AI 驅動的資安告警優化～從疲乏到精準

蘇柏鳴 / 金融聯合徵信中心 資安部

告警疲乏如何拖垮資安防線

在多數組織的 SOC（Security Operations Center）與資安團隊裡，「告警疲乏（Alert Fatigue）」不是情緒或工作態度問題，而是一種會直接削弱防線的系統性風險，當 SIEM 每天湧入成千上萬筆告警，資安人員面對的是無止盡的「判讀、關單、再判讀」，久而久之就會出現兩個後果，其一是人力被雜訊消耗，導致沒時間挖掘的真正的事件；其二是心理與流程上的麻木，導致高風險訊號被當成一般雜訊忽略。

很多團隊曾經以為「告警越多越安全」，但現實往往相反，告警越多誤報越高，越可能讓真正的入侵更遲被看見。傳統規則式偵測在早期能快速建立基本盤，但進入規模化營運後就暴露極限，規則維護成本高、例外越開越多、攻擊手法快速變形、跨系統脈絡不足，導致規則要麼太鬆造成海量誤報，要麼太嚴造成漏偵。更典型的是「規則只加不減」，沒有人敢刪也沒時間回頭檢視，最後 SIEM 變成雜訊製造工廠，SOC 變成告警處理工廠。要把告警從「量」變成「質」，關鍵不是單純換工具，而是把資安監控當成一個端到端工程系統來經營，用資料工程把日誌資料變得可用、可對齊、可稽核，用偵測工程把規則變成可治理、可度量、可迭代的產品，用自動化（SOAR）把高頻流程標準化並可追溯；再用AI/LLM做去重、聚合、風險排序、調查協作與報告產出。目標不是「少告警」，而是「對的告警在對的時間到對的人手上」，讓有限的人力投入在真正重要的風險與事件上。

告警從哪裡來：日誌資料的全景圖(收集、可觀測性、正規化與治理)

任何告警治理（Alert Governance）都必須先瞭解「告警從哪裡來、資料怎麼走、品質如何被保證」。典型Log包含：

端點（Endpoint / EDR）

- ●程序/行程啟動（Process start）
- ●可疑指令與腳本（PowerShell、CMD、Bash、WMI 等）
- ●檔案落地/下載/解壓（Drop / Download）
- ●登錄檔修改、服務/排程建立、持久化行為

身分與目錄（AD / IdP / IAM）

●登入成功/失敗、異常地理位置/裝置

●MFA 啟用/停用、驗證失敗

●權限變更（Role/Policy）、特權提升

●帳號新增/停用/鎖定、群組變更（加入/移除）

網路設備（Firewall / VPN / Proxy / DNS / IDS / IPS）

●連線行為（來源/目的/埠/流量特徵）

●阻擋/放行紀錄（Policy hit）

●DNS 查詢與回應（可疑網域、DGA 特徵）

●VPN Tunnel、代理行為、異常通道（Tunneling）

雲端控制面（CloudTrail / Defender / Workspace 等）

●管理操作（Console/API 管理事件）

●權限授與/變更（IAM Role/Policy）

●金鑰/憑證建立與輪替（Access key、Service account）

●資源建立/刪除/變更（VM、Storage、Network、Security Group）

應用程式與 API

●認證/授權事件（AuthN/AuthZ、Token）

●敏感操作：匯出、刪改、批次下載、管理功能

●錯誤碼與異常模式（暴力嘗試、撞庫、邏輯濫用）

●API 呼叫行為（高頻、異常路徑、異常參數）

資料庫（DB）

●登入/失敗/來源異常

●查詢模式異常（大量掃描、可疑語句、時間分佈）

●權限變更（Grant/Revoke）

●資料匯出/備份/大量讀取

WAF / CDN

●攻擊流量（SQLi、XSS、RCE、L7 DDoS）

●Bot 行為與自動化特徵

●規則命中（Rule hit）與封鎖原因

●異常地區/ASN/來源分布

真正的挑戰往往不在「有沒有收」，而在「收上來能不能形成可關聯的證據」。常見痛點包括：資料遺漏（Agent 覆蓋不足、API 連線失敗、設備傳輸中斷）、格式不一致（同類事件欄位名不同、值域不同）、時間不同步（時區、NTP、延遲造成排序錯誤）、欄位品質差（使用者/資產識別不清，host 名稱與資產編號對不起來）、成本不可控（把大量低價值事件塞進最昂貴的 Hot storage）。因此需要「讓 SIEM 看得懂」的正規化與語意一致性：建立 Schema/Taxonomy（欄位統一、事件分類、資產/使用者識別規則），並把關鍵脈絡一併補上（例如：資產重要性、系統用途、部門/地點、身分角色、高權限/服務帳號標記、弱點與修補狀態，以及 CMDB 與 IAM 群組資訊），同時要做資料治理：保留策略（哪些留 90 天、哪些留 1 年、哪些只留摘要）、分層儲存（Hot/Warm/Cold）以控管成本、機敏資料遮罩與存取權限（避免因集中化而擴大資料外洩風險），以 Splunk 為例，若沒有先把 sourcetype、時間戳、欄位抽取、CIM（Common Information Model）對齊與資

