Regular Paper

Communications of the CCISA
Vol. 28 No. 4 Nov. 2022

運用網路封包分析與機器學習之勒索病毒偵測技術

蔡文淙[1*]、林韶如[2] 、劉得民[3]、周兆龍[4*]
[1,4]國防大學理工學院資訊工程學系、[2]國家中山科學研究院、[3]中華民國網路封包分析協會
[1]wentsung.tsai@gmail.com、[2]oceankeep@gmail.com、[3]dmliu99999@gmail.com、[4]chaolung.chou@gmail.com

摘要

企業及政府機構遭勒索病毒攻擊的資安事件，近年來逐漸登上新聞或資安網站的版面，駭客透過駭侵手法滲透使用者電腦，甚至運用社交工程，藉由執行勒索病毒將其電腦文件檔案進行加密，受害者如急於取回文件，避免造成組織營運停滯、個人利益受損，有可能依駭客指定方法進行贖付。為減少損害，在受這類攻擊的當下，爭取應變時間就是首重目標，因此需要動態分析的方式，即時偵測出勒索病毒的攻擊。

本研究基於勒索病毒在網路環境發動攻擊時所產生特殊的異常行為，提出「勒索文件封包數」與「異常封包數」兩項指標，偵測同一區網內部電腦，是否遭受勒索病毒攻擊，並運用決策樹、循序最小優化及簡單邏輯迴歸等機器學習演算法，依所提出之兩項指標數值對不同勒索病毒進行分類。經 600 次的實驗，實驗結果平均準確率可達 99.25% 以上，證明本文提出之方法可有效地偵測並分類勒索病毒。

關鍵詞：勒索病毒、動態分析、網路封包、機器學習

* 通訊作者 (Corresponding author.)

Regular Paper

Communications of the CCISA
Vol. 28 No. 4 Nov. 2022

Ransomware Detection Technique by using Network Packet Analysis and Machine Learning

Wen-Tsung Tsai[1], Shao-Ru Lin[2], Te-Min Liu[3], Chao-Lung Chou[4*]
[1,4]Department of Computer Science and Information Engineering, Chung Cheng Institute of Technology, National Defense University, [2]National Chung-Shan Institute of Science and Technology (NCSIST), [3]Network Traffic Packets Analysis Association, R.O.C
[1]wentsung.tsai@gmail.com、[2]oceankeep@gmail.com、[3]dmliu99999@gmail.com、[4]chaolung.chou@gmail.com

Abstract

In recent years, information security incidents about enterprises and government agencies being attacked by ransomware viruses have gradually appeared on the news. Hackers penetrate users' computers through social engineering or insidious methods and encrypt their files by using ransomware viruses. Suppose the victim is eager to restore the files to avoid stagnation of the organization's operations and damage to personal interests. In that case, the ransom payment may be made according to the method specified by the hacker. To reduce damage, gaining response time is the primary goal while attacked by such attacks. Therefore, a real-time dynamic analysis method is required to detect ransomware attacks.

Because of the abnormal behaviors of ransomware attacks in the network environment, this research proposes two indicators, that is, the ransom file (RF) and abnormal packets (AP), to detect whether computers are attacked by ransomware and use machine learning algorithms such as decision tree, sequential minimum optimization (SMO) and simple Logistic regression to classify different ransomware according to the two indicators. After 600 rounds of experiments, the results show an average classification accuracy rate of 99.25%, indicating that the proposed method can effectively detect and classify ransomware.

Keywords: Ransomware, Dynamic Analysis, Packet, Machine Learning

壹、前言

1.1 研究背景與動機

1988 年世界出現第一隻網路蠕蟲，在當時更癱瘓了幾千部電腦，可見早在電腦和網路萌芽之初，就受到不小的安全威脅。時至今日的物聯網時代，網路所帶來的經濟效益日與俱增，全球各國家之間的銀行往來、影音娛樂產業、購物平臺無一不依賴網路，企業與政府的機要文件也都會運用內部網路傳遞，可以說無論內外部網路上的資訊，都是相當重要的資產，而所有網路使用者所面臨的安全威脅，更不可同日而語。

企業與個人的電腦所存放的文件檔案中，不乏重要的資訊，同時很可能是貿易往來、公司營運、學術研究或產業研發等關鍵文檔，這也引起不肖人士覬覦，透駭侵手法竊取機敏資訊，而今年來運用加密技術的勒索病毒日益猖獗，駭客利用受害者倚賴重要文檔的心理，透過勒索病毒加密受害者重要檔案，以勒索要求支付贖金，才會給出加密金鑰；在未取得金鑰前，被加密的檔案通常難以解開，將導致業務無法推展而遭受更大的損害。

2021 年資安公司 Check Point 的年度報告指出，駭客攻擊手法中，高達 93%的社交工程攻擊，而且全球平均每 10 秒就有一起勒索病毒攻擊，加密勒索如此頻繁的原因是有利可圖及受害者耽心失去重要資訊的心理，也因此只要有受害者交付贖金，勒索攻擊就會持續進行[1]。我們無法控制惡意程式攻擊不發生，但確有機會在被攻擊的當下，保護機敏檔案及減少災損。

勒索病毒一旦感染主機，存放在磁碟裡的檔案將很快地被加密，如何減少損害，並避免從宿主主機攻擊內部網路電腦，必須即時偵測出網路異常行為，以爭取資安事件處置的緩衝時間。

1.2 研究目的

透過網路異常行為所產生的封包特徵值，在解析侵入的勒索病毒攻擊特性過程中，找出可偵測勒索病毒活動並分辨種類的指標數值。

1.3 論文架構

本論文共分五節：第一節前言，說明研究背景、動機及目的。第二節文獻探討，針對電腦病毒、勒索病毒種類、加密方法以及檢測方式等相關的背景知識加以闡述；第三節為本文所提出的方法；第四節為驗證與分析，介紹實驗環境、樣本與成果分析；第五章說明結論及未來研究方向。

貳、文獻探討

本節介紹近年重大勒索軟體及特性，並探討惡意程式分析技術，以奠定本研究基礎。

2.1 惡意程式分析技術

通常區分靜態分析與動態分析兩種，原理及比較概述如後：

1. 靜態分析：

靜態分析是對惡意程式進行反組譯程式編碼型態，從這些程式碼中找到可能執行的行為及命令，其中較特殊的程式碼片段，很有可能是特徵辨識用途之病毒碼，彙整之後放入防毒軟體資料庫中供使用者進行病毒防護。

靜態分析的方式主要是針對程式碼本身進行資料剖析、擷取病毒的模式、屬性和元件，並標記異常，對於已知的惡意程式，能夠快速且精準取得重要特徵，但是檢測到複雜編碼的程式病毒。

另外，靜態分析有賴於技術人員的能力與經驗除錯或反組譯器，如果攔截到未知的惡意程序碼，則，如：操作反組譯器、分析編碼結構 (脫殼) 等等能力。在靜態分析中，一般常見運用的方法有四項[2]：

