macOS Persistence Checklist

v1.0 (20/04/2026)

macOS persistence mechanisms, ordered by frequency in real-world malware. Each entry includes the exact paths, plists, and commands needed to verify it. Legend: ✓ = checked by osquery, ⚙ = requires osquery configuration, ✗ = not checked, unless you're a forensic analyst. Tags: Persistence = finding implies persistence, Execution = runs on login, Hardening = finding implies a misconfiguration, Recon = useful for enumeration, Lateral = enables lateral movement.

Mechanism

The search bar and legend filter apply here too, so the rows stay in sync with the checklist.

Core Launch & Login Persistence

Launch Agents — ✓ / ✓ — Persistence, Execution
Per-user daemons that run when a user logs in.
`~/Library/LaunchAgents`
`/Library/LaunchAgents`

Launch Daemons — ✓ / ✓ — Persistence, Execution
Root-level daemons that start at boot, before any user logs in.
`/Library/LaunchDaemons`

launchd overrides / disabled — ✓ / ✓ — Persistence, Hardening
Disabled services can be forced back on by a single command.
`/private/var/db/com.apple.xpd.plist`

BTM / SMAppService — ✓ / ✗ — Persistence, Execution
Modern Ventura+ Background Task Management.
`/private/var/db/com.apple.backgroundtaskmanagement/BackgroundItems-v4.btm`

Login Items (legacy / classic) — ✓ / ✗ — Persistence, Execution
Old-style login items.
`~/Library/LoginItems`

Embedded Login Helpers — ✓ / ✗ — Persistence, Execution
Helper apps bundled inside other apps, launched at login.
`/Applications/*/Contents/Library/LoginItems`

Privileged Helper Tools — ✓ / ✗ — Persistence, Execution
Root helpers installed via SMJobBless.
`/Library/PrivilegedHelperTools`

Authorization & Access Persistence

SSH authorized_keys — ✓ / ✗ — Persistence, Lateral
Grants remote access without password.
`~/.ssh/authorized_keys`
`/etc/ssh/sshd_config`

TCC (privacy permissions) — ✓ / ✗ — Persistence, Recon
Grants apps access to camera, mic, screen, files, etc.
`/Library/Application Support/com.apple.TCC/TCC.db`
`~/Library/Application Support/com.apple.TCC/TCC.db`

Authorization DB — ✓ / ✗ — Persistence, Hardening
Controls who can install privileged helpers.
`/Library/Security/authorization.plist`

PAM modules — ✓ / ✗ — Persistence, Hardening
Can intercept authentication system-wide.
`/etc/pam.d/*`
`/Library/Security/pam_appsec/*`

sudoers / sudo policy — ✓ / ✗ — Persistence, Hardening
Controls who can run commands as root.
`/etc/sudoers`
`/etc/sudoers.d/*`

Authorization Plug-ins — ✓ / ✗ — Persistence, Hardening
Legacy login window plug-ins.
`/Library/Security/SecurityAgentPlugins/*`

Local accounts / Groups — ✓ / ✗ — Persistence, Recon
Hidden accounts are a classic persistence trick.
`/Local/Default/Users`
`/Local/Default/Groups`

Shell & Scheduled Execution

Shell rc/profiles — ✓ / ✗ — Persistence, Execution
Runs on every new shell.
`~/.zshrc`
`~/.zprofile`
`~/.bash_profile`
`/etc/zshrc`
`/etc/zprofile`

Shell init (bash / zsh) — ✓ / ✗ — Persistence, Execution
Runs on every new shell.
`~/.bashrc`
`/etc/bashrc`

Periodic jobs — ✓ / ✗ — Persistence, Execution
Runs daily / weekly / monthly.
`/etc/periodic/{daily,weekly,monthly}/*`

Cron — ✓ / ✗ — Persistence, Execution
Classic scheduled execution.
`/private/var/at/tab/*`

Emond — ✓ / ✗ — Persistence, Execution
Legacy event-driven daemon.
`/private/var/db/emondClients/*`
`/private/var/db/emond.plist`

Extensions, Browser, Enterprise, and Handler Software

System Extensions — ✓ / ✗ — Persistence, Execution
`/Library/SystemExtensions/*`

Kext — ✓ / ✗ — Persistence, Hardening
`/Library/Extensions/*`
`/System/Library/Extensions/*`
`/Library/StagedExtensions/*`

