Communications of the CCISA
Vol. 29 No. 3 Aug. 2023

基於卷積神經網路與特徵機制之整合式網路入侵偵測告警系統

黃正達[1]、李佳軒[2]、翁浩宇[3]、王尉任[4]
[1]元智大學資訊管理學系、[2]元智大學資訊英語學士班、
[3]國立中央大學資訊工程學系、[4]國立中央大學資訊工程學系
[1]cthuang2020@saturn.yzu.edu.tw、[2]s1093526@mail.yzu.edu.tw、
[3]pony71909@gmail.com、[4]wjwang@csie.ncu.edu.tw

摘要

隨著網際網路的蓬勃發展，人們的日常活動幾乎無一不與網際網路扯上關係，科技雖然帶給人們方便，卻也造成資安事件數量劇增。因此，為了有效防範惡意網路攻擊，在本論文中，我們提出了一個基於卷積神經網路 (Convolutional Neural Network, CNN) 與基於特徵 (Signature-based) 之整合式網路入侵偵測告警系統 (Network-based Intrusion Detection and Alarm System)。在系統中，我們使用開源軟體 Snort 作為基礎的特徵式入侵偵測系統，然而這種基於特徵的偵測技術往往會受到變形多變的攻擊手法而無法有效偵測出攻擊封包，因此我們使用卷積神經網路訓練網路流量分類器來改善 Snort 對於未知攻擊不易偵測之缺陷。在一般的情況之下，Snort 針對於已知攻擊具有良好的偵測效果；而卷積神經網路的分類器則善於偵測未知攻擊，我們得以透過兩種基於不同原理的入侵偵測機制互相配合，以有效地提升網路之安全性，因此我們將兩種機制的入侵偵測系統進行整合，並透過 Elastic Stack (ELK) 進行日誌管理。在實驗結果中，我們使用入侵偵測的標竿資料集 CICIDS-2017 Dataset 進行模型的訓練與測試，從實驗數據中，顯示了本論文的研究方法能夠達到 99.04%的預測準確率，同時，我們能夠透過分類器的預測結果來修改與新增 Snort 規則，以提高 Snort 的偵測率與降低誤報率，基於實驗結果得以說明利用本論文之方法能夠建立一個更加可靠的入侵偵測系統。

關鍵詞：網路安全、入侵檢測系統、深度學習、卷積神經網路、CICIDS2017

Regular Paper

Communications of the CCISA
Vol. 29 No. 3 Aug. 2023

Integrated Network Intrusion Detection and Alarm System based on Convolutional Neural Networks and Feature Mechanisms

Cheng-Ta Huang [1], Chia-Hsuan Lee [2], Hao-Yu Wang [3], Wei-Jen Wang [4]
[1] Yuan Ze University, Information Management,
[2] Yuan Ze University, International Bachelor Program in informatics,
[3] National Central University, Department of Computer Science and Information Engineering,
[4] National Central University, Department of Computer Science and Information Engineering,
[1] cthuang2020@saturn.yzu.edu.tw, [2] s1093526@mail.yzu.edu.tw,
[3] pony71909@gmail.com, [4] wjwang@csie.ncu.edu.tw

Abstract

With the rapid growth of the Internet, people's daily activities are closely tied to it. While technology brings convenience, it also leads to a significant increase in cybersecurity incidents. To effectively prevent malicious cyber-attacks, this paper presents an Integrated Network Intrusion Detection and Alarm System (Network-based Intrusion Detection and Alarm System) that combines Convolutional Neural Networks (CNN) and signature-based feature mechanisms. In this system, we employ the open-source software Snort as the foundation for a signature-based intrusion detection system. However, such feature-based detection techniques often struggle to identify attack packets due to the diversity of evolving attack methods. To address this limitation, we utilize Convolutional Neural Networks to train a network traffic classifier, enhancing Snort's capability to detect previously unknown attacks.

In typical scenarios, Snort performs well at detecting known attacks, whereas the CNN classifier excels at identifying unknown attacks. By integrating these two intrusion detection mechanisms based on different principles, we enhance network security. The two mechanisms are integrated and managed using the Elastic Stack (ELK) for log management. Experimental results using the benchmark CICIDS-2017 Dataset for training and testing demonstrate a predictive accuracy of 99.04% using the proposed research approach. Furthermore, we leverage the classifier's predictions to modify and add Snort rules, thereby increasing detection rates and reducing false positives. The experimental results substantiate that this paper's methodology enables the establishment of a more reliable intrusion detection system.

Keywords: Cybersecurity, Intrusion Detection System, Deep Learning, Convolutional Neural Network, CICIDS2017

壹、前言

處於現今的數位化時代，數據可以是財務紀錄、員工信息、客戶資料、商業或知識產權中的任何內容。而近年來互聯網技術的快速發展，卻使有心人士藉由駭客、惡意軟件和網絡釣魚等方法，對於系統做出攻擊行為並竊取以上敏感資料，導致資安事件的發生情況頻繁增加。針對這些挑戰，入侵偵測系統 (Intrusion Detection System, IDS) 是其中一種安全性防禦工具，主要工作原理通過持續監控網路流量，再進一步分析與辨別是否有潛在的異常或可疑行為。為了事先預防以避免遭受這些惡意攻擊，可以透過網路分析與系統安全維護兩個關鍵策略，於測試中識別系統弱點，加以修補，降低資料外洩的風險。

