Mad Ad Slingers — Full Platform Audit

Friday, November 20, 2026 · Ordered by the owner. Six workstreams: full audit, security test, OWASP ZAP scans, smoke test, memory tests (Elephant + Goldfish), and competitive ratings. Everything was **read-only** against the live platform — no code was changed, nothing was restarted, nothing was sent, bought, or generated for a fee. The one exception: the Elephant memory test taught the Brain one nonsense test fact, on purpose, to see whether it could be remembered (details in §5). Findings only — every fix is a separate decision for the owner.

The short version

Security: every hole found in the November 12 security audit that could be re-tested is **fixed and verified live**. This audit found **0 Critical and 0 High** security problems. One Medium functional bug: the Stripe checkout-status check returns an error for every real checkout session (root cause found; payments themselves are unaffected).

ZAP scans: real OWASP ZAP scans of both public sites — **0 High, 0 Critical** alerts on either. Everything found is hardening-level (content-security-policy tightening, script integrity tags).

Smoke test: 7 of 8 core flows pass. The one failure is the known Slingy `/brain` public route (404 — the pending server paste). A separate addendum: **8 of the videos in the owner's Creatives Gallery are dead files** and cannot play.

Memory tests: both **failed**, honestly. The Brain stores teachings it is then not allowed to read back (a wiring defect — the November TikTok and Pinterest teachings are stuck the same way), and the portal chat has no working in-conversation memory. Nothing leaked between sessions.

Ratings: 4.7 / 10 overall (plain average of seven categories) against Smartly, Madgicx, AdEspresso, and Birch (formerly Revealbot). Strongest: pricing/value (7). Weakest: market traction (1 — zero paying customers so far, stated plainly).

Site content: no broken links anywhere (all 71 pages, 87 link targets, 48 assets return 200). Pricing is complete on almost every surface; the ChatGPT Ads page is the one real failure. Platform-status wording is stale in several places — worst, the Microsoft Ads page says "not available yet" about a platform that is fully working.

1 · Full audit — every page and feature

Method. Every page in the site map (71 URLs on mad-ad-slingers.com) was fetched live; every internal link target (87) and every internal file the pages load (48) was checked. Portal and admin findings come from the source code plus the actual JavaScript files served to visitors, which were fetched and searched. Ground truth used throughout: the catalog is **40 platforms, 9 Working** (Google, Meta, LinkedIn, Pinterest, Snapchat, Reddit, Vibe, YouTube, Microsoft); TikTok, X, Nextdoor, Spotify and eBay are in progress.

Broken links — clean

All 71 site-map pages return HTTP 200. All 87 internal link targets and all 48 internal assets return 200. Pages for the six platforms removed on Nov 17 (Perplexity, Etsy, AdMob, Angi, Pandora, iHeart) correctly return 404 and are not linked from anywhere. **Zero broken internal links.** One coverage gap, not a breakage: there is no dedicated YouTube marketing page, even though YouTube is a Working platform.

Stale copy — the retired "Magic Ad Maker" name

- **Marketing site: zero occurrences** in all 71 live pages and in the site source. Clean, as required.
- **Portal source:** one leftover page file still titled "Magic Ad Maker" — but that page is not routed anywhere and does not ship to visitors (dead code; see "Dead pages" below).
- **Backend:** the label "Magic Ad Maker" still sits in a live API feature list (`coming_soon.py`), so it can surface in the portal's "coming soon" gate wording. This is the most live of the remnants.
- **Admin tree's older embedded copy of the portal:** a credit-cost explainer still says "Magic Ad Maker". The admin tree carries a second, older copy of the portal — that drift is how stale labels survive in one place after being fixed in another.

- Two mentions are intentional and correct: Slingy's knowledge base explains the rename ("Magic Ad Maker is the old name — those tools are now called the Image Generator and Video Generator"), and the admin app redirects the old `/magic-ad-maker` address to the new one.

