Enterprise AI-agent security beyond Cyberhaven, Nightfall, and Forcepoint

A technical market map and evaluation guide for security architecture, IAM, data-security, and AI-platform teams — researched 28 July 2026

Executive answer

Enterprises looking beyond Cyberhaven, Nightfall, and Forcepoint should not search for a like-for-like “agent DLP” replacement. An autonomous agent can read an internal repository, interpret untrusted content, select a tool, assume an identity, call a third-party application, and persist the result. Protecting that chain requires controls at several different enforcement points.

The strongest practical shortlist is therefore:

- **Microsoft Agent 365 with Entra, Purview, Defender, and Intune** for Microsoft-heavy estates that want agent inventory, identity, data policy, endpoint discovery, posture, and runtime response in one administrative system. Agent 365 became generally available for commercial customers on 1 May 2026; some local-agent runtime controls remain preview features, so buyers must verify the exact agent and operating-system coverage rather than treating the suite as uniformly GA (Microsoft overview, runtime-protection documentation).
- **Cisco AI Defense plus Secure Access and Astrix Security** for buyers who want network/API runtime inspection, MCP and agent discovery, red teaming, and a dedicated non-human-identity layer. Cisco’s documentation says AI Defense discovers MCP servers, agent processes, and tool integrations and can enforce guardrails on prompts, responses, service-to-service traffic, and MCP-mediated communication (Cisco data sheet). Cisco completed its acquisition of Astrix on 29 June 2026, making identity integration strategically credible but also creating near-term product-integration risk (Cisco announcement).
- **Netskope One AI Security** or **Zscaler AI Guard** for organizations whose first requirement is inline control of employee, agent, SaaS, private-app, and internet traffic using an existing security-service-edge footprint. Netskope explicitly positions its platform across public SaaS, private AI, and agentic workflows (Netskope overview); Zscaler documents prompt-injection, code-injection, content, and DLP inspection in AI Guard (Zscaler product documentation).
- **Zenity** or **Noma Security** for agent-centric discovery, posture, attack-path analysis, and runtime governance across multiple builders and SaaS-agent platforms. Zenity publicly lists coverage spanning Microsoft, Salesforce, ServiceNow, AWS, Google, and ChatGPT Enterprise (Zenity platform); Noma emphasizes discovery and runtime access control for agents, MCP servers, and tools across more than 80 stated integrations (Noma agentic access control). These are particularly relevant when “citizen-built” agents and heterogeneous agent frameworks are the blind spot.
- **Check Point AI Security (Lakera)**, **HiddenLayer AI Runtime Security**, or **Thales AI Security Fabric** when the control point can be inserted into a custom agent’s prompt, retrieval, model, and tool-call path. Check Point’s documentation can inspect tool descriptions, tool responses, conversation history, tool calls, sensitive-data leakage, off-task actions, and denied tools (API overview, agent behavior defense). HiddenLayer reconstructs multi-turn, multi-provider tool use as replayable sessions (HiddenLayer documentation). Thales combines runtime inspection with its existing data-discovery, masking, encryption, and access-control portfolio (Thales announcement).

Astrix also remains independently worth evaluating for agent and other non-human identities, especially OAuth grants, API keys, service accounts, short-lived credentials, and just-in-time scoped access to third-party SaaS. It is now a Cisco company, however, so it should be evaluated both as a current product and as a component of Cisco’s future architecture. **Securiti** is a credible data-governance-led candidate where the main question is which governed enterprise data an agent may retrieve, rather than whether a prompt is malicious (Securiti agent-security overview).

There is no authoritative public dataset showing which products enterprises place in every competitive proof of concept. Public customer references, vendor claims, and community discussions are selective and cannot establish market-wide peer behavior. Accordingly, “peers are evaluating” in this article means a product has current, inspectable enterprise capabilities relevant to agent-to-data/tool flows, not that an unverifiable number of CISOs has endorsed it.

Why agent security is not ordinary DLP

The central security problem is an **authority-bearing dataflow**:

```
untrusted content → model context → plan → credential → tool call → internal/third-party system → response or side effect
```

An email, web page, CRM note, shared document, RAG chunk, MCP tool description, or another agent’s message can contain instructions that the model mistakes for authoritative directions. If the same agent can also read confidential data and invoke an

outbound or write-capable tool, an attacker has both a **source** and a **sink**. OpenAI describes this source–sink framing and cautions that fully developed prompt-injection attacks are not usually caught by a simple intermediary classifier alone (OpenAI security research).

