



SIEM and Detection Engineering

Blue team track | Detection lifecycle, Sigma, SPL/KQL/EQL, ATT&CK coverage, alert quality, tuning, SOAR

#SIEM

#DetectionEngineering

#Sigma

#SOC

≡ Vocabulary and identifiers used on this sheet

① Sources and how to read the ids

Grounded in NIST SP 800-92 (Guide to Computer Security Log Management) and SP 800-61r3 (Incident Response Recommendations, 2025), plus practitioner knowledge. **ATT&CK ids**: MITRE ATT&CK (Adversarial Tactics, Techniques and Common Knowledge) is a public catalogue of attacker behaviour. TA00xx = tactic (the attacker's goal, e.g. TA0002 Execution), T1059 = technique (how: Command and Scripting Interpreter), T1059.001 = sub-technique (PowerShell). **Windows event ids** are numbers the OS writes in a log channel: 4624 (successful logon), 4769 (Kerberos service ticket requested) and the other numbers below are always spelled out where used. CySA+ CS0-003 = CompTIA Cybersecurity Analyst exam; GCIH = GIAC Certified Incident Handler.

SIEM — Security Information and Event Management: platform that collects logs from many sources, normalises them, stores them for search and runs correlation rules that raise alerts (Splunk, Microsoft Sentinel, Elastic, QRadar).

SOAR — Security Orchestration, Automation and Response: playbook engine that enriches alerts and runs response actions (block, disable account) through APIs.

EDR / XDR — Endpoint Detection and Response: agent that records process, file, network and registry activity on each host and can isolate it; XDR (Extended DR) adds email, identity and cloud telemetry.

SOC — Security Operations Centre: the team that monitors alerts 24/7, triages them and escalates incidents.

Sigma — open, vendor-neutral YAML format for detection rules, converted by sigma-cli into the query language of each SIEM.

SPL / KQL / EQL — query languages: Splunk Search Processing Language, Kusto Query Language (Microsoft Sentinel and Defender), Event Query Language (Elastic, for event sequences).

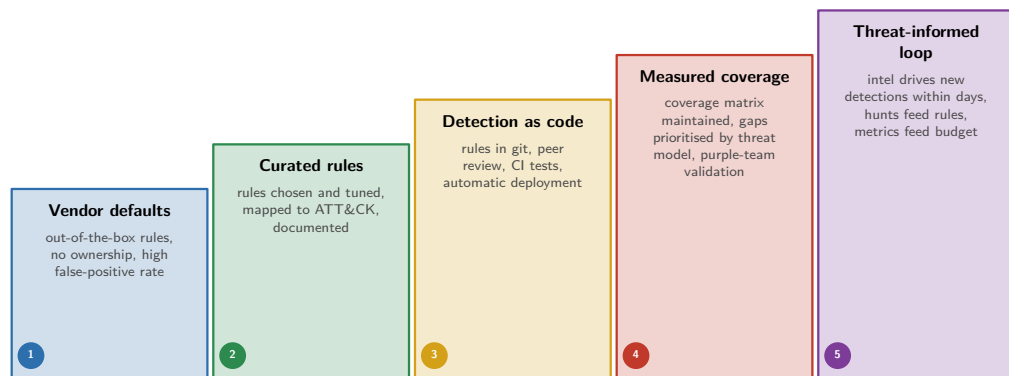
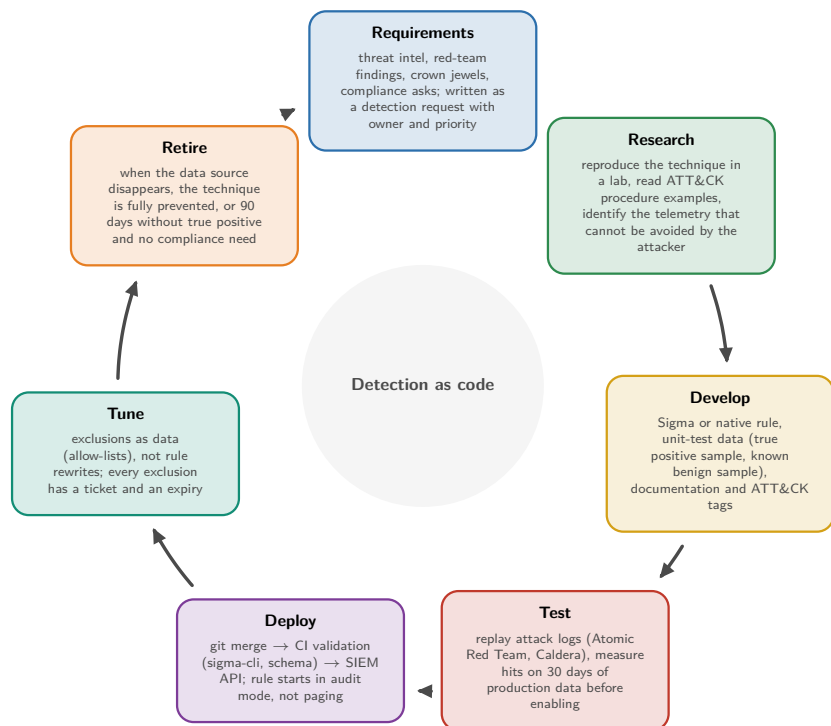
UEBA — User and Entity Behaviour Analytics: baselines normal behaviour per user or host and scores deviations (new country, unusual volume).

