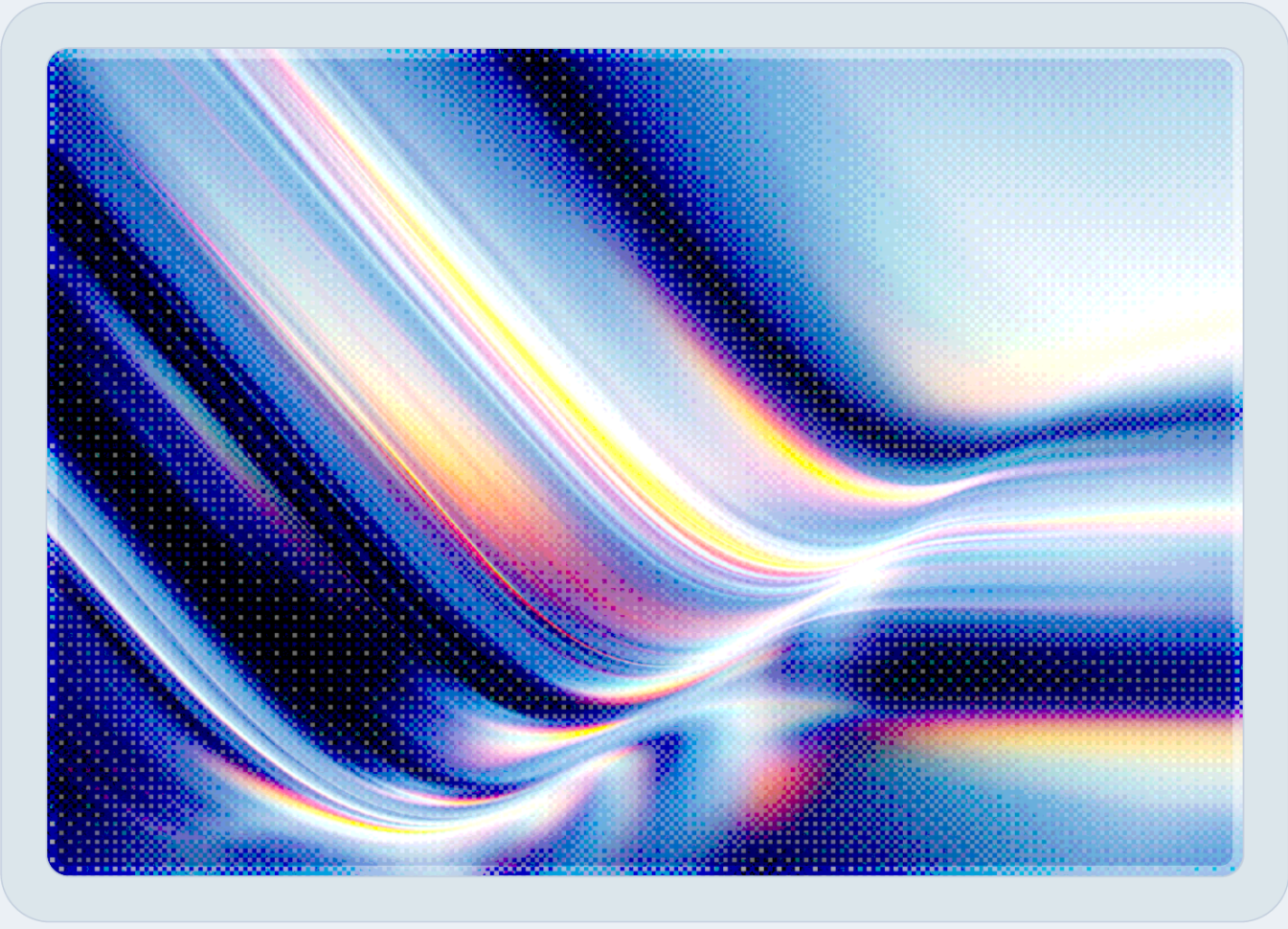
CSAI Foundation | Cloud Security Alliance

Legacy Infrastructure: The AI Agent Security Blind Spot

Systemic Risk in Hybrid Agentic Deployments

2026-06-22

AI-assisted Rapid Research



CSAI

CSA cloud security alliance®

Key Takeaways

Enterprise organizations deploying agentic AI face a risk they have been slow to address: the legacy infrastructure those agents connect to. A January 2026 CSA survey of 418 security professionals found that 82% of enterprises already have AI agents running in their environments that IT had not officially provisioned or discovered [1] — evidence that deployment has outpaced security review of the integration layer. These agents interact with operational technology, legacy databases, flat-file archives, and identity systems designed a decade or more before autonomous AI actors existed, inheriting every vulnerability those systems carry within their reach. Only 8% of organizations report high confidence that their legacy identity and access management infrastructure can handle the demands of AI and non-human identity (NHI) management [2]. Until enterprises treat the legacy integration boundary as a primary attack surface — rather than a secondary concern beneath the AI layer — agentic deployments will remain systematically exposed to threats that AI-focused security controls cannot address.