由於病毒與攻擊行為的不斷演進，傳統的入侵檢測系統越來越顯得力不從心。早期預警措施為重新啟動被入侵的主機並將所有連線至該主機的電腦斷線、將中毒的電腦進行重灌等方法，然而，這些做法不僅可能造成原有資料的遺失，也可能使無法提前警告入侵攻擊事件的發生，並且不易追蹤與蒐集這些事件的證據和日誌。開發一個良好的入侵偵測系統，搭配可以提供即時警訊的告警措施，於受保護的電腦遭到入侵時，能夠提供攻擊追蹤和防範措施，並通知管理者依據突發狀況作出應對。

因此，本論文採用以卷積神經網路 (CNN) 訓練之深度學習模型與 Snort 特徵偵測模式，利用兩種不同的入侵偵測機制，同時分析通過機台網路的封包流量，再將日誌記錄傳至 ELK 管理。根據 ElastAlert 告警規則，倘若日誌信息匹配到相關規則，則發出對應警告信息。讓使用者即時接收和查看攻擊跡象，而容易追蹤和防止網絡攻擊。

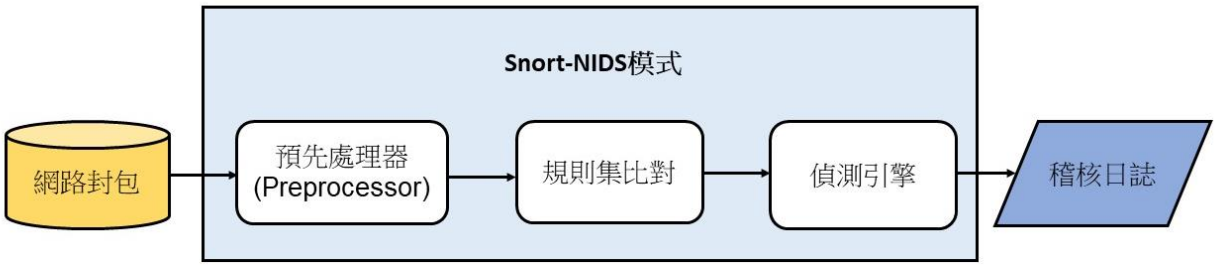
貳、相關文獻

2-1 A Novel Intrusion Detection Model for Detecting Known and Innovative Cyberattacks Using Convolutional Neural Network [1]

在 2020 年，SAMSON HO 等人提出了一個基於卷積神經網路的網路攻擊入侵偵測模型，在他們的方法當中，首先將 CIC-IDS 2017[10]資料集進行前處理 (Pre-Processing)，將帶有缺失值以及數值無限大的資料進行丟棄，接著再將資料進行標準化 (Normalization)。由於套用卷積神經網路進行模型訓練，輸入的資料需要滿足 nxn 的格式，然而，在 CIC-IDS 2017 的資料集當中具有 77 維度的特徵，因此需要透過零填補 (Zero Padding) 的方式將特徵維度重組成 81 維，並將 81 維特徵分解成 9x9 的格式，此時資料得以滿足卷積神經網路的訓練。在他們的方法當中，使用了兩層的卷積層 (Convolution Layer)、兩層池化層 (Pooling Layer)，最後在使用完全連接層 (Fully Connected Layer) 進行攻擊手法的分類。透過他們的卷積神經網路架構進行模型的訓練與測試，實驗結果顯示他們的方法的預測準確率高於一些著名的機器學習演算法如：貝氏分類法 (Naïve Bayes)、支持向量機 (Support Vector Machine)、決策樹 (Decision Tree) 等等。

2-2 Snort NIDS 模式

Snort 為一個開放 (Open Source) 的入侵偵測系統，它可基於已知的攻擊特徵值來檢測任何具有此類特徵的傳入網路流量。其主要目標是為了響應可能危及網路完整性和安全性的惡意活動。廣泛的使用領域包括但不限於企業網路已應對潛在網路攻擊與保障重要數據。也可應用在教育機構，對於學校和大學可保護他們的網路並保護學生和教職員工的隱私與安全。Snort 主要為三種工作模式，分別為 嗅探器模式 (Sniffer)、封包紀錄器模式 (Packet Logger) 以及網路入侵偵測模式 (NIDS)。而本文所討論的方法使用 Snort 的 NIDS 模式，其功能為識別與預定義規則相匹配的流量封包，以確定是否發生匹配條件。將符合規則集比對的數據包記錄至 Log 檔案，產生稽核日誌集作為輸出，指示潛在的安全威脅。如圖一顯示該模式的運作流程，本文方法之 Snort 模擬偵測實驗將在第四章詳細說明。



