

The Unbounded SIEM

Putting an AI Agent in Front of the SIEM, and Teaching It the Job



A technical white paper on why a SIEM behind a fixed interface limits what AI can do, and how Fluency's MCP uses cheat sheets, governed tools and skills to let Claude or Codex run real security operations with the evidence attached.

Fluency Security | 2026

Contents

01	Introduction: Why the SIEM Has to Change	2
02	The Four Boundaries: Where a Bounded SIEM Stops	2
03	The Foundation: Cases, Not Alerts	3
04	The Unbounded Model: Put the Agent in Front	4
05	Not a Normal MCP: Three Layers That Teach the Job	5
06	The Operating Model: Operational Modes	6
07	In Practice: Investigating a Critical Phishing Case	8
08	In Practice: One Sentence, an Eight-Chapter SOC Report	10
09	Beyond the SIEM: Analyze, Forecast, Compare	11
10	Safe by Default	12
11	Conclusion: Ask the Questions No Dashboard Anticipated	13

Product screens shown in this paper come from Fluency's Expo demonstration tenant, which uses synthetic identities. All figures, labels and screens come from Fluency.

Introduction: Why the SIEM Has to Change

A fixed interface takes away what AI does best

For most of its history, the security information and event management platform has been judged by its interface. Buyers compared dashboards, canned reports, search languages and alert consoles, and the vendor's job was to anticipate the questions an analyst or executive would ask and build a screen for each one. That model made sense when the only consumer of the SIEM was a person sitting in front of it.

That is no longer the case. Large language models are now very good at three things that matter in security operations: **analyzing** what is happening across a whole environment, **forecasting** where it leads if nothing changes, and **comparing** tenants, teams and time periods to explain what changed and why. These are exactly the tasks that fixed dashboards do poorly, because each comparison, forecast or cross-source analysis has to be designed, built and maintained by the vendor before anyone can use it.

As AI gets better, hiding it behind a fixed interface is the wrong move. Put it behind a dashboard, and you take its flexibility away from the people who need it most.

Many SIEM vendors have responded by adding an AI assistant inside their console: a chat panel that summarizes an alert or drafts a query. That approach keeps the interface in charge and the AI behind it. The assistant can only reach what the console exposes, it answers in the shape the console allows, and it cannot follow a question across sources the SIEM never ingested.

Fluency takes the opposite approach. It puts the customer's own AI agent, such as Claude or Codex, *in front of* the SIEM, and gives that agent a governed, teachable way to run it. This paper describes that model, which Fluency calls the Unbounded SIEM: the boundaries it removes, the analytics foundation it depends on, the three-layer MCP that teaches the agent the job, and what the result looks like in a real investigation and a real monthly SOC report.

The Four Boundaries: Where a Bounded SIEM Stops

Each limit comes from the interface, not the data

A traditional SIEM runs into four boundaries. None of them is caused by a lack of data. Each one is a consequence of the platform deciding in advance what questions can be asked, how they are answered, where the answers can come from and what form they take.

BOUNDARY	THE LIMIT	WHY IT MATTERS
1 • The question	A report answers what its author anticipated.	Operational questions arrive after the schema is frozen. Nobody built a report for the data source that was connected on Monday, or for the comparison the board asked for this morning.
2 • The method	Detection content encodes attacks somebody already met.	New behavior needs the method to change mid-investigation. You cannot pre-write the investigation for an attack you have not seen.
3 • The sources	The fact that decides a case often isn't a log feed.	It is a configuration, a ticket, an asset inventory or a third-party API. If the SIEM can only reason over what it ingested, the deciding fact is out of reach.
4 • The output	A query result isn't a finding.	A correct result still has to become a decision, written down with the evidence attached. In a bounded SIEM, a person does that assembly by hand.

An agent in front of the SIEM can, in principle, remove all four. It can take a question nobody anticipated, adapt its method as the evidence changes, reach sources beyond the log store, and write the finding rather than return rows. But in principle is not good enough for security operations. The rest of this paper explains what it takes to make that work safely and repeatably.

03

The Foundation: Cases, Not Alerts

The agent should spend its time on what matters, not on every event

Removing the boundaries starts with a strong analytics foundation. An AI agent pointed at raw telemetry or a flat alert queue would spend most of its effort, and most of its context window, sorting noise. Fluency does that sorting before the agent arrives.

