OWASP Newcastle upon Tyne

Large Language Threat Detection

Supercharging the SOC Tier 1 with AI

Michael Lamb - Director of Security Operations Precursor Security & SANS Instructor Candidate (LDR551)

Precursor

SANS

```
$ whoami

> Michael Lamb
> Dir. of Security Operations @ Precursor Security
> 5,000 hours of Incident Response experience
> Responded to Akira, BlackCat (ALPHV), Lockbit2/3, NoEscape incidents
> Passionate about building awesome threat detection teams
> GitHub/Twitter (X) @mikecybersec
```


Who We Are

CREST Certified

NEBRC Trusted Partner

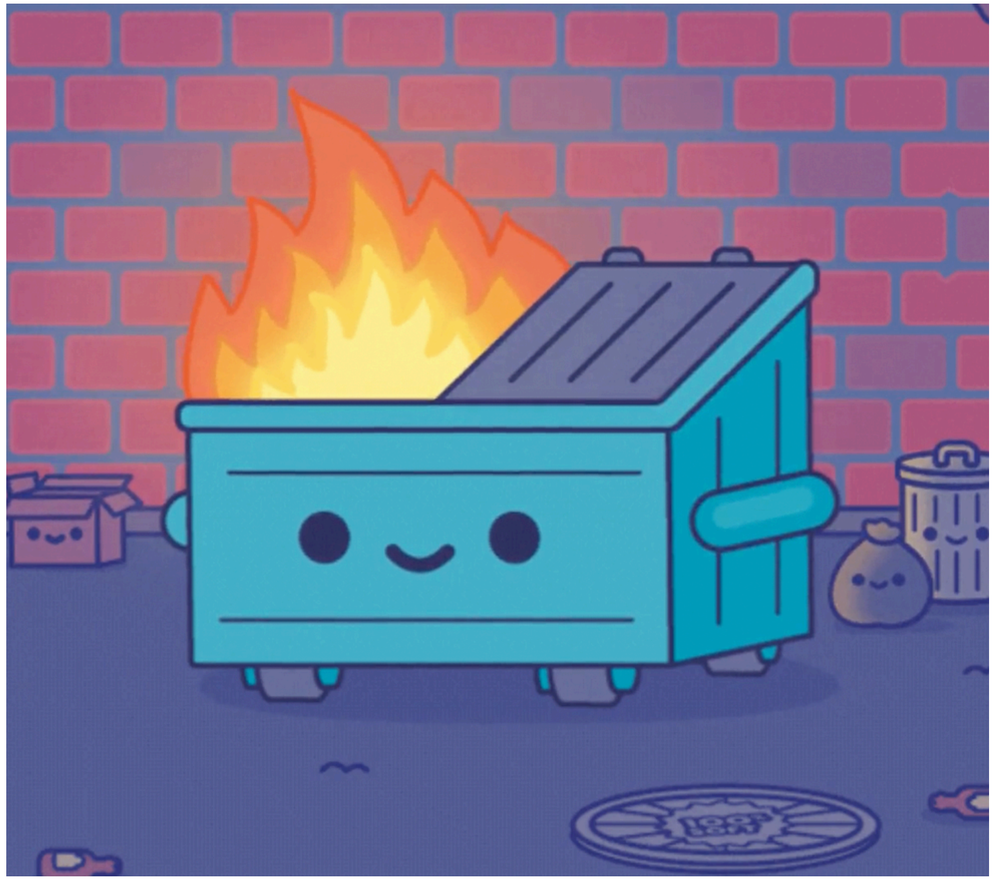
ISO27001
ISO9001

Cyber Essentials Certification Body

