wallarm

DATA SHEET

PRODUCT DATA SHEET

Wallarm AI Hypervisor

See every AI workload. Attribute every call. Stop the bad ones at the kernel.

SUMMARY

200 external calls in 90 seconds. Do you know what happened?

An unapproved AI agent in your cluster makes 200 external calls in 90 seconds. It sends customer data to a model provider your security team has never reviewed. It operates under a service account with broad permissions. You find out a week later, from a log you weren't watching.

This is not a hypothetical. It is happening in organizations with mature security programs right now, because the tools built to govern API traffic were not built for agentic AI.

AI Hypervisor is the runtime governance layer that closes that gap.

Minutes	0	11+
TO INSTRUMENT AFTER LABELING	APPLICATION CODE CHANGES	MODEL PROVIDERS COVERED

OVERVIEW

Runtime Governance for AI.

Instrument every AI workload without touching application code. Observe every outbound call. Attribute it back to the user who triggered it. Stop it when it crosses a line.

Most organizations deploying AI on Kubernetes can tell you how many clusters they have. They cannot tell you what those clusters are actually doing. Which agents are calling which model providers. Whose requests are triggering which actions. Whether customer data is crossing a boundary it shouldn't. AI Hypervisor answers all of it, without asking your developers to change a line of code.

You get complete visibility into every AI workload the moment it appears, whether it was approved or not. You get attribution from the user request all the way through every service hop and model call. You get real-time detection when sensitive data moves where it shouldn't. And when something goes wrong, you can shut it down in seconds, not hours, without touching the workload.

Deployment takes minutes. Coverage is immediate. The governance evidence your compliance team needs is generated continuously, so when an audit arrives, the record is already there.

THE WALLARM AI CONTROL PLATFORM — TWO PRODUCTS

HYPERVISOR GOVERNS THE AI ITSELF

PRODUCT 01
Infrastructure Discovery
MAPS THE AWS ESTATE AI RUNS ON

DISCOVERS
What exists in AWS. Where findings go.

PRODUCT 02
AI Hypervisor

OBSERVES & ENFORCES
What AI is doing. On whose behalf. Stop it if it misbehaves.

WHO IT'S FOR

AI Hypervisor is built for organizations running AI workloads on Kubernetes that process PII through AI models, face EU AI Act compliance, or operate in regulated industries — fintech, healthcare, enterprise SaaS.

CTO / PLATFORM
No bespoke monitoring
- Know what AI is doing without building it yourself.

CISO
Explain the alert
- Enforce AI policy in real time, without code changes in your applications.

SECURITY ENG
Attribution, not just alerts
- Every AI action traced to the user or session that triggered it.

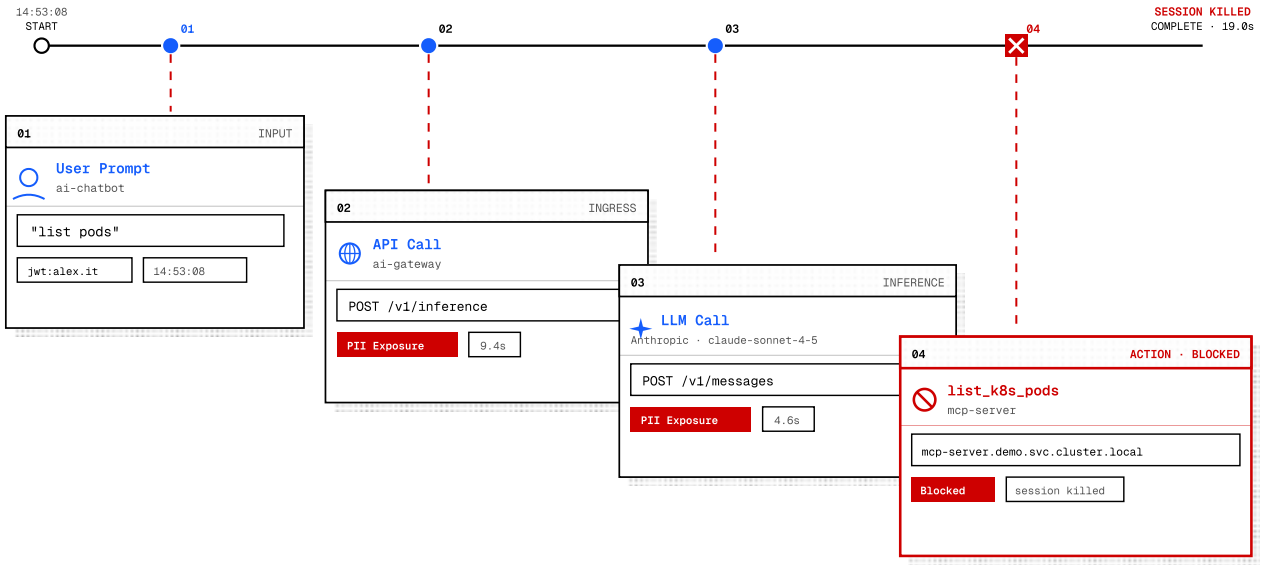
COMPLIANCE
Continuous evidence
- Audit-ready governance, not a spreadsheet assembled the week before.

