SNAPSMACK SECURITY AUDIT

Audit F — Post-Release Code Quality & Tooling Review | Alpha 0.7.29

Date	2026-04-29	**Auditor**	Claude (Anthropic Sonnet 4.6)
Scope	0.7.29 release verification, inline code audit, tooling scaffold	**Version**	Alpha 0.7.29 "Lock-Off"
Prior Audit	Audit E — 2026-04-29 (login hardening)	**Status**	Complete

1. Release Verification (0.7.29)

Full integrity check on the 0.7.29 release files. All new and modified files were verified for truncation, duplicate includes, prepared statement usage, and proper HTML/JS structure.

PASS **File Integrity — snap-in.php** **PASS**

Location: `snap-in.php`

374 lines. Ends with </html>. No truncation. Duplicate CSS link tags from reconstruction were identified and removed — head section now contains exactly one geometry master link and one skin CSS link.

PASS **File Integrity — All New 0.7.29 Files** **PASS**

Location: `migrations/045, smack-2fa-verify.php, assets/js/*`

All six new/modified files verified: smack-passphrase.js ends with closing brace (line 73). ss-engine-updater.js ends with IIFE closure }()); (line 460). migrations/045_login_protection.php ends with execute() call. smack-fingerprints.php ends with admin-footer include. smack-2fa-verify.php ends with </html>. No truncation found in any file.

PASS **Prepared Statements — snap-in.php** **PASS**

Location: `snap-in.php lines 50-78, 123-126`

All five database queries in snap-in.php use parameterised statements with ? placeholders: snap_rate_limits INSERT/SELECT/DELETE and snap_ip_bans INSERT and SELECT. No string interpolation in query construction.

PASS **Prepared Statements — migrations/045** **PASS**

Location: `migrations/045_login_protection.php`

Both queries use prepared statements: existence check and INSERT IGNORE. Fixed from Audit E finding where raw string interpolation was present.

INFO **LIMIT/OFFSET Interpolation — smack-fingerprints.php** **INFO**

Location: `smack-fingerprints.php lines 135, 176, 219`

Three queries use string interpolation for LIMIT/OFFSET values. PDO does not support binding LIMIT/OFFSET with ? placeholders reliably across all drivers. The correct mitigation is explicit integer casting — values are calculated from hardcoded $per_page=50 and integer math on $page. Risk is LOW. No action required; pattern is correct for this use case.

PASS **HTML Closing Tags — smack-fingerprints.php IP SMACKER tab** **PASS**

Location: `smack-fingerprints.php lines 374-384`

IP SMACKER tab panel HTML verified: .tab-content, .box, and #ip-smacker-list all have correct closing tags. No missing or mismatched tags.

2. Naming Consistency — IP SMACKER

The admin feature was renamed from IP Shield to IP SMACKER for brand consistency with the other security layer names. All references updated this session.

FIXED **IP Shield references in smack-fingerprints.php** **FIXED**

Location: `smack-fingerprints.php`

All IP Shield references updated to IP SMACKER: tab button label, tab comment, div id (ip-shield → ip-smacker), h3 heading, list element id, JS tab switch condition, and JS section comment. Verified: grep finds zero remaining 'IP Shield' or 'ip-shield' strings.

FIXED **IP Shield references in smack-help.php** **FIXED**

Location: `smack-help.php`

Help topic title updated to 'IP SMACKER & Login Security'. All three body references (h3 heading, Troll Control navigation reference, lift-ban instruction) updated to IP SMACKER.

FIXED **IP SHIELD in snapsmack.ca index.html** **FIXED**

Location: `projects/snapsmack-ca/index.html`

Layer 7 heading updated to IP SMACKER. Lede updated to reference 'IP SMACKER admin login hardening'. Layer numbering updated to 8 layers with SMACKATTACK as its own dedicated Layer 4 card.

3. Inline JS/CSS Technical Debt Audit

Full codebase scan for inline <style> and <script> blocks in PHP files, measured against the architecture rule: no inline styles, no inline scripts (documented exception: skin-header.php PHP-conditional overrides).

DEBT **Pre-existing Inline Style Blocks** **NOTED**

Location: `~44 locations across codebase`

Approximately 44 inline <style> blocks found. Notable concentrations: core/meta.php (dynamic CSS output — architectural, required for runtime PHP values), install.php (installer UI), smack-edit.php, smack-skin.php, smack-post-carousel.php, sc-* admin pages. These are pre-existing technical debt, not 0.7.29 regressions. The architecture rule applies to new code going forward.

