Communications of the CCISA
Vol. 26 No. 2 May. 2020

LeNet-5 卷積神經網路應用於勒索病毒分類

王平[1*]、洪維謙[1]、蔡東霖[1]、周明勝[1]
[1]崑山科技大學資訊管理學系
pingwang@mail.ksu.edu.tw, yamn18345@gmail.com, bryant84920@gmail.com, s7920378@gmail.com

摘要

近年來駭客透過不當下載進而安裝勒索病毒，綁架組織重要檔案，進行勒索金錢或比特幣，尤其鎖定工業控制系統、商業銀行、醫療機構與上市櫃公司，造成人心惶惶並增加企業資訊安全管理的高風險！故本研究針對近期發生的勒索病毒(Ransomware)威脅，透過沙盒及正規化概念分析法建立勒索病毒之行為特徵矩陣以提供模式預訓練(pre-training)，再透過深度學習網路(Deep Learning Networks)之 LeNet-5 卷積神經網路(Convolutional Neural Networks, CNNs)進行病毒行為的學習及特徵影像識別。實驗結果證明病毒之行為特徵矩陣能明確定義病毒與攻擊行為間之關聯，透過知識本體抽象資料模型可作為勒索病毒分類(classification) 與變種鑑定的參考依據，並將其轉化為規則可應用於再生能源預測平台之病毒即時偵測，提高偵測的精確度並降低誤判率。

關鍵詞：勒索病毒、病毒分類、卷積神經網路、LeNet-5、行為特徵

* 通訊作者 (Corresponding author.)

Regular Paper

Communications of the CCISA
Vol. 26 No. 2 May. 2020

Ransomware Classification Using LeNet-5 Convolutional Neural Networks

Ping Wang[1*], Wei-Qian, Hong[1], Dong-Lin Tsai[1], Ming-Sheng Jhou[1]
[1]Department of Information Management, Kun Shan University
pingwang@mail.ksu.edu.tw, yamn18345@gmail.com,
bryant84920@gmail.com, s7920378@gmail.com

Abstract

Recently, the ransomware were installed thru the use of malicious links and downloads, that kidnapped important files of organizations for money blackmail or bitcoins, especially focused on commercial banks and medical services, and public companies. Consequently, it raised a high crisis of information security management for corporates. Accordingly, the present study proposes a formal concept analysis-based security management system for Ransomware detection with malware sandbox analysis platform by analyzing the bahivoral features of malware. Then, using LeNet-5 Convolutional Neural Networks to learn the behavior of the ransomware classes for classify the pattern by using behavior characteristic matrix of the ransomware. Experimental data show that our model is capable of performing the missions including of i) explicitly identifying the mapping relations between Ransomware classes and their behavioral features, ii) As a basis of detection rules for network intrusion detection to classify the Ransomware families and their variations, and (iii) assist manager detect the malicious intrusion or illegal downloads for Ransomware from cyber threats with high accuracy and low false rate.

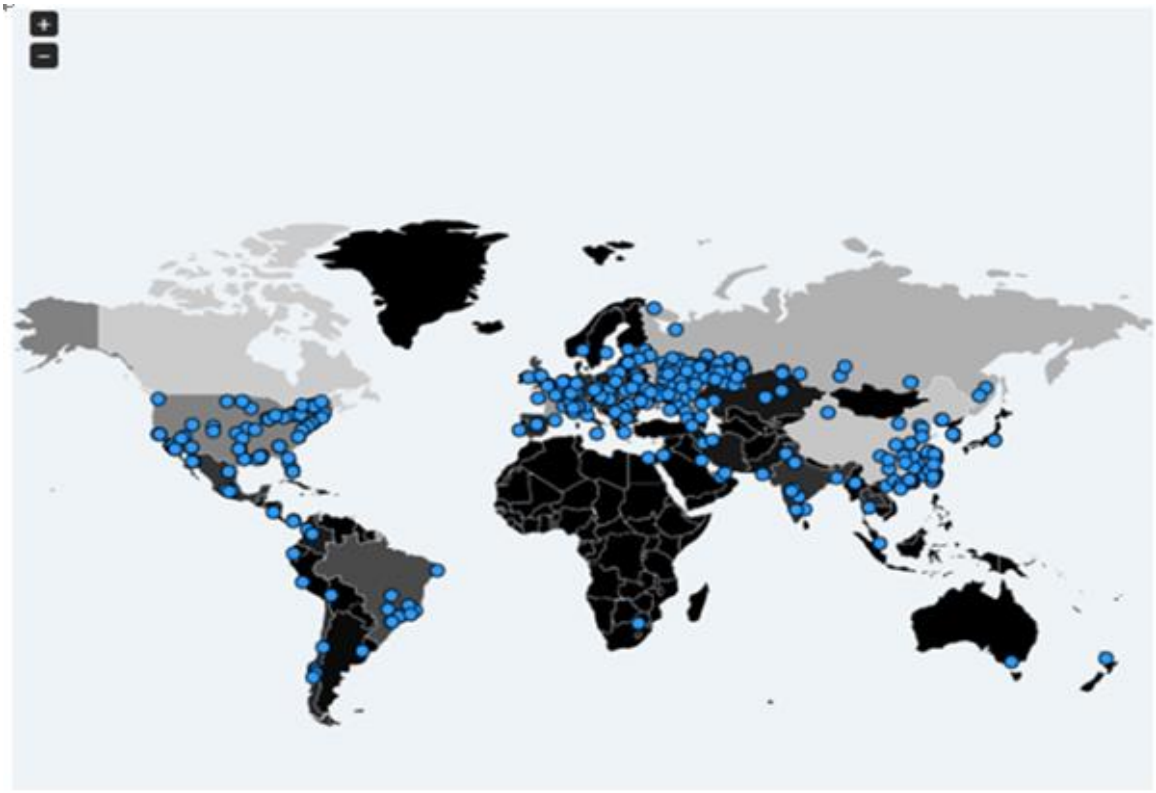
Keywords: Ransomware, Virus classification, Convolutional Neural Networks, LeNet-5, Behavior feature

壹、前言

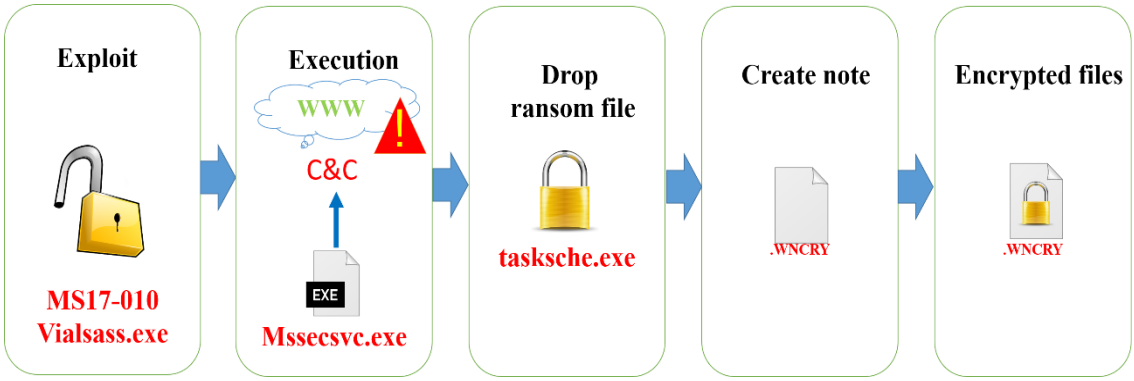
勒索病毒是一種特殊的惡意軟體，與其他病毒不同於目的在於收取贖金，會將受害者電腦上鎖、將受害者硬碟上的檔案加密，會要求受害者繳納贖金以取回電腦控制權，但是付完贖金之後只是「有機會」救回檔案資料勒索病毒於近幾年內開始流行，這些勒索病毒通常皆透過殭屍網路與電子郵件內夾帶偽裝檔案進行傳遞，如讓受害者執行後便寫入機碼內，開始進行加密行為，加密成功後便出現提示受害者發現系統內檔案已被加密，並要求受害者支付數位貨幣如比特幣等解鎖，各病毒家族間差異處主要為加密系統啟動是利用之系統漏洞不同，故傳遞手法也有所不同，如相當知名的 WannaCry[18]便是其中一隻勒索病毒。

2017 年勒索病毒爆發，讓世界各地從一般民眾到公司行號政府機關電腦內重要檔案被被綁架如圖一，防毒軟體固然可以即時偵測電腦病毒，但勒索病毒的行為似毒非毒，防毒軟體幾乎無法預先察覺是惡意程式，除非已經有感染案例並且被加入病毒碼，但是勒索病毒推出太快，一般更新病毒碼式的防毒軟體對於新的病毒幾乎沒有招架之力，等到發現時已經為時已晚[18]。

在物聯網中可連網之 IP 設備如偵測器內含精簡型電腦，如街頭攝影機含樹莓派等皆須面對惡意程式之入侵、攻擊、竊取隱私資料等威脅，做為中繼跳板，若再誘使下載網路勒索病毒(Ransomware)，引發後果不堪設想。趨勢科技於 2017 年 4 月中首次監測到勒索病毒（RANSOM_WCRY.C），最初它透過網路釣魚攻擊誘使使用者從 Dropbox 網址下載惡意程式，感染的程序如圖二；後來，趨勢科技發現這個肆虐全球的勒索病毒「WannaCry/Wcry」已進化為結合了 SMB 伺服器漏洞 EternalBlue 與新勒索病毒家族（RANSOM_WCRY.I / RANSOM_WCRY.A）的新變種。由於網路勒索病毒 WannaCry 肆虐，台電也受害，全公司有 779 台電腦中毒，受害單位遍及台電上下，包括總管理處、水火力電廠及業務單位等，中毒電腦皆已網路斷線並隔離搶修。教育領域有 10 所學校（包含台大、台師大）的 59 台電腦中毒，但因為都不屬於核心系統，電腦重灌以後就沒問題。[14]



