PWOS at a glance

Protocol Wealth Operating System · fact sheet

2026-06-16

Protocol Wealth Operating System (PWOS) is the integrated advisor IDE, client portal, and compliance engine Protocol Wealth built for its own SEC-registered investment advisory practice.

PWOS is architecturally an *integration wrapper* around open-source primitives. The open primitives (`pwos-core` Apache 2.0, `nexus-core` Apache 2.0) are published and inspectable without an NDA. The integration logic on top is what turns those open primitives into a governed system operated inside a live SEC-registered practice.

The six structural layers

Layer	What it does	Basis
Zero Data Retention (ZDR)	Anthropic ZDR posture, enforced contractually at the API workspace level. Inputs and outputs are not retained beyond the duration of a single inference request; content is contractually excluded from training, fine-tuning, and model improvement. Inference runs on US-based infrastructure.	Contracted at the workspace level; effective April 21, 2026
Schema-level PII tagging	Every client-data field is tagged high-, medium-, or low-sensitivity at ingestion. High-sensitivity fields (SSN, full account numbers, wallet seed phrases, biometrics, authentication artifacts) are structurally stripped from every model-bound payload by middleware — not by developer discipline.	Enforced in middleware; drift-checked in CI

Layer	What it does	Basis
Independent PII egress canary	A second-layer guard runs at every model call site. The pattern set is *deliberately* re-implemented independently so the two layers cannot share a single bug. Any residual high-sensitivity data aborts the call and surfaces as a structured, alerted event.	Defense-in-depth; fail-closed posture
WORM audit retention	A retention-locked, write-once object store mirrors every row written to the canonical audit log. Bucket-level retention lock means no PW operator — including a full cloud-organization administrator — can delete or modify an object within the window. Each object carries a content hash and the source revision in metadata.	SEC Rule 17a-4(b)(4) + 17a-4(f)(2)(ii)
Sentinel-row reconciliation	Reconciliation of a failed mirror write cannot `UPDATE` the audit log (the table is immutable at the database-trigger level). Instead, a daily job emits a *new* sentinel row referencing the failed row's ID. The pattern applies to every immutable evidence table.	Append-only; immutable evidence tables
Canonical webhook receiver	Every vendor callback flows through the same stages: verify (HMAC signature), dedup, parse, process, audit, dead-letter. The pattern is shared across vendor integrations; cross-component tracking is mechanical, not per-route judgment.	Uniform across all vendor integrations

What runs today

The onboarding pipeline that takes a prospective client from first contact to a signed advisory relationship runs on PWOS today:

- **Portal authentication** — per-client passkey sign-in.
- **Identity verification + AML** — document and biometric identity checks, with sanctions and risk screening, processed through the canonical webhook receiver.
- **Risk-tolerance assessment** — a validated questionnaire producing a composite score, with an advisor override required before any material change in classification.
- **E-signature onboarding** — the advisory agreement, Form ADV Part 2A, privacy notice, and investment policy statement delivered and executed as one envelope, with annual Form ADV re-delivery.
- **Account-data aggregation** — read-only aggregation of client-held accounts.
- **Onboarding status** — a single view of where each new relationship stands.

The structural-over-disciplinary frame

Where a control can be enforced by code, PWOS enforces it by code. The PII egress guard does not rely on every developer remembering to scrub a payload; middleware enforces it, and CI flags any model call routed outside the middleware. The canonical action catalog, the model alias map, and the PII tag map are each enforced automatically rather than by convention.

The credential-layer thesis — *high-sensitivity client data is never sent to any LLM* — is enforced by code that runs regardless of caller behavior, not by caller behavior itself.

How to verify any of this

Concrete pathways are documented in the public security posture (`/security` § "Verify"):

1. **Confirm ZDR with Anthropic directly** — workspace posture and effective date attestable on request to credentialed reviewers under NDA.
2. **Review the substrate architecture decision records** — covering PII tagging, the WORM audit mirror, the webhook-receiver primitive, and the onboarding state machines. Available to qualified reviewers via the verification pathway on `/security`.
3. **Sample anonymized audit-log and WORM-mirror rows** demonstrating action-verb structure, principal-chain capture, and WORM round-trip.
4. **Vendor due-diligence bundle** — SOC 2 reports, DPAs, and vendor risk assessments under NDA.
5. **Written attestation on CCO letterhead** — formal attestation of current AI posture (subprocessor list, ZDR status, inference region, PII controls in effect at time of request) available on qualified request.

Open source

- `pwos-core` — Apache 2.0-licensed; compliance audit-logging patterns + PII redaction toolkit. Reference code, not deployed.
- `nexus-core` — Apache 2.0 defensive licensing; MCP server foundation; ~243 financial-data tools surfaced; live at nexusmcp.site.

The intent is not marketing surface; it is to demonstrate, in code, the structural-over-disciplinary patterns described above.

Cross-references

- **Public security posture:** `/security`
- **Architecture decision records:** available under NDA via `/security`
- **Form ADV Part 2A:** adviserinfo.sec.gov/firm/brochure/335298
- **Subprocessor list:** protocolwealthllc.com/subprocessors

Protocol Wealth, LLC · SEC-Registered Investment Adviser · CRD #335298 Engineering substrate transparency · aggregate substrate material · not investment advice · not advisory performance.