- 防毒軟體掃描 (Antivirus Scanning)：

是最早以簽名檔進行惡意程式分析的方式，若發現惡意程式，此簽名檔會註記。此方法乃是透過研究並分析程式碼內容，找出其中唯一的程式碼，作為辨識特徵之病毒碼，搭配防毒軟體中之掃描檔案功能，就能自遭受感染的電腦中，找出有問題的惡意程式，是一款能快速準確檢測已知病毒的方法。不過，倘若簽名檔不在資料庫中，就無法檢測到未知與變種的惡意程式。現今，不同的掃毒軟體會使用不同的辨識方法，因此會同時使用多款不同的防病毒程序一起進行惡意程式偵測作業，以增加惡意程式的樣本檢測率，也正因為如此，VirusTotal、Jotti 和 VirSCAN 等多種掃毒網站[3][4][5]，會同時使用不同的掃毒方式進行檢測作業，以達成上述目的。

- 程式碼字串分析 (String Analysis)：

在目標檔案中，搜索特定的字串可能會發現惡意程式的訊息，例如搜索到「URL」文字，就可能會連結到惡意程式 C&C 中繼位址，如果是搜索到「Email」則可能連結到駭客信箱。此方式是屬於快速且有效果的方法。

- 雜湊函數 (Hashing)：

用雜湊函數來判別惡意程式是一項很特別的方法，只不過現行的雜湊函數檢測方式，僅能針對已知惡意程式進行比對檢測作業。

- 逆向編譯 (Reverse Compiling)：

逆向編譯檢測技術將惡意程式檔案拆解為作業系統執行檔結構 (Execute Structure) 與可執行程式碼 (Execute Code)，並依照惡意程式的可執行程式碼，對應 CPU 的 Machine

Code，進一步轉為可閱讀的組合語言碼 (ASM Code)，藉此分析惡意程式碼已使用的 CPU 功能 (例如暫存器) 和堆疊狀態、獲知程序或程式碼執行的呼叫順序。逆向編譯分析惡意程式碼時，可能可以提供更多惡意程序的相關線索，但需要深入了解 CPU 組合語言與作業系統運作架構。

2. 動態分析：

動態分析方法主要是在沙箱 (SandBox) 環境中，刻意執行惡意程式並從中觀察其行為，簡言之，動態分析的方法是一邊執行惡意程式，同時一邊進行分析[6]。動態分析的過程中，可以在虛擬機器中觸發可疑的檔案，並查看及分析觸發後行為，不需倚靠特徵碼來識別威脅，因此能夠識別與以往不同的威脅。

依前述靜態分析與動態分析的論述，綜整兩者特性如表 1 [7][8]。

表 1：靜態分析與動態分析比較表

項目	靜態分析	動態分析
原理分析	先將惡意程式進行反編譯，，並取得惡意程式可能執行的行為及命令等進行相關分析，研究人員將其中特殊之程式碼片段，指定為特徵辨識用途之病毒碼，彙整之後放入防毒軟體資料庫中供使用者進行病毒防護。	建立一個虛擬且與真實系統隔離的環境，並將惡意程式放入該虛擬環境中，觸發惡意程式，並在該環境中運作，除可避免惡意程式對實際系統造成影響外，同時可以詳盡地分析該惡意程式的所有行為。
檢測方式	(1) 防毒軟體掃描 (2) 程式碼字串分析 (3) 雜湊函數 (4) 逆向編譯	(1) 虛擬機 (2) 沙箱 (3) 網路封包分析
優點	(1) 病毒碼(Malicious Code)易於快速分析：通過檢測程式碼片段特徵，並將其與先前觀察到的病毒碼進行比對，可以快速進行靜態分析。 (2) 快速產生結果，不需要以實際執行勒索病毒。	(1) 唯一能零時差偵測威脅的方法。 (2) 不易混淆，可識別未知的病毒，並且可分析加密病毒碼。
缺點	(1) 病毒碼容易受到混淆。 (2) 無法識別未知病毒。 (3) 病毒碼被加密時，無法進行分析。 (4) 壓縮檔案導致可視性降低。	(1) 在分析病毒的環境時，容易識別病毒狀況，但過程費時。 (2) 分析的過程中，容易受到病毒攻擊。

2.2 近年常見勒索病毒

勒索病毒在 2017 年虛擬貨幣開始盛行之際，攻擊強度達到了高峰[9]，襲捲全球的 WannaCry 勒索病毒，就是駭客要求必須使用比特幣支付贖金；專家指出因虛擬貨幣匿名性及交易過程不易追蹤的特性，導致網路犯罪增加，相對地也使勒索病毒攻擊增加[10]。隨著時間推移，綜整近年常見的勒索病毒如表 2 [11]。

表 2：近年常見勒索病毒一覽表

項次	發現時間	種類	特性
1	2016	Petya (Windows)	以感染硬碟方式進行，受感染系統於下次啟動時加密檔案，完全阻止系統重啟，直至交付贖金為止。
2	2017	Bad Rabbit (Windows)	誘使使用者下載偽裝成 Adobe Flash 的更新程式，趁機加密電腦的檔案文件。
3	2017	WannaCry (Windows)	利用美國國家安全局 (NSA) 零日漏洞的永恆之藍，透過網路進行，並利用演算法惡意加密用戶檔案。
4	2019	Conti (Windows)	藉由木馬病毒遠端操控或 Windows 漏洞進行散播與操控，且使用 AES-256 演算法加密文件，每個二進位文件都是專為每個受害者門製作，密鑰皆不同，另外也以程式加密文件，讓使用者無法以命令和控制伺服器打開文件。
5	2019	Maze (Windows)	以 ChaCha20 和 RSA 兩種演算法加密文件，加密後會在每個文件名稱的尾末加上隨機 4~7 個字符，當所有文件加密完成後，也會修改桌面顯示已遭加密之文字，並發出聲音說明受害者電腦文件已遭加密。
6	2022	Quantum (Windows)	駭客利用電子郵件散佈金融木馬 (IcedID)，之後再於受害者組織網域部署名為「量子」(Quantum) 的勒索病毒來勒索用戶，整個加密勒索過程不到 4 小時，是歷來速度最快的勒索病毒。

2.3 勒索病毒加密方式

勒索病毒對受害者電腦加密方式，一是螢幕加密 (Screen-Lockers; Locky)，另一種則是檔案加密 (File-Lockers)，兩種加密方式說明如下[12]：

1. 螢幕加密：

螢幕加密的目的是透過鎖定電腦桌面，強迫使用者無法操作，通常在登入電腦時就會發現被勒索。如 2010 年的 WinLock 病毒，就是以色情圖片遮擋使用者的電腦螢幕，並要求受害者繳付 10 美元簡訊費後才會接收解鎖密碼，攻擊畫面如圖 1 [13] [14]。

圖 1：WinLock 病毒攻擊成功畫面[14]