圖一: 2017 年勒索病毒全球肆虐情形地圖(藍色代表通報案例)
資料來源：引用自[18]

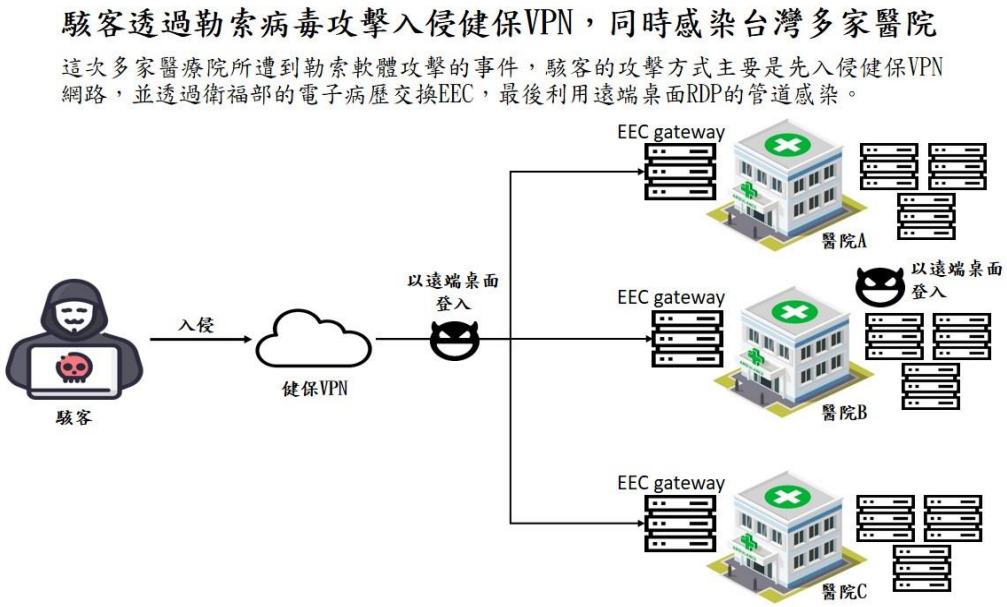


圖二：勒索病毒之感染途徑 [27]

修改自: https://buzzorange.com/techorange/2017/05/15/trendmicro-wannacry/

針對面臨的資安困境，美國資安公司 ZingBox 提出的主動式解決方案是透過人工智慧(Artificial Intelligence, AI)與機器學習(Machine learning, MA)等創新技術，可以在幾個小時內自動偵測出特定區域內絕大部分的物聯網設備，並僅需將所有蒐集數據的 1%上

傳雲端進行分析，即可以快速識別每一個裝置的獨特性，及時監測裝置的流量，判別異常行為以利防禦工作。[25] 此外，在台灣在 2019 年 8 月底傳出醫院大規模受到勒索病毒攻擊的狀況，造成醫院應用系統檔案被加密，成為去年臺灣最大的勒索軟體攻擊事件，引發社會大眾的關注，攻擊過程如圖三所示。



圖三: 2019 年 8 月勒索病毒在台醫療機構入侵與攻擊
資料來源：引用自[26]

傳統網路威脅分析方法常以行為關聯分析技術，搭配封包蒐集、過濾與精確特徵比對檢視，需要大量資安專家人工作研判，常無法及時判斷或誤認新型態網路威脅，故目前網路入侵偵測與防護已結合 AI 之深度學習 (Deep Learning，DL)功能模組。深度學習使用人工神經網路來模擬大腦生理運作，學習威脅攻擊程序的共通點，有助於識別潛在威脅並輔助專家準確判斷威脅類別，並推薦對應防護的決策。近期資安工程師嘗試應用深度學習網路搭配機器學習方式不同威脅的行為特徵，嘗試以自動化方式偵測來協助網管工程師執行網路安全的防衛；基本上，各種深度學習模型各有專長，但特定深度學習模型常可解決單一項問題，例如神經網路為基礎之影像文字解說(Neural Image Caption，NIC)是透過卷積神經網路(Convolutional Neural Networks, CNNs)做影像識別，加上遞迴神經網路(Recurrent Neural Networks, RNNs) 語言產生器共同來完成此一先進功能，最有名是 Microsoft 發展的 MSR 系統，故近期複合式深度學習模型興起是此領域一個新發展的趨勢。

1.1 研究動機及目的

為加速網路入侵偵測對勒索病毒之威脅辨識精度並降低誤判率(False Positive Rate, FPR)，本研究針對序列性攻擊資訊提出一個基於深度學習為基礎之網路異常偵測模式，採用深度學習網路 CNN 模式開發病毒鑑別模式，運用 Cuckoo 沙盒[6]分析行為特徵 (behavioral features)，再運用正規化概念分析 (Formal Concept Analysis, FCA)[15]客觀建構勒索病毒之行為特徵矩陣，並轉換成特徵向量陣的影像格式，透過 LeNet-5 卷積神經網路[17]模式學習以進行萃取特徵向量與影像類別映射。

1.2 研究特色

本研究為一在實驗室之多階段與延續性的研究，先前學長透過 GitHub theZoo 專案 (https://github.com/ytisf/theZoo)[20]、GitHub MalwareDatabase 專案 (https://github.com/Endermanch/MalwareDatabase)[1]、ANY.RUN - Interactive Online Malware Sandbox[4]、Tutorial Jinni | Hub of Tutorials[9]蒐集到的病毒樣本，運用 Cuckoo 沙盒動態分析萃取出其行為特徵，搭配 Protégé 知識庫發展環境[13]，運用知識本體理論與 FCA 確認病毒種類與特徵間關聯，將特徵與病毒種類之關聯轉換成為行為特徵矩陣，以作為 CNN 模式的學習資訊與模式預訓練的輸入，以求取隱藏層特徵矩陣之最佳化權重，搭配 Softmax 分群函數以識別不同來源的威脅類別，提高病毒分類模式的精確度和降低誤判率，以利管理者正確識別已知勒索病毒及變種病毒種類。本研究期望達到以下目的：

(一)、蒐集近期已知的網路勒索病毒樣本，以 Sandbox 動態分析與實作一個的勒索病毒特徵行為特徵矩陣，系統化分析勒索病毒種類與其行為特徵。

(二)、建立勒索病毒的正規化抽象模型，明確定義病毒與攻擊行為間之關聯，以做為 LeNet-5 卷積神經網路學習病毒特徵，求取隱藏層特徵矩陣之最佳化權重，，搭配 Softmax 分群函數以識別不同來源的威脅類別。

(三)、為了提高模式判斷精確度，加入倒傳播最佳演算法遞迴更新，以修正深度網路學習網路之模式誤差，求取最佳模式參數，以提高網路型入侵偵測系統 (Network-based Intrusion Detection System, NIDS)對威脅分類預測的精度與降低誤判率。

1.3 論文章節架構

本篇論文架構共分為四章，第一章為緒論，說明研究背景、動機與目的。第二章為文獻與技術回顧，探討勒索裝置病毒種類及行為特徵分析方法、深度學習網路及 LeNet-5 卷積神經網路（Convolutional Neuron Network, CNN）等文獻。第三章為研究方法，說明本研究的研究方法、流程及架構。第四章為勒索病毒家族分類，透過 Cuckoo 工具執行病勒索毒動態行為資料之分析，運用 TensorFlow 及 Python 進行 LeNet-5 模式學習。

貳、文獻探討

本章節介紹病毒動態行為特徵分析、深度學習網路及 LeNet-5 卷積神經網路的相關研究。

2.1 病毒動態行為特徵分析技術

病毒動態行為特徵分析技術包括分析程式結構之靜態分析與沙盒為主的動態分析。

2.2.1 動態分析

惡意程式動態行為是具有時間先後順序的，透過統計分析統計特定事件序列，稱為情節(scenarios) 例如開啟通訊埠號(port)、修改機碼(registry) 、查詢 DNS、對外連線、檔案下載、開機自動執行惡意程式、啟動 superuser 權限及停止資安紀錄等情節。換句話說，觀察特定之序列事件發生頻率及數量，可以確認威脅的行為特徵。進一步分析，特定行為特徵可組成情節。一般而言，時間序列數據的預測運算一般有兩個基本要求：1)網路輸入 $x_0, ..., x_t$ 時序資料，期望輸出相同長度的 $y_0,..., y_t$ 的預測； 2)對時刻 t 的預測 y_t 只能通過 t 時刻之前的歷史輸入資料 $x_1,…, x_{t-1}$ 來判別。

