# 以軟體定義網路實現具手機遠端警示與控制之網路入侵偵測與防禦系統


鄔啟瑞
臺中教育大學資工系
transworldwide576@yahoo.com.tw

林志仁
臺中教育大學資工系
zx21940@gmail.com

曾煒婷
臺中教育大學資工系
j830902@yahoo.com.tw

李宗翰*
臺中教育大學資工系
thlee@mail.ntcu.edu.tw

張林煌
臺中教育大學資工系
lchang@mail.ntcu.edu.tw



## 摘要

在傳統的網路架構中，網路路由器須各別設定使用者所需的功能，在設定上的複雜度、所需設定的網路設備數量，容易因人為疏失導致網路中斷，且因多數的網路路由器，其軟、硬體架構均為封閉式系統，導致不同廠牌網路設備間的系統相容性差，造成設備管理上的負擔。有鑑於此，本計畫將對針對一般市售的無線網路路由器進行修改，使其成為具備 SDN 能力的網路路由器，並透過 Snort IDS，將一般市售網路設備也能如同企業級高端路由器擁有網路入侵偵測與防禦系統，同時本專題也在 Android 行動裝置上開發能進行遠端監控與配置路由器的 APP，使一般不具備相關網路專業知識的使用者，也輕易與方便的進行 SDN-enabled Switch 的遠端監控與配置。



**關鍵詞**：軟體定義網路、入侵偵測系統、Android。



## Abstract

In the conventional network architecture, network routers needs setup each function individually. In the more complex function configuration, it easily leads to network interruption due to human error. In addition, the closed systems in the firmware of network devices resulting in the lack of system compatibility and management between different brands of network devices. Therefore, this project was modified ordinary wireless routers to SDN and IDS enabled switches by porting openFlow and Snort IDS. Also, the Android-based smart phone can remote monitoring and configuring the SDN-enabled Switch by the proposed Android APP. Thus, users can monitor and configure the SDN-enabled switch remotely do not have the related expertise of network.



**Keywords**: Software Defined Networks, Intrusion detection system, Android.


## 1. 前言

在現今的網路架構下的路由器包含了控制層(control plane)與資料層(data plane)，各個路由器透過內部的 routing rule 在現今的網路下傳送資料，因此使用者必須了解與如何熟知各種網路參數的設定。例如在設置上需要將每台路由器逐一設定，況且不同廠牌有著不同的管理介面，導致並非一般使用者均可自行設定這些路由器內的功能，而採用內建的原始參數，如此反而造成網路安全上的疑慮。

Software Defined Networks (SDN)[1]是透過軟體定義來改變網路架構與機能，與現今網路最大的差別在於，它改變了網路的控制模式，將原先網路管理的功能交給控制層的 SDN Controller 負責。開發者可以開發應用軟體部屬於控制器內，再透過 OpenFlow 與 Controller 傳達並下達指令給資料層的網路裝置，例如 SDN-enabled Switch。SDN-enabled Switch 則負責封包的傳送。SDN 可以自動控制路徑，簡單而言就是自動化。透過此項技術能夠更有效率地運用在網路管理上，為企業省下不少的網路管理的成本支出。不過上述優點必需要透過能支援 SDN 網路技術的 SDN-enabled Switch 才能進行，而這種 SDN-enabled Switch 往往價格不斐。

有鑒於此本計畫將對一般市售的無線網路路由器改寫成具備 OpenFlow 協定的 SDN-enabled Switch，並開發 REST API[2]使行動裝置上也能透過 SDN Controller 進行 SDN-enabled Switch 的遠端監控與配置，提供 SDN-enabled Switch 更即時與便利的服務。本計畫將著重在開發具 SDN 功能之無線網路路由器，移植 openFlow 協定架構至一般市售的路由器(例如 TP-Link 1043 ND[3])，並透過 REST API 監控 SDN Controller 對所開發之可程式化路由器進行控制。