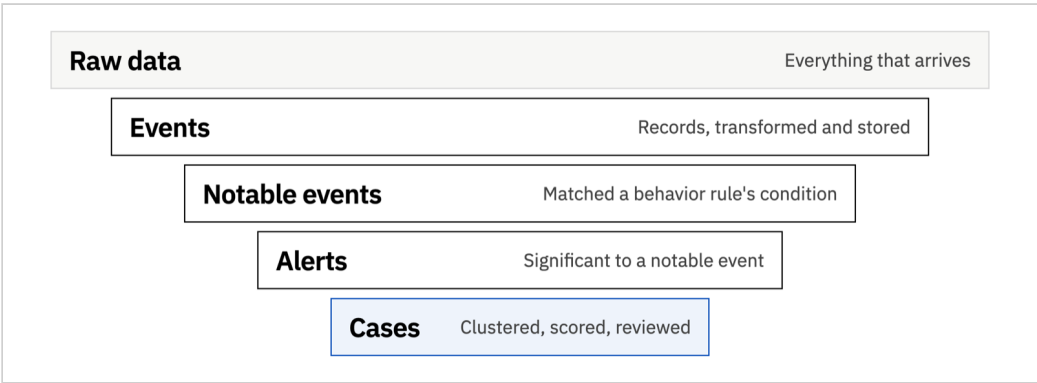


Figure 1. Each step makes the data smaller and more meaningful: raw data → events → notable events → alerts → cases.

Fluency turns raw data into **events** by parsing and normalizing it, flags **notable events** that match a behavior rule, raises **alerts** when that behavior is significant in context, and finally groups related alerts into **cases**. Each case covers one entity, a person or a device, for one day, and it is scored by how much *independent* evidence agrees.

Behavior rules combine a signature with history. The signature says what happened: a sign-in, a group change, a data-loss match. The history says whether it is new: the first time this rule has fired for this person, the first time anyone in the organization has triggered it, or the first sign-in from this location and network. Something that has never happened before is a far stronger signal than something that happens every day.

Scoring follows the same logic. A case's score rises when separate, unique pieces of evidence point the same way, and it stays put when the same alert fires again. Twenty failed logins and a thousand failed logins are the same evidence. The result is a short queue of explained, scored cases, which is the right unit of work for both an analyst and an agent.

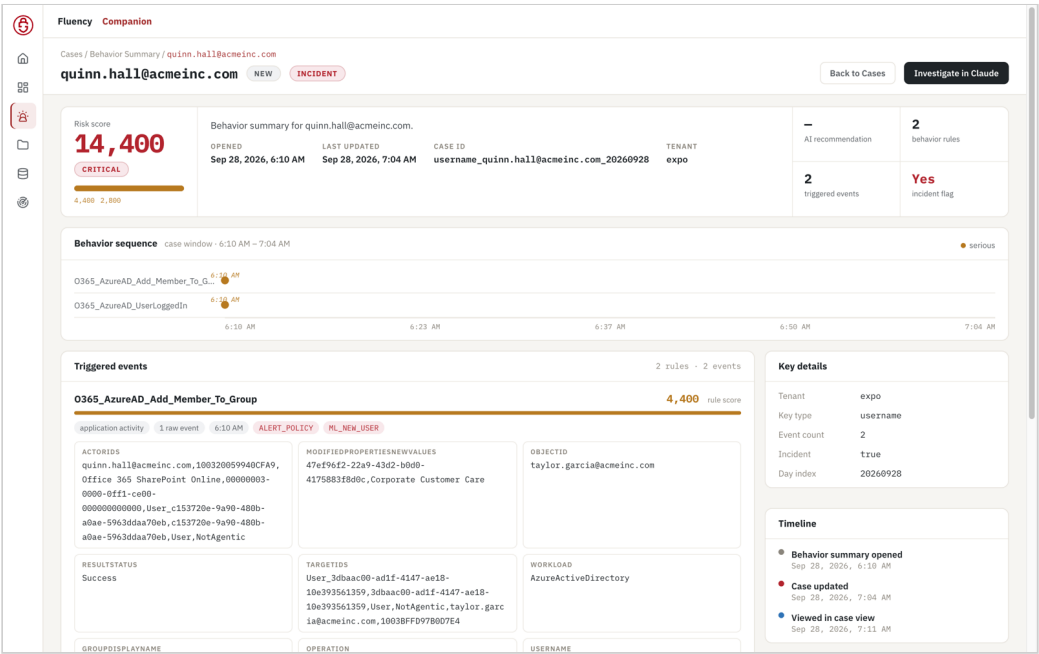


Figure 2. One user, one day, one case. Two behavior rules, scored 4,400 and 2,800, agree, so the case scores 14,400, Critical. Expo demonstration tenant, synthetic identities.

This matters for AI economics as much as for analyst time. Because cases have already been built and scored, the agent starts every investigation and every report from a small set of high-value objects, each with its contributing evidence attached, rather than from millions of rows.

