基於軟體定義網路技術之響應式網路威脅感知與欺敵系統建置

黃文源　鍾旻哲　郭懿萱　鄭欣恬　黃連億　李柏毅

財團法人國家實驗研究院國家高速網路與計算中心

{wunyuan, 2303002, 2103075, n00cht00, 2303021, 1203007}@ narlabs.org.tw

摘要

本論文分享了一個 TWAREN 上響應式網路威脅感知與欺敵系統建置規劃案例。在 TWAREN 骨幹上引入軟體定義網路技術，將流量導至網路威脅感知設備之上分析，當確定為威脅後，便將威脅流量導往特定欺敵之系統上，收集威脅情資，並搭配惡意程式知識庫，將所得之威脅資料分析並儲存，如此一來除了已知的威脅漏洞外，也能獲得未知的網路威脅情資(Cyber Threat Intelligence, CTI)與戰術、技術、程序(Tactics, Techniques, and Procedures, TTP)資訊，有益於各種組織單位的資安防護。

關鍵詞：SDN、Honeynet、Honeypot、TWAREN

1. 前言

網際網路在近十年中蓬勃的快速發展，也使得網際網路上的各種應用與技術相繼出現與提出，例如:大數據傳輸和雲端計算。這些技術的出現，大大改變了現今人們使用設備的習慣，從以往的地端設備慢慢遷移至雲端的虛擬系統，也漸漸習慣了採用雲端上的系統和服務來開發與配置各式各樣的技術和特定服務功能。雲端技術也結合了其它電腦領域的技術，讓原本的能力有了大幅度提升，像是雲端計算與 AI 的結合，減少了運算時間、資料傳輸與儲存的時間。

由於雲端技術的所提供的便利服務，因此增加了許許多多的應用和個人化服務，而這導致全世界的網路使用者數量與頻寬用量暴增，因此服務供應商們也一直增加資料中心（Data center）的建置和擴增網路頻寬，大幅度的提高原本的服務量能，避免因為設備能力不足而流失許多的客戶。

雲端服務的發酵與全球 Data Center 服務量能持續擴增之下，使得與此兩項技術相關的 IOT 主題也趨於熱門。圖1為 IOT 的架構[1]，在此架構之下，Edge Device Layer 中的 End Devices，例如: 智慧型手機, 筆記型電腦, 或是藍芽設備等會藉由不同的協定，例如: 5G, TCP/IP, 或是 Zigbee，透過網路將資料傳送至 Fog Layer，然後 Gateway 會將接收的資料送至 Cloud Layer 的 Data Center 或 Computing Service 來儲存或計算。在此種架構下， Fog Layer 中充當 Gateway 角色的網路設備就非常重要，它不只需要轉送資料外，也必須擁有能相容不同的網路協定，成為 IOT 架構中的骨架。因此 Gateway 的保護不夠完善或是產生漏洞，導致駭客入侵或是出現漏洞攻擊，將會使得資料遺失或竊取，延伸出許多重大影響，例如:公司信譽損毀、詐騙、個資濫用或帳號盜用，因此就必須針對 Fog Layer 的環境下實作資訊安全的防護措施。