產/使用者映射做扎實，後續再強的 correlation search 也只是在處理不一致資料；反之，資料層穩定後，偵測與關聯才能真的「事件化」，而不是「堆告警」。

SIEM：告警匯聚與關聯分析的核心（偵測工程與規則優化）

SIEM 的角色應該從「收集/查詢平台」升級成「決策輔助中樞」：把各系統的訊號整合成可追溯的時間線與可理解的情境脈絡，支援更快的判斷與決策，提供集中查詢、關聯分析（Correlation）、儀表板與稽核追溯。許多組織導入 SIEM 後成效不彰，典型原因往往是只把資料堆進來卻沒有偵測工程，規則越寫越多且缺乏治理，導致同樣雜訊日復一日，偵測工程的核心精神是把規則當成「可管理的產品」而不是一次性腳本。

實務上可先做三件事。

(1) 規則分級與事件化設計

把規則區分為高置信度（命中後可快速進入處置）、需脈絡（必須結合資產重要性/身分角色/time window才有意義）、需人工判讀（偏提示，避免用高優先級干擾）。例如「新增網域管理員」在多數環境可視為高置信度；「凌晨異常登入」若未結合地理位置、裝置指紋、是否伴隨敏感操作，可能只能是需脈絡。

(2) 白名單與例外管理制度化

例外必須具備理由、範圍、期限與風險接受人，並定期複查，避免「永久例外」成為攻擊者的長期通道。

(3) 以 MITRE ATT&CK 做覆蓋盤點與缺口管理

不要只看告警量，而要看攻擊鏈段落是否有偵測，包括初始存取、憑證存取、權限提升、橫向移動、持久化、資料外傳等是否有對應控制點。

治理上要把指標納入日常營運與例行檢視，例如：誤報率（False Positive Rate）、MTTA/MTTR、告警壓縮率、事件命中率（Alert-to-Incident conversion）、規則貢獻度（哪些規則帶來最多雜訊/哪些最有效命中），以及規則維護成本（每月調整次數、例外數量），以 Splunk 的使用情境來說，可透過 correlation search + notable event，把同一使用者短時間內的異常登入、權限變更、敏感資料存取串成事件，同時以風險型告警（risk-based alerting）把多個中低風險訊號累積成高風險事件，而不是每個訊號都丟給人，當規則能版本化、能回溯驗證、並由指標持續驅動調整，SIEM 才會真正從「雜訊來源」轉成「事件工廠」。

情資收集與脈絡：把「訊號」變成「意義」

威脅情資（Threat Intelligence， TI）常被誤解為「黑名單比對」，只要 IOC 命中就告警、就封鎖，這種用法往往帶來反效果，其實IOC是有時效性的，過期後仍命中會製造雜訊，資料來源品質不一，污染與重複會造成誤傷；更常見的是對組織情境不具相關性（lack of contextual relevance），某些攻擊基礎設施在別的產業有效，但在你所在環境可能全是誤報。情資的正確價值應該是「風險排序」與「脈絡補強」。

情資可分為 IOC（IP/Domain/Hash/URL）、TTP（戰術技術程序）、漏洞/Exploit 趨勢（哪些弱點正在被大量利用）、惡意家族與攻擊基礎設施（C2、投遞網域、工具鏈），來源可以是商用 TI、OSINT、產業共享、以及內部事件發現；接入可透過 TAXII/STIX、API、Feed 管理自動化更新，落地時需要兩個關鍵機制：信心分數（Confidence Score）與情境加權（Context Weighting），信心分數包含來源可信度、是否被多來源交叉驗證、觀測時間距今多久、是否與已知攻擊活動相關，情境加權則把「資產重要性 + 使用者角色 + 行為異常 + 現場關聯結果」一起納入，例如同樣命中惡意網域：若發生在測試環境的一般帳號、且無後續敏感操作，風險分數應低，但若發生在核心系統的高權限帳號、同時伴隨權限提升與大量資料查詢，風險分數應立即拉高並觸發事件化。

進一步可把 TI 的用途從「比對命中」升級到「調查加速」，當事件發生時，自動回填該 IOC 的家族歸屬、常見 TTP、相關 CVE、已知攻擊活動時間線，協助資安人員更快形成假設與查證路徑。TI 若能與 SIEM 的事件關聯、資產清單、身分資料、弱點資料一體化，就能避免「情資越多越吵」的困境，反而把告警排序做得更精準，讓 SOC 的注意力被引導到真正值得處理的風險上。