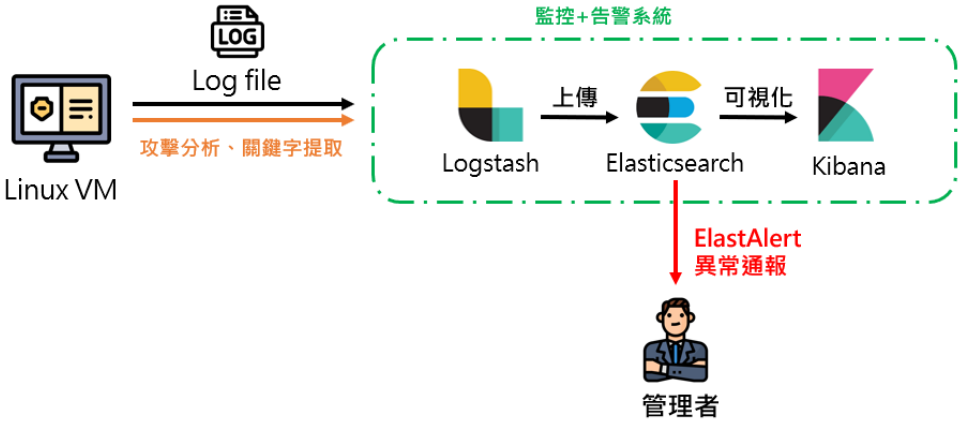
圖一：Snort NIDS 模式運作流程

2-3 ELK 日誌告警

ELK Stack 是一個開源的日誌管理平台，結合了三種產品組成的堆疊：Elasticsearch、Logstash 和 Kibana。Elasticsearch 是一個去中心化的搜索和分析搜索引擎，它集中存儲數據以供檢索和分析。Logstash 是一個服務器端數據處理管道，收集分散的日誌，進行自定義過濾，並將數據輸出到指定位置或服務器存儲。Kibana 是一個免費開源的 Web 前端應用程序，它與 Elasticsearch 配合使用，為存儲在 Elasticsearch 中的數據提供搜索和視覺化，例如儀表板、圖表和其他可視化。而越來越多的 IT 基礎設施遷移到公共雲，ELK Stack 作為一種經濟實惠、靈活且可擴展性的數據分析工具，憑藉存儲和分析大量數據的能力，且高度的客製化功能，允許使用者根據他們的需求設計他們的日誌分析解決方案。並針對潛在的安全威脅或其他問題提供實時警報，可以幫助各種規模的組織改進其 IT 基礎架構監控和管理。而 ElastAlert 是一種第三方插件的警報工具，旨在定期查詢 Elasticsearch，搜尋符合特定條件的數據，之後根據已設定規則觸發告警響應。觸發警報時，ElastAlert 會向指定端點發送通知，例如電子郵件地址、Slack 通道或 Line 聊天室頻道。

本文重點介紹如何使用 ELK 配合第三方插件 ElastAlert，將日誌記錄存儲在 Elasticsearch 中，以供進一步分析。該平台可定制為基於特定關鍵字觸發條件 (使用 KQL 語法)，並在特定

事件發生時向管理人員發送即時電子 Gmail 郵件警報，圖二所示為日誌檔案於 ELK 平台的資料串接流程。

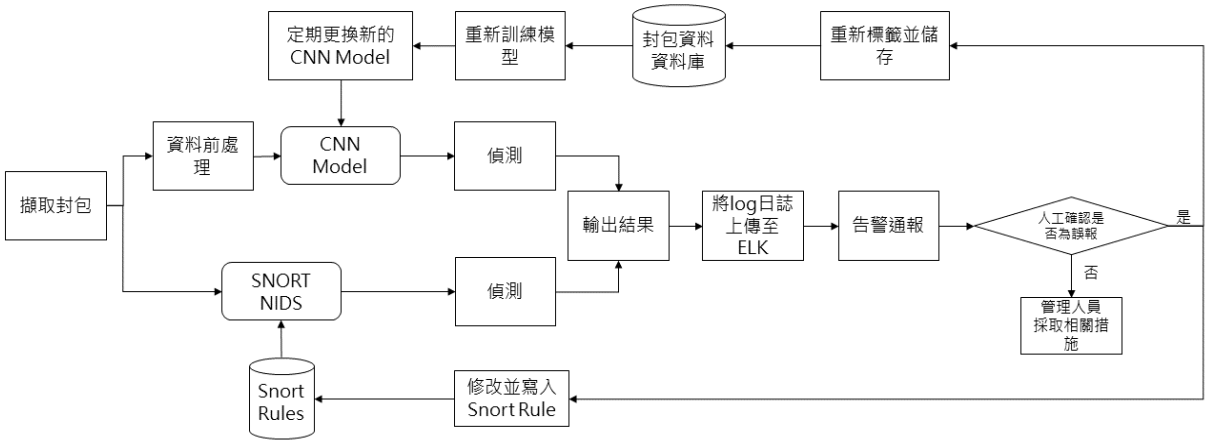


圖二：ELK 平台資料串接流程

參、提出的方法與系統

3-1 系統架構圖

根據 Windows (VM) 中 Snort 的網路入侵偵測模式 (NIDS Mode) 擷取該網域之流量封包，實時監控流量數據並檢測惡意活動。同一時間，將數據包作為預處理資料進行特徵萃取，通過卷積神經網路 (CNN) 演算法所訓練出的系統分類器，判斷是否有攻擊行為的情況。之後通過 Python 程式從產出的日誌文件，提取所需要的關鍵字內容，將其轉換為 CSV 文件格式。之後進一步轉換為分布式搜索和分析引擎 Elasticsearch 需要的特定 JSON 檔案，並使用 SMTP 傳輸協定傳送 JSON 檔案至 Linux (VM) 指定資料夾中。再定期使日誌蒐集及輸出的 Logstash 讀取資料夾內的 JSON 文件，並上傳至 Elasticsearch 進行日誌管理。最後，根據告警規則比對傳輸的 Log 紀錄，倘若發現匹配條件，則向管理人員發出警告訊息。如圖三顯示本論文提出的系統架構圖。