IdP — Identity Provider: the service that authenticates users and issues tokens (Microsoft Entra ID, Okta); its sign-in logs are a key detection source.

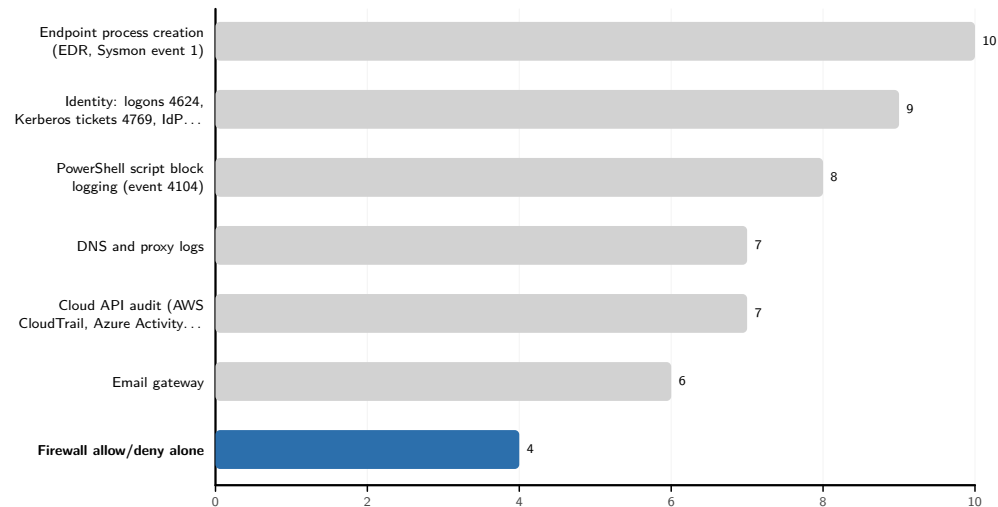
Sysmon — free Microsoft Sysinternals driver that writes detailed process, network and file events to the Windows event log (event 1 = process created, 10 = process accessed another process).

Metric	Formula	What it tells you
Precision (fidelity)	$\frac{TP}{TP + FP}$	share of alerts worth an analyst's time; target ≥ 0.8 on paging rules
Recall (coverage)	$\frac{TP}{TP + FN}$	share of real attacks the rule catches; measured with purple-team replays
MTTD	$\overline{t_{detect} - t_{start}}$	Mean Time To Detect: average delay between first malicious action and alert
MTTR	$\overline{t_{contain} - t_{detect}}$	Mean Time To Respond: alert to containment; lower is better
Alert load	$\frac{alerts/day}{analysts}$	above about 30 per analyst per day triage quality drops (estimate)

☀ Detection engineering lifecycle



Detection maturity (author's synthesis of common SOC maturity models)



Data sources by detection value for a typical enterprise (author estimate, relative score out of 10)

Detection — a rule plus its documentation, test data, response playbook and owner; a query alone is not a detection.

Benign true positive (BTP) — the rule fired correctly on activity that was authorised (admin, scanner, red team). Tune with an exclusion, not by deleting the rule.