此種方式只是鎖住電腦系統，通常可經由重新啟動或是安全模式下恢復正常狀態，危害程度較小。

2. 檔案加密：

對電腦內的文件進行加密，讓使用者無法開啟、存取文件，直到使用者支付贖金後才能解開文件，近年出現的勒索病毒多屬於這一類型。經研究統計，Microsoft Office 文件檔案通常含有重要的業務資訊，因此成為多數勒索病毒的攻擊目標，另外，勒索病毒對檔案加密通常三種模式[15]：

(1) 對檔案加密，但不重新命名或更改檔案存放位置。
(2) 對檔案加密並重新命名，但不更改檔案存放位置。
(3) 對檔案加密並重新命名，且更改檔案存放位置。

2.4 勒索病毒攻擊模式

勒索攻擊大致可分為三個階段：

1. 入侵階段 (Delivery Stage)：

通過假冒成合法的電子郵件等方法，誘使受害者點擊連結並下載病毒。

2. 破壞階段 (Sabotage Stage)：

入侵階段及破壞階段，都可能產生異常的網路封包，如 DNS、HTTP/HTTPS、SMTP、FTP、RDP 或 SMB 等通訊協定異常的封包。根據勒索家族的差異，進行網路封包分析，可以找到部分共同行為表現[16]。

3.勒索階段 (Extortion Stage)：

這個階段使用者電腦會出現極明顯的三項狀態：

- 讀寫磁碟的位元組不斷增加。
- CPU使用率遽增。
- 對網路磁碟機 (分享目錄、磁碟) 表現出特殊網路行為，包括探測、異常權限存取、異常檔案名稱寫入、大量網路讀取寫入封包 (其主要通訊協定為SMB)。

2.5 運用網路封包分析惡意程式的相關研究

封包分析是追溯網路行為主要的方式，獲得的封包數據越詳細，就可以越了解網路狀況。Usha Banerjee 在 2010 年的研究中，利用 Wireshark 進行惡意程式入侵偵測，以 ACL (Access Control List) 過濾封包資料，查得惡意程式在封包裡的字串符號、使用的通訊協定項目[17]。

2013 年 Pallavi 等學者使用 Network Interface Card 擷取封包工具，監聽駭客的攻擊手法，了解各種資訊流向、監控與分析，藉以監測可疑的攻擊[18]。2017 年 Aishwarya 等學者認為所有的電腦資訊都是通過 TCP 進行，因此利用 Wireshark 進行封包流量分析，並對不同的時序、封包往來時間、資訊量等多種參數進行分析[19]；2020 年 Juraj 等學者以封包的特性為基礎，開發視覺化工具進行數據分析，以了解惡意訊息傳遞的過程[20]。

以上相關研究顯示深入分析網路封包，可以發掘惡意程式在網路活動時，特有的異常行為，因此本研究依循學者先進的作法，分析勒索病毒發動攻擊時的封包特徵，進而達成依特徵值偵測並分類勒索病毒。

參、本研究提出之方法

當勒索病毒藉由宿主電腦，在區域網路中進行橫向擴散時，因其目的是加密檔案或磁碟，所以在網路中會不斷地搜索網路硬碟，必然會產生網路活動的封包，如果加上再設定網路、檔案或磁碟的存取權限，將會使勒索病毒產生不同的反應，只要將這些特殊反應的網路封包篩選出來並加以分析、歸納後，應不難判斷及預測出，有某種網路行為相以性的程式，是否為勒索病毒，甚至可以為其分類，屬於哪一種勒索病毒。

秉持上述的構想，找出判定勒索病毒的特性，以作為篩選封包的條件，即為本研究核心工作。勒索病毒最一般性的特徵，就是會夾帶文字檔案，以告知受害者付贖條件和方式，所以必須將此特性做為篩選封包的條件。另一方面，為使病毒在網路環境下，產生其他合法程式不會有的不尋常行為，則透過網路芳鄰開啟共用資料夾，且外部電腦僅被允許讀取權限；當勒索病毒頻繁地對共用資料夾加密時，將產生大量網路服務存取被拒 (Access Denied) 的訊息，如此，藉由不正常的網路服務被拒的資訊，作為篩選封包的條件。

獲得上述二項篩選封包條件後，必須量化封包才能作為偵測勒索病毒的指標。第一項篩選條件是針對勒索病毒的隨附文件，本研究稱之為勒索文件 (Ransom File, RF)，依條件所獲封包定義為勒索文件封包，累計數量以 P_{RF} 表示；第二項篩選條件是因網路攻擊產生的異常封包 (Abnormal Packets)，累計數量以 P_{AP} 表示。基於 SMB 通訊協定，勒索病毒才能夠在網路上存取共用資料夾，因此，本研究提以 P_{RF} 及 P_{AP} 各別在所有 SMB 封包中所佔比例，定為偵測勒索病毒的兩項指標，並以 S_{RF} 與 S_{AP} 表示。最後，依 S_{RF} 及

S_{AP} 數值做為資料集，運用機器學習方法建立分類器，以預測網路中含有何種勒索病毒正在活動，研究方法流程圖 2。

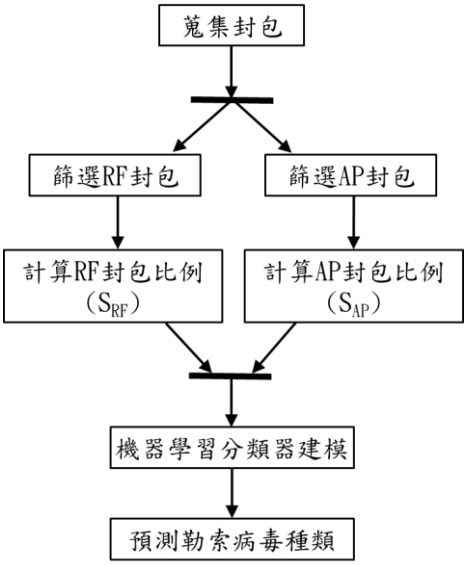

圖 2：本研究所提方法之流程

3.1 蒐集封包

本研究使用微軟 Windows 系列作業系統為實驗環境，並開啟網路芳鄰功能，以網路側錄軟體—Wireshark，擷取 SMB 通訊協定網路封包。

3.2 篩選特徵封包

1. 勒索文件封包數 (P_{RF})：

定義為某特定攻擊行為中，在一定時間內，所累積勒索文件的網路封包數量，如式(1)，k 為不同病毒產生的勒索文件類別，t 為時間區段(範圍 1 到 n)。

$$P_{RF} = \sum_{t=1}^{n} P_t^k \tag{1}$$

2. 異常封包數 (P_{AP})：

定義為某特定攻擊行為中，在一定時間內，所累積相對應的異常網路封包數量，如式(2)，a 為不同病毒產生的異常網路活動類型，t 為表時間區段(範圍 1 到 n)。

$$P_{AP} = \sum_{t=1}^{n} P_t^a \tag{2}$$

3.3 計算 S_{RF} 及 S_{AP} 指標

利用式(3)及式(4)，計算 S_{RF} 及 S_{AP} 指標，分別代表 RF 及 AP 封包相對於所有 SMB 通訊協定封包中的比例，其中$\sum P_{SMB}$代表 SMB 協定網路封包總數。

$$S_{RF} = \frac{P_{RF}}{\sum P_{SMB}} \tag{3}$$

$$S_{AP} = \frac{P_{AP}}{\sum P_{SMB}} \tag{4}$$

3.4 機器學習分類器

機器學習依資料集訓練模式，經常分為監督式機器學習 (Supervised Learning) 及非監督式機器學習 (Unsupervised Learning) 兩大類型[21]。

