Consultiply

• AI

MIP RESOURCE CENTER · MSP GUIDE · OCT 14, 2026

Let the AI technician suggest the fix. Keep the approval with your team.

NinjaOne launched Katana AI on Oct 13, 2026, an agentic technician that summarizes, recommends and runs nothing without a technician's approval. What it changes on your service desk and how to pilot it.

ninjaOne®

INDEPENDENT ANALYSIS OF NINJAONE'S ANNOUNCEMENT
NOT AFFILIATED WITH OR ENDORSED BY NINJAONE

What NinjaOne announced

On October 13, 2026, NinjaOne announced Katana AI, which it calls "an agentic AI technician," built directly into the NinjaOne platform. It shipped alongside a NinjaOne MCP server that lets outside AI agents read NinjaOne data and take actions in the platform. In NinjaOne's launch post, the company describes the problem it is aiming at: the time a technician spends gathering context before fixing anything.

Katana AI is in Early Access. NinjaOne has not published pricing, a general availability date, or any speed or accuracy figures for it.



Image: NinjaOne, Katana AI and MCP press release (Oct 13, 2026).

What Katana AI does

- **It summarizes the device.** Device health, recent activity and correlated alerts arrive in one summary, so a technician sees what is happening before digging through logs.
- **It recommends the next step.** Relevant scripts, proven fixes and related incidents appear inline, where technicians already work.
- **It searches in plain language.** Technicians can ask about their environment instead of building filters.
- **It reasons over your own environment.** NinjaOne says Katana AI uses the customer's own NinjaOne data, including device history, alerts and activity. Its FAQ adds: "It does not pull in outside data, and customer data is not used to train the underlying model."

The controls NinjaOne built in

NinjaOne puts control at the center of the launch. Its post says: "Katana AI recommends; it doesn't act on its own; every suggested fix or script requires a technician's explicit approval before anything runs."

Katana AI inherits the permissions of the technician or service account using it. Per-AI controls let IT leaders decide where it is active, down to the individual technician, and a global kill switch can shut off access instantly across the organization. The MCP server follows the same model: NinjaOne says external agents "work through NinjaOne rather than around it, governed by the same role-based permissions IT teams already manage."



Illustration: NinjaOne, 'Katana AI: How NinjaOne is Helping Technicians Resolve Issues Faster' (Oct 13, 2026). Marketing art, not a product screenshot.

NinjaOne's numbers, and what they are

The launch posts contain no measured results for Katana AI. Every figure below is NinjaOne's own statement, and some describe the company rather than the product.

Claim	Measured	Status
Katana AI helps technicians "resolve issues faster"	No figure published	vendor
Architecture "designed for token efficiency" to help customers manage and predict costs	No cost figure or pricing published	vendor
43 minor releases and six major releases in the last year	NinjaOne's own company-wide count	vendor
Patch and remediate in minutes "instead of the industry average of 43 days"	Refers to NinjaOne's vulnerability management, citing Verizon's 2026 Data Breach Investigations Report; not a Katana AI figure	vendor
Nearly 40,000 customers in 140+ countries	NinjaOne's company description	vendor

What it means for your service desk

- **Context is the bottleneck, and NinjaOne is naming it.** NinjaOne argues that the gap between noticing a problem and knowing what to do about it is where most of a technician's day disappears. If that matches your desk, test the summary first, before the fix suggestions.
- **Junior technicians get a senior starting point, if it works.** NinjaOne says a technician new to an environment "gets the same starting point as someone who's been troubleshooting it for years." Treat that as a claim to test on your own tickets.
- **NinjaOne frames it as margin.** "For MSPs specifically, that's a margin story too: the faster a whole team can resolve issues consistently, the more capacity there is to take on new clients without things slipping." Capacity becomes margin only if you measure it.
- **Early users talk about control, not speed.** The one customer quoted, Randy Hyder, MSP Manager at Range MSP, said: "Being able to set what the Katana agent can access and having the ability to approve its actions gives us confidence. It is reassuring to know that adopting Katana AI does not mean giving up control of our environment."
- **Approval is a responsibility, not a formality.** When a technician approves a recommended script, your MSP owns the outcome. An approval step moves the decision to a person; it does not move accountability to the vendor.

Who approves what: delivery responsibility

- **Write approval tiers before the pilot.** Decide which recommendations a level 1 technician may approve, which need a senior, and which are never run from a suggestion alone, such as scripts that change security settings, backups or many devices at once. These tiers are our example, not a NinjaOne feature.
- **Scope by technician.** NinjaOne says per-AI controls reach "down to the individual technician." Start with a small, named group.

- **Record the approval with the ticket.** Keep the recommendation, the approver and the result in the PSA ticket so you can show a client who decided what. The launch posts do not say whether NinjaOne records approvals automatically; confirm it during Early Access.
- **Set a policy for MCP connections.** The MCP server lets external agents take actions in NinjaOne. Treat each connected agent like a new technician account: a named owner, a scoped role and a regular review.
- **Rehearse the kill switch.** Know who is allowed to use it, and try it once on a test tenant.

> **Governance note.** Approval before action, inherited permissions and an off switch are the three controls to ask every RMM, PSA and security vendor for. NinjaOne has published all three for Katana AI. Whether they behave as described is something to verify in your own environment during Early Access.

What to do this quarter

- **Pilot internally first.** If you run NinjaOne, ask about Early Access and start on your own endpoints before any client environment.
- **Measure before and after.** Pull time to first action, reopen rate and escalations per technician from your PSA for a month before the pilot. NinjaOne published no baseline, so yours is the only number that counts.
- **Share your approval tiers.** Clients are asking how AI touches their environment. A one-page approval policy answers that.
- **Review agent connections monthly.** List every agent connected through MCP, its owner and its role.
- **Ask your other vendors the same three questions.** Approval before action, inherited permissions, an off switch.

Read with care

- **Early Access only.** No pricing, general availability date or regional availability was published.
- **No measured results.** Faster and more consistent resolution are NinjaOne's claims. The customer quote describes confidence, not numbers.
- **Token efficiency is not a price.** NinjaOne says the design helps customers predict costs but published no cost.
- **Inherited permissions cut both ways.** If a technician's role is too broad, Katana AI's reach is too.

> This guide is Consultiply's independent analysis of NinjaOne's own announcement, translated for MSP operators. It is not affiliated with or endorsed by NinjaOne, and it is not a product recommendation.

Read the full announcement: NinjaOne press release.

Sources

NinjaOne Brings Governed Agentic AI to Unified IT Operations with Katana AI and MCP (NinjaOne press release, Oct 13, 2026)

Katana AI: How NinjaOne is Helping Technicians Resolve Issues Faster (NinjaOne, Nick DeStefano, Oct 13, 2026)

Read the full guide: consultiply.ai/mip-resource-center/ninjaone-katana-ai-approval-first-msp