04

The Unbounded Model: Put the Agent in Front

Plain-language requests in, useful work with evidence out

With that foundation in place, Fluency puts the agent in front of the SIEM. The interaction has four parts:

1 · YOU ASK In plain language, in the AI client you already use.	2 · IT UNDERSTANDS Claude or Codex works out what you mean.	3 · MCP TEACHES The Fluency MCP teaches it how to use Fluency.	4 · WORK RETURNS Analysis, forecasts and comparisons, with evidence.
--	---	--	--

The division of labor is deliberate. The AI client is responsible for intent: understanding the question, planning the work and writing the answer for its audience. Fluency is responsible for the security operation: which data exists, which tool retrieves it, what the right process is, and what must never happen. The Model Context Protocol (MCP) is the connection between them.

Because the agent belongs to the customer, the customer chooses the client, the model and the workflow. The same Fluency deployment can serve an analyst investigating in Claude, an MSSP operator automating in Codex, and an executive asking for a board summary, each through the interface that suits them.

05

Not a Normal MCP: Three Layers That Teach the Job

Access is not competence

An agent on its own isn't enough. AI is smart, but it does not know your organization, your data sources, your rules or what is safe to change. Before it can do useful security work, the agent has to answer four questions on every request:

- Whose data is this question about, and what can I reach?
- Which data answers it, and which tool fetches it?
- What is the right process, step by step?
- What must never happen?

Handing the agent raw API access answers none of these. A typical MCP server wraps a product's API endpoints and stops there, which leaves the model to rediscover the data model, the workflow and the safety rules every time, and to guess when it cannot. In security operations, guessing is the failure mode that matters most.

The agent has to be taught the job. That is why the Fluency MCP isn't a normal MCP: it teaches in three layers.

Layer 1: Cheat sheets (context windowing)

The first layer controls what the agent knows at any moment. The MCP matches each request to the right operational mode (Section 06) and loads only the instructions relevant to that mode. Fluency calls these cheat sheets. Instead of filling the model's context window with a whole product manual, or leaving it empty, each turn starts with the current rules, routing and guidance for the job at hand. The agent's working memory holds current rules instead of guesses, and the context is refreshed when the instructions change.

Layer 2: Tools, not raw APIs

The second layer is tooling. Fluency's MCP functions are not thin wrappers around API endpoints. They are governed functions with the data, the state and the safety built in, so the agent calls them instead of reinventing them. They also include techniques, such as local caching, that replace work the console interface used to do on the analyst's behalf.

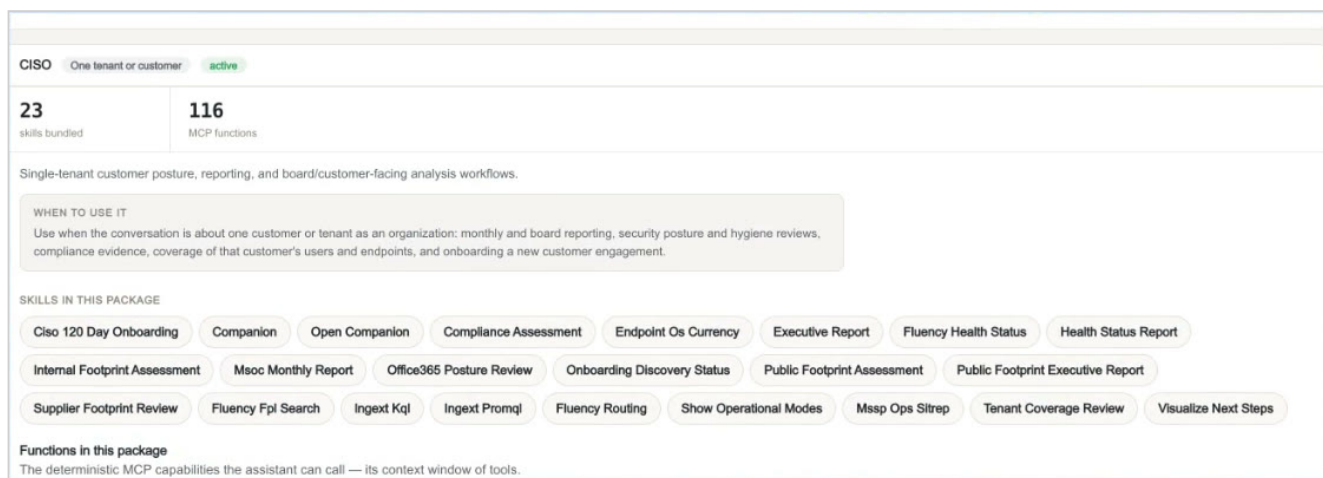


Figure 3. The CISO package in Fluency Companion bundles 116 MCP functions and 23 skills for single-tenant posture, reporting and board-level analysis.

