行政院國家科學委員會專題研究計畫 成果報告

異質無線多網安全檢測平台建置計畫
研究成果報告(完整版)

計 畫 類 別 ：個別型
計 畫 編 號 ：NSC 98-2219-E-009-003-
執 行 期 間 ：98 年 01 月 01 日至 98 年 12 月 31 日
執 行 單 位 ：國立交通大學資訊工程學系（所）

計 畫 主 持 人：謝續平
共 同 主 持 人：曾文貴、黃能富、楊武、黃育綸、趙禧綠
黃世昆
計畫參與人員：碩士級-專任助理人員：陳柏愷
碩士級-專任助理人員：卓政逸
學士級-專任助理人員：郭明華
學士級-專任助理人員：謝政翰
碩士班研究生-兼任助理人員：黃琨翰
碩士班研究生-兼任助理人員：邱世欣
碩士班研究生-兼任助理人員：黃佑鈞
碩士班研究生-兼任助理人員：許國祥
碩士班研究生-兼任助理人員：張詠承
碩士班研究生-兼任助理人員：彭博群
碩士班研究生-兼任助理人員：黃啟彥
碩士班研究生-兼任助理人員：王雅萱
碩士班研究生-兼任助理人員：許鴻生
碩士班研究生-兼任助理人員：鄭偉強
碩士班研究生-兼任助理人員：黃晉澤
碩士班研究生-兼任助理人員：黃錦銘
碩士班研究生-兼任助理人員：顏豪緯
碩士班研究生-兼任助理人員：官振傑
碩士班研究生-兼任助理人員：宋穎昌
碩士班研究生-兼任助理人員：施汎勳
碩士班研究生-兼任助理人員：劉雨芊
碩士班研究生-兼任助理人員：徐蘇偉

碩士班研究生-兼任助理人員：朱信儒
碩士班研究生-兼任助理人員：張書綸
碩士班研究生-兼任助理人員：游釗俊
碩士班研究生-兼任助理人員：朱慶峰
碩士班研究生-兼任助理人員：江孟寰
碩士班研究生-兼任助理人員：蘇修醇
碩士班研究生-兼任助理人員：王嘉偉
碩士班研究生-兼任助理人員：劉芳瑜
碩士班研究生-兼任助理人員：許銘佩
碩士班研究生-兼任助理人員：林晏蔚
博士班研究生-兼任助理人員：張宏義
博士班研究生-兼任助理人員：高迦南
博士班研究生-兼任助理人員：蔡欣宜
博士班研究生-兼任助理人員：甄元彬
博士班研究生-兼任助理人員：林煥宗
博士班研究生-兼任助理人員：沈宣佐
博士班研究生-兼任助理人員：林孝盈
博士班研究生-兼任助理人員：林佳純
博士班研究生-兼任助理人員：王繼偉
博士班研究生-兼任助理人員：李秉翰
博士班研究生-兼任助理人員：許家維
博士班研究生-兼任助理人員：姜淑華
博士班研究生-兼任助理人員：郭子綺

報 告 附 件 ：出席國際會議研究心得報告及發表論文

處 理 方 式 ：本計畫可公開查詢

中　華　民　國　99 年 03 月 24 日

行政院國家科學委員會補助專題研究計畫 ■ 成 果 報 告 □期中進度報告

資通安全人才培育-國立交通大學資通安全研究與教學中心

計畫類別： □ 個別型計畫 ■整合型計畫
計畫編號： NSC 98-2219-E-009-003
執行期間： 2009 年 1 月 01 日至 2009 年 12 月 31 日
計畫主持人：謝續平
共同主持人：曾文貴
協同主持人：黃能富、黃世昆、黃育綸、楊武、趙禧綠、吳育松

成果報告類型(依經費核定清單規定繳交)：□精簡報告 ■完整報告

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、列管計畫及下列情形者外，得立即公開查詢
□涉及專利或其他智慧財產權，□一年□二年後可公開查詢

執行單位：國立交通大學資訊工程學系

中 華 民 國 99 年 03 月 31 日

摘要

隨著無線網路的盛行，無線網路攻擊行為層出不窮，這對政府機關、財團法人與高科技廠商已經造成重要內部資訊的洩露及金錢的嚴重損失。對政府機關 (例如：國安局、中科院等) 而言，期待有全面性的安全檢測工具可檢測其內部所使用的無線網路設備及無線軟體是否有安全漏洞及弱點。而財團法人 (例如：資策會、工研院等)所推行的校園無線漫遊整合計劃及 WiMAX 科學園區建置計劃也希望有合適的安全檢測工具能夠檢測與滲透分析無線行動裝置、Base Station (BS)及無線漫遊伺服器 (Roaming Server)的安全性。產業界 (例如：中華電信、微軟、聯發科技、友訊、威播、明泰、宏碁、阿碼科技等)而言，希望能夠有完整的檢測工具可以檢測他們開發的無線設備或者無線設備內的系統與應用軟體上是否存在安全漏洞。然而目前市面上並沒有完整及合適的安全檢測工具可以提供檢測服務給上述單位。為了滿足政府機關、財團法人與高科技廠商對於無線網路安全檢測服務的迫切需求，本計畫在 98 年執行初期便邀請工研院、資策會、國安局、中科院、明泰科技等單位共同協助規劃，建置一異質無線多網安全檢測實驗室，並開發一異質無線多網安全檢測平台 (WiSec@NCTU)，目前此平台可分成異質無線多網核心網路安全檢測以及行動裝置滲透檢測兩大部份。在 98 年的計畫執行期間我們建置與開發總共 13 個子系統與工具用以檢測異質多無線網路 (WiFi、WiMAX 及 3.5G)與有線網路（wired）互動下無線網路設備、無線行動裝置、軟體程式的安全性。在 98 年，本計畫有豐碩的成果。我們在 98 年發表於國際期刊之論文共有 8 篇，而發表於國際研討會之論文數共 7 篇。另外，我們在技術移轉與產學合作方面的成果包括技術移轉（軟體授權）共 5 件，總金額為 290 萬、技術服務共 4 件，總金額為 220 萬、產學合作共 6 件，總金額為 877.5 萬。另外我們也提供了 5 件檢測服務。藉由此平台的建置與檢測工具的開發，我們可提供政府機關、財團法人及高科技廠商無線網路安全檢測的服務，幫助上述單位發現漏洞及弱點。此外，我們所建置的工具可為產業界創造上億元以上的經濟效益、提升無線產品附加價值、節省因網路攻擊或系統弱點所消耗的產值、節省專業檢測人力並且有效減少無線網路環境的攻擊。

關鍵字: 異質無線多網安全檢測、行動裝置滲透檢測、惡意軟體檢測、軟體程式安全檢測

Abstract

With the increasing prevalence of wireless networks, numerous attacks have occurred frequently these years. These wireless network attacks have great impact not only on government agencies, but also on private sectors. The illicit behaviors may cause sensitive information leaked and serious monetary lost. Unfortunately, the current tools are not designed for the security testing of wireless systems and software. To ensure our efforts will fit into the need of government agencies, research institutes, and wireless equipments manufactures, the engineers and technical leaders of ITRI, III, CSIST, NSB, and Alpha Networks have been invited to participate the planning of this project at the beginning stage. The aim is to construct a heterogeneous wireless multiple network penetration testing lab, and develop a heterogeneous wireless penetration testing platform (WiSec@NCTU). We have developed and established thirteen penetration testing systems and tools. These systems and tools can be divided into two categories: wireless network penetration test and mobile devices penetration test. These developed subsystems and tools can examine and test security issues on heterogeneous wireless network (WiFi, WIMAX, and 3.5G), wired wireless devices, mobile wireless devices and software. TWISC@NCTU has close relationship with Taiwan high-tech companies, government, and research institutes. In respect to technology transfer, we have transferred five novel technologies to National Security Bureau (NSB), Industrial Technology Research Institute (ITRI), and Chunghwa Telecom. In regard to industrial collaboration, Chung-Shan Institute of Science & Technology, MediaTek, ITRI, and D-Link are currently involved in our center. TWISC@NCTU also provides technical services to NSB, ITRI, Taipei Computer Association (TCA), and Institute of Information Industry (III). We hope government agencies and Taiwan industries can benefit by using these tools to examine and evaluate their wireless networks or mobile devices.

