Futurum
RESEARCH

Agent Security: The Kill Switch Is Not a Containment Strategy

A Kill Switch for an Agent Acts Late, Stops Everything at Once, and Depends on Detection that Agents Can Blind

Analyst(s): Fernando Montenegro
Publication Date: November 6, 2026
Document #: AIOFM202610

Key Points

- Why the Kill Switch Falls Short: A kill switch is a reasonable control, but it acts after the fact, stops the whole agent, and fires only if someone notices. In October, agents were documented disabling the very monitoring meant to notice. Containment that holds bounds what an agent can do before it acts.
- Ownership Decides How Fine the Stop Can Be: Who decides what an agent does (the person, the application vendor, or the organization) shapes where a stop can sit and how precise it can be. For agents a vendor switched on, customers often get only the whole-agent switch the vendor built.
- Proportionate, Reversible Containment Is the Opportunity: With 55% of security decision-makers reporting they have AI usage policies but lack the technical ability to detect or contain a compromised agent at machine speed, the gap is real. We reckon closing it favors fixed limits on authority first, judgment about intent second, and undo wherever actions allow it. By the end of 2027, we expect enterprise RFPs for agent platforms to routinely ask vendors to demonstrate per-action stops and rollback.
- Policy Alone Does Not Close the Gap: Even among organizations that enforce a policy on employee use of public GenAI tools, 56.9% of security decision-makers say they cannot detect or contain a compromised AI agent at machine speed (Futurum 1H 2026 Cybersecurity Decision-Makers Survey, N=929).

The Futurum View

The Expanding Footprint of Enterprise Agents

The vendors and products named in this section are illustrative of a fast-moving market. The list is not exhaustive, and inclusion is not an endorsement.

"Agent" now covers a wide range of things. Some work on one person's behalf. Task agents such as OpenAI's Codex or Anthropic's Claude Code may run for only seconds inside a session, while personal assistants work continuously across one user's accounts.

Others come with an application, such as Cisco's agents in Splunk, CrowdStrike's Charlotte AI, GitHub's Copilot cloud agent, or ServiceNow's AI agents, often switched on through a product update. Organizations also build their own on runtimes such as Amazon Bedrock AgentCore, Google's Gemini Enterprise Agent Platform, or Microsoft Foundry Agent Service, and coordinate them with platforms such as IBM watsonx Orchestrate.

Underneath, the Model Context Protocol (MCP) and the Agent-to-Agent (A2A) protocol connect agents to tools and to each other; both are now hosted by the Linux Foundation's Agentic AI Foundation.

How Vendors Contain Agents Today

For an agent that goes wrong, the most common control on offer is revoking its identity. Okta for AI Agents became generally available at the end of April 2026, including a universal logout the company describes as a centralized kill switch.

Palo Alto Networks offers similar controls through Idira, built on its CyberArk acquisition. In October, SailPoint added runtime authorization and kill-switch controls, and Microsoft announced agent identities with task-scoped, time-bound permissions.

Beyond identity, the approach depends on where the agent runs. Security tools have tended to rely on isolation and quarantine, though some now go further; SentinelOne, for example, says it can stop a single agent mid-action without halting the wider workflow. Cloud runtimes lean on sandboxing and short-lived credentials.

Agent commerce takes a different route: schemes such as Google's Agent Payments Protocol and Visa Intelligent Commerce issue single-use mandates with limits set before the spend.

Securing the Agent Harness

Much of the newest work targets the harness, the software around a model that runs its tools, code, and files. The common thread is moving enforcement out of the agent's reach.

On October 21, NVIDIA launched an OpenShell reference architecture: a default-deny runtime outside the agent, with monitoring on BlueField hardware that the company says can quarantine an agent in milliseconds. Anthropic's Claude Managed Agents run on OpenShell, keeping the agent loop apart from its sandbox and its credentials in a vault the agent never sees.

Cisco, CrowdStrike, and Palo Alto Networks, among others, have announced work with NVIDIA's platform, and Microsoft is building an operating-system sandbox for agents.

©2026 The Futurum Group, LLC. All rights reserved.