圖三：系統架構圖

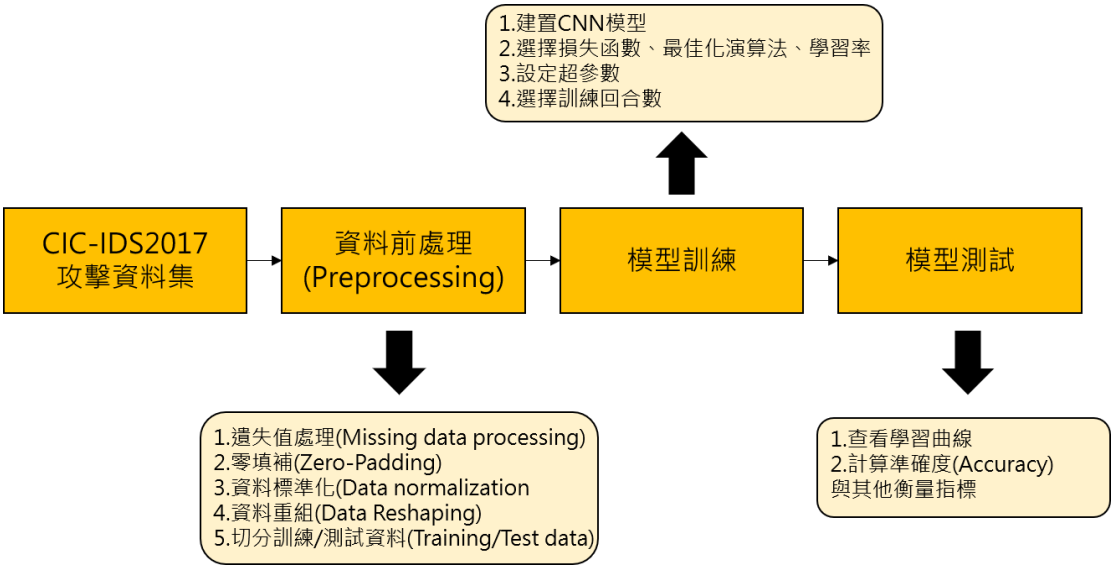
根據整合式系統可能發生的攻擊偵測事件，如表一所示。Snort 能夠良好地判斷已知的惡意流量封包。一旦 CNN 深度學習模型遇上錯誤判斷的情況，可以透過 Snort 規則的正確檢測，新增模型的學習資料並重新訓練模型。反之，CNN 深度學習模型較為準確反映出未知的惡意流量。如果前述情況發生，則修改 Snort 的比對規則，使兩個入侵偵測系統達到整合的實施，以減少高誤報率的發生。而針對 CNN 模型重新訓練、Snort 規則集修改的周期，同樣為一星期一次。

表一：整合式系統可能發生的偵測事件

	假設事件	應對措施
(1)	Snort 偵測到惡意流量封包，CNN 卻沒有判斷到。	將新的 traffic flow 進行 label，重新 CNN 模型的訓練。
(2)	Snort 沒有偵測到，CNN 有正確偵測。	修改與新增 Snort 規則集。
(3)	Snort 跟 CNN 都沒有偵測到，但經調查發現是攻擊行為。	修改與新增 snort 規則集, 針對攻擊的 flow 進行 label 重新訓練 CNN 模型。
(4)	Snort 跟 CNN 都有準確偵測。	最期望的情況， 進行下一步的修補與防禦。

3-2 基於卷積神經網路的入侵偵測模型

卷積神經網路 (Convolutional Neural Network, CNN) 屬於一種深度學習 (Deep Learning) 中的一種演算法，CNN 起初被用來作為的電腦視覺中的辨識領域，並具有非常佳的辨識效果，後來 CNN 被發現在各項不同的應用範疇，如：語音辨識、瑕疵檢測、入侵偵測系統等領域當中也能夠有良好的效果，因此被各個領域廣泛應用。本文方法當中採用 CNN 做為神經網路的演算法，並以網路封包做為模型訓練資料以訓練一個基於 CNN 的入侵偵測模型，並整合到系統當中做使用，圖四顯示了本文方法模型的訓練流程。



圖四：本文方法模型的訓練流程

3-2-1 CIC-IDS2017 攻擊資料集

首先，我們採用網路入侵偵測系統的標竿資料集 CICIDS-2017[10]進行模型的訓練，在 CIC-IDS 2017 的資料集當中，總共包含了 1 種安全連線與 14 種不同攻擊手法的分類項，相對應的資料與筆數如表二所示。在該資料集當中，除了標籤為 BENIGN 的為正常封包以外，其他封包皆為具攻擊行為之封包。我們能看到標前為 BENIGN 的樣本數佔了整個資料集當中的 80.301%，且部分的攻擊樣本數量非常低，例如 Heartbleed 與 Infiltration 的標籤項，因此我們必須先針對資料不平衡 (Data imbalance) 的問題進行處理，我們的作法是將標籤屬於 BENIGN 的樣本進行採樣，任意採樣其中的 20%，即 423934 筆資料進行模型的訓練，調整後的資料集如表三所示。

表二：CICIDS 2017 攻擊資料集的資料分布情形

標籤 (Label)	樣本數 (Samples)	佔比 (Composition)
BENIGN	2273097	80.301%
FTP-Patator	7938	0.281%
SSH-Patator	5897	0.209%
Bot	1966	0.07%
DDoS	128027	4.523%
DoS GoldenEye	10293	0.364%
DoS Hulk	231073	8.163%
Dos Slowhttptest	5499	0.195%
DoS slowloris	5796	0.205%
Heartbleed	11	0.001%