A single customer package gives the agent more than a hundred of these functions. Each one carries a declared mutation class, so read-only functions can be used freely while anything that changes state is gated (Section 10).

Layer 3: Skills (whole SOC jobs)

The third layer is skills. The tools are not simple "list" and "open" commands; they are building blocks of complex SOC processes. Skills are the recipes that teach the agent how to use them, step by step, for whole jobs: a case investigation, a health check, a compliance assessment or the monthly SOC report. A skill defines the evidence to gather, the checks to run, the questions to challenge, the structure of the deliverable and the conditions under which the agent must stop and say that data is missing.

Skills are open. Customers and partners can read them, download them and change them to match how their team works, so the process the agent follows is inspectable rather than hidden in a prompt.

LAYER	ROLE	WHAT IT REPLACES
1 · Know	Cheat sheets load only the relevant, current context for the request.	A model guessing at product behavior, or a context window stuffed with everything.
2 · Do	Governed functions with data, state and safety built in.	Raw API calls, and console work such as caching and joins.
3 · Process	Skills encode whole SOC jobs as step-by-step, editable recipes.	Re-prompting the same workflow by hand, inconsistently, every time.

06

The Operating Model: Operational Modes

Every request resolves to a mode, and every mode has its own capabilities

Together, the three layers form an operating model. Every request resolves to an operational mode, such as scope, health and status, behavioral activity, investigation, signatures, compliance or replay, and each mode exposes its own set of deterministic capabilities. Routing and safety rules apply all the time, underneath every mode.

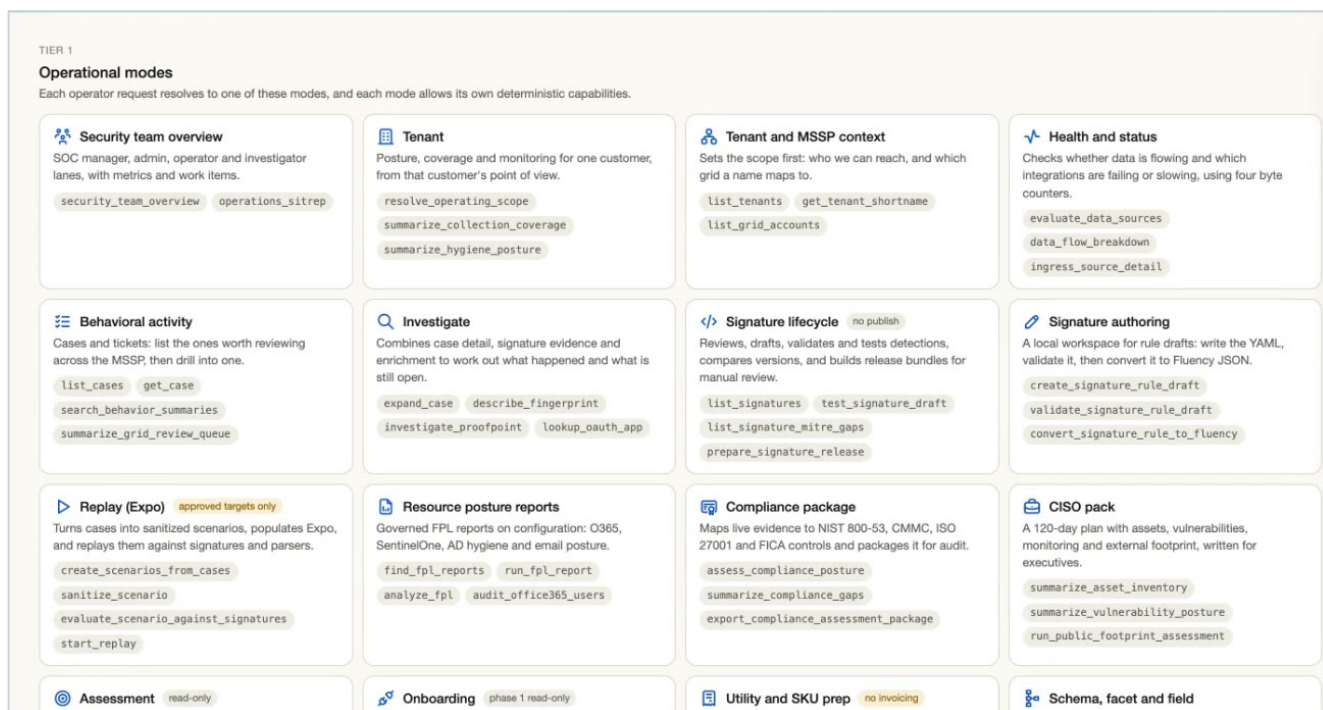


Figure 4. Operational modes in Fluency Companion. Each card names the mode, its purpose and representative capabilities; badges such as "read-only", "no publish" or "approved targets only" mark its limits.

