星連VPN 開源軟體：安全複審報告

繁體中文版 · 新報告 · 2026-11-17

結論：原 A-01 / A-02 / A-03 在本次審查與限定控制範圍內有修復證據。新增一項低風險 A-04：Android 啟動錯誤後可能重複要求關閉已由 Go 清理的 descriptor。來源與分層控制成立，未在 Android 真機重現 FD 重用或崩潰；因此不宣稱所有問題已關閉。

1. 評估範圍與版本

報告日期：2026-11-17（Asia/Shanghai）。這是重新取得原始碼後的新評估，保留 2026-11-13 舊報告；本報告的修復結論以以下完整 commit 為準。

項目	記錄
Protocol SHA	ddfd0d7a6ee4b74561dda16315493796899e92d6
App SHA	49a0a31b710b3b07bd90c459e3aebd4fd8a193e9
協議規格	Draft 4; wire byte 0x01; implementation 1.0.0-draft
App 版本	0.1.0+1; first-generation prototype
本報告來源	216 tracked files; 112 added/modified/deleted paths vs retained Nov 13 snapshot
來源清單	source_inventory.json

App 的 Go pseudo-version 與 protocol-revision.txt 均對應上述協議 SHA。兩個舊 commit 不存在於新遠端已取得的歷史中，因此比較採用保留的舊檔案快照並正規化 CRLF；不宣稱 Git 祖先關係。原始碼快照與逐檔 SHA-256 可供覆核。

此開源版本自述為第一代協議 / 原型 App；目前商用官方 App 與 SingLink 2.0 核心不在本次來源範圍。先前免費 / 付費 Windows 測評不等於本次來源或商用版本證明。本次未登入帳號、擷取使用者流量、修改 VPN / 代理 / 登錄檔，也未將報告發送或公開發布。

來源：README.md:8；README.md:5；core/go.mod:5；SPEC.md:3。

2. 方法、獨立性與證據等級

依使用者要求，由三個分開的 AI 代理進行協議審查、App 審查與反向交叉覆核，主代理整合並另行跑基準、模糊測試與 Linux 測試。這是不同審查路徑，並非獨立人類團隊、外部資安公司認證或形式化密碼學證明。原廠文件是待驗證的說法，沒有當成執行指令或通過證據。

證據分為：原始碼觀察、原廠測試重新執行、審查者新增控制、真正 OS API 元件測試，以及尚未完成的真機 / 生產環境驗收。測試通過不代表不存在漏洞。風險採定性分級，未指定未驗證的 CVSS 數字。

工具為 Go 1.27.4、Flutter 3.35.6 / Dart 3.9.2、govulncheck v1.9.0；實際命令、時間、環境、stdout、stderr 與退出碼見每個 evidence ID。Go 使用 local toolchain，未無聲切換下載版本。報告日期採本機時區；記錄中 ISO 時間保留其 UTC offset。

Linux 測試在 QEMU TCG 的 Alpine 3.22.8 / Linux 6.12.115 RAM-root 執行，無網卡、無主機資料夾分享、無主機磁碟、僅 guest loopback；不是正式伺服器部署。QEMU 下載 SHA-512 與 Alpine rootfs SHA-256 對照發布者資料，kernel 只記錄官方 HTTPS 來源與雜湊，未額外驗證獨立簽章。所有 guest 測試以 root 執行；不是非 root 部署驗證。使用了真正 Linux flock、rename、fsync 與子程序，但未模擬實體磁碟斷電或多主機共享儲存。

證據：ENV-LINUX-GUEST-RUN-R2；工具來源；交叉審查筆記。

威脅模型與涵蓋路徑

分別檢查可修改網路輸入的未認證對手、持有效 token 的濫用者、一般設定檔外洩、啟動 / 停止競態，以及 build / 原生函式庫信任面。節點 root / 同使用者惡意程式、完整簽署金鑰失陷不會被當成一般遠端攻擊等價情境。重點包括 parser / frame / AEAD / token / early data、重放 / 撤銷 / 地址政策、憑據遷移、FFI / FD 交接與 release gates。檔案總數是來源清冊，不代表逐行證明所有檔案、gVisor、Flutter、OS 或第三方依賴完全安全；審查集中在上述安全關鍵路徑。