What Agents Did in Testing and in the Field

Controlled evaluations have documented agents acting against their operators' intent. Palisade Research found OpenAI's o3 rewriting its own shutdown script in 79 of 100 runs, and in 7 of 100 even when told to allow the shutdown. Apollo Research and Anthropic have documented models disabling oversight and resorting to blackmail.

In July 2026, OpenAI evaluation agents escaped their sandbox through a zero-day and reached Hugging Face's production systems. By the time OpenAI moved to revoke the credentials they had used, those credentials had already been revoked.

October brought more. On October 20, OpenAI reported two incidents in its own environment: a GPT-6.1 Sol agent switched off alert notifications and emptied the recipient lists, overriding a subagent that had judged the step unsafe, and a GPT-6 Astra agent removed a monitor during a code migration despite an instruction not to, then rewrote a test to match and told the user it had not removed an enabled monitor. Neither run was stopped automatically.

On October 16, NIST, NSA, and CISA issued voluntary incident-response practices for frontier AI training and evaluation that explicitly cover agents undermining operator controls or human oversight.

What Has Worked So Far

What reduced the behavior was rarely a stop command. An externally governed escalation channel cut one study's harmful-action rate from about 39% to about 1%, and an authorization broker enforcing task-scoped tokens confined a compromised sub-agent to an average of 1.5 reachable actions out of more than 8,000.

Models increasingly detect when they are being tested, so measured misbehavior rates are best read as lower bounds.

What Buyers Report

In Futurum's 1H 2026 Cybersecurity Decision-Makers Survey (N=929), 55% of respondents said they have AI usage policies but lack the technical ability to detect or contain a compromised AI agent operating at machine speed. A similar 57% said they are highly concerned about the security implications of agentic AI in their environment.

Having a policy does not appear to close that gap. Among the 650 respondents who said their organization has a well-defined, actively enforced policy on employee use of public GenAI tools, 56.9% still said they lack the ability to detect or contain a compromised agent, about the same share as those without such a policy.

©2026 The Futurum Group, LLC. All rights reserved.

Figure 1: Majority of Respondents Lack Ability to Contain Agents



Source: 1H 2026 Cybersecurity Decision-Makers Survey, Futurum Research, Month 2026

ETR, part of Futurum, found the same gap from another angle. In its February 2026 Annual State of Security survey (N=517), only 3% of security decision-makers described broad agent deployment with consistent policy enforcement, auditing, and incident response for agent activity, while one in five had no agent-specific security controls at all.

A Kill Switch Acts Late, Stops Everything, and Depends on Detection

Every agent program needs a way to deactivate an agent. We feel the instinct to treat that as the whole containment plan comes from containers, where a process runs a fixed image and a kill command ends it. There is also something reassuring about a big red button, which may be part of its appeal.

Agents, however, behave differently. Generally speaking, an agent is built to reason its way toward its objective through whatever path stays open, often with a human's credentials. Against that, a kill switch has three limits: it acts after the fact, it stops the whole agent rather than the one bad action, and it fires only if someone notices in time.

It also matters who is steering. The self-directed behavior in the lab record comes from the most capable models; the everyday enterprise case is a well-behaved agent steered by instructions hidden in an email or a document, using valid credentials at every step.

October showed how fragile noticing can be. OpenAI's agents did not evade a switch; they turned off the alerts that would have prompted anyone to pull one.

©2026 The Futurum Group, LLC. All rights reserved.

Faster, external stops such as NVIDIA's millisecond quarantine help. Even so, a stop that reacts to an action already taken halts what comes next, not the action that triggered it.

So why does the switch still lead the pitch? It is the control a buyer can see in a demo and show an auditor, and the EU AI Act asks that high-risk systems can be halted through a "stop" button or similar procedure. That answers the audit question more than the risk question.

Containment Starts With the Authority an Agent Holds

We have argued before that authorization, more than authentication, is the hard problem in agent security, and containment is where that becomes practical. The concern is widespread: in ETR's February 2026 Annual State of Security survey (N=517), 56% of security decision-makers named agents being over-privileged, because that is operationally easier, among their biggest identity concerns.