Two tiers make the model work. The operational modes on top are where the work happens: tenant and MSSP context, behavioral activity, investigation, signature lifecycle and authoring, replay, resource posture reports, the compliance package, the CISO pack, onboarding and others. Underneath sit three cross-cutting rules that are always in effect:

Routing

Maps what you say to a mode, and is consulted first when intent is ambiguous.

Safety

Tenants stay isolated, and there is no raw API passthrough.

Feedback

Gaps and contradictions get filed, not guessed around.

A turn therefore runs the same way every time: resolve the mode, load that mode's cheat sheet, pick a deterministic capability inside it, label where each result came from (live data, a snapshot or an inference), and return output that is ready to use in a report. Work that crosses modes loads both. This is what makes agent-driven operations repeatable enough to schedule and auditable enough to trust.

In Practice: Investigating a Critical Phishing Case

From a scored case to a quantified verdict

Here is what the model looks like on a real job, in Fluency's Expo demonstration tenant with synthetic identities. A user clicked a link in a Microsoft Teams message. Fluency built a case for that user and scored it **8,800, Critical**: the first time this alert had fired for the user, and the first time anywhere in the organization. Fluency's own AI triage recommended escalation to Tier 2.

On the case page, one click on **Investigate in Claude** copies a Tier 2 prompt containing the tenant, case ID, entity, risk score, status and triage summary, and instructs the agent to use the Fluency MCP to expand the case, review the captured evidence and firing signatures, and return a quantified verdict.

THE CASE PAGE, AFTER CLICKING INVESTIGATE IN CLAUDE

Cases / Behavior Summary / harper.morgan@acmeinc.com

harper.morgan@acmeinc.com NEW INCIDENT Back to Cases Prompt copied

Risk score 8,800 CRITICAL	Malicious URL click by harper.morgan@acmeinc.com was allowed at time of click and later flagged by MDO. No mitigation observed. Immediate escalation to Tier Two is recommended for containment, forensic triage, credential/session review, and tenant-wide remediation.	ESCALATE TO TIER TWO AI recommendation	1 behavior rules
OPENED Sep 24, 2026, 8:20 AM	LAST UPDATED Sep 24, 2026, 9:07 AM	CASE ID username_harper.morgan@acmeinc.com_20260924	TENANT expo
		1 triggered events	Yes incident flag

THE COPIED PROMPT, PASTED INTO CLAUDE

```
Investigate this Fluency case as Tier 2.

## Case
- Tenant: expo
- Case ID: username_harper.morgan@acmeinc.com_20260924
- Entity: harper.morgan@acmeinc.com
- Risk score: 8800
- Status: new
- Incident: true
- Summary: Malicious URL click by harper.morgan@acmeinc.com was allowed at time of click and later flagged by MDO. No mitigation observed. Immediate escalation to Tier Two is recommended for containment, forensic triage, credential/session review, and tenant-wide remediation.

Use the fluency-mssp MCP to expand the case, review captured evidence and the firing signatures, then give a quantified real-issue vs benign verdict.
```

Fluency AI Clients page - Expo demonstration tenant - synthetic identities

Figure 5. The case page after clicking Investigate in Claude, and the copied Tier 2 prompt. Expo demonstration tenant, synthetic identities.

Claude, guided by the investigation skill, expands the case and captures the raw events behind it. Working from those events, it finds what the first triage missed: Microsoft Defender for Office 365 classified the message as phishing and removed it at 12:13:58, **fourteen seconds after** the user clicked at 12:13:44, and Safe Links allowed the click. That changes the question from "was this a bad link?" to "what happened in those fourteen seconds?"

