



VenomX: An AI-Powered Penetration Testing Assistant

Nick Guyette, Jordan Martin, Khalid Mohammed, Coleman Pagac

Drake University



What is Model Abliteration?

Large language models encode behaviors as **directions in the residual stream**: high-dimensional vectors that persist and accumulate across layers. Safety-trained models develop a *refusal direction*: a consistent geometric feature that, when activated by a harmful prompt, steers the model toward declining to respond.

- **Identify** the refusal direction $r = b - g$ by contrasting mean residual vectors of harmful vs. harmless prompts at the layer of peak separation
- **Project out** r from the model's weight matrices $-\Delta W = -\lambda(vv^\top)W$ so the direction can no longer be read from or written to the residual stream
- **Result** is a weight-space edit requiring no retraining; the model retains all other capabilities while losing the geometric "hook" that triggers refusal

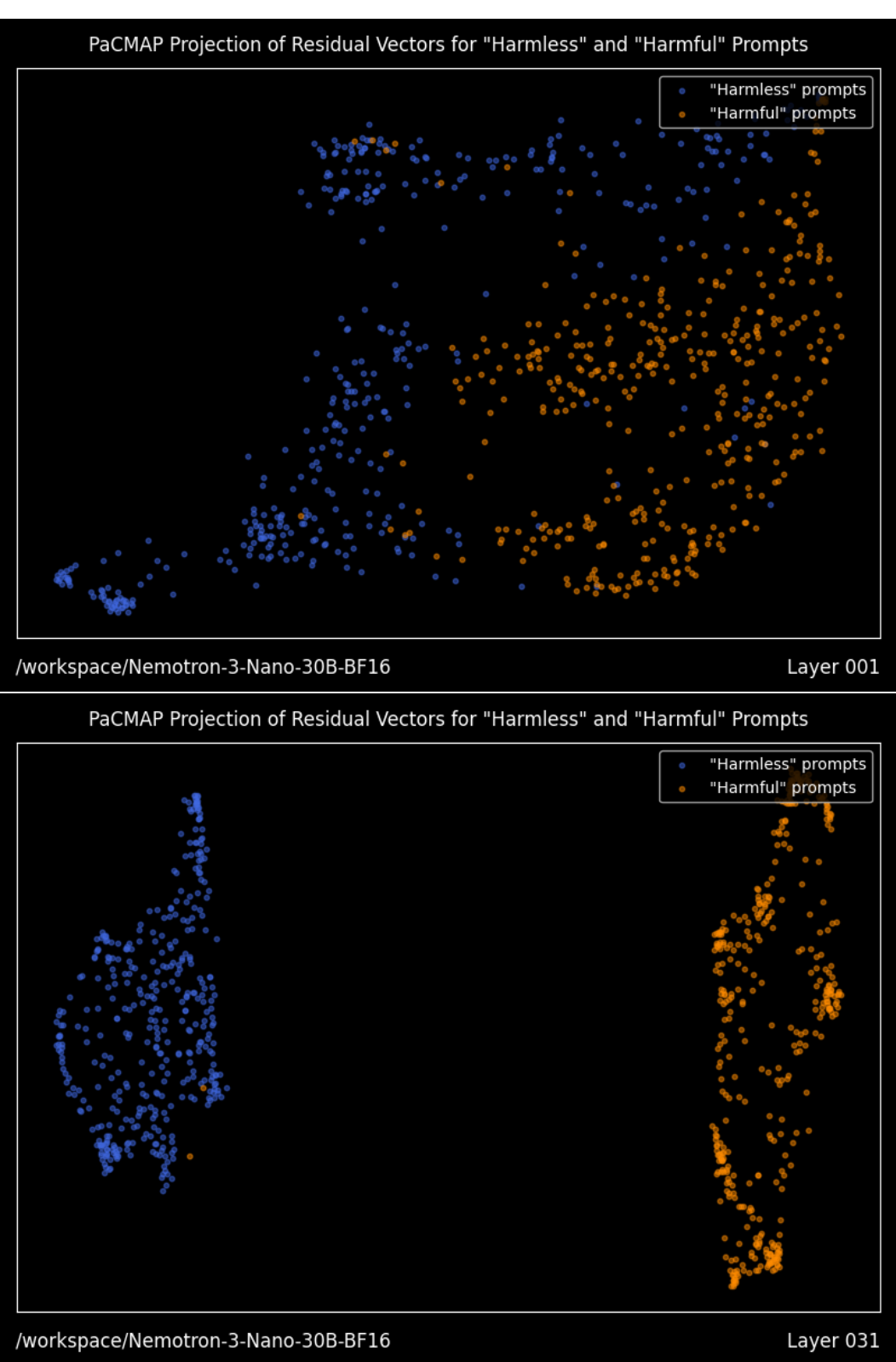
Unlike fine-tuning, ablation is a **targeted surgical edit** where it only modifies the subspace responsible for refusal behavior, leaving the rest of the model's knowledge and reasoning intact.