## 2. 系統架構

如圖 1 所示，本專題的系統架構由四個主要的子系統組成，第一個為 Ryu SDN Controller，Controller 可以控制底下 SDN-enabled Switch 封包之間的傳送；第二個是 OpenFlow SDN-enabled Switch，本專題使用 TP-Link 1043ND，並改寫其韌體以支援 OpenFlow 協定；第三個是入侵偵測系統，主要用來偵測透過 Switch 傳輸的封包當中，是否有攻擊的存在；第四個為使用者的 SDN 智慧型手機 APP，當入侵偵測系統偵測到疑似攻擊的時候，將會傳送警告訊息(alert)通知使用者。

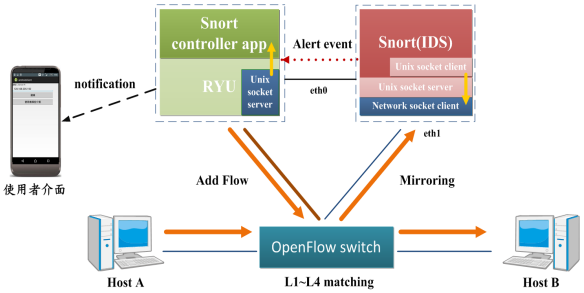


**圖 1. 系統架構**

### 2.1 運作流程

透過 SDN Controller 設定 SDN-enabled Switch 複製所有經過交換器的網路封包，以交換器上的 port 3 作為監聽，將封包資訊傳送至入侵偵測系統進行相關比對。

入侵偵測系統以封包特徵進行比對，判斷是否為攻擊或異常封包，如為異常或攻擊封包，即透過 Network socket 傳送警訊至 SDN 智慧型手機 APP。

SDN 智慧型手機 APP 在收到 Alert Message 之後，會將其拆解成三個部分，包括警告語、攻擊端 IP 與被攻擊端 IP 並將它們寫入資料庫供使用者查閱，且同時根據 IP 位址與通訊協定下達 Flow Enrty 進行封包的阻斷。

### 2.2 OpenFlow Switch

本專題使用 OpenWrt[4]以及 Open vSwitch (OVS) [5]來完成開發，SDN 實驗設備如圖 2 所示。OpenWrt 是一個適合於嵌入式裝置的 Linux OS，使用者可以模組化選擇相關的應用程式和套件，而不必受裝置提供商的限制，並可以使用一些適合某方面應用的軟體包來客製化系統。在本專題中，第一步驟是將 TP-Link 1043ND 的韌體刷新成 OpenWrt，再將 Open vSwitch 安裝於 OpenWrt 上，透過 Cross-Compiler 編譯適用於 TP-Link 1043 ND 且具 OVS 的韌體，待編寫完韌體時，再將原本 TP-Link 1043 ND 中的韌體替換成編寫出來的新韌體，更換完成後，TP-Link 1043 ND 將具備支援 SDN 功能。

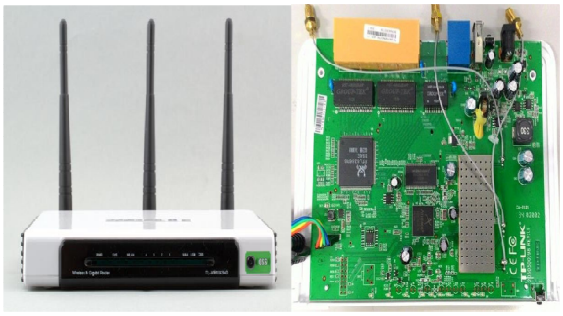

**圖 2. SDN-enable Switch**

### 2.3 Ryu Controller

如圖 3 所示，本專題使用的 SDN Controller 為 Ryu 以支援 OpenFlow1.3 版本的相關協定。Ryu 是一個 component-base 的 SDN Controller，主要提 well-defined API 讓開發者更容易開發網路管理與控制應用，component-base 的設計模式能幫助使用者達到特定的需求，開發者可以快速的修改現存的 component 或是再創建其他 component 來確保底下的網路會照著應用程式所指定的行為運作。