監督式學習特點是訓練資料集必須先經過標記化 (Labeled)，以在機器學習輸出時判斷誤差，標記過的資料就像標準答案，電腦在學習過程中一邊對比誤差，一邊修正以達到更精準預測，這使監督式學習有準確率高的優點；但是資料前處理需大量以人工作業標記資料，相當繁瑣耗時，當範圍擴大或資訊量增加，便難以對資料標記所有特徵。常見的監督式學習演算法有線性迴歸、多項式迴歸、決策樹、簡單邏輯迴歸、SVM、簡單貝氏及 KNN 等。

非監督式與監督式學習不同處在於，不需事先對資料作標註，是依資料關聯性進行歸類並找出潛在規則而形成分群 (Clustering)，也因如此，其特點是聚集相似度高的資料，而不是精準預測，常見的非監督式演算法有聚合式階層分群、階層式分群、K-Means、DBSCAN、及主成份分析 (PCA) 等。

為明確地對 S_{RF} 及 S_{AP} 兩項指標作分類，以驗證偵測勒索病毒有效性，本研究採用監督式學習，並運用決策樹、循序最小優化 (Sequential Minimal Optimization, SMO) 及簡單邏輯迴歸等三項演算法，預測異常封包為何種類型的勒索病毒，三項演算法概述如後：

1. 決策樹 (Decision Tree)：

一種決策分析的方法，依資料之間的關聯性來產生預測，運用樹的階層概念，分類過程如同樹狀結構，從最上層的根節點開始，在每個節點為一項特徵，經由分支去做出決策，再繼續往下一層分類推進，分析示意如圖 4，其中根節點為資料集全部樣本，內部節點是對應「特徵」屬性測試 (即分類測試)，葉節點即為決策、分類的結果[22]。

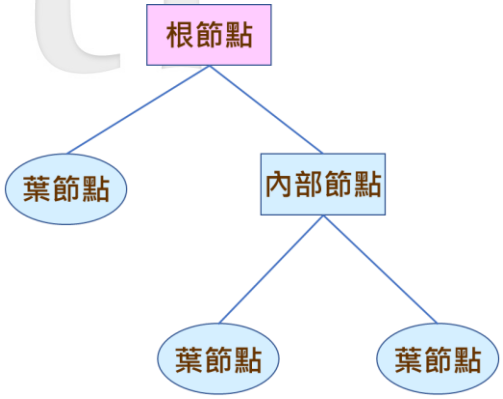

圖 3：決策樹分析示意圖

決策樹有 ID3、C4.5、C5.0、CHAID 及 CART 等多個演算法，本研究採用修進自 ID3 的 C4.5 演算法，C4.5 不使用資訊增益 (Information Gain)，而是用「資訊增益比」(Gain Ratio) 作為特徵的選擇依據，因資訊增益比考量整體效益，更好地決定該採取何項特徵作為決策條件。

2. 循序最小優化演算法 (SMO)：

SMO 演算法是由微軟研究院 (Microsoft Research) 的研究員 John Platt 所提出[23]，主要是對在訓練支援向量機 (Support Vector Machine, SVM) 時進行優化問題的改善；該演算法的核心思想是將原問題分解成多個小問題，並分別進行優化求解，SMO 每次只優化兩個變數，將其他的變數都視為常數。也是說 SMO 演算法是為了解決 SVM 中的優化目標函數，尤其是在線性的 SVM 和資料較少的情形下，效能更佳。在 SMO 提出之前，SVM 訓練方法必須使用複雜的方法，並耗費龐大的運算資源。

3. 簡單邏輯迴歸演算法 (Simple Logistic Regression)：

主要在探討依變數與自變數之間的關係，是基於線性邏輯迴歸模型的分類器。一般的迴歸分析，限制依變數必須是連續型變數，當要分析的變數不是連續型態則無法進行，因此，當依變數為類別型變數時，需改為使用簡單邏輯迴歸。實務運用上，簡單邏輯迴歸主要是用於資料分類，而線性迴歸主要是預測資料[24]。

肆、驗證與分析

4.1 實驗環境

本研究實驗使用 VirtualBox 虛擬機平臺，並建置微軟系列作業系統分別為 WindowXP、Window 8 以及 Window10 等三部主機，主機之間透過 Windows 的 SMB 功能，建立資料夾分享，藉以觀察勒索病毒在相同網段下，對於三台不同作業系統的電腦，是否產生異常通訊行為；如圖 4，A、B、C 三部主機 (虛擬機) 分別安裝 Windows 10、

Windows XP 及 Windows 8 作業系統，三部主機形成同網段的區域網路，並由 B 主機開啟網路芳鄰功能，共享資料夾供對 A、C 主機作存取。

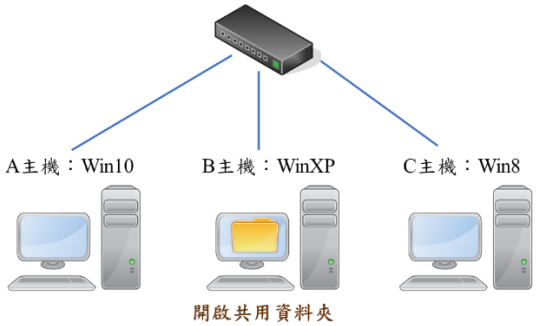

圖 4：實驗環境示意圖

4.2 勒索病毒樣本選用

1. Maze：

最早出現於 2019 年 5 月，又稱 ChaCha，利用漏洞工具並通過網頁掛馬方式、偽裝成合法的虛擬貨幣交換應用程式或夾帶於垃圾郵件進行散播。不同的 Maze 變種病毒會產生不同的勒索文件。當遭受 Maze 攻擊，會循環播放檔案文件被加密的錄音檔，告知受害人電腦已經遭勒索病毒感染，如果受害者拒絕支付贖金，駭客組織會威脅洩露在加密前竊取的資料，受其攻擊後出現勒索訊息如圖 5 [25]。

