Detection Engineering in the Age of AI

Samson Adewale · Threat Detection & Response Engineer

KSU Cybersecurity Speakers Series · April 28, 2026

01

About me!

- KSU Alumni - 2016 🦉
- GA Tech Alumni - 2025 🐝
- Senior Threat Response Engineer @ Klaviyo 🛡️
- Passionate about AI and Automation

What is Detection Engineering?

The discipline of building and maintaining the systems that identify threats.

NIST Cybersecurity Framework - Govern, Identify, Protect, **DETECT**, Respond and Recover

DATA LAYER

1

Collect

Logs, events, and telemetry from endpoints, cloud, and network infrastructure

LOGIC LAYER

2

Define

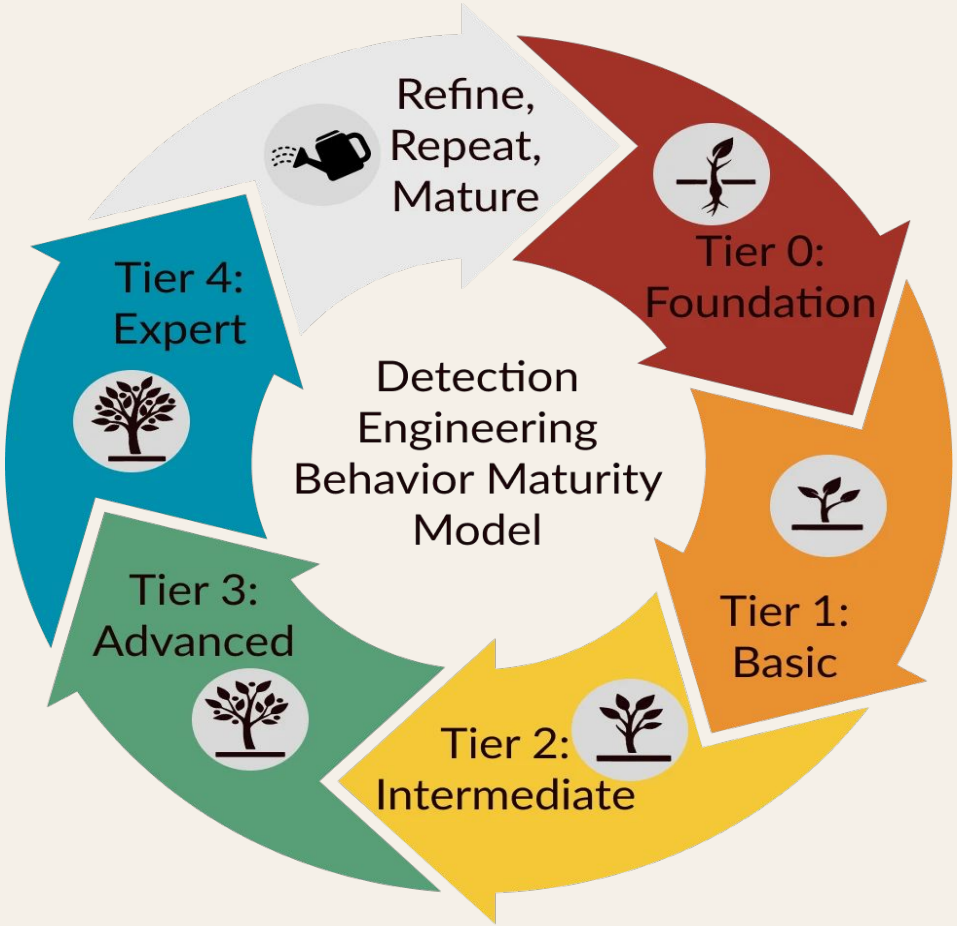
Write detection rules and logic — SIGMA, KQL, YARA, SPL. This is where human expertise lives.

OUTPUT LAYER

3

Alert

Rule fires, alert is created, analyst queue fills. The output of the whole pipeline.



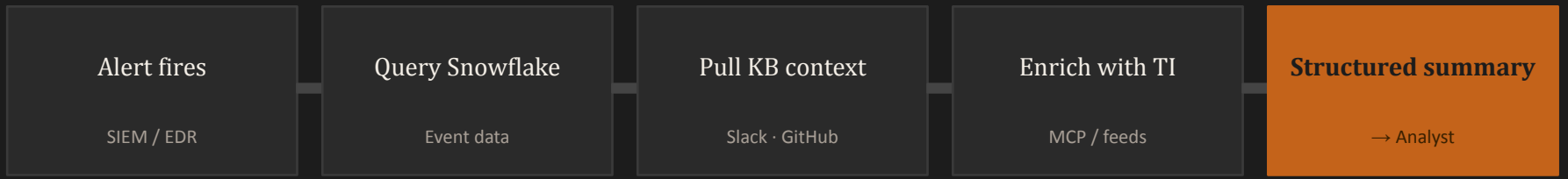
Resource: https://www.elastic.co/security-labs/elastic-releases-debmm

The current model has cracks.

Reactive by design	You only detect what someone already thought to look for. Novel TTPs slip through every time.
Human bottleneck	Detection rules are hand-written. Coverage is capped by analyst bandwidth and knowledge.
Alert fatigue	Too much noise, too little signal. Analysts burn out. Real threats hide in plain sight.
Context is fragmented	Alert fires. Analyst opens 4-6 tools to manually reconstruct what happened.

AI enters the picture : Post-Alert

First generation: AI as investigative partner, not replacement.



- Reduces mean-time-to-understand from 30+ minutes to <2 minutes
- Analyst receives assembled context — not just an alert number
- Consistent investigation quality regardless of analyst experience level

Before the alert.

What if AI could reduce noise and surface threats before a rule ever fires?

NOISE REDUCTION

Pre-alert scoring

AI scores every potential alert before it reaches the queue. Behavioral baselines, peer group analysis, historical patterns. Most noise never reaches an analyst.

SIGNAL DISCOVERY

AI-proposed detections

Agent reads Snowflake telemetry continuously, identifies anomalous patterns with no matching rule, and surfaces candidate detections for a human to validate and promote to production.

The data is already in Snowflake. The architecture makes this tractable today.

The detection pipeline, reimagined.

TODAY	WITH AI
Telemetry collected	Telemetry collected
Human writes detection rule	AI + human co-create detections
Rule fires ⟶ alert created	AI pre-scores ⟶ noise filtered
Analyst manually investigates	AI auto-investigates ⟶ context ready
Analyst queries 4–6 systems	Structured summary ⟶ analyst
Analyst writes up findings	Human validates, decides, acts

You don’t need a SOC to begin.

01

Learn the data model

Pick any free SIEM — Splunk free tier, Elastic. Ingest logs. Write your first detection rule. Understand what a field looks like.

02

Experiment with AI-assisted triage

Take a sample alert. Feed it to an LLM with context. See what it surfaces. Notice what's missing or wrong.

03

Build a simple agent

Use LangChain or CrewAI. Give it one tool — a CSV of logs. Ask it to investigate a fake alert. Watch it reason through the problem.

04

Study the guardrails

Learn structured outputs, human-in-the-loop patterns, least-privilege tool access. This is where the craft separates good from great.

The practitioners who understand this shift will define what SecOps looks like.

09

Thank You

Samson Adewale · Threat Response Engineer

Questions? Let's talk. - samsonadewale.com - samwale008@gmail.com