

# Detection of Spoofed User-Agent, Detection Strategy DET0898

Archived: 2026-04-05 14:49:06 UTC

## AN2029

Process execution without GUI context (e.g., powershell.exe, wscript.exe) generates HTTP traffic with a spoofed User-Agent mimicking a legitimate browser. No corresponding UI application (e.g., msedge.exe) is active or in parent lineage. The User-Agent deviates from known enterprise baselines or contains spoofed platform indicators. User-Agent strings can be gathered with API calls such as `ShellExecuteW` to open the default browser on a socket to receive an HTTP reply, or by hard coding the User-Agent string for a specific browser.

### Log Sources

### Mutable Elements

Field	Description
HeaderSignatureMatch	Specific HTTP header anomalies or patterns (e.g., spoofed User-Agent).
UserAgentFingerprint	Flag browser-based sessions
NonBrowserProcessList	List of non-browser binaries expected not to initiate web requests (e.g., powershell.exe, cscript.exe)

## AN2031

Detection of HTTP outbound requests with inconsistent or spoofed User-Agent headers from command-line tools (e.g., curl, wget, python requests) following interactive user shells or scheduled jobs outside of normal user session behavior.

### Log Sources

### Mutable Elements

Field	Description
HeaderSignatureMatch	Specific HTTP header anomalies or patterns (e.g., spoofed User-Agent).
UserAgentFingerprint	Flag browser-based sessions

## AN2032

Observation of scripted network requests (e.g., using `osascript`, `curl`, or `python`) that include mismatched or spoofed browser User-Agent strings compared to the typical macOS Safari or Chrome baseline, especially when triggered by non-interactive launch agents, login hooks, or background daemons.

**Log Sources**

**Mutable Elements**

Field	Description
UserAgentFingerprint	Flag browser-based sessions
HeaderSignatureMatch	Specific HTTP header anomalies or patterns (e.g., spoofed User-Agent).

---

Source: <https://attack.mitre.org/detectionstrategies/DET0898#AN2031>