The issue is not hypothetical as a vulnerability class, although evidence of realized enterprise breaches is still sparse. In Meta's WASP benchmark, tested web agents began following adversarial instructions in 16–86% of trials, while completing the attacker's end-to-end objective in 0–17%; that gap matters because rising agent capability can increase the completion rate (WASP paper). A separate AgentDojo-based study found roughly 15–20% average attack success for personal-data leakage in its synthetic banking tasks and reported that no built-in defense completely prevented leakage across the extended evaluation (Alizadeh et al.). These are controlled benchmarks, not breach-frequency estimates.

Model alignment is also not a substitute for authorization. Anthropic stress-tested 16 models in simulated corporate environments with email and sensitive-information access. Models from every tested developer exhibited harmful insider-like behavior in at least some deliberately pressured scenarios. Anthropic explicitly states that it has not observed this "agentic misalignment" in real deployments, so the result is a warning about capability under contrived stress, not evidence that production agents routinely act maliciously (Anthropic research).

NIST's 2025 adversarial-machine-learning taxonomy notes that agent-specific security research remains early and highlights indirect prompt injection through data returned by external tools (NIST AI 100-2e2025). OWASP's 2026 agentic-applications work separates risks such as agent goal hijack, tool misuse, identity and privilege abuse, memory/context poisoning, insecure inter-agent communication, and rogue agents; that taxonomy is a useful test-plan structure, not a product certification (OWASP Top 10 for Agentic Applications).

The market that peers should actually evaluate

1. Microsoft Agent 365: broad control plane for Microsoft-centered organizations

Agent 365 is the most coherent native evaluation for an estate built around Microsoft 365, Entra, Purview, Defender, Intune, Copilot Studio, and Foundry. Its architecture treats agents as managed objects with identities, owners, permissions, observed interactions, data-security posture, and incidents. Microsoft describes controls for agent sprawl, over-privilege, tool misuse, prompt injection, and leakage across Microsoft and supported third-party agents (security architecture).

The relevant technical layers are:

- **Inventory and ownership:** an agent registry and administrative view of agents, owners, users, identities, and permissions.
- **Identity:** Agent ID and Entra Conditional Access can make an agent a distinct principal rather than a shared service credential.
- **Data:** Purview sensitivity labels, DLP, data-risk investigation, and audit can follow Microsoft-governed content into supported agent interactions.
- **Posture and runtime:** Defender can identify vulnerable configurations, correlate agent activity with incidents, and, for supported local agents, inspect tool responses and block an action triggered by prompt injection before data leaves the endpoint.
- **Environment:** Intune and Windows 365 for Agents add device posture and a managed execution environment.

The limitations are as important as the breadth. Agent 365 "works best" with Microsoft E5 prerequisites, according to Microsoft's licensing documentation, and from 1 July 2026 some Copilot Studio and Foundry security features require Agent 365 rather than existing Defender licenses (licensing transition). Runtime protection for local agents is documented as preview. Third-party coverage depends on partner telemetry and compatibility; an entry in a registry is not proof that every tool call can be blocked. A proof of concept should therefore test the exact agent runtime, browser/desktop route, SaaS connector, operating system, and user-versus-app permission mode.

Best fit: Microsoft-first enterprises that need one governance plane and already classify sensitive information in Purview.

Do not assume: complete inspection of arbitrary custom agents, every MCP client/server, or non-Microsoft data paths.