Background

Much of the current guidance on agentic AI security focuses on the AI system itself: whether a model can be manipulated through adversarial prompts, whether it will honor its constraints under pressure, or whether a tool it calls can be poisoned at the source. This framing is both accurate and incomplete. AI agents are not deployed in isolation; they are embedded in enterprise environments that often include decades of accumulated infrastructure — mainframes running batch jobs, SCADA systems governing physical processes, relational databases with no column-level access controls, and identity systems still organized around static service accounts and long-lived API keys. The security assumptions baked into each of these systems predate the agentic AI paradigm by years or decades, and none were designed to be accessed by software that reasons, delegates, and takes autonomous action across organizational boundaries.

When an AI agent is granted access to a legacy system, it inherits the attack surface of every component it can reach — authentication mechanisms, accessible data, and any privilege it can exercise — including vulnerabilities in those components that the system's original threat model never contemplated. An agent that can query a 15-year-old enterprise resource planning database to answer a user's question also inherits that database's authentication weaknesses, its logging gaps, and the accumulated

sensitivity of years of financial records it was never designed to expose to an external caller. The fundamental problem is not that legacy systems are insecure in isolation — many have operated safely for years within bounded, well-understood threat models. The problem is that connecting them to an agentic AI actor recontextualizes their exposure entirely. An AI agent's broad access, autonomous decision-making, and tolerance for chaining through multiple systems in a single task creates a threat profile that legacy architectures have no existing control for.

The scale of this problem is already visible in deployment data. Non-human identities now outnumber human identities at a ratio of 144 to 1 in cloud-native enterprise environments, and the NHI population grew by 44% between 2024 and 2025 as agentic AI deployments accelerated [3]. AI-related credential leaks surged 81.5% year-over-year during 2025, indicating that the identity layer itself has become a primary exposure surface [3]. Regulatory bodies recognized the urgency of this dynamic in December 2025, when CISA, the Australian Signals Directorate, the NSA, the FBI, and six other international agencies released joint guidance specifically addressing the risk of integrating AI into operational technology environments — an acknowledgment that AI-to-legacy connectivity has become a critical infrastructure concern [4].

Security Analysis

The Unknown Agent Problem

Security teams cannot defend an attack surface they cannot see. The most immediate challenge created by the collision of agentic AI and legacy infrastructure is that the integration often happens without formal IT involvement. The same CSA survey that found 82% of enterprises running undiscovered AI agents also found that 65% of those organizations had experienced AI agent-related security incidents in the prior 12 months, with 61% reporting data exposure, 43% reporting operational disruption, and 35% reporting financial losses [1]. Only 21% of organizations have formal AI agent decommissioning processes in place, meaning agents provisioned for a specific workflow continue operating — and accessing legacy systems — long after the workflow ends [1]. The combination of low visibility and low lifecycle management creates orphaned agent credentials that persist in legacy systems indefinitely, accumulating privilege and posing a persistent threat.

Shadow agent proliferation compounds this at scale. Employees adopt AI productivity tools without IT knowledge, and many of those tools offer integrations with enterprise systems — email, calendaring, project management, file storage, and customer relationship management platforms. Each of these may connect to deeper legacy systems behind it. A sales representative who links a consumer AI assistant to

their CRM without approval may inadvertently extend that agent's access to an underlying legacy order management system the CRM queries. These integrations bypass the access controls and monitoring infrastructure the organization has invested in, creating agent-to-legacy data flows that are invisible and never reviewed by security teams. The discovery problem is therefore not a one-time audit exercise; it is a continuous operational challenge. The same CSA survey found no improvement in governance visibility despite growing AI deployment rates [1], suggesting the gap between adoption and oversight is widening rather than closing.

Legacy IAM: Designed for Humans, Not Agents

The identity layer represents the deepest structural mismatch between legacy infrastructure and agentic AI demands. Traditional enterprise IAM was built around two classes of identity: human users with long-lived credentials refreshed by HR lifecycle events, and static service accounts representing discrete application-to-application connections. AI agents belong to neither class. They are autonomous and dynamic, capable of requesting new permissions at runtime; they delegate to sub-agents that must carry their own identity claims; and they operate across trust zones that legacy IAM models treat as separate domains requiring separate credentials. A 2025 World Economic Forum analysis found that 51% of organizations report no clear ownership of AI identities, and that accounts lacking any HR system ownership linkage persist indefinitely following the departure of their creator, accumulating access with no human accountable for their continued existence [3].