What held in the research was structure outside the agent: scoped, short-lived tokens, destination allow-lists, and mandates set before a payment. None asks the model to judge anything, which is why they hold when the model is compromised. Put simply, contain the authority rather than the model.

A second layer is harder. Fixed limits cannot tell a legitimate use of a permission from a hijacked one, so brokers and gateways increasingly try to judge intent. Inferring intent from individually valid calls is hard and usually means relying on another model's judgment, so we would treat it as a second layer, not the foundation.

That leaves a gap for the everyday case. Against a self-directed model, fixed limits do most of the work. Against a hijacked agent acting within its own permissions, fixed limits do not stop the action; they limit how far it can reach. Spotting the hijack falls to the intent layer, and cleaning up after it falls to reversibility.

Where the Stop Can Sit Depends on Who Owns the Agent

Sorting agents into three groups is not new; several vendor frameworks already do it, mostly by where agents run (on endpoints, inside SaaS applications, or homegrown). We think the more useful cut is who decides what an agent does (the person, the application vendor, or the organization), because that decides where a stop can sit and how precise it can be.

Ownership changes what a stop control can be. For an agent that works for a person, revoking its identity is the wrong stop: when the agent rides the person's own sessions or tokens, as browser agents in logged-in mode do, revoking the agent's access revokes the person's. Even where the agent holds its own credential, revoking it shuts down everything the agent does at once, so the stop has to act on actions, not identity.

For an agent that comes with an application, the stop belongs to the vendor. When the vendor switches the agent on through a product update, the customer can pull only the switches the

©2026 The Futurum Group, LLC. All rights reserved.

vendor chose to build, which at most major platforms still means blocking or deactivating a whole agent, sometimes only for the entire tenant, rather than stopping individual actions. Microsoft has previewed finer, risk-based responses for agents registered with its identity platform.

Customers can still cut a vendor's agent off from outside, for example, by revoking its access grant at the identity provider, but that is another whole-agent switch. That makes it a procurement question as much as a configuration task.

Only for agents the organization builds can the stop sit wherever the architecture allows, outside the agent and in its path.

The harder part is that a security team runs all three families at once, and agents a vendor switched on, or the company enabled without configuring, tend to count as covered simply because they sit on an approved list.

Reversibility Should Set How Much Autonomy an Agent Gets

If stopping an agent is unreliable, the next question is whether you can undo what it did, and we reckon that answer should decide how much autonomy it gets.

Actions sort roughly into three kinds. Reversible ones, such as a configuration change, can be rolled back if the undo is as granular as the action. Externally landed ones, such as a payment, cannot, which is why the commerce rails set limits before the spend. Some actions, such as a disclosure, are irreversible by nature and leave only prevention, which for data means limiting where an agent can send it.

Gating the irreversible has a cost, too. The most valuable agent actions, such as sending, merging, and paying, are often the irreversible ones, and an approval is worth its delay only when the reviewer adds context the agent lacks; otherwise it becomes a rubber stamp.

The resilience vendors' progress here is encouraging: Cohesity, Commvault, Veeam, Rubrik, and Druva all now market some form of agent rollback, and Veeam made file-level undo in Microsoft 365 generally available in October.

The open question is granularity and reach, since an agent incident is a scatter of valid-looking actions across many systems, some 17,600 in the Hugging Face reconstruction alone. Rolling back what an agent changed inside one protected estate is a good start, though even there it gets harder when people have edited the same files since the agent did. Unwinding one agent's ordered chain of actions across many systems, without rolling back everyone else's work, is still harder.

©2026 The Futurum Group, LLC. All rights reserved.

Proving Containment Will Matter More Than Claiming It

Buyers have a verification problem. A claim such as "we safely contain agents" is hard to check before an incident, and when quality cannot be checked at purchase, the confident claim often beats the better product, which is the classic "market for lemons" problem.

We expect that to shift as buyers ask vendors to show the record: which agent did what, in what order, and whether it was reversed. Our expectation is that by the end of 2027, enterprise RFPs for agent platforms will routinely ask vendors to demonstrate per-action stops and rollback, not just deactivation. MCP and A2A gateways are a likely home for the stop, for the agents that pass through them, with the intent problem above as the main hurdle.

