SnapSmack Security Audit

Audit E — Login Hardening & 0.7.29 Changes | 2026-04-29 (rev 2: LOW items fixed)

Scope

Changes shipped in SnapSmack 0.7.29 "Lock-Off": three-layer login hardening in **snap-in.php**, the **snap_ip_bans** table and migration, IP Shield tab in smack-fingerprints.php, and the GOBSMACKED Scanner companion tool. UX-only changes (tooltip conversions, HTML reconstruction) noted but not individually audited.

Executive Summary

The login hardening adds three meaningful layers to the existing slug-obscurity approach: UA filtering, rate-limited auto-banning, and an admin-visible IP ban list. Implementation is sound. Two LOW findings (raw string interpolation in migration, undocumented XFF trust assumption) were fixed in the same session. CSRF protection remains the open deferred HIGH from prior audits — no new surface added in 0.7.29 but the IP lift handler is another unprotected POST endpoint.

Findings

PASS	**UA filter: regex anchoring and scope**	NO ACTION
	snap-in.php — $_ua_bot_rx	
	Regex uses # delimiter, i flag, and correct ^$ anchor for blank UAs. Alternation list covers all major scripting clients. No injection surface — UA is matched only, never echoed. Legitimate browsers not caught by any arm.	
FIXED	**X-Forwarded-For: single-hop trust assumption undocumented**	FIXED 2026-04-29
	snap-in.php — snap_client_ip()	
	The function trusts the leftmost XFF hop (correct for single Cloudflare proxy). Risk: if a second untrusted proxy is added upstream the trust model breaks. **Fixed:** added explicit comment documenting the single-proxy assumption and flagging the revisit condition (validate against Cloudflare IP ranges if topology changes).	
PASS	**Brute-force window reset logic**	NO ACTION
	snap-in.php — snap_record_login_failure()	
	ON DUPLICATE KEY UPDATE resets count to 1 (not count+1) when window_start is stale — correct. Window comparison consistent. Ban INSERT uses ON DUPLICATE to extend rather than error. Counter cleared after ban issuance.	

FIXED	**Migration 045: raw string interpolation in prepared-statement context**	FIXED 2026-04-29
	migrations/045_login_protection.php	
	Migration name check used: SELECT ... WHERE migration_name = '$migration_name'. Variable is hardcoded (no user input), so no injection risk existed. However the pattern is inconsistent and unsafe to copy-paste. **Fixed:** both the existence check and the INSERT now use PDO prepared statements with ? placeholders.	
PASS	**IP ban gate: fail-open on missing table**	NO ACTION
	snap-in.php — IP ban gate try/catch	
	Ban check wrapped in try/catch(PDOException), fails open — older installs without the snap_ip_bans table continue to reach the login form. Correct: gate must not lock out legitimate users on installs that have not yet run migration 045.	
PASS	**IP Shield AJAX: auth, authorization, SQL**	NO ACTION
	smack-fingerprints.php — fetch_ip_bans / lift_ip_ban	
	Both handlers inside admin-auth guard. lift_ip_ban uses prepared statement DELETE FROM snap_ip_bans WHERE id=?. fetch_ip_bans uses a parameterless query with LIMIT 200. No injection surface.	
PASS	**snap_ip_bans schema integrity**	NO ACTION
	migrations/045_login_protection.php — CREATE TABLE snap_ip_bans	
	UNIQUE KEY uq_ip(ip) prevents duplicate ban records. ON DUPLICATE KEY UPDATE in the ban INSERT extends existing bans. idx_expires supports efficient expiry queries.	
INFO	**GOBSMACKED Scanner: credentials in plaintext INI**	ADVISORY
	tools/smackattack-scanner/config.py — gobsmacked-scanner.ini	
	MySQL password and API key stored in plaintext .ini next to the exe. Acceptable for a local desktop admin tool (not a web component, not committed to git). User should be aware the file is not encrypted. No action required unless multi-user desktop access is a concern.	

HIGH	**CSRF protection: deferred from Audit C**	DEFERRED
	All admin POST handlers — smack-.php*	
	No CSRF token validation on any admin form. SameSite=Lax provides partial mitigation. Carried forward from Audit C. No new attack surface in 0.7.29, but lift_ip_ban is another unprotected POST handler. Recommend per-session CSRF token in core/auth.php validated on all admin POST handlers. Complexity: medium. Priority: HIGH before public/open-source release.	

PASS	**snap-in.php HTML: XSS surface**	NO ACTION
	snap-in.php — form output	
	All user-supplied values in form output use htmlspecialchars(). Error string escaped. No raw output of user input.	

Action Items

Priority	Item	Status
HIGH (deferred)	Implement CSRF tokens on all admin POST handlers	Deferred — pre-release
LOW → FIXED	Prepared statements in migration 045	Fixed 2026-04-29
LOW → FIXED	Document XFF single-proxy assumption in snap_client_ip()	Fixed 2026-04-29
ADVISORY	Note plaintext credential storage in Scanner README/help	Open

---

Auditor: Claude (Anthropic) | SnapSmack 0.7.29 "Lock-Off" | 2026-04-29 | Audit series: A B C D (prior) E (this document, rev 2)