Keywords: heterogeneous wireless network penetration test, mobile devices penetration test, malware detection, software security analysis

一、 背景

以下將針對本計畫之背景，分成國內需求現況、異質無線多網安全議題與行動裝置安全議題三大部份來分別介紹。

● 國內需求現況

現代人追求無線的便利性，促成無線網路的盛行。然而無線網路攻擊行為層出不窮，這對政府機關、財團法人與高科技廠商已經造成重要內部資訊的洩露及金錢的嚴重損失。對政府機關(例如：國安局、中科院等) 而言，會希望有全面性的安全檢測工具可檢測其內部所使用的無線軟體及無線網路設備是否有安全漏洞及弱點。而財團法人(例如：資策會、工研院等)所推行的校園無線漫遊整合計劃及 WiMAX 科學園區建置計劃也希望有合適的安全檢測工具能夠檢測無線漫遊伺服器(Roaming Server)、Base Stataion (BS)及核心網路的安全性。高科技廠商(例如：友訊集團、明泰科技、中華電信與宏碁等)而言，希望能夠有完整的檢測工具可以檢測他們開發的無線設備或者無線設備內的應用軟體上是否存在安全漏洞。然而目前市面上並沒有完整及合適的安全檢測工具可以提供檢測服務給上述單位。為了滿足政府機關、財團法人與高科技廠商對於無線網路安全檢測服務的迫切需求，本計畫建置一個異質無線網路安全檢測平台，開發相關的安全檢測工具與系統，提供(1)異質無線多網安全檢測與(2)行動裝置之系統與軟體安全檢測，前者包含無線區域網路 WiFi 滲透檢測、3.5G 和 WiMAX 核心網路滲透檢測，後者包含行動裝置軟體弱點檢測、惡意軟體 malware 偵測、惡意網頁滲透惡意程式檢測以及無線系統滲透檢測，以滿足國內目前對於無線網路安全檢測上的需求。

● 異質無線多網安全議題

無線網路提供使用者無拘無束的網路使用環境，但無線網路封包在開放的空間中以電波傳送，一般人可以輕易地就抓取封包，因此帶來更多安全的威脅。為了讓使用者即使在移動的環境下仍享有高速且穩定的網路服務，於是各種不同的無線網路標準因應而生，如：Wi-Fi、3G、3.5G 和 WiMAX。但由於無線網路通訊協定的設計不良、加密方式等缺陷，導致許多無線網路安全上的問題亟待解決。以下我們列舉一些在 Wi-Fi、3.5G 和 WiMAX 上常見的安全議題。

■ 無線區域網路(Wi-Fi)的安全議題

IEEE 於 1999 年提出了 802.11 的 WLAN 通訊標準，以此標準為基礎的 Wi-Fi 為現今普及的無線網路應用。目前以 2.4GHz 頻段之 11b 及 11g 為主流，最高傳速達到 54Mbps(11g)。而 802.11n 的傳輸速率為 300Mbps。在資料傳輸過程，可使用 WEP 或 WPA 加密認證機制來加密傳遞的資料。以下列舉常見發生於無線區域網路安全上的問題：

- 通訊協定的設計不良：通訊協定的設計不良造成阻斷服務攻擊 (DoS Attack)。阻斷服務攻擊最為常見也最難防範的攻擊之一。現今的駭客不需要有高超的技術或

者知識才可進行攻擊，因為許多 DoS 攻擊程式都可相當容易取得，使得網路攻擊事件更為泛濫。

- 加密方式存在漏洞：雖然 WEP 或 WPA 這兩種加密認證機制號稱可提供資料機密性 (Data Confidentiality)，但已有研究指出 WEP 金鑰可以在很短的時間之內被破解，而 WPA 金鑰亦有可能被攻擊者使用字典攻擊的方式破解，這使得資料的安全性受到嚴重的威脅。
- 隱私及秘密洩漏的風險：由於資料在無線環境中是經由無線傳輸，在傳送範圍內的所有人皆可收到網路封包，因此更容易遭到他人的竊聽，所以如何保持資料的安全不被惡意人士所竊聽及盜取是無線網路使的一大問題。
- 無線網路硬體上的缺陷：利用 802.11 直接序列展頻 (direct-sequence spread spectrum, DSSS)協定的漏洞，雖然不能截擊或修改封包資料，但是卻可以對資料封包傳送的可靠度(reliability)造成威脅，使得在範圍內的使用者無法正常使用線上資源或是溝通訊息。

■ 無線都會區域網路(WiMAX)的安全議題

WiMAX 系統架構如圖 1-1 所示，包括 SS/MS、ASN、CSN 及 ASP，其定義與功能如下所述：

- SS/MS：固定式用戶端 (Subscriber Station, SS)與行動式用戶端 (Mobile Station, MS)。
- BS：WiMAX 基地台 (Base station, BS)，它使用方向性天性來增加其覆蓋範圍與無線頻寬，強化傳輸效果，使得 BS 覆蓋範圍下的 SS 可透過 BS 存取網路服務。
- 存取服務網路 (Access Service Network, ASN)：提供 WiMAX 用戶無線存取服務，並傳送認證、授權及交易計價訊息 (Authentication, Authorization, and Session Accounting)訊息給 WiMAX 用戶端所屬的家庭網路服務供應商(Home Network Service Provider, H-NSP)，以維持連線狀態。
- 連線服務網路 (Connectivity Service Network, CSN)：提供 WiMAX 用戶 IP 連線服務，包括 MS 的 IP 位址及連線參數設定，亦可作為 AAA 代理伺服器來管理及控制用戶端存取能力，支援用戶端漫遊於不同 CSN 所需的服務。
- 應用服務供應商 (Application Service Provider, ASP)：提供各種應用或服務的商業組織，通常為 V-NSP 或是 H-NSP。

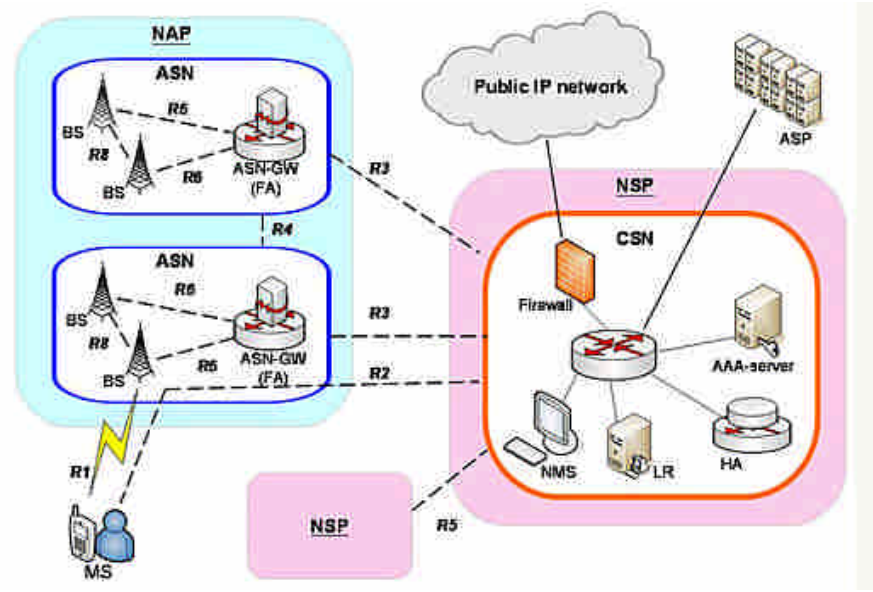


圖 1-1、WiMAX 網路架構圖

在 WiMAX 的環境下，攻擊者可藉由監聽得到使用者傳送的私人資訊，並發送惡意的錯誤訊息。因此，如何判斷並提醒使用者所處的 WiMAX 環境私密性以及使用者進行的網路使用行為是否被監聽亦是一個很重要的議題。被監聽的資訊在經過攻擊者有系統或長時間的蒐集後，會對使用者造成一定程度的威脅。除了沒加密過的資訊可以直接被攻擊者得知外，攻擊者還可針對各種協定封包的分析來進行更進一步的破壞動作。而即使是加密後的資料，攻擊者也能蒐集一定的數量後，破解出使用者所使用的加密成鑰或是解得一些資訊等。所以監聽對於 WiMAX 使用者而言是非常危險的攻擊手法。另外，由於核心網路的重要性使其經常成為被攻擊的對象。若核心網路沒有受到任何保護，很容易被攻擊者攻擊進而無法服務合法使用者。

