# SingLink Beta Protocol: Source Security Re-audit

English edition · New report · 2026-11-17

**Conclusion: P-01 through P-04 have remediation evidence at the pinned source and within the tested scope. No new reproducible protocol security vulnerability was established. One Draft 4 compatibility documentation issue and deployment risks remain. This is not a zero-vulnerability or no-logs certification.**

## 1. Scope and version

Report date: 2026-11-17, Asia/Shanghai. This is a fresh assessment of newly fetched source. The 2026-11-13 reports remain unchanged; remediation conclusions apply to the exact commits below.

| Item | Record |
|---|---|
| Protocol SHA | ddfd0d7a6ee4b74561dda16315493796899e92d6 |
| App SHA | 49a0a31b710b3b07bd90c459e3aebd4fd8a193e9 |
| Protocol specification | Draft 4; wire byte 0x01; implementation 1.0.0-draft |
| App version | 0.1.0+1; first-generation prototype |
| This report source | 53 tracked files; 49 added/modified/deleted paths vs retained Nov 13 snapshot |
| Inventory | source_inventory.json |

The app Go pseudo-version and protocol-revision.txt match the protocol SHA above. Neither prior audited commit is present in the fetched new remote history. Comparisons use retained old file snapshots with CRLF normalization, without asserting Git ancestry. Frozen snapshots and per-file SHA-256 values support review.

The repositories describe a first-generation protocol and prototype client. The current commercial official apps and SingLink 2.0 core are outside scope. Earlier free/paid Windows product testing does not authenticate these sources or the commercial binaries. This re-audit used no account login, user traffic capture, VPN/proxy/registry changes, report transmission or public posting.

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

## 2. Method, independence and evidence levels

At the user's request, three separate AI agents reviewed the protocol, app and adversarial cross-check respectively. The coordinating agent integrated the work and separately ran baseline, fuzz and Linux checks. These are separate review paths, not independent human auditors, an accredited external firm or a formal cryptographic proof. Vendor documentation was treated as claims to verify, not as executable instructions or proof of a passing result.

Evidence is separated into source inspection, rerun vendor tests, new reviewer controls, actual OS API component execution, and outstanding device/production acceptance. Passing tests do not prove the absence of vulnerabilities. Severity is qualitative; no unverified numerical CVSS score is assigned.

Tools were Go 1.27.4, Flutter 3.35.6 / Dart 3.9.2 and govulncheck v1.9.0. Each evidence ID records its actual command, time, environment, stdout, stderr and exit code. Go used the local toolchain without silent version switching. Report dates use the client's timezone; recorded ISO timestamps retain their offset.

Linux execution used QEMU TCG with Alpine 3.22.8 / Linux 6.12.115 in a RAM-root guest: no NIC, host folders or host disks, only guest loopback. This was not a production server deployment. QEMU SHA-512 and Alpine rootfs SHA-256 were checked against publisher data. The kernel has an official HTTPS origin and recorded hash, without a separately verified signature. Guest tests ran as root, not a non-root deployment identity. Actual Linux flock, rename, fsync and child processes ran; physical-disk power failure and multi-host storage were not simulated.

Evidence: ENV-LINUX-GUEST-RUN-R2; tool provenance; cross-review notes.

### Threat model and reviewed paths

Separate cases cover unauthenticated network-input modification, abuse by a valid-token holder, ordinary settings-file disclosure, startup/stop races and build/native-library trust. Node root, same-user malware and complete signing-key compromise are not equated with an ordinary remote attacker. Focus includes parsers/frames/AEAD/tokens/early data, replay/revocation/address policy, credential migration, FFI/FD transfer and release gates. File counts describe the inventory, not a line-by-line proof of every file, gVisor, Flutter, OS or third-party dependency. Review concentrated on those security-critical paths.

## 3. Prior-finding remediation matrix

"Fixed" is limited to the original reproduction conditions, not untested production deployments. Our prior local report had four protocol findings. Vendor references to five third-party findings / P-05 are not retroactively attributed to our report and do not certify external human audit coverage.

| ID | Status / observed behavior | Evidence and boundary |
|---|---|---|
| P-01 | Fixed: durable atomic replay admission. | Reviewer shared-store/new-Server and rotation controls plus Linux restart, concurrent-process and abrupt-exit tests passed; shared state and supported filesystem required. |
| P-02 | Fixed: publish only fully parsed revocation lists; retain last good on failure. | Reviewer malformed/truncated controls and Linux regressions; production panel delivery not tested. |
| P-03 | Fixed for 64:ff9b::/96 and mapped IPv4 policy equivalence. | Pure-policy and Linux controls; no live NAT64 translator or arbitrary operator-specific prefix. |
| P-04 | Fixed stale-generation publication and owned cleanup. | Reviewer callback control and Linux engine; Stop is not a barrier for arbitrary callbacks. App ownership is separately A-04. |