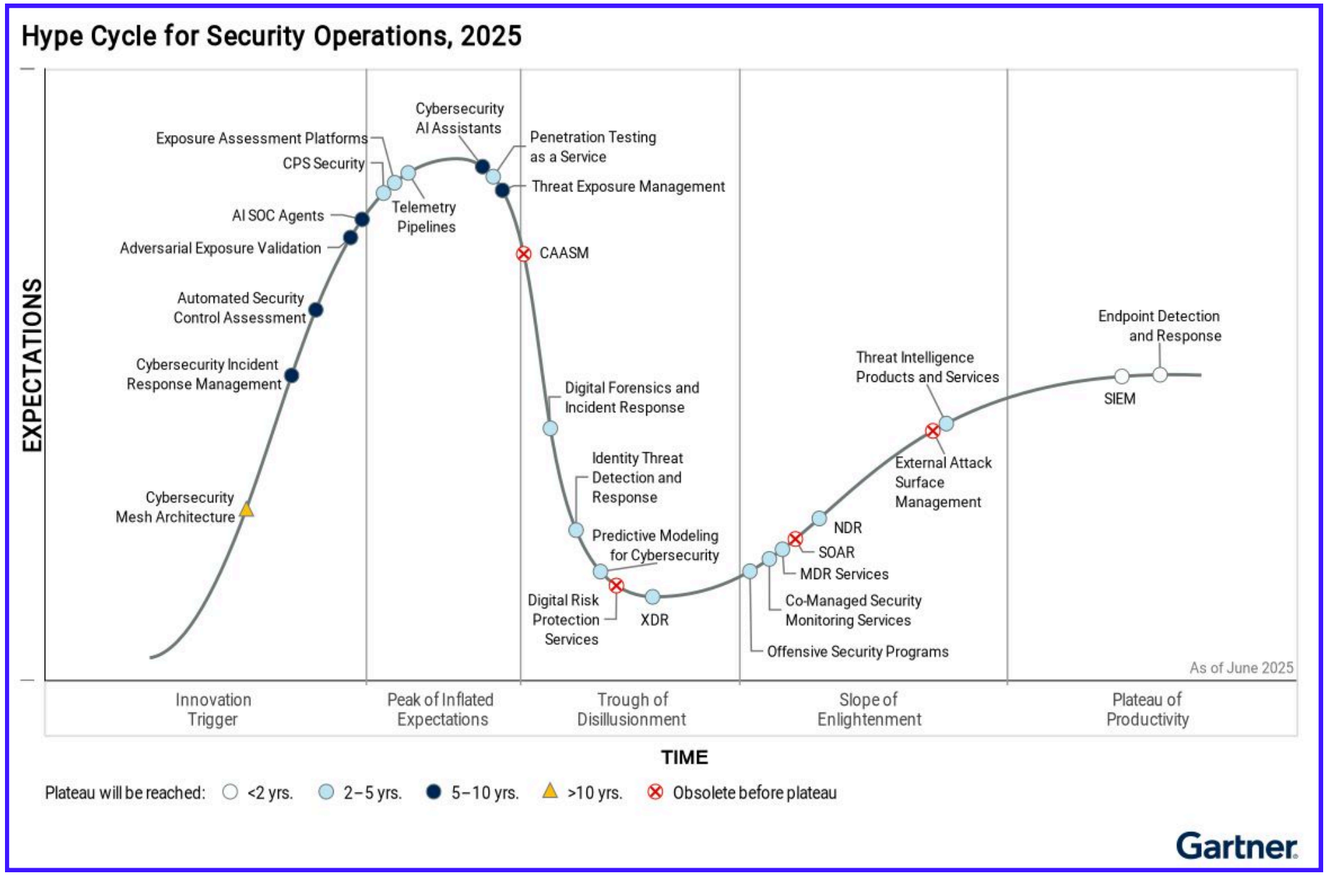
CCS Supplier

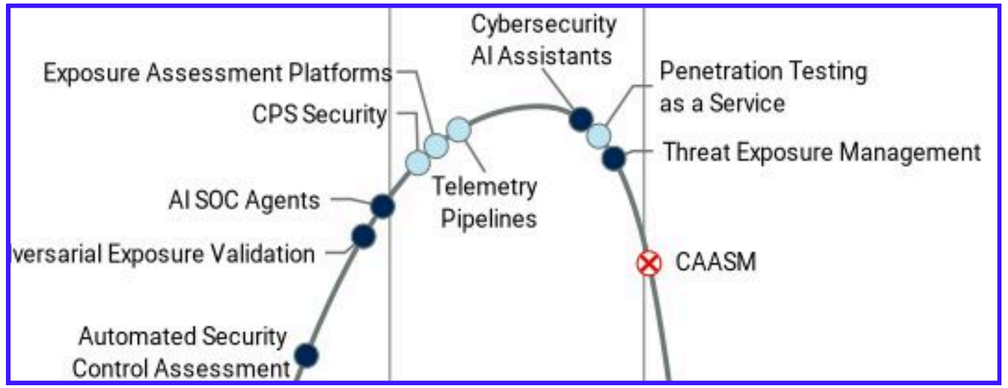
Highly Qualified Staff

The Current State of SOC



Agentic AI is the new SOAR



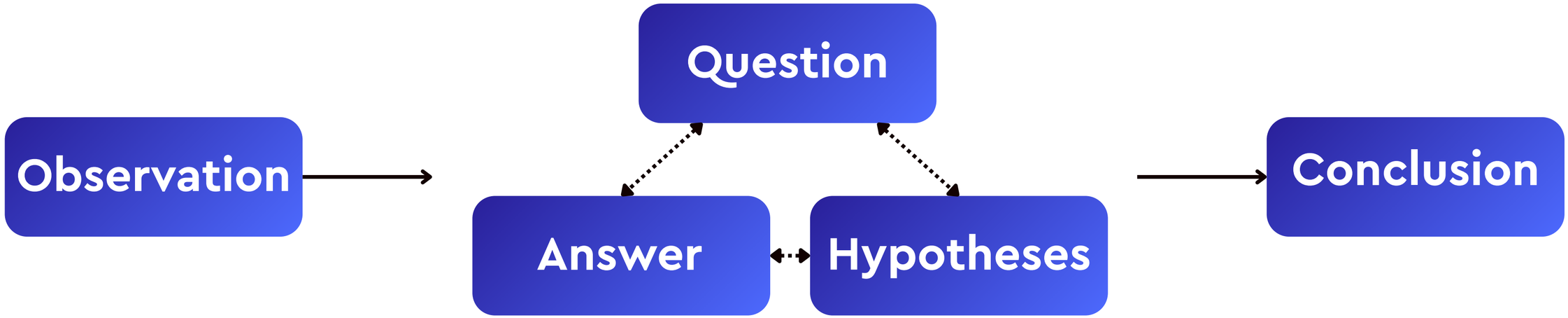


- SOAR is obselete before market plateau
- AI SOC Agents are in an innovation trigger with 5-10 years before plateau

