



MARKET INTELLIGENCE

RESEARCH BRIEF

# Why Claude Can't Replace Your Security Stack

A plain-language breakdown of the five components of a security analytics platform, and where large language models actually create value.

---

PREPARED BY

**Frank Dickson**

DATE

**September 28, 2026**

STATUS

**Final**

---

**SECTION 01**

# The Question Being Asked

---

Those outside the discipline are increasingly asking a version of the same question: why can't Claude be the primary tool in a security operation? It's a fair question from that vantage point. The leading frontier models have shown they can address real security use cases, and they are exceptional at finding vulnerabilities in code. We use Claude as shorthand because it is the model most often named in the question, but the argument applies to any frontier model. It is also an argument about running a security operation, not about finding flaws in code, where these models already excel.

My answer is simple. A frontier model is an extraordinary analytics engine, and analytics is one of five components a security platform requires. The rest of this brief explains the other four, and where the models actually earn their place.

## The Four Questions

The question usually shows up as one of four, asked on a regular basis:

- Why can't Claude replace Splunk or other SIEM tools?
- Why can't Claude replace CrowdStrike Falcon or other EDR?
- Why can't Claude replace Zscaler or other cloud-based SSE offerings?
- Why can't Claude replace Palo Alto Networks or other network security stacks?

For readers outside the discipline, a quick decoder. A SIEM (security information and event management) platform collects and searches logs from across the organization. EDR (endpoint detection and response) watches laptops and servers for malicious behavior and stops it. SSE (security service edge) is a cloud service that inspects and controls user traffic to the web, SaaS applications, and private apps. A network security stack is the firewalls and related controls that sit in the path of traffic between networks.

Frontier models are now powerful enough that the question is fair. But it rests on an assumption that doesn't hold up: it treats security analytics as a single, homogeneous capability and leaves out how analytics are actually applied within a cybersecurity program.

*“Asking why Claude cannot replace CrowdStrike is like asking why a brilliant detective cannot replace the police department.”*

The detective may be the best reasoner in the building, but think of him as one newly transferred from another city. He knows how to work a case. He does not yet know these streets, these informants, or this precinct's history, and the department has to supply all of it. He still needs patrol cars on the street watching for trouble, a desk sergeant screening which reports are worth his time, radios and a dispatcher who can move twelve units to the right address in under a minute, and an evidence locker with a chain of custody that will hold up later. Reasoning is one component. It is not the department.

That maps directly onto how a security platform works. The patrol cars are telemetry. The desk sergeant is insight and intelligence. The radios and the dispatcher are orchestration. The evidence locker is data management. The detective, the one doing the reasoning, is analytics. Here is how each of those five roles works in practice.

## SECTION 02

# The Five Components of a Security Analytics Platform

Each of the four questions above fails on a different component. Splunk solves a data management and search problem. CrowdStrike is telemetry plus runtime response. Zscaler is insight and intelligence applied inline, at the network edge. Palo Alto Networks is orchestration across a platform estate. None of them are primarily analytics problems, which is the one component an LLM actually is. Each of these companies spans all five components, as the closing section notes. The mapping here isolates the component that defines each company's core identity, and analytics is deliberately absent from the list.

## 1. Telemetry, the Patrol Car

Gathering telemetry is a specialized engineering discipline. It's necessary, and it has real consequences for system performance, so the work has to be done carefully.

---

A model cannot reason about an event with no sensor recorded, the same way a detective cannot work a crime no patrol car drove past. Collection is a kernel-level engineering problem: the sensor has to observe process creation, memory injection, credential use, and network connections without measurably slowing the machine it protects. That is domain-specialized work, built on deep knowledge of operating system internals, driver stability, and attacker tradecraft, and it has to stay light on CPU, memory, and battery across every endpoint in the fleet.

Coverage is the constraint. Endpoint, identity, network, cloud control plane, and SaaS each require a different collector, and the seams between them, like the seams between patrol beats, are where attackers operate. None of this is the job of a general-purpose LLM, even one wrapped in agents. A model can call an API. It cannot sit in the kernel of every laptop and server, watching continuously without getting in the way. The detective doesn't patrol the streets.

## 2. Insight and Intelligence, the Desk Sergeant

Telemetry is voluminous. Dickson Research estimates that a typical endpoint security agent generates roughly 150 to 200 MB of raw security telemetry per day on the device, before any filtering or upload.

Only a small fraction of that data has value, and the desk sergeant's job is to know which fraction. That means performing preliminary analysis to illuminate potential malicious activity and doing it fast enough to matter: runtime blocking and other automated responses become increasingly important when a compromise is measured in milliseconds. Ransomware encryption is the classic case: the endpoint agent has to recognize the pattern and stop it locally, before any round trip to the cloud could complete.

The same logic holds at the network edge, which is where Zscaler and Palo Alto Networks live. An SSE proxy or a next-generation firewall sits in the path of the traffic, inspecting each session and deciding whether to allow or block it in the time it takes for the traffic to pass. There is no room in that path for a model to deliberate. The desk sergeant has to make the decision at the front door, not after the visitor is already in the building.

