

How Manifold Security Built Production-Ready Observability for AI Agents with Dash0



⚠️ The Problem:

Manifold Security needed production-ready observability early, with visibility across single-tenant customer environments and the ability to investigate issues at scale.

💡 The Solution:

Dash0 provided an OpenTelemetry-native observability platform with automated Kubernetes instrumentation, MCP integration, AI-assisted investigation, and built-in cost controls.

✅ The Result:

Manifold Security moved from limited observability to production-ready visibility in about a week, uses Dash0's MCP server every day to investigate alerts, and reduced telemetry cost by 2–3x with low-effort filtering.

Securing AI Agents Starts with Runtime Visibility

Manifold Security is building a platform to secure AI agents.

As AI agents spread across organizations, the security challenge is no longer just understanding what exists in code. It is understanding what agents actually do at runtime: which tools they call, what systems they access, what data they touch, and how often.

"You cannot really infer what an agent is doing from just reading the code," says Oleksandr Yaremchuk, CTO and co-founder at Manifold Security. "You need to understand what is actually being called, how often, and who is calling it."

That same principle applied to Manifold's own infrastructure. As the company moved toward enterprise customers, it needed to build operational maturity early. The team runs single-tenant deployments, which keeps each customer environment isolated, but also makes observability more complex.

"We have deployments, we have something to show customers, but how do we ensure that all these instances we are running for every customer are working at scale?" Oleksandr says. "That was our motivation to bring in a platform like Dash0."

The Challenge: Enterprise-Grade Visibility Without Enterprise-Size Overhead

Manifold was moving quickly toward production with a small team. They needed visibility across logs, metrics, traces, web events, and synthetic checks, without the operational overhead a traditional platform would add.



» When you're building a company that wants to sell to enterprises, you need to build maturity early. You need to be reliable, you need to monitor all that stuff. «

says Oleksandr Yaremchuk.

Manifold needed an observability platform that matched how modern AI-native teams build: OpenTelemetry, Kubernetes, Terraform, automation, and AI-assisted workflows.

Why Manifold Security Chose Dash0

Dash0's Kubernetes operator was one of the first things that caught the team's attention.

"We looked at the Kubernetes operator. It was really satisfying the way it automatically instruments certain workloads. And we really like the documentation," Oleksandr says.

Implementation was fast. After the first technical call, Manifold's DevOps engineer installed Dash0 and started sending telemetry. The team added tracing, frontend web events, and synthetic checks without heavy manual setup.

"It was very straightforward. He immediately installed everything, started populating data. We just managed to do it all within like one week," Oleksandr says. "Everything was completely automated. Helm charts, Terraform, and almost no click ops at all."

But the biggest differentiator was Dash0's MCP server.

"We use it every day, many times a day," Oleksandr says. "During the POC, we found it so useful that we were like, okay, we need to actually choose it."

For Manifold, this was not AI observability as a marketing claim. It worked in real engineering workflows.

» A lot of companies offer AI features and they market a lot of AI features, but usually when you try them they don't work well in production. For us, the agentic functionality of Dash0 was really the best. «

says Oleksandr Yaremchuk.

The Impact: Investigating Alerts in Minutes, Not Hours



One of the team's most important workflows starts in Slack. When an alert comes in, Oleksandr copies it and uses Dash0's MCP server to investigate.

"I just copy the alert itself and use the Dash0 MCP server and just say, hey, I have this alert, can you figure this out?" he says.

Because Dash0 brings logs, metrics, and traces together in one place, the investigation starts with full context, not a blank prompt.

» It will understand what's happening, look at spans, look at logs, metrics, look at everything that is there — after five minutes tell me what exactly is happening and propose me the potential fix. «

says Oleksandr Yaremchuk.

A Real Issue Resolved in Half an Hour

The value became clear during a recent issue involving synchronization between Manifold's graph database and Postgres.

An alert indicated the two databases were out of sync. Instead of opening a war room, Oleksandr passed the alert into Dash0 through MCP. Dash0 identified that the alert itself was too noisy, caused by a Kubernetes reconciliation behavior that was already recoverable, and surfaced both a short-term fix for the noise and a longer-term improvement around connection pooling.

» It figured it out just because of logs combined with tracing, combined with metrics. It really helped to resolve this issue basically in half an hour, while I was on another meeting, not even looking in an incident mode in War Room. «

says Oleksandr Yaremchuk.

Controlling Cost Without Cutting Visibility

As Manifold expanded its telemetry coverage, cost control mattered just as much as visibility. With legacy vendors, reducing noise meant going back into agent configuration or application code.

“With legacy vendors, we would need to go to agent configuration or actually even code, which is the worst,” Oleksandr says. “Now we can just go to the UI and quickly check some filter, decide okay, for now it’s spam and we don’t want to pay for this.”

That workflow had a direct impact on the bill.

“It reduced the cost by 3x or 2x with really low effort,” he says.

Built for OpenTelemetry-Native and AI-Native Teams

For Manifold, OpenTelemetry was not a checkbox. It was the architecture the team wanted to build on, which made Dash0 a natural fit.

“If you have OpenTelemetry already, it’s very straightforward,” Oleksandr says. “You just change the endpoints you push data to, and that’s it.”

For teams building AI agents, runtime visibility is becoming critical. Agents behave differently in production than they appear to in code, and a small team cannot manually watch all of it.

» The difference it made is getting the critical visibility into our customers’ environments at scale. If you have a small team, you cannot just monitor logs all the time. You need automation, you need visibility. «

says Oleksandr Yaremchuk.

Ready to build observability for AI-native systems?

See how Dash0 helps OpenTelemetry-native teams gain visibility, control cost, and investigate issues faster with MCP and AI-assisted workflows.

[Book a demo](#)