Infiltration	36	0.002%
PortScan	158930	5.615%
Web Attack-Brute Force	1507	0.054%
Web Attack- Sql Injection	21	0.001%
Web Attack - XSS	652	0.024%
Total	2830743	100%

表三：調整後 CICIDS 2017 攻擊資料集的資料分布情形

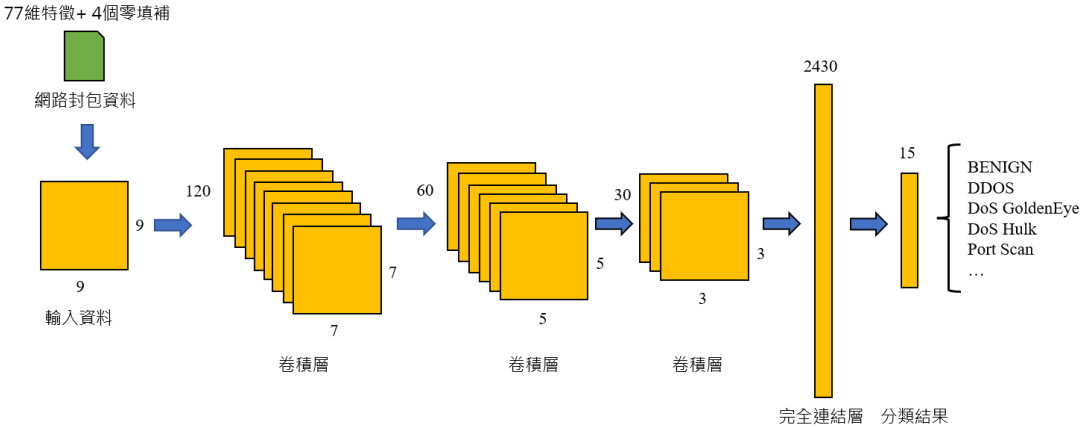
標籤 (Label)	樣本數 (Samples)	佔比 (Composition)
BENIGN	423934	43.19%
FTP-Patator	7938	0.81%
SSH-Patator	5897	0.60%
Bot	1966	0.20%
DDoS	128027	13.04%
DoS GoldenEye	10293	1.05%
DoS Hulk	231073	23.54%
Dos Slowhttptest	5499	0.56%
DoS slowloris	5796	0.59%
Heartbleed	11	0.00%
Infiltration	36	0.00%
PortScan	158930	16.19%
Web Attack-Brute Force	1507	0.15%
Web Attack- Sql Injection	21	0.00%
Web Attack - XSS	652	0.07%
Total	2830743	100%

3-2-2 資料前處理

接著，進行資料的前處理 (Data preprocessing)，在資料前處理的過程中，我們將具有缺失值與包含無限大值的欄位進行丟棄，並將標籤進行編碼，總共 15 個分類項將其編碼為 0-14。完成編碼後，為了使資料滿足訓練 CNN 的輸入格式，我們將資料進行零填補 (Zero Padding)，將 77 維的特徵透過 4 維的零填補，使其重組成 81 維的格式，接著將每一筆的資料轉換成 9×9 的格式。

完成轉換後，我們進行 CNN 模型的建置，本文方法所採用的模型架構如圖五所示。我們使用了三個卷積層 (Convolution layer) 以及一個完全連結層 (Fully Connection Layer)，其中卷積核尺寸使用 3×3，且三個卷積層的特徵圖 (Feature map) 數量分別為 120、60 與 30、學習率設

置為 0.0002，損失函數使用交叉熵 (Cross entropy)，優化器 (Optimizer) 使用 Adam、訓練回合數 epoch 設置為 30 回合、批次數量為 128 筆；其中訓練資料與測試資料切分比例為 80%及 20%。