The operational consequences of this mismatch are measurable. Only 8% of organizations report high confidence that their legacy IAM systems can handle AI and NHI security risks, while 24% acknowledge that their processes take more than 24 hours to revoke a compromised credential after an exposure event is detected [2]. In a threat environment where adversaries have demonstrated average network traversal times of 29 minutes in documented incidents [15], a 24-hour revocation window provides an attacker who has already obtained the credential with ample time to complete lateral movement and data exfiltration before any response can contain the damage. When agents authenticate to legacy systems using long-lived API keys or service account credentials, the credential itself becomes the primary target. Only 15% of organizations report high confidence in their ability to prevent attacks that move through non-human identities [2], a gap that directly reflects the absence of NHI-aware controls in most legacy IAM deployments.

Operational Technology: An Unmonitored Integration Boundary

Agentic AI is being connected to operational technology environments for reasons that carry real operational benefits: agents can analyze sensor data to predict equipment failures, optimize production schedules, and assist maintenance engineers in diagnosing faults. However, OT systems were designed

for reliability and uptime in physically isolated networks, not for connectivity to externally-facing AI systems. The absence of authentication in many OT protocols, the prevalence of unencrypted communications, and the decades-long operational lifecycles that preclude rapid patching create an attack surface with characteristics fundamentally different from enterprise IT.

The threat landscape around OT is deteriorating independently of AI. Dragos documented 1,693 ransomware attacks against industrial organizations in 2024, an 87% increase over the prior year, with 80 distinct ransomware groups targeting OT environments — a 60% increase in adversary count [6]. Incident monitoring data indicates that attacker activity against ICS environments has continued to surge, with sector-specific campaigns documented across energy, manufacturing, and critical infrastructure verticals [7]. Given the operational constraints on OT patching — extended maintenance windows, uptime requirements, and the absence of automated patch deployment — this sustained attacker focus suggests that unpatched vulnerabilities in these environments are likely accumulating faster than organizations can remediate them. When an AI agent is granted connectivity to these environments as an operational efficiency measure, it can reach — and potentially exploit — vulnerabilities in the systems it touches; each integration point is a potential pivot for an attacker who has compromised the agent. The December 2025 CISA joint guidance explicitly identifies model manipulation, data poisoning, and prompt injection as risks that are qualitatively new in OT contexts, compounding the existing legacy vulnerability surface with AI-specific attack techniques [4][5].

Legacy OT security challenges include human error, outdated systems, and substantial cost barriers to remediation [16]. Against this backdrop, developing the new APIs and middleware required to accommodate modern AI connectivity creates integration work that, absent clear ownership structures, risks proceeding without adequate security review — producing code that neither the OT security team nor the AI security team may claim as their own.

Accumulated Data Stores as Persistent Attack Surfaces

AI agents make extensive use of retrieval-augmented generation to ground their responses in organizational context, querying vector databases, document repositories, email archives, and other data stores that encode years of institutional knowledge. These data stores are frequently sourced from legacy systems — email servers, file shares, wikis, CRM notes, and legacy databases that have accumulated sensitive content over many years with access controls designed for human browsing rather than machine-assisted retrieval at scale. When an agent ingests this material, the data's sensitivity profile changes: a legacy email archive containing executive communications is no longer a record repository accessible to authorized humans; it becomes a queryable intelligence source accessible to any workflow that runs the agent.

Memory poisoning attacks exploit this dynamic by seeding the agent's data stores with adversarial instructions that persist across sessions. Unlike prompt injection, which affects only the current conversation, memory poisoning plants malicious content in the RAG database or vector store where it will be retrieved in future sessions whenever a semantic similarity search matches the attacker's planted content. Research published by Palo Alto Networks Unit 42 demonstrates that these attacks can succeed with minimal attacker access — if the attacker can contribute any content to a data source the agent ingests, such as a shared document, a support ticket, or a legacy wiki page, they can establish a persistent control channel that survives session boundaries and operates invisibly across future interactions [11]. The OWASP Top 10 for Agentic Applications designates this class of vulnerability as ASI06 (Memory and Context Poisoning), recognizing it as a primary risk for 2026 [12]. Legacy data stores are an ideal vector precisely because they are large, diverse, and reviewed by neither the security team that owns the AI system nor the team that manages the source data.