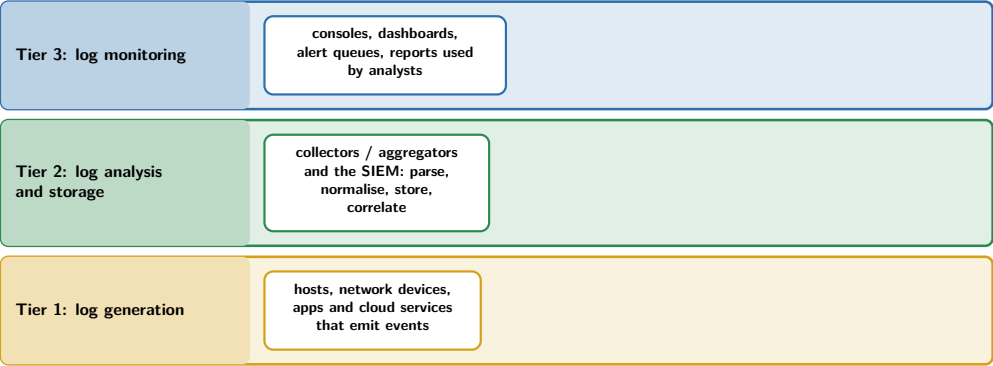
Detection gap — an ATT&CK technique relevant to the threat model with no rule or no telemetry; tracked in the coverage matrix.

Alert fidelity — share of alerts that are true positives; high-fidelity rules page, low-fidelity rules feed dashboards and hunts.

EXAM TIP

CySA+ (CS0-003, objective 1.3 and 3.1) uses the terms **true positive**, **false positive**, **true negative**, **false negative** and asks which one a *missed* attack is: a false negative. It also expects you to know that tuning *reduces false positives*, while adding data sources reduces false negatives.

📖 Log pipeline: from source to alert (NIST SP 800-92)



- 1 Collect** agent push (Winlogbeat, Splunk UF, Azure Monitor Agent) or pull (API polling of SaaS and cloud); syslog on 514/udp is lossy, prefer TLS on 6514/tcp
- 2 Parse** split raw text into named fields (user, src_ip, process); unparsed fields cannot be searched efficiently
- 3 Normalise** map vendor fields to one schema (Splunk CIM, Elastic ECS, OCSF) and one time zone (UTC) so one rule works on every source
- 4 Enrich** add asset owner, criticality, geo-IP, threat-intel match at ingest time
- 5 Filter / reduce** drop proven-useless noise (debug logs) to control licence cost, never security-relevant events
- 6 Store** hot (searchable, 30-90 days) then cold / archive (cheap, 1 year or more per regulation)
- 7 Correlate and alert** scheduled or streaming rules, thresholds, sequences, risk scores per entity

What happens to one event between the host and the analyst

Clock sync	every source on NTP (Network Time Protocol); a 5-minute skew breaks sequences and timelines (SP 800-92 lists inconsistent timestamps as a core challenge)
Log rotation	closing a log file and starting a new one on schedule or size; size local logs so they never roll before the collector reads them
Integrity	forward off-host in near real time; an attacker with admin rights can clear local logs (event 1102)
Retention	driven by regulation and dwell time: keep at least 12 months searchable or restorable (PCI DSS requires 1 year, 3 months immediately available)

Alerting on single events vs risk-based alerting (RBA)

✓ RBA: each rule adds risk points to a user or host; alert when the sum crosses a threshold	✗ Limits of RBA
+ noisy low-fidelity rules become useful signals instead of being disabled	- needs good entity resolution (same user in AD, IdP, VPN)
+ one alert tells a story (5 weak signals on one host) instead of 5 separate tickets	- a fast single-step attack may not reach the threshold: keep direct paging rules for high-fidelity events
+ fewer alerts per analyst, higher precision	- risk scores need tuning and are harder to explain to auditors

🔗 Sigma: write once, run everywhere

```
Sigma rule anatomy: encoded PowerShell command line
title: Suspicious Encoded PowerShell Command Line
id: 6d9e4b5a-1c2f-4a8e-9b3d-2f5c7e8a9b10
status: test
description: Detects -EncodedCommand with bypass flags
tags: [attack.execution, attack.t1059.001, attack.t1027] # T1027 = obfuscation
logsource:
  category: process_creation
  product: windows
detection:
  selection_img:
    Image|endswith:
      - '\powershell.exe'
      - '\pwsh.exe'
  selection_enc:
    CommandLine|contains:
      - '-enc '
      - '-EncodedCommand '
      - '-ec '
  selection_flags:
    CommandLine|contains: ['-nop', '-w hidden', '-ep bypass']

filter_sccm:
  ParentImage|endswith: '\OcmExec.exe'
condition: all of selection_* and not filter_sccm
falsepositives:
  - Software deployment tools using encoded commands
level: high
```