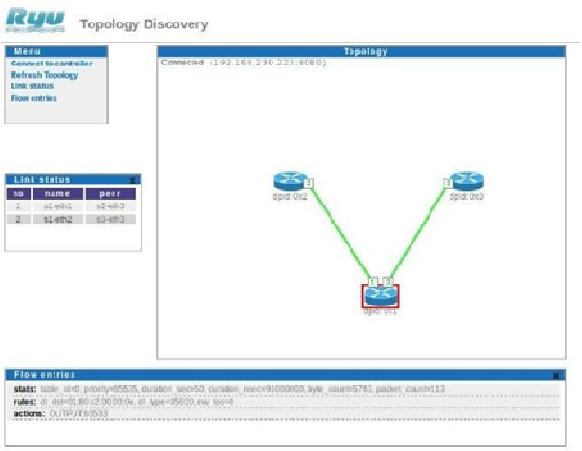


**圖 3. Ryu 之管理者圖形介面**

本專題利用 Ryu 提供之 REST API 下達 Flow Entry，當使用者透過 SDN 智慧型手機 APP 接受到由入侵偵測系統發出的警告訊

息時，可以透過 REST API 傳輸指令到 Ryu，阻擋發生問題的 Switch 有關於網路方面的權限，例如發現是某 IP 正傳輸大量封包給某一組 SDN-enabled Switch，Ryu SDN Controller 將可以立即設定受攻擊之 SDN-enabled Switch 的防火牆系統阻擋該 IP 與 Switch 之間的網路傳輸，進而阻擋來自該 IP 的攻擊。

## 2.4 Snort 入侵偵測系統

本專題採用 Snort [6]作為 IDS (Intrusion detection system)之核心，將所有通過 SDN-enabled Switch 的封包進行複製，並即時傳送至 Snort 中，Snort 會在分析封包的過程中，逐一比對 Snort Rule 中描述著各種網路攻擊的型態，例如：DDoS，SQLinjectio…等。當發現疑似有網路攻擊的行為特徵時，會將該訊息透過 Unix Domain Socket 建立 Snort 與 Ryu SDN Controller 間的橋梁，透過 Network Socket 傳送 Alert Message 給 Ryu。如圖 4.所示，Ryu 將立即接收來自 Snort 的入侵封包警告訊息，當 Ryu 收到 Alert message 時，就會自動將警訊寫入資料庫並同時下達 Flow Entry 進行阻斷。