針對 WiMAX 核心網路，我們分別探討 AAA 伺服器 (Authentication-Authorization-Accounting server) 及 DHCP 伺服器所遭受到的安全風險：

- AAA 伺服器: AAA 伺服器主要提供認證 SS 的服務，因為 WiMAX 中只有 BS 透過 AAA 伺服器對 SS 進行身份認證程序，卻缺乏 SS 對 BS 的身份認證。因此，在沒有對 BS 進行驗證的同時，惡意的攻擊者可以假造 BS 進行 Man-in-the-Middle 攻擊，即 SS 以為自己在跟預期中的 BS 進行溝通，但其實中間是透過一個惡意的攻擊者在轉送雙方的資訊。除了認證上的安全性之外，由於 AAA 伺服器還需負責計算 SS 的費率，配置上需注意 AAA 伺服器所能承載 SS 的數量，避免因攻擊者偽造流量或偽造大量 SS 來降低 AAA 伺服器效能，甚至癱瘓 AAA 伺服器，使其無法再進行認證與計費的服務。
- DHCP 伺服器: 在 WiMAX 的設計中，每一個 SS 需要有一個網路卡位址。SS 在與 BS 取得連繫之後，可要求 DHCP 伺服器將網路卡位址轉成一個 WiMAX IP。DHCP 伺服器必須提供穩定的服務品質，否則可能大量 SSs 同時註冊或偽造許多網路卡位址來換取大量的 WiMAX IP 時，會造成後端 DHCP 伺服器當機，無法

再繼續服務其他合法使用者。

■ 3.5G 無線網路的安全議題

3.5G 提供無所不在上網通訊便利性，然而 3.5G 無線網路的安全議題也逐漸受到電信業者與研究學者的重視。3.5G 無線網路的安全議題包括可能發生於服務核心網路 (core network) 的攻擊行為或用戶端手持式行動裝置 (mobile device)的滲透行為。

核心網路的攻擊行為可能造成某些網路節點無法提供路由服務，使得某些網路區段中斷網路連線服務；也可能使某些核心服務伺服器因惡意的服務要求，造成系統過載，而無法對正常的要求提供適當的服務。

舉例來說，如果核心網路裝置或行動網路裝置中在上線運轉前，未檢測出其潛在的系統漏洞或應用軟體弱點，則可能使這些裝置容易遭受到駭客攻擊，而成為殭屍網路的一員。經分析當成為殭屍網路的網路裝置或行動裝置數量達到約萬台時，擁有控制此殭屍網路的主事者，便可協調在特定時間大量傳送不當的服務要求 (如: call forwarding 要求)，使此網路系統因遭受 DDoS 攻擊，而造成整個核心網路的癱瘓。 這些攻擊，都很可能導致業者的重大損失。在行動網路裝置方面，常見的攻擊包括：

- 網頁內嵌木馬攻擊：此為目前最流行的攻擊方式之一。攻擊者將惡意程式碼植入使用者經常瀏覽的網頁中。當使用者開啟該網頁時，惡意程式碼將可以竊取隱私資料、植入後門程式，甚至獲得該系統的管理者權限。此攻擊可導致使用者資料外流，也可能讓使用者的系統崩潰或者成為攻擊者可控制的殭屍機器。
- 網際網路釣魚攻擊：此攻擊目的是誘使使用者進入偽造的網站，騙取使用者的帳號、密碼、甚至個人重要機密資料。常見的誘使手法須搭配網域名稱伺服器(Domain Name Server, DNS)攻擊、竄改 ARP 封包、偽造重要信件使受害者相信信件指示、或是使用相似的網址使受害者一時不察而誤以為偽造網址是可信任的。
- 駕駛攻擊 (War Driving)：駕駛攻擊是被動式攻擊的一種，藉由簡單的設備(如接收力強的天線)搭配軟體，沿街搜集行動網路拓樸資訊，在網頁上以地圖方式公佈特定區域的網路路由資訊。有心人或許會利用安全防護不足的網路節點發動其他攻擊。
- 冒充 (Masquerade)：攻擊者冒充他人的身份欺騙認證系統，得到進入某個網路的資格。以下簡單介紹可應用於行動網路之 IP 假造與欺騙方法。攻擊者先利用阻絕服務攻擊(DoS attack) 讓使用者的裝置無法對外連線，避免自己假冒使用者裝置的行為被發現。隨後修改自己送出的封包之標頭 (header)，把 IP 位址改成原使用者的 IP 位址，欺騙路由器 (router)或防火牆這個連線是來自可信任的網路，因而得到連線的資格。
- 竊聽 (Eavesdropping)：無線通訊經空氣傳播的特性，可以讓攻擊者輕易取得與儲存整個網路中流通的資料；即使訊息經過加密，仍需提防攻擊者從得到的部分訊息中分析取得重要資訊。攻擊者可以為了自己攻擊的需要而在網路中傳播某些訊息；攻擊者也可以透過被記錄下來的封包或明文裡的資訊，找到加密的金鑰並用來解密其他封包。
- 被動竊聽/流量分析 (Passive eavesdropping/ Traffic Analysis)：攻擊者利用自己的

設備竊聽無線網路上的封包，並針對網路通訊的流量、內容，以及行為等進行分析；透過通訊內容或流量分析，可得到目標網路的資料，如伺服器位址、通訊模式等。由於攻擊者只在電腦前觀看網路上傳送的封包，所以不會留下任何線索。利用監聽模式配合軟體如 Kismet，可竊聽並儲存通訊內容；某些竊聽軟體可以將通訊的內容以模擬終端機的形式展現，竊聽者可以看到與使用者一模一樣的終端機畫面。

- 中間人攻擊 (Man-in-the-Middle Attack)：攻擊者冒充合法的使用者與基地台建立連線，同時冒充合法的基地台提供行動裝置進行連線服務，如此，合法的基地台與使用者之間傳送的資料必須通過攻擊者的裝置。攻擊者讓自己的裝置能在使用者與基地台之間讀取或修改傳遞的訊息，卻不被通訊雙方所察覺。此攻擊必須在攻擊者具有基地台通訊裝置之前提下方可能達到。

以現行機制來說，目前對於用戶端行為之防範只能採取較消極的控管方法。例如使用干擾器阻斷 3.5G 封包傳輸，或是在用戶端電腦安裝監控軟體以進行監控。但是前法會造成行動裝置無法通話、無法發送簡訊等問題；後者則因涉及使用者隱私，使得使用者接受度低。

● 行動裝置之安全議題

以下分別討論攜帶型電腦與 Android 應用軟體的安全議題。

■ 攜帶型電腦(NoteBook、NetBook)之安全議題

攜帶型電腦如同桌上型電腦，只要裝有作業系統就無可避免的存在著系統安全的問題，攻擊者可能會利用作業系統的漏洞或是弱點來進行攻擊。為了讓系統減少受到攻擊，使用者應當正確的使用電腦，但是一般的使用者最常發生的就是缺乏安全意識的使用行為導致系統有弱點與漏洞存在，舉例如下：

- 使用者沒有定期更新軟體：根據 AOL/NCSA 的研究指出只有百分之二的人有定期更新最新版本的軟體，而有將近百分之八十的電腦中含有間諜程式或是廣告程式，平均每位使用者的電腦中有九十三個廣告和間碟軟體。如此多的惡意軟體以及只有少數的人能做到更新的動作，使得絕大多數的電腦有漏洞存在，但使用者卻不自知。
- 使用者沒有將未使用的服務關閉：絕大多數的使用者都未必知道自己系統的哪些服務端口(Port)是開啟的，因此駭客可能會利用一些未曾使用的服務端口來滲透系統進行攻擊或竊取資料，甚至系統有可能被植入惡意軟體讓電腦變成跳板進而幫助駭客去竊取別人的資料，但使用者本身卻不知道。

除了以上的安全性問題，攜帶型電腦 (NoteBook、NetBook)還會因為使用者在使用 Wi-Fi 無線上網時可能會遭遇到駭客的攻擊，而一般攜帶型電腦在使用 Wi-Fi 無線網路時會遭遇到的安全問題舉例如下：

- 加密方式存在漏洞：使用者使用 Wi-Fi 無線上網時選擇了較不安全的加密機制，

例如：WEP 加密認證機制。目前已經有研究指出美國 FBI 的一個團隊可以在三分鐘之內成功破解 WEP 加密 [1]。因此，如果使用者選擇使用 WEP 加密並且在傳輸個人資料時，例如：銀行帳號。那麼很有可能加密的資訊將會被駭客攔截並且遭到破解，使得資料的安全性受到嚴重的威脅。

