INTERRAIT
A FERFIER GROUP COMPANY

WHITEPAPER

Securing the Agentic Enterprise

A delivery playbook, not a prediction.

Abstract

The market does not need another prediction. Enterprise AI agents now hold credentials, call tools, and act across production systems with little human oversight; yet most security stacks were never designed to see them.

This paper is written from the delivery chair: for the CISOs, VP-Engineering, and platform leaders accountable when an agent does something they shouldn't. Rather than forecasting the threat, it sets out an operating model — how to inventory, scope, test, monitor, and govern agents in production.

It closes with an Agentic Security Maturity Model teams can use to locate themselves honestly and a 90-day path to act — ensuring control is reflected in how agents are built and run, not only in policy documentation.

1. The Shift Nobody Budgeted For

1.1 The Perimeter Is Now the Agent

The security perimeter has moved before, from the network to human identity. In 2026, it has moved again, and this time it is far harder to see.

A New Class of Identity
An AI agent is a non-human entity that holds credentials, invokes tools and APIs, and operates continuously at machine speed. Enterprises are deploying them by the thousand.

Built for Humans, Not Machines
Nearly every access model in your estate assumes people: working hours, familiar places, a bounded set of actions. Agents are always-on and can be handed far more privilege than the task in front of them requires.

Adoption Outpacing Control
The spending is real; the visibility is not. Most organizations shipped the agent before they secured it, and few can name every agent is now running, let alone governing it.

1.2 Why This Matters Now

The gap between how fast agents are adopted and how slowly they are secured is the risk this paper addresses.

$240B
Projected global information- security spending in 2026, up ~12.5%. (Gartner)

82%
Of analysts fear missing real threats under alert volume. (Google Cloud, 2026)

4/5
Enterprise stacks not equipped to detect a compromised agent.

1/4
Enterprises with a dedicated AI-security function.

Governance has barely started while adoption accelerates into the core of the business. The chapters that follow move from what goes wrong to why current tooling misses it to a delivery playbook for addressing it.

2. Five Ways an Agent Gets Turned Against You

These are not hypotheticals; they are the failure modes security teams are already investigating. Read each as a question: could it happen here, and would we know if it did?

A. Prompt injection
Instructions hidden in content the agent reads; a web page, a ticket, a PDF. It bypasses the perimeter because it is not an exploit; it is input, and the agent was built to trust input.

B. Tool misuse & privilege escalation
An agent granted broad access “to be safe” is steered into tools the task never needed. The over-permissioning is the vulnerability.

C. Memory poisoning
A malicious instruction planted in long-term memory persists across sessions; the agent acts on it days later, long after the entry point is forgotten.

D. Stolen agent identity
A captured token carries valid credentials, so the network can’t tell the real agent from the impostor. The blast radius is brutal when one orchestrator holds the keys to many systems.

E. Supply-chain compromise
Attackers poison the models, libraries, and tools on which agents depend. The agent runs as designed; the design was compromised upstream.

3. Why Your Current Stack Can’t See It

Your tools were engineered to catch anomalies in human behavior. Agents defeat that logic by being relentlessly, perfectly consistent.

> An agent that runs code flawlessly ten thousand times looks normal to tools built to catch people.

The signal hides inside legitimate activity
The malicious action doesn’t arrive as a spike; it hides in a stream of machine-speed, legitimate-looking calls. Static rules written for yesterday’s threats can’t keep pace.

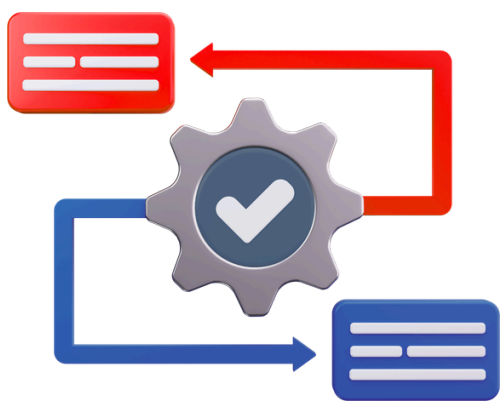
You can’t simply hire your way out
Demand for adversarial-AI testing far outstrips supply, and fragmented best-of-breed tooling fractures visibility across discovery, identity, posture, and data.

> You can’t hire your way out fast enough. You have to engineer your way out, turning agent security into a repeatable, automated discipline built into how software ships.

4. The Delivery Playbook

Six disciplines that combine people, process, and automation. None requires a tool you can't buy today; they require that someone owns them and runs them on a cadence.

01 **Discover & inventory**	Stand up an agent registry: identity, owner, purpose, data class, and permitted tools for every agent; then hunt the shadow AI that isn't on it. Good: new agents are registered before production; unknown ones are flagged automatically.
02 **Scope identity & least privilege**	Unique per-agent credentials, short-lived tokens, and tool access scoped to the task. Kill the single orchestration identity that holds standing access to everything. Good: a breached agent reaches only what its task needs, and its token has already expired.
03 **Harden before you ship**	Prompt-injection and tool-abuse test suites in the delivery pipeline; red-team before production; re-run on every prompt or model change. Good: every agent is a release candidate with a security gate to pass; not a science experiment.