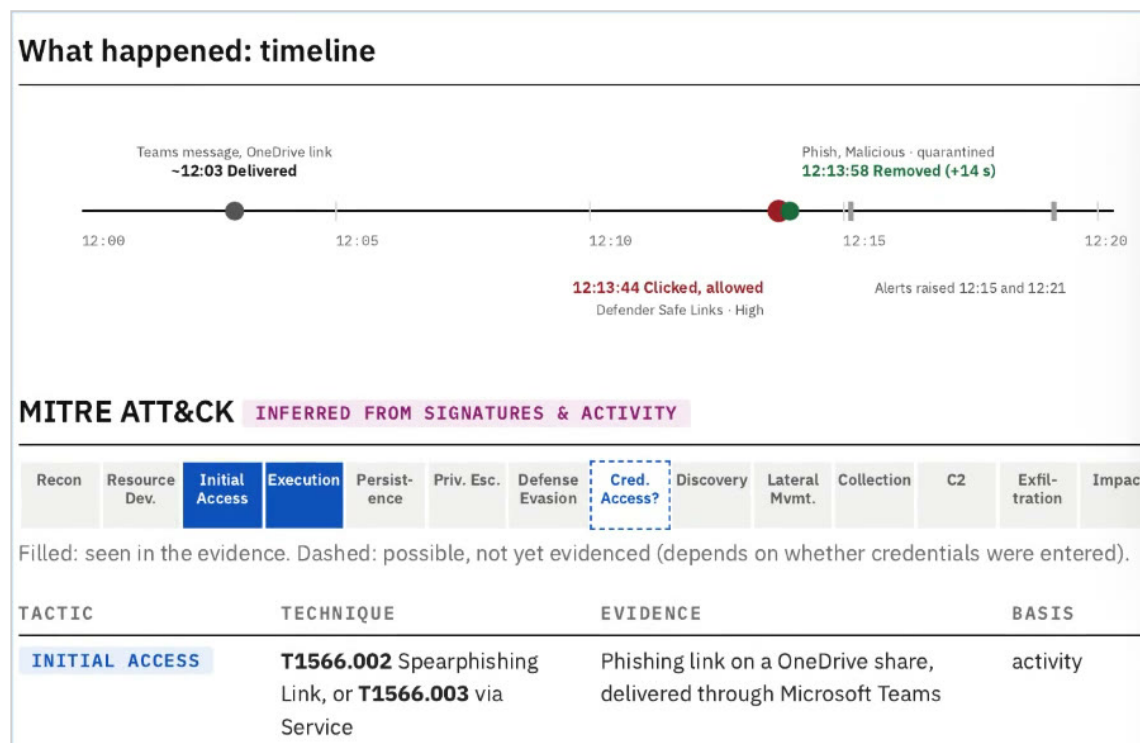


Figure 6. The timeline and MITRE ATT&CK mapping, inferred from signatures and activity. Filled tactics are seen in the evidence; the dashed one is possible but not yet evidenced. The full mapping also lists Execution (T1204.001).

The agent then maps the activity to MITRE ATT&CK: Initial Access through a spearphishing link on a OneDrive share delivered via Teams (T1566.002, or T1566.003 via service), and Execution through the user's click (T1204.001). Credential Access is marked as possible but not yet evidenced, because it depends on whether credentials were entered.

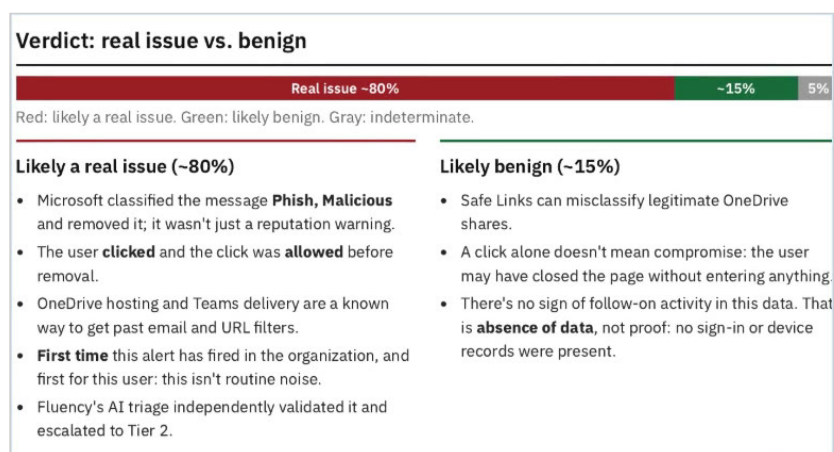


Figure 7. A quantified verdict with the reasoning on both sides.

The output is a finding, not a query result. The verdict is about **80% likely a real issue**, 15% likely benign and 5% indeterminate, with the evidence for each side written down. It also names the one check that would settle it: confirming whether the user entered credentials after the click. The absence of follow-on sign-in or device records in this data is stated plainly as an absence of data, not treated as proof of safety.

Every one of the four boundaries is crossed in this single job. The question was specific to this case. The method changed as the evidence did. The deciding fact came from Microsoft's own message handling, not from a detection rule. And the output is a decision with its evidence attached.

08

In Practice: One Sentence, an Eight-Chapter SOC Report

Reporting that SIEMs have traditionally not done at all

Investigation is only one job. The monthly SOC report is a larger one, and it is where the gap between a bounded SIEM and an agent-driven one is widest. The request is a single sentence:

Build the monthly SOC report for August.