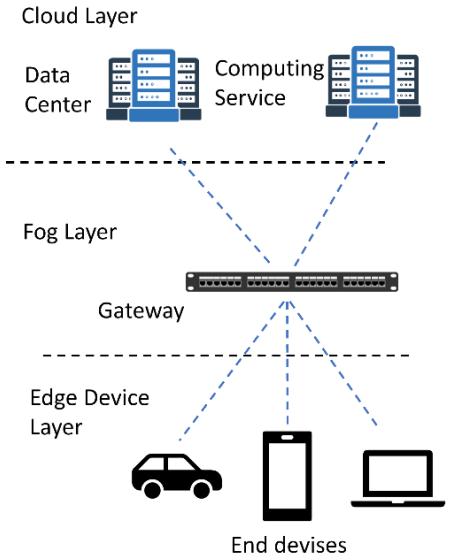

圖 1 IOT 架構

資安防護上，通常藉由分析現有的攻擊樣本了解駭客手法，和防堵設備的漏洞，然而，在時代的進步下，駭客的攻擊方法變得非常多元，進階持續性威脅(Advanced Persistent Threat, APT)也轉為主流，因此樣本的多寡也將會影響著防禦的能力，因此可以採用威脅情資獵取(Thread Hunting)的方式，來獲得更多的威脅情資。TH 的方法中，其中之一是藉由部署誘捕系統來吸引駭客並記錄所有的操作紀錄，經由分析，就能得知其攻擊手法相關資訊，例如:攻擊面(Attack Surface)和攻擊向量(Attack Vector)等，如此一來，便能強化組織單位的資安防護能力。

在台灣，國家高速網路與計算中心，除了有 IOT 架構中代表 Fog Layer 的台灣高品質學術研究網路(TWAREN)和 Cloud Layer 的計算與儲存服務外，也有針對資安防護相關的平台服務。TWAREN 它是為學術研究而設的網路，其擁有著100G 超高速骨幹網路，提供高達100Gbps 網路頻寬，有著多層次的網路服務，這是因為除了傳統網路架構外，也擁有軟體定義網路(SDN)[2]，能自定義特定功能來相容許多的不同網路協定，另外也擁有高容錯備援性網路架構設計，來保障連線的品質。圖2為 TWAREN

現階段的網路架構，TWAREN 是由五個骨幹核心主節點以及12個區網中心(GigaPOP)建構而成的，詳細資訊可以參考 TWAREN 官網[3]。

圖 2 TWAREN 網路架構[3]

本中心的於 TWAREN 提供了資安防護相關的平台服務，服務方面包含了雲端資安攻防平台(CDX)[4]、OWL 惡意程式知識庫[5]與誘捕系統。CDX 提供學研單位申請，讓使用者利用此平台去實驗資安相關技術。OWL 惡意程式知識庫則存放了許多的惡意程式的樣本。誘捕系統如同前面所提，是用來取得威脅情資。

誘捕系統的缺點是，一旦部署建置，就保持固定了，基本上不會再去變動，因此沒有依需求動態部署蜜罐(Honeypot)的彈性存在。本中心的誘捕系統是很久以前建置完成，此系統過於老舊，且也有著彈性不佳的缺點存在，故本團隊參考相關文獻[6][7]，再依據本中心的網路環境和 TWAREN 架構，來建置軟體定義網路技術之響應式網路威脅感知與欺敵系統，將 SDN 技術、資安的威脅感知技術、誘捕網路與 OWL 的惡意程式知識庫的服務結合，藉由響應式網路威脅感知與欺敵系統來取得額外的惡意樣本與威脅情資，從而讓 OWL 擁有更豐富多樣的惡意樣本資訊。

本篇論文中，首先會簡單介紹本中心的 OWL 惡意程式知識庫與部署 Honeypot 的軟體，以及 SDN 的介紹，然後提出系統建置的設計想法與運作流程，最後在說明全部節點的建置規劃與後續的建置計畫。

2. 相關知識

這一章中將會先介紹 OWL 以及和 Honeynet 有關的軟體:MHN，接著會說明 SDN 在網路上其工作原理和架構，並介紹兩款 SDN 中較受歡迎的主控平台:ONOS 與 Ryu。藉由本章的說明，便能簡單的了解這些軟體所需負責的工作。

2.1 OWL 惡意程式知識庫

惡意程式知識庫(Malware Knowledge Base，以下簡稱 OWL)，為國家高速網路與計算中心(NCHC)所研發與建置，如下圖3所示。惡意程式知識庫是一個開放的服務平臺，主要具備惡意程式樣本自動化之蒐集、儲存、分類、過濾及分析機制，並提供樣本快速檢索功能，可便利地進行樣本分類查詢、分析報告查詢、進階查詢及樣本下載，供國內外研究人員與研究單位進行相關的研究與應用。服務示意圖如下圖4所示。