Stale copy — platform status wording (live, checked page by page)

Where	What it says	The truth	Verdict
Microsoft Ads page	"In Development … not available yet"	Microsoft is **Working** (connected and proof-tested Nov 20)	**Worst error on the site**
Snapchat Ads page + homepage card + site-wide tagline "(TikTok, X and Snapchat Ads rolling out)"	"Now Rolling Out" / "Rolling out"	Snapchat is **Working**	**Stale**
Reddit Ads page + homepage "Coming Next" strip	"Now Rolling Out" / listed as "in development"	Reddit is **Working**	**Stale**
Madison page	"seven live today, with TikTok and X coming soon"	9 Working	**Stale count**
Portal browser-tab description	"run campaigns across 10+ ad platforms"	40-platform catalog	**Badly understates**
Slingy's built-in knowledge (portal)	"We publish to 7 platforms" — then lists 6, includes in-progress TikTok, omits Working Reddit, Vibe, YouTube, Microsoft	9 Working	**Slingy will quote this to clients**
Spotify / eBay pages; Nextdoor homepage card	"In Development / not available yet" / "Coming soon"	All three are in progress	**Understates**
TikTok, X, Nextdoor pages; Google, Meta, LinkedIn, Pinterest, Vibe pages; all coming-soon and adult-network pages	Statuses as written	Match the truth	**Correct**

The "40 platforms" count itself is correct everywhere it appears, and the backend catalog math checks out exactly: 40 ad platforms with statuses matching the truth.

The known Snapchat label bug — fixed

The portal Campaigns page's built-in launch list now says `snapchat: working` — verified in the source **and** in the JavaScript actually served to visitors. One residual gap: that built-in fallback list has only 7 platforms and omits Working Reddit, Vibe and Microsoft. It is backstopped by a live-catalog merge and by the deploy picker list, which is fully correct (9 Working / 5 in progress / 26 coming soon) — so the risk only materialises if the live merge ever fails.

Pricing completeness — the rule: every price appears everywhere pricing shows

The full set is 12 items: Base $49/mo, Unlimited $250/mo, Agency $399/mo, Managed $999/mo, the 5% AI fee, creative fees $10/mo per image and $25/mo per video, Studio Pass $50/mo, and the ChatGPT Ads plans ($499 one-time, $697/mo, $997/mo, $3,000/mo).

Surface	Result
Homepage, /pricing, /compare, all 7 trade-niche pages, /madison, /advertising-costs-explained,	**All 12 prices present**

/what-is-an-ai-ad-agency, portal Pricing page	
/chatgpt-ads	**The one clear failure. Micro Pilot and AI Search Starter are named and described with no price at all ($997 and $3,000 appear nowhere). Agency $399, Managed $999, the 5% fee, creative fees and Studio Pass are also absent. The page does point readers to the pricing page — a deferral, not completeness.**
/ad-cost-comparison	**Shows the *platforms'* prices for all 40 platforms and none of Mad Ad Slingers' own 12. The page says so itself and defers to the pricing page. Judgment call for the owner under the strict rule.**
Portal ChatGPT Ads tab (and the admin equivalent)	**Shows only the 4 ChatGPT prices — by design it is the dedicated ChatGPT tab, with the full pricing page one tab away. Flagged under the strict rule for the owner's call.**

Dead pages and placeholders

- **Portal:** one orphaned page file (`MCPAdGeneratorPage` — the "Magic Ad Maker" remnant above) is not routed anywhere and does not ship. The new UGC Ads page contains a "Built-in presenters" section that is three dashed boxes literally reading "Coming soon" — a visible placeholder on an otherwise real page. Every other portal tab maps to a substantive page (41 page components).
- **Admin:** about a dozen page-shaped files are not referenced by the router (AdStudio, AgenciesPage, AdminCreativeLibrary, AdBudgets, AdChatbot, BotQuotas, CollectionsView and similar) — they look like superseded pages. Unrouted means unreachable, so this is cleanup debt, not visitor-facing breakage. No routed admin page was found to be an empty stub.

Mobile rendering — not fully testable here

Plainly: this audit had no live browser, so **no page was actually rendered at phone width**. Static checks only: the viewport tag is present on all 71 pages; the stylesheet's large fixed widths are mostly guarded by media/container queries or scroll-wrapped; the homepage hero and demo video scale fluidly. The known sideways-scroll class of bug (tickers, transforms, absolutely-positioned elements) **cannot be confirmed or cleared without rendering**. A rendered pass at 390px on the homepage, /pricing, /ad-cost-comparison (its table is the highest overflow risk) and /chatgpt-ads is still owed — it is listed in §8.