SOC AI Ideas

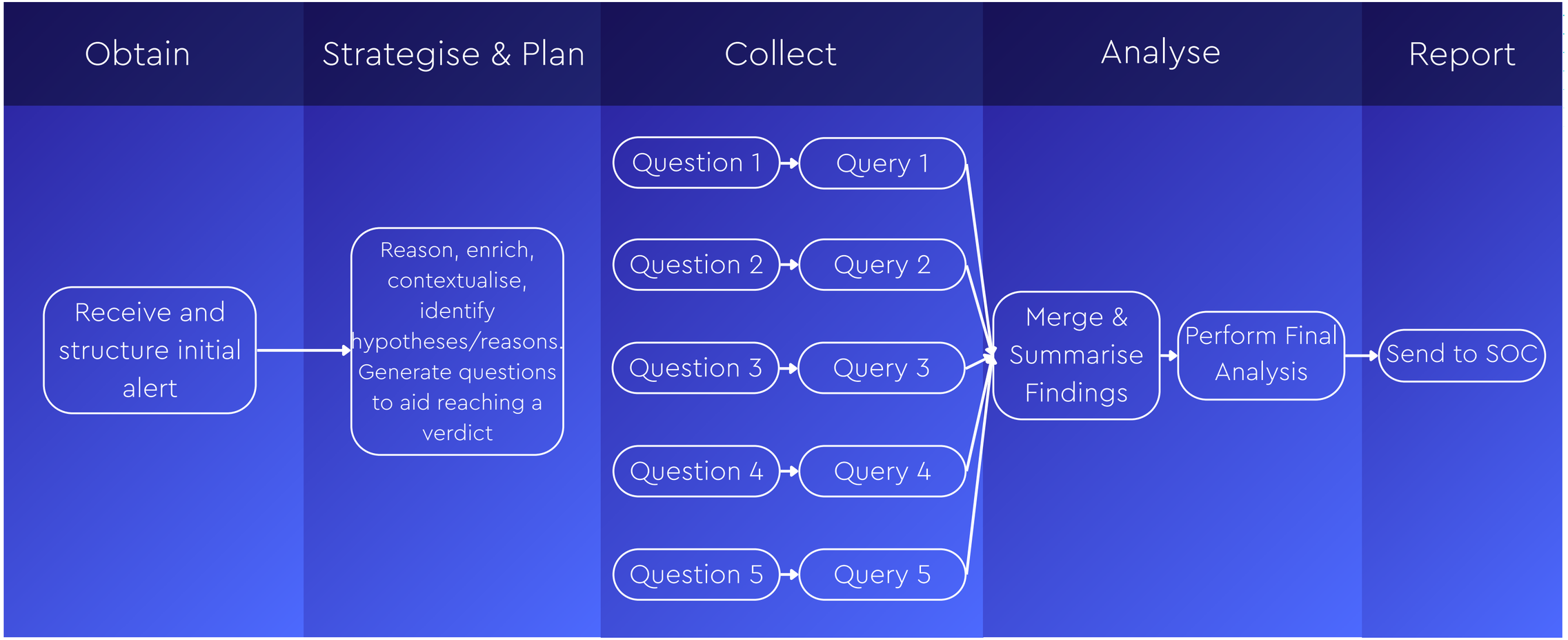
- Summarising alerts & incidents
- Generating queries for investigations
- Copilots/Bots
- Analysing (un)structured data
- Alert correlation
- **Performing Tier 1 Activities on low-fidelity alerts ✅**

Diagnostic Inquiry



"Analysts asked relevant questions when they based them on the interpretation of existing evidence within the current investigation or other investigations involving similar components and techniques." - **The Analyst Mindset: A Cognitive Skills Assessment of Digital Forensic Analysts Chris Sanders, Ed.D.**

OSCAR



Source: Dropzone, Why SOCs Rely on OSCAR: A Proven Investigative Framework

Reducing TOIL in SOC

TOIL	SOC Mapping
Manual	Analysts manually investigate similar alerts (e.g., false positives, benign behavior).
Repetitive	High volume of similar alerts (e.g., failed logins, known scanning activity).
Automatable	Many steps (e.g., enrichment, triage, correlation) can be automated or semi-automated.
Reactive	Analysts respond to alerts after they fire — no proactive value added unless feedback loops exist.