該系統亦結合 Honeynet 誘捕系統，蒐集台灣高品質學術研究網路(TWAREN)上監測與捕捉到的惡意程式攻擊紀錄，並將蒐集到的攻擊紀錄與樣本進行歸納與彙整，經過分析後的樣本報告將整合至惡意程式知識庫，供使用者免費檢索與下載使用。使用者可以透過 OWL 的進階檢索功能，藉由樣本類別與標籤，快速查找到目標樣本，也能將所需樣本，透過批次下載功能一次打包帶走。

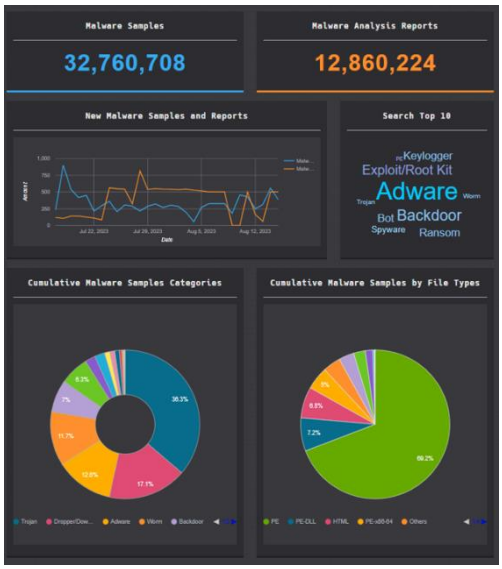

圖 3 惡意程式知識庫首頁畫面圖

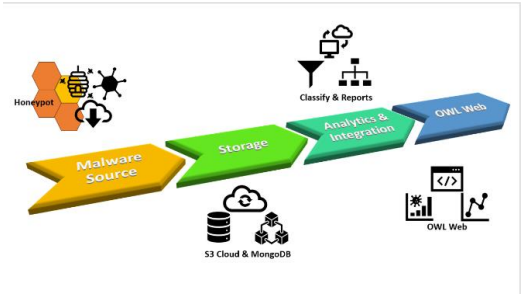

圖 4 惡意程式知識庫示意圖

2.2 Honeypot Platform: MHN

Modern Honey Network(MHN)[8]，是一款能管理與收集作為感測器 honeypot 資料的軟體，它的功能如下:

- 簡單部署 Honeypot: 搭載著 Honeypot 部署腳本，只要執行腳本便能輕鬆部署。
- 威脅情資收集: 整合 MongoDB 與 HPFeeds[9]軟體，HPFeeds 能讓 Honeypot 主動傳送資料至 MHN 上，然後將資料記錄至 MongoDB 之上。
- 威脅情資紀錄展示: MHN 提供網頁介面，從網頁上能檢視收集的各種攻擊資訊，
- 攻擊地圖: 提供攻擊地圖的檢視，當下被攻擊時，能從地圖中得知攻擊者來自世界的哪個地點。
- 第三方軟體的串接: 資料串接方面，其支援 ELK 與 Splunk 的串接。

透過這些功能，使用者將能輕鬆獲得與分析威脅情報。

MHN 支援14種 Honeypot 的部署與資料收集，分別如下所列:

- Conpot: 工業控制系統蜜罐。
- Drupot: 內容管理平台 Drupal 的 Honeypot。
- Magenpot: Magento 電子商務平台的 Honeypot。
- Wordpot: wordpress Honeypot。
- Shockpot: 是一個 Web App Honeypot，目的為尋找試圖利用 Bash 遠端程式碼漏洞 CVE-2014-6271的攻擊者。
- P0f: 用來檢測主機連接方式的工具。
- Suricata: 開源的入侵偵測系統。
- Glastopf: Web Application 的 Honeypot。
- ElasticHoney: Elastic 的 Honeypot。
- Amun: 低互動性的 Honeypot。
- Snort: 知名的開源入侵偵測系統。
- Cowrie: ssh 和 telent 的 Honeypot。
- Dionaea:低互動性的 Honeypot。
- Shockpot Sinkhole:用來偵測 shellshock 和 shinkholing。