SOAR：讓回應從手動流程變成可控的自動化（分級自動化與治理控制）

當告警品質改善後，SOC 的下一個瓶頸通常會移到「回應流程」：資安人員仍花大量時間做重複工作，例如開票、查詢、蒐證、通知、彙整時間線，甚至在不同系統間複製貼上，SOAR 的價值在於把這些高頻、可標準化、可稽核的流程工程化，讓回應更快、更一致、更可追溯，並把 SIEM 的偵測結果轉成可行動的工作流。典型 Playbook 可以涵蓋：告警分流（依風險分數、資產等級、攻擊類型自動分派到適當隊列）、證據蒐集（自動查詢同 time window內的相關 log、拉取端點狀態、取得雲端審計紀錄、彙整使用者近期活動）、處置動作（封鎖惡意 IP/網域、隔離端點、鎖定帳號、要求密碼重置或 MFA 強制、建立工單與通知），以及結果回寫（誤報/真事件標註、處置動作、耗時、影響範圍）。衡量 SOAR 成效的 KPI 應落在營運面：減少 L1 手工步驟比例、縮短 MTTR、提高回應一致性、降低重工率、提升稽核可用性。

但自動化一定伴隨風險，因此必須「分級自動化」並加入治理控制點：先從建議模式（提出封鎖建議與證據）、再到半自動（人審核後執行）、最後才是全自動（僅在高置信度且低誤傷風險情境）。所有動作都要可稽核，誰在何時基於何證據觸發何行為，並保留回滾機制，並且還要有熔斷（Circuit Breaker），當偵測規則異常爆量、外部情資來源誤導、或整合系統不穩定時，SOAR 能自動降級到建議或人工模式，避免誤封造成業務中斷，把這些治理與保護機制做進去，自動化才會是「可控的加速」，而不是「失控的風險」，最重要的

是，SOAR 也應成為閉環的一環：把結案原因與處置結果回饋到 SIEM 規則與 AI 標註資料中，讓整個系統越用越準。

AI 驅動告警優化 + LLM 在 SOC 的角色

在資料、偵測、情資與流程逐步工程化後，AI/LLM 才能真正落地並放大效益，AI 在 SOC 最務實的價值，通常集中在四類：

(1) 去重與聚合（Alert Dedup/Clustering）

把同源多告警合併成事件，避免分析師被重複訊號消耗。

(2) 異常偵測（UEBA/NBA）

建立行為基線，抓出不尋常的登入時間、地點、頻率、資源存取模式，彌補規則追不上變形攻擊的缺口。

(3) 風險評分（Risk Scoring）

把資產、身分、行為、TI 與關聯結果匯成優先序，讓 SOC 先處理高風險事件

(4) 誤報抑制（False Positive Reduction）

透過相似案例比對、root cause分類與模型判別降低雜訊，技術路線可依成熟度選擇，有足夠歷史結案標註可採監督式分類，標註不足可用非監督式。

若要處理跨實體的關聯（使用者、主機、IP、程序、雲資源），可以用圖模型把關係串起來（例如 Entity Graph、Attack Path），若要善用文字型知識（工單、IR 報告、分析紀錄），則可用 Embedding/RAG 做語意檢索，快速找到相似案例與處置脈絡，但要讓這些方法真正落地，關鍵前提仍是「閉環」，標註要能從 SOC 結案、工單、IR 報告與 SOAR 執行結果持續回收；再透過主動式學習，讓分析師把時間花在最有價值的樣本上，同時模型需具備版本管理與可回溯驗證機制，並持續監控模型漂移與概念漂移，確保效果不隨環境變化而失真。

至於 LLM更像「調查與決策副駕」，把雜亂欄位轉成一致敘事摘要、提出下一步假設與查詢建議、透過 RAG 檢索過往案例與 Runbook、協助產出事件時間線與通報草稿，為避免胡說，LLM 必須綁在可引用證據上，資料來源包含 SIEM 事件、SOAR 工單、資產清單、IR Runbook、威脅情資，檢索策略以實體（Entity）+ time window + 相似度多路並行，輸出必須附證據引用與信心描述，查不到就明確回答「未知/資料不足」並且重大處置仍由人工把關，系統只提供建議，避免模型直接觸發不可逆操作。

整體參考架構可總結為，Log Pipeline（收集、正規化、Enrichmen及分層儲存）、 SIEM（規則、關聯及風險評分）、TI（Feed 管理、可信度/時效及情境加權）、AI（聚合/異常/排序/根因）、SOAR（Playbook、工單及治理控制）及 LLM 助理（RAG、調查建議、報告自動化且不做不可逆操作），精準不是告警少，而是對的告警在對的時間到對的人手上，成功關鍵也不是單點 AI，而是資料工程、偵測工程、流程治理與 AI/LLM 輔助共同形成可持續的營運系統。