Content-Security-Policy compliance

The marketing site's CSP deliberately allows **no inline styles** (the hard lesson of Nov 7–8). Checked across all 71 served pages: **zero `<style>` blocks**, and exactly **one dead inline style** — `style="margin-bottom:16px"` on a shared header paragraph, present on the homepage, /pricing and /madison. Impact is cosmetic only (that one paragraph loses its bottom margin in real browsers); it is reported because it is exactly the pattern that caused the November slogan fight.

2 · Security test

Method. Every hole from the November 12, 2026 security audit was re-tested live, plus security headers on all hosts, unauthenticated probes of sensitive endpoints, and targeted checks the owner named. Rules: no credential attacks, no valid commands sent to any command endpoint — only empty, wrong-secret and unauthenticated probes. No secret values are printed anywhere in this report; presence only.

The November 12 holes — all fixed

Nov 12 finding	Then	Now (re-tested live Nov 20)
Server logs readable by anyone	**Critical**	**FIXED — every log path now returns 401 without staff login; 200 with it. No web path to the log files exists in the server config.**
Unlock endpoint could mint a superadmin	**Critical**	**FIXED — welded shut, as ordered. The PIN (HUD_PIN) is not set anywhere, so the endpoint refuses everyone: "PIN unlock is not**

token		**configured." Token life was also cut from 365 days to 12 hours in code.**
Phone-command webhook needed no password	**High**	**FIXED — closed at two independent layers: the CSRF middleware stops unsigned calls (403), and the handler itself rejects a wrong secret (401) before reading any command. The webhook secret is set.**
Client reports readable across accounts (IDOR)	**High**	**FIXED IN CODE — every report endpoint now checks ownership and returns 404 to non-owners; unauthenticated requests get 401. A live cross-account read could not be proven in this run (needs a second client's login — see §8).**
Operations numbers (HUD / metrics / events) public	**Medium**	**FIXED — all return 401 unauthenticated; 200 with a staff token.**
Public payment lookup	**Medium**	**FIXED — lookup and status endpoints now return 401 unauthenticated.**
MFA secrets stored/returned unsafely	**Medium**	**FIXED IN CODE — MFA secrets are now Fernet-encrypted at rest, and the admin client view no longer includes them (or password hashes).**
Encrypted client keys visible in dashboard payloads	**Low**	**Code shows the fields are now excluded from the serializer; not re-proven live (no client login available to this run).**

Also verified: no `.env` file is reachable on any host (404/401); EMAIL_OUTREACH_PAUSED is still `true`; the Stripe, cron and SMS secrets are all present (values never read).

New findings from this run

Severity	Finding
Critical	**None.**
High	**None.**
Medium	**Stripe checkout-status check is broken for every real session.** Asking the backend "did this checkout get paid?" (`GET /api/stripe/checkout/{id}`) returns a bare 500 for genuine sessions — reproduced with both ChatGPT Ads sessions created the night of the audit. Root cause pinned: the Stripe library now returns the session's metadata as a `StripeObject` instead of a plain dictionary, and the response serializer chokes on it *after* the handler succeeds — so the handler's own error handling never sees it. Impact: any page polling payment status after checkout gets an error. **No money is affected** — checkout creation works and payment completion flows through the separate, verified Stripe webhook. Fix direction (not applied — read-only audit): convert the metadata to a plain dictionary where the session is read back.
Low	**Agreement gate fails open on internal error.** If the agreement check itself throws an error, the request is allowed through (a deliberate design choice, unchanged since November). The gate is otherwise correctly scoped — blocked users get 403, the allowlist is small. The risk only materialises if the gate errors while an unaccepted user is calling.
Low	**The vps subdomain serves a thinner header set.** Responses served directly by Caddy on vps.mad-ad-slingers.online lack X-Frame-Options, Referrer-Policy and Permissions-Policy (the API responses behind it carry the full set). The host serves no rich app of its own, so impact is low today.
Info	Six observations, none exploitable: the ChatGPT Ads plans endpoint is documented as public but in practice requires login (it fails *closed* — 401 — so nothing is exposed; it just means public pricing pages can't read it anonymously); an old middleware exemption for `/api/logs/` still exists but no routes sit behind it (404s — worth removing while empty); the Command Center's static app shell

	is publicly fetchable (shell only — all data behind it returns 401); all three main hosts' CSPs allow inline *scripts* (common for tag managers; weakens CSP as an XSS backstop); the marketing site lacks the legacy X-XSS-Protection header the other hosts carry (minor — its CSP and nosniff do the real work); backend API responses carry the full, strong header set.