Chromium Extensions — ✓ / ✗ — Persistence, Execution
`~/Library/Application Support/Google/Chrome/*/Extensions/*`

Firefox Extensions — ✓ / ✗ — Persistence, Execution
`~/Library/Application Support/Firefox/Profiles/*/extensions/*`

Safari App Extensions — ✓ / ✗ — Persistence, Execution
`~/Library/Safari/Extensions/*`

Safari Web Extensions — ✓ / ✗ — Persistence, Execution
`~/Library/Safari/WebExtensions/*`

Finder Sync Extensions — ✓ / ✗ — Persistence, Execution
`~/Library/Application Support/FinderSync/Extensions/*`

Application Support — ✓ / ✗ — Persistence, Execution
`~/Library/Application Support/*`

App preference hijacking — ✗ / ✗ — Persistence, Execution
`~/Library/Preferences/*.plist`

Handler Software — ✓ / ✗ — Persistence, Execution
`~/Library/Preferences/com.apple.LaunchServices/com.apple.launchservices.secure.plist`
`/Applications/*/Contents/Info.plist`

Persistence Persistence Persistence

Persistence Persistence Persistence

Persistence Persistence Persistence

Mechanism	Required privilege	Source of logs	Collection/Triggs	Trigger	What to review
Shell Startup Scripts (T1037.004)	Root	Sysmon	Archive/Hash /etc/profile /etc/profile.d/* /etc/zshenv /etc/zprofile /etc/zshrc /etc/zlogin /etc/zlogout ~/.zshenv ~/.zprofile ~/.zshrc ~/.zlogin ~/.zlogout /usr/local/etc/zshenv /usr/local/etc/zprofile /usr/local/etc/zshrc /usr/local/etc/zlogin /usr/local/etc/zlogout Content /etc/ld.so.preload /etc/ld.so.conf.d/* Commands ldconfig Sysmon Event ID 7 (Image loaded)	Logon shell initialization shell (login)	Baseline hidden root-level persistence. Malicious entries in /etc/profile.d/ or /etc/zshenv run silently for every user. Verify ldconfig runs only when /etc/ld.so.conf.d/ changes. Hash-monitor /etc/ld.so.preload — any non-empty entry is suspicious.
Shell Init (zsh) (T1546.004)	Root	Sysmon	Archive/Hash ~/.zshenv ~/.zprofile ~/.zshrc ~/.zlogin ~/.zlogout /usr/local/etc/zshenv /usr/local/etc/zprofile /usr/local/etc/zshrc /usr/local/etc/zlogin /usr/local/etc/zlogout Content ~/.zsh_history ~/.zsh_logout ~/.zsecrets ~/.zsh_history ~/.zprofile Commands zsh -c zsh -i Sysmon Event ID 1 (Process Creation)	Logon shell initialization shell	Baseline hidden root-level persistence. Malicious entries in /etc/profile.d/ or /etc/zshenv run silently for every user. Verify ldconfig runs only when /etc/ld.so.conf.d/ changes. Hash-monitor /etc/ld.so.preload — any non-empty entry is suspicious.
Cron (T1053.003)	Root	Sysmon	Archive/Hash /etc/crontab /etc/cron.hourly/* /etc/cron.daily/* /etc/cron.weekly/* /etc/cron.monthly/* /etc/cron.d/* /var/spool/cron/crontabs/* Commands crontab -l crontab -u Sysmon Event ID 1 (Process Creation)	Scheduled/periodic	Baseline hidden root-level persistence. Malicious entries in /etc/profile.d/ or /etc/zshenv run silently for every user. Verify ldconfig runs only when /etc/ld.so.conf.d/ changes. Hash-monitor /etc/ld.so.preload — any non-empty entry is suspicious.
AC/Lotus (T1546.002)	Root	Sysmon	Archive/Hash /var/spool/cron/crontabs/* Commands /usr/bin/crontab Sysmon Event ID 1 (Process Creation)	One-shot initialization	Baseline hidden root-level persistence. Malicious entries in /etc/profile.d/ or /etc/zshenv run silently for every user. Verify ldconfig runs only when /etc/ld.so.conf.d/ changes. Hash-monitor /etc/ld.so.preload — any non-empty entry is suspicious.
Periodic Jobs (T1546.003)	Root	Sysmon	Archive/Hash /etc/periodic/daily/* /etc/periodic/weekly/* /etc/periodic/monthly/* /usr/local/etc/periodic/daily/* /usr/local/etc/periodic/weekly/* /usr/local/etc/periodic/monthly/* /usr/local/bin/periodic /usr/bin/periodic Sysmon Event ID 1 (Process Creation)	Daily / weekly / monthly maintenance	Baseline hidden root-level persistence. Malicious entries in /etc/profile.d/ or /etc/zshenv run silently for every user. Verify ldconfig runs only when /etc/ld.so.conf.d/ changes. Hash-monitor /etc/ld.so.preload — any non-empty entry is suspicious.
Calendar Alerts/EventKit (T1053.005)	Root	Sysmon	Archive/Hash ~/Library/Calendars/* ~/Library/Preferences/com.apple.ical.plist Commands /usr/bin/calendar Sysmon Event ID 1 (Process Creation)	Event alert/reminder	Baseline hidden root-level persistence. Malicious entries in /etc/profile.d/ or /etc/zshenv run silently for every user. Verify ldconfig runs only when /etc/ld.so.conf.d/ changes. Hash-monitor /etc/ld.so.preload — any non-empty entry is suspicious.
Emond (T1546.014)	Root	Sysmon	Archive/Hash /private/var/db/emond/* /System/Library/LaunchDaemons/com.apple.emond.plist Commands /usr/sbin/emond Sysmon Event ID 1 (Process Creation)	Event-driven daemon, with payload	Baseline hidden root-level persistence. Malicious entries in /etc/profile.d/ or /etc/zshenv run silently for every user. Verify ldconfig runs only when /etc/ld.so.conf.d/ changes. Hash-monitor /etc/ld.so.preload — any non-empty entry is suspicious.