2.2.2 靜態分析

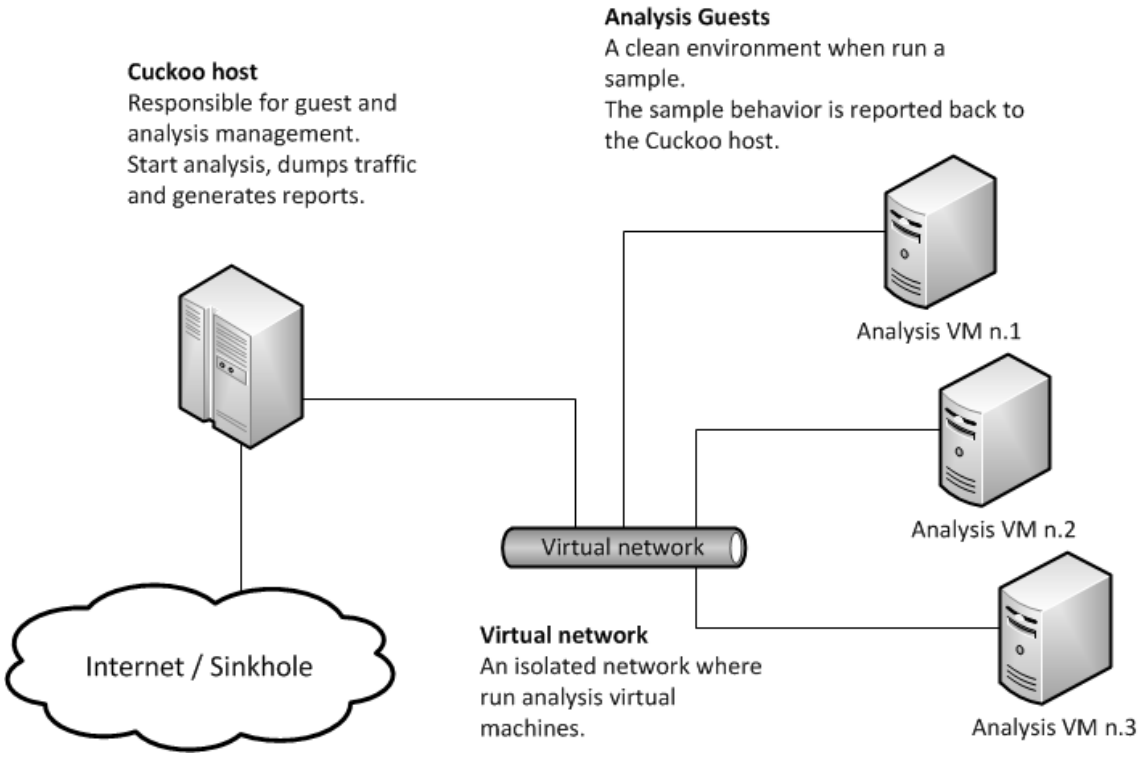
靜態分析方法的特點是運用圖形理論產生呼叫關係圖 (call graphs)、控制流程圖分析:以呼叫函數為單位，分析可能拜訪各分支點、基於路徑的靜態系統調用分析；2.程式都沒有被實際執行，故靜態分析執行較快速也較簡單；靜態分析方法使用限制 1.程式未經實際執行，容易產生大量的錯誤回報；.以靜態分析為基礎之汙染分析，因為程式未真實被執行，預測準確率較低。應用程式中常有應用程序 API 呼叫 (API calls)及呼叫待決定傳遞參數，靜態分析可同時模擬多條分支，但常無法實際決定執行路徑，此一問題可由動態分析運用標示資訊分之與流向，迅速找出程式實際執行的路徑。

2.2 沙盒分析技術

沙盒(sandbox)是一個抽象的概念，是指「在某個特定的環境中，根據所需要的安全性限制程式的行為」，讓程式放在模擬真實環境的沙箱內執行，對病毒做動態的追蹤，觀察是否有異常行為，透過此方式可不影響真實的系統環境檢測出病毒。知名沙盒包括 CWSanbox、Cuckoo、TRUMAN。動態分析讓程式在虛擬的環境下執行並監控觀察其行為，由於程式必須持續執行，因此耗費的運算時間明顯相對比靜態分析多。沙箱可依目的不同分為多種技術，像是要進行檢查而停在某個程式呼叫執行點、防止潛在的危險而中止程式、在程式運行時監控及紀錄程式活動 Cuckoo 沙盒是在參加 2010 年 Google 程式專案夏令營的作品，最近在 GPL 許可下開源發佈。任何人可以將其加到自行研發的

專案中，打造自己的惡意軟件行為分析工具。基本上，Cuckoo 是一個輕量級的 windows 二進制文件行為自動動態分析工具。它能夠給出程序運行過程中詳細的關鍵 API 調用和網路活動。 由於 Cuckoo 的開源特性和廣泛的模塊化設計，可以客製化分析環境，可將各階段的分析結果處理以和產出報告。為使用者可透過指令要求 Cuckoo 提供服務，可以按照您希望的方式輕鬆地將沙箱架構再到現有的計算機組態和後台中，只要滿足開放源碼 GPLv3 許可要求。建置一個可以執行程式的環境紀錄可疑行為及資訊洩漏(information leakage)，透過 Cuckoo 沙盒可以觀察程式所造成的影響的行為，如圖四，分析完成可提供以下資訊[27]:

1. 由惡意程式生成的所有進程執行的 win32 API 呼叫的追蹤。
2. 惡意程式在執行期間創建，刪除和下載的文件。
3. 惡意程式進程的內存轉儲。(Memory dumps of the malware processes)
4. 網路流 PCAP 格式紀錄。(Network traffic trace in PCAP format)
5. 執行惡意軟件期間拍攝的 Windows 桌面的螢幕截圖。
6. 主機的完全內存轉儲，包括自動運行 Volatility。(Full memory dumps of the machines, including automatic running of Volatility).
7. 工具更多資訊及下載：http://www.cuckoobox.org/[19]



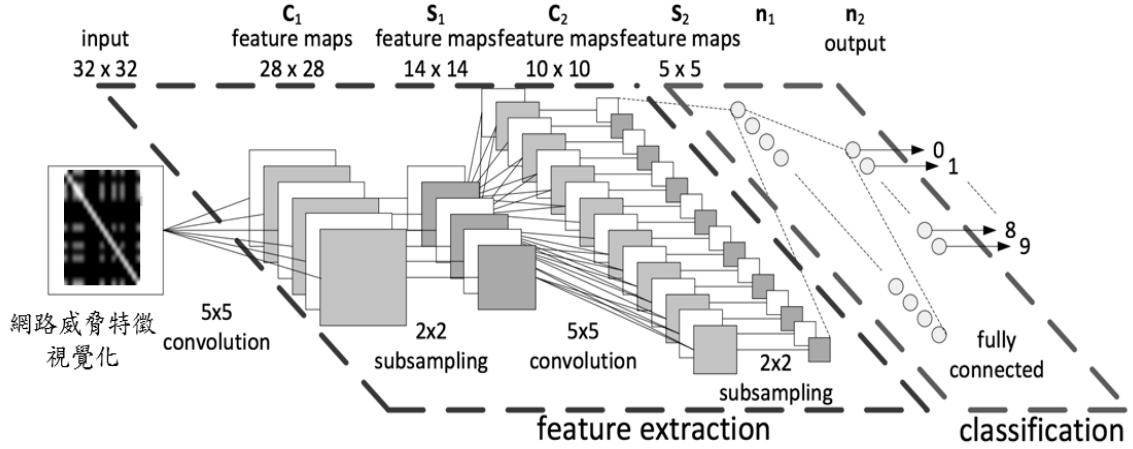
圖四: Cuckoo 的主要架構 [19]

2.3 LeNet-5 卷積神經網路

卷積神經網路(Convolutional neural network，CNN)設計的目標是用來處理以多陣列型態表達的資料。卷積神經網路(CNNs)在圖像和語音識別科學領域表現出色，尤其對於大型圖像識別性能表現優異，因為卷積神經網路可直接輸入原始圖像並省掉對圖像的前處理，故獲得更廣泛的應用與高的辨識精度。

在 CNNs 發展的歷史中，手寫數字的識別問題中獲得成功 LeNet-5 是值得討論；LeNet-5 是基於 LeNet 的基礎上，LeNet 卷積神經網路是在 1993 年由貝爾實驗室（AT&T Bell Laboratories）開發並被部署於支票辨識系統，1998 年 Yann LeCun 及其合作者構建了更加完備的卷積神經網路 LeNet-5 並在手寫數字的識別成功率中獲得成功。LeNet-5 沿用了 LeCun (1989)的學習策略並在原有網路設計中加入了池化層對輸入特徵進行篩選，提高分類的精度。基本上，LeNet-5 及其後產生的變體定義了現代卷積神經網路的基本結構，其構築中交替出現的卷積層-池化層被認為能夠提取輸入圖像的平移不變特徵。微軟公司在 2003 年使用卷積神經網路開發了光學字符讀取系統 ，目前應用研究也得到展開包括人像識別、手勢識別等。[16]

實務上，CNN 的基本結構包含兩層，一為特徵擷取層-包括卷積層、線性整流層(Rectified Linear Units layer, ReLU layer）與池化層，其神經元的輸入與前一層的局部感知區域相連，擷取輸入資訊局部特徵；另一層為特徵映射層即為全連通層內含特徵權重與 SoftMax 分類層，卷積神經網路的計算層由多個特徵擷取與映射層而組成，最後將特徵權重映射至一個平面作分類運算。綜合上述，卷積神經網路已被視為一種頗具吸引力的深度學習結構，比起其他前饋神經網路或深度神經網路，其結構簡潔與所需估計的參數較少。由圖五可知，LeNet-5 卷積神經網路通常由 2 個卷積層、池化層（pooling layer）和連接輸出的全連通層(fully connected layer)與分類層(dense layer)層所組成。而全連通層內容包括特徵權重(weights)和 SoftMax 分類層。



圖五: 應用 LeNet-5 卷積神經網路於網路威脅分類示意圖 source:修改自[2]

卷積神經網路運用卷積運算、池化運算與分類運算來辨識特定圖像，CNN 架構設計具有以下三個主要特點[5]:

1. 感知區域(Receptive field)：一般採用 3 維的圖像資料與神經元連接方式，實際上也可以直接採用 2 維的圖像資料作為輸入，但隱含層內部的神經元只與原本圖像的某一小塊區域做連結，該區塊我們稱之為感知區域。
2. 局部連結 (Local connectivity)：根據上述感知區域的概念，CNN 使用卷積運算為基礎之過濾器(filters)增強與該局部圖形空間的相關性，然後堆疊許多層以達到非線性濾波的功能，萃取圖形的抽象特徵，也就是允許網路架構從小區塊圖形的明確特徵值，組合後變成大區塊的特徵值。
3. 3.共享權重(Shared weights)：當在原始圖像產生一特徵圖(feature map)時，其模式的權重向量及偏誤(bias)是共用的，確保在該卷積層所使用的神經元會偵測相同的特徵(抽象特徵)，並且即使圖像位置或是有旋轉的狀態仍然可被偵測，換句話說卷積神經網路主可應用在於辨識因縮放、位移等不同形式扭曲的圖形。

這三個特點使得 CNN 在圖像辨識上比傳統機器學習有更好的辨識效果。

參、勒索病毒特徵分析與辨識

本章介紹知識本體應用於勒索病毒概念全文之建立與行為特徵的篩選過程。

3.1 勒索病毒之知識本體

本研究將運用本體論及正規概念分析法(Formal Concept Analysis, FCA)建置本研究之知識本體，再使用美國史丹佛大學生物醫學信息學研究中心開發的 Protégé 建置「物聯網裝置病毒知識本體庫」。學者 Uschold & Gruninger 在 1996 年提出一般本體論的構成方式，是由實體(entities)、屬性(attributes)以及關聯(relations)三個概念所組成[10]，實體指的是在一領域中有形式或無形式的重要事物；屬性是用以敘述概念的特性以及可能的範圍；關聯則用以說明實體或實體與概念之間的規則及關係[21]。知識本體應有的基本要素為：類別(Class)、資料槽(Slot)、實例(Instance)及基本元素(Axiom)[11]，類別與類別之間可存在繼承關係，子類別可繼承父類別所定義之資料槽，資料槽可用以描述類別與類別之間的關係(relationship)，包含類別與類別的實例是一組完整的知識概念，亦即知識基礎。以電腦病毒為例說明病毒知識本體的基本要素定義與案例如表二。

表二: 勒索病毒知識本體的基本要素[23][24]

	基本定義	案例
類別 (Class)	・類別:具有相同屬性的物件群體，大型類別常常可再細分子類別 ・子類別:具有與類別部分相同之屬性的物件群體	・類別:電腦病毒一般區分主機型或網路型病毒 ・子類別:勒索病毒、僵屍病毒、蠕蟲、特洛伊木馬可視為網路型病毒的子類別，勒索病毒或僵屍病毒可視為網路型病毒的子類別
資料槽 (Slot)	・在知識本體論中用來描述概念的屬性或概念之間的關聯 ・其中父類別與子類別間的關聯也是一種 Slot	・勒索病毒子類別包括 WannaCry、CryptoLocker、Lucky、BadRabbit、Santna 、 Ryun 和 Keypass 等
實例 (Instance)	・知識本體中為一個概念或類別的案例，實例會繼承類別的所有屬性或關聯	・電腦中 WannaCry 勒索病毒案例，該病毒進入目標主機之後，就會對主機硬碟和儲存裝置中許多格式的檔案進行加密。
基本元素 (Axiom)	・於知識本體中是原則或限制，其功能在於制定概念間關聯或限制 ・與 Slot 不同之處在於，Slot 清楚定義兩個類別之間的關聯	・勒索病毒的限制是受感染主機之通常透過木馬病毒的形式傳播，將自身為掩蓋為看似無害的檔案，通常會通過假冒成普通的電子郵件等社會工程學方法欺騙受害者點擊連結下載感染。

Protégé 知識本體編輯器[13]是由史丹佛大學發展，已應用於知識探索之用，其已可應用於電腦病毒特徵儲存、病毒分類與識別；Protégé 資料庫定義三個主要的知識模組: (1)說明特定的類別(class)，(2)描述每一個概念的性質與屬性稱為資料槽(slot)，(3)描述資料槽位的限制稱為限制面向(facet)，類似前述的 axiom，類別所產生的知識實例稱為 instance。Protégé 工具繪製知識本體之類別關係圖可以清楚表達出每一個類別(class)所階層中包含的資料槽、類別和類別之間的關聯、描述屬性限制面向、屬性的型態達到共同的認知，可釐清類別與類別之上、下位及相鄰關係，減少概念之衝突，架構出概念清楚的知識本體。

Protégé 工具可外掛 FcaView Tab 功能模組，將知識本體轉換成非同形式於知識本體的本文(context)，並藉由物件中相同屬性的關聯產生隱含的資料槽。本文中的物件為知識本體的類別、屬性為知識本體的特徵，兩者作為本文交叉關聯表的 X 軸與 Y 軸；本文交叉關聯表中若同一列的物件跟同一欄的屬性有關聯則成為一個概念，勒索病毒知識本體的概念全文，如表三所示。

表三: 勒索病毒家族概念全文(33 家族) [24]

	Checks_m...	generate a...	Steals priv...	Allocates r...	sandbox e...	DNS query	Installs its...	Attempts to...	monitors k...	Runs bcde...	Deletes a l...	Likely insta...	Writes a p...	Detects Vir...
Cerber	X					X		X	X				X	X
Cryptowall				X	X	X								
Locky						X						X		
Petrwrap												X		
Petya				X										
Radamant	X						X							
Satana						X						X	X	X
Vipasana														
WannaCry	X	X	X	X		X	X	X		X	X			
TeslaCrypt			X	X	X		X	X					X	X
BadRabbit	X			X			X							X
Ryun							X							
CryptoNar	X			X			X							
Hermes				X										
LockerGoga	X		X					X					X	X
Termite	X						X							X
KeyPass	X		X	X			X	X						X
CMB Dharma			X	X			X	X			X			X
GandCrab				X		X		X						X
dexec	X						X				X			X
KeyPass	X		X	X			X							X
Ryun					X		X							
satan	X						X							
Satanamem						X	X					X	X	X
cerber	X			X		X		X					X	X
BigBobRoss	X												X	
Krotten							X							
NoMoreRansom	X			X		X	X							
Matsnu				X										
Termite	X				X		X							
Xyeta				X			X							
WinlockerVB6Blacksod	X			X										

本研究將運用本體論及正規概念分析法(Formal Concept Analysis, FCA)建置本研究之知識本體，再使用美國史丹佛大學生物醫學信息學研究中心開發的 Protégé 建置「物聯網裝置病毒知識本體庫」。學者 Uschold & Gruninger 在 1996 年提出一般本體論的構成方式，是由實體(entities)、屬性(attributes)以及關聯(relations)三個概念所組成[15]，實體指的是在一領域中有形式或無形式的重要事物；屬性是用以敘述概念的特性以及可能的範圍；關聯則用以說明實體或實體與概念之間的規則及關係[21]。知識本體應有的基本要素為：類別(Class)、資料槽(Slot)、實例(Instance)及基本元素(Axiom)[14]，類別與類別之間可存在繼承關係，子類別可繼承父類別所定義之資料槽，資料槽可用以描述類別與類別之間的關係(relationship)，包含類別與類別的實例是一組完整的知識概念，亦即知識基礎。

3.2 病毒行為特徵的篩選

實務上，CNN 在處理網路入侵偵測系與分類方面的問題有不錯的表現，但過多的行為特徵卻造成執行工作負擔。本研究選擇決策樹枝 ID3 演算法[7]之資訊增益(information gain)進行重要行為特徵的篩選：假設每個類別的資料個數為 $x_{.j}$，N 為所有資料個數，$p_j = x_{.j}/N$ 維個類別出現的機率，因此 k 個類別的資訊總和 Info(D)為：

$$\text{Info(D)} = -\frac{x_{.1}}{N}\log_2\left(\frac{x_{.1}}{N}\right) - \frac{x_{.2}}{N}\log_2\left(\frac{x_{.2}}{N}\right) - \cdots - \frac{x_{.k}}{N}\log_2\left(\frac{x_{.k}}{N}\right) = -\sum_{j=1}^{k} p_j \cdot \log_2(p_j) \quad (1)$$

Info(*D*)又稱熵(entropy)常用來衡量資料離散程度或亂度，可作為評估訓練資料集合 *D* 下所有類別的期望訊息。假設該資料集合 *D* 要根據屬性 *A* 分割成 *L* 個資料分割集合 D_i，

其中 $x_{i.}$ 為各屬性值 A_i 下的分割資料總個數，x_{ij} 為屬性值 A_i 下且為類別 C_j 的個數，故屬性 A_i 的資訊 $\text{Info}(A_i)$ 為：

$$\text{Info}(A_i) = -\frac{x_{i1}}{x_{i.}}\log_2\left(\frac{x_{i1}}{x_{i.}}\right) - \frac{x_{i2}}{x_{i.}}\log_2\left(\frac{x_{i2}}{x_{i.}}\right) - \cdots - \frac{x_{ik}}{x_{i.}}\log_2\left(\frac{x_{ik}}{x_{i.}}\right) \quad (2)$$