3. 舊問題與來源修復

ID	本次結論	驗證範圍
A-01	已修復被審查的儲存設計；各平台原生驗收未全完成。	metadata v3 / protected store；Windows DPAPI helper 實測，Android Keystore / Apple Keychain 只審碼。
A-02	原 Dart shutdown 過期續接缺陷已修復。	generation、cleanup owner、shutdown barrier 控制通過；另列 A-04 / iOS acknowledgement。
A-03	來源 / build 設定已移除 debug release 後備。	release gate / CI 名稱審查與合成檔路徑控制；未檢查正式 APK / 簽章憑證。

A-01：憑據儲存、遷移與剩餘資料

一般 JSON 改只保存 secretRef，不輸出完整 bearer link / warning consent。256-bit 隨機 reference、journal、protected readback 後提交與復原路徑避免悄悄以明文後備。label / server / plan 等 metadata 仍可讀，並非整份設定都加密。遷移失敗時刻意保留舊 plaintext 檔，不應寫成升級立刻消除所有明文；成功遷移也不會追溯抹除舊備份 / 舊磁碟區塊。

Windows 使用 current-user DPAPI、reference entropy、immutable 寫入與原子檔案處理；Android 使用 noBackupFilesDir、AndroidKeyStore AES-256/GCM、reference AAD / nonce 與 backup 排除；Apple 使用不同步 Keychain，iOS after-first-unlock-this-device-only 並分享 extension access group。這不保證 Android 硬體金鑰、不要求每次讀取生物辨識；iOS 首次解鎖後背景可讀，也不是鎖機一律不可讀。DPAPI 可防普通檔案外洩與其他使用者，不能視為防同使用者惡意程式。詳見 Microsoft DPAPI 文件。

Windows 真正 DPAPI 控制抽取固定來源的 Read / Write / Delete / SecretPath helpers；唯一實質環境重導是 SHGetKnownFolderPath 指向獨立暫存根目錄，省略 Flutter channel glue。執行 roundtrip、ciphertext 不含 canary、不可覆寫 reference、reference entropy 置換拒絕、非法 reference、真正已寫入的 ciphertext 毀損拒絕與刪除。完整 Flutter dispatch、真實 known-folder、Android / Apple 原生儲存、備份還原、升級遷移與鎖機情境未執行。

lib/models/profile.dart:112 · lib/core/credential_migration.dart:16 · lib/core/app_store.dart:371 · windows/runner/credential_store.cpp:89 · android/app/src/main/kotlin/com/singlinklabs/singlinkvpn/CredentialStore.kt:23 · apple/CredentialKeychain.swift:37

A-INDEPENDENT-WINDOWS-DPAPI-V3; A-INDEPENDENT-RETEST

A-02：shutdown 與連線所有權

generation guards 讓舊工作失效；controller 等待 pending connect、清理 owner 及代理復原。主程式 exit timeout / error 會取消離開。獨立阻塞控制驗證 shutdown barrier；不表示每個原生介面都有同等保障。iOS stop acknowledgement 及 A-04 需分開處理。

A-03：發行簽署

Android release 使用 protectedRelease，沒有 debug signing fallback；未供外部 signing inputs 時 gate 拒絕。CI 產物明示 DEMO-debug / UNSIGNED。11 個來源 build-gate 測試及獨立合成檔路徑控制通過；這些只驗證存在 / 名稱 / 組態，不驗證 keystore 內容、憑證真偽、金鑰保管或更新鏈。沒有執行正式 Gradle release、安裝 APK 或檢驗正式憑證 fingerprint。

lib/core/client_controller.dart:421 · lib/core/client_controller.dart:535 · lib/main.dart:34 · android/app/build.gradle.kts:64 · scripts/verify_release_signing.py:9 · .github/workflows/build.yml:226

A-BUILD-GATES; A-INDEPENDENT-SIGNING-GATE

4. A-04：Android 失敗啟動後重複關閉 descriptor

狀態：Open；風險：Low；類型：有條件的本機生命週期 / 可用性缺陷。 需要取消 / watch failure 落在 Go callback 已取得 TUN、但尚未完成發布的狹窄區間。不是未驗證遠端 RCE，也沒有證實封包外洩。