- **logsource** category (process_creation, network_connection, file_event, registry_set, dns_query, image_load, ps_script), product (windows, linux, macos, aws, azure), service (security, sysmon, powershell)
- **Modifiers** |contains, |startswith, |endswith, |re, |base64offset|contains (matches any base64 alignment), |cidr, |all (every value must match), |windash (matches - and / switches)
- **Condition** selection and not filter, 1 of selection_*, all of them; correlation rules (v2) add event_count, value_count, temporal with a timespan
- **Level** informational, low, medium, high, critical; high and critical page, medium goes to a queue, low feeds hunts

sigma-cli	Purpose
pip install sigma-cli pysigma-backend-splunk pysigma-backend-elasticsearch	install the converter and two backends
sigma convert -t splunk -p sysmon rule.yml	SPL with the Sysmon field mapping (pipeline)
sigma convert -t lucene -p ecs_windows rule.yml	Elastic Lucene query with ECS field names
sigma convert -t kusto -p microsoft_xdr rule.yml	KQL for Microsoft Defender XDR advanced hunting
sigma check rules/ ; sigma list targets	validate syntax, list installed backends

Splunk SPL	Microsoft KQL	Elastic KQL / EQL
Pipeline of commands separated by ; stats, eval, tstats on accelerated data models	Tabular operators piped with ; where, summarize, extend, join kind=inner	KQL for filtering in Kibana; EQL for sequences (sequence by host.id [process ...] [network ...])
Index-time fields + search-time extraction; CIM data models for vendor-agnostic searches	Strongly typed tables (SecurityEvent, DeviceProcessEvents, SigninLogs); functions in Sentinel	ECS schema; event.category, process.command_line; Elastic prebuilt rules ship as Sigma-like TOML
Scheduled searches → notable events (ES); risk-based alerting sums risk scores per entity	Analytics rules (scheduled, NRT, fusion); entity mapping feeds incidents and UEBA	Detection rules: query, threshold, EQL, indicator match, ML; alerts to cases

```
Same detection, three engines
# Splunk SPL (Sysmon TA, sourcetype XmlWinEventLog)
index=endpoint source=XmlWinEventLog:Microsoft-Windows-Sysmon/Operational
EventCode=1 (Image="*\powershell.exe" OR Image="*\pwsh.exe")
CommandLine IN ("* -enc *","* -EncodedCommand *","* -ec *")
NOT ParentImage="*\CcmExec.exe"
| stats count min(_time) as first by host, User, CommandLine

// Microsoft Defender XDR / Sentinel KQL
DeviceProcessEvents
| where FileName in- ("powershell.exe","pwsh.exe")

| where ProcessCommandLine has_any ("~enc ","~EncodedCommand","~ec ")
| where InitiatingProcessFileName != "CcmExec.exe"
| summarize count(), min(Timestamp) by DeviceName,
    AccountName, ProcessCommandLine

# Elastic EQL (winlogbeat/elastic-agent, ECS)
process where event.type == "start" and
process.name in ("powershell.exe", "pwsh.exe") and
process.args : ("~enc", "~EncodedCommand", "~ec") and
not process.parent.name : "CcmExec.exe"
```

The Sigma rule above firing in the SIEM queue, with the analyst pivots (illustrative; worked triage in the last section)

Time	Level	Source	Message
09:14:02	INFO	sysmon/1	FIN-LT-042 j.doe: WINWORD.EXE → powershell.exe -nop -w hidden -enc JAB...
09:14:03	ALERT	sigma	Suspicious Encoded PowerShell Command Line (high, T1059.001) ▶ parent is Word, not CcmExec: the filter did not apply
09:14:04	NOTICE	dns	cdn-update-srv[.]com A 203.0.113.45 (domain registered 3 days ago)
09:15:04	WARN	proxy	GET /api/check, 312 bytes, repeats every 60 s
09:31:40	WARN	dns	same domain from FIN-LT-017 and HR-LT-008
09:33:10	SUCCESS	edr	3 hosts isolated, incident INC-2291 opened

The alert alone is ambiguous; DNS and proxy pivots (same new domain on 3 hosts, 60 s beacon) make it a true positive

🕒 Coverage and alert quality

Data source	Initial access	Execution	Persistence	Priv. esc.	Cred. access	Lateral mvt
Email gateway	5	1			1	
Endpoint process / Sysmon	2	5	5	4	4	3
PowerShell 4104	1	5	3	3	4	3
Windows Security (AD)	1	2	4	4	5	5
Network / DNS / proxy	3	1	1		1	4
Cloud control plane	3	2	4	4	3	3