Extensions, Browser, Interpreter, and Handler Surfaces

Only application-level persistence is discovered in unprivileged extension-store logs (Chrome, Edge, Firefox, Safari, VS Code, JetBrains, JupyterLab, Jupyter, Zeppelin, Thunderbird, Brave, Opera, Vivaldi, Arc, Cursor, Windsurf, Zed, Sublime Text, Notepad++, Obsidian, Raycast, Alfred, Karabiner, Hammerspoon, Spotlight, Quicksilver, Launchbar, Keyboard Maestro, Shortcuts, Automator, Terminal, iTerm, Ghostty, Kitty, Alacritty, Wezterm, Warp, Hyper, Tabby, Mailspring, Apple Mail, Outlook, Thunderbird, Slack, Discord, Zoom, Teams, Spotify, Figma, Notion

Application / deamon plug-ins

Linux | Mac | Windows

Windows: Application

Modifying Finder Sync plug-ins

Linux | Mac | Windows | Application

Mac: Application

Windows: Application

Linux

Sysfs Hijacking / Persisting

Linux | **Persist** | **Sysfs**

Prerequisites: root privileges

Detection: Monitor /sys/class/net/ for new entries; watch for udev rules in /etc/udev/rules.d/ and /lib/udev/rules.d/ that reference network interfaces

Mitigation: Restrict write access to /sysfs; use kernel lockdown mode; monitor udev rule changes

Sysfs Hijacking / Persisting

Persist | **Sysfs**

Prerequisites: root privileges

Detection: Monitor /sys/class/net/ for new entries; watch for udev rules in /etc/udev/rules.d/ and /lib/udev/rules.d/ that reference network interfaces

Mitigation: Restrict write access to /sysfs; use kernel lockdown mode; monitor udev rule changes

Other sysfs Persist / Hijack

Persist | **Sysfs**

Prerequisites: root privileges

Detection: Monitor /sys/class/net/ for new entries; watch for udev rules in /etc/udev/rules.d/ and /lib/udev/rules.d/ that reference network interfaces

Mitigation: Restrict write access to /sysfs; use kernel lockdown mode; monitor udev rule changes

Sysfs Hijacking / Persisting

Detection: Monitor /sys/class/net/ for new entries; watch for udev rules in /etc/udev/rules.d/ and /lib/udev/rules.d/ that reference network interfaces

Mitigation: Restrict write access to /sysfs; use kernel lockdown mode; monitor udev rule changes

Login Hooks

Linux | **Persist** | **Shell**

Prerequisites: root, kernel 5.19+

Detection: Monitor /etc/login.hooks and /etc/logout.hooks for unexpected entries; auditd -w /etc/login.hooks -p wa