Residual Geometry

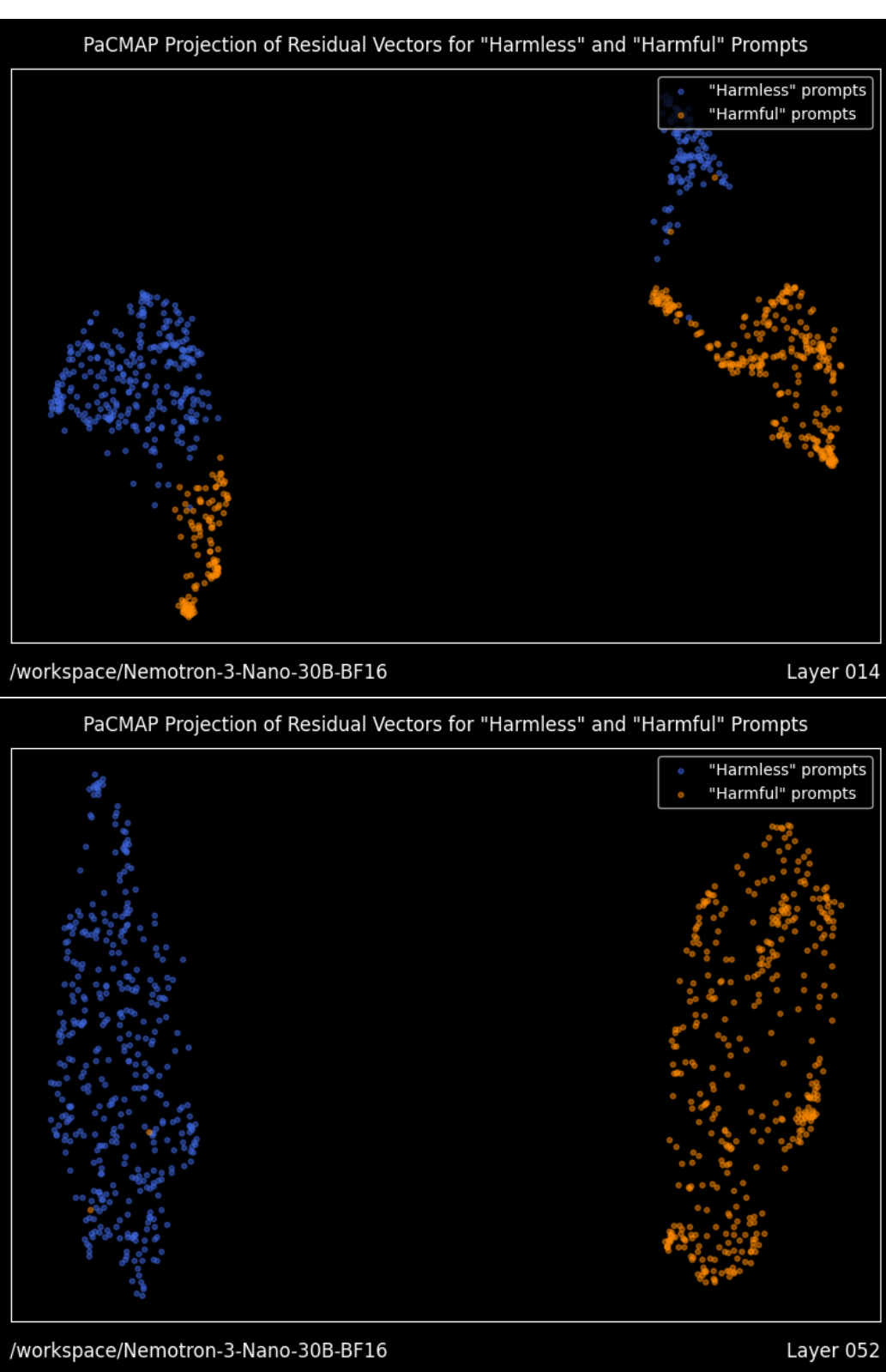
NVIDIA-Nemotron-3-Nano-30B-A3B-Base-BF16

PaCMAP projection of residual stream vectors across depth.