Security headers — summary

All three main hosts send HTTPS-only HSTS with preload, clickjacking protection, MIME-sniffing protection, referrer and permissions policies, and a real Content-Security-Policy. The admin host's set is the strongest (framing fully denied). The marketing site's CSP keeps its deliberate rule: inline *styles* are not allowed at all; inline scripts are (see the Info note above).

What this workstream could not prove live (details in §8): the agreement gate and scoped-access 403s end-to-end — the only login available was a staff token, and staff are exempt from both controls by design, so a staff 200 proves nothing about client-side enforcement. Testing them needs a real unaccepted client account and a scoped client profile.

3 · OWASP ZAP scans

Tool: OWASP ZAP 2.17.0 — a real ZAP scan (spider + passive scan, the baseline scan's phases), run the night of Nov 20, 2026 against both URLs the owner named. One honest environment note: ZAP was first installed in the audit workspace, but this sandbox's internet egress blocks ZAP's direct connections, so the scans were executed with ZAP 2.17.0 on the owner's own server, reaching the public sites over the internet exactly as an external scan would. The server install was shut down and removed afterwards; the full ZAP reports are saved with this audit's working files. No active attack scan was run, and the scans were unauthenticated — ZAP saw what any visitor sees.

Target	URLs scanned	Alert instances	High / Critical	Medium	Low	Informational
mad-ad-slingers.com	142	559	**0 / 0**	196	72	291
mad-ad-slingers.online	21	61	**0 / 0**	20	6	35

What the Mediums actually are (both sites): the CSP allows inline scripts (78 instances on the marketing site — the same observation as the security workstream's Info note; the *style* side stays strict); third-party scripts load without Subresource Integrity tags (72 + 6); the marketing site's static forms carry no anti-CSRF tokens (46 — the forms post to the backend API, which the security workstream checked separately); and on the app host, one CSP directive is a wildcard (7). The Lows are cross-domain script includes — the site's known third parties (Google Tag Manager, Facebook, Microsoft Clarity). The Informational bulk is cache behaviour and modern-app fingerprints. **Nothing ZAP found is an open door; it is a hardening list.** The app host's low URL count (21) is expected: it is a single-page app, so an unauthenticated crawl sees the login shell and its assets.

4 · Smoke test — do the key flows answer correctly?

#	Flow	Result
1	Backend health (`GET /api/health`)	**PASS — 200, `{"status":"ok","pqc":"ready"}`, public and on the server itself**
2	ChatGPT Ads plans (`GET /api/chatgpt-ads/plans`)	**PASS — exactly 4 plans in order: Test Pilot $499 one-time, Starter Pilot $697/mo, Micro Pilot $997/mo, AI Search Starter $3,000/mo (login required; anonymous gets 401 — see §2 Info)**
3	UGC Ads Generator (`GET /api/client/ugc/config` + route probe)	**PASS — config returns all three lanes with cost tables (Runway recipe, Soul 2 actor lane, talking-head lane); the generate route is registered and validates input (400 "Write at least**

		one hook line") instead of 404. No generation was run and no cost incurred
4	Marketing pages: home, pricing, ad-cost comparison, ChatGPT Ads, free audit, adult hub	**PASS — all 200**
5	Portal and admin apps load	**PASS — both 200 with correct titles**
6	Homepage Book a Call pill + Meta pixel	**PASS — Calendly link present twice; Meta pixel 2058937894756740 present**
7	Slingy public brain route (`POST /brain`, no key)	**FAIL — 404, not 401. The public route is still missing: the site sits behind Vercel, which never passes /brain to the server. This is the known pending fix (a Vercel rewrite plus the server route, awaiting the owner's paste) — not a new discovery.**
8	sitemap.xml + robots.txt	**PASS — both 200; sitemap lists 71 URLs**