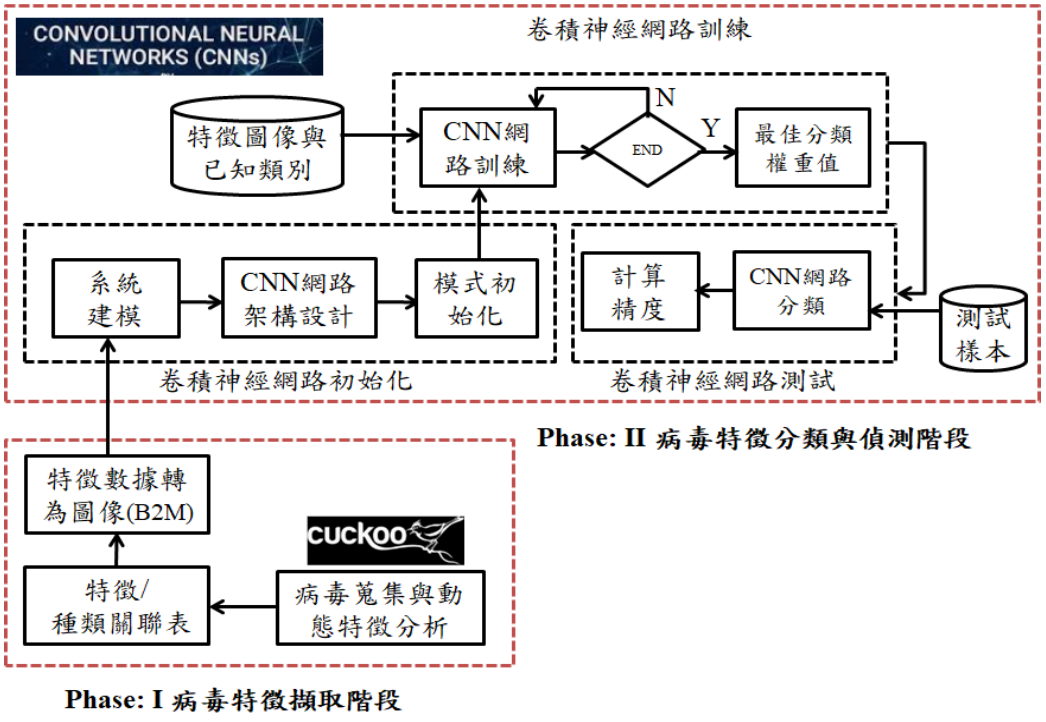
而屬性 A 的資訊 $\text{Info}_A(D)$ 則根據各屬性值下的資料個數多寡決定為

$$\text{Info}_A(D) = \frac{x_{1.}}{N}\text{Info}(A_1) + \frac{x_{2.}}{N}\text{Info}(A_2) + \cdots + \frac{x_{l.}}{N}\text{Info}(A_l) = \sum_{i=1}^{l}\frac{x_{i.}}{N}\text{Info}(A_i) \quad (3)$$

資訊增益 $Gain(A) = \text{Info}(D) - \text{Info}_A(D)$ 可表示屬性 A 作為分枝屬性對資訊的貢獻程度，計算各個屬性作為分枝對資訊的貢獻程度後可找出最佳資訊的分枝屬性。

3.3 LeNet-5 卷積神經網路之病毒行為分類

透過 Cuckoo 動態實驗分析病毒特徵，運用知識本體的資料庫之系統化建立病毒種類與種類、種類與屬性間之關聯，再運用 LeNet-5 卷積神經網路做病毒行為分類與辨識，其實驗規劃流程為圖六，分類的精確度可藉由機率理論估算電腦病毒與各特徵間關聯之支持度。



圖六: 研究流程

圖六中 CNN 模式已被應用於網路惡意程式行為之特徵學習、識別與偵測，近期 CNNs 已轉為在新型網路威脅及電腦病毒特徵的識別應用。雖然樣本學習過程耗用較多時間訓練以設定隱藏的參數(卷積核)，但完成訓練部署的主機可即時且精確偵測網路惡意程式的入侵。本研究參考[25][27][22][28][26]先前在網路惡意程式分析研究，將應用深度學習卷積神經網路(CNN)之 LeNet-5 模式於對網路惡意程式特徵的監督學習過程，進而擷取抽象行為，透過核心演算法的最佳化過程以學習網路惡意程式特徵中隱藏的參數以及種類間關係，最終用於網路惡意程式之特徵分析，研究將區分兩個階段，階段一為以 cuckoo 擷取病毒行為特徵(於 4.1 節說明)，階段二為以卷積神經網路分析勒索病毒家族之行為特徵(於 4.2 節說明)。

肆、驗證與確認

本章透過 Cuckoo 動態實驗分析病毒特徵，運用知識本體的資料庫之系統化建立病毒種類與種類、種類與屬性間之關聯，再運用 LeNet-5 卷積神經網路做病毒行為分類與辨識。

4.1 Cuckoo 動態分析

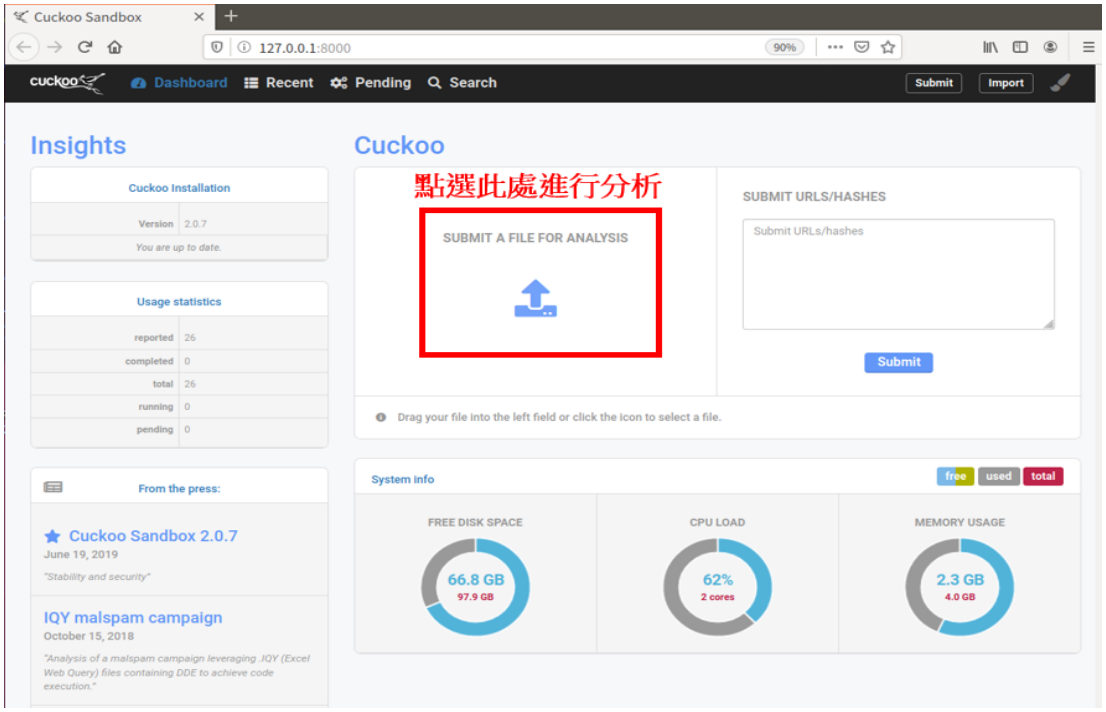
以著名的 WannaCry 勒索病毒為例，WannaCry 被認為利用了美國國家安全局的「永恆之藍」（EternalBlue）工具以攻擊 Microsoft Windows 作業系統的電腦，為 WanaCrypt0r 1.0 的變種。「永恆之藍」利用了某些版本的微軟伺服器訊息區塊（SMB）協定中的數個漏洞，而當中最嚴重的漏洞是允許遠端電腦執行程式碼。修復該漏洞的安全修補程式已經於此前的 2017 年 3 月 14 日發布，但並非所有電腦都進行了安裝與安全修補。本研究利用 Cuckoo 動態分析病毒樣本，監測到木馬程式會以的加密型勒索軟體兼蠕蟲病毒（Encrypting Ransomware Worm）進行攻擊。[18]該病毒利用 AES-128 和 RSA 演算法惡意加密用戶檔案以使用 Tor 加密通訊並勒索受害者之比特幣。

當受害者被安裝惡意程式時，會進行以下活動，包括(1) 該病毒進入目標主機之後，就會對主機硬碟和儲存裝置中許多格式的檔案進行加密 (2) 利用網路檔案分享系統的漏洞，傳播到任意的其他聯網的主機 (3) 處於同一區域網路的相鄰主機也會被感染。該惡意程式安裝完後會連線至後台，此時受駭主機已遭受到程序監控，傳送出行動裝置使用的基本訊息至特定的遠端監控主機，如圖七。

圖七: WannaCry 勒索病毒受駭的畫面(取自:https://applealmond.com/posts/5171)

此案例利用動態紀錄模擬真實行動裝置系統來監看勒索病毒樣本執行過程，並把過整分類為利用 windows api 生成密鑰、檢察系統的記憶體容量(可以探測記憶體容量較少的虛擬機)、竊取本地瀏覽器個資資料、與未執行 DNS 查詢的主機通訊、在 Windows 啟動時自動執行、從作業系統安除大量下載惡意文件，作為建置 protégé 病毒特徵庫之動態分析屬性，其實驗步驟如下：