— 中間人攻擊 (Man In The Middle Attack)：在無線網路的環境中，由於資料是經由無線傳輸，而駭客可能會利用假冒的無線 AP 存取點 (Access Point)扮演著中間人的角色，進行中間人攻擊，駭客將會監聽使用者或是竊取使用者的個人私密資訊。

近年來於攜帶型電腦的發展上，比起 Notebook，廠商更致力於 Netbook 的發展。由於 Netbook 的產品定位為更專注於網際網路的漫遊，所以當 Netbook 沒有安裝防毒軟體或是防惡意程式的軟體，那麼系統中電腦病毒的機率就會大大的增高。根據 2009 年 Symantec MessageLabs 網路安全報告指出，web 網頁目前仍然是惡意程式的主要來源，並且約有 34.2%為全新出現的類型 [2]。所以，Netbook 的使用者在使用 Netbook 瀏覽網際網路時更需提防惡意程式。但是，在 2008 至 2009 年期間，隨著 Netbook 快速的發展與大量的銷售 [3]，也因為市場導向機制，企業必須要降低生產的成本，因此有可能導致 Netbook 的安全性也跟著降低，舉例如下：

— 在硬體部分 Netbook 比起 Notebook 缺少了可信任平台模組 (Trusted Platform Module, TPM)，或是生物辨識技術的安全保護機制，因此若 Netbook 不小心遺失或是被竊取時，根本無法達到資料保密的防護性 [4]。

— 企業為降低成本，價格較低的 Netbook 並沒有預先安裝試用版的防火牆或是防止惡意程式的軟體，所以使用者很有可能第一次使用 Netbook 上網瀏覽網頁或是收取電子郵件時，系統就中了電腦病毒或是被植入惡意程式 [5]。

目前主要購買 Netbook 的用戶，較多為一般用戶而非企業用戶，對於一般用戶在使用 Netbook 可能不會察覺到的安全事項舉例如下：

— 定期更新軟體，例如：更新瀏覽器。一般使用者多半使用 Netbook 上網瀏覽網頁或是收發電子郵件，使用者必須定期的更新軟體，這樣系統才不會存在漏洞以致於遭到攻擊。

— 安裝防毒軟體：Netbook 出廠時所搭載的防毒軟體均為試用版本，目的為避免可能第一次在網際網路瀏覽網頁時就中了電腦病毒或被植入惡意程式。但試用期過後，使用者往往沒有察覺試用版的防毒軟體已經過期，導致電腦沒有更新病毒碼以致於增加系統中電腦病毒的風險。

對於企業用戶，目前 Netbook 出廠時所搭載的作業系統版本多為 Windows XP Home Edition，安全性較 Windows XP Professional 不足。因此，如果企業用戶使用 Netbook 透過 Wi-Fi 無線網路瀏覽網頁時，有可能系統會中電腦病毒或是被植入惡意程式導致企業的資料被竊取。

目前無論是 Notebook 或是 Netbook 的使用者都必須要多注意自己的使用行為，

定期更新防毒軟體的病毒碼和一般軟體的補丁程式(patch)，避免在瀏覽網頁或收發電子郵件時系統中了電腦病毒或者惡意程式，並且在使用 Wi-Fi 無線上網時，要選擇具安全性的無線存取點(Access Point)，避免遭受到中間人攻擊，個人資料被監聽和竊取。

■ **Android 應用軟體之安全議題**

隨著智慧型手機的發展以及市場的需求，Google 在 2007 年推出 Android 手機作業系統，並且結合各家手機廠商，讓手機作業系統趨向開放且可自由修改。然而在智慧型手機上，因為使用者可以修改作業系統底層的元件，而使得安全問題的重要性逐漸被重視。

Android 發展的動機，除了提供一個 Open Source 的環境讓廠商可以節省平台授權費，不必被層層剝削，而開發者也能在開發應用程式上獲取更多的自由外，還有一個重要的因素：讓應用程式可以在不同平台上通用，不必花費時間及精力對應用程式作跨平台的改寫。Android 系統底層以 Open Source 界最著名 Linux Kernel 為基礎，它只提供最基本的運作功能。而最有附加價值的應用軟體部份則可自行開發，以 Java 作為編寫程式的一部分。Android 採用了軟體堆層 (Software stack)的架構，主要分為三部分：

— 語言部分

Android 使用的語言與 Java 並沒有不同。

— 虛擬機器

Android 使用的虛擬機器稱為 Dalvik，它認識的指令集並不是 Java bytecode，而叫 Dalvik executable，簡稱 dex。Android 軟體開發套件 (Software development kit, SDK)裡提供了一個工具程式叫 dx，可以把 Java bytecode 再轉換成 Dalvik 所支援的 dex，這樣 Dalvik 就知道怎麼執行它。為了適合在電話這種比較小型的平台上使用，Dalvik 做了許多最佳化的處理，例如減低記憶體的使用、偵測某些裝置是否閒置而暫時關閉以節省電力以及可以有效率的同時執行多個程式。

— 函式庫

Android 提供了大部分的 Standard Java Library (來自於 Apache Harmony Project)，並把他們轉換成 dex 的格式，如此 Dalvik 才認得。除此之外，還提供了很多獨有的函式庫，讓使用者可以直接呼叫來使用電話、GPS 等元件，或者是一些視覺的元件來取得跟其他 Android 程式相同的外觀。Android 虛擬機器與函式庫合稱 Android Runtime。

在 Android 平台上，目前出現的攻擊可以簡單分為兩大類：

— 第一類是 JNI (Java Native Interface)所呼叫函式庫的漏洞，因為 Android 本身的函式庫是用 Java 撰寫而成，Java 程式本身會經過 JVM (Java Virtual Machine)的檢查，因此可以減少大量的漏洞及問題。但是外部呼叫的函式庫 (例如：瀏覽器引擎、藍芽函式庫等等)絕大部分是用 C/C++編寫的，執行時可能含有 buffer overflow 的問題而導致整支手機被入侵。

— 第二類為應用程式標準上的問題，例如藍芽、瀏覽器等等。這些攻擊往往是利用

通訊標準制定上的漏洞而形成的攻擊，例如在藍芽資料傳輸中，可能遭受 DoS、Password crack 的攻擊。

二、 相關研究

以下將針對各研究範疇，介紹目前國內外相關研究。

● 異質無線多網檢測方面

以下我們將無線網路分成兩大塊討論，一為3G 核心網路之滲透系統，另一為 WiMAX 核心網路之滲透系統。

■ 3G 核心網路之滲透系統

滲透測試 (Penetration Test)為一種檢測網路資訊安全的手段 [24]，不同於傳統的資訊安全測試方法，滲透測試的概念是用真實駭客的思維角度與手法，來對目標網路、主機或是系統進行攻擊測試。此外，滲透測試之成效取決於滲透路徑之選取，這也是與傳統弱點檢測極為不同之處 [25]。滲透測試的過程就如同真實的攻擊事件，可以呈現受測者實際脆弱的環節，以及得知受測者的安全強度與攻擊者可能的攻擊路徑與方法。受測者再藉由測試的結果報告，進一步的補強安全性或是更新系統以達到最佳的防護效果。

滲透測試由一連串的測試步驟、搭配使用不同的軟體和分析方法組合而成。其可粗分成三大步驟，分別為探測、弱點掃描與攻擊和結果分析 [26]。探測是滲透測試的第一步，合適的受測目標能幫助系統管理者更佳且有效率地掌握網路系統的弱點；反之，若是受測目標選取不當，即便測試結果也能成為系統管理者的參考依據，但將無法反映出網路的核心問題。3.5G 行動網路滲透測試與有線網路滲透測試或是無線網路滲透測試相同，測試的第一步皆是先探索與推測網路拓樸，進而分析探測結果找出關鍵的受測目標。而不同網路層級的滲透測試皆有其意義，多層級的測試可以幫助系統管理者或是網路服務提供者 (Internet Service Provider，簡稱 ISP) 瞭解更全面同時也更精確的系統弱點。

3.5G 行動網路滲透測試可以分為許多不同層級的測試，如路由 (router) 層級測試、ISP 層級測試或是自治系統 (autonomous system)層級測試等等，因此全面且完整的 3.5G 行動網路滲透測試應涵括多個層級的測試目標。