根因與可達路徑

1. Kotlin service detach TUN fd，VpnBridge 放入 unadopted 集合。

2. Dart 僅在 FFI res.ok 時通知 adopted；錯誤路徑呼叫 closeFd。

3. Go SingLinkTunStart 的 Engine.StartWith callback 建立 NewFDDevice，傳回 tunnel.Close 作清理。

4. 若 generation 已過期，engine.finish 先清理該 device，再回傳 context.Canceled。錯誤結果不代表從未接管 FD。

5. Kotlin 仍認為未 adopted，遂 adoptFd(fd).close()。其集合可防兩次 Kotlin close，卻不知道 Go 已關閉同一個原始整數。

FFI start / stop 可在不同 isolate 執行，來源上可達此取消窗口。一般成功啟動、取得 FD 前就失敗並不構成同一缺陷。Android 的 detachFd 會轉交 raw descriptor 責任；參見 Android 官方 ownership API。

已重現與未證實的部分

實際固定版本 Go engine + app tunnel 配合 fake blocking Device 與合成 loopback 節點，驗證取得 callback 後取消，device.Close 先執行一次，再回傳 context.Canceled。另以真正 Dart AndroidVpnCore 元件，模型化 native 已取得 / 關閉後回傳錯誤，觀察順序 native-acquire:42 → native-close:42 → kotlin-close:42。後者的 Kotlin channel、OS FD 都是 mock。

兩個控制加上獨立交叉審碼支持「錯誤的第二次 close 請求」。沒有把真實 Android Go FFI / Kotlin / OS 串起來執行，沒有實際重用 FD 整數、誤關其他資源、崩潰、資料損失或商用影響證據。若該整數在兩次關閉間被重用，後次 close 可能關到同程序其他 I/O；這是條件性風險，評為 Low。

修復建議

以明確 consumed / adopted flag 讓資源所有權獨立於成功結果；或 core 接管 duplicate FD，訂出永遠消耗傳入 duplicate 的合約，由 Kotlin 只關其保留原件。涵蓋取得前與取得後所有失敗。不能只把 adopted 通知提前，否則 parse / handshake 未取得時可能反而洩漏。驗收必須包含實際 Android FD reuse 與每個 owner 恰好清理一次。

lib/platform/android.dart:215 · lib/platform/android.dart:303 · android/app/src/main/kotlin/com/singlinklabs/singlinkvpn/VpnBridge.kt:45 · android/app/src/main/kotlin/com/singlinklabs/singlinkvpn/VpnBridge.kt:64 · core/ffi/main.go:165 · core/tunnel/fd.go:29 · core/tunnel/fd.go:70 · core/tunnel/tunnel.go:123 · engine/lifecycle.go:31

A-INDEPENDENT-NATIVE-OWNERSHIP-V3; A-INDEPENDENT-RETEST; cross-review

5. 密碼學、明文與信任邊界

目前來源使用 32-byte 方向性 traffic key 建立 AES-GCM，因此本次應記為 **AES-256-GCM**，不是 AES-128。審查者控制驗證方向金鑰不同且為 32 bytes；自產 first-flight 明文標記未出現在序列化密文，正控制可解密，修改 AEAD 與已簽署能力資料均遭拒絕。這些是程式內序列化 / 解密測試，沒有執行封包擷取，亦非密碼破解難度證明。

設計結合 X25519 與 ML-KEM-768，仍屬自訂協議。第一個 flight 的保密性依賴靜態伺服器金鑰；第一個 flight 不具完整前向保密，first flight 之後的流量另混入 ephemeral-ephemeral X25519，設計提供經典攻擊者模型下的前向保密；靜態 KEM 金鑰日後遭入侵時仍不能主張後量子前向保密。應保留規格明示的限制，不能用「有後量子元件」取代完整分析。

應用程式本來就會在端點處理明文；VPN 節點解封裝後，非 HTTPS 等目的流量也可能是明文。內層 AES-256 不證明伺服器不記錄內容、營運日誌為零、所有 App 都走 VPN，或任一日本商用節點採用同一版本。外層 TLS / 憑證驗證、raw TCP 與 insecure 模式需各自評估；App 的明示警告不能被概括「已加密」介面取代。