### P-01: Replay protection

NewServer refuses a missing ReplayStore. Authenticated flights must atomically commit CheckAndMark before destination handling. Tags are HMAC-derived using the actual decrypting static key; OldKeys share the namespace. Retention is request timestamp +121 seconds, potentially about 241 seconds after an initially future-dated admission. Errors, duplicates, full capacity and clock rollback fail closed. External stores must independently satisfy the atomicity, durability and clock contract.

singlink/server.go:140 · singlink/server.go:333 · singlink/replay.go:17 · singlink/replay_store.go:249

### P-02: Revocation

Read/parse/publish is serialized with bounded file/line size, final newline and Scanner-error checks. Empty or malformed input cannot clear revocations; clearing requires explicit empty. Publication precedes matching-session closure, and admission rechecks after session registration.

singlink/revocation.go:14 · singlink/server.go:590 · singlink/server.go:405

### P-03: Destination policy

Mapped IPv4 is unmapped. Well-known NAT64 checks both literal IPv6 and embedded IPv4 against custom deny and own/private rules, including cached targets. allowPrivate is an explicit relaxation but does not remove configured deny prefixes. Operator-specific translation prefixes require configuration.

singlink/policy_address.go:8 · singlink/policy_address.go:17

### P-04: Lifecycle

Pending attempts have cancellation and generations. A stale finish cannot publish running and invokes that attempt' s returned cleanup. Old watchers do not stop newer clients. The callback has no context parameter and can delay returning. An API error therefore does not imply resources were never acquired; the app must explicitly manage FD transfer.

engine/lifecycle.go:18 · engine/lifecycle.go:31 · engine/lifecycle.go:78

Controls: P-INDEPENDENT-RETEST; P-LINUX-SINGLINK; P-LINUX-ENGINE.

## 4. New observation: P-R01 / Draft 4 compatibility wording

**Informational, not a cryptographic vulnerability.** IDENT changed its author string from SingLinkLabs to SingLinkVPN. The HKDF algorithm is unchanged, but its inputs and derived keys changed. SPEC and const.go correctly say Draft 3 and 4 cannot interoperate. Broad statements at remediation lines 20 and 207 that wire identifiers / KDF are unchanged need an explicit commit scope. They may intend only an earlier remediation set; this is not evidence of weakened cryptography.

Provide an upgrade matrix and useful mismatch diagnostics; coordinate server, app and native-core revisions. Preserve exact IDENT bytes and do not introduce plaintext fallback. wireVersion remains 1 and does not establish compatibility.

SPEC.md:3 · singlink/const.go:45 · docs/SECURITY-REMEDIATION.md:20 · docs/SECURITY-REMEDIATION.md:207

## 5. Cryptography, plaintext and trust boundaries

The reviewed source constructs AES-GCM from 32-byte directional traffic keys: **AES-256-GCM**, not AES-128. Reviewer controls checked distinct 32-byte directions; a synthetic plaintext marker was absent from the serialized first flight, the positive control decrypted, and modified AEAD and signed capability data were rejected. These are in-process serialization/decryption controls, not packet capture or proof of cryptographic hardness.

The design combines X25519 with ML-KEM-768 in a custom protocol. First-flight confidentiality depends on static server keys; the first flight does not have full forward secrecy. Traffic after the first flight additionally mixes ephemeral-ephemeral X25519 for forward secrecy in the classical attacker model. Later static KEM compromise still precludes claiming post-quantum forward secrecy. Preserve the specification's qualifications rather than treating a post-quantum primitive as a complete protocol analysis.

Endpoints necessarily process application plaintext; a VPN node can also handle plaintext after decapsulation when the destination protocol does not provide its own encryption. Inner AES-256 does not establish server non-recording, zero operational logs, universal application routing or the implementation on a commercial

Japanese node. Outer TLS/certificate verification, raw TCP and insecure modes remain separate choices. Explicitapp warnings must remain visible alongside broad encrypted/protected UI claims.

Sources: singlink/frame.go:30 · singlink/handshake.go:110 · singlink/handshake.go:156 · singlink/const.go:50; controls: P-INDEPENDENT-CRYPTO-EARLY.