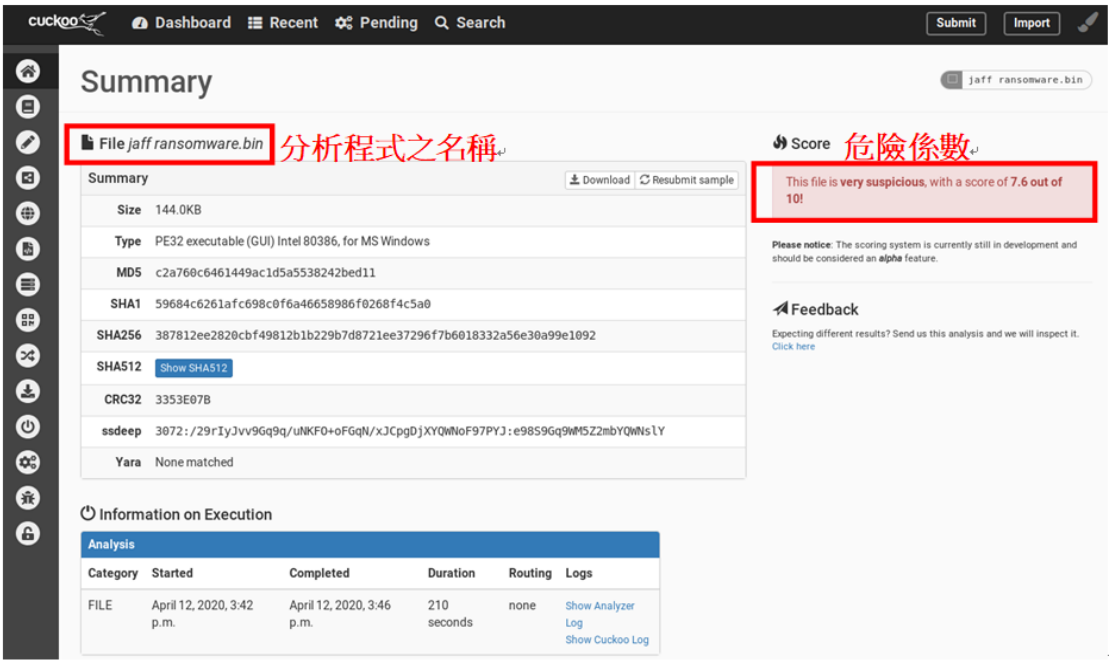
步驟 1.以除錯模式開啟 cuckoo 如圖八。



圖八: 開啟 cuckoo web 分析 WannaCry 勒索病毒

步驟 2. Cuckoo web 執行勒索病毒特徵分析如圖九。

步驟 3. 勒索病毒特徵分析結果之顯示如圖九。



圖九: 顯示分析之結果

步驟 4. 勒索病毒從中毒至發作詳細感染過程之清單如圖十。

圖十: 顯示勒索病毒執行流程之異常行為屬性

步驟 5. 以 Protégé 編輯病毒資料庫

依據前面步驟 1~4 的分析結果，彙總出勒索病毒之行為屬性欄位，如表四，在依據表 4 內的欄位輸入至 Protégé 特徵庫內，並將其轉換為 xml 格式，提供給 Concept Explorer 發展病毒概念全文及概念格，實驗步驟 5~8 如下：

表四: Cuckoo 動態分析資訊洩漏部分

編號	行為屬性描述
1	偵測是否為虛擬機環境（Detects the presence of VirtualBoX）
2	將新的文件擴展名或內容附加到數個文件中，以指示勒索軟體加密過程 (Appends a new file extension or content to Serveral files indicative of a ransomware file encryption process)
3	解析可疑的上級網域（TLD）(Resolves a suspicious Top Level Domain TLD)
4	創建一個已知的 Satana 勒索軟體解密指令/密鑰文件(Creates a known Satana ransomware decryption instruction / key file)
5	通過將可執行檔或 DLL 寫入另一個程序的來進行代碼注入(Code injection by writing an executable or DLL to the memory of another process)
6	在程序內存轉儲中找到與 Tor 相關的 URL (Found URLs related to Tor in process memory dump)
7	刪除數種未知 mime 類型文件，這些類型表明勒索軟件將加密文件寫回到硬碟(Drops several unknown file mime types indicative of ransomware writing encrypted files back to disk)
8	嘗試找到瀏覽器的安裝位置(Tries to locate where the browsers are installed)
9	使用 UPX 壓縮可執行程式 (The executable is compressed using UPX)
10	流量攔截攻擊(Traffic Interception Attack)
11	一個或多個緩衝區包含一個嵌入式 PE 文件(One or more of the buffers contains an embedded PE file)
12	創建可疑文件、程序或服務(Creates a suspicious file or process or service)
13	從系統中刪除大量文件，指示勒索軟體或系統破壞(Deletes a large number of files from the system indicative of ransomware, wiper malware or system destruction)

步驟 6. 勒索病毒家族概念全文之建立

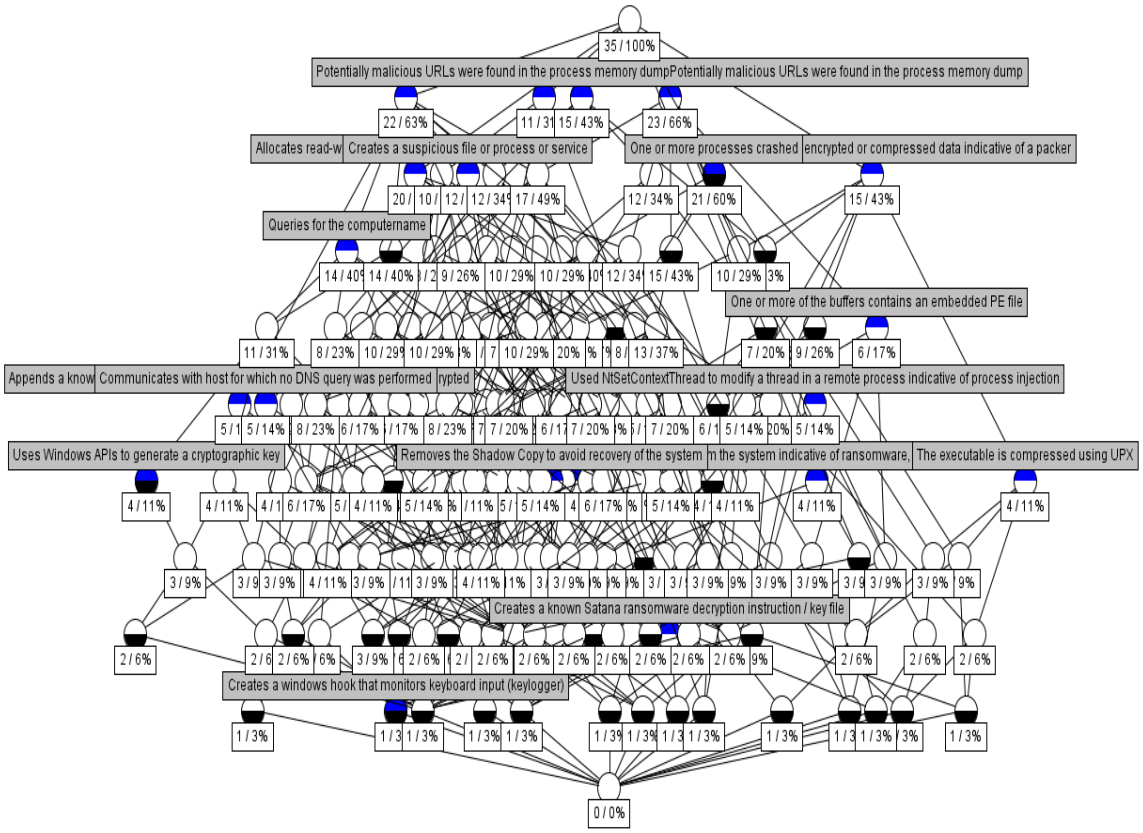
為清楚說明所蒐集勒索病毒家族概念全文，九個家族病毒經由 Cuckoo 動態分析出來的數據結果建立概念全文，病毒命名方式皆使用防毒軟體公司賽門鐵克(Symantec)病毒數據庫為依據，每一行代表一個病毒物件，每一列代表病毒的屬性特徵，交叉關聯矩陣註記「X」表示同一列的物件關聯同一行的物件，如表五。

表五: 勒索病毒家族概念全文(50 家族)

	Queries for...	Checks if p...	Uses Wind...	Command...	Tries to loc...	Collects inf...	This execu...	The execut...	The file co...	One or mo...	Creates a ...	One or mo...	Communic...	Resolves ...
BadRabbit	X	X		X						X	X	X		
BigBobRoss										X	X			
Birele								X		X				
cerber	X	X		X		X				X	X	X	X	
CMB Dhar...							X			X				
CryptoNar							X			X				
cryptowall										X		X		
PolyRanso...							X			X				
dexec	X	X					X				X	X		
GandCrab	X	X	X								X			X
HannaSo...														
hrms										X				
KeyPass	X	X					X		X	X	X			X
kkk_pr3										X		X		
Krotten														
LockerGoga				X	X					X	X			
Locky								X		X				X
Matsnu									X			X		
NoMoreRa...	X													
Petrwrap										X				
Petya														
Radamant	X									X				
Ryun		X		X							X	X		
satan	X	X		X						X	X	X		X
Satana	X	X		X			X			X		X		
Termite	X			X	X			X	X	X	X			
TeslaCrypt				X						X	X	X		
Vipasana														
ViraLock							X			X				
WannaCry	X		X	X				X	X	X	X			
WinlockerV...	X						X		X					
Xyeta														
Xcry														
Sigrun	X	X	X											
Jaff	X	X	X					X				X		
NetWalker	X			X	X						X			
Csrs	X										X			
Maze	X	X	X							X				
Ako	X		X	X	X						X	X		
RagnarLoc...	X			X							X			
Phobos	X	X			X						X			
LockBit	X	X		X							X			
Spora	X	X	X	X	X					X	X	X		
MaMoCrypt...	X	X	X	X						X	X			
MZ Revenge	X	X	X	X						X	X			
Mr.Ddec	X	X	X		X					X	X	X		
MedusaLo...	X		X	X	X						X	X		
AlphaCrypt				X				X			X	X		X
Makop														