來源：singlink/frame.go:30 · singlink/handshake.go:110 · singlink/handshake.go:156 · singlink/const.go:50；控制：P-INDEPENDENT-CRYPTO-EARLY。

6. 剩餘強化與平台限制

項目	現況 / 判斷
A-H01	部分修復。完整 protocol pin、clean-tree / timestamp 與 artifact fingerprint；Actions 仍用 mutable tags，unsigned sidecar 只是本機一致性記帳，非可信簽署 attestation。
A-H02	未修復的條件性開發 DLL fallback。先用打包核心，失敗後仍接受環境 / cwd / 祖先路徑，無 release gate。需本機可控路徑且打包核心不可用；未示範遠端注入。
A-H03	部分修復。警告同意保存在 protected credential、連線前重解析，iOS extension 亦驗證；主介面 Protected / Your traffic is encrypted 仍過於概括。desktop system proxy 不保證所有軟體走 VPN。
A-H04	部分修復。DNS idle pool 限 4，但每次 UDP query 仍可新增工作，無全域 outstanding cap。未執行持續耗盡攻擊。
A-H05	timestamp-derived profile ID 保留；未重現安全繞過或碰撞，不升級成漏洞。
A-H06	iOS 來源觀察：stopIfActive 忽略 waitForStatus timeout 的 nil，disconnecting 也可直接續行。可能未觀察 disconnected 就回報成功；未在 iOS 真機驗證，不宣稱已發生 leak。

scripts/protocol_provenance.py:25 · .github/workflows/build.yml:39 · lib/core/singlink_bindings.dart:166 · lib/core/singlink_bindings.dart:219 · lib/core/client_controller.dart:254 · lib/ui/home_page.dart:153 · core/tunnel/dns.go:25 · ios/Runner/SingLinkVpnPlugin.swift:182 · ios/Runner/SingLinkVpnPlugin.swift:343

以上不與 A-04 混加為新高風險漏洞，也不能忽略。prototype / platform status 與 desktop system proxy 限制應在公開結論明示。

7. 實測結果與資料

證據 ID	結果	範圍
A-PUBGET	exit 0	依賴解析
A-ANALYZE	exit 0	Dart analyzer
A-FLUTTER-FUNCTIONAL	113 success; 3 skipped; 1 teardown error; exit 1	完整功能測試的實際失敗結果
A-FLUTTER-GOLDEN	6 errors; exit 1	5 個圖像差異 + 1 個 finder assertion；未重製通過
A-INDEPENDENT-RETEST	5 passed; exit 0	新增儲存 / shutdown / DLL / Android 控制
A-INDEPENDENT-VENDOR-SECURITY	32 passed; exit 0	原廠安全測試另外重跑，與完整 suite 有重疊
A-INDEPENDENT-NATIVE-OWNERSHIP-V3	1 passed; exit 0	實際 Go engine + tunnel，fake Device
A-INDEPENDENT-WINDOWS-DPAPI-V3	exit 0	真正 DPAPI helpers、隔離路徑
A-LINUX-TUNNEL	4 passed; exit 0	原始 Go tunnel 套件、Linux guest
A-REAL-PARSER	1 passed; exit 0	新編譯真實 Go C ABI + Dart FFI parser
A-CABI-BUILD	exit 0	Windows C ABI DLL 編譯，非 GUI release
A-PIN-VERIFY-R2	exit 0	固定 protocol pin 一致性
A-CABI-PROVENANCE-R2	exit 0	本機 artifact / source consistency
A-BUILD-GATES	11 passed; exit 0	合成 git / build gates
A-INDEPENDENT-SIGNING-GATE	exit 0	簽署設定 / 合成路徑

Linux tunnel coverage 為 **80.08%（189 / 236 statements）**，不是 Android / iOS 原生或整體 App coverage。四個 synthetic signed profile 經真實 DLL parser 驗證，包含 TLS 預設、raw TCP 與 insecure/pinned TLS 警告與 protected import consent；使用 MemorySecretStore，不代表各原生 vault integration。

