PORT1

TRACEFORCE FOR MSPs

What to look for in an AI security tool for your clients

A checklist for MSPs choosing an AI security tool for clients: what it should see, what it should control, how it handles client data and how it fits an MSP.

November 2026 | Jason Makevich, CISSP

An AI security tool for MSP clients should see AI wherever it runs on a computer, control it on the client's terms, handle client data carefully and fit the way an MSP works across many clients. The questions below work with any vendor, and each group explains why it's worth asking.

Does it see all the AI on a computer?

AI runs in the browser, in desktop apps and in command line tools, so a tool that sees only 1 of those leaves gaps. Local connectors (MCP servers) are easy to miss too: under the Model Context Protocol's stdio transport, the AI app launches a local MCP server as a subprocess on the same computer.

- Does it see AI in the browser, in desktop apps, and in command line and coding tools?
- Does it cover coding agents working inside dev containers?
- Can it tell personal accounts from company accounts, and free plans from business and enterprise ones?
- Does it find the connectors (MCP servers) people add to AI tools, and rate each one's risk?
- Does it show which AI models people are calling?
- Does it spot sensitive data in prompts, tool calls and attachments, including encoded text, compressed files and PDFs?
- Does it rate risk against a recognized framework, and keep a registry of known AI agents and connectors?
- Does it keep up as new AI tools, versions and connectors arrive?

Our guide to finding shadow AI on client computers covers what each of these turns up.

Can it control AI as well as report on it?

A good tool does both: it reports on AI use, then backs the client's rules with controls. OWASP ranks prompt injection first among risks for LLM applications, as LLM01 in 2025 and again in its 2026 Top 10.

- Can it keep AI tools on company accounts and block personal sign-ins?
- Can it warn or block when sensitive data goes into AI tools, with custom rules for data specific to a client?
- Can it stop risky AI agent actions, like deleting data or force-pushing over shared history, before they run, with the choice to block, require approval or report?
- Does it detect prompt injection?
- Can it allow or deny connectors, AI models and AI apps?
- Is it clear, for each AI tool, which controls are enforced and which only monitor and alert?
- Can new policies start inactive and go to a pilot group first?

For how agent controls work in detail, see our guide to securing AI agents and MCP servers.

How does it handle client data?

A tool that watches AI use will see sensitive information, so its own data handling deserves the same questions as the AI tools it watches.

- Does it store prompts by default? If it does, where, for how long, and who can read them?
- Does sensitive data detection run on the computer?
- Can the client keep detailed records in its own cloud storage, with its own encryption and retention settings?
- Is data encrypted in transit and at rest, and is each client's data kept separate?
- Does the vendor hold ISO/IEC 27001 certification or a SOC 2 Type II report, and can you see the report?
- What does it upload from the computer, and what does it leave behind, like credentials it finds?
- Does it inspect only AI traffic, and if inspection fails, do people's AI tools keep working?
- Is it visible on the computer, and can users switch it off?

Does it fit the way an MSP works?

A tool built for 1 company at a time adds overhead when you run it for dozens of clients.

- Is it multi-tenant, with each client in its own organization and a view of every client at once?
- Does it deploy through your RMM or MDM, with nothing for the client's people to install?
- When a client moves from an assessment to ongoing service, does anything need redeploying?
- Does it support the Windows and macOS versions your clients run?
- Can its data feed your SIEM, ticketing or reporting tools through an API?
- Does each computer show its coverage, so you can confirm every part of the tool is installed?
- Does it produce client-ready reports in your branding?
- Who supports your team when something goes wrong, and how does an issue get escalated?

For turning a tool's findings into rules a client can follow, see our guide to AI governance for MSP clients.

How do you use this checklist?

The checklist works best sent to each vendor in writing, so the answers can be compared side by side, followed by a pilot at a real client to see whether they hold up. A feature that isn't available yet counts as a no for now. It also helps to know which controls are enforced and which only report, for the AI tools your clients actually use.

How do you test a tool before you commit?

You test it on a pilot group, with made-up data and success criteria agreed at the start.

- Agree success criteria, sign-off and a decision date before the pilot begins.
- Give each control 2 tests on a pilot computer, with made-up values in a real format and never real data: one that should be stopped and one that should go through.
- Check that every computer in scope is reporting, including the browser side of the tool.
- Watch for disruption beyond expected warnings, and for complaints about slower computers.

PORT1's free AI Risk Assessment is one way to see the visibility questions answered at a real client. PORT1 brings enterprise-grade security and AI solutions to MSPs and MSSPs and sells only through them, and our partner program page covers how working with PORT1 works. See how PORT1 handles AI security for MSP clients.

Sign up with PORT1 and run a free AI Risk Assessment at your first client.

Sources

- Model Context Protocol specification, stdio transport, version 2026-07-28
- LLM01:2025 Prompt Injection, OWASP GenAI Security Project
- OWASP GenAI LLM Top 10 2026, OWASP GenAI Security Project, 2026

Start Now