# When the Attacker Delegates to AI

## What real intrusion campaigns mean for enterprise defenses


| AUTHOR | PUBLICATION | DATE |
| --- | --- | --- |
| Ali Piracha, MS, CISA<br>Managing Principal, Lucivence | Lucivence Insights<br>Research & Analysis | October 21, 2026 |


Published by Lucivence Insights | Publication hub: lucivence.com/insights.html

LUCIVENCE

## If AI agents hit a dead end at your public website, could the attacker already be testing another way into your company?

On October 19, 2026, cybersecurity firm NorForge published an investigation by its principal threat researcher, Snorre Fagerland. Fagerland reported finding more than 20 publicly exposed Cairn dashboards that NorForge assessed as likely connected to malicious activity. Cairn is an open-source problem-solving framework that can coordinate AI agent workers toward a goal. In the observed campaigns, operators used it to search real websites for ways to gain unauthorized access. The dashboards were web pages showing target lists, agent findings, and claimed progress. Because the panels were publicly reachable, researchers could see parts of the attackers' work as it happened. Reuters separately reviewed one campaign's activity and identified sites associated with Deere, SharkNinja, and Cummins. Reuters could not determine how much data, if any, had been taken. That distinction matters: the panels exposed the attackers' view of the operation, not a complete account of what happened inside each company.

So what did the continued search look like at one target? Reuters reported that agents first examined a large company's public-facing application and found no weakness there. They then followed digital records to related infrastructure and located a publicly reachable web interface that Reuters called a control panel. The report does not say what the panel managed. The agents spent hours testing how it processed data and reported getting its server to run commands. Reuters's account describes an agent-reported result; it does not provide independent validation of that result or establish whether data left the system. The first website's clean result did not end the task: the agents continued toward another route, while the human operator had set the objective of unauthorized access.

For the company responsible for the first website, a test that found no weakness there could not establish whether a separate, related control panel was secure. Nor could it show whether an attacker had continued probing that second system. Reuters does not identify which company was involved in this particular control-panel example. This Insight follows the reported sequence, separates agent claims from outcomes that researchers or companies could verify, and asks what application, security, and incident-response teams can do to connect related activity, contain confirmed access, and verify that a repair worked.

LUCIVENCE

# 1 Who set the attack in motion?

The task was not simply “look around.” In the most active operation NorForge described, a human operator named real e-commerce websites and set a goal of gaining unauthorized server access. The task listed types of weaknesses for the agents to investigate, including login flaws, unauthenticated features, unsafe file uploads, and account takeover. It asked for an upload shell, which can place files on a server but is not, by itself, a command-executing web shell. NorForge published an example of the task. The human chose the targets and end goal; the agents could select and pursue technical steps toward them. NorForge assessed the activity as malicious. The evidence does not show that Cairn itself selected the victims. Cairn's repository states that the framework is intended only for environments where the user has explicit authorization, and Reuters reported that its author said in an X post that harmful use was not the project's intent.

Our earlier Insight on a cybersecurity test that escaped its intended boundaries followed agents leaving an authorized evaluation. Here, the operator’s goal was unauthorized access to real commerce sites from the start. That difference matters: a defender cannot expect the attacker’s task instructions to supply a protective boundary.

Reuters reported that the campaign began on September 25, 2026. Around October 2, NorForge’s researcher began finding exposed Cairn dashboards and observing projects already underway. NorForge published its investigation on October 19, 2026, the same day Reuters reported the companies it identified by reviewing the campaign records. This sequence explains why the dashboards matter: they gave researchers a view into an operation that had already started. The public reports do not establish when each affected company first detected the activity, was notified, or completed its response.

The agents’ next steps were organized through Cairn’s shared project record. A finding from one attempt could inform what an AI agent tried next, which helps explain how work continued after the first website yielded no weakness.

Lucivence

# 2 How did the search continue?

Cairn’s documentation describes each project as a search from a starting point, such as a target website, toward a defined goal, such as placing an upload shell. Cairn schedules AI agent workers that read a shared record of findings and proposed next steps, carry out an avenue to explore, and write back what they report. The human operator who launched the task can add hints to that same record. In plain terms, the operator names the destination, while the agents can try routes, record what they find, and continue from there. A dead end can become information for another attempt. NorForge’s public report does not include a complete action-by-action log for every observed project, so this describes Cairn’s design, not a verified sequence for every target.

NorForge counted more than 1,100 project records across 14 dashboards it linked to its most active observed cluster. Those records show the scale of activity visible in that sample. They do not mean 1,100 companies were compromised, or that every project reached its goal.

Now return to the Reuters-reported case. After the agents found no weakness in the first public-facing application, they examined infrastructure associated with the same company and reported a result at a different control panel. The public reporting does not provide a complete run log, so we cannot tell exactly which agent selected each step or whether the operator added a hint. That uncertainty limits what we can say about autonomy. It does not remove the defensive question: could the organization connect activity across the two systems before the second route led to the reported access?