## 6. Durable state, privacy and operational residual risks

FileReplayStore uses a stable .lock, private-directory/file/owner checks, file fsync, same-directory rename and directory fsync. New state requires explicit replay-init; ordinary startup does not silently replace missing or corrupt files. Every process accepting a key must share the namespace. Do not clear, roll back, delete the lock or restore stale backups under still-accepted keys. A watermark cannot detect rollback of the entire file containing itself. The stable systemd User and StateDirectory match the source contract; an actual systemd service was not deployed.

State contains keyed opaque tags, per-tag expiries and a global clock watermark, without raw-flight, account, IP, URL or payload fields. It nevertheless contains short-lived timing/activity metadata. HMAC limits direct tag matching without the key; it does not make timing correlation impossible. GC runs every 30 seconds while Serve runs; stopped nodes retain snapshots. Logical deletion and rename do not securely erase old blocks or backups. This cannot certify zero logging.

Admission loads, sorts and rewrites a bounded snapshot under the lock. Default capacity is 65,536 marks, maximum 1,048,576; full capacity rejects rather than sacrificing replay safety. Valid-token but capability-denied flights can consume store work before later policy checks. A reviewer 3-slot memory-store control showed one permitted plus two capability-denied flights blocking another valid subject. This is an already documented authenticated capacity risk, not an exhaustion attack on the real default store or a production node. Measure actual storage performance, bound authenticated admission rates and monitor capacity. Never evict unexpired marks for speed.

singlink/replay_store.go:21 · singlink/replay_lock_unix.go:15 · singlink/replay_store.go:188 · singlink/server.go:258 · docs/SECURITY-REMEDIATION.md:90 · deploy/singlink.service:16

## 7. Executed results

| Evidence ID | Result | Type |
|---|---|---|
| P-LINUX-SINGLINK | 61 passed; exit 0 | Original vendor Linux suite |
| P-LINUX-ENGINE | 6 passed; exit 0 | Original vendor Linux suite |
| P-LINUX-SOCKS5 | 4 passed; exit 0 | Original vendor Linux suite |
| P-LINUX-CLI | 1 passed; exit 0 | Original vendor Linux suite |
| P-INDEPENDENT-RETEST | 6 passed; exit 0 | New reviewer controls |
| P-INDEPENDENT-CRYPTO-EARLY | 4 passed; exit 0 | New reviewer controls |

The four Linux protocol suites passed 72 top-level tests. Ten additional reviewer top-level controls are separate runs, not 82 disjoint comprehensive security scenarios. The full Linux guest took 74.391 seconds including app tunnel. The singlink suite executed actual concurrent child-process and process-exit tests. The exit fixture calls os.Exit(0) after successful CheckAndMark acknowledgement; no mid-write/fsync SIGKILL or power loss was injected.

| Package | Executed coverage / statements |
|---|---|
| singlink | 82.80% (1261/1523) |
| engine | 52.44% (86/164) |
| socks5 | 75.78% (244/322) |
| cmd/singlink | 0.00% (0/258) |

Coverage is instrumented package statements, not vulnerability detection probability. The passing CLI control covered no main-package statements; full CLI coverage is not claimed.

| Target | Final elapsed | Executions | New interesting |
|---|---|---|---|
| P-FuzzDecodeAddr | 20 s | 576,239 | 7 |
| P-FuzzParseHeader | 20 s | 483,215 | 15 |
| P-FuzzParseToken | 20 s | 679,330 | 3 |
| P-FuzzParseLink | 21 s | 3,258 | 3 |
| P-FuzzParseFirstFlight | 21 s | 3,831 | 1 |
| P-FuzzReadFrame | 20 s | 712,716 | 7 |

Each target requested 20 seconds with two workers; final counters report about 20-21 seconds. Total: **2,458,589 executions**, all exit 0. Initial compilation adds wall time. “New interesting” means coverage corpus additions, not vulnerabilities. No crash in this budget does not predict unlimited fuzzing results.

go vet and govulncheck for Windows and Linux exited 0 with no finding events. govulncheck used source/symbol scanning; database last-modified was 2026-11-11T02:51:11Z. The protocol module uses the standard library

only. Zero findings excludes neither unknown vulnerabilities nor design mistakes and says nothing aboutarbitrary commercial binaries.

P-VET-WINDOWS; P-VET-LINUX; P-VULN-WINDOWS; P-VULN-LINUX; P-MODULES

## 8. Failures, limitations and unexecuted work

The original Windows protocol baseline exited 1: 29 pass / 43 fail. Many fixtures require the Unix-only FileReplayStore, which intentionally fails closed on Windows. This is neither 43 confirmed vulnerabilities nor an all-pass run. A memory adapter was used only in reviewer controls, never proposed for deployment; subsequent Linux execution used the original persistent backend. The first QEMU launch could not locate BIOS under a CJK Windows path; an ASCII temporary path resolved it, with the failure retained.