Blue = harmless Orange = harmful



Layer 1



Layer 14

Read Left to Right, Top to Bottom

Layer 31 (peak)

Layer 52

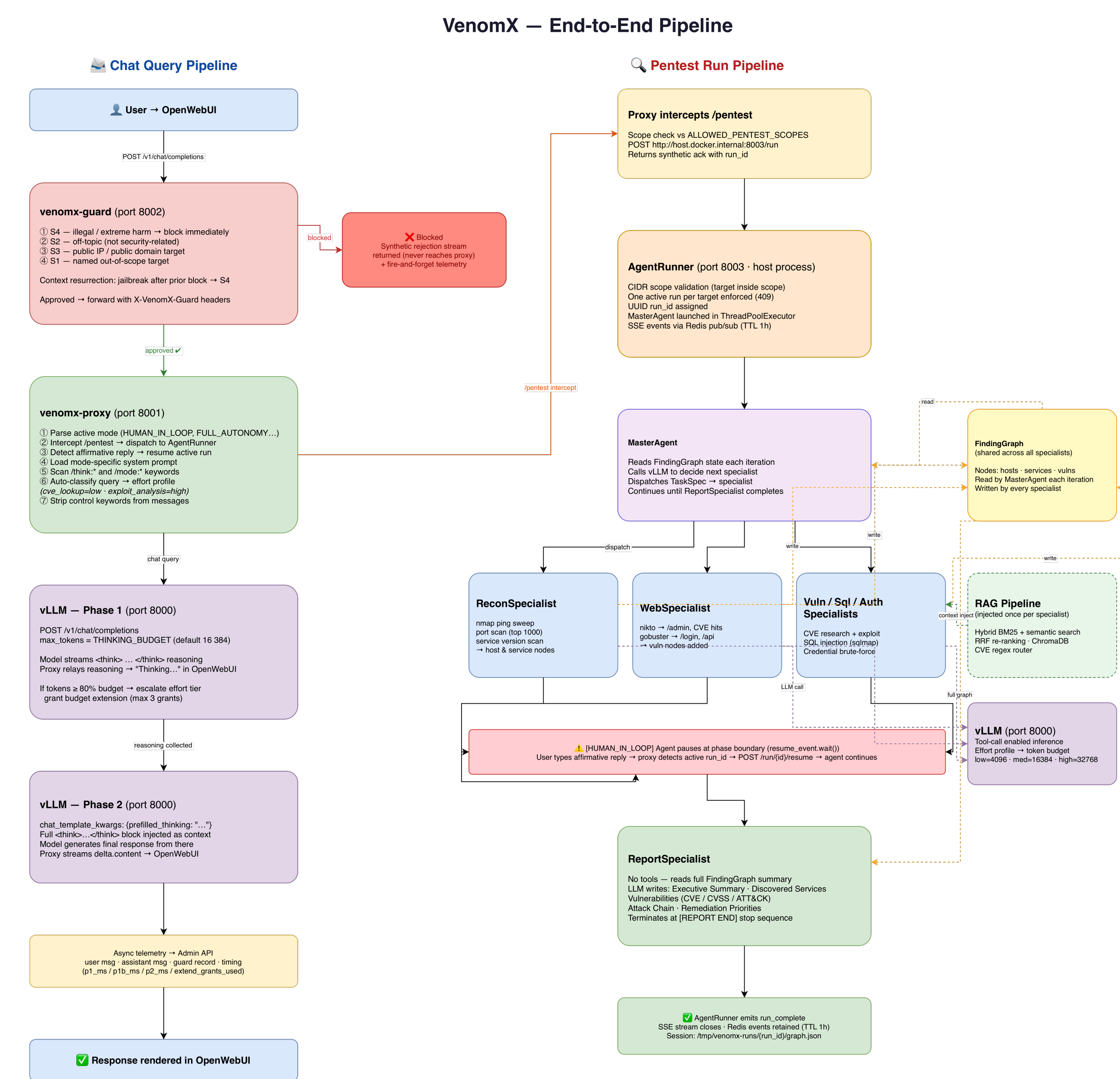
To locate where refusal is geometrically encoded, we scan all 52 layers of the residual stream, computing cluster statistics over a balanced set of harmless and harmful prompts. Each layer yields a mean and geometric median for both classes, a refusal direction $r = b - g$, and a silhouette score measuring how cleanly the clusters separate.