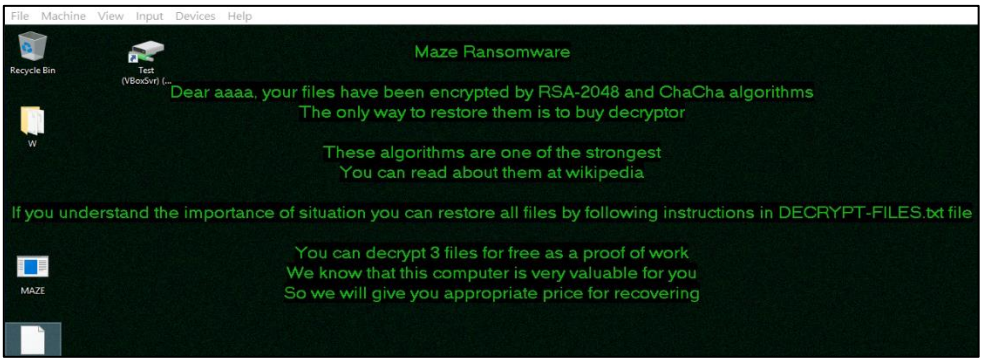

圖 5：Maze 勒索病毒警示受害者訊息

2. Conti：

2019 年 Conti 發跡於俄羅斯，具雙重勒索特性，在勒索加密前，會先下載機密資料的明文，查找機密文件中企業財務狀況，且對目標植入勒索病毒前，會先上傳備份工具，並取得網域管理者權限建立通訊連線，再偵察及竊取網域內的財務資料，作為受害者拒付贖金的勒索籌碼。

2022 年 1 月我國某科技大廠即遭受 Conti 攻擊，駭客要求支付 1500 萬美元 (約新台幣 4.12 億元) 的巨額贖金換取解密金鑰[26]，其攻擊畫面如圖 6。

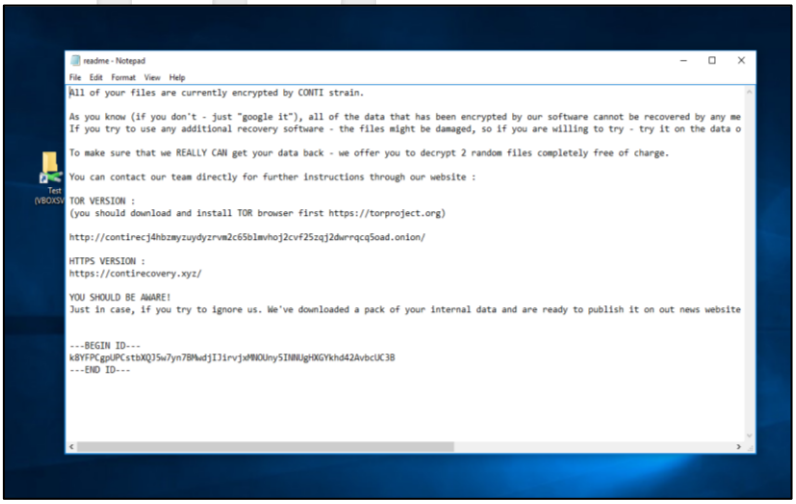

圖 6：Conti 勒索病毒攻擊成功畫面

3. WannaCry：

WannaCry 在 2017 年 5 月間襲捲全球，超過 150 個國家在短短數小時內遭受侵害，這場史詩級的網路災難肇因於 WannaCry 有主動感染的能力，除透過釣魚郵件、惡意連結或惡意文件散播，只要發現具有 SMB 漏洞的電腦，就結合惡意程式—永恆之藍入侵被害者電腦，一方面加密的檔案，另一方面繼續入侵其他電腦。具備這樣的能力，使得 WannaCry 不再單單只是勒索病毒，可以稱為勒索蠕蟲，也因此能在全世界疾速擴散的關鍵。

WannaCry 使用 RSA-2048 加密技術，並針對 180 種檔案格進行加密，而且檔案加密後副檔名修改為“.WNCRY”。受害者必須以比特幣支付贖金，才可取回解密金鑰，超過 7 天未付贖金，解密金鑰就會被銷毀，原來的檔案極可能等同遺失或刪除[27][28]，攻擊成功畫面如圖 7。

圖 7：WannaCry 勒索病毒攻擊成功畫面

4.3 資料集蒐整方式

1. 攻擊模擬場景：

實驗環境如 4.1 小節所述，攻擊模擬場景為 A 主機感染 Maze、Conti 及 WannaCry 等三項勒索病毒，並透過區域網路對其他二部主機攻擊；B 主機開啟共用資料夾分享給 A、C 主機存取，C 主機則不分享資料夾，以對照勒索病毒對電腦開啟與不開啟共用資料夾的反應，模擬場景如圖 8。

2. 網路封包側錄：

本實驗使用開源軟體--Wireshark 側錄攻擊場景中 A、B、C 三部主機所產生封包，A 主機執行 Maze 勒索病毒感染後，並確認病毒透過網路攻擊 B、C 主機後，立即還原虛擬機至初始未遭受感染、攻擊狀態，如此重複 200 次，以蒐集足量的網路封包進行分析。Conti 及 WannaCry 二項勒索病毒也使用此方式重複 200 次 (三種病毒共 600 次)。

圖 8：攻擊模擬場景示意圖

4.4 封包篩選結果與分析

因 Windows 系統中的網路芳鄰上共用資料夾，是藉由 SMB 通訊協定來達到資訊通聯的目的，所以蒐集 SMB 的封包，並產生 S_{RF} 及 S_{AP} 兩項指標為本研究的重點，依 3.2 小節定義，需要蒐集 SMB 封包總數、篩選出 RF 及 AP 封包。

使用 Wireshark 篩選 AP 封包的條件可設為“ smb.nt_status==0xc0000022 or smb2.nt_status==0xc0000022 ”，而每種勒索文件都略有不同，篩選 RF 封包條件及計算平均 S_{AP}、S_{RF} 指標結果分述如後：

1. Maze 勒索病毒：

篩選 RF 封包條件：smb.file contains “ DECRYPT-FILES.txt ” or smb2.filename contains

“DECRYPT-FILES.txt”。A、B、C 三部主機平均 S_{AP} 分別為 47.42%、47.87%及 0%，平均 S_{RF} 分別為 0.49%、0.50%及 0%。

2. Conti 勒索病毒：

篩選 RF 封包條件：smb.file contains “ CONTI_README.txt ” or smb2.filename contains “ CONTI_README.txt ”。A、B、C 三部主機平均 S_{AP} 分別為 0.10%、0.10%及 0%，平均 S_{RF} 分別為 0.20%、0.21%及 0%。

3. WannaCry 勒索病毒：

篩選 RF 封包條件：smb.file contains “ @Please_Read_Me@.txt ” or smb2.filename contains “ @Please_Read_Me@.txt ”。A、B、C 三部主機平均 S_{AP} 分別為 0.10%、0.10%及 0%，平均 S_{RF} 分別為 0.33%、0.36%及 0%。