04 **Observe at machine speed**	Log every tool call, baseline each agent, and alert on deviation. Wire in circuit breakers and a kill switch to halt an agent the instant it acts outside its envelope. Good: an abnormal agent is stopped automatically, and a human is told why; in seconds, not hours.
02 **Contain the blast radius**	Segment agents, scope secrets, rate-limit high-impact actions, and gate anything irreversible behind a human- in-the-loop. Good: the worst, realistic outcome of one compromised agent is small, reversible, and quickly visible.
03 **Govern & prove**	Map controls to NIST AI RMF and the OWASP LLM Top 10 and to EU AI Act obligations. Assign accountability per agent; treat governance as continuous, not annual. Good: a current, evidenced answer whenthe board, an auditor, or a regulator asks how you control your agents.

FROM THE FIELD

AI-powered vulnerability management

Vulnerability management isoneofthe first placesenterprises are giving AI agents real authority. In a recent deployment, a large enterprise put a team of agents to work across the workflow:

- **Rapid7 vulnerability analysis**
- **ServiceNow ticket creation**
- **Teams notifications**
- **Patch-deployment recommendations**

The automation brought obvious upside — and a new attack surface, because each agent now held standing access to security findings and change systems. Running just the first two moves of this playbook, **build an agent inventory**, then **implement least privilege;** InterraIT eliminated a large portion of that attack surface **before investing in advanced monitoring, testing, or runtime defense.**

5. The Agentic Security Maturity Model

Locate your organization honestly; mark where you actually are today. The goal isn't Level 3 everywhere overnight; it's knowing which row is weakest and fixing that one next.

Dimension	L0 Ad hoc	L1 Aware	L2 Managed	L3 Resilient
Visibility	No inventory; shadow AI unknown.	Manual list of known agents.	Registry: owner, purpose, data class.	Continuous discovery; shadow AI auto-flagged.
Identity & access	Shared keys; standing admin.	Per-agent keys, broad scope.	Least-privilege; short-lived tokens.	Just-in-time access; auto scope reviews.
Testing & hardening	Ships untested against abuse.	Occasional manual checks.	Injection & tool-abuse suite in pipeline.	Continuous red-teaming; regression on change.
Runtime defense	No agent-specific monitoring.	Basic tool-call logging.	Baselines & alerting per agent.	Real-time anomaly detection & kill switch.
Governance & proof	No ownership or policy.	Informal guidelines.	Mapped to NIST AI RMF & OWASP.	Audit-ready; continuous compliance.

Most organizations sit between L0 and L1 on visibility and identity, with little on testing and runtime defense. The first two moves — inventory and least privilege — remove a disproportionate share of the risk before you spend on advanced tooling.

6. Where to Start: The First 90 Days

You don't need a transformation programme to begin — you need three focused sprints.

Days 1–30 · See it
Stand up the agent registry, run a shadow-AI sweep, and rank every agent by data sensitivity and business impact.

Days 31–60 · Shrink it
Give high-risk agents unique, short-lived credentials and the least privilege; run an injection & tool-abuse suite; remove any all-powerful identity.

Days 61–90 · Sustain it
Add runtime monitoring and a kill switch; gate irreversible actions behind a human; map controls to NIST AI RMF & OWASP and assign an owner per agent.

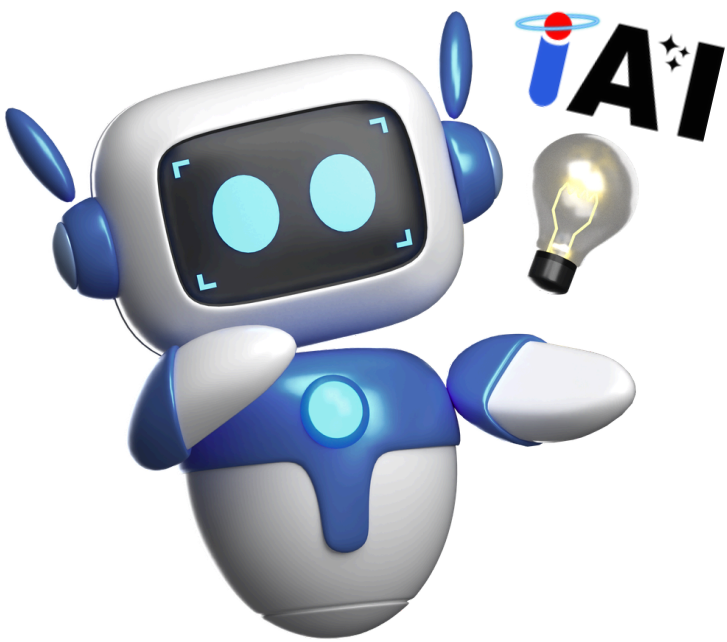
Ninety days in, you move from not knowing what your agents can do to controlling what they're allowed to do; the entire difference between the two classes of organization that 2026 will produce.

Conclusion

Predictions are cheap; operating models are rare. The organizations that come through this era well won't have the sharpest forecast about agentic risk.

They'll be the ones who quietly did the unglamorous work: inventoried their agents, scoped their access, tested them before shipping, watched them at runtime, and could prove all of it.

That work is engineering, not prophecy, and engineering is something you can start this quarter.



InterraIT

A global technology consulting firm with over 25 years of experience in Data Engineering and Digital Transformation with a history of automation.

Our iAI Services empower enterprises with intelligent automation, document governance, and data-driven compliance solutions.

United States I Canada I India I Australia

Let's Connect

InterraIT's AI Services

Cybersecurity Services