Tactic →

ATT&CK tactic coverage by data source (author estimate: 0 none, 5 primary source)

≥ 80%

precision on paging rules

target; true positive + benign true
positive over all alerts

< 1 h

time to detect (TTD)

target for endpoint techniques;
measured with purple-team replays

< 4 h

time to respond (TTR)

target to containment for critical alerts

≤ 30

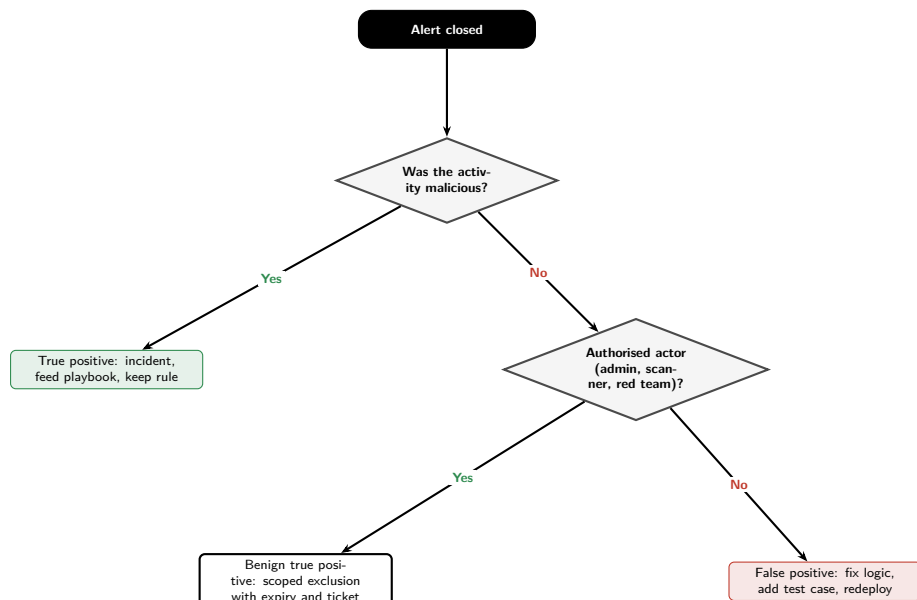
alerts / analyst / day

target; above this, triage quality drops

Actual	Predicted		Total
	Attack	Benign	
Attack	18 TP	4 FN	22
Benign	27 FP	951 TN	978
Total	45	955	1000
Precision	0.40	1.00	
Recall	0.82	0.97	
F1	0.54	0.98	

Accuracy 96.9% · n = 1000

One rule, one month (illustrative): rows = truth after triage or purple-team replay, columns = the rule's verdict. Precision 18/45 = 0.40 (tune out false positives), recall 18/22 = 0.82 (4 misses to cover)



Tuning decision after triage

Privileged group change

Member added to a security group: 4728 (global), 4732 (domain-local), 4756 (universal) targeting Domain Admins, Administrators (SID ending -544) or Enterprise Admins. Rarely benign outside change windows: page always.

Kerberoasting burst

Kerberoasting = requesting service tickets to crack service-account passwords offline. Detect ≥ 10 distinct SPNs (Service Principal Names) in event 4769 with ticket encryption 0x17 (RC4, weak and crackable) from one account in 5 minutes.

LSASS access

LSASS (Local Security Authority process) holds credentials in memory. Sysmon event 10 (process access) on lsass.exe with GrantedAccess 0x1010 (memory read) or 0x1ffff (full access) from a process that is not your EDR = credential dumping (ATT&CK T1003.001).

Audit tampering

Defenders blinded on purpose: 1102 (Security log cleared), 104 (System log cleared), 4719 (audit policy changed), Sysmon event 4 (Sysmon service state changed / stopped). ATT&CK T1070.001 and T1562.

New service on Tier 0

Event 7045 (new service installed, System log) on a domain controller or PKI server (Tier 0 = assets that control the whole domain). Red flags: PSEXESVC (PsExec remote execution), random 8-character names, binary in a temp folder.