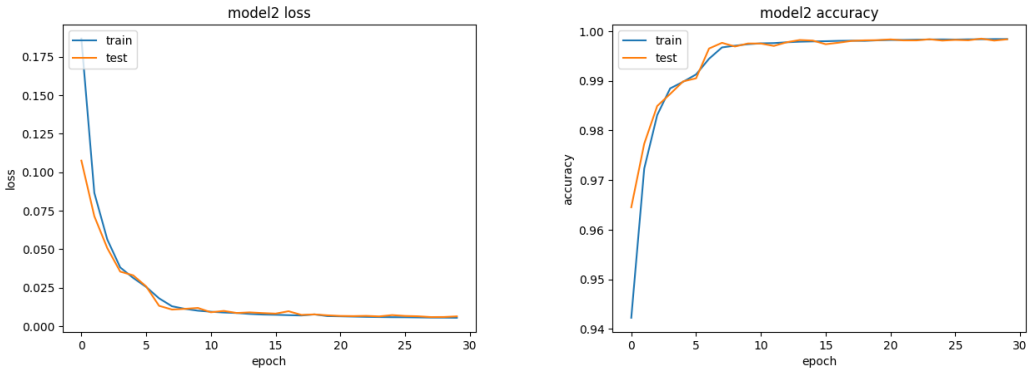
圖五：本文方法的 CNN 模型架構

待相關的參數設定完成後，即開始進行模型訓練，本文方法之實驗數據將在第肆部份實驗結果與討論進行詳細說明。

肆、實驗結果與討論

4-1 CNN 模型的實驗數據

在本節中，將探討本文方法的實驗結果，本文的實作環境使用 Google Colaboratory 作為執行環境，並使用 Google Colaboratory 提供的 NVIDIA Tesla T4 作為模型訓練之 GPU。圖六中子圖 (a) 與子圖 (b) 表明了本文入侵偵測系統的學習曲線，從圖中可以得知訓練的 30 個回合當中，訓練資料與驗證資料都能夠隨著訓練回合數的增加，而降低損失值 (Loss) 並提高準確度 (Accuracy)，能夠說明本文所提出的 CNN-IDS 模型並無發生過度適配 (Overfitting) 之問題，在 10%的驗證資料當中，準確度 (Accuracy) 能達到 99.85%。



(a) 損失曲線　　(b) 準確度曲線

圖六：本文方法的模型學習曲線

針對本文所提出的 CNN-IDS 進行實驗結果之評估，我們使用混淆矩陣 (Confusion Martrix) 與兩種衡量深度學習模型效能的指標：準確度 (Accuracy) 與精確度 (Precision) 進行評估，混淆矩陣示例如表四所示，其中 True Positive (TP) 表示實際上為該類別且同時預測為該類別，即真陽性；False Positive (FP) 表示實際上並非該類別，卻預測成該類別，即偽陽性；False Negative (FN) 表示實際上為該類別，卻預測成非該類別，即偽陰性；True Negative (TN) 則是實際並非該類別，並且也預測為非該類別，即真陰性，我們能夠透過這四項指標計算模型的準確度 (Accuracy) 與精確度 (Precision)，計算方法如公式 (1) 與公式 (2)。

表四：實驗衡量指標–混淆矩陣 (Confusion Matirx)

	True Class (正確結果)		
Predicted Class (預測結果)		**Positive**	**Negative**
	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

表五中顯示了本文方法在測試資料集中的實驗數據，在測試集中包含了 263728 筆的資料，列出了每一種攻擊類型的 TP、FP、TN 以及 FN 的資料筆數，經實驗數據能夠說明本文所提出的 CNN-IDS 能夠達到 99.05%的準確率 (Accuracy)，以及 99.04%的精確率 (Precision)。唯有一些攻擊類型如 Web Attack- Sql Injection、Web Attack-Brute Force 等等，這些類型的資料由於樣本數量過少，因此導致較低的精確度 (Precision)，對於此問題，未來需要透過手動錄製的方式來蒐集封包以補足這些樣本數太少的資料類型。除此之外，本文所提出的 CNN-IDS 模型對於分散式阻斷服務攻擊 (DDoS) 與阻斷服務攻擊 (DoS) 具有良好的預測準確率與精確率。

表五：本文所提出的 CNN-IDS 之實驗數據

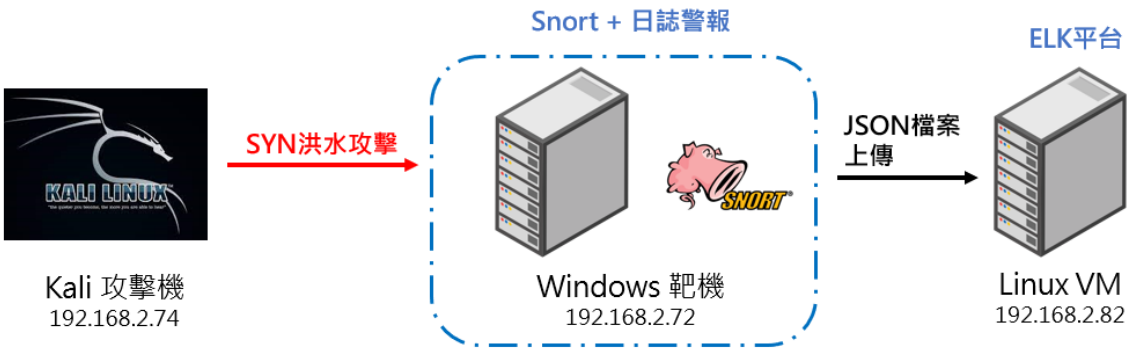
攻擊類型	**TP**	**FP**	**TN**	**FN**	**Accuracy**	**Precision**
BENIGN	104364	1620	156993	751	0.9910	0.9847
Bot	296	55	263280	97	0.9994	0.8433
DDoS	25585	7	238115	21	0.9999	0.9997
DoS GoldenEye	2033	7	261662	26	0.9999	0.9966
DoS Hulk	92429	17	171281	1	0.9999	0.9998
DoS Slowhttptest	1089	7	262598	11	0.9999	0.9936
DoS slowloris	1145	7	262562	14	0.9999	0.9939

FTP-Patator	1582	2	262138	6	0.999970	0.9987
Heartbleed	1	0	263726	1	0.999996	1.0000
Infiltration	6	0	263721	1	0.999996	1.0000
PortScan	31773	1532	230410	13	0.9941	0.9540
SSH-Patator	614	129	262549	565	0.9974	0.8264
Web Attack-Brute Force	269	129	263298	32	0.9994	0.6759
Web Attack-Sql Injection	0	0	263724	4	0.9999	0.0000
Web Attack-XSS	3	2	263596	127	0.9995	0.6000
Overall	**261189**	**2539**	**266189**	**2539**	**0.9905**	**0.9904**

