The CISO's Guide to Securing MCP Servers.

Board-level risk framing for the unmonitored layer of the agentic stack.

91%
of enterprises deploy or plan to deploy AI agents

0
native security controls shipped with MCP protocol

$4.45M
average cost of a data breach — IBM, 2024

> *"MCP servers are the new shadow IT. They give AI agents real-world capabilities — and most enterprises have no idea what those servers can access, who authorized them, or what they log."*

EXECUTIVE SUMMARY

MCP is the unmonitored layer of the agentic stack.

AI agents are no longer experimental. Financial services firms use them to process claims. Healthcare systems use them to triage patient data. Retailers use them to manage supply chains. In nearly every case, those agents take action through an emerging protocol called MCP — the Model Context Protocol.

MCP servers are the connective tissue between AI reasoning and enterprise systems. They give agents the tools to read files, query databases, call APIs, send messages, and execute workflows — autonomously, at machine speed.

They are also almost entirely ungoverned.

Unlike APIs, which organizations have spent years learning to secure, MCP servers sit outside existing security frameworks. They are not tracked in your CMDB. They are not scanned by your vulnerability management tools. They are not logged by your SIEM. They were stood up by a developer, a vendor, or an AI platform — and they are running right now, with no visibility and no controls.

The board-level risk When an AI agent takes an unauthorized action through an MCP server — exfiltrating customer data, executing a fraudulent transaction, or escalating its own privileges — the board will ask three questions: What did we know? When did we know it? What controls did we have in place? Today, most organizations cannot answer any of them.

29M+	2.8B	0
AI agents deployed globally in 2025	Projected AI agents by 2030	Native security controls shipped with MCP protocol

THE PROBLEM

Why MCP creates board-level risk.

Five structural gaps your security team cannot close without purpose-built tooling.

01

No native authentication or authorization

The MCP specification does not mandate authentication. Most MCP servers ship with no built-in access controls. Any agent — or attacker — that can reach an MCP server can invoke its tools. This is not a configuration gap. It is a protocol gap.

02

No audit trail by default

MCP servers do not log by default. When an agent takes an action — reading a file, querying a database, calling an API — there is no native record. Security teams cannot reconstruct what happened, when, or by whom. Forensics after an incident become nearly impossible.

03

Shadow MCP infrastructure

Developers spin up MCP servers without involving security teams. Vendors ship AI integrations that bundle MCP servers as a dependency. Organizations routinely have dozens of MCP servers running in production that no one in the security function knows exists. This is the same problem as shadow IT, running at agent speed.

04

Prompt injection and tool poisoning

Attackers can craft inputs that cause an AI agent to invoke MCP tools in unintended ways. A malicious document can trick an agent into exfiltrating data through an MCP server it already has access to. Unlike traditional injection, there is no human review step before the action executes.

05

Excessive permissions and privilege escalation

MCP servers are frequently configured with broader tool access than any agent requires. An agent that needs to read calendar data may have access to an MCP server that can also send email, query HR systems, and modify access controls. Least-privilege is the right policy. Almost no organization can enforce it across MCP today.

GOVERNANCE FRAMEWORK

What CISOs need to govern the MCP layer.

A practical framework for agentic security built for the boardroom.

Governing MCP servers requires the same disciplines that mature organizations apply to APIs and cloud infrastructure — discovery, posture management, and runtime protection — adapted for the unique characteristics of agentic systems. The framework has four pillars.

Discover everything

You cannot govern what you cannot see. The first requirement is a complete, continuously updated inventory of every MCP server in your environment — who deployed it, what tools it exposes, what APIs it connects to, and which agents have access. Discovery must extend to shadow MCP infrastructure, including servers deployed by vendors and embedded in third-party AI products.

Assess posture continuously

Every MCP server represents a posture risk. Is it running without authentication? Does it expose sensitive tools to agents that should not have access? Is it connected to a critical API with no rate limiting or abuse detection? Posture assessment must run continuously — because MCP infrastructure changes faster than any periodic scan cycle can track.

Enforce least-privilege at the tool layer

Agents should have access only to the specific MCP tools they require for their defined function. Policy enforcement must operate at the tool level, not just the server level. An agent that reads contracts should not have access to an MCP tool that can also post to Slack or execute SQL. Enforcing this at scale requires automated policy, not manual configuration review.