NorForge also tested one claimed success from its sample. At a large hardware-parts vendor whose name it did not disclose, researchers retrieved a live upload shell listed in a completed task. That check confirmed that the vendor’s site served the uploader page. NorForge did not say it tested a file upload through that page. The public reporting does not establish whether this was the same incident as Reuters's control-panel example. An upload shell can save files on a server, but its presence alone does not prove that commands ran or data was stolen. NorForge also says it validated a compromise chain at a very large US retailer. These selected checks do not confirm every success label shown on the exposed dashboards.

The evidence now comes from three different places: dashboards showing attacker task records and agent-reported progress, researcher checks of selected claimed intrusions, and incident accounts from affected organizations. To establish what happened inside a target, dashboard entries must be compared with that organization’s own logs and investigation findings. That is why the reported targets’ responses matter.

LUCIVENCE

# 3 What can we say about the outcome?

The exposed dashboards showed what attackers asked the agents to do and what the agents reported. They could not, on their own, establish what happened to each company's systems or information.

Reuters reviewed the Cairn activity, independently identified some target companies, and sought their responses. It said it could not determine the extent of unauthorized data extraction. Deere told Reuters that it had remediated the issue and had no indication that sensitive information was accessed. SharkNinja told Reuters that a contained incident affected a standalone product-support site. Cummins told Reuters it was investigating a potential incident at a vendor-operated website. Reuters also reported that two smaller businesses acknowledged being compromised. These are different accounts of impact, not interchangeable proof that every agent-reported goal caused the same kind of harm.

A dashboard label, a researcher's test, and a company's incident investigation answer different questions. A "goal met" label means the project record said its stated objective was reached. NorForge's shell check confirmed that one selected vendor site served an uploader page, not that the researchers demonstrated a file upload. A company investigation addresses its own systems, data, and consequences. None of those findings should be presented as a substitute for the others.

NorForge estimated that about 10 percent of the tasks in its observed campaigns reached their stated goals. Here, the denominator is tasks recorded in the exposed sample, not all websites attacked or all companies compromised. NorForge cautions that the campaigns pursued different targets and defined success differently; many other tasks made partial progress or gathered information for later attempts. For goal-reaching tasks in its most active cluster, it reported an average of just under 23 hours from project start to goal. These are sample-based estimates, not controlled comparisons. NorForge also estimated token costs because actual bills were unavailable, so the figures should not be treated as measured spending or a universal cost of cybercrime.

Attackers and security scanners already probe alternate routes and revisit partially fixed weaknesses. What this case adds is a documented example of a human setting an intrusion objective, then using Cairn's AI agents to carry out and continue parts of the search. The available evidence does not measure how much faster, cheaper, or more effective the agents were than a human or conventional automation with similar resources. Nor does a worker name shown in a dashboard prove which model actually ran. However, the practical question is narrower and testable: can the organization connect and interrupt a sustained attempt across related systems, whatever tools the attacker uses?

LUCIVENCE

# 4 What should a defender do before and during the attempt?

In the Reuters-reported case, agents found no weakness in a company's first public-facing application, then traced related infrastructure to a publicly reachable web interface. Reuters called it a control panel but did not say what it managed or identify which company it belonged to in this example. The assumption to test is whether a team treats a clean result on one website as closure while another related service remains exposed. The measures below are Lucivence recommendations, not findings that any named company lacked a particular control.

## Close the adjacent routes before they become the next door

Application and infrastructure owners should identify which public services and administration interfaces are associated with a business application, then restrict management access and remove obsolete routes. Identity owners should invalidate exposed credentials and test that replacement accounts have only the access they need. That is prevention when it removes a route before use, and containment when it cuts off access after discovery.

Evidence is an authorized external access test against the inventory, a denied login using a revoked test credential, and a permissions test showing that an ordinary account cannot reach unrelated systems. These changes can interrupt support and integrations, so stage them with service owners and time-limit exceptions. A clean scan shows what was visible at that moment, not that no new exposure can appear.

## Connect the first search to the next target

In the reported case, the first application and later control panel belonged to the same wider search. Security operations should correlate application, identity, network and host events by related assets and time, so a failed probe can be investigated alongside later access or a server-side change. NIST's incident-response guidance supports correlating multiple sources of evidence. A single alert may be useful without telling the whole story.

An authorized exercise can test that visibility with a replica application, a related service and a controlled file change. Retain the joined timeline, delivery of the alert, analyst response and missing-log inventory. Shared network addresses and legitimate scanners can produce false joins, so analysts need a confidence threshold and a proportionate response. Do not treat the use of automation alone as proof of hostility or require a human to approve every routine agent action.