實驗結果綜整如表 3，由表中的數據可發現，勒索病毒在區網裡的攻擊目標，只有開啟共用資料夾的 A 主機 (宿主) 與 B 主機產生 AP、RF 封包，而區網內的 C 主機則沒有受到勒索病毒攻擊的影響，也就是說電腦未遭受病毒攻擊，S_{AP} 與 S_{RF} 應為 0%，而遭受攻擊的電腦，這兩項指標的數值均大於 0%；另一方面，如圖 9 所示，重複數百次蒐集網路封包的實作後，三種勒索病毒的 AP 及 RF 在 SMB 封包中所佔比例，經圖示化後，明顯形成 3 個群集。依此實驗結果呈現的圖表，可推論本研究所提出之 S_{AP} 與 S_{RF} 指標可有效針對不同類別之勒索病毒呈現差異數值。

表 3：S_{AP} 與 S_{RF} 計算結果

主機別 / 指標 / 病毒名稱	A 主機		B 主機		C 主機	
	S_{AP}	**S_{RF}**	**S_{AP}**	**S_{RF}**	**S_{AP}**	**S_{RF}**
Maze	47.42%	0.49%	47.87%	0.50%	0%	0%
Conti	9.77%	0.20%	9.72%	0.21%	0%	0%
WannaCry	0.10%	0.33%	0.10%	0.36%	0%	0%

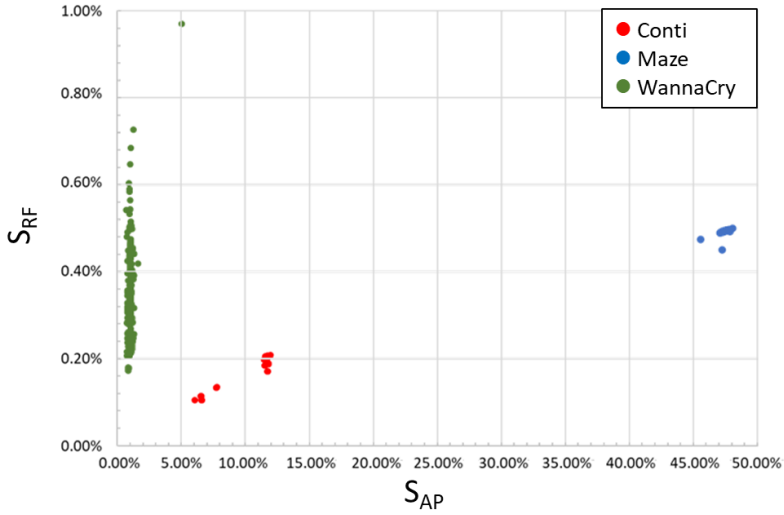

圖 9：三種勒索病毒 S_{AP} 與 S_{RF} 特徵值分布

4.5 機器學習分類訓練及測試

本研究使用開源軟體—Weka，針對實驗所得 S_{AP} 及 S_{RF} 兩項指標實作機器學習分類器的訓練與測試，以驗證這兩項數值可作為偵測勒索病的指標，並以決策樹、循序最小優化演算及簡單邏輯迴歸三種方分類法實作訓練與測試。

分類所用資料集即經實驗所得 1,200 筆 S_{AP} 及 S_{RF} 數據 (A、B 主機各遭受三種病毒攻擊，每種病毒攻擊 200 次，合計 1,200 筆資料)，並基於實驗結果資料數量，以兩種資料集比例進行機器學習：第一，訓練及測試資料集各為 50% (各 600 筆)，第二，訓練及測試資料集分為 70% (840 筆)、30% (360 筆)。

1. 決策樹分類法：
   Weka 重要參數設定，如圖 10。
   - 信任度 (Confidence Factor)：用於決策樹建立後修剪的參數依據，數值越小代表修剪越多，本研究設定為0.25。
   - 折數 (Numfold)：用於減少錯誤修剪的資料量，本研究設定為3。
   - 最少物件數量 (MinNumObj)：控制每個葉子的最小資料數，本研究設定為2。
   - 演算法 (Calibrator)：C4.5演算法(Weka名稱為“J48”)。
2. 循序最小優化演算分類法 (SMO)：
   Weka 重要參數設定，如圖 11。
   - 參數 (C)：複雜參數，本研究設定為1.0。
   - 演算法 (Calibrator)：使用Logistic進行。
   - 捨入誤差 (Epsilon)：1.0E-12。

使用的核心 (Kernel)：Polykernel。

3. 簡單邏輯迴歸分類法：

Weka 重要參數設定，如圖 12。

- 啟發式停止 (HeuristicStop)：如果此數值大於0，再反覆運算中，如果沒有達到錯誤的新最小值，則停止LogitBoost演算法，本研究設定為50。
- LogitBoost最大反覆運算次數 (MaxBoostingIterations)：本研究設定為500。