CHALLENGES & SOLUTIONS

Real problems we hear from real customers. Real solutions.

	CHALLENGE	AI HYPERVISOR
01	"We don't know what AI is running in our cluster, including the workloads that bypassed IT approval."	Auto-discovers every AI workload from runtime traffic and supply-chain signals — model providers, MCP servers, and agent frameworks including LangChain, LangGraph, AutoGen, and CrewAI — including shadow AI never declared in a manifest.
02	"We can see an AI agent did something, but we can't tell whose request triggered it."	Every outbound connection is fully traced and attributed back to the user or session that triggered it, across service hops, via kernel-level thread ID correlation. No trace headers required.
03	"We don't know if PII or sensitive data is leaving our AI boundary."	Detects credit cards, SSNs, passport numbers, DOBs, emails, API keys, JWTs, and passwords inline in AI traffic in real time, before they reach external destinations.
04	"By the time we detect a bad AI action, the damage is done."	Blocks outbound LLM calls on pattern-match enforcement rules at the egress boundary. Revokes compromised sessions by JWT subject, auth-token hash, session cookie, source IP, or destination endpoint with kernel-level connection termination. No pod restart, no deploy cycle, no downtime.
05	"We can't prove to auditors that our AI stack is governed."	Continuous compliance evidence — coverage heatmap, AI inventory with components + CVEs, session audit logs, sensitive data flow records. Audit-ready at any time.
06	"We have a GuardDuty finding on an AI workload and no idea what the agent was actually doing."	Captures full behavioral context per agent: every outbound connection, every model call, every API touched, attributed to the triggering user. GuardDuty findings land on the same graph as runtime context.

PII EXPOSURE, BLOCKED AT THE KERNEL — 19.0S · SESSION KILLED



BENEFITS

For the four roles every AI conversation touches.

FOR CIOS / CTOS
Helm. Label. Done.

- Zero application code changes.
- Language and application agnostic
- eBPF at the kernel — no sidecar.

FOR CISOS
Stop it in real time.

- Know what AI runs, what it does, who triggered it.
- Real-time enforcement
- EU AI Act evidence, continuously generated.

FOR SECURITY ENG
Attribution, end to end.

- Per-call user attribution across service hops.
- Inline sensitive data detection on live traffic.
- Session audit logs with payload traces.

FOR COMPLIANCE
Audit-ready, always.

- Coverage heatmap, AI-SBOM, audit logs, data flows.
- EU AI Act enforcement: Aug 2026.
- Full CVE inventory across the AI stack.

How does it work.

DEPLOY · INSTRUMENT · GOVERN

DEPLOYMENT POSTURE

01 Kubernetes DaemonSet on Amazon EKS, deployed via Helm.

02 Patented non-invasive analysis (US Patent 12,505,228) + eBPF at the kernel.

03 Languages: Python, Go, Node, Java, Ruby, Rust, generic containers.

04 11+ providers including Anthropic, OpenAI, Bedrock, Azure OpenAI, Hugging Face, Together, and others.

05 BPF-optional enforcement: eBPF tc by default, userspace proxy fallback on Fargate, GKE Autopilot, hardened clusters.

06 Enforcement stays inside your EKS. No traffic leaves the boundary for decisions.

INSTRUMENTATION LIFECYCLE

01 Label a deployment — instrumented within minutes. No restart.

02 Outbound calls captured at the kernel: LLM, S3, APIs, DBs, anything reached.

03 Runtime task-ID correlation stitches each call to the triggering user across hops.

04 Inline sensitive data detection at the boundary; detections logged with payload traces.

05 Enforcement blocks at egress; revocation via kernel RST injection + conntrack cache flush.

06 Compliance evidence generated continuously: heatmap, AI inventory, audit logs, data-flow records.

NEXT STEP

See what your AI is actually doing.

A 30-minute walkthrough on a sample workload — instrumentation, attribution, enforcement, evidence. No code changes required to follow along.

WALLARM.COM / REQUEST-DEMO —— 30 MIN · LIVE

Schedule a Demo →