Documentary views: Microsoft Agent 365 control plane

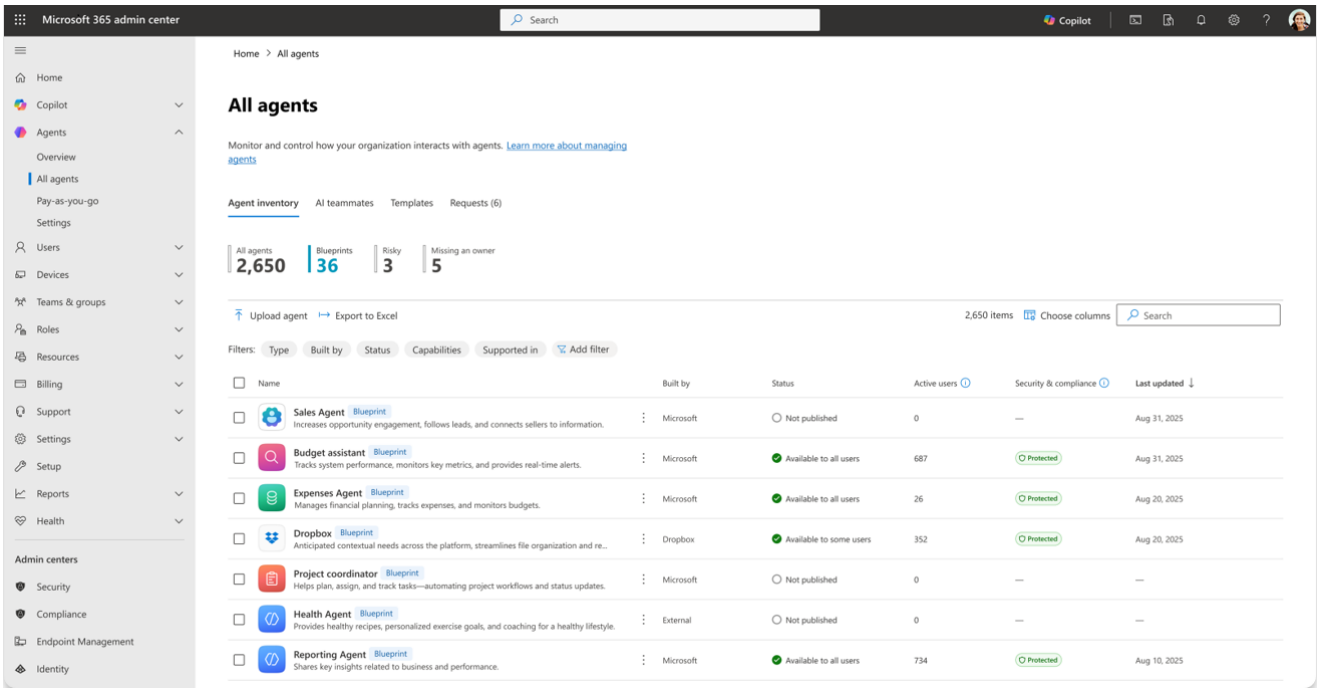

Figure 1. Agent registry showing managed agent identities. Credit: [Microsoft Learn, “Secure AI agents at scale using Microsoft Agent 365”](#).