The report skill takes it from there. It confirms the customer and the month, asks the data between eight and sixteen questions, runs an evidence QA pass that challenges its own first draft, and writes the report. Where data is missing, it says so, and marks the gap rather than filling it.

The result is eight chapters of written analysis, each section readable on its own without a chart, with charts and tables serving as evidence for the text:

CH.	TITLE	WHAT IT COVERS
01	Where things stand	Executive summary in business language, with two ratings: threat level and operational readiness.
02	What happened this month	Security events overview and rolling trends against prior months.
03	What the patterns tell us	Repeat offenders and root-cause themes.
04	What we can and cannot see	Operational exposure and monitoring blind spots.
05	What's at risk	Attack pathways and a likelihood-by-impact risk register.
06	How our defences performed	Controls, response and the escalation log.
07	What needs to happen next	Remediation plan and client actions: who owns each fix, and by when.
08	How we are moving forward	Improvements, plus appendices including the evidence QA record.

Along the way the report covers health and hygiene, validates detections against MITRE ATT&CK, and gives executives the decisions and forecasts they need to run the business. The ratings are deliberately split into two axes: **threat level** describes what adversaries did this period, and **operational readiness** describes whether the defences and processes could see, stop and close threats. A process gap never raises the threat level, and a red threat level never implies fault.

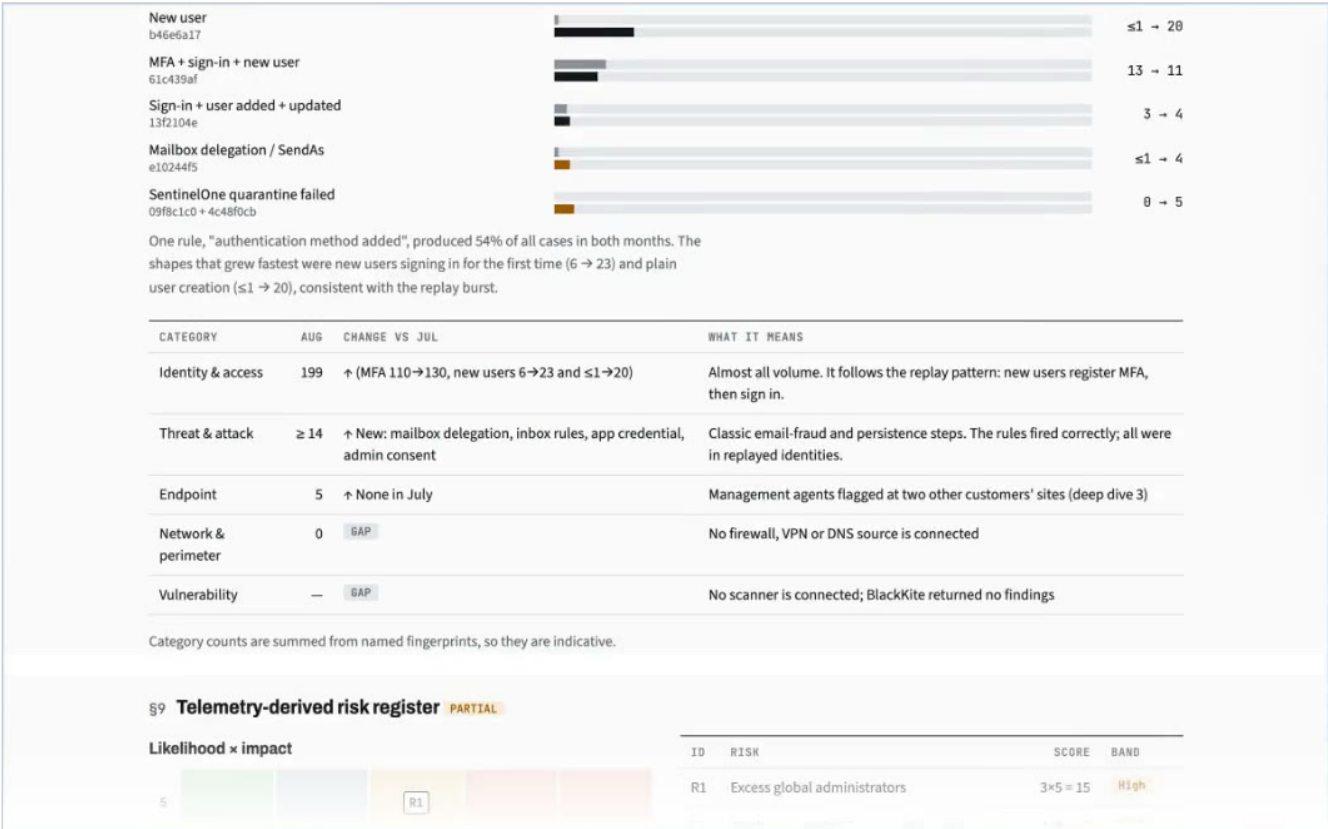


Figure 8. A page from the generated August report: headline categories compared with July, and the start of the telemetry-derived risk register. Expo demonstration tenant, synthetic identities.