Impossible travel

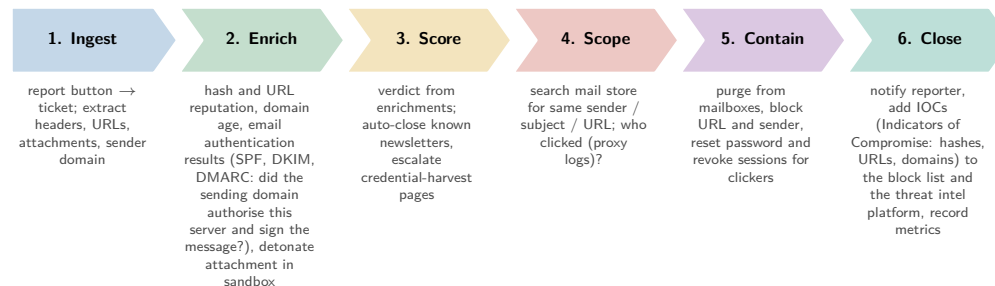
IdP sign-ins for one user from two countries within 1 h, faster than physically possible: likely stolen credentials or token. Exclude known VPN egress ranges to avoid false positives.

Beaconing

Proxy/DNS: same destination, near-constant interval, low jitter, small payloads over 24 h.

Cloud key abuse

AWS CloudTrail API calls: CreateAccessKey (new long-lived key = persistence), AttachUserPolicy with AdministratorAccess (privilege escalation), StopLogging (blinding the audit trail).



SOAR playbook: phishing triage (user-reported email)

⚠ DANGER

Common pitfalls: enabling every vendor rule on day one (alert flood, then rules are muted wholesale); exclusions written as NOT user=svc* that silence the technique for attackers who use service accounts; rules that depend on a field only one log forwarder populates; no test data, so a schema change silently breaks the rule (zero hits looks like zero attacks); measuring the SOC on alert volume instead of precision and TTD.

🔍 Correlation patterns and alert triage

	One event	Several events
Known bad pattern	Signature - one Sigma rule on one event - e.g. encoded PowerShell, 1102 log cleared - cheap, precise, easy to evade by renaming	Threshold / sequence - count or ordered chain in a time window - e.g. 20 failed logons (4625) then a success (4624) from one IP in 10 min - EQL sequence, Sigma correlation
	First seen / rare - value never observed before for this entity - e.g. new service name on a server, new parent-child process pair - needs a 30-day baseline table	Behavioural / UEBA - statistics over many events - e.g. user downloads 10x their daily volume - finds novel attacks, higher false positive rate

Security log cleared (1102) admins rarely clear logs; ask why

LSASS memory read by unknown process credential dumping

Office app spawns PowerShell or cmd malicious macro; some add-ins do it legitimately

Encoded PowerShell command line also used by deployment and monitoring tools

Failed logon burst (4625) password spraying or an expired service password

Connection to newly registered domain useful as risk points, noisy as a page

Typical fidelity of common detections (5 = almost always malicious; author estimate)



Task	L1	L2	L3 / IR	Det. eng.	CTI	SOC lead
Triage within SLA, enrich	R	C			C	A
Escalate a true positive	R	A	I			I
Scope and contain		R	A		C	I
Forensics, root cause		C	R A		C	I
Tune or retire a rule	C	C		R A		I
New detection from intel		C	C	R A	C	I

R Responsible A Accountable C Consulted I Informed

SOC tiers on one alert (L1 triage, L2 investigation, L3 incident response; small SOC's merge L2 and L3). R = does it, A = accountable (one per row), C = consulted, I = informed

Worked triage: encoded PowerShell alert

- 1. **Context:** host = finance laptop, user = standard account, parent = WINWORD.EXE (not a deployment tool).
- 2. **Decode:** the base64 string decodes to a download cradle fetching a script from a domain registered 3 days ago.
- 3. **Pivot:** same domain in DNS logs on 2 other hosts; proxy shows beacon every 60 s.
- 4. **Verdict:** true positive, initial access by macro (T1566.001 spearphishing attachment). Isolate the 3 hosts with EDR, block the domain, reset the user's credentials, open an incident.
- 5. **Improve:** add the parent-process pattern as a separate high-fidelity rule; add the sample to the rule's test data.

Q1 Which metric improves when you tune out false positives, and which when you on-board a new log source?

A Tuning raises precision (fewer FP). A new data source raises recall (fewer false negatives, i.e. missed attacks).

Q2 Why normalise logs to a common schema (CIM, ECS, OCSF)?

A So one rule using `user` or `process.command_line` works on every vendor's logs, instead of one rule per log format.

Q3 Sigma modifier that matches a string at any base64 alignment?

A `|base64offset|contains`

Q4 EQL keyword for ordered multi-event detections?

A `sequence by <field> [event1] [event2] with maxspan=...`

Q5 Difference between a false positive and a benign true positive?

A FP: the logic matched something it was not written for. BTP: the logic matched exactly the technique, performed by an authorised actor.