步驟 7. 勒索病毒家族概念格之建立

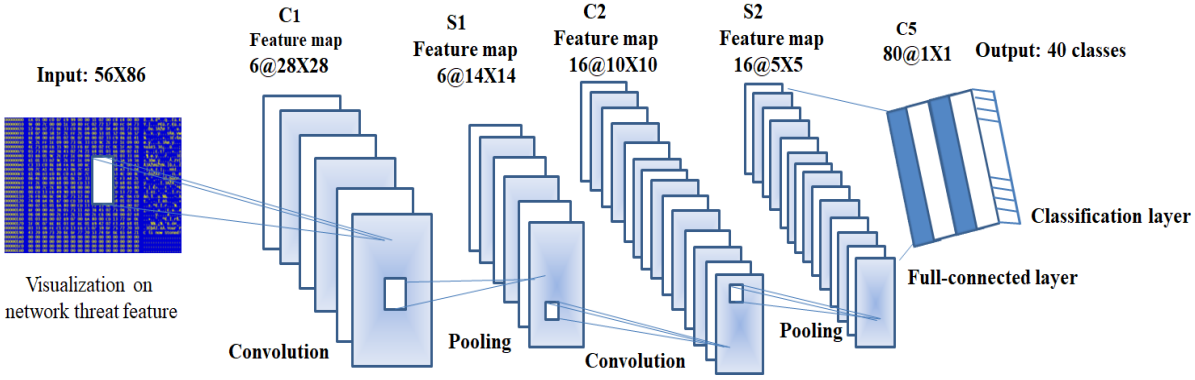
為了解勒索病毒家族與特徵之完整關聯，本研究將 50 個家族進行特徵動態分析，實驗結果先概念全文，再以 Protégé 資料庫提供之 FCA 進行病毒間相同屬性之合併，，繪製其概念格如圖十一。



圖十一: 以機率顯示行動裝置病毒樣本概念格之關聯

4.2 病毒分類與辨識

整理三十六個勒索病毒家族之分類與行為特徵聯，本研究使用 CNN 的經典代表 LetNet-5 卷積神經網路，本研究使用 LetNet-5 神經網路設計的用於檢測勒索病毒的卷積神經網路結構如圖十二所示。



圖十二: LetNet-5 為基礎的勒索病毒分類的卷積神經網路結構

LeNet-5 是一個應用於圖像分類問題的卷積神經網路，其學習目標是從輸入為設定時間窗網路流 50 家族行為特徵資訊共 86 組，使用 ID3 選擇權重值高的 60 個行為特徵組成次集合，轉換成的 60×60 的灰階圖像，再由 60×60×1 灰度圖像作為 CNN 輸入以識別不同類別(0-50)的勒索病毒家族。LeNet-5 的隱藏層由 2 個卷積層、2 個池化層構築和 2 個全連接層組成，如表六構建：

表六: 卷積神經網路結構

隱藏層	隱藏層的結構
卷積層C_1	使用感知階段預訓練得到的 6 個 5×5 大小的卷積核(權重)，對輸入圖像進行卷積，得到 6 張 28×28 的特徵映射圖。
池化層S_1	以 2×2 大小的視窗對 C1 層的 6 張圖像次抽樣得到 6 張 14×14 的特徵映射圖。
卷積層C_2	使用預訓練的 96 個 5×5 的濾波器對 S2 的 6 張特徵映射圖進行卷積，得到 16 張 10×10 的特徵映射圖。
次池化層S_2	以 2×2 大小的視窗抽樣，抽樣得到 16 張 5×5 的特徵映射圖。
全連接層C_3	以 1920 個 5×5 大小的濾波器將 16 張 5×5 的特徵映射圖全連接為一個 80×1 的向量。
輸出層(Dense)	以 Soft-max 分類器，輸出結果最多為 50 類，本研究最大值（編碼為 1~50）和正常程式（編碼為 0）。

卷積神經網路模式訓練

實作上須將圖像資料透過 ImageDataGenerator 進行讀取，然後藉由 flow 方法將 train data 與 train labels 進行包裝，再藉由 fit_generator 將包裝之訓練資料丟入神經網路進行訓練，如圖十三所示。

```
model.fit_generator(datagen.flow(train_img_data, train_label, batch_size=32),
steps_per_epoch=len(train_img_data) / 32, epochs=10,
validation_data=datagen.flow(test_img_data, test_label, batch_size=32),
validation_steps=int(len(test_img_data) / 32))
```

圖十三: 神經網路訓練

執行 ID3 演算法如式(4)~(6) ，將行為特徵依據墒值大小進行權重排列，擷取的前 60 個特徵如表七。

表七: ID3 將權重前 60 項特徵進行排序

Feature	IG	Rank	Feature	IG	Rank
Allocates read-write eXecute memory (usually to unpack itself).	1.143.	1.	Writes a potential ransom message to disk.	0.913.	17.
The binary likely contains encrypted or compressed data indicative of a packer.	1.132.	2.	Resumed a suspended thread in a remote process potentially indicative of process injection.	0.884.	18.
Installs itself for autorun at Windows startup.	1.132.	3.	Performs file moves indicative of a ransomware file encryption process.	0.884.	19.
Queries for the computername.	1.125.	4.	Drops an executable to the user AppData folder.	0.825.	20.
One or more processes crashed.	1.113.	5.	Allocates execute permission to another process indicative of possible code injection.	0.823.	21.
Detect the presence of VirtualBoX.	1.108.	6.	Uses suspicious command line tools or Windows utilities.	0.821.	22.
Potentially malicious URLs were found in the process memory dump.	1.084.	7.	One or more of the buffers contains an embedded PE file.	0.804.	23.
A process attempted to delay the analysis task.	1.075.	8.	Used NtSetContextThread to modify a thread in a remote process indicative of process injection.	0.758.	24.
Creates a suspicious file or process or service.	1.053.	9.	Executed a process and injected code into it, probably while unpacking.	0.731.	25.
Checks for the Locally Unique Identifier on the system for a suspicious privilege.	1.041.	10.	Communicates with host for which no DNS query was performed.	0.729.	26.
One or more potentially interesting buffers were eXtracted, these generally contain injected code, configuration data, etc..	1.041.	11.	Appends a known multi-family ransomware file extension to files that have been encrypted.	0.725.	27.
Checks if process is being debugged by a debugger.	1.022.	12.	traffic interception attack.	0.723.	28.
Command line console output was observed.	0.982.	13.	Deletes eXecuted files from disk.	0.720.	29.
Appends a new file extension or content to Several files indicative of a ransomware file encryption process.	0.982.	14.	The executable uses a known packer.	0.720.	30.
This executable has a PDB path.	0.937.	15.	The file contains an unknown PE resource name possibly indicative of a packer.	0.717.	31.
Uses Windows utilities for basic Windows functionality.	0.922.	16.	Deletes eXecuted files from disk.	0.712.	32.

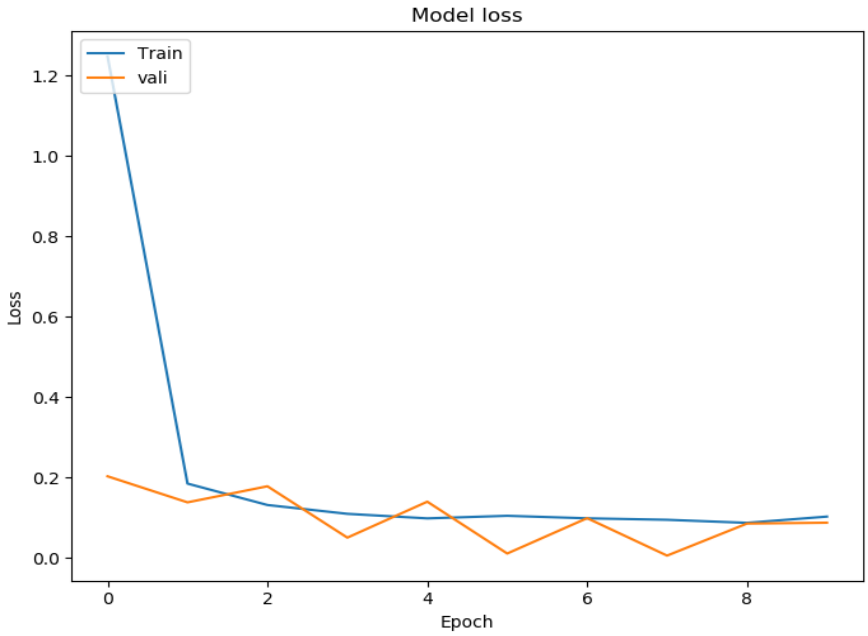
Feature	IG	Rank	Feature	IG	Rank
Uses Windows APIs to generate a cryptographic key.	0.681	33	Steals private information from local Internet browsers.	0.490	49
Resolves a suspicious Top Level Domain (TLD).	0.675	34	Changes read-write memory protection to read-execute.	0.472	50
Terminates another process.	0.675	35	Looks up the external IP address.	0.467	51
The executable is compressed using UPX.	0.675	36	Drops a binary and eXecutes it.	0.462	52
Code injection by writing an executable or DLL to the memory of another process.	0.661	37	Executes one or more WMI queries.	0.457	53
Deletes a large number of files from the system indicative of ransomware, wiper malware or system destruction.	0.657	38	Moves the original executable to a new location.	0.455	54
Removes the Shadow Copy to avoid recovery of the system.	0.651	39	Repeatedly searches for a not-found process, you may want to run a web browser during analysis.	0.455	55
Found URLs related to Tor in process memory dump.	0.632	40	Creates a known Satana ransomware decryption instruction / key file.	0.448	56
Checks whether any human activity is being performed by constantly checking whether the foreground window changed.	0.593	41	Operates on local firewall's policies and settings.	0.443	57
Expresses interest in specific running processes.	0.593	42	Installs Tor on the machine.	0.432	58
Creates a thread using CreateRemoteThread in a non-child process indicative of process injection.	0.582	43	Connects to Tor Hidden Services through a Tor gateway.	0.429	59
Manipulates memory of a non-child process indicative of process injection.	0.577	44	Performs Several file moves indicative of a ransomware file encryption process.	0.429	60
Connects to IP addresses that are no longer responding to requests.	0.577	45			
Drops serveral unknown file mime types indicative of ransomware writing encrypted files back to disk.	0.572	46			
Tries to locate where the browsers are installed.	0.490	47			
Starts servers listening.	0.490	48			

