicious command-line arguments (e.g., --gpu-launcher) and modified app resources (.asar). Behavior chain: suspicious parent process (Electron app) → unusual command-line args → child process creation → optional DLL/network artifacts.

Log Sources

Mutable Elements

Field	Description
TimeWindow	Correlation window tying app launch, file tampering, child process, and network events (5–10 minutes typical).
UserContext	Flag admin/service accounts versus standard users executing Electron apps.
AllowedElectronApps	Baseline of Electron-based executables expected in the enterprise.
AllowedChildProcesses	Whitelist normal child processes (chrome.exe → crashpad_handler.exe) versus anomalies (powershell.exe).
ElectronAppDomainAllowlist	Approved service domains for Teams, Slack, etc. to suppress benign traffic.
AsarIntegrityHash	Expected hash/signature of app.asar resources to detect tampering.

AN0072

Abuse of Linux Electron binaries by modifying app.asar or config JS files and spawning unexpected child processes (bash, curl, python).

Log Sources

Mutable Elements

Field	Description
AsarIntegrityCheck	Baseline of expected asar package signatures per app.

Field	Description
SuspiciousChildProcesses	Flag shells/python spawned from Electron parent.

AN0073

Abuse of macOS Electron apps by modifying app.asar bundles and spawning child processes (osascript, curl, sh) from Electron executables.

Log Sources

Mutable Elements

Field	Description
AllowedAppBundlePaths	Baseline of legitimate Electron app paths under /Applications.
SignedToUnsignedTransition	Alert when signed Electron parent spawns unsigned child.