不論是針對哪一個網路層級進行探測，網路探測的相關研究多半是建立在 traceroute 這個探測工具得到的單一路徑之上，進而從有限的路徑擴展至全面性的網際網路或是網路服務提供者的拓樸。traceroute 的工作原理如下：首先，traceroute 送出 ICMP (Internet Control Message Protocol 網際網路訊息控制協定, ICMP)封包至受測的目標主機，每個 ICMP 封包指定不同的存活時間 (Time To Live, TTL)，TTL 長短會決定封包是否可以到達受測目標。若是 TTL 太短以致封包無法到達，亦即，TTL 的長度只允許該封包到達發送主機和受測目標主機之間的某一路由器，則該路由器會丟棄

封包並且回傳 ICMP 逾時封包給發送主機。發送主機接收到逾時封包後，便會累加 TTL 數值並且重新傳送 ICMP 封包，傳送封包的動作會持續到發送主機接收到由目標主機回傳的 ICMP 回應封包為止。藉由這樣發送 ICMP 封包的探測方式，發送主機便能收集得到其與目標主機之間的路徑。

Mercator [20] 是一個探測網際網路路由層級拓樸的程式，Mercator 採用一種名為 informed random address probing 的啟發示 (heuristic) 技術進行探測，它可以從任一位置上的單一主機進行發送探測封包，且從有限的跳躍點 (hop) 的探索來推演得到全面的網路拓樸。 此外，基於此啟發示技術，Mercator 能以之前的探測紀錄篩選出可行的受測 IP，它不需要如邊界閘道協定 (Boarder Gateway Protocol，簡稱 BGP)路由表格等外部資訊亦能大幅減少不必要的探測次數。Mercator 亦支援路由器接口別名解析 (alias resolution)，它可以判別哪些 IP 位址是屬於同一個路由器，以增加網路拓樸探測的準確度與保真度。Magoni 等人 [21] 在 2005 年提出了名為 nec 的網路拓樸繪製軟體，該軟體是以 traceroute 工具為基礎以取得基本的路由路徑。與 Mercator 不同，nec 是由大量的位於不同位置的主機對事先決定的目標主機進行路徑探索，而非隨機選取目標主機。另一個與 Mercator 的差異是，nec 除了可以探索路由層級的網路拓樸外，nec 也支援自治系統 (autonomous system)層級的拓樸探測。

除了路由層級與自治系統層級，許多專家學者亦提出了針對網路骨幹 (backbone) 拓樸的探測方法。Barford 等人 [22] 認為進行網路骨幹拓樸探索時，與被動的受測目標主機的數量相比，主動探索的來源主機的數量相對地小，因此骨幹拓樸的探索方法必須是基於此特性的設計。Barford 等人亦討論了增加來源主機和目標主機的數量對於探測準確度的影響。從他們的實驗結果可以得知，對探索網路拓樸而言，增加目標主機的數量的影響性和重要性遠勝於增加來源主機的數目。

Rocketfuel [23] 是由 Spring 等人設計開發的 ISP 網路拓樸探測引擎。與先前介紹的其他方法或工具不同的是，Rocketfuel 專注於單一的 ISP 骨幹拓樸的探索，只對進出該 ISP 的路徑感興趣。Rocketfuel 利用邊界閘道協定 (Boarder Gateway Protocol，簡稱 BGP)路由表格過濾與 ISP 骨幹無關或是沒有進出該 ISP 的路徑。此外，若是有兩條不同的路徑但都經由同一個路由位址進入或是離開 ISP 骨幹，則 Rocketfuel 只會保留其中一條；這是因為 Rocketfuel 的重點在於探測單一 ISP 的骨幹網路，Spring 等人認為當數條路徑皆由同一個路由器進入/離開骨幹時，可以推測這些路徑在骨幹內部的片段路徑是相同的，因此只需保留一條以降低分析的運算量、提昇分析效能。Spring 等人以 Rocketfuel 探測得到 10 家 ISP 的骨幹網路拓樸，並且徵詢 ISP 的協助以確認探測結果的保真度。其中有三家 ISP 回應了 Spring 等人的問卷，他們同意 Rocketfuel 得到的探測結果具有很高的保真度。

■ **WiMAX 核心網路之滲透系統**

WiMAX 之保護資訊和網路安全的途徑有很多種，滲透測試是其中最有效的一種途徑，可以在攻擊者分析、滲透、攻擊某個系統之前，先行瞭解該系統的安全問題與弱點。也因此，網路滲透分析工具近幾年來開始逐漸受到各界重視。這些滲透測試與

分析主要是透過信任的第三方(Trusted Third Party, TTP)所進行的一種評估網路系統安全等級的活動。利用駭客攻擊工具，TTP 可以對企業網路進行各種攻擊，並試著找出網路系統中所存在的安全漏洞,從而評估該網路存在的安全風險。常見幾種 Penetration Testing Tools 可以分為兩大類：偵測工具與探勘工具。前者包括 Nmap、Nessus 及修改封包與破解密碼的工具。Nmap 是一種常見的連接埠掃描工具，常用於系統滲透的偵查階段。由於不同作業系統對不同的網路封包會有不同的反應，所以 Nmap 通常會利用資料庫內所記錄的資料及封包的回應狀況猜測作業系統的種類，其結果通常可以提供攻擊者有用的資訊片段，使攻擊者有機可趁。Nessus 是一種常見的弱點掃描工具，是許多網路安全專家常用的系統。常見於掃描、測試及辨識系統弱點。大部分情況下，Nessus 會依據系統的反應作判斷，而不需要真的勘測系統內部。許多時候，使用者會讓 Nessus 搭配 nmap 掃描待測系統。Hping 是一種竄改封包的工具，可用來產生或傳送各種精心設計的 TCP/IP 封包，用以測試或偵測網路安全保護機制 (防火牆或 IDS/IPS 等)。John the Ripper、Cain&Able 都是破解密碼的工具，可用於偵測、破解強度較差的密碼。

在探勘工具方面則有 SAINTexploit 、Metasploit、SecurityForest、CORE IMPACT 等。SAINTexploit 是 SAINT 公司所開發的一套用於找出網路系統中潛在安全漏洞的工具。可以用於找出攻擊者可能入侵網路的漏洞、計算網路風險值，並能讓 IT 管理人員更有效率管理網路資源、保護其資訊資產。Metasploit 是一套已經發展多年的滲透工具,提供攻擊函式庫及攻擊封包,可以滲透系統,取得系統權力,如進入 command prompt 畫面。專家學者發現 Metasploit 是解決擾人的資訊安全問題不可或缺的工具之一。SecurityForest 是一個開放原始碼工具，這個工具收集許多 exploit 程式碼，並建置出 ExploitTree，因此測試人員可以透過網頁瀏覽器啟動 exploit 程式碼，並進行滲透。CORE IMPACT 是一套商用滲透測試工具,主要可用以找出各程式中的可能弱點。CORE IMPACT 可以在被破解的電腦中植入一個代理程式，並由該電腦啟動其他的網路攻擊。在實際的滲透測試分析中，這是一個很有用的功能，可以讓測試人員破解一台電腦，並由被破解的電腦啟動自動掃描，尋找系統中的下一個可能受害主機。

- **行動裝置檢測方面**

以下我們將行動裝置相關研究分成兩大塊討論，一為行動裝置系統安全滲透檢測，另一為行動裝置惡意軟體檢測。

- **行動裝置系統安全滲透檢測**

Android 行動裝置系統以 linux 為底層，在其上面用 dalvik 虛擬機器運行特有的 dex 檔，但是實際上和原本的 java 並無差別，檢測 android 應用程式有以下幾種方法：

- 將 dex 檔轉回 java bytecode 並支援 android 特有函式庫。
- 在 JVM 上模擬 dalvik 虛擬機器並支援 android 特有函式庫。
  (以上兩種方法可直接套用現有檢測工具)
- 直接開發一個針對 android 行動裝置的檢測工具。

目前在軟體安全檢測的方法大致上分為兩大類：

1. 靜態分析

靜態測試是對軟體程式碼作靜態程式碼分析(static code analysis)來檢驗程式碼中是否有錯誤或可能被攻擊的弱點。然而靜態測試有誤判率 (false positive, or false alarm)過高的問題。因為靜態分析程式碼沒有實際執行，因此無法保證找到的程式錯誤是真正的錯誤。靜態分析後的結果需要靠測試人員進一步檢查才能確定是誤判或是真實的錯誤，因此需要花額外的人力來判斷，相關研究工具包括 Findbugs、PMD 等。