Mitigation: Restrict write access to /etc/login.hooks; use kernel lockdown mode; monitor /etc/login.hooks with auditd

Logon Hooks

Persist | **Shell**

Prerequisites: root, kernel 5.19+

Detection: Monitor /etc/login.hooks and /etc/logout.hooks for unexpected entries; auditd -w /etc/login.hooks -p wa

Mitigation: Restrict write access to /etc/login.hooks; use kernel lockdown mode; monitor /etc/login.hooks with auditd

Logout Hooks

Persist | **Shell**

Prerequisites: root, kernel 5.19+

Detection: Monitor /etc/login.hooks and /etc/logout.hooks for unexpected entries; auditd -w /etc/login.hooks -p wa

Mitigation: Restrict write access to /etc/login.hooks; use kernel lockdown mode; monitor /etc/login.hooks with auditd

Logon Hooks

Detection: Monitor /etc/login.hooks and /etc/logout.hooks for unexpected entries; auditd -w /etc/login.hooks -p wa

Mitigation: Restrict write access to /etc/login.hooks; use kernel lockdown mode; monitor /etc/login.hooks with auditd

Startup Items

Linux | **Persist** | **Boot**

Prerequisites: root

Detection: Monitor /Library/StartupItems/ and /Library/LaunchDaemons/ for unexpected entries

Mitigation: Restrict write access to /Library/StartupItems/; use System Integrity Protection (SIP); monitor launchd plist changes

Startup Items

Persist | **Boot**

Prerequisites: root

Detection: Monitor /Library/StartupItems/ and /Library/LaunchDaemons/ for unexpected entries

Mitigation: Restrict write access to /Library/StartupItems/; use System Integrity Protection (SIP); monitor launchd plist changes

Boot

Persist | **Boot**

Prerequisites: root

Detection: Monitor /Library/StartupItems/ and /Library/LaunchDaemons/ for unexpected entries

Mitigation: Restrict write access to /Library/StartupItems/; use System Integrity Protection (SIP); monitor launchd plist changes

Startup Items

Detection: Monitor /Library/StartupItems/ and /Library/LaunchDaemons/ for unexpected entries

Mitigation: Restrict write access to /Library/StartupItems/; use System Integrity Protection (SIP); monitor launchd plist changes

BC Scripts

Linux | **Persist** | **Shell**

Prerequisites: root

Detection: Monitor /etc/bc.local and /usr/share/bc/init/ for unexpected entries; auditd -w /etc/bc.local -p wa

Mitigation: Restrict write access to /etc/bc.local and /usr/share/bc/init/; use kernel lockdown mode; monitor /etc/bc.local with auditd

BC Scripts

Persist | **Shell**

Prerequisites: root

Detection: Monitor /etc/bc.local and /usr/share/bc/init/ for unexpected entries; auditd -w /etc/bc.local -p wa

Mitigation: Restrict write access to /etc/bc.local and /usr/share/bc/init/; use kernel lockdown mode; monitor /etc/bc.local with auditd

BC Scripts

Detection: Monitor /etc/bc.local and /usr/share/bc/init/ for unexpected entries; auditd -w /etc/bc.local -p wa

Mitigation: Restrict write access to /etc/bc.local and /usr/share/bc/init/; use kernel lockdown mode; monitor /etc/bc.local with auditd

Configuration Profiles & MDM

Configuration profiles (`/usr/bin/profiles`) and MDM enrollment provide a platform-native, cloud-managed persistence and control layer. Profiles can install certificates, VPN configurations, Wi-Fi networks, Kerberos realms, Safari extensions, Gatekeeper exceptions, and arbitrary preferences. MDM enrollment adds remote command execution, app installation, and continuous policy enforcement. Malicious profiles can be used to disable security controls, install rogue certificates for MITM, or establish persistent access that survives reinstallation. Detection requires monitoring `/var/db/ConfigurationProfiles/` and `/Library/Managed Preferences/`.

Mechanism	Required privilege	Scope	Collection / Trigger	Trigger	What to review
Configuration Profiles (/usr/bin/profiles) Persist · Profile · Enterprise · T1176.1 · macOS 10.7+ · Requires root or device supervision, MDM enrollment for remote deployment	root	system	/var/db/ConfigurationProfiles/Store/ /var/db/ConfigurationProfiles/Settings/ /Library/Managed Preferences/<username>.plist /Library/Managed Preferences/mcx/*.plist /Library/managed_installs/Manifests/ profiles -P -v -L -C -D -I -R -N -S -E -U -e -f -r -p -c -d -n -v -s -V -h -v -v -v

ATT&CK coverage for emond rules, queues, and event-driven actions on macOS.