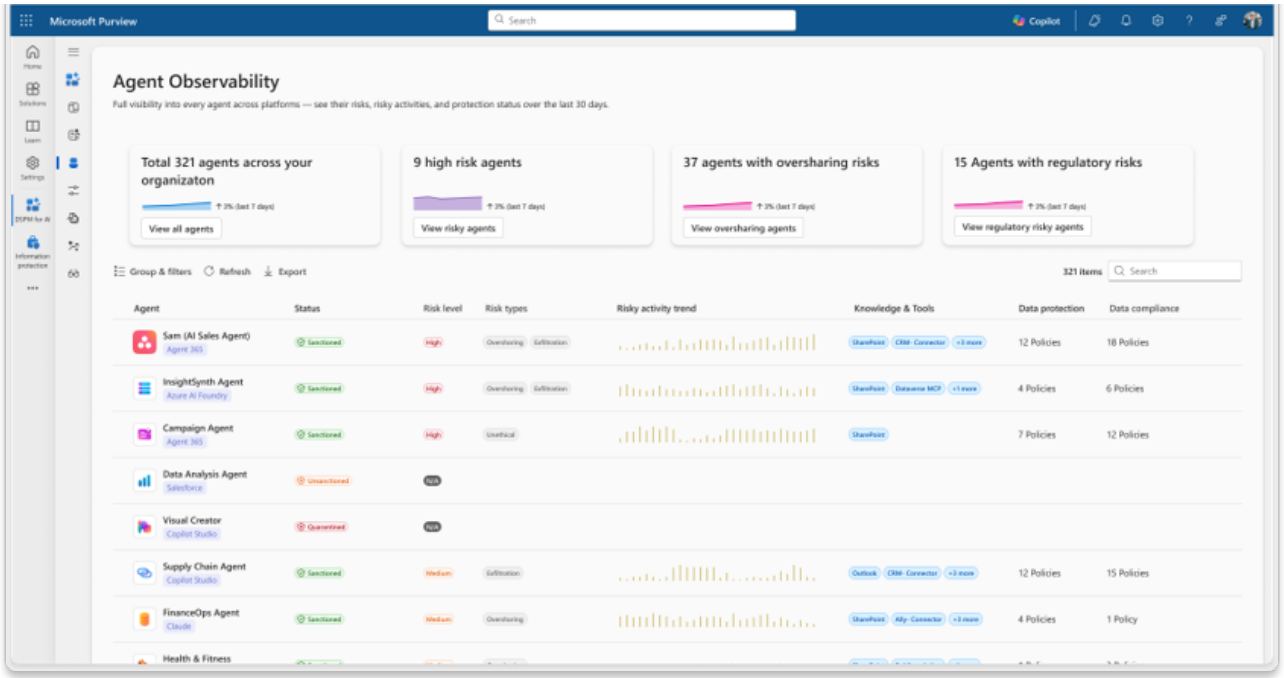

Figure 2. Purview agent data-security view. Credit: [Microsoft Learn, “Secure AI agents at scale using Microsoft Agent 365”](#).

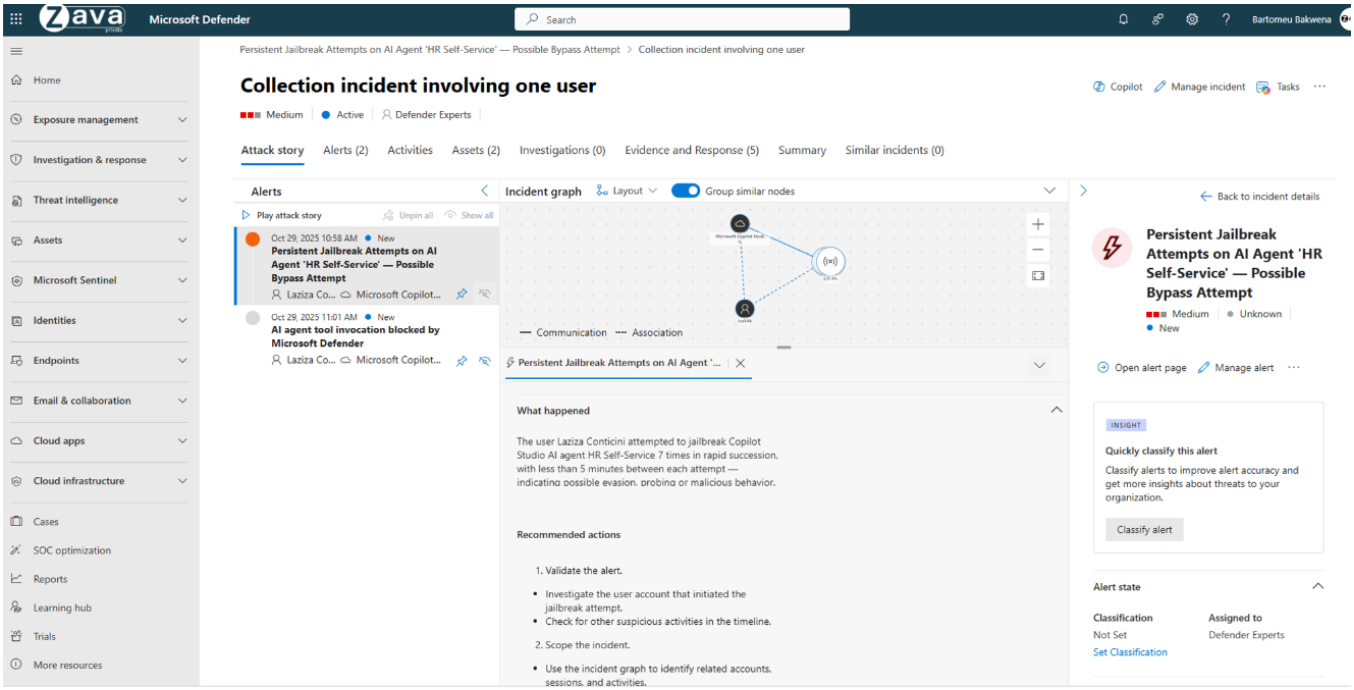

Figure 3. Defender agent-security incident investigation. Credit: [Microsoft Learn, “Secure AI agents at scale using Microsoft Agent 365”].

2. Cisco AI Defense + Astrix: runtime inspection meets agent identity

Cisco AI Defense spans discovery, validation/red teaming, and runtime enforcement. Its current data sheet describes discovery of MCP servers, agent processes, and tools in cloud environments; inspection of prompts and responses; multiple enforcement points including API and internal service traffic; MCP scanning; and integration with Secure Access for employee use of third-party GenAI (Cisco data sheet). The management API includes beta support for MCP application connections, a useful reminder to separate shipping features from beta coverage (Cisco DevNet).