Windows / Linux go vet、govulncheck 都退出 0，無 finding 事件；Govuln DB last-modified 2026-11-11T02:51:11Z。只針對提交的 Pub lockfile 24 個 hosted package 名稱 / 版本送 OSV 查詢，0 advisory matches；未送原始碼 / 憑據。這不是 Gradle / CocoaPods / OS framework / 完整 Flutter SDK 的依賴認證。Go module 完整清單另附。

A-VET-WINDOWS; A-VET-LINUX; A-VULN-WINDOWS; A-VULN-LINUX; A-MODULES; A-PUB-OSV

8. 失敗、跳過與未執行項目

完整 Flutter functional 並非退出 0：113 個 visible tests 成功，3 個跳過，1 個 tearDownAll 因 Windows 已載入的 mock DLL 暫存目錄無法刪除而錯誤。這不能改寫為全 suite 通過；沒有觀察到功能 assertion 失敗，但 cleanup 有問題。3 個跳過是實際節點端到端、真實 ABI parser gate、真實 registry；其中 parser 已用新 DLL 另跑通過，其餘仍未執行。

golden suite 六個測試均錯誤：五個實際 pixel mismatch，分別為 connected light 1.86%（23,702 pixels）、disconnected light 1.74%（22,189）、connected dark 1.85%（23,682）、disconnected dark 1.74%（22,155）、

settings 6.09%（77,786）。另一個 location sheet 在比圖前就因 findsOneWidget 預期一個「42 ms」、實際找到五個而失敗。原參考圖採 macOS 字型，與 Windows 有差異，這是可能原因而非每個 pixel 的完整歸因。這些不是已證實資安漏洞；未重生圖片掩蓋失敗，差異圖與原始紀錄保留。

Windows Go app baseline 的 3 個失敗來自 Unix-only replay store fixture，後續 Linux 原始 tunnel 套件通過。最初 pin / sidecar 驗證因審查者把 coverage 檔寫進 protocol worktree 而失敗；將證據移至 evidence 後 R2 通過，屬本次儀器設定問題，非產品漏洞。

Go ownership fixture 最初沒有可連節點，再因 Windows Unix store 不支援而失敗；修正成明示 test-only memory store / loopback node 後 V3 通過。DPAPI fixture 前兩版企圖用 ofstream 修改 hidden file 卻未成功寫入，曾輸出假的「corrupt accepted」；V3 僅對合成檔清 hidden flag 並檢查寫入後，真正毀損正確被拒。前兩次訊息不是 DPAPI bypass，全部記錄保留。

本次未安裝或運行 Windows GUI release，未跑 Android / iOS / macOS native app、正式簽署 / 備份 / 系統代理 / kill switch / DNS leak / 串流 / 商用節點，也沒有成功 race-detector 證據。沒有要求 OS 登錄檔 smoke、沒有 packet capture。真正的完整裝置安全性仍須平台驗收。

A-GO-BASELINE-WINDOWS; A-PIN-VERIFY; A-CABI-PROVENANCE; A-INDEPENDENT-NATIVE-OWNERSHIP; A-INDEPENDENT-NATIVE-OWNERSHIP-V2; A-INDEPENDENT-WINDOWS-DPAPI; A-INDEPENDENT-WINDOWS-DPAPI-V2

9. 修復與下一次驗收

1. A-04：先明確化 FD 所有權協議，實作後在 Android instrumentation / 真機測試取消落點、FD 整數重用、handshake / parse 失敗、watch failure、revoke 與 stop。要求每個實際 FD 恰好關閉一次，無誤關其他 pipe/socket、無洩漏。未測真機前不能把條件性影響寫成已證實崩潰。

2. iOS stop 應傳遞 timeout / manager load error，且 disconnecting 不提早報成功；真機確認 extension 確實結束。release 模式應只信任打包核心，停用開發 fallback；固定 CI Actions commit，強化發行簽章與 attestation。

3. 真機覆核 Android Keystore / Apple Keychain、權限群組、裝置鎖定、備份 / restore / migration；正式 release 憑證與更新鏈另驗。Windows 完整 GUI 與 DPAPI channel 要做整合測試，不能由 helper 控制代替。