[S36]
Chromium Extensions
Firefox Extensions

ATT&CK coverage for browser extension persistence, including Chromium and Firefox profile or policy abuse.

[S40]
Installer Packages

ATT&CK coverage for preinstall and postinstall execution paths that can drop or enable persistence.

[S40b]
Mach-O Binary Execution
MITRE ATT&CK · Reviewed 2026-04
Source: attack.mitre.org
Mechanisms: Mach-O Binary

ATT&CK technique for trojanized or modified host binaries that re-execute on normal app launch.

[S40c]
Mach-O Load Command Execution
MITRE ATT&CK · Reviewed 2026-04
Source: attack.mitre.org
Mechanisms: LC_LOAD_DYLIB Addition

ATT&CK mapping for malicious Mach-O load commands that force dylib execution on launch.

[S37]
Boot or Logon Initialization Scripts: Login Hook
MITRE ATT&CK · Reviewed 2026-04
Source: attack.mitre.org
Mechanisms: Login Hooks

ATT&CK coverage for loginwindow hook persistence and its privileged system-wide variant.

[S38]
Boot or Logon Initialization Scripts: Startup Items
MITRE ATT&CK · Reviewed 2026-04
Source: attack.mitre.org
Mechanisms: Startup Items

ATT&CK coverage for legacy StartupItems directories and their executable ordering model.

[S39]
Boot or Logon Initialization Scripts: RC Scripts
MITRE ATT&CK · Reviewed 2026-04
Source: attack.mitre.org
Mechanisms: RC Scripts

ATT&CK coverage for historical rc.local and rc.common execution paths on macOS.

[S40a]
Event Triggered Execution
MITRE ATT&CK · Reviewed 2026-04
Source: attack.mitre.org
Mechanisms: Mail.app Rules (AppleScript actions)

ATT&CK umbrella technique for event-triggered execution; Mail rules with AppleScript actions fit the pattern even though no dedicated sub-technique exists for Mail-rule abuse.

Community Research

2 sources

[S28d]
Folder Actions for Persistence on macOS
Community Research · Reviewed 2026-04
Source: specterops.io
Mechanisms: Folder Actions / Automator / Quick Actions

Defender-relevant research on Folder Actions as event-triggered persistence and the FolderActionsDispatcher preference state.

[S31]
Click File, App Opens
Community Research · Reviewed 2026-04
Source: objective-see.org
Mechanisms: LaunchServices / URL Handlers

Reverse-engineering write-up on Launch Services document-handler abuse for stealthy macOS persistence.

MDN Documentation

1 source

[S21d]
Native manifests
MDN Documentation · Reviewed 2026-04
Source: developer.mozilla.org
Mechanisms: Firefox Extensions

Adjacent pivot for Firefox extensions that use native messaging, managed storage, or PKCS#11 manifests on macOS.

Mozilla Documentation

1 source

[S21b]
Policy Templates for Firefox
Mozilla Documentation · Reviewed 2026-04
Source: mozilla.github.io
Mechanisms: Firefox Extensions

Official Firefox enterprise policy reference, including policies.json and ExtensionSettings. On macOS, policies.json is placed in Firefox.app/Contents/Resources/distribution.

Mozilla Support

2 sources

[S21a]
Deploy Firefox with extensions
Mozilla Support · Reviewed 2026-04
Source: support.mozilla.org
Mechanisms: Firefox Extensions

Official Mozilla enterprise deployment guidance for pre-installing extensions through the Firefox distribution/extensions directory.

[S21c]
Manage policies on macOS desktops
Mozilla Support · Reviewed 2026-04
Source: support.mozilla.org
Mechanisms: Firefox Extensions

macOS-specific Firefox policy support, including org.mozilla.firefox preference payloads and Firefox 64 / ESR 60.4 policy support.

Tool Documentation

4 sources

[S25]
KnockKnock
Tool Documentation · Reviewed 2026-04
Source: objective-see.org
Mechanisms: Application Support helpers

Defender-oriented catalogue of macOS persistence locations that is useful for user-space helper staging sweeps.

[S26]
KnockKnock
Tool Documentation · Reviewed 2026-04
Source: objective-see.org
Mechanisms: Application startup scripts

Useful operational reference when startup logic is hidden in app resources or user-scoped helper scripts.