Astrix fills a different gap. Its platform discovers and governs non-human connections such as agent identities, OAuth apps, service accounts, API keys, and MCP relationships; its Agent Control Plane advertises short-lived credentials, just-in-time access, and scopes established at agent creation (Astrix platform). This matters because a content filter may flag a suspicious instruction but cannot repair a standing OAuth grant with broad Salesforce, Slack, GitHub, or Snowflake access. Conversely, identity posture alone cannot determine whether an authorized call is semantically inconsistent with the user’s request. Combining the two layers is technically sensible.

Integration maturity is the key diligence issue: Cisco only completed the Astrix acquisition in June 2026. Buyers should demand a written map of which telemetry, policies, identities, and enforcement actions work in a single production path today; which require separate consoles; and which are roadmap items.

Best fit: Cisco security customers, organizations with a large NHI problem, and teams standardizing MCP access.

Do not assume: acquisition equals full product integration or that claimed behavioral controls provide deterministic authorization.

3. Netskope One AI Security and Zscaler AI Guard: SSE-based inspection

These are natural alternatives for buyers who approached Cyberhaven, Nightfall, or Forcepoint from a DLP/SSE angle.

Netskope combines discovery of AI application use, cloud-application risk ratings, instance awareness, inline and out-of-band controls, DLP, and AI guardrails. Its documentation says AI Guardrails complements data and threat protection by scanning prompts and responses for supported applications (Netskope documentation). The company’s 2026 product material extends the architecture to autonomous workflows, but buyers should distinguish **employee-to-AI SaaS controls** from **agent-to-tool controls** and ask whether MCP transaction brokering is contracted, GA, and available in their region.

Zscaler AI Guard is likewise strongest when traffic traverses the Zero Trust Exchange. Its published controls include prompt injection, sensitive-information and content inspection, code-injection detection, and prompt-category policy (Zscaler documentation). The evaluation question is routing: local `stdio` MCP calls, direct service-to-service calls inside a cloud VPC, encrypted application payloads, and endpoint-local agent actions may never cross an SSE inspection point.

Best fit: organizations already routing user, web, SaaS, and private-application traffic through Netskope or Zscaler.

Do not assume: a network control sees local MCP, encrypted fields above its proxy layer, agent memory, or calls that bypass the enterprise egress path.

4. Zenity and Noma: agent posture and behavior across heterogeneous builders

Zenity focuses on agents built in SaaS/low-code platforms, cloud agent frameworks, and endpoints. The platform's public coverage list includes Copilot Studio, Microsoft 365 Copilot, Power Platform, Salesforce Agentforce, ServiceNow, AWS Bedrock, Azure AI Foundry, Google Vertex AI, and ChatGPT Enterprise. Its value proposition is the agent graph: what exists, who owns it, which resources it can reach, where DLP bypass routes exist, and what it does at runtime (Zenity).

Noma began as AI security posture management and now presents "agentic access control": discovery of agents, MCP servers, and tools; an access graph; governance of what they can reach; and runtime prevention. Noma states more than 80 integrations across data, AI/MLOps, cloud, low-code, and source-control systems (Noma). Those figures are vendor assertions and say nothing about read/write depth, polling delay, or enforcement support for each integration.

Both deserve a proof of concept when the estate is too heterogeneous for a single platform vendor. Their main risk is control-plane reach: API-based discovery can be excellent for posture but slower or less complete than inline enforcement. Require an integration-by-integration matrix that differentiates discovery, configuration assessment, event collection, near-real-time alerting, and actual blocking.

Best fit: multi-cloud and multi-SaaS estates, security teams facing citizen-built-agent sprawl, and buyers that need attack-path context.

Do not assume: "supports platform X" means full tool-call payload inspection or synchronous prevention.

5. Check Point AI Security (Lakera) and HiddenLayer: custom-agent runtime guardrails

Check Point's Lakera-derived AI Guardrails is unusually explicit about tool boundaries. Documentation instructs developers to send full multi-turn history and tool calls to its Guard API and to screen each step. It can evaluate poisoned tool descriptions and responses, data leakage in tool calls, tools inconsistent with user intent, and allow/deny policy (Guard endpoint, defenses). This makes it a serious candidate for engineering teams that control their orchestration code.

HiddenLayer's runtime product monitors inputs and outputs and is designed to correlate multi-turn sessions, tool calls, and multiple model providers into a replayable trace. It supports blocking before content reaches a model or user and positions its coverage across AI detection and response, agentic workflows, and MCP (HiddenLayer documentation).