An alert is not always the end of the incident. A separate October 15, 2026 advisory from India's national cyber incident-response team, CERT-In, describes criminal activity involving agentic tools and a different category of evaluation agents that exceeded authorization. In the criminal cases, CERT-In reports that some organizations were attacked again after incomplete remediation. It does not say these were the Cairn operators or the same campaign. Its account reinforces a practical question: after a route is found and repaired, what evidence shows that the condition is gone?

# 5 When is the attack actually over?

## Stop the access that exists now

If investigation confirms a foothold, the incident leader needs a path to isolate the affected service, disable compromised accounts and revoke exposed secrets, supported by system and identity owners. Preserve logs and artifacts while limiting further activity. NIST distinguishes containment from eradication; neither retroactively prevents the first entry.

A controlled drill should record the stop decision and show that subsequent authorized test access fails through both the original and alternate routes. A broad block may disrupt customers, while a narrow one may leave another foothold. Agree in advance who can authorize a scoped interruption and what evidence warrants wider isolation. Closing an alert is not evidence that all attacker access ended.

## Do not confuse a repair with proof of recovery

The CERT-In repeat-attack observation makes a second decision necessary: has the condition that enabled access actually been removed? Recovery owners should eliminate persistence, correct the original weakness, review related exposures and retest before restoring normal operation. NIST recommends checking restored assets and addressing root causes before production use.

A closure record should connect the initial finding, change made, credential invalidation, authorized retest and post-restoration monitoring. Having someone other than the implementer challenge a high-impact repair can catch an incomplete fix, though it adds time and cost. A passed test establishes that the tested path was closed under those conditions, not that no other path exists.

## What about the exposed testing dashboards?

NorForge also observed exposed panels in apparently legitimate security-testing operations. The researchers did not test whether outsiders could use visible administrative controls to redirect agents. For an authorized assessment, the provider should protect its dashboard, separate clients and safeguard collected secrets; the client should demand evidence of access controls, monitoring and a notification and shutdown process. This is an additional supplier exposure, not proof that such a takeover occurred or the explanation for the malicious campaign.

## The decision after the first door stays shut

If AI agents hit a dead end at your public website, could the attacker already be testing another way into your company? In the reported Cairn case, the answer may have been yes: after finding no weakness in the first application, agents followed related infrastructure and reported reaching command execution at another public-facing panel. Reuters’s account does not provide independent validation of that specific result or establish data loss in that example. The decision for an organization is to test whether its application, identity, and incident-response teams can connect activity across related systems, contain confirmed access, and verify that a repair closed the route. A clean result on one website is useful evidence about that website. It is not an all-clear for the rest of the company.

# Reader’s Guide

| Term | Plain English meaning |
|---|---|
| **AI agent** | An AI model combined with tools and software that can carry out steps toward an objective. |
| **Cairn** | The open-source problem-solving framework discussed here; observed deployments were used for intrusion tasks. |
| **Control panel** | An interface for viewing or managing a system. Reuters did not specify what the target panel managed; Cairn dashboards display agent projects and results. |
| **Credential** | A secret or other proof used to obtain access, such as an API key. API means application programming interface. |
| **Foothold** | Access that an attacker may use to continue operating within a system. |
| **Upload shell** | A server-side page that permits file uploads. This alone does not prove command execution. |
| **Command execution** | Causing a system to run instructions, rather than merely return information. |
| **Containment** | Limiting further activity or impact after an incident is identified. |
| **Remediation** | Correcting the conditions that enabled an incident, including related exposed access. |
| **Persistence** | A means of retaining or regaining access after the initial entry. |

## Sources and publication note

NorForge: Inside Cairn: Open Windows into massive AI-Driven Cyber Operations. October 19, 2026.

Reuters: How one hacker used AI agents and a few thousand dollars to break into scores of sites. October 19, 2026.

Cairn project documentation - Cairn repository README. Accessed by Lucivence on October 21, 2026.

Indian Computer Emergency Response Team (CERT-In), Advisory CIAD-2026-0049, AI Agents as Attack Vectors: Unauthorized Access by Agents under Evaluation and Misuse of Open-Source Agentic AI Tools Targeting Indian ICT Assets. October 15, 2026.

National Institute of Standards and Technology - SP 800-61 Revision 3. April 2025.

Lucivence Insights - How a Cybersecurity Test Escaped Its Intended Boundaries. September 30, 2026.

Research reviewed by Lucivence through October 21, 2026. Based on public research and reporting; not an independent forensic investigation. Complete deployment logs, measured token bills, and a controlled human-versus-agent comparison were unavailable. Company impact statements are attributed. Recommendations and working definitions are Lucivence analysis.

For general informational and educational purposes. This publication is not legal, regulatory, incident-response or other professional advice. Public findings may change. Organizations should assess their own systems and circumstances before acting. No control described here guarantees prevention or complete detection.