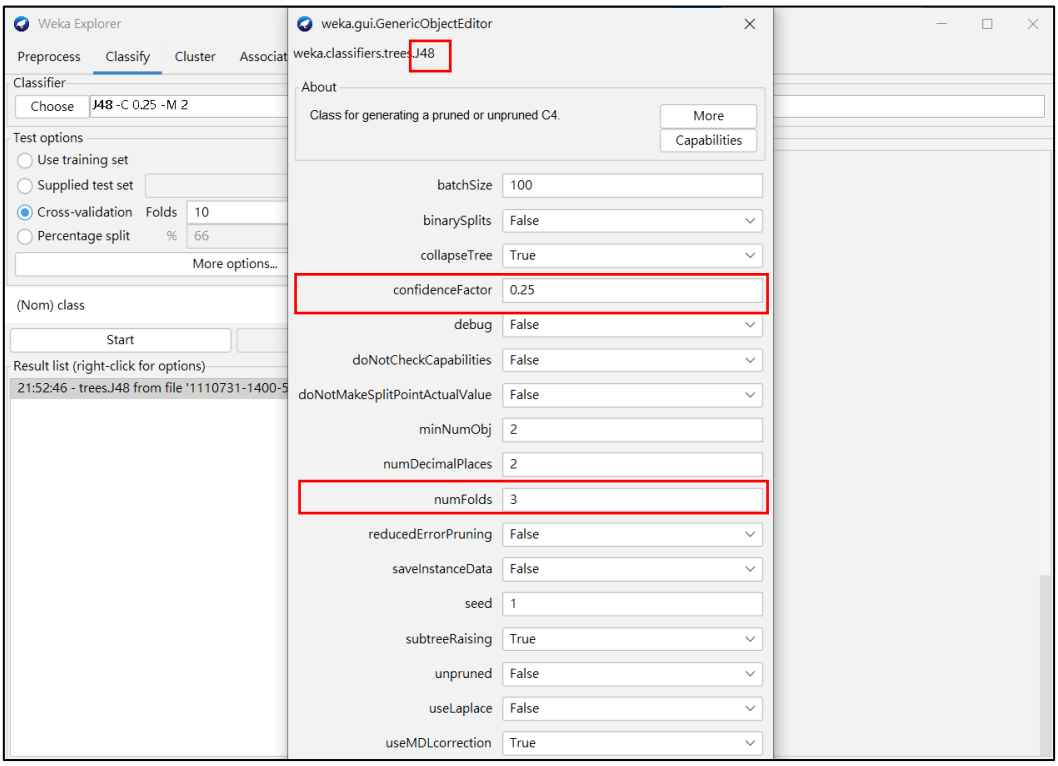

圖 10：使用 Weka 實作決策樹重要參數設定

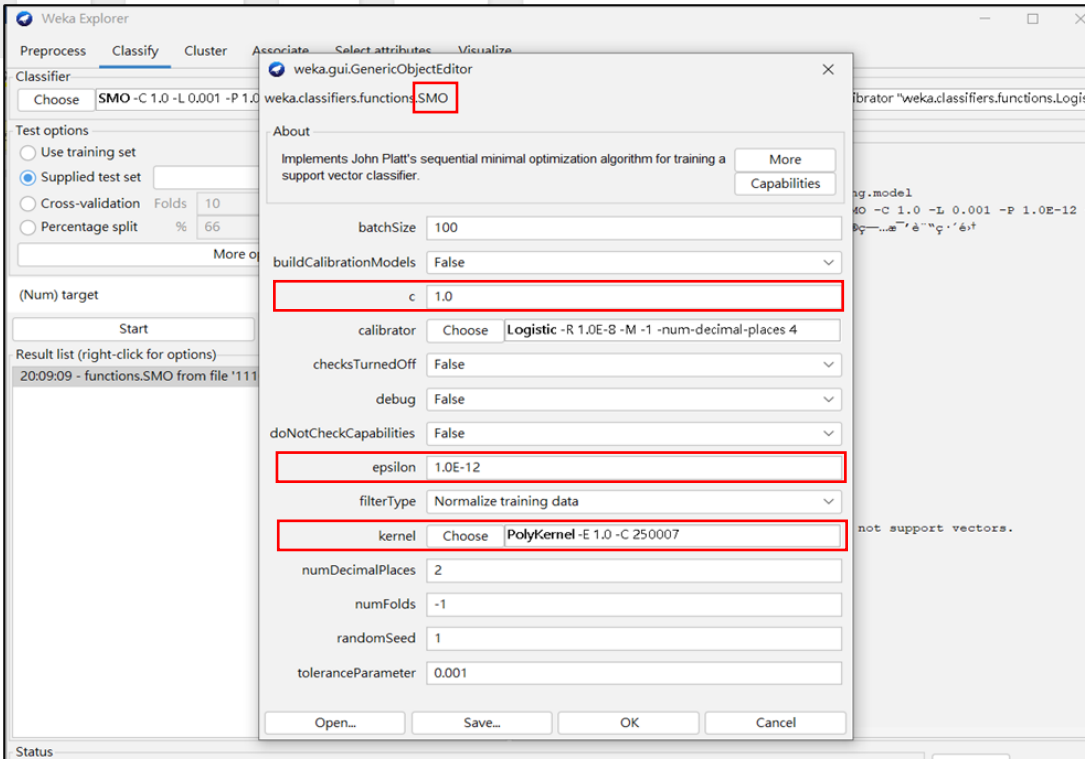

圖 11：使用 Weka 實作序列最小優化 (SMO) 重要參數設定

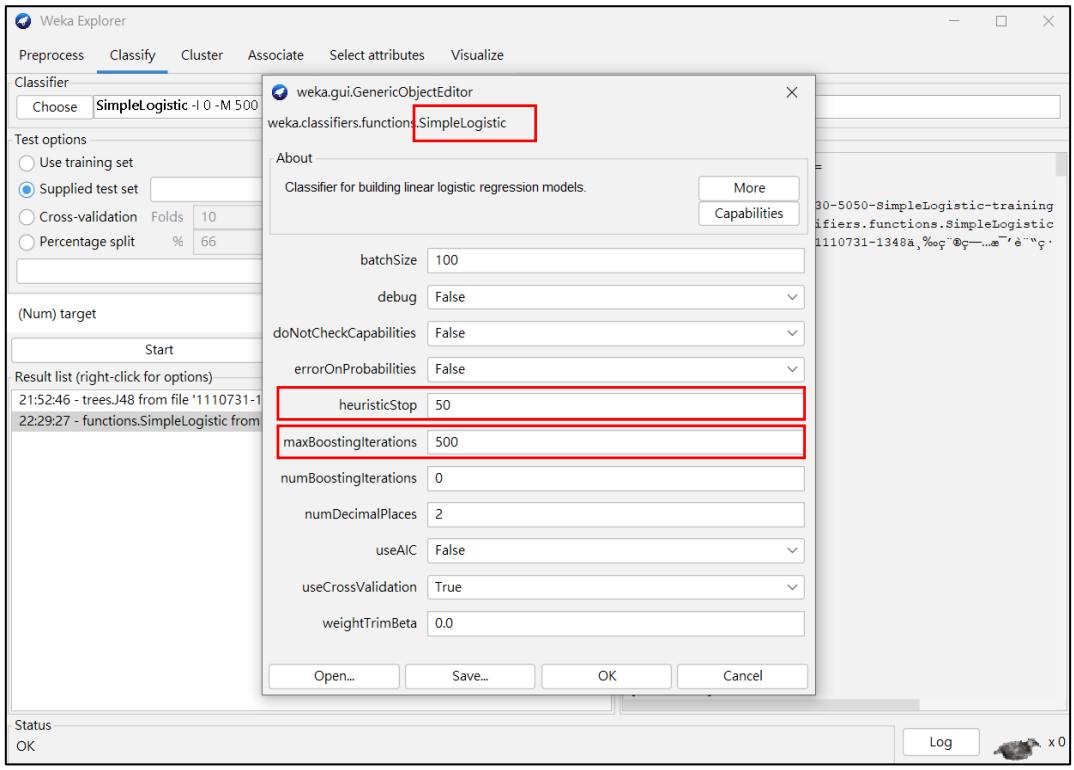

圖 12：使用 Weka 實作簡單邏輯迴歸重要參數設定

本研究以決策樹分類法、序列最小優化演算法及簡單邏輯迴歸分類法進行分類，其

準確率計算公式如式 5 所示，其中 $n_{samples}$ 為病毒樣本數量、$\hat{y}$為實際病毒類別、y_i則代表演算法分類病毒類別。

$$accuracy(y, \hat{y}) = \sum_{i=0}^{n_{samples}-1} (\hat{y} = y_i) \quad (5)$$

經式 5 計算，訓練資料集與測試資料集比例各為 50%時，準確率均在 95%以上；訓練集與測試集分別為 70%、30%，準確率更可提高至 99%以上，統計如表 4。由此可知本研究以 S_{AP} 與 S_{RF} 兩項偵測指標進行機器學習，其可有效分類出不同的勒索病毒種類。