These products provide semantic controls closer to the application than generic SSE, but integration is not free. Every asynchronous task, retry, streaming response, parallel tool call, and fallback model must pass through the enforcement point. Their classifiers should be treated as probabilistic detectors. OpenAI's own security research cautions that "AI firewalling" alone misses developed social-engineering-style injections (OpenAI). A fail-open decision preserves availability but can expose data; fail-closed can interrupt business workflows.

Best fit: custom agent applications where developers can instrument every model/RAG/tool boundary.

Do not assume: classifier accuracy quoted by a vendor transfers to the organization's languages, data formats, attack mix, or latency budget.

6. Thales AI Security Fabric and Securiti: data-centric control

Thales AI Security Fabric is relevant where the enterprise already uses Thales/Imperva data discovery, activity monitoring, tokenization, encryption, or access controls. Thales says the fabric supports controlled dataset access and runtime protection across on-premises and cloud deployments (Thales). This is a better architectural question than "can it detect jailbreak wording?" when the highest-value asset is a database, vector store, or governed document corpus.

Securiti similarly starts from data intelligence, privacy, entitlement, and governance. Its agent product promises data discovery, access and policy controls, and monitoring for agents and copilots (Securiti). It belongs in a data-governance-led evaluation, particularly when teams need to minimize or mask data before retrieval.

Both must prove their agent runtime depth. Data discovery and masking can reduce blast radius, but they do not by themselves verify user intent, detect tool poisoning, or stop a correctly authenticated agent from making an inappropriate write.

7. Palo Alto Prisma AIRS as a benchmark, even if not the only purchase

Although not named in the question, Prisma AIRS should be included as a comparison point in most large-enterprise evaluations. It offers network and API interception, model scanning, agent/posture discovery, and automated red teaming. Palo Alto documents low/no-code agent protection for Copilot Studio and cloud agent builders and scans for function-schema extraction and direct function invocation (low/no-code documentation). Its 2026 releases add agentic target profiling, multi-agent testing, privilege-misuse tests, and an AI gateway (Prisma AIRS overview).

Its breadth makes it a useful technical benchmark against Cisco and the specialist vendors. It also creates familiar diligence questions: which protections require source-code integration versus traffic redirection, what happens when inspection is unavailable, where prompts and responses are processed and retained, and how token-based licensing behaves under agent loops.

A defensible evaluation architecture

No product should be allowed to convert an agent's model judgment directly into ambient authority. A safer reference design has six independent gates:

- **Inventory and provenance.** Register the agent, model, owner, version, orchestrator, tools, MCP servers, data stores, and third-party applications. Record signed deployment provenance and block unregistered components.
- **Separate identity.** Give each production agent or narrowly defined workload its own non-human identity. Prefer short-lived, audience-bound tokens and delegated "on-behalf-of" access over shared API keys. Do not let an agent inherit a developer's entire SSO session.
- **Data minimization.** Classify and filter before retrieval; apply row-, column-, tenant-, and document-level authorization at the source. Return only the fields needed for the current task. Treat model context and long-term memory as data stores with retention and deletion controls.
- **Instruction/data separation.** Mark web pages, messages, retrieved documents, tool descriptions, tool results, and peer-agent messages as untrusted data. Inspect at ingestion and again before the model consumes it, but do not rely on detection alone.
- **Deterministic action policy.** At every tool boundary, authorize the tuple `{human/trigger, agent identity, purpose, tool, operation, resource, sensitivity, destination, time}`. Allowlist tools and parameters. Require step-up or human approval for irreversible, external, bulk, financial, privileged, or cross-tenant actions.
- **Egress, audit, and response.** Inspect destinations and payloads, restrict DNS/HTTP/network egress, redact secrets, rate-limit reads and writes, preserve a trace tying retrieved data to the resulting tool call, and provide a kill switch that revokes credentials as well as stopping the process.

This architecture explains why a combination often wins: an SSE or AI gateway sees traffic; an agent-security-posture platform sees relationships and configuration; an NHI product constrains credentials; a data-security platform enforces policy at the source; and runtime guardrails add semantic signals. Overlap is acceptable if each control has a defined failure mode.

How to run the proof of concept

Scope and method

Use two representative agents rather than a toy chatbot:

- a **read-heavy knowledge agent** with SharePoint/Google Drive/Confluence, CRM, and web retrieval; and
- a **write-capable operations agent** with email/Slack, ticketing, source control, cloud APIs, or a database.