Selecting preliminary indicators for centralized analysis prevents a flood of data that would otherwise show up as latency and cost, much like a desk sergeant who forwards every call keeps the detectives from ever reaching the ones that matter.

The arithmetic makes the point. Taking the top of that range at roughly four bytes per token, 200 MB of telemetry is on the order of 50 million tokens per endpoint per day. Four bytes per token is a rule of thumb for English prose, and structured logs usually tokenize less efficiently, so the true figure is likely higher. An organization with 10,000 endpoints would generate roughly 500 billion tokens daily, before adding a single log from identity, network, or cloud. Even at a deliberately low assumption of \$1 per million input tokens, sending all of it to a

---

frontier model would cost about \$500,000 a day. No security budget survives that, and none should.

In practice, the screening happens in two tiers. The first is on the endpoint, where the agent triages its own telemetry and acts in milliseconds. What survives that filter moves up to the XDR (extended detection and response) layer, where it is correlated with signals from identity, email, network, and cloud. AI can augment that correlation, but it still has to happen fast, and much of it runs on rule sets written by engineers who understand what each sensor can and cannot see. That is where AI is earning its keep today: helping vendors write and tune detections faster, drawing on real-time and historical threat intelligence, on behalf of customers who rarely have the time or staff to do detection engineering themselves. The desk sergeant isn't being replaced. He's getting a faster way to write the standing orders.

### 3. Orchestration, the Radios and the Dispatcher

Orchestration is the most undervalued and underappreciated attribute of a security analytics platform. Collecting data from thousands of points, across tens of unique platforms, for centralized analysis, and then orchestrating a response across those platforms and across physical and virtual endpoints, takes engineering discipline and design genius. APIs, certificates, protocols, and OAuth assertions all have to be properly maintained, the way a dispatcher has to keep every radio channel and every unit's location straight in real time.

Orchestration also has to be secured and held to a higher standard of care than almost anything else in the stack. Security platforms are intimate. The SolarWinds breach, in which attackers turned a trusted IT management tool with broad network access into a delivery mechanism, shows what happens when that trust is compromised, the equivalent of an attacker seizing the dispatch desk itself.

The strongest counterargument is that an agent can simply call the APIs these platforms already expose, through the Model Context Protocol (MCP) or otherwise, and let the model act as the dispatcher. It is a fair point, and agents will increasingly do exactly that. But an agent calling APIs is a consumer of orchestration, not a replacement for it. Someone still has to build, secure, and maintain every one of those integrations, and handing a model the keys to dispatch raises the standard of care rather than lowering it. A new dispatcher with a radio still depends on the channels, the call signs, and the units someone else put on the street.

### 4. Data Management, the Evidence Locker

Homogenizing data is critical, because you cannot run a data transformation every time you want to do an analysis. That's slow, and latency kills. CrowdStrike's 2026 Global Threat Report puts average eCrime breakout time at 29 minutes, down from 48 minutes in 2024, with the fastest observed breakout at 27 seconds. In one 2025 intrusion, data exfiltration began within

---

four minutes of initial access. A large data transformation does not let you hit that clock, any more than reorganizing the evidence locker mid-chase gets an officer to the scene faster.

It's also expensive, and security does not have an infinite budget. That's why security is increasingly treated as a data pipeline problem. The more data you can string together ahead of time (especially at or near runtime), cataloged and ready, the simpler the search for maliciousness becomes. PowerShell shows exactly why.

## The PowerShell Conundrum: A Data Problem, Not a Reasoning Problem

Consider how this plays out in a case every SOC analyst recognizes: a PowerShell script of uncertain intent. PowerShell is a scripting language built into Windows for systems administration. It's powerful for administrators, which is exactly why it's also powerful for attackers running living-off-the-land attacks that use legitimate system tools instead of custom malware.

It can be very difficult to decide whether a PowerShell script is malicious by looking at the script alone. But if we see that the script was launched from a Word document, that the document arrived in an email from outside the organization, and that the sender's email came from a known malicious IP, we can say with high confidence that the script is malicious.

That analysis doesn't hinge on a flash of insight. A frontier model can reason about likely attack paths, drawing on threat intelligence and known attacker techniques, but only over evidence that has already been assembled. Once the evidence locker produces the case file, the conclusion is close to obvious. Without it, even the best detective is guessing.

## 5. Analytics, the Detective

Analytics are hardly homogeneous. Every security analytics platform runs a sea of machine learning algorithms and deterministic analytics engines side by side.