Every chart is built from that month's data: headline numbers against the prior month, trends, risks and a remediation plan. The finished report is posted to Fluency Companion as a deliverable in the customer's Monthlies folder.

Any audience, any format

The conversation does not end with the report. The same session can continue: *compare it with July, add recommendations for the board, format it as a PDF, post it to Companion*. Without an agent, a vendor has to build every one of those variants as a product feature. With Fluency, every audience gets the same evidence, shaped the way it needs it.

09

Beyond the SIEM: Analyze, Forecast, Compare

The unbounded advantage reaches past detection

Once the agent can run the SIEM, the same approach applies to work that has traditionally lived outside it. This is the unbounded advantage:

Analyze

The whole environment, not one console view at a time.

Forecast

Where it leads if nothing changes.

Compare

Tenants, teams and time periods: what changed, and why.

Evidence and certification readiness

Fluency maps live evidence to NIST SP 800-53 Rev. 5, with related views for CSF 2.0, NIST 800-171, CMMC and ISO 27001, and marks each control as **supported**, **partial** or **not observed**. This supports readiness: it shows what the organization's own telemetry can actually demonstrate before an audit. Certification decisions stay with the assessor, and the output works alongside compliance platforms such as Vanta and Drata rather than replacing them.

Put the work on a schedule

None of this has to wait for someone to ask. Because skills make each job repeatable, a scheduled prompt runs the same way every time:

`MONTHLY, 1ST, 7:00`

"Build the monthly SOC report, post it to Companion, send the link to the team."

`WEEKLY, MONDAY, 6:00`

"Run the health status report for all tenants; message me if any source is broken."

`DAILY, 6:00`

"Summarize open cases scored 8,000 or more from the last 24 hours."

For MSSPs, this turns the platform from a tool the SOC watches into a service that produces deliverables: the monthly report on the first of the month, a weekly health check across every tenant, and a daily summary of critical cases.

10

Safe by Default

Reading is safe. Changing anything needs your say-so.

Putting an agent in front of a security platform raises an obvious question: what stops it from doing something it shouldn't? In Fluency the answer is built into the tools, not left to the prompt.

Read-only by default	Reports, searches and case reviews change nothing. Most of the work in this paper, including the full investigation and the monthly report, uses read-only functions only.
Changes are confirmed	Any change to a case, such as acknowledging or closing it, is previewed, confirmed by a person and recorded.
Your scope, every call	Role and scope are checked on every call. An operator works on one customer or the whole fleet, never a mix, and tenants stay isolated.
Secrets stay out of chat	Credentials go through a secure intake page and never enter the conversation with the model.

There is also no raw API passthrough: the agent can only do what a governed function allows. That is what lets security teams give an AI client real operational reach without giving it the keys to the platform.

11

Conclusion: Ask the Questions No Dashboard Anticipated

From a fixed interface to a governed, teachable operation

The SIEM is changing because the way people use it is changing. When the primary interface was a console, the vendor's job was to anticipate every question and build a screen for it. When the interface is an AI agent, the job is different: provide analytics strong enough that the agent starts from cases rather than noise, and teach the agent to run the security operation correctly and safely.

Fluency does both. Its behavioral analytics and evidence-based scoring turn telemetry into a short queue of explained cases. Its MCP teaches Claude or Codex the job in three layers: cheat sheets that load only the relevant context, governed tools instead of raw APIs, and skills that encode whole SOC jobs. The result removes the four boundaries of a traditional SIEM, so the question, the method, the sources and the output are no longer fixed in advance.

This is the Unbounded SIEM: ask the questions no dashboard anticipated, and get answers backed by evidence, through the AI you already use.

Getting started

Fluency offers a free 30-day trial, and Microsoft 365 is enough to start. Connect Claude or Codex, investigate a case, check coverage and build a report on your own data.

- Watch the video: youtu.be/gMeq1QFrqs
- The Unbounded SIEM: fluencysecurity.com/unbounded
- Connect Claude or Codex (free 30-day trial): fluencysecurity.com/ai-clients
- Sample SIEM and SOC reports: fluencysecurity.com/reports
- Book a demo: fluencysecurity.com/demo