我們將訓練好的模型輸出本文提出的 CNN-IDS 模型輸出，並將其與基於特徵的入侵偵測系統 Snort 同時搭建於內部網路當中，並配合 ELK 作為日誌管理平台可以處理大量數據資料的特色，將所蒐集的 Log 檔案，經過提取關鍵字的相關資料前處理，之後上傳到該平台的 Elasticsearch 進行匹配告警規則的分析，並根據符合條件，觸發並傳送警報訊息。

4-2 Snort 模擬攻擊偵測

圖七中顯示上述的模擬攻擊測試流程圖，於 Windows (VM) 上安裝 Snort 套件作為基本的入侵偵測系統，並使用 Kali 作為攻擊機對其發動 SYN 洪水攻擊測試，以確保 Snort 接收攻擊封包並判斷相關資訊的正確性。測試情境以 IP 位址為 192.168.2.74 的攻擊機，針對已架設 Snort 的靶機使用 hping3 進行 Dos 洪水攻擊，其中靶機 IP 位址為 192.168.2.72。該測試的情境如圖八與圖九所示。



圖七：模擬攻擊測試流程圖

```
┌──(root💀kali)-[/home/kali]
└─# hping3 -c 1000 -d 120 -S -p 21 --flood 192.168.2.72
HPING 192.168.2.72 (eth0 192.168.2.72): S set, 40 headers + 120 data bytes
hping in flood mode, no replies will be shown
^C
--- 192.168.2.72 hping statistic ---
2233804 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

圖八：從 192.168.2.74 針對 192.168.2.72 發動 SYN 洪水攻擊

```
[**] [129:2:2] Data on SYN packet [**]
[Classification: Generic Protocol Command Decode] [Priority: 3]
04/23-12:33:38.843710 192.168.2.74:1282 -> 192.168.2.72:21
TCP TTL:64 TOS:0x0 ID:32366 IpLen:20 DgmLen:160
******S* Seq: 0xBE30ABD  Ack: 0x1D5A6580  Win: 0x200  TcpLen: 20
[Xref => http://www.securityfocus.com/bid/34429][Xref => http://cve.mitre.org/cgi-bin/cvename.cgi?name=2009-1157]
```

圖九：從 Snort 接收到攻擊資訊與 Log File

將 Windows 虛擬機上 Snort 紀錄偵測的警告日誌，利用 Windows 附帶的工作排程器，驅動批次檔案中的 Python 程式，先經過該程式從 Log 檔案提取所需的重要訊息並轉換成 JSON 檔案，再將檔案傳送至 Linux 虛擬機的指定資料夾。後續則是 Linux (VM) 上架設的 ELK 平台中的 Logstash 定期將指定資料夾內的 JSON 檔，推送至 Elasticsearch 進行資料檢索與分析，如圖十、圖十一與圖十二顯示使用工作排程器驅動的自動化封包蒐集。

202304231436	2023/4/23 下午 02:37	Comma Separat...	9,526 KB
202304231436	2023/4/23 下午 02:37	JSON 來源檔案	20,562 KB
202304231436	2023/4/23 下午 02:36	文字文件	30,826 KB
202304231441	2023/4/23 下午 02:42	Comma Separat...	5,246 KB
202304231441	2023/4/23 下午 02:42	JSON 來源檔案	11,333 KB
202304231441	2023/4/23 下午 02:41	文字文件	16,908 KB
202304231455	2023/4/23 下午 02:55	Comma Separat...	5,176 KB
202304231455	2023/4/23 下午 02:55	JSON 來源檔案	11,168 KB
202304231455	2023/4/23 下午 02:55	文字文件	16,840 KB

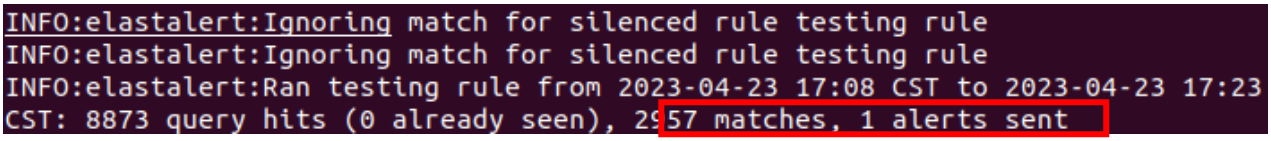
圖十：自動化封包蒐集(1)