Addendum — Creatives Gallery: 8 videos cannot play (owner capture + audit verification)

Observed live by the owner at ~8:51 PM CT on audit night, signed into the portal's Creatives Gallery: the gallery holds 12 creatives (4 images, 8 videos); **all 8 videos fail to play** (buffering spinner, 0:00, "Unable to play media.", controls disabled; all 8 marked BAD), and **gallery images cannot be opened** — clicking does nothing; there is no lightbox or detail view. The 4 images themselves render (2 marked ACTIVE).

The audit then verified the media layer directly (database + file checks): the owner's business has 15 stored generated videos — **6 point at expired CDN links** (WaveSpeed/CloudFront URLs that now return 403), **2 point at files that no longer exist on the server** (404), and 7 are healthy. That accounts for exactly the 8 dead gallery videos: **they are dead at the file level, not just in the player**. A second business's 5 videos all serve correctly. Player behaviour itself could not be re-verified by the audit (no live browser), but the file-level evidence fully explains it. Severity: **Medium (functional)** — client-facing creative work is unplayable, and the expired-CDN pattern will keep eating older generated videos unless originals are kept on the server.

5 · Memory tests — Elephant (long-term) and Goldfish (short-term)

Why these tests exist. "Elephant" checks whether the Brain remembers what it is taught, forever. "Goldfish" checks whether a chat remembers what was said earlier in the same conversation — and whether it wrongly "remembers" it later. Full transcripts are saved with this audit's working files. Both tests were run against the real backend surfaces; what follows is exactly what happened.

Elephant — FAILED (the fact is stored, but nothing can read it back)

The test fact, taught through the established teach path on audit night: *"the Zorblax Index for Mad Ad Slingers is exactly 47.3, and the audit test passphrase word is 'quokka-lantern'"*. The teach **succeeded as a write** — the row sits in the Brain's knowledge store (id 1229320739224158209, verified directly in the database). Then, in fresh sessions:

- **Portal Slingy chat:** generic fallback — "I don't have a number for that one yet." No 47.3, no quokka-lantern.
- **Admin Brain Ask box, in the domain the fact was taught into:** refused — HTTP 400 "unknown domain".
- **Brain Ask in the two allowed domains:** 0 results. The Brain's own retrieval service raises the same "unknown domain" error.

Root cause — a wiring defect, not a storage or search defect. The Brain only recognises two domains ("core" and "chatgpt_ads"), but the teach script writes each platform's teachings under that platform's own

domain name, unchecked. Anything taught that way lands in a domain every reader is forbidden to open — and the Brain's own overview doesn't even count those rows. The chat surfaces users actually talk to never consult the Brain store at all.

This is not just the test fact. The real platform teachings are stuck the same way: the **TikTok teaching (Nov 20)** — refresh-token rotation from Dec 1, 2026, save-the-latest-token by Jan 31, 2027 — and the **Pinterest teaching (Nov 17)** are both write-only in the Brain; asking for them returns unrelated activity stubs or the same domain refusal. A control test proves the chain itself works: the **ChatGPT Ads teaching (Nov 15) recalls perfectly** — because ChatGPT Ads happens to be one of the two allowed domains, and the asker named it. One nuance in the platform bots' favour: the teach script's other half — per-bot lessons — is stored and loaded correctly, so the bots may still apply these updates while they run; what is broken is the Brain half and every chat/Ask recall path for non-ChatGPT platforms. **Plainly: the standing order "teach every platform update to the Brain" currently only produces a retrievable Brain record for ChatGPT Ads.**

Decision for the owner: the nonsense test fact was left in the Brain store, as instructed. It is unreachable by every reader, so it cannot affect any answer. Removing it takes a one-row database delete whenever you say so.

Goldfish — FAILED in-session; nothing leaked

One session on the portal chat surface, 12 messages: a codeword (`silver-badger-42`) and a number (812) planted first — with the full conversation history supplied in every request, giving the surface its best case — then ten ordinary exchanges, then the recall question.