- **Peak silhouette at Layer 31** ($Silh = 0.3688$) — identified as the primary ablation target; refusal direction alignment $S(g, r) = -0.47$ at Layer 29 and remains strongly negative across Layers 29–3.
- **Refusal signal first appears at Layer 6** ($S(g, r) = -0.17$) but does not sustain — consistent negative alignment begins at Layer 14 ($S(g, r) = -0.35$) and persists through Layer 38.
- **63 GB model hosted on RunPod** — $2 \times A40$ (96 GB VRAM combined) required to hold the full model in memory for the complete 52-layer residual stream scan

Despite L_2 norm growth from $|g| = 0.86$ at Layer 1 to 1180 at Layer 52, cosine similarity $S(g, b) > 0.97$ throughout — good and bad residuals scale together, diverging only along the refusal direction.

Website: venomxai.com

End-to-End Pipeline



RAG System

VenomX augments Nemotron 30B with a hybrid RAG pipeline, injecting security knowledge at query time across 487,327 records from 13 sources (NVD CVEs, Exploit-DB, MITRE ATT&CK, HackTricks, GTF0Bins, and others). Records are embedded with BAAI/bge-m3 into dense (1024-dim) and sparse (learned token-weight) vectors stored in PostgreSQL 16 with pgvector. Each query fans out across four search paths, blending cosine and sparse dot-product scores at 70/30 and overretrieving 40 candidates before a CrossEncoder re-ranks to the final top 5. The hybrid blend prevents exact identifiers like **CVE-2021-44228** or **Apache 2.4.49** from being diluted by version-mismatched semantic neighbors. Embedding and re-ranking run on CPU, keeping both GPUs free for vLLM tensor-parallel inference.

VenomX Guard

VenomX Guard is a deterministic safety gate inserted between OpenWebUI and the inference stack, intercepting every chat request before it reaches the model. Approved requests are forwarded to the thinking-proxy (:8001) and onward to vLLM (:8000); blocked requests return a valid OpenAI-shaped refusal rendered as a normal assistant message. A Llama-Guard-3-1B model is loaded for future LLM-based classification but currently bypassed in favor of fast deterministic pattern checks.

Check sequence (first match wins):

- S4** Illegal-harm keywords (e.g. bomb, drugs, suicide)
- S4** Context-resurrection attempt following a prior block
- S2** No cybersecurity-domain signal detected
- S3** Public IP or domain target outside an isolated lab
- S1** Offensive intent paired with a named organizational target