儘管 MHN 能支援14種 Honeypot 的部署，然而 MHN 架構與部署腳本均採用 Python 2撰寫而成，因此必須修改設定方能順利部署。

2.3 SDN

軟體定義式網路(Software Defined Network, SDN)是新的網路型態與架構，其當初提出的目的是為了解決傳統網路遇到的問題，例如: 現今 IP address 不足，網路架構複雜等。

SDN 與傳統網路的差別在於，傳統網路的控制平台(Control Plane)和資料平台(Data Plane)是整合在一起，SDN 則是分開的，且 SDN 擁有著可程式化的特性，所以減少了被網路設備商限制的機會和成本。

圖5為 SDN 的網路架構，由下而上分別為基礎架構層(Infrastructure Layer)、控制層(Control Application)與應用層(Application Layer)。基礎架構層主要是 SDN 資料平台(Data Plane): SDN 交換器(SDN Switch)，SDN 交換器由兩部分組成，軟體部分為安全通道(Secure Channel)，負責使用 OpenFlow Protocol 與控制平台通訊，硬體部分則是流表(Flow Table)，裡面存放著許多的串流策略(Flow Entry)，主要是讓 SDN 交換器知道如何處理配對到策略的封包。

控制層部分是 SDN 的控制平台:SDN 控制器(SDN Controller)，主要負責決定針對網路的各種決策，並將處理規則透過 OpenFlow Protocol[10]傳送至 SDN 交換器記錄至流表中成為串流策略，此外也開放 API 與應用程式通訊。應用層則是由使用者撰寫的應用程式組成，他們必須參考 SDN 控制器所開放的 API 文件並實作於應用程式之上，如此才有辦法與 SDN 控制器交流。

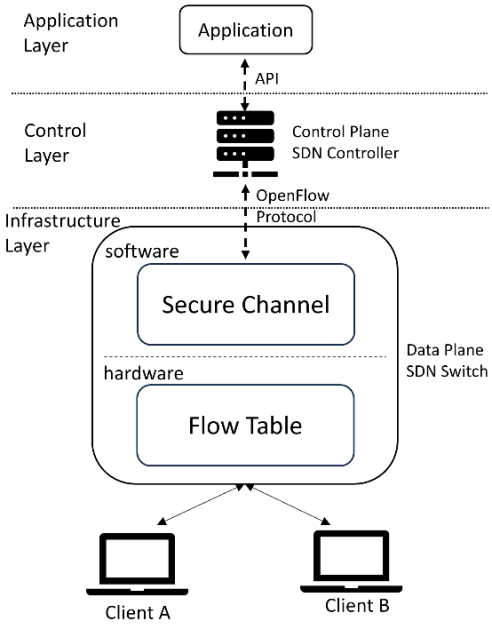

圖 5 SDN 網路架構

SDN 運作流程部分，Client A 送資料串流至 Client B，當 SDN 交換器收到從 Client A 的資料串流中的第一個封包時，會比對流表的串流策略，當沒配對到串流策略，此封包會轉送至 SDN 控制器，SDN 控制器決定串流策略後，便請 SDN 交換器將此策略記錄至流表之中，緊接著將此封包轉送回

SDN 交換器之中，此時封包就會根據串流策略往 Client B 前進，而資料串流的後續封包也將根據此串流策略往 Client B。

SDN 的開放性可程式化特性，可以讓使用者能於控制平台或是應用層上新增特定的功能或應用程式來符合網路建置所需的設定和需求。

2.4 SDN 控制器

現今較熱門的 SDN 控制器為 Ryu[11]和 ONOS[12][13]，其中 Ryu 是由 Python 撰寫而成的。Ryu 是以組件為基礎組成的一個 SDN 框架，Ryu 除了支援 OpenFlow 外，也支援了常用的管理網路協定，例如: NetConf 等，因此使用者能以組件的方式撰寫網路管理的功能或是其他所需之功能，並且直接執行此程式，即可順利的啟動 Ryu 控制器，讓 SDN 交換器順利連線，然後從而管理 SDN 網路與得知網路狀態。