首先以選擇前 60 個高權重值的特徵進行以下實驗:模式訓練過程最佳化成本函數選擇則 cross-entropy，最佳函數 Optimizer 選擇 adadelta，疊代參數 batch_size=32，epoch=10，詳如表八。在監督的機器學習算法中，期望在學習過程中最小化訓練樣本的誤差，通常

是透過對成本函數的梯度下降來完成優化，並以損失函數(loss function)評估誤差下降速度與大小，若不佳則須修改模式成本函數或修正訓練參數，本實驗訓練過程之成本函數誤差下降如圖十四所示，epoch=8 後誤差下降達到穩定。接下來，將訓練資料進行 k-fold 交叉驗證，其中 k 值為 2~6 進行驗證，交叉驗證各-fold 運算結果如表九，使用不同特徵之準確度之驗證證模式測試平均準確率為 94.55%(60 個特徵) 如表十。

表八、模式訓練選擇的參數

Parameter in the proposed model	Value used in the proposed model
Image size	(60 x 60, Gray_level)
Loss in object function	Categorical cross entropy
Optimizer	adadelta
Learning_rate	0.1
Batch_per_epoch	32
Epoch	10



圖十四: 模型訓練的誤差收斂

表九:六類各 k-fold 最高驗證準度(60 個特徵)

k-fold	Epoch	Loss	Accuracy (%)
2	10	0.2866	94.11
3	10	0.2142	94.29
4	10	0.1376	95.17
5	10	0.1762	94.49
6	10	0.1532	94.70
平均	10	0.1935	94.55

表十:不同特徵數量之準確度驗證

No. of feature	Loss	Accuracy (%)
43	0.2650	86.47%
50	0.2041	92.26%
60	0.1935	94.02%

伍、結論

本研究導入深度學習網路(Deep Learning Networks)技術於勒索病毒分類研究，透過行為特徵監督學習，搭配 Cuckoo 病毒動態分析與特徵歸納，運用 LeNet-5 是卷積神經網路，將勒索病毒行為特徵轉換成灰度圖像，再由灰度圖像以識別不同 50 類別的勒索病毒家族，用以網路入侵偵測之勒索病毒之偵測與危害分析，其成果彙整如下：

(一) 針對近期發生的勒索病毒威脅，透過 Cuckoo 沙盒分析 50 類的勒索病毒家族的特徵集，再運用正規化概念分析 (FCA)建構勒索病毒之知識本體模型(ontological model)；

(二) 勒索病毒之知識本體雛型明確定義病毒與攻擊行為間之關聯，作為 LeNet-5 是卷積神經網路的輸入，再由 60x60 灰度圖像來識別 50 類的勒索病毒家族鑑定的參考；

(三) LeNet-5 病毒分類模式依據每一網路勒索病毒攻擊特徵與運作順序，可協助管理者系執行自動化之病毒類別偵測與變種病毒鑑定，作為評估網路病毒威脅之參考。

針對時間序列性之威脅資訊流，未來在病毒變種(variants)鑑定與判斷將導入時間卷積網路(Temporal Convolution Network, TCN)技術，以機器學習方式來學習複雜多類變種病毒行為特徵，並與 FCA 分析的結果作比較，提高病電腦病毒分類與偵測的正確性。

[誌謝]

本研究承蒙行政院科技部計畫（MOST 108-3116-F-168-001-CC2, TWISC2.0, and MOST 108-2410–H -168-003）

參考文獻

[1] A. Endermanch, MalwareDatabase, https://github.com/Endermanch/MalwareDatabase) (2019/06/11)

[2] A. Rosebrock, Rants, “Get off the deep learning bandwagon and get some perspective, *Machine* Learning, 2014,https://www.pyimagesearch.com/2014/06/09/get-deep-learning-bandwagon-get- perspective/

[3] A. Y. Javaid, Q. Niyaz,W. Sun, and M. Alam, “A Deep Learning Approach for Network Intrusion Detection System”, *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies*, 2015

[4] ANYRUN - Interactive Online Malware Sandbox, https://app.any.run/ (2019/06/13)

[5] C.Szegedy, V.Vanhoucke, S. Ioffe, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision, *Proceedings of the IEEE on Computer Vision and Pattern*, 2016.

[6] HoneyNet, Cuckoo Sandbox, https://github.com/cuckoosandbox/cuckoo) (2019/06/10)

[7] I. H.Witten and E. Frank, “*Data Mining: Practical Machine Learning Tools and* Techniques.” Morgan Kaufmann Publishers, 2005.

[8] J. Saxe, K. Berlin, “Deep Neural Network Based Malware Detection Using Two Dimensional Binary Program Features”, 2015, https://arxiv.org/abs/1508.03096v2.

[9] Jinni, Hub of Tutorials, tutorialjinni.com, https://www.tutorialjinni.com(2019/06/13)

[10] M.Uschold, M.Gruninger, Ontologies: Principles, Methods and Applications, *The Knowledge Engineering Review*, Vol.11, No.2, 1996.

[11] N.F. Noy and D.L. McGuinness.“Ontology Development 101: A Guide to CreatingYour First Ontology”, *Stanford Knowledge Systems LaboratoryTechnical Report* KSL-01-05, 2001.

[12] S. Tobiyama, Y. Yamaguchi† H. Shimada, T. Ikuse and T. Yagi, “Malware Detection with Deep Neural Network using Process Behavior,” *IEEE 40th Annual Computer Software*

and Applications Conference, pp.577-582, 2016

[13] Standard University, Protégé, https://protege.stanford.edu/ (2019/03/10)

[14] TechOrange，“ 趨勢科技教你逃離勒索病毒 WannaCry，從今天開始備份、拒當人質！”，2017-05-15，https://buzzorange.com/techorange/2017/05/15/trendmicro-wannacry/. (2019/11/08)

[15] Wikipedia, “Formal Concept Analysis, http://en.wikipedia.org/wiki/Formal_concept_analysis). (2019/06/10)

[16] Wikipedia, Convolutional neural network, https://en.wikipedia.org/wiki/Convolutional_neural_network (2017/09/15)

[17] Wikipedia, LeNet, https://en.wikipedia.org/wiki/LeNet (2019/11/11)

[18] Wikipedia, WannaCry, https://zh.wikipedia.org/wiki/WannaCry.(2019/10/27)

[19] WordPress, “Fighting malware for better online gaming experiences,” http://www.cuckoobox.org/ (2019/03/10)

[20] Y. Nativ, theZoo, (available online at https://github.com/ytisf/theZoo) (2019/03/08)

[21] 呂星學，“軟體發展規範中程序性知識與非程序性知識粹取與應用”，*碩士論文*，國防大學國防資訊研究所，2003

[22] 寇廣,湯光明,王碩,宋海濤,邊媛,“深度學習在僵屍雲檢測中的應用研究”. *通信學報*, 第 37 卷,第 11 期, 2016,頁 114~128.

[23] 林孝忠,王平,洪維謙,網路勒索病毒的特徵分析與知識本體模型建構, *Communications of the CCISA* , 第 25 卷, 第 2 期, 2019 ,頁 37-58。

[24] 洪維謙,王平,“網路勒索病毒的特徵分析與知識本體模型建構”, *碩士論文*,崑山科技大學資訊管理系,民 2019.

[25] 科技新報,“深度學習助網路攻擊偵測率升至 99%，NVIDIA 出資力挺”，2017-07-13 , https://technews.tw/2017/07/13/nvidia-investment-deep-instinct/ (2019/11/11)

[26] 羅正漢,“徹底揭露 2019 年臺灣最大規模病毒攻擊事件 勒索軟體衝擊！全臺醫療院所資安拉警報”, *iThome*, 2019-11-14, https://www.ithome.com.tw/news/134108 (2019/11/11)

[27] 陳智德,“醫療產業駭客威脅日益增加 零信任網路成為安全架構之一”, *DIGITIMES*, 2018-03-09, https://www.digitimes.com.tw/iot/article.asp?cat=158&cat1=20&id=0000525540_ 9IT4KNVC5HNXBAL6NI23B (2018/11/08)

[28] 韓曉光, 曲武, 姚宣霞等,“基於紋理指紋的惡意程式碼變種檢測方法研究，*通信學報*，第 35 卷,第 8 期, 2014 頁, 125-136.

[作者簡介]

王平，交通大學資訊管理研究所博士，現為崑山科技大學資訊管理系教授，同時兼任副研發長，研究方向為深度學習神經網路、資訊安全、網路服務及技術創新與專利佈局之研究。

洪維謙，為崑山科技大學資訊管理系碩士，研究專長為電腦病毒特徵分析、網路入侵偵測與系統弱點掃描之研究。

蔡東霖，現為崑山科技大學資訊管理系研究生，研究專長為電腦病毒特徵分析、網路入侵偵測與深度學習網路之研究。

周明勝，現為崑山科技大學資訊管理系研究生，研究專長為網路入侵偵測、影像辨識與深度學習網路之研究。