[S27]
KnockKnock
Tool Documentation · Reviewed 2026-04
Source: objective-see.org
Mechanisms: App preference triggers

Helpful defender reference for preference-backed execution surfaces that are easy to miss in broad triage.

[S36a]
Dylib Hijack Scanner
Tool Documentation · Reviewed 2026-04
Source: objective-see.org
Mechanisms: Dylib Hijacking / Proxying

Practical defender reference for @rpath abuse, weak imports, and proxy dylib detection on macOS.

Vendor Documentation

11 sources

[S12b]
Use secure token, bootstrap token, and volume ownership in deployments
Vendor Documentation · Reviewed 2026-04
Source: support.apple.com
Mechanisms: Local accounts / hidden users / Remote Login

Apple deployment guidance for SecureToken, bootstrap token, and volume ownership context that helps DFIR assess durable account control and FileVault-adjacent access implications.

[S21e]
Native messaging
Vendor Documentation · Reviewed 2026-04
Source: developer.chrome.com
Mechanisms: Chromium Extensions

Official Chrome native messaging host reference. On macOS, host manifests live in fixed browser-specific system or user locations; the manifest maps a host name to an absolute native binary path, uses stdio transport, and restricts callers through allowed_origins.

[S21f]
Native messaging
Vendor Documentation · Reviewed 2026-04
Source: learn.microsoft.com
Mechanisms: Chromium Extensions

Microsoft Edge native messaging reference. Useful for Chromium-family DFIR because Edge uses its own macOS NativeMessagingHosts locations and the same host-manifest concepts: native binary path, stdio transport, and allowed extension origins.

[S30]
site - Site-specific configuration hook
Vendor Documentation · Reviewed 2026-04
Source: docs.python.org
Mechanisms: Python Startup Hooks

Official Python documentation for .pth files, sitecustomize.py, and usercustomize.py execution at interpreter start.

[S43]
Privacy Preferences Policy Control device management payload settings for Apple devices
Vendor Documentation · Reviewed 2026-04
Source: support.apple.com
Mechanisms: High-impact MDM payloads

Apple deployment reference for the com.apple.TCC.configuration-profile-policy payload, including service-specific allow or deny behavior and PPPC interpretation boundaries.

[S44]
Managed Login Items device management payload settings for Apple devices
Vendor Documentation · Reviewed 2026-04
Source: support.apple.com
Mechanisms: High-impact MDM payloads

Defines com.apple.servicemanagement managed login item rules and the matching keys used to auto-approve items.

[S45]
Manage login items and background tasks on Mac
Vendor Documentation · Reviewed 2026-04
Source: support.apple.com
Mechanisms: High-impact MDM payloads; Declarative Device Management / Managed background tasks

Operational guidance for background task management, rule matching, notifications, and sfltool or log-based validation on managed Macs.

[S46]
Certificates device management payload settings for Apple devices
Vendor Documentation · Reviewed 2026-04
Source: support.apple.com
Mechanisms: High-impact MDM payloads

Documents certificate payload identifiers that install roots, intermediates, or identities onto managed devices.

[S47]
Intro to certificate management for Apple devices
Vendor Documentation · Reviewed 2026-04
Source: support.apple.com
Mechanisms: High-impact MDM payloads

Explains certificate trust anchors, identities, and enterprise trust deployment through configuration profiles.

[S48]
Background task management declarative configuration for Apple devices
Vendor Documentation · Reviewed 2026-04
Source: support.apple.com
Mechanisms: Declarative Device Management / Managed background tasks

Apple deployment reference for macOS 15+ Declarative Device Management background task configuration, including MDM-delivered executables, scripts, and launchd configurations in managed service workflows.

[S48a]
Use declarative device management to manage Apple devices
Vendor Documentation · Reviewed 2026-04
Source: support.apple.com
Mechanisms: Declarative Device Management / Managed background tasks

Apple deployment overview for declarative device management, declarations, status reporting, and managed configuration context used to distinguish legitimate management state from suspicious policy-plane persistence.

macOS Automation

1 source

[S28c]
Automator Quick Actions
macOS Automation · Reviewed 2026-04
Source: macosxautomation.com
Mechanisms: Folder Actions / Automator / Quick Actions

Operational reference for Automator Quick Actions / Services workflows and their user-library Services integration.