ONOS 是由 JAVA 撰寫而成的，其本身除了充當電信專業等級的 SDN 控制器外，也能作為資料中心虛擬化角色和雲端運算之基礎架構。

ONOS 擁有分散式叢集架構，若一台 ONOS 失效時，將會立刻將工作轉至其他它正常的 ONOS 之上。此外，ONOS 也同樣支援 NetConf 等網路管理協定，因此使用者也能用這些協定來管理 OpenFlow 設備、白牌交換機(White-Box Switches)和傳統網路設備。使用者若於 ONOS 之上撰寫所需功能，則必須透過較繁雜的設定或是新增額外的設定檔案，才能讓實作的功能順利嵌入 ONOS 之內。

ONOS 相對於 Ryu 而言，其架構較為龐大且較為複雜，對於使用者而言較不易上手，因此我們便使用 Ryu 作為 SDN 控制器。

3. 系統架構與運作流程設計

由於先前本中心建置的誘捕系統網路，雖然與 OWL 惡意程式知識庫結合，然而誘捕系統其年代較久遠，系統維護方面較為困難，導致容易出現問題或錯誤，使得誘捕系統與 OWL 惡意程式知識庫之間相容性不佳，兩系統地結合並未能完全發揮出原本預期的功效，反而偵錯上耗費了維運人員大量的時間，故我們決定建置軟體定義網路技術之響應式網路威脅感知與欺敵系統，提升本中心的資訊安全防護能力。

在此系統功能設計考量上，除了必須與 OWL 惡意程式知識庫結合外，我們也參考文獻的實例，將 SDN 導入做為系統網路傳輸的骨幹，與增加威脅感知功能，並根據本中心的網路架構來微調來設計。

圖6為此系統的設計，圖中 SDN 交換器與 External Network、Control Net 與 Honeynet 連接，讓 External Network 進來的流量一開始能傳送至 Control Net，然後根據決策讓流量轉至 Honeynet，而 Control Net 也會與 Honeynet 和 OWL 連接，其目的是於 Honeynet 中動態啟動 Honeypot，以及蒐集威脅情資，之後再把惡意樣本傳給 OWL。

Control Net 主要工作是控制 SDN 交換器的導流、流量分析與蒐集情資，因此裡面需要部署 SDN 控制器、支持威脅感知功能的設備或軟體以及資料蒐集設備。Honeynet 裡面則是由許多的 Honeypot 組成，這些 Honeypot 可以是容器、虛擬機器或實體設備，SDN 控制器會動態的啟動或是部署它們。OWL 部分則是本中心的惡意程式知識庫，他會於特定時間跟 Control Net 中的資料蒐集設備取得 Honeynet 傳回來的各種惡意樣本以及情資。

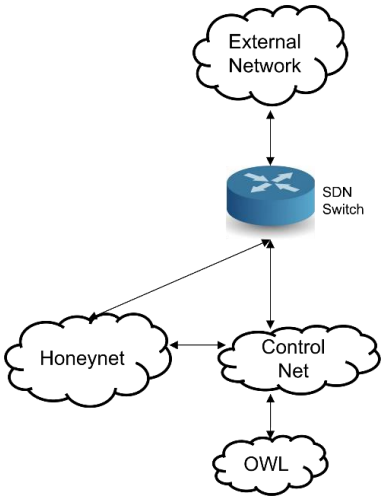

圖 6 誘捕系統網路架構設計

因此在此架構下，我們的整體運作流程如圖7所示，其步驟說明如下:

- Setp1: 當串流的第一個封包進入 SDN 交換器後，由於無法匹配流表上的任一串流策略，因此封包轉送至 Control Net 上的 SDN 控制器。
- Step2: 然後 SDN 控制器設定 SDN 交換器，將此串流轉送至目的地的 Port，並 Mirro 流量至 Control Net 上的擁有威脅感知能力的設備/軟體。
- Step3: 當確認流量為威脅活動的行為，便通知 SDN 控制器執行防護處置。
- Step4: SDN 啟動特定的 Honeypot 並設定 SDN 交換器，將流量導流至已啟動並設定完畢的 Honeypot。
- Step5: Honeypot 將攻擊情資與樣本送至 Control Net 的資料蒐集設備。
- Step6: OWL 與資料蒐集設備聯繫，並下載新的惡意軟體樣本。
- Step7: OWL 取得樣本後便開始分析此樣本，並將分析結果儲存至資料庫之上。

藉由此網路架構與運作流程，便能讓此系統除了順利與 OWL 結合外，也能擁有威脅感知能力與動態的根據需求配置 Honeypot，能減少電力的消耗和硬體資源的佔用。

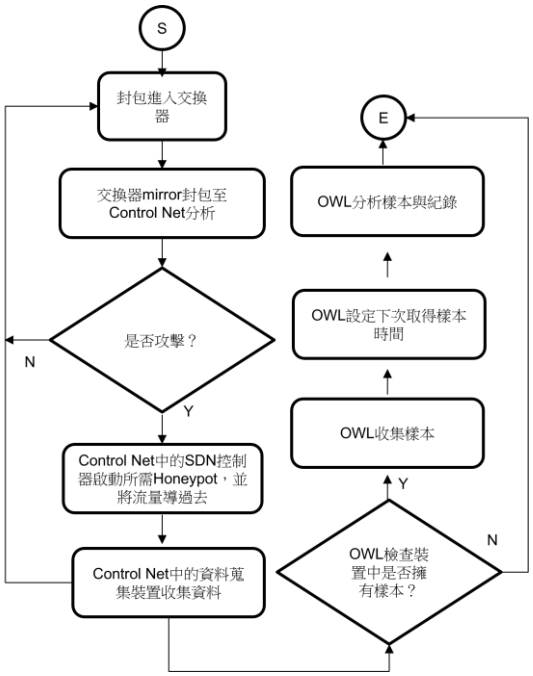

圖 7 運作流程設計

4. 建置規劃

由於本中心有三個分部，分別位於新竹、台中和台南，為了讓本中心資安防護有大的提升，因此我們規劃建置的系統需要部署在這三點上，圖 8 為我們根據前述架構設計所規畫的系統整體建置圖，圖中的規劃為，各節點的均放置一台 SDN 交換器，這三台 SDN 交換器也會採用 VPLS 的連線方式將來串連，而 SDN 交換器也將會如前述設計的架構與 Control Net 和 Honeynet 連接。OWL 連接部分，由於它是位於不同網段的網域之內，因此 OWL 會採用 Tunnel 的方式與三點的 Control Net 連線來獲得惡意樣本與威脅情報。

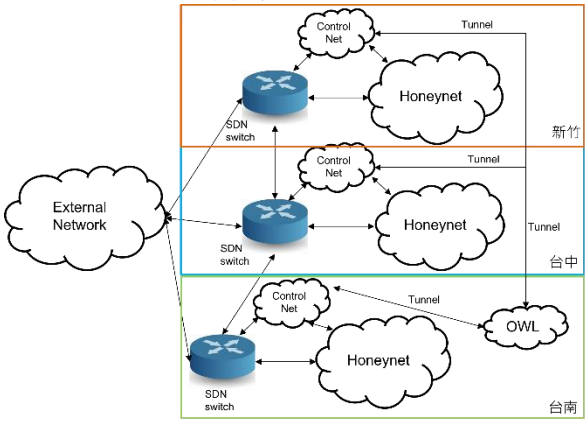

圖 8 系統整體建置圖

今年我們預計先針對台南的節點建置，圖9為台南節點的建置規畫圖，交換器部分將會採用 OVS 來做為 SDN 交換器的角色，Control Net 中各角色的採用的軟體如下:

- SDN 控制器:Ryu。
- 威脅感知軟體:Snort[14]。
- 資料蒐集軟體:MHN。

這些設備的連線方式為，Ryu 與 Snort、Honeynet 和 OVS 互連，Snort 則是與 Ryu 和 OVS 連接，MHN 部分，為了獲得 honeypot 的情資和傳送惡意樣本到 OWL，因此它將會與 Honeynet 和 OWL 連線，OWL 部分則是採用 Tunnel 的方式。

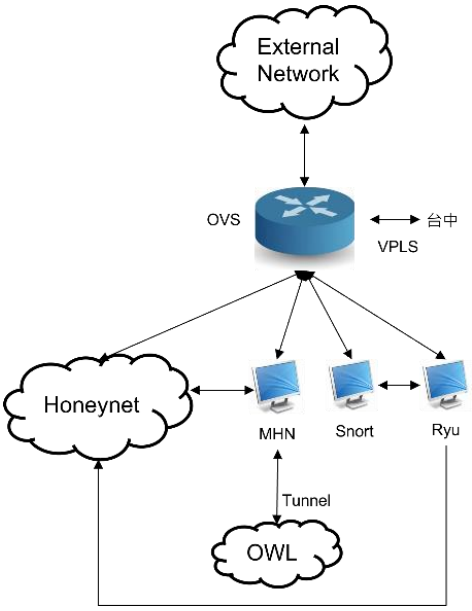

圖 9 運作流程設計

台南節點建置順利完成後，後續將考量依照此規劃持續重建台中、新竹的誘捕系統並根據兩地的網路架構來微調，以便串聯三個節點，遵循整體建置規劃來完成軟體定義網路技術之響應式網路威脅感知與欺敵系統建置。

5. 結論

本論文中，以參考文獻的實例為基礎，於 TWAREN 上規劃與建置跨分部軟體定義網路技術之響應式網路威脅感知與欺敵系統，與 OWL 系統的結合，讓其擁有更多的惡意樣本來源。除了改善了中心老舊的誘捕系統外，並分享了於 TWAREN 上規劃與建置此系統的經驗，能供有同樣有此部署需求的先進或組織單位參考。

未來方面，除了遵循整體的建置規畫外，我們將再改善我們的設計，更加強化與 OWL 的結合或是結合中心其他服務，來讓中心的資安防護能力更加完善。

參考文獻

[1] A3 Menachery, “IoT technology Trends with a Focus on Applications” 2022 IEEE 4th Eurasia Conference on IOT, Communication and Engineering (ECICE), vol. 46, pp. 77-82, Oct. 2022.

[2] Software Defined Network, https://www.opennetworking.org/sdn-definition/

[3] 台灣高品質學術研究網路, https://www.twaren.net/

[4] Cyber Defense Exercise: CDX, https://cdx.nchc.org.tw/

[5] OWL 惡意程式知識庫, https://owl.nchc.org.tw/

[6] 李宇哲、王偉齊、史碩三、黃柏勝、周立德, ” SDN 網路中針對惡意行為之智慧誘捕系統”, TANet2016論文集，花蓮，2016年10月。

[7] Zheng Minjiao, Ma Yufeng, Wu Bo, Qian Zhang, “A Dynamic Deceptive Honeynet System with A Hybrid of Virtual and Real Devices” 2022 5th International Conference on Computing and Big Data (ICCBD), vol. 46, pp. 113–117, Dec. 2022.

[8] Modern Honey Network, https://github.com/pwnlandia/mhn

[9] HPFeeds, https://github.com/hpfeeds/hpfeeds

[10] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, J. Turner, “OpenFlow: enabling innovation in campus networks,” ACM SIGCOMM Computer Communication Review, vol. 46 Issue 2, pp. 69-74, Apr. 2008.

[11] Ryu, https://osrg.github.io/ryu/

[12] P. Berde, M. Gerola, J.Hart, et al. ONOS: Towards An Open, Distributed SDN OS. Proceedings of the 3rd Workshop on Hot topics in Software Defined Networking. ACM, 2014: 1-6.

[13] ONOS, https://opennetworking.org/onos/

[14] Snort, https://www.snort.org/