2. 動態測試

動態測試利用實際執行受測程式來找出錯誤或弱點。動態測試並不會再有誤判的問題，但是如何提供這程式所需要的輸入 (例如函式的參數或標準輸入等)是另一個關鍵的問題。最簡單的方式是亂數產生輸入給程式的測試資料，這樣的工具稱作 fuzzer。因為是亂數產生程式的輸入，fuzzer 很容易產生一堆輸入資料給受測程式，有一定的機率讓程式行為異常。Fuzzer 把受測程式視為黑盒子，只了解程式的輸入輸出，對其內部結構完全沒有分析，所以無法測試的範圍無法涵蓋所有程式的路徑，沒辦法測到需要特殊條件的路徑或結構，相關研究有 jCute、java path finder 等等。

近幾年來，一種結合靜態測試與動態測試的方法被提出，叫做 concolic testing，一方面利用 symbolic execution 對所有被程式輸入影響到的變數作符號分析(symbolic analysis)，以解決產生輸入的問題；另一方面用 concrete execution 確定目前輸入所引導的程式執行路徑，來避免誤判率高的缺點。Conclic testing 將靜態與動態分析結合以達到優缺點互補的效果，近幾年有許多相關的研究顯示這個方法比起傳統的測試方法有很大的改善，而且這個方法可被應用在系統核心的測試以及多線程(multi-thread)的軟體測試上。

■ **行動裝置惡意軟體檢測**

Google 於 2007 年推出 Android 手機作業系統，並讓手機作業系統趨向開放且可自由修改，因而使用者可以任意修改作業系統的底層元件，促使得安全問題的重要性逐漸被重視。隨著智慧型手機的發展以及市場的需求，Google 在 2010 年 1 月 15 日正式推出搭載 Android 系統的 Nexus One 智慧型手機。然而防毒軟體公司卡巴斯基美國研究室 (Kaspersky Lab Americas) 的資深惡意軟體研究員 Roel Schouwenberg 預言，在 2010 年 Android 手機將會遭受大量的網路攻擊，而從實際面來看，Android 手機也已在 2008 年即陸續發生安全問題，目前出現的攻擊可以簡單分為兩大類：

第一類是 JNI (Java Native Interface)所呼叫函式庫的漏洞 [27][28]，因 Android 函式庫是用 Java 撰寫而成，經由 JVM (Java Virtual Machine)的檢查，可減少大量的漏洞及問題。但是外部呼叫的函式庫 (例如：瀏覽器引擎、藍芽函式庫等等)絕大部分是採 C/C++編寫而成，執行時可能含有 buffer overflow 的問題而導致整支手機被入侵。

第二類為應用程式標準上的問題 [28]，例如藍芽、瀏覽器等等。這些攻擊往往是利用通訊標準制定上的漏洞而形成的攻擊，例如在藍芽資料傳輸中，可能遭受 DoS、

Password crack 的攻擊。

然而一般電腦上的防毒軟體對於智慧型手機來說是沉重的負荷，因為處理器的速度、記憶體的大小，以及硬碟所存放空間的限制，讓智慧型手機不能使用一般的防毒軟體。而且過去的防毒軟體乃透過尋找病毒特徵碼的方式進行偵測，但現今使用多型技術的惡意程式越來越常見，舊式的比對方法已失去效用 [29][30]。而目前若要在 Android 上建置惡意程式動態檢測工具，除了須考慮處理器的問題之外，因為在智慧型手機上大部分是使用 ARM 的處理器，而且 ARM 的處理器在每個版本的指令集不盡相同，例如 Android 支援 ARMv4 跟 ARMv5te，但是這兩個版本的指令集不完全相同，所以檢測工具也必須要有兩個版本方能正確運作。更加嚴重的問題在於 ARM CPU 本身執行的速度遠遠低於 X86 CPU，所以建置惡意程式動態檢測工具的效率將會非常差。而 Android 本身是實作 Software Stack，其實架構上跟 x86 極為相似，因此若我們能把 Android porting 到 X86 的虛擬機器上，就能啟動動態偵測的功能。事實上，隨著 Android 的發展，Google 也在 Android 的原始碼內部增加了 X86 instruction sets 的部份，但因為 ARM 的部份是 Android 開發的重心，所以在 X86 部分的缺失則沒有受到重視，例如：硬體的驅動程式、周邊配件的支援，以及指令集的完整度。若可以把以上所述的部分補齊，就可以順利的在 QEMU 上動態偵測 Android 的資訊流向。

目前在 DIFT 方面，最大的特點在於 Instruction Level 的污染源偵測，因為在最底層往上看，所以可以跨處理程序，單純針對相關的指令進行行為分析。若是以一個 Process 為出發點，則有可能遺漏了其他程序，因此 DIFT 方能找出最完整的資訊流動。

三、 貢獻

本計畫建置一套異質無線網路安全檢測平台，對於異質多無線網路與各類行動裝置，開發相關的安全檢測工具與系統，以提供使用者安全的無線網路使用環境。異質無線網路安全檢測平台分為兩部分：(1) 異質無線多網安全檢測與 (2) 行動裝置之系統與軟體安全檢測。異質無線多網安全檢測包含無線區域網路 WiFi 滲透檢測、3.5G 和 WiMAX 核心網路滲透檢測；行動裝置之系統與軟體安全檢測包含行動裝置軟體弱點檢測、惡意軟體 malware 偵測、惡意網頁滲透惡意程式檢測、及無線系統滲透檢測。本計畫提出完善的無線網路安全檢測方案，以滿足國內目前對於無線網路安全檢測上的需求。本計畫在 2009 年已建置並開發 13 種安全檢測工具，對使用者系統及其行動裝置以及所在之無線網路環境，進行完整的安全滲透檢測及惡意攻擊偵測，提供安全性上的評估標準，以及主動式的系統防護，以滿足國內目前對於無線網路安全檢測上的需求，並提供使用者安全可靠的行動環境。

四、 異質無線多網安全平台介紹

異質無線多網安全檢測平台 (WiSec@TWISC)可支援多種異質網路 (例如 WiMAX、3.5G、WiFi、有線網路)、支援多種行動設備 (例如：筆記型電腦、迷你筆電、智慧型手機)以及支援多種作業系統 (例如：Windows XP、Linux、Android)，如圖 4-1 所示。

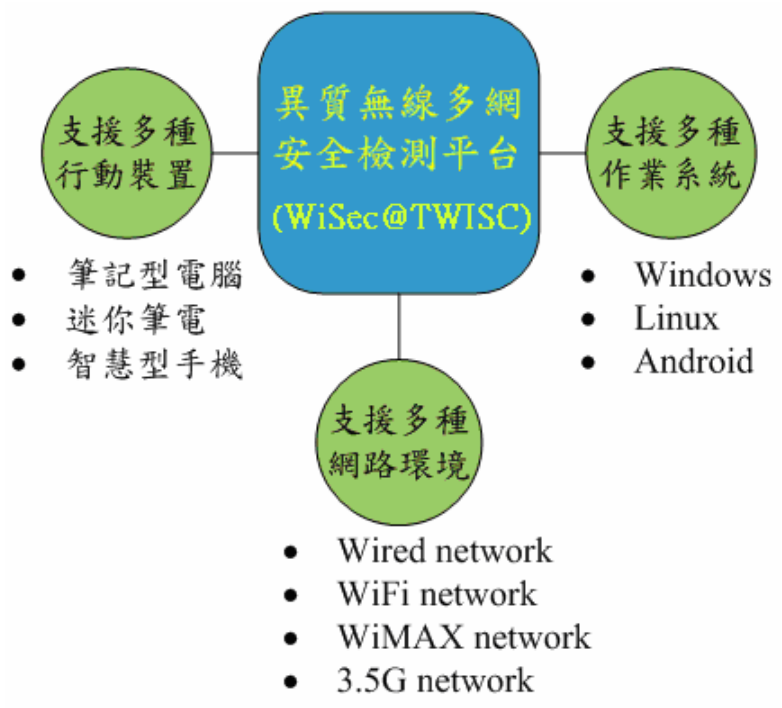


圖 4-1、WiSec@TWISC 的支援性

依據檢測目標和特性，異質無線多網安全檢測平台(WiSec@TWISC)可分為兩大部份 (如圖 4-2 所示)。第一部份為異質無線多網核心網路滲透檢測，其檢測對象包括 3G、WiMAX、Wi-Fi 的核心網路；第二部份為行動設備滲透檢測，其檢測對象包括筆記型電腦、Netbook、智慧型手機等，而這些行動設備可為 Windows、Linux 或者 Android 平台。