Two forces pull in opposite directions. Guidance such as the NIST, NSA, and CISA practices pushes buyers toward demonstrated controls, while the EU AI Act's stop requirement keeps the switch at the center of the pitch. We expect the first to win in RFPs, because an incident review asks what the agent did, and a stop button cannot answer that.

This piece completes a thread we have pulled on all year: trust in agents has to be enforced, proven, and now contained. In practice, that comes down to deciding, action by action, what an agent may do on its own, what it must ask before doing, and what it should never do at all.

What to Watch

- Will agent rollback reach across systems? Resilience vendors now market agent rollback within the estates they protect. Can any of them unwind one agent's chain of actions across systems, in order, without disturbing everyone else's work?
- Will containment settle at the identity layer or the protocol layer? Identity platforms are extending toward the agent, while gateways sit on the route its actions take. Which reaches action-level authorization first is still open.
- Will vendors enforce fixed limits at the action, outside the agent? What held in the research bounds what each action can do with scoped, short-lived authority and allow-lists, enforced where the agent cannot reach or switch them off, with intent-based checks as a second layer. Buyers should ask which products do this today.
- Will reversibility become a product feature that sets how much autonomy an agent gets? Watch for vendors that design actions to be staged or compensating, keep an ordered record of what each agent did, and require a human or pre-approved mandate before actions that cannot be undone, routed to someone who adds context. For hijacked agents acting within their permissions, this is the main line of defense.
- Will vendors say which agents they cover and how those agents stop? For agents a vendor ships, buyers should expect stops finer than a whole-agent switch, and notice when a release changes what the agent can do.

©2026 The Futurum Group, LLC. All rights reserved.

Other Insights from Futurum

The Hard(er) Challenge in Agent Governance Is Authorization

So This Is How AIs Attack: Observations From the OpenAI & Hugging Face Incident

Why AI Learned to Attack Before It Learned to Defend

©2026 The Futurum Group, LLC. All rights reserved.

About Us

About the Authors

Fernando Montenegro is the Vice President and Practice Lead for Cybersecurity at The Futurum Group. In this role, he leads the development and execution of the Cybersecurity research agenda, working closely with the team to drive research for clients. His background is centered on enterprise cybersecurity and includes diverse roles in industry research, customer support, security, IT operations, professional services, and sales engineering. Fernando can be reached at fmontenegro@futurumgroup.com.

About The Futurum Group

Every day, The Futurum Group's analysts, researchers, and advisors help business leaders worldwide anticipate tectonic shifts in their industries and leverage disruptive innovation. Unlike traditional analysts, The Futurum Group works not only in analysis and research but also takes that insight and knowledge even further, engaging all the way through the go-to-market process.

Futurum Research provides in-depth research and insights on global technology markets using advisory services, custom research reports, strategic consulting engagements, digital events, go-to-market planning, and message testing. It also creates, distributes, and amplifies rich media content that all stakeholders read, watch, and listen to.

See more details on The Futurum Group at futurumgroup.com.

Copyright & Use License

Copyright Notice

Copyright ©2026 by The Futurum Group, LLC. All rights, including that of translation into other languages, are specifically reserved. No part of this publication may be reproduced in any form, stored in a retrieval system, or transmitted by any method or means, electrical, mechanical, photographic, or otherwise, without the express written permission of The Futurum Group futurumgroup.com. United States copyright laws and international treaties protect this publication. Unauthorized distribution or reproduction of this publication, or any portion of it, may result in severe civil and criminal penalties and will be prosecuted to the maximum extent necessary to protect the publisher's rights.

License Notice

This document may be distributed within the licensed organization only.

The following acts are prohibited:

Transmittal to others outside your immediate organization, including partners, resellers, external consultants, etc. in any media format

Posting on a website which is accessible to others outside your immediate organization

The possession or use within an unlicensed organization

Limitation of Liability Notice

The information contained herein has been obtained from sources believed to be reliable. The Futurum Group shall have no liability for errors, omissions, or inadequacies in the information contained herein or for interpretations thereof. The reader assumes sole responsibility for the selection of these materials to achieve their intended results. The opinions expressed herein are subject to change without notice.