- **Plant reply:** a stray demo answer about cake orders from the built-in knowledge file. The codeword was never acknowledged.
- **Recall reply:** the generic Slingy intro. Neither the codeword nor the number. The endpoint accepts a `history` field and **never reads it** — and the real portal clients don't even send it.
- **Fresh-session check:** asked "What is my test codeword?" — generic fallback. **No leak.** There is nothing to leak: the surface stores nothing per conversation.

Verdict: the backend chat has no working conversation memory at all — it does not hold facts within a session, and (vacuously) does not carry them across sessions either.

One surface could not be driven live — stated plainly

The mobile app's Slingy-brain server could not be tested live: it needs a portal API key, and issued keys are unrecoverable by design (only a hash is stored; the secret is shown once). Minting a new one would have been a write beyond the one authorised teach, so it was not done. Its full source was read instead — it is small and unambiguous: a static system prompt, no database access, no server-side memory; it forwards whatever history the app itself re-sends. Neither Elephant nor server-side Goldfish could pass there even with a key. What a live test would need: the owner revealing a fresh portal key once, plus any caller LLM key (a Groq key already exists in place on the server).

6 · Ratings — 1 to 10 against the top competition

Method. Scored against competitors *as they are today* (their public sites and pricing pages were checked on audit night), not against other startups. 9–10 = proven best-in-class at scale; 5 = credible mid-market parity; 1–2 = absent or unproven. The competitors: **Smartly** (enterprise; ~10–12 live channels incl. TikTok and CTV, 800+ brands, $7B+ spend managed, no public pricing — reported median ≈ $90K/yr), **Madgicx** (Meta-only automation, from $45/mo, ~200K claimed advertisers), **AdEspresso** by Hootsuite (Facebook/Instagram only, from $49/mo, effectively in maintenance — no product updates since early 2024), and **Birch**, formerly Revealbot (Meta/Google/TikTok/Snapchat rules automation, from $49/mo, $2B spend, actions every 15 minutes).

Category	Score	Evidence in one line	The bar
Platform coverage	**6**	9 platforms proven Working beats Birch (4), Madgicx (1) and AdEspresso (1) — but no TikTok yet, and 26 of the 40 catalogued	Smartly (9)

		platforms are not working.	
Pricing & value	**7**	$49 entry matches every SMB rival while covering 9 platforms, and the flat tiers badly undercut Smartly's ~$90K/yr — but value is unvalidated (0 paying customers) and the 5% AI fee stacks on top.	Madgicx (9)
Creative tooling (incl. UGC generator)	**6**	18 image + 42 video models across 3 providers plus a dedicated UGC generator is broader generation access than any rival here — but it launched the night of the audit, with zero usage proof, and model access is not a proven catalog workflow.	Smartly (10)
Client portal experience	**5**	Scoped access, agreement gate, view-only preview and per-client connections are genuinely differentiated — but no paying client has ever used the portal, and there are no reviews.	Smartly (9) / Madgicx (8)
Automation / bot fleet	**5**	A 115-bot fleet with orchestrator, staging/approval pipeline and CAPI senders for 5 platforms is substantial internal automation — not yet a proven client-facing rules engine running on client spend.	Birch (10)
Security posture	**3**	The Nov 12 holes are fixed and email DMARC is at quarantine — but MFA is still off on the owner's superadmin account, the single point of total compromise, with no SOC 2 or published pen-test equivalent.	Smartly (9)
Market traction	**1**	0 paying customers, 0 reviews, solo operator, 18 days since launch — no market traction yet, whatever the build speed.	Smartly (10)

Overall = the plain average: (6 + 7 + 6 + 5 + 5 + 3 + 1) ÷ 7 = 33 ÷ 7 = 4.7 / 10. A remarkably fast, broad build with SMB pricing that already out-covers three of four named competitors on working integrations — held below 5 by what the competitors have and it does not yet: TikTok, proven automation at client scale, an MFA-secured/certified security posture, and any paying traction at all.

Note on the creative counts: the 42-video-model count was the catalog as observed during the audit (evening of Nov 20). During the audit the owner ordered one video model family (MagiHuman — the UGC talking-head lane, observed live in the UGC config that night) removed completely, and that removal plus its replacement was already in flight in parallel. Model counts may therefore shift; the removal is an owner-ordered change in progress, not an audit defect.