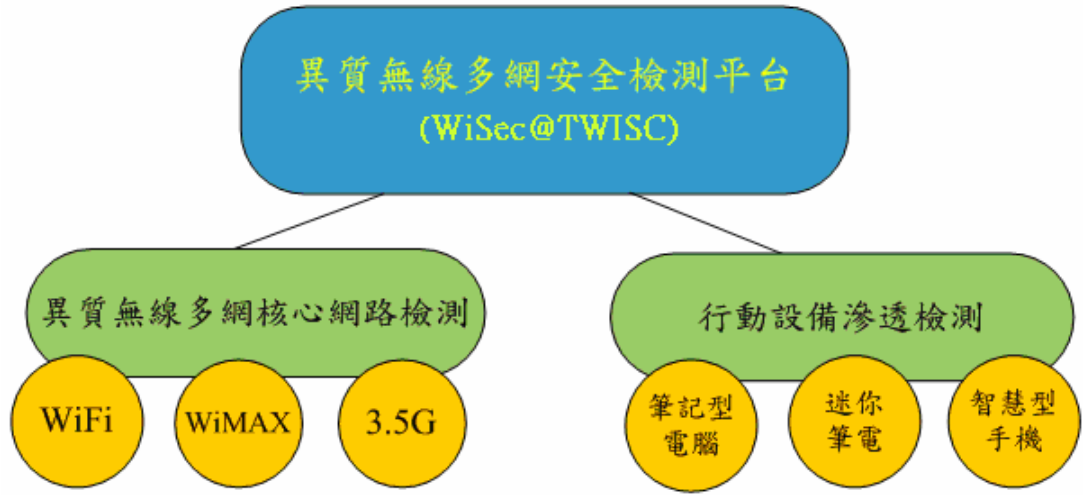


圖 4-2、異質無線多網安全檢測平台之架構圖

針對異質多核心網路之滲透檢測以及行動設備之滲透檢測，我們已開發與建置以下 13 種子系統與工具，並已在實際運轉與測試中。以下針對異質無線多網核心網路滲透檢測與行動設備滲透檢測兩大部份分別介紹我們所建置與開發的工具。

- **異質無線多網與核心網路檢測**

在異質無線多網核心網路檢測方面，我們建置與開發以下五種子系統與工具來滲透與檢測 WiFi、WiMAX 以及 3.5G 核心網路的安全性，藉此找出不當的核心網路設定管理方式，並且檢測不良的通訊協定與加密設計。這 5 五工具分別為：3.5G 滲透檢測系統(MoPT)、WiMAX BS Denial of service Testing System (WBDT)、Wireless Penetration Testing System (WiPT)、Wireless Security Monitor (WiMon)以及入侵偵測系統強度評估系統(Simple IDS Informer，簡稱 SIDSI) 。以下我們將分別介紹各個子系統與工具。

- **MoPT: 3.5G 滲透檢測系統**

在滲透檢測工具方面，在 98 年度我們已經完成檢測系統 MoPT 之系統架構設計，如圖 4-3 所示。

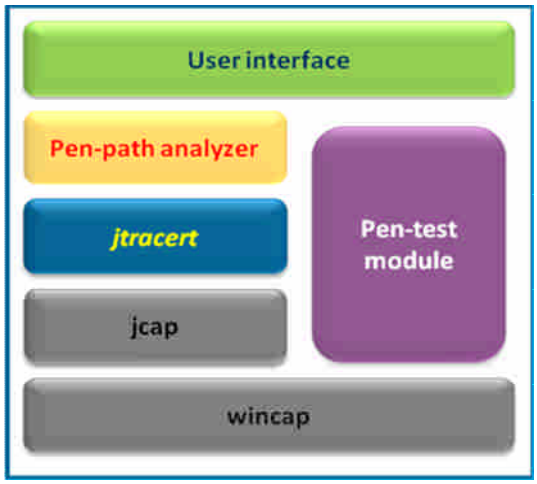


圖 4-3、MoPT 系統架構圖

本系統除提供圖形化使用者介面外，還設計了三個主要的模組：路徑追蹤工具(jtracert)、路徑分析器 (pen-path analyzer) 與滲透測試模組 (pen-test module)。其中，jtracert 為本系統中的一個工具，以 jcap、wincap 等函式庫為基礎，此工具可用於探索指定區域的網路拓樸。Pen-path analyzer 可進一步地分析 jtracert 所提供之資訊，並找出系統中的弱點節點，以 pen-test 模組進行滲透測試分析。

我們目前已經完成 jtracert 之雛型實作，並針對台灣數家 ISP 業者的 3.5G 網路進行拓樸探索。 此拓樸探索為進行滲透測試之第一步，透過拓樸探索實驗可瞭解各家 ISP 在某些定點的網路拓樸結構。依據這些測得的網路拓樸資訊，搭配拓樸分析演算法，則可推測出可能遭遇到攻擊的網路節點。這些資訊將有助於對 3.5G 網路設備進行進階檢測，進而找出潛在的錯誤設定與安全漏洞，增強 3.5G 網路抵擋攻擊之強健度。滲透檢測行動網路可以分為下列幾個步驟，如圖 4-4 所示：

- 追蹤路由路徑 (trace routes)：指定不同的目的節點 (destination)，利用 icmp echo/reply 或 TCP 等協定，找出來源節點 (source)與目的節點之間的路由路徑，以下稱為 traces。
- 收集路由路徑 (collect traces)：先將所收集到的路由路徑檔案，匯入本系統中，以來源節點為起始點，將各路徑串接成一個網路區域拓樸圖 (graph)。
- 分析節點類型 (analyze type of nodes)：分析上述網路區域拓樸圖中各節點的內分

支度 (in-degree)與外分支度 (out-degree)，並決定各點的類型。本系統中，定義數種節點類型，包括來源節點 (s-node)、目的節點 (d-node)、中間節點 (i-node)、共享節點 (x-node)、受害節點 (v-node)等。一個節點可以同時具有多種類型，如某個節點可能同時是目的節點 (d-node)與共享節點 (x-node)。

- 選擇受害節點 (choose v-nodes)：依據節點資訊與其類型分類結果，可以找出網路區域拓樸圖中可能的數個受害節點。這些節點都可能成為攻擊者攻擊的目標。
- 攻擊受害節點 (attack v-nodes)：找出受害節點之後，可以透過現有的滲透分析工具 (nmap、nessus、metasploit 等)對這些受害節點進行滲透分析。由滲透分析結果，可進一步地評估出此區域網路的安全等級。這些評估結果將可提供相關 ISP 業者或系統管理員參考，以茲作為加強其網路安全等級之依據。

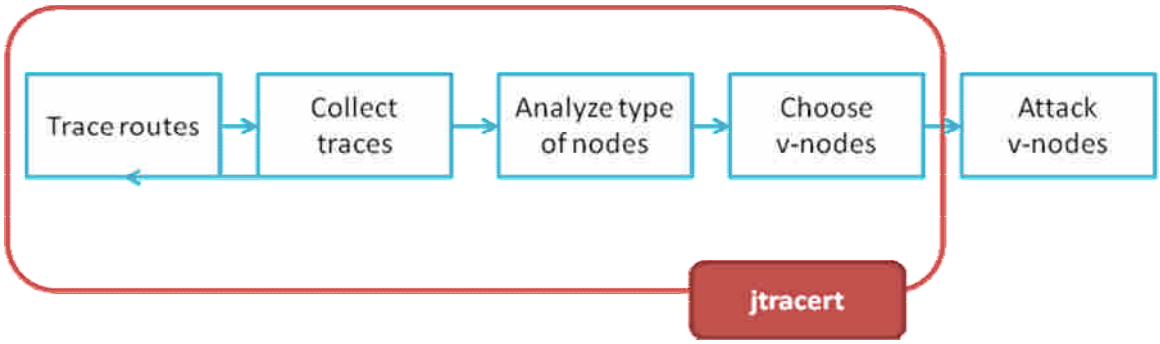


圖 4-4、行動裝置滲透檢測之步驟

我們已經將圖 4-4 所示之追蹤路由路徑、收集路由路徑、分析節點類型、選擇受害節點等模組，整合為一網路拓樸探索工具(即 jtracert)，未來並將整合於 MoPT 滲透檢測系統中。jtracert，一套以 Java 程式語言開發的網路拓樸探索工具，可單機運作，亦可整併於 MoPT 中，成為其滲透檢測的第一步。此工具主要以路由角度來探索網路中的各節點。透過不同路徑的整併，可歸納得到部分的行動網路拓樸。圖 4-5 為 jtracert 的操作介紹。啟動 jtracert 之後，使用者可以利用右鍵選單新增或移除目的節點，亦可勾選所想要的目的節點(d-nodes)，並收集其與來源選單之間的路由路徑。