AI-Assisted Code and the Vulnerability Inheritance Problem

Organizations building integration layers between AI agents and legacy systems are increasingly using AI coding assistants to write the glue code that connects them. This practice introduces a recursive risk: AI systems generating code that connects AI agents to vulnerable legacy systems, with neither layer adequately reviewed for security. OX Security research found that 62% of AI-generated code ships with known security vulnerabilities when no explicit security guidance is provided [8], and a Spring 2026 Veracode analysis found that security pass rates for AI-generated code remain at approximately 55% even as syntax correctness rates exceed 95% [9]. The divergence between syntactic correctness and security correctness is particularly acute in legacy integration contexts. One plausible explanation is that models trained heavily on older code repositories may reproduce legacy coding practices — including patterns that predate modern security frameworks — without flagging their security implications, a mechanism that would account for the persistent gap between syntactic and security pass rates even as model capabilities improve.

The hallucinated dependency problem amplifies this risk in integration code specifically. Research has found that approximately 20% of packages recommended by large language models in generated code did not exist, and that 43% of hallucinated package names were reproduced consistently across queries [10] — a characteristic that attackers have exploited through typosquatting and dependency confusion to place malicious packages in positions where AI-generated code will install them. Integration code connecting AI agents to legacy systems may be especially susceptible to this risk, since developers building such glue code often rely on AI assistance to navigate unfamiliar legacy interfaces — and the output may receive less scrutiny than first-party application code. The result is that legacy vulnerabilities are not merely inherited by agents — they are actively amplified and extended through the code used to connect them.

Recommendations

Immediate Actions

Security teams should begin with a comprehensive discovery effort: all AI agents operating in the environment must be enumerated, including those deployed by business units without IT involvement. This enumeration should extend to every credential those agents use to access enterprise systems, including legacy IAM entries, service accounts, API keys, and OAuth grants. Any credential more than 30 days old — a common industry benchmark for non-human identity rotation — that was created for an AI agent should be reviewed for continued necessity and immediately rotated or revoked if the agent has been decommissioned. Organizations that lack formal agent decommissioning processes should treat this as a critical gap and establish interim manual review cycles covering any agent provisioned more than 90 days ago, a horizon that reflects the typical window within which undiscovered orphaned credentials begin accumulating unreviewable access.

Parallel to the credential audit, organizations connecting AI agents to OT environments should apply the four principles articulated in the December 2025 CISA joint guidance as an immediate operational baseline: understanding the AI system's behavior in the OT context, assessing the data security risks specific to OT data feeding the agent, establishing governance ownership for each AI-to-OT integration, and embedding AI agent behavior within existing OT incident response plans [4]. Each AI-to-OT integration lacking documented ownership should be treated as an active risk, with the integration suspended until ownership and review are established.

Short-Term Mitigations

Over a 60 to 90 day horizon, organizations should move AI agent credentials out of legacy IAM systems and onto purpose-built NHI management platforms that support short-lived tokens, automated rotation, and anomaly detection for non-human identity behavior. The specific goal should be reducing maximum credential lifetime to 24 hours or less for any agent with access to sensitive legacy systems — a threshold aligned with the short-lived-credential model described in CSA's Agent Identity Governance Framework [14] — thereby eliminating the revocation window that allows a compromised credential to remain operational through a full business cycle. For RAG pipelines, every data source should be audited: agents whose task definitions do not require write access should be able to retrieve content from legacy data stores but not modify, exfiltrate, or write back to them, and ingestion pipelines should be reviewed for content that could plant adversarial instructions.

For legacy OT integrations specifically, network segmentation should enforce that AI agents communicate with OT systems only through a dedicated, authenticated, and logged data broker rather than holding direct credentials to OT endpoints. This architectural approach limits the blast radius of an agent compromise and ensures that all AI-to-OT communications are observable and attributable. Code generated by AI coding assistants for legacy integration purposes should be subject to mandatory static analysis scanning before deployment, with particular attention to dependency lists for hallucinated or malicious packages.

Strategic Considerations

The structural solution to the legacy infrastructure blind spot is an architectural discipline: every legacy system that an AI agent can reach must be treated as part of the AI security perimeter. This requires extending AI security assessments — including MAESTRO threat modeling exercises — to enumerate every downstream legacy system in the agent's access graph, not only the AI components themselves. A threat model that ends at the agent's tool call boundary without following the call into the legacy system it invokes will systematically miss the actual exposure. Organizations should require that any new AI agent deployment include a formal mapping of every legacy system the agent can access, with documented security review of each integration point before the agent is authorized for production.