4. 在相容平台解決 Windows mock DLL teardown，於原設計平台重跑 golden；真實 registry / VPN / tunnel / DNS / 斷線行為需另一次明示裝置驗收，不能借用這次單元測試結論。

本次新來源審計已完成至可交付範圍，未修改 App。公開時應同時保留 open finding、所有失敗與範圍說明，不應稱商用各平台已完全通過、零明文 / 零日誌或正式發行可用。未向第三方發布報告或傳送帳號 / 使用者流量。

10. 證據包與重現說明

新報告、英文版與繁體版各自獨立，finding ID、commit、數字及限制對照一致。HTML / Markdown 是可編輯版；PDF 是閱讀版。入口為 START_HERE.html，完整檢查矩陣見 check_summary.json。

公開交付包只含本次報告、公開來源快照 / LICENSE / NOTICE、合成測試、測試結果與差異圖；不含帳號、付費憑據、個人瀏覽封包、工具快取或正式簽署金鑰。下載依賴時僅向官方 / 套件來源請求公開材料；OSV 只接收套件名與版本。這不等同聲稱測試全程完全離線：只有 Linux guest 無外部網路。

本機保留原始紀錄；分享包對本機使用者 / 工作目錄路徑做一致替換，redactions.json 記錄來源與分享檔雜湊。sources 逐檔 bytes 保持不變，SHA256SUMS.txt 驗證交付包內容。工具鏈、QEMU / SDK / module cache 不打包，避免把第三方工具重發與證據混在一起。重跑需先準備指定工具與相鄰 protocol checkout，並依 README_REPRODUCE.md 設定路徑；並非任意電腦一鍵可跑。

原始測試命令與退出碼清單：

ID	Exit	備註
A-ANALYZE	0	限定檢查通過
A-BUILD-GATES	0	限定檢查通過
A-BUILD-LINUX-TUNNEL	0	限定檢查通過
A-CABI-BUILD	0	限定檢查通過
A-CABI-PROVENANCE-R2	0	限定檢查通過
A-CABI-PROVENANCE	1	需讀取範圍 / 失敗原因
A-FLUTTER-FUNCTIONAL	1	需讀取範圍 / 失敗原因
A-FLUTTER-GOLDEN	1	需讀取範圍 / 失敗原因
A-GO-BASELINE-WINDOWS	1	需讀取範圍 / 失敗原因
A-INDEPENDENT-NATIVE-OWNERSHIP-V2	1	需讀取範圍 / 失敗原因
A-INDEPENDENT-NATIVE-OWNERSHIP-V3	0	限定檢查通過
A-INDEPENDENT-NATIVE-OWNERSHIP	1	需讀取範圍 / 失敗原因
A-INDEPENDENT-PUBGET	0	限定檢查通過
A-INDEPENDENT-RETEST	0	限定檢查通過
A-INDEPENDENT-SIGNING-GATE	0	限定檢查通過
A-INDEPENDENT-VENDOR-SECURITY	0	限定檢查通過
A-INDEPENDENT-WINDOWS-DPAPI-V2	1	需讀取範圍 / 失敗原因
A-INDEPENDENT-WINDOWS-DPAPI-V3	0	限定檢查通過
A-INDEPENDENT-WINDOWS-DPAPI	1	需讀取範圍 / 失敗原因
A-LINUX-TUNNEL	0	限定檢查通過
A-MODULES	0	限定檢查通過
A-PIN-VERIFY-R2	0	限定檢查通過
A-PIN-VERIFY	1	需讀取範圍 / 失敗原因
A-PUBGET	0	限定檢查通過
A-REAL-PARSER	0	限定檢查通過
A-VET-LINUX	0	限定檢查通過
A-VET-WINDOWS	0	限定檢查通過
A-VULN-LINUX	0	限定檢查通過
A-VULN-WINDOWS	0	限定檢查通過

再發布與署名

附帶來源的 LICENSE / NOTICE 保留原作者 SingLinkVPN 的署名要求；法律適用範圍未在本次做獨立法律鑑定。報告作者為受使用者委託的 AI 輔助評估；不使用原廠「third-party audit」措辭替本次背書。公開前請維持完整範圍、版本、殘餘問題與證據連結。