表 4：分類準確率統計表

資料集比例 / 分類法	**訓練：50%，測試：50%**	**訓練：70%，測試：30%**
決策樹	98.00%	99.40%
循序最小優化	97.84%	99.17%
簡單邏輯迴歸	96.33%	99.17%
平均準確率	**97.39%**	**99.25%**

伍、結論

本研究提出 S_{RF} 及 S_{AP} 兩項特徵值，實驗結果驗證可作為偵測勒索病毒的指標 (未受攻擊及感染的電腦，S_{RF} 及 S_{AP} 兩項數值均為 0)，並運用機器學習中的決策樹、循序最小優化 (SMO) 及簡單邏輯迴歸等三項分類演算法，發掘這兩項指的潛在規則，也就是不同的勒索病毒所衍生出來的 S_{RF} 及 S_{AP}，會成對地落在某個數值範內，也因此可以利用這個特性對勒索病毒做分類。

得到這兩項富含意義的指標性特徵值，在未來的研究方向，除可導入資通安全防護管理中心 (Security Operation Center, SOC) 作為安全資訊事件管理系統 (Security Information and Event Management , SIEM)，對於惡意程式攻擊事件警示與回報的閾值 (Threshold)；另應可結合入侵預防系統 (Intrusion Prevention System, IPS) 或端點防護系統，即時掌握企業、組織內部及末端使用者所受威脅，以即時掌控惡意攻擊動態及減少災損。

[誌謝]

本研究由國家科學及技術委員會計畫支持，計畫編號 111-2221-E-606-013。

參考文獻

[1] Check Point Software Technologies, https://pages.checkpoint.com/cyber-security-report-2021.html (2021/12/10).

[2] A. Omer and S. Refik, "Investigation of possibilities to detect malware using existing tools," *IEEE/ACS 14th International Conference on Computer Systems and Applications,* 2017.

[3] VirusTotal, https://www.virustotal.com/gui/home/upload (2022/11/28).

[4] Jotti, https://virusscan.jotti.org (2022/11/28).

[5] Virscan, http://r.virscan.org (2022/11/28).

[6] 台灣電腦網路危機處理暨協調中心, https://www.twcert.org.tw/tw/cp-14-4502-000a2-1.html. (2022/11/28).

[7] 林韶如，"基於網路異常行為與機器學習技術之勒索病毒檢測"，*碩士論文*，國防大學理工學院資訊工程學系，2022。

[8] G. Ekta, B. Divya and S. Sanjeev, "Malware analysis and classification: a survey," *Journal of Information Security*, vol.5, no.2, pp.56-64, 2014.

[9] O. Philip, S. Sakir and C. Domhnall, "Evolution of ransomware," *IET Journal*, vol. 7, no.5, pp.321-327, 2018.

[10] S. Veronika, A. Gabor and D. Akos, "Introduction of the ARDS—anti-ransomware defense System model—based on the systematic review of worldwide ransomware attacks," *Applied Science*, vol. 11, no.13, 2021.

[11] H.C. Lin, P. Wang and W.Q. Hong, "Using signature analyses to construct an ontological model of ransomware," *Communications of the CCISA*, vol. 25, no.2, pp.37-58, 2019.

[12] C.V. Bijitha, S. Rohit and H.V. Nath, *Secure Knowledge Management in Artificial Intelligence Era*, Springer, Singapore, pp.55-68, 2020.

[13] D. Sgandurra, L. Muñoz-González, R. Mohsen and E.C. Lupu, "Automated dynamic analysis of ransomware: benefits, limitations and use for detection," https://arxiv.org/abs/1609.03020 (2016/9/10).

[14] The Winlock case – I'm taking bets! ttps://securelist.com/the-winlock-case-im-taking-bets/29623 (2022/11/28).

[15] S.H. Kok, A. Azween, N.Z. Jhanjhi and M. Supramaniam, "Ransomware, threat and

detection techniques: a review," *International Journal of Computer Science and Network Security*, vol. 19, no.2, pp.136-146, 2019.

[16] S. Nolen, C. Henry, T. Patrick and K.R.B. Butler, "CryptoLock (and drop it): stopping ransomware attacks on user data," *International Conference on Distributed Computing Systems*, 2016.

[17] S.H. Kok, A. Abdullah and N.Z. Jhanjhi, "Early detection of crypto-ransomware using pre-encryption detection algorithm," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 5, pp.1984-1999, 2020.

[18] A. Pallavi and S.Vishal, "Network monitoring and analysis by packet sniffing method," *International Journal of Engineering Trends and Technology*, vol. 4, no. 5, pp.2133-2135, 2013.

[19] B. Aishwarya, G. Samala, T.K. Koirala and I.M. Ruhul, "Packet sniffing and network traffic analysis using TCP-a new approach," *Advances in Electronics, Communication and Computing*, pp.273-280, 2018.

[20] U. Juraj, "Visual analysis of network packet capture files," Master's Thesis, Masaryk University, Czech Republic, 2020.

[21] K. Herleen and G. Naveen, *Innovative Data Communication Technologies and Application*, Springer, U.S.A., pp.266-275, 2020.

[22] H. Hans and P.H. Swain, "The decision tree classifier: design and potential," *IEEE Transactions on Geoscience Electronics*, vol. 15, no. 3, pp.142-147, 1977.

[23] J.C. Platt, "Sequential minimal optimization: a fast algorithm for training support vector machines," Microsoft Research Technical Report MSR-TR-98-14, U.S.A., pp.1-21, 1988.

[24] A.S. Dominguez, P. Benitez and R.A.R. Gonzalez, "Logistic regression models," *Allergol Immunopathol*, vol. 39, no. 5, pp.295-305, 2011.

[25] M.K. Marisa and P. Alexandra, "Current ransomware threats," Carnegie Mellon University, Technical Report, AD1110335, pp.1-84, 2020.

[26] Etoday, https://finance.ettoday.net/news/2179744#ixzz7Q2YclPLT (2022/11/28).

[27] A. Maxat, V.G. Vassilakis and M.D. Logothetis, "WannaCry ransomware: analysis of infection, persistence, recovery prevention and propagation mechanisms," *Journal of Telecommunications and Information Technology*, vol. 1, pp.113-124, 2019.

[28] Q. Chen and R.A. Bridges, "Automated behavioral analysis of malware- a case study of WannaCry ransomware," *International Conference on Computer Communication and Informatics*, 2017.

[作者簡介]

蔡文淙，國防大學國防科學研究所資工組博士生，研究方向為資訊安全及深度學習。
林韶如，國防大學理工學院網路安全在職專班碩士，現任職於國家中山科學研究院，研究方向為資訊安全及網路封包分析。
劉得民，中華民國網路封包分析協會，現為中華民國網路封包分析協會理事長，研究領域為網路安全、惡意程式分析及網路封包分析等。
周兆龍，國防大學國防科學研究所博士，現為國防大學理工學院資訊工程學系副教授，研究領域為資訊安全、深度學習及生物辨識等。