7 · All findings by severity

Severity	Findings
Critical	None — in the security test or the ZAP scans.
High	None — in the security test or the ZAP scans.
Medium	**Security/functional:** Stripe checkout-status endpoint 500s for every real session (root cause pinned; payments unaffected). **Functional:** Creatives Gallery — 8 of the owner's 15 stored videos are dead files (6 expired CDN links, 2 missing from the server) and cannot play; gallery images have no open/detail view. **Functional:** Brain teach→recall wiring broken — teachings for every platform except ChatGPT Ads are stored in domains no reader may open; the TikTok (Nov 20) and Pinterest (Nov 17) teachings are affected right now. **Functional:** Portal chat has no conversation memory (history field accepted and ignored); Goldfish test failed in-session. **Hardening (ZAP):** CSP inline-script allowances and a wildcard directive; missing Subresource Integrity tags; anti-CSRF token flags on static forms.
Low	Agreement gate fails open on internal error · vps subdomain's thinner header set · /chatgpt-ads pricing gaps (two of its own four plans unpriced; 7 of the 12 standard prices absent) · stale platform-status copy (Microsoft page worst; Snapchat/Reddit "rolling out"; Madison "seven live"; Slingy's "7 platforms" knowledge) · "Magic Ad Maker" remnants (backend API label; dead portal

	page; admin's older portal copy) · one dead inline style on 3 pages · portal fallback launch list missing Reddit/Vibe/Microsoft · UGC page "Coming soon" presenter placeholder · orphaned/dead page files in portal and admin (cleanup debt).
Info	ChatGPT plans endpoint requires login despite being designed public (fails closed) · empty `/api/logs/` middleware exemption worth removing · Command Center static shell publicly fetchable (no data) · marketing site missing legacy X-XSS-Protection header · no dedicated YouTube marketing page · two pricing surfaces defer to the pricing page (/ad-cost-comparison, portal ChatGPT tab) — owner's call under the strict completeness rule.

Known items confirmed, not new findings: the Slingy `/brain` public route 404 (pending the owner's server paste); MFA off on the owner's superadmin account (already on the owner's list — it is also why Security posture scores 3); EMAIL_OUTREACH_PAUSED still on.

8 · Not tested / not proven

- **Mobile rendering at phone width** — no live browser was available to any workstream; only static checks were done. A rendered 390px pass (homepage, /pricing, /ad-cost-comparison, /chatgpt-ads) is still owed.
- **Authenticated portal/admin use** — pages were inventoried from source and served code, not clicked through as a client. The Creatives Gallery player behaviour rests on the owner's live capture plus the audit's file-level verification.
- **Cross-account report access (IDOR), live** — fixed in code and closed to anonymous callers; a live client-A-vs-client-B read needs a second client's login, which this read-only audit did not have and did not create.
- **Agreement gate and scoped-access 403s, live** — verified in code; not exercised end-to-end, because the only available login (staff) is exempt from both by design.
- **Slingy-brain live chat** — could not be driven (portal keys are unrecoverable by design); assessed from its full source instead.
- **ZAP active (attack) scanning and authenticated ZAP scanning** — out of scope for this pass; the scans were baseline (spider + passive), unauthenticated.
- **Bot application of taught lessons while running** — lesson storage/loading verified in code; not live-proven.
- **UGC generator output** — config and routes verified; no paid proof take was run (the cheapest is ≈ $2.28 on the key owner's credits and still needs the owner's word).
- **Anything involving spend, sends, or submissions** — excluded by the audit's read-only rules.

Full working papers for this audit (per-workstream reports, complete ZAP HTML/JSON reports and alert exports, and the Elephant/Goldfish transcripts) are kept with the audit files: `~/workspace/full-audit-20261120/` (ws1-full-audit.md · ws2-security.md · ws3-zap.md · ws4-smoke.md · ws5-memory.md + ws5-transcripts/ · ws6-ratings.md · zap/). Audit run Friday, November 20, 2026, ~8:45–9:05 PM CT. Read-only throughout; the single authorised write was the Elephant teach described in §5. No secret values appear in this report.