There is no successful race-detector execution evidence, physical power-loss test, NFS/multi-host test, production key-custody/control-plane validation, real NAT64, macOS backend or commercial end-to-end service test. Reviewer callback/first-flight controls used synthetic keys and loopback. Cryptographic review was not a formal proof.

P-BASELINE-WINDOWS; ENV-LINUX-GUEST-RUN

## 9. Recommended order and closure criteria

1. Before release, scope Draft 4 compatibility wording and verify actual client/server/FFI pairing without downgrade. Resolve the app report A-04 FD ownership contract alongside this integration.

2. Validate crash/restart/rotation/capacity/clock rollback on production-like Unix storage and a non-root identity. Add physical power-loss and backup-restore exercises, multi-host external-store contract tests and real admission benchmarks. RAM-root results do not substitute for these.

3. Run a compatible native race detector, sustained fuzz/load tests, a real NAT64 path and revocation control-plane acceptance. Windows nodes need a supported store.

4. Retain failed runs and their corrections. Vendor APFS timings, internal 31-findings counts and third-party wording are not measurements or certifications from this assessment.

This review is complete within its stated scope. The items above are release/operational acceptance criteria, not claimed executions. Product code was not modified. Do not restate this conclusion as comprehensive safety of all versions, absence of plaintext, or proof of server non-recording.

## 10. Evidence package and reproduction

The English and Traditional Chinese editions are separate files with matching finding IDs, commits, measurements and limitations. HTML/Markdown are editable; PDF is the reading edition. Start at START_HERE.html; see check_summary.json for the full matrix.

The share package contains the new reports, public source snapshots/LICENSE/NOTICE, synthetic tests, outcomes and diff images. It excludes accounts, paid credentials, personal browsing packets, tool caches and production signing keys. Dependency downloads requested public materials from official/package sources; OSV received package names/versions only. This does not mean the whole assessment was offline: only the Linux guest had no external network.

Original logs remain local. Share copies consistently replace local username/workspace paths; redactions.json records original/share hashes. Source bytes remain unchanged. SHA256SUMS.txt verifies delivered contents. Toolchains, QEMU/SDK/module caches are excluded rather than republishing third-party tools as evidence.

Reproduction requires the specified tools, sibling protocol checkout and paths described inREADME_REPRODUCE.md; this is not an arbitrary-machine one-click harness.

Recorded check IDs and exit codes:

| ID | Exit | Note |
|---|---|---|
| P-BASELINE-WINDOWS | 1 | Read scope / failure explanation |
| P-BUILD-LINUX-CLI | 0 | Bounded check passed |
| P-BUILD-LINUX-ENGINE | 0 | Bounded check passed |
| P-BUILD-LINUX-SINGLINK | 0 | Bounded check passed |
| P-BUILD-LINUX-SOCKS5 | 0 | Bounded check passed |
| P-FuzzDecodeAddr | 0 | Bounded check passed |
| P-FuzzParseFirstFlight | 0 | Bounded check passed |
| P-FuzzParseHeader | 0 | Bounded check passed |
| P-FuzzParseLink | 0 | Bounded check passed |
| P-FuzzParseToken | 0 | Bounded check passed |
| P-FuzzReadFrame | 0 | Bounded check passed |
| P-INDEPENDENT-CRYPTO-EARLY | 0 | Bounded check passed |
| P-INDEPENDENT-RETEST | 0 | Bounded check passed |
| P-LINUX-CLI | 0 | Bounded check passed |
| P-LINUX-ENGINE | 0 | Bounded check passed |
| P-LINUX-SINGLINK | 0 | Bounded check passed |
| P-LINUX-SOCKS5 | 0 | Bounded check passed |
| P-MODULES | 0 | Bounded check passed |
| P-VET-LINUX | 0 | Bounded check passed |
| P-VET-WINDOWS | 0 | Bounded check passed |
| P-VULN-LINUX | 0 | Bounded check passed |
| P-VULN-WINDOWS | 0 | Bounded check passed |

### Redistribution and attribution

Source LICENSE/NOTICE are retained, including the original-author SingLinkVPN attribution requirement. Its legal application was not independently assessed. Report authorship is an AI-assisted assessment commissioned by the user; vendor “third-party audit” wording is not used to endorse this work. Preserve scope, revision, residual findings and evidence links when publishing.