Google SRE - Operational Efficiency: Eliminating Toil

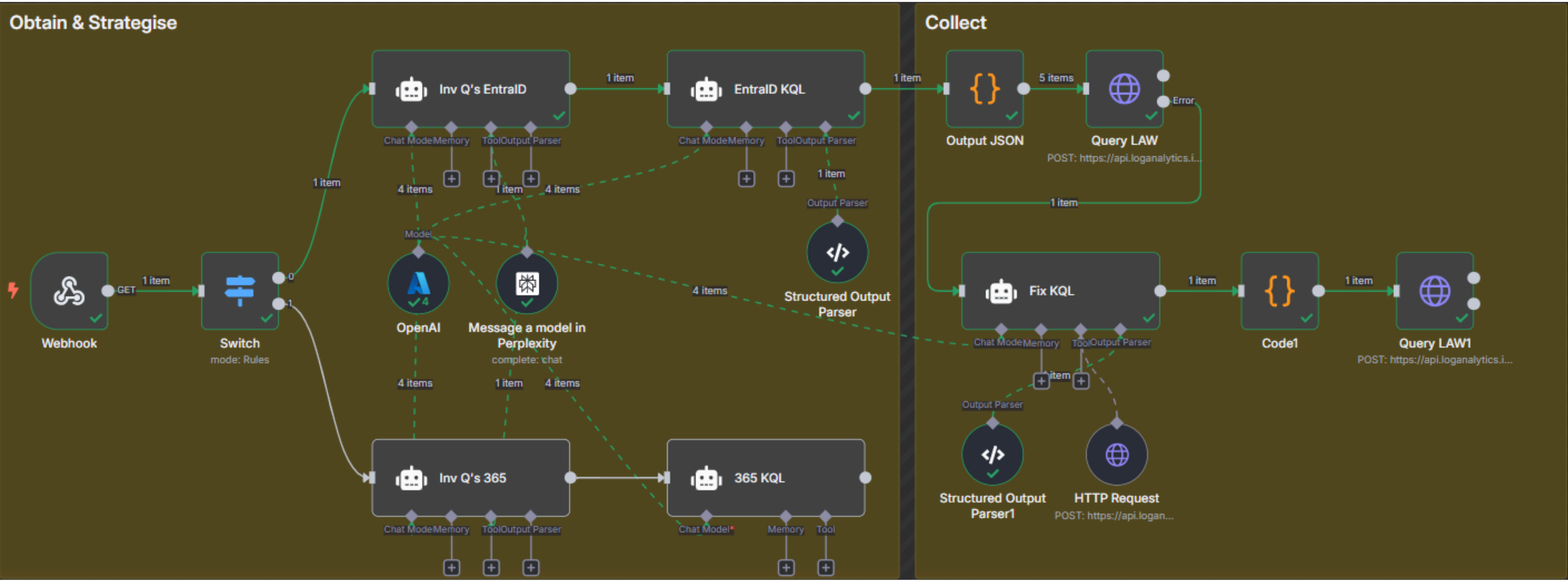
What I've Tried

- HP DL380 server w/ NVIDIA GPU & DeepSeek (self-hosted)
  - N8N Local Server
  - CrewAI
  - Langchain
- Demos with other vendors

What's Working

- Azure Open AI (GPT-4o) + N8N Cloud

The Flow


Obtain & Strategise
Collect
Webhook
GET
1 item
Switch
mode: Rules
0
1
1 item
Inv Q's EntraID
Chat Model
Memory
Tool
Output Parser
1 item
EntraID KQL
Chat Model
Memory
Tool
Output Parser
1 item
4 items
1 item
4 items
1 item
Output Parser
Structured Output Parser
Model
OpenAI
Message a model in Perplexity
complete: chat
4 items
4 items
1 item
4 items
Inv Q's 365
Chat Model
Memory
Tool
Output Parser
365 KQL
Chat Model*
Memory
Tool
1 item
Output JSON
5 items
Query LAW
POST: https://api.loganalytics.i...
Error
1 item
Fix KQL
Chat Model
Memory
Tool
Output Parser
1 item
Code1
1 item
Query LAW1
POST: https://api.loganalytics.i...
item
Output Parser
Structured Output Parser1
HTTP Request
POST: https://api.logan...

The Result

User Consent Denied for OAuth Application

🔍 **Investigative Question**

What is the name, application ID, and registered owner of the OAuth application that requested consent?

🧠 **Investigative Rationale**

Identifying the app and its creators helps determine whether it's a known benign application or could be associated with an attacker. Many malicious apps have names that mimic reputable services. This question helps attribute the activity for further investigation.

```
AuditLogs
| where OperationName has "Consent to application"
| where Result == "failure"
| extend AppName = tostring(TargetResources[0].displayName),
AppId = tostring(TargetResources[0].id),
RegisteredOwner = tostring(InitiatedBy.user.userPrincipalName)
| project TimeGenerated, AppName, AppId, RegisteredOwner
```


Other Questions

User Consent Denied for OAuth Application

From which IP address and geographic location was the OAuth consent request initiated?

Did the user receive prior login or unusual activity requests from the same IP or location within the last 24 hours?

Were there any subsequent user actions, such as elevated privileges or application usage, following the denied consent event?