```
root@wifi-virtual-machine:/data/ftp/pub# ls
202304231436.json  202304231441.json  202304231455.json
```

圖十一：自動化封包蒐集(2)

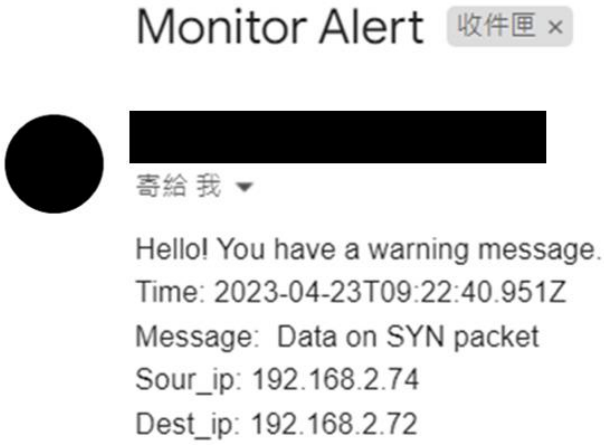
圖十二：自動化封包蒐集(3)

根據 Elastalert 規劃自訂的告警規則，以每 5 分鐘為一週期於實時上傳的數據中，以數據(KQL 語法) 查詢的方式，將數據傳遞至規則類型，並匹配相關的關鍵字訊息。一旦發現符合規則條件，則採取串接至 Email、Line 頻道等告警方式，輸出警報訊息。如圖十三所示為 ElastAlert 匹配關鍵字訊息的畫面，而其中的紅框顯示符合已設定規則條件的數量、告警訊息的廣播次數。



圖十三：ElastAlert 匹配關鍵字訊息

圖十四所示則為傳送 Email 至指定管理人員的詳細告警信息。而告警通報可以從資料前處理中特定提取的關鍵字內容，將幾個重要資訊與警報消息一併輸出。方便管理人員藉由事件發生的相關資訊，從中找到關聯性和漏洞，並加以反省修正。例如圖十四所提及的事件發生的時間戳記、告警日誌中附帶的訊息、事件發生的來源 IP 位址和事件發生的目的 IP 位址。



圖十四：Email 告警訊息

伍、結論

綜合以上所述，本論文所提出的系統是一種基於 Snort 的 NIDS 模式與卷積神經網路 (CNN) 的集成網路入侵偵測系統和告警響應。通過使用開源軟體 Snort 以特徵值篩選作為基本的入侵偵測系統，結合 CNN 所訓練的流量封包分類器，再使用 ELK (Elastic stack) 進行日誌管理。根據兩個系統不同工作原理的協作，以集成系統的互補性質，可以提高對於已知和未知攻擊的檢測。實驗結果表明，本文的研究方法可以達到 99.04%的預測準確率，並且可以透過分類器的預測結果，後續修改和添加 Snort 規則，以提高 Snort 的正確檢測率，降低誤報的機率。目標在駭客利用不同的攻擊手法，造成網路癱瘓或服務中斷等資安事件發生時，提高資訊安全和網路系統防護水平，防止重要數據的丟失或被盜。另一方面，利用優秀且直觀操作的視覺化管理介面，進行實時監控，能夠提供各類型機構更加安全穩定的網路環境，以及提高網路系統的資訊安全。

[誌謝]

This work was partially supported by the National Science and Technology Council of the Republic of China under the Grant No. MOST111-2221-E-155-038.

參考文獻

[1] E. E. Abdallah, W. Eleisah, and A. F. Otoom, "Intrusion Detection Systems using Supervised Machine Learning Techniques: A survey," *Procedia Computer Science*, vol. 201, pp. 205-212, 2022, ISSN 1877-0509.

[2] S. Arra and K. R. Devi, "Evaluation, prediction and implementation patterns of network traffic malware using machine learning," *Materials Today: Proceedings*, 2021, ISSN 2214-7853.

[3] L. Ashiku and C. Dagli, "Network Intrusion Detection System using Deep Learning," *Procedia Computer Science*, vol. 185, pp. 239-247, 2021, ISSN 1877-0509.

[4] T. Daniya, K. S. Kumar, B. S. Kumar, and C. S. Kolli, "A survey on anomaly based intrusion detection system," *Materials Today: Proceedings*, 2021, ISSN 2214-7853.

[5] G. A. Grimes, "Network security managers' preferences for the Snort IDS and GUI add-ons," *Network Security*, vol. 2005, no. 4, pp. 19-20, 2005, ISSN 1353-4858.

[6] S. Ho, S. A. Jufout, K. Dajani, and M. Mozumdar, "A Novel Intrusion Detection Model for Detecting Known and Innovative Cyberattacks Using Convolutional Neural Network," *IEEE Open Journal of the Computer Society*, vol. 2, pp. 14-25, 2021, doi: 10.1109/OJCS.2021.3050917.

[7] Intrusion Detection Evaluation Dataset (CIC-IDS2017). Database. Accessed: April. 1, 2023. [Online]. Available: https://www.unb.ca/cic/datasets/ids-2017.html

[8] A. Kim, M. Park, and D. H. Lee, "AI-IDS: Application of Deep Learning to Real-Time Web Intrusion Detection," *IEEE Access*, vol. 8, pp. 70245-70261, 2020, doi: 10.1109/ACCESS.2020.2986882.

[9] M. Mazini, B. Shirazi, and I. Mahdavi, "Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms," *Journal of King Saud University - Computer and Information Sciences*, vol. 31, pp. 554-553, 2019, doi: 10.1016/j.jksuci.2018.03.011.

[10] K. Salah and A. Kahtani, "Performance evaluation comparison of Snort NIDS under Linux and Windows Server," *Journal of Network and Computer Applications*, vol. 33, no. 1, pp. 6-15, 2010, ISSN 1084-8045.

[11] A. Varanda, L. Santos, R. L. C. Costa, A. Oliveira, and C. Rabadão, "Log pseudonymization: Privacy maintenance in practice," *Journal of Information Security and Applications*, vol. 63, article 103021, 2021, ISSN 2214-2126, doi: 10.1016/j.jisa.2021.103021.

[12] A. Waleed, A. F. Jamali, and A. Masood, "Which open-source IDS? Snort, Suricata or Zeek," *Computer Networks*, vol. 213, pp. 109116, 2022, ISSN 1389-1286.