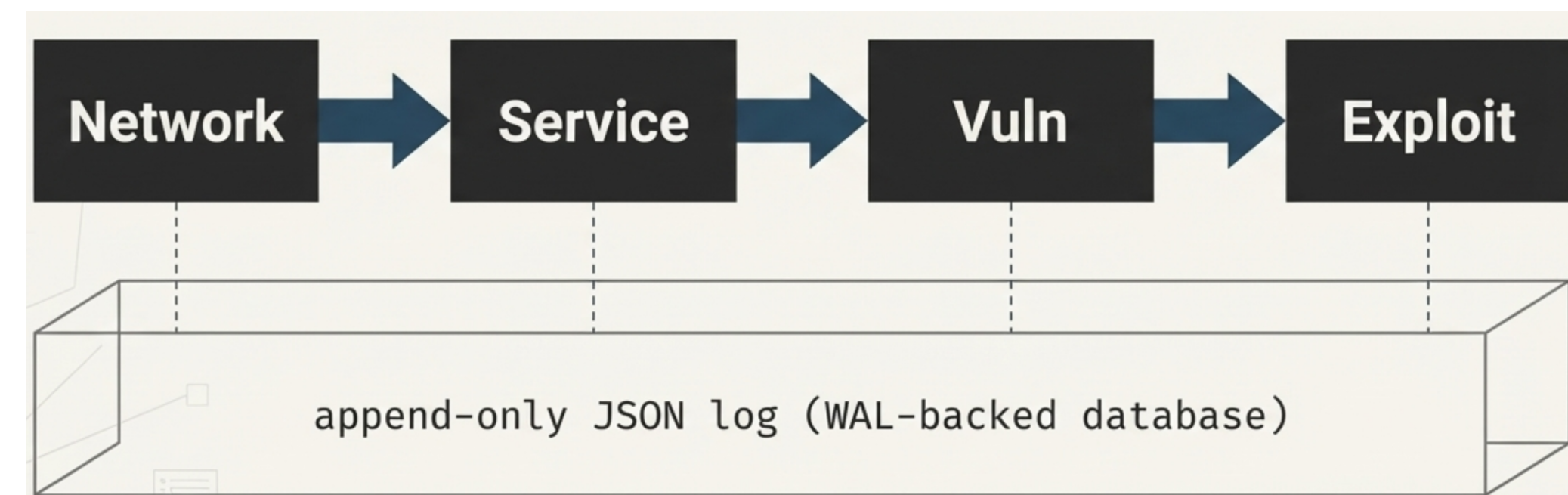
Multi-Agent Architecture

The proxy intercepts `/pentest` commands before any vLLM call. AgentRunner runs on the host machine, granting direct access to installed security tools. Every agent action emits an event to Redis; SSE clients stream live output to OpenWebUI in real time. After Phase 1 (recon), MasterAgent blocks on a **threading.Event** until the user sends an affirmative (*yes / go ahead / proceed*, fuzzy-matched at 0.8 cutoff) via the chat UI.

Specialist Agent	Designated Tools	Dispatch Condition
recon	nmap	always (Phase 1)
web	nikto, gobuster	HTTP/S service found
auth	hydra	login surface found
vuln	searchsploit	service version known
exploit	metasploit	exploit candidate found
sql	sqlmap	injectable endpoint found
smb	enum4linux	port 139/445 in graph
report	(none)	always (Phase 2 close)

Attack Path Traversal & FindingGraph

All specialists share a single **WAL-backed graph** where every `add_node` and `add_edge` call appends one JSON line to `graph.wal` (O(1), append-only), and the materialized `graph.json` snapshot is written only on session close, which is the same write-ahead pattern used by PostgreSQL and SQLite. On crash recovery the WAL is replayed from the last checkpoint with no data loss.



After each specialist run, **AttackPathFinder** performs a DFS over typed edges (**has_service**, **has_vulnerability**, **has_exploit**) and classifies every reachable chain by completeness, sorting all paths by a CVSS-weighted priority score before injecting them into the master's context:

$$P = \begin{cases} 100 + CVSS & \text{complete — full chain, actionable} \\ 50 + CVSS & \text{known gap — CVE confirmed, no exploit yet} \\ 10 + CVSS & \text{unknown gap — service found, no CVE yet} \end{cases}$$

Known gap paths receive a CVSS-stratified recommendation injected directly into LLM context. **Unknown gap** paths auto-generate a targeted RAG query (e.g. “CVE vulnerabilities for ProFTPD 1.3.2c on port 21”) that resolves *before* LLM context is assembled, potentially upgrading the path's own state before the model sees it.

References

- [1] Andy Ardit, Oscar Obeso, Aaquib Syed, David Bau, Arthur Conmy, and Neel Nanda. Refusal in language models is mediated by a single direction. *arXiv preprint arXiv:2406.11717*, 2024.
- [2] Timothy Jaeryang Baek. Open WebUI: User-friendly AI interface. <https://github.com/open-webui/open-webui>, 2023.
- [3] Philipp Emanuel Weidmann. Heretic: Fully automatic censorship removal for language models. <https://github.com/p-e-w/heretic>, 2025.

Consortium for Computing Sciences in Colleges, Central Plains Region 2026

contact@venomxai.com