Monitor and respond to runtime behavior

Even well-configured MCP servers can be abused. Runtime monitoring must detect behavioral anomalies — an agent invoking tools outside its normal pattern, a spike in data access, a sequence of tool calls that resembles exfiltration or privilege escalation. Response must be fast enough to interrupt actions before they complete.

The CISO mandate MCP governance is not optional. As AI agents take on consequential business functions, regulators, insurers, and boards will demand evidence that agentic systems are governed with the same rigor as other critical infrastructure. CISOs who build that evidence now will be ahead of the mandate. Those who wait will be explaining a breach.

BOARD-LEVEL RISK NARRATIVE

The language your board already understands.

Translating MCP risk into fiduciary and regulatory terms.

The technical details of MCP security do not belong in a board presentation. What does belong is a clear articulation of the business risk, the current control gap, and the investment required to close it.

Operational risk

AI agents are executing business processes — approving transactions, generating documents, communicating with customers, modifying system configurations. An uncontrolled agent action carries the same operational risk as any other automated system failure, with the added complication that it may have been deliberately induced by an external actor.

Regulatory exposure

Data protection regulations require organizations to demonstrate control over how personal data is accessed and processed. When an AI agent accesses data through an MCP server, that access must be attributable, auditable, and constrained to what was authorized. Organizations that cannot produce this evidence face regulatory risk that is quantifiable and growing.

Third-party liability

Many MCP servers are deployed as part of vendor AI products. Your organization may have no visibility into what those servers can access, how they are configured, or whether they meet your security standards. Third-party MCP servers represent supply chain risk at the AI layer — a category most vendor risk management programs do not yet evaluate.

Cyber insurance implications

Insurers are beginning to ask about AI agent governance in policy renewals. Organizations that cannot demonstrate MCP server inventory and controls may face coverage limitations for AI-related incidents. The actuarial models are still forming, but the direction is clear: governance evidence will determine coverage terms.

THE SALT SOLUTION

Agentic security built for MCP.

The only platform purpose-built to discover, govern, and protect the complete agentic stack.

Salt Security's Agentic Security Platform extends Salt's proven API security foundation to cover the full agentic stack — AI agents, MCP servers, the tools they expose, and the APIs those tools invoke. It is the only platform that operates natively across all three layers of the agentic architecture.

01

Agentic Discovery

Automatically discovers every AI agent, MCP server, and connected API across your environment — including shadow infrastructure deployed without security review. Produces a continuously updated visual map of the Agentic Fabric: Agents → MCP Servers → Tools → APIs → Enterprise Systems.

02

MCP Posture Management

Continuously evaluates every MCP server against a library of security policies covering authentication, authorization, tool exposure, data access controls, and external API connectivity. Risk is automatically classified and prioritized so security teams address the highest-exposure servers first.

03

Agentic Runtime Protection

Monitors agent behavior across MCP server interactions in real time. Detects anomalous tool invocations, data access patterns consistent with exfiltration, prompt injection attempts, and privilege escalation sequences. Response is fast enough to interrupt actions before they complete.

04

Policy Governance Engine

Enforces least-privilege tool access policies across the MCP layer. Define which agents can invoke which tools, under what conditions, and with what approval requirements. Policies are applied at the tool level — enabling fine-grained governance at scale.

05

Board-ready reporting

Translates agentic security posture into the metrics executives and boards need: MCP server inventory, risk trend over time, policy violations, incident summaries, and regulatory evidence packages. Designed to be included directly in board-level security reporting.

Why Salt

Salt invented API security and is now defining its successor: Agentic Security. The Agentic Security Platform is built on the same behavioral analysis engine that has protected APIs for the world's largest financial institutions, healthcare systems, and retailers. MCP security is not a new product line. It is the next layer of a platform that already knows your environment.

What to do in the next 90 days.

30	**Run a discovery scan** Inventory every MCP server across your environment. You will almost certainly find servers you did not know existed. Present the results to your leadership team as evidence of the current exposure surface.
60	**Complete a posture assessment** Evaluate every discovered MCP server against your security baseline. Identify servers running without authentication, with excessive tool exposure, or connected to sensitive APIs without adequate controls. Begin remediation on the highest-risk findings.
90	**Establish an MCP governance policy** Define deployment approval requirements, authentication standards, tool access policies, logging requirements, and incident response procedures for MCP servers. Integrate MCP server inventory into your existing CMDB and vendor risk management program.