Were there any other OAuth consent requests (approved or denied) involving the same user or app across the organization in the last 7 days?

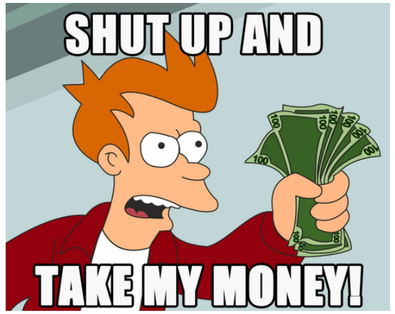
Example Output & Cost Breakdown

Input Prompt	8,896
Output	2,390
Estimate	£0.0336

Pricing (1M Tokens)

Input: £1.82037
Cached Input: £0.9102
Output: £7.2815

Daily estimate of ~£3 per day based on 80 similar alerts everyday



KQL Bench

Comprehensive AI evaluation framework testing large language models' ability to generate cybersecurity detection rules using real-world attack scenarios

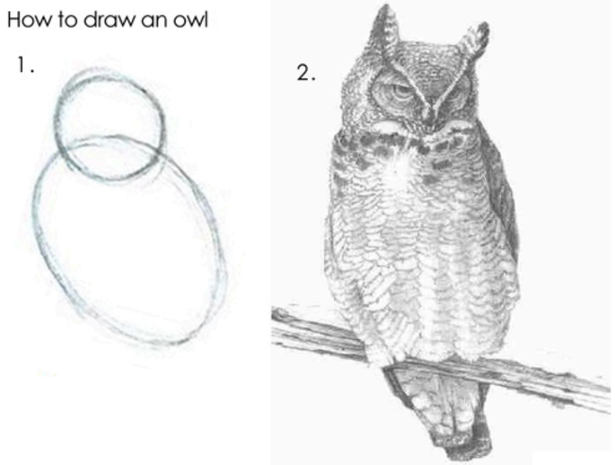
Model	Overall Success Rate (%)	Avg. Attempts	Avg. Exec Time (s)	Total Cost ($)
o1-low	63.3%	2.60	37.90s	$93.89
o1-high	63.3%	2.71	40.35s	$98.50
gpt-4.1	61.7%	2.74	6.93s	$5.36
grok-3-mini-beta	58.5%	2.53	16.55s	$0.75
o3-mini-low	51.6%	2.85	23.32s	$5.24

Tips for Success

- Provide known-good examples in the prompt (Few-Shot Prompts)
- Provide tips/guidelines/principles clearly
- Handle errors/common mistakes with a seperate agent
- Avoid repeating points throughout prompts
- Connect knowledge sources (RAG) - See vectorize.io
- Use the 'Pause & Wait for Human Response'
- Use prompt files (like .prompty) to manage prompt development and testing
- Use ReACT for reasoning traces
- Connect 'Tools'/MCP Servers to perform additional lookups and add context
- **Use a good model** (honourable mention: kqlbench.com)

SOC AI Workflow Maturity

Level	Description	Capabilities	Limitations
L0 — Basic Prompting	Copy-paste questions into ChatGPT or Copilot	- Answers basic KQL or incident triage questions- No schema knowledge	- Hallucinates fields - Wrong syntax - No memory
L1 — Prompt-Only Agent	Uses static system prompt via Azure OpenAI	- Uses KQL-only scaffold - Can generate valid queries - Can call SIEM API	- No awareness of tenant schema - No learning from feedback
L2 — Schema-Grounded + Memory	Adds RAG or hardcoded schema + stores query feedback	- Picks correct tables/fields - Avoids SQL syntax - Learns from corrections - Feedback loop improves future responses	- Needs schema updates as data sources change - Needs retrieval tuning
L3 — Autonomous Workflow + Evaluation	Fully integrated with query execution, correction, and scoring	- Generates query - Executes it - Corrects and retries - Scores quality - Learns from outcomes	- Higher complexity - Requires state/memory - Needs cost control (API usage, retries)



1. Draw some circles 2. Draw the rest of the fucking owl

Thank You

Questions

GitHub/Twitter(X): @mikecybersec