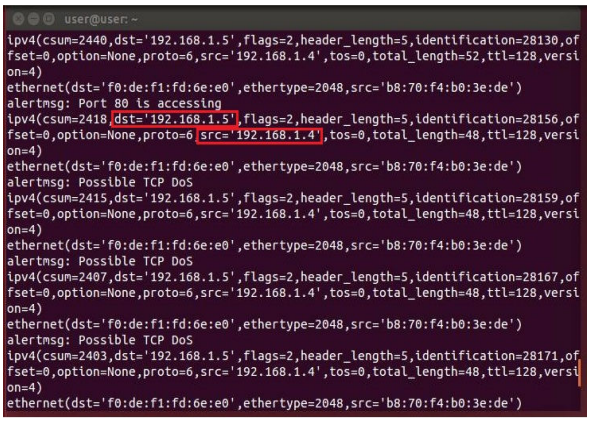


**圖 4. Ryu 接收 Snort 的入侵封包警告訊息**

## 2.5 遠端管理 APP

本專題將 SDN Controller 的監控與配置功能，同時實現在可攜式的行動裝置上，且不受裝置系統的使用限制，例如 Android 智慧手機噢平板電腦。因此本專題將以 Android APP 呈現在行動裝置上，讓裝置只具有連網能力就可與 SDN Controller 連線，故在此採用 Ryu 所提供的 REST API 開發元件，並以 REST API 所提供的 HTML5 框架，動態呈現出目前 SDN Controller 上各 SDN-enabled Switch 之即時狀況，透過 Ryu 所提供的 REST API 來與遠端 Controller 進行溝通，可攜式裝置只需使用支援 HTML5 的瀏覽器，就可以透過網頁來與 Controller 溝通，並且讓裝置點擊 Android APP 即可連接至上述所開發的 REST API，而當攻擊發生時，Ryu 會發出攻擊訊息給 Android APP 來通知使用者，圖 5 為 Ryu REST API 與 Android APP 的連線示意圖。

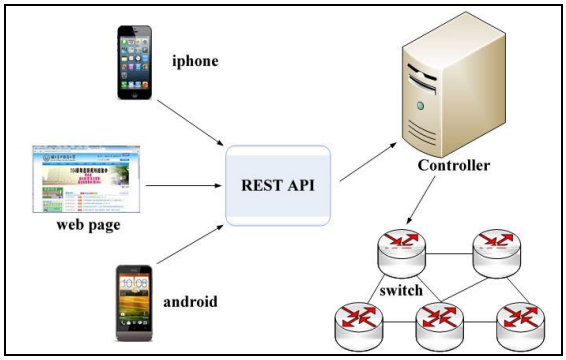


**圖 5. Ryu REST API 與 Android APP 連線示意圖**

## 2.6 REST API –Flow Entry

當使用者收到 APP 所送來的即時入侵警告訊息時，透過 Android APP 使用者可查看所受到的攻擊紀錄以及 Ryu 針對攻擊自動下達的 flow entry。如使用者確認此類型封包並非是來自網路的惡意攻擊，可以透過 Android APP 將被隔離起來的 IP 加以還原。使用者也可以透過下達指令去新增刪除存在於 Flow Table 當中的規則(entry)，能更方便的遠端操控 SDN Controller。

## 3. 實驗結果

在此章節我們將介紹 SDN-enabled Switch 與 Snort 的連線架構、Controller Ryu 與入侵偵測的系統整合與 Ryu 與智慧型裝置 APP 的溝通等實驗結果。

## 3.1 整合 SDN-enabled Switch 與 Snort 之網路架構

在本專題中，我們將 Snort 安裝在一台主機上，作為 IDS 的核心，該主機需要具備二張網卡，eth0 連接到 Internet，讓 Snort 透過 network socket 傳送 alert 至 Ryu。，而 eth1 連接到 SDN-enabled Switch 的實體 port 3，用以複製流經 Switch 封包，我們使用兩台 PC 分別

連接到 SDN-enabled Switch 的實體 port 1 及 port 2，如圖六所示：

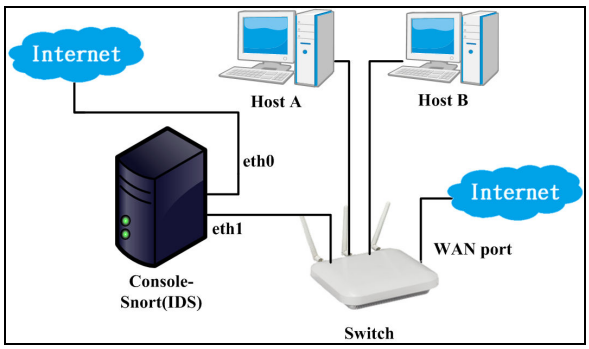


**圖 6.** SDN-enabled Switch 與 Snort **網路架構圖**

### 3.2 整合 Ryu 與入侵偵測系統

RyuSDN Controller 除了會賦予 Switch 基本的封包傳輸功能，也會將流經 Switch 的封包導向至 IDS，而我們以自行修改 Ryu 使其可以驅動 SDN-enabled Switch 自動阻擋網路攻擊，並以手機 APP 通知使用者。專題使用 Snort 與 Ryu 協同作業來完成入侵偵測與防禦的工作，Snort 會將封包與自行定義的 rule 做比對，當發現可能的網路攻擊時，Snort 會發出 alert 至 Unix domain socket，再透過 network socket 傳送給 Ryu。並根據 alert 上的資訊寫入 database 中並下達 flow entry 至被入侵的 Switch 並加以阻擋網路攻擊。其實驗成果如圖 7 所示。