Even if the industry once thought an LLM could be the sole analytics engine, the rise of the harness in solutions from CrowdStrike and Microsoft points toward a better answer, one designed to deliver stronger performance at a lower cost. A harness is the layer that wraps the models, and it does three jobs. It routes each question to the right engine, a deterministic rule, a purpose-built ML classifier, or an LLM, rather than sending every question to the most expensive one. It supplies context and memory, handing the model the assembled case file, the asset and identity details, prior incidents, and the organization's own policies, so the model reasons about this environment rather than about security in the abstract. Finally, it enforces controls and guardrails, limiting what the model can see and do, checking its outputs, and requiring human approval before any consequential action. It is, in effect, the captain deciding

---

which detective works which case, handing him the file, and signing off before anyone makes an arrest.

Frontier models lack the foundations of telemetry, insight and intelligence, orchestration, and data management. That's not a knock on the models. It's a description of what they were built to do. The detective was never supposed to drive the patrol car.

There is also the matter of trust. A deterministic rule gives the same answer every time, and that answer can be explained to an auditor or a court. An LLM can give different answers to the same question, can state a wrong conclusion with complete confidence, and can be manipulated by prompt injection, where an attacker plants instructions inside the very logs, emails, or files the model is asked to analyze. That is why the harness's guardrails matter, and why the chain of custody in the evidence locker cannot be handed to the detective. A detective who can be talked out of a conclusion by the suspect's own notes needs a supervisor.

## A Word about XDR

XDR has fallen out of favor as a marketing term, as vendor attention has shifted to AI. However, the principle that telemetry from multiple security technologies is aggregated by an analytics platform to improve detection and subsequent response across multiple control points is alive and well. XDR is the art and application of telemetry, insight and intelligence, orchestration, data management, and analytics, the five components of a security platform. Trend Micro built TrendAI, the enterprise business it renamed in March 2026, on these principles, integrating its Vision One solution with native sensor coverage across endpoints, identity, email, network, and cloud workloads. For TrendAI, operational efficiency comes at the intersection of executing on all relevant analytics disciplines.

## Where Frontier Models Do Add Value

Frontier models add real value when they run on top of that foundation, which is why security vendors are building them directly into their platforms. The strongest use cases:

- **Investigation narrative:** turning a confirmed detection into a readable account of what happened, for an audience that does not read EDR consoles.
- **Analyst time:** summarizing, triaging, and prioritizing hundreds of correlated alerts so the most urgent are actioned first, drafting the incident timeline, writing the post-incident report.
- Natural-language access to data the platform has already collected, homogenized, and stored.
- **Threat hunting:** surfacing suspicious patterns and likely attack paths across telemetry the platform has already collected, informed by threat intelligence and known attacker techniques.

- 
- Judgment on genuinely ambiguous cases that deterministic rules deliberately leave open.
  - **Remediation:** recommending the right response actions for a confirmed incident and, where the platform's orchestration layer allows, executing them with an analyst in the loop.

### SECTION 03

## The Bottom Line

---

An LLM is an analytics engine. A security analytics platform includes telemetry, insight and intelligence, orchestration, data management, and an analytics engine. It is the whole department, not just the detective.

To be precise: Splunk, CrowdStrike, Zscaler, and Palo Alto Networks each address all five components, not only the one used above to make each company's core identity legible. The framing was simplified deliberately to keep the argument easy to follow. The simplification doesn't change the conclusion: all five components matter, and none of these companies built their platform around analytics alone.

For security leaders, the practical guidance follows directly. First, evaluate AI features on the foundation underneath them: ask every vendor what telemetry the model sees, how that data is homogenized, and what the model is allowed to do on its own. Second, measure AI by analyst time recovered and mean time to respond, not by the polish of the demo. Third, keep an analyst in the loop for any response action a model recommends until the platform can provide evidence for it. Fourth, treat any pitch to rip out the stack and replace it with a model as what it is: a proposal to disband the department and keep the detective.

*“Claude does not replace Splunk, CrowdStrike, Zscaler, or Palo Alto Networks. It sits on top of the machinery they build and makes the output of that machinery legible and actionable. That is a real job. It is not the same job.”*

FRANK DICKSON, DICKSON RESEARCH

**COPYRIGHT NOTICE**

This Dickson Research document was published as part of Dickson Research's independent market intelligence coverage. Visit [dicksonresearch.com](https://dicksonresearch.com) to learn more about Dickson Research. Please contact Dickson Research at [frank@dicksonresearch.com](mailto:frank@dicksonresearch.com) for information on reprints, web rights, or permission to cite this research in external materials.

This document is provided for informational purposes only. It is based on sources Dickson Research believes to be reliable, but Dickson Research makes no warranty as to its accuracy, completeness, or fitness for any particular purpose and accepts no liability for decisions made in reliance on it. Opinions reflect Dickson Research's judgment at the time of publication and are subject to change without notice. Nothing in this document constitutes investment, legal, or financial advice.

Dickson Research's analysis is independent. References to specific vendors, products, or services do not constitute an endorsement.

All product and company names mentioned in this document are trademarks or registered trademarks of their respective owners.

**Copyright © 2026 Dickson Research. Reproduction is forbidden unless authorized. All rights reserved.**