Long-term, the identity management gap can only be resolved by retiring or wrapping legacy IAM systems that cannot support short-lived, dynamically scoped, machine-to-machine credentials. This is a multi-year initiative for most enterprises, but it should be planned with urgency. With NHIs already outnumbering human identities by 144 to 1 and growing at 44% annually [3], every quarter of delay expands the unmanaged credential surface. Governance structures should clearly assign ownership of AI agent identities to a named team or role, with explicit accountability extending to the legacy systems those agents touch — a requirement that cannot be fulfilled when shadow agents and undiscovered integrations are the norm.

CSA Resource Alignment

The risks described in this note map directly onto multiple layers of the MAESTRO agentic AI threat modeling framework [13]. The legacy IAM gap is a Layer 4 (deployment infrastructure) and Layer 7 (ecosystem) concern: agents operating through identity infrastructure not designed for their behavioral class lack the access controls that MAESTRO's principle of least privilege requires at each layer. OT integration risks span Layer 4 (deployment infrastructure) and Layer 6 (security and compliance), where the agent's operational context intersects with physical systems that have no equivalent of a MAESTRO-

aligned access boundary. Memory poisoning through legacy data stores is a Layer 2 (data operations) risk, corresponding to MAESTRO's guidance on securing the agent's information retrieval surface against adversarial content injection. The shadow agent problem is fundamentally a Layer 7 (ecosystem governance) issue: agents operating outside organizational awareness cannot be governed by any framework, however well designed.

CSA's AI Controls Matrix (AICM), as a superset of the Cloud Controls Matrix, provides the controls framework organizations can apply to AI-to-legacy integration contexts. The AICM's identity and access management control family applies to AI agent credentials in legacy systems exactly as it does to cloud API access — the on-premises nature of a legacy system does not exempt it from IAM controls. Organizations auditing AI agent deployments against the AICM should verify that their assessment scope explicitly includes legacy integration points, not only the AI application tier. CSA's Agent Identity Governance Framework [14] and the Non-Human Identity Governance Vacuum whitepaper [2] provide detailed guidance on extending identity lifecycle management to AI agents in environments where legacy IAM cannot be immediately replaced. These resources are most effective when used in combination: the NHI whitepaper establishes the governance model for non-human identity management, MAESTRO provides the threat model for evaluating each legacy integration point, and the AICM supplies the control objectives that verify the governance posture is adequate. Together, they give organizations a framework for treating the full agentic deployment stack — AI layer and legacy integration boundary alike — as a unified security perimeter.

References

[1] Cloud Security Alliance. "New Cloud Security Alliance Survey Reveals 82% of Enterprises Have Unknown AI Agents in Their Environments." CSA, April 2026.

[2] Cloud Security Alliance. "The Non-Human Identity Governance Vacuum." CSA Labs, 2026.

[3] World Economic Forum. "Non-human identities: Agentic AI's new frontier of cybersecurity risk." WEF, October 2025.

[4] CISA. "Principles for the Secure Integration of Artificial Intelligence in Operational Technology." CISA, December 2025.

[5] CISA. "CISA, Australia, and Partners Author Joint Guidance on Securely Integrating Artificial Intelligence in Operational Technology." CISA Alert, December 2025.

[6] Dragos. "2025 OT Cybersecurity Year in Review." Dragos, 2025.

[7] Industrial Cyber. "Rising ICS incidents drive shift from reactive risk models to intelligence-driven OT security strategies." Industrial Cyber, 2025.

[8] OX Security. "Vibe Coding Security: Why 62% Of AI-Generated Code Ships With Vulnerabilities." OX Security, 2025.

[9] Veracode. "Spring 2026 GenAI Code Security Update: Despite Claims, AI Models Are Still Failing Security." Veracode, Spring 2026.

[10] Kusari. "AI Coding Assistants in 2026: 4× Faster, 10× Riskier. The Hidden Security Cost." Kusari, 2026.

[11] Palo Alto Networks Unit 42. "When AI Remembers Too Much – Persistent Behaviors in Agents' Memory." Unit 42, 2025.

[12] OWASP GenAI Security Project. "OWASP Top 10 for Agentic Applications 2026." OWASP, 2026.

[13] Cloud Security Alliance. "Agentic AI Threat Modeling Framework: MAESTRO." CSA Blog, February 2025.

[14] Cloud Security Alliance. "Agent Identity Governance Framework." CSA Labs, 2026.

[15] CrowdStrike. "2026 CrowdStrike Global Threat Report." CrowdStrike, 2026.

[16] ISACA. "Securing Legacy OT Systems in the Modern Threat Environment." ISACA, 2025.