DEBT **Pre-existing Inline Script Blocks** **NOTED**

Location: `~32 locations across codebase`

Approximately 32 inline <script> blocks found. Notable: smack-albums.php (featured posts picker XHR), smack-settings.php (Akismet key test fetch()), smack-mosaics.php (mosaic asset picker), smack-manage.php (batch select logic), gallery-wall.php (mobile redirect). core/meta.php window.SMACK_CONFIG injection is architectural (PHP→JS data bridge). Pre-existing debt, not regressions.

INFO **Skin Archive Script Blocks** **INFO**

Location: `skins/50-shades-of-noah-grey/archive-layout.php, skins/rational-geo/archive-layout.php`

Two skin archive-layout.php files contain inline <script> blocks for skin-specific layout logic. These should eventually move to the skin's declared JS engine in the manifest system. Lower priority than admin page cleanup.

Remediation plan: inline code debt is a long-term refactoring track, not an emergency. Priority order when addressing: (1) admin pages with substantial form logic, (2) skin archive scripts, (3) install/setup pages, (4) dynamic CSS blocks in meta.php (last — these are architectural).

4. Changelog and Help System Coverage

PASS **CHANGELOG.md — 0.7.29 Entry Completeness** **PASS**

Location: `CHANGELOG.md`

All 0.7.29 changes verified present in CHANGELOG.md: snap-in.php reconstruction, UA filter, IP ban gate, migration 045, canonical schema update, IP SMACKER (formerly IP Shield) tab, help topic, tooltip conversion across 16 admin pages, 2fa-verify truncation fix, CSS/JS new files, SMACKATTACK scanner tool, wotcha.html articles, version bumps.

PASS **Help System — 0.7.29 Feature Coverage** **PASS**

Location: `smack-help.php`

All four required help topics verified present with substantial content: ip_shield/ip_smacker (30 lines — login hardening, auto-ban, recovery), updater_modal (40 lines — 5-stage process, rollback), smackback (31 lines — file integrity, BREACH skin), gobsmacked (128 lines — stylometric vectors, cluster analysis, privacy).

5. Claude Commander Scaffold

Initial Python sidecar scaffold committed to tools/claude-commander/ (gitignored, development only). Security review of the scaffold architecture.

PASS **Credential Storage Architecture** **PASS**

Location: `python-sidecar/remote_ops.py`

All credentials retrieved from OS keychain via keyring library. No credentials stored in swapfile, plaintext files, or code. Credential service names are namespaced (claude-commander-ssh:host, claude-commander-ftp:host). get_credential() raises CredentialError if keyring unavailable rather than falling back to plaintext.

PASS **Project Scope Enforcement** **PASS**

Location: `python-sidecar/action_executor.py _assert_in_scope()`

Hard boundary enforced via Path.resolve().relative_to() check. FileWrite and FileDelete operations raise PermissionError if target path resolves outside the registered project directory. Cannot be bypassed by symlinks or relative path tricks — resolve() follows symlinks before comparison.

PASS **Protected Branch Push Gate** **PASS**

Location: `python-sidecar/git_ops.py push()`

Push to main, master, release, production, prod branches raises PermissionError unless require_confirmation=False is explicitly passed. require_confirmation=False is only set by action_executor.py when processing an operator-approved queue item. Direct push() calls default to require_confirmation=True.

PASS **Audit Log Append-Only Design** **PASS**

Location: `python-sidecar/swapfile.py audit()`

audit() opens file with mode 'a' (append). No method exists to modify or delete audit entries. Each entry is a JSON line with timestamp, session ID, operation, target, and outcome. Log is written before execution (pending) and after (ok/failed) for pre-execution accountability.

PASS **Destructive Operation Safeguards** **PASS**

Location: `python-sidecar/action_executor.py`

file_delete requires confirmed=True in action params — raises NeedsReview if absent. git_remove_index_lock requires explicit call, not automatic. PANIC command halts all operations and writes current queue to swapfile before stopping. All destructive operations logged before execution.

INFO **Bridge Read Audit Trail** **INFO**

Location: `python-sidecar/action_executor.py _bridge_read()`

Bridge reads from other projects are logged to audit.log with the source project name and file list. The source project scope check ensures the source path is a registered project (not arbitrary filesystem access). Hard rule enforced: bridge is always read-only from source.

6. Carry-Forward from Previous Audits

HIGH **CSRF — Deferred from Audit C** **DEFERRED**

Location: `All admin POST handlers`

SameSite=Lax partial mitigation in place. Full CSRF token implementation not yet scheduled. This remains the highest-priority unaddressed security item. No new POST surfaces were added in 0.7.29 that worsen exposure.

Summary

Category	Result
Release integrity (0.7.29)	All files clean
Naming consistency (IP SMACKER)	All references updated
Changelog coverage	Complete
Help system coverage	All topics present
Inline code debt	76 pre-existing violations — documented, not regressions
Commander scaffold security	All checks pass
CSRF (carry-forward)	Deferred HIGH — unchanged