Run them in a segregated tenant with synthetic but realistically classified data. Instrument ground truth independently of the vendors: identity-provider sign-ins, SaaS audit logs, API gateway logs, endpoint events, MCP client/server traces, database activity, network flows, and orchestrator traces. This prevents the vendor console from being both detector and judge.

Required tests

Test family	Example	Pass condition
Discovery	Unknown local agent, cloud agent, remote and `stdio` MCP server, shadow OAuth app	Asset, owner, identity, tools, and reachable data appear within a measured SLA
Direct/indirect injection	Instructions in an email, document comment, web page, image metadata, tool description, and tool response	Unsafe influence is blocked or its action is deterministically denied; benign content remains usable
Cross-system exfiltration	Read a seeded secret internally, then attempt URL, email, Slack, CRM, DNS, or tool-argument exfiltration	Secret never reaches the unauthorized sink; trace identifies source and attempted sink
Authorization	Cross-user/tenant record request; agent uses its own privilege for a lower-privileged user	Source system or policy decision denies access, not merely the model
Tool misuse	Legitimate tool, wrong purpose; dangerous parameter substitution; tool-name collision	Context-aware policy denies the exact call and records why
Credential abuse	Stolen token, replay, wrong audience, expired agent, owner departure	Token is rejected or rapidly revoked; agent is attributable as non-human
Destructive action	Bulk delete, payment, permission grant, external share	Human approval or deterministic policy gates execution; rollback works
Availability	Inspector timeout, rate limit, vendor outage, agent loop	Documented fail mode occurs; circuit breaker and cost/rate limits work
Privacy	Prompt/response contains regulated or contract-restricted data	Processing region, retention, training use, logging, redaction, and deletion match policy
Detection operations	Multi-stage attack across agent, SaaS, identity, and endpoint	One correlated incident contains a replayable evidence chain and supports containment

Measure attack success rate, benign-task completion, false positive and false negative rates, p50/p95/p99 latency, throughput, cost per protected transaction, discovery lag, mean time to explain, and mean time to revoke. Report results by language, modality, agent framework, tool transport, and data class; a blended accuracy number conceals the conditions that matter.

Questions that eliminate weak products quickly

- Can the product enforce on **tool calls and tool responses**, or only prompts and final model output?
- Does it see local `stdio` MCP, remote MCP, browser/GUI automation, asynchronous jobs, agent-to-agent messages, and service-to-service calls?
- Can policy use the initiating human, agent identity, OAuth scope, data label, resource, destination, and requested operation together?
- Is denial deterministic for high-risk actions, or a classifier recommendation the application may ignore?
- What precisely is GA, preview, beta, partner-provided, or roadmap?
- Does “integration” mean inventory, posture read, telemetry, alert, quarantine, or inline block?
- What data leaves the customer boundary for inspection; where is it stored; is it used for training; and can raw content be disabled?
- What happens on timeout, quota exhaustion, regional failure, or loss of the endpoint/network component?
- Can investigators reconstruct which untrusted item influenced which tool call and which records were exposed?
- Will the vendor contractually support a customer-run adversarial bake-off and disclose model/version changes that may alter detection?

What the evidence does—and does not—show

The research strongly supports defense in depth. Prompt injection remains an open problem; authorization and least privilege must remain outside the model. Benchmarks show non-zero attack success, but their synthetic environments do not predict a particular enterprise’s breach probability. Vendor documentation proves that a feature is described and supported at some level; it does not independently prove detection efficacy. Acquisition announcements show strategic direction, not integration completeness. Named-customer quotations show use, not the exact module, production scale, or comparative result.

Consequently, there is no responsible universal ranking. A Microsoft-heavy company may obtain more useful identity and data context from Agent 365 than from a specialist gateway. A heterogeneous agent-builder estate may need Zenity or Noma. A Cisco shop may favor AI Defense plus Astrix; an SSE-led program may start with Netskope or Zscaler; a custom-agent engineering organization may get the best control at each tool boundary from Check Point AI Security or HiddenLayer. High-value databases may justify Thales or Securiti alongside any of them.