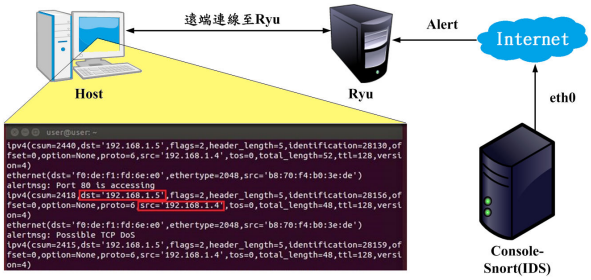


**圖 7. Snort 發出的 alert 封包資訊顯示在 Ryu 上**

### 3.3 Ryu 與智慧型裝置之溝通

在計畫中，希望在 Ryu 接收到來自 Snort 發出的警告訊息時，可以即時的發出通知到使用者的智慧型裝置上，例如: 手機。因此，Android App 會與 Ryu 建立一條 TCP 連線，在此連線不斷線的前提下，Ryu 收到警告訊息時，可以傳送 socket 給手機，手機會立即將收到的警示推播通知使用者，使用者可以透過此通知知道目前哪台 SDN-enabled Switch 或是電腦正遭受攻擊，除 SDN Controller 會主動立即處理可疑 IP 的封鎖動作外，在 Android APP 中也可以簡易的圖形介面下達 Flow Entry 查看 Ryu 中 Flow Table 之功能，讓使用者可以針對被攻擊主機所連線到的 SDN-enabled Switch 下達進一步的阻擋規則，也可透過此 App 進行規則的新增及刪除。

## 4. 結論

本專題成功的將 SDN 技術實現在一般的無線網路路由器中，能以 OpenFlow 協定在一般市售路由器下，透過 SDN Controller 來控制所有底下的 SDN-enabled Switch，實現 SDN 入侵偵測與防禦系統。也成功結合 Ryu 與 Snort，使 Ryu 可以自動的根據 Snort 發出的警訊，依照被攻擊的類型去自動下達封鎖動作，不需使用者手動執行下達指令的步驟，已達到更迅速地抵擋來自網際網路上的攻擊。

在未來研究方面，由於專題所使用的 Snort 是架設在一台實體主機上，並利用內網接收來自 SDN-enabled Switch 的封包，這樣的架構套用在大規模數量的 Switch 會產生許多額外的成本與複雜性。因此，在未來希望能夠做到以 Snort 嵌入式系統結合 SDN-enabled Switch，除了被動遭受攻擊再產生阻斷指令來抵禦網路攻擊之外，也希望能夠根據忘錄攻擊的特徵與歷史資訊來作為能夠預先抵擋攻擊的智慧型學習與分析系統。

## 5. 誌謝

本計劃之執行承蒙科技部專題研究計畫(編號：104-2221-E-142 -002)及 科技部大專學生研究計畫(編號：104-2815-C-142-002-E)之支持，特此致謝。

## 參考文獻

[1] “Open Networking Foundation, SDN definition”[online].J. Breckling, Ed., The Analysis of Directional Time Series: Applications to Wind Speed and Direction, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61

[2] “Wikipedia, REST” [online]. http://en.wikipedia.org/wiki/Representational_state_transfer

[3] “TP Link 1043 ND”[online]. http://www.tplink.tw/products/details/?model=TL-WR1043ND
[4] ”OpenWrt”[Online]. https://openwrt.org/
[5] Open vSwitch – An Open Virtual Switch. http://www.openvswitch.org, September 2014.
[6] “Snort” [Online] https://www.snort.org/