SnapSmack Security Audit

Audit #010 · Version 0.7.152 · 2026-05-18 · Scope: 0.7.147–0.7.152 release delta

---

Executive Summary

This audit covers security-relevant changes introduced between 0.7.146 and 0.7.152 (five commits ahead of GitHub/master at time of review), plus two targeted fixes applied during this session: skin exclusion corrections in sc-release.php and null-byte removal from smack-help.php. Seven items were examined. No exploitable vulnerabilities were identified. Two LOW housekeeping items were resolved. All database writes use parameterised queries. All new session values follow the established load-at-login pattern.

Scope

File / Endpoint	Role
`login.php`	Auth/session bootstrap — loads ui_mode from snap_users
`smack-admin.php`	Per-user ui_mode switch (Pimpmobile / Big Wheel offer)
`core/sidebar.php`	Reads ui_mode from $_SESSION for nav rendering
`core/updater.php`	Version comparison fix; migration registration
`install.php`	New Step 0 security-advice page; step default change
`archive.php`	Variable rename fix ($archive_layout to $grid_layout)
`smack-update.php`	Flex layout for stage-action buttons (CSS only)
`smack-central/sc-release.php`	always_exclude additions — 4 non-release skins
`smack-help.php`	Auto-advance prose update; null-byte tail removed

Findings

SS-010-01 INFO CLEAN ✓ Per-user ui_mode — session assignment and DB writes

ui_mode was migrated from site-wide snap_settings to snap_users (per user). The value is loaded into $_SESSION at login and written via UPDATE on the authenticated user's own row. Only the two hardcoded literals 'pimpmobile' and 'bigwheel' are ever written. Both the SELECT at login and the UPDATE in smack-admin.php are fully parameterised.

Before

```
// site-wide — any admin's toggle affected all users $_ui_pimpmobile = ($settings['ui_mode'] ?? 'bigwheel') === 'pimpmobile'; $pdo->prepare("INSERT INTO snap_settings (setting_key, setting_val) VALUES ('ui_mode','pimpmobile') ON DUPLICATE KEY UPDATE setting_val=VALUES(setting_val)")->execute();
```

After

```
// per-user — read from session (set at login from snap_users.ui_mode) $_ui_pimpmobile = ($_SESSION['user_ui_mode'] ?? 'bigwheel') === 'pimpmobile'; // parameterised UPDATE on authenticated user's own row $pdo->prepare("UPDATE snap_users SET ui_mode = 'pimpmobile' WHERE id = ?") ->execute([$_SESSION['user_id']]);
```

No injection risk. Value constrained to enum literals. Clean.

SS-010-02 INFO CLEAN ✓ Installer step default (1 -> 0) and new Step 0 security page

The installer's default step changed from 1 to 0. Step 0 is a static security-advice page (password hygiene, 2FA recommendation) with no user-supplied output — all content is hardcoded HTML. Advancing to Step 1 uses a hidden form field value="1", consistent with the existing step-progression pattern. The installer's lock checks are unaffected.

Before

```
$raw_step = $_POST['step'] ?? 1; // default: land on env check
```

After

```
$raw_step = $_POST['step'] ?? 0; // default: land on security advice page // Step 0: hardcoded HTML only — no user input echoed
```

Positive change: new installs are prompted for strong password and 2FA before proceeding.

SS-010-03 INFO CLEAN ✓ updater_check_status() — false-positive update fix

The updater's checksum comparison previously ran whenever versions were equal. The fix adds an explicit equality guard so the comparison only fires when the remote version is neither greater nor less than installed — preventing a stale remote server from triggering a spurious "update available" prompt, which could lead a user to install an older or unintended package. hash_equals() usage preserved.

Before

```
if ($installed_checksum !== '' && !empty($release_info['checksum_sha256']) && !hash_equals(...)) { return 'update_available'; // fired even when remote < installed }
```

After

```
$versions_equal = !snap_version_compare($release_info['version'], $installed_version, 'lt') && !snap_version_compare($release_info['version'], $installed_version, 'gt'); if ($versions_equal && $installed_checksum !== '' && ... && !hash_equals(...)) { return 'update_available'; // only fires on true version equality }
```

Clean. Correctness fix with positive security implication.

SS-010-04 LOW RESOLVED ✓ sc-release.php: four non-release skins shipping in base package

chaplin, galleria, photogram, and rational-geo were absent from the always_exclude list and were bundled into every base release zip. rational-geo includes a large SVG asset. Shipping unintended files increases attack surface (extra PHP entry points) and package size. Fixed during this session by adding all four to always_exclude.

Before

```
$always_exclude = [ 'skins/52-card-pickup/', 'skins/a-grey-reckoning/', // chaplin, galleria, photogram, rational-geo absent ];
```

After

```
'skins/true-grit/', // Skins distributed via Skin Packager only 'skins/chaplin/', 'skins/galleria/', 'skins/photogram/', 'skins/rational-geo/',
```

SS-010-05 LOW RESOLVED ✓ smack-help.php: null bytes appended after EOF marker

smack-help.php contained a run of null bytes (\x00) after the SNAPSMACK EOF marker, with no trailing newline. Null bytes in PHP source are a known vector for path traversal and file-inclusion attacks on older PHP versions. While PHP 8.x handles embedded nulls more robustly, their presence is anomalous. Removed during this session; file now terminates cleanly with a single newline after the EOF marker.

Before (tail -5 | cat -A)

```
<?php include 'core/admin-footer.php'; ?>$ <?php // ===== SNAPSMACK EOF =====$
^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@^@ (no newline at EOF)
```

After (tail -3 | cat -A)

```
<?php include 'core/admin-footer.php'; ?>$ <?php // ===== SNAPSMACK EOF =====$
```

SS-010-06 INFO CLEAN ✓ archive.php — $archive_layout renamed to $grid_layout

Variable rename for naming consistency. No logic change, no new user input. Value originates from $settings (DB-loaded). No security impact.

SS-010-07 INFO CLEAN ✓ smack-update.php — flex layout for stage-action buttons (CSS only)

CSS-only change. No PHP logic touched, no new input surfaces, no template output changes. No security impact.

Summary

ID	Severity	Status	Title
SS-010-01	INFO	CLEAN	Per-user ui_mode session/DB pattern
SS-010-02	INFO	CLEAN	Installer Step 0 security advice page
SS-010-03	INFO	CLEAN	Updater version comparison fix
SS-010-04	LOW	RESOLVED	Four non-release skins in base package
SS-010-05	LOW	RESOLVED	Null bytes after EOF marker in smack-help.php
SS-010-06	INFO	CLEAN	archive.php variable rename
SS-010-07	INFO	CLEAN	smack-update.php flex layout (CSS only)

No exploitable vulnerabilities identified. Two LOW housekeeping items resolved. Codebase is clear for the 0.7.152 launch.

Audited by Claude (Anthropic) · SnapSmack co-author · 2026-05-18