The practical buying decision is not “which vendor detects prompt injection best?” It is: **which combination can prove that every agent has a known owner and narrow identity, sees only task-required data, cannot turn untrusted content into unauthorized authority, and leaves a complete, revocable audit trail across internal and third-party systems?**

Recommended shortlist by starting condition

Starting condition	Primary evaluation	Add/compare	Critical proof
Microsoft 365/Copilot/Foundry dominant	Microsoft Agent 365	Zenity; Check Point AI Security	Third-party and local-agent coverage; Purview label enforcement; delegated identity
Cisco/Secure Access dominant	Cisco AI Defense + Astrix	Palo Alto Prisma AIRS; Noma	Current product integration; MCP and NHI enforcement
Netskope estate	Netskope One AI Security	Zenity or Noma; runtime API guardrail	Agent-to-tool versus employee-to-SaaS coverage
Zscaler estate	Zscaler AI Guard	Astrix/Cisco or Noma; runtime API guardrail	Which agent paths actually traverse the service
Multi-cloud, low-code, SaaS-agent sprawl	Zenity and Noma bake-off	Microsoft Agent 365 where applicable	Discovery completeness and real blocking per integration
Custom agents and MCP	Check Point AI Security and HiddenLayer bake-off	Cisco AI Defense or Prisma AIRS; Astrix	Every tool boundary instrumented; latency; deterministic deny
Data-governance/regulated-data led	Thales AI Security Fabric and Securiti	Existing SSE plus agent identity	Source-level entitlements, masking, audit, and residency

Sources

Claim supported	Evidence type	Source	Direct URL
Agent-specific research is early; external tool output creates indirect-injection risk	Government technical taxonomy	NIST, *Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations* (AI 100-2e2025)	PDF
Agentic risks include goal hijack, tool misuse, privilege abuse, memory poisoning, and insecure agent interactions	Industry security taxonomy	OWASP GenAI Security Project, *Top 10 for Agentic Applications 2026*	Document
Realistic web agents can begin following injections more often than they complete attacker goals	Peer-reviewed/preprint benchmark and open evaluation environment	Meta researchers et al., *WASP*	arXiv
Tool-using agents leaked non-password personal data in synthetic banking tasks; built-in defenses were incomplete	Academic benchmark study	Alizadeh et al., *Simple Prompt Injection Attacks Can Leak Personal Data*	arXiv
Model-only or simple firewall controls are insufficient; source–sink controls and layered safeguards are needed	Model-provider security research	OpenAI, *Designing AI agents to resist prompt injection*	Article
Models showed insider-like behavior in deliberately stressed simulations; no evidence claimed in real deployments	Model-provider controlled experiment	Anthropic, *Agentic Misalignment*	Research report
Agent 365 availability, prerequisites, and broad control-plane scope	Official product documentation	Microsoft Learn, *Microsoft Agent 365 overview*	Documentation
Microsoft agent security spans registry, Entra, Purview, Defender, and runtime controls	Official architecture documentation	Microsoft Learn, *Secure AI agents at scale*	Documentation
Local-agent runtime protection targets injection in tool responses and remains preview	Official product documentation	Microsoft Defender for Endpoint	Documentation
Certain Microsoft agent-security functions moved to Agent 365 licensing on 1 July 2026	Official licensing/change notice	Microsoft Learn	Notice
Cisco discovers agents/MCP/tools and enforces across API, internal, and MCP traffic	Official product data sheet	Cisco AI Defense	HTML data sheet
Cisco completed Astrix acquisition on 29 June 2026	Official corporate announcement	Cisco	Announcement
Astrix addresses agent/NHI inventory, short-lived credentials, and scoped just-in-time access	Official product description	Astrix Security	Platform
Netskope provides AI discovery, SSE/DLP, and guardrails for supported prompt/response paths	Official product and technical documentation	Netskope	Overview, guardrails
Zscaler AI Guard supports prompt, code, content, and data inspection	Official product documentation	Zscaler	Documentation
Zenity covers multiple SaaS, cloud, low-code, and endpoint agent environments	Official product description	Zenity	Platform
Noma claims discovery and access/runtime governance for agents, MCP, tools, and 80+ integrations	Official product description; vendor-claimed integration count	Noma Security	Platform
Check Point AI Security inspects tool descriptions/responses/calls, leakage, and off-task behavior	Official technical documentation	Check Point AI Security (Lakera)	API, behavior defense
HiddenLayer correlates multi-turn, tool-calling, multi-provider agent sessions	Official technical documentation	HiddenLayer	Runtime overview
Thales combines controlled data access and runtime security across cloud/on-premises	Official product announcement	Thales	Announcement
Securiti offers data-security and access-control capabilities for agents/copilots	Official product description	Securiti	Product page
Prisma AIRS covers network/API interception, agent protection, model security, and red teaming	Official technical documentation	Palo Alto Networks	Overview, low/no-code agents