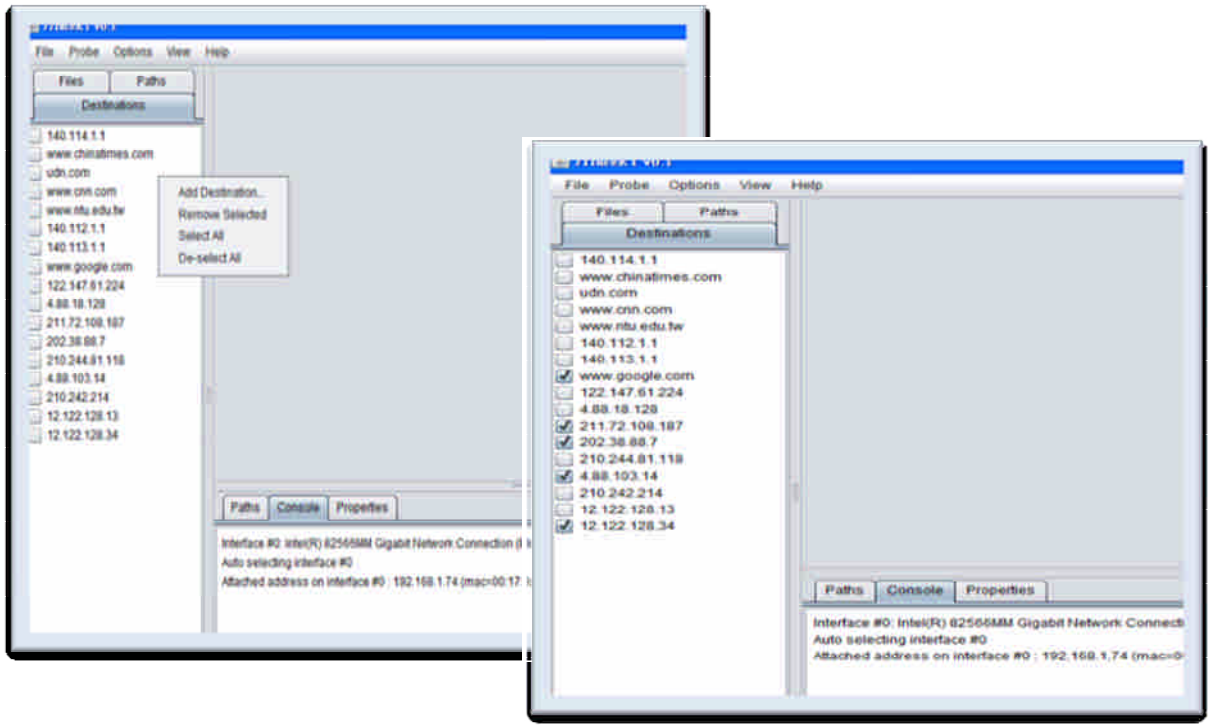


圖 4-5、jtracert 之操作界面

這些路由路徑將以檔案形式存在，使用者可以於需要時，將其載入系統中，或獨立顯示，或與其他路徑整併，以得到較完整的網路區域拓樸圖，如圖 4-6 所示。

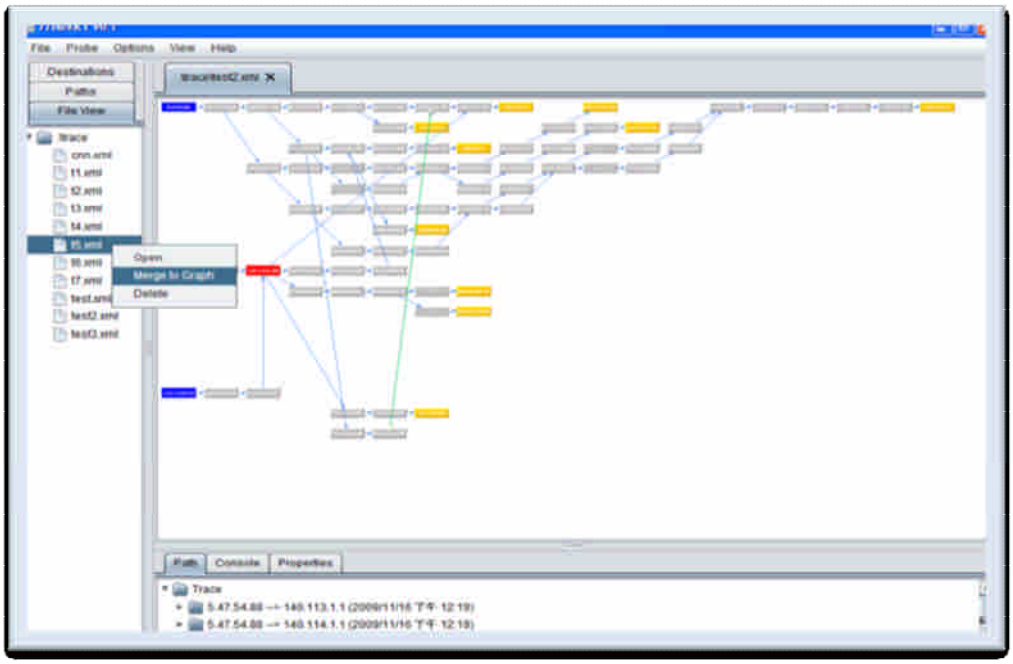


圖 4-6、jtracert 之拓樸探測與分析結果

圖 4-6 中所列之藍色節點為 s-nodeS、黃色為 d-nodes，紅色為 v-nodes。所收集

的路由路徑愈多，則更能勘得該網路中可能的弱點節點。透過本工具所評估的資訊，系統管理人員可以針對這些弱點節點進行安全功能補強，或改變整個網路的路由機制，這些都將有助於提昇整體網路的安全度。

■ **WBDT：WiMAX BS Denial of service (Dos) Testing System**

在WiMAX網路中，若是Base Station (BS) 接受Subscriber Station (SS) 傳輸大量流量，使得SS佔用大量頻寬而造成BS無法再供給頻寬給新進入WiMAX網路的SS時，表示此WiMAX網路面臨潛在的DoS attack的威脅。有鑑於此，我們開發了WBDT檢測系統，以實地傳輸大量網路流量的方式來檢測BS對頻寬分配的能力與頻寬控管是否有效，進而可分析此控管能力是否足以抵擋DoS attack的攻擊。

WBDT以實驗的方式，透過分析不同數量與不同權限的SS所獲得的頻寬來分析與瞭解BS頻寬分配的能力。WBDT以流量產生軟體Iperf與Tfgen產生異常的大量封包(abnormal traffic)，再透過以winpcap函式庫為基礎之流量監測模組 (traffic monitor) 觀察各個SS的上傳流量，藉此瞭解與分析BS頻寬分配的能力。圖4-7為WBDT之功能示意圖。

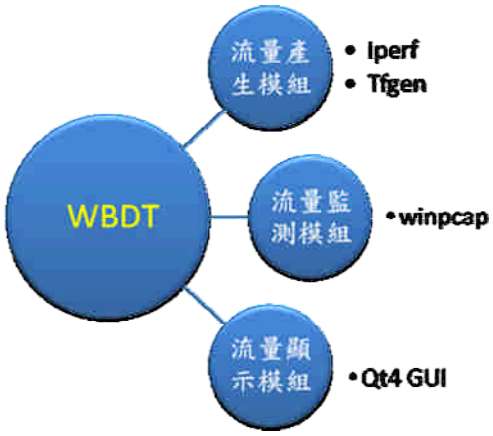


圖 4-7、WBDT 功能示意圖

我們以圖4-8這個測試環境為例，詳細介紹WBDT的檢測方法與分析結果。此測試環境中有四個不同權限的SS，它們的上傳 (uplink) 與下載 (downlink)頻寬可依權限之不同而有所差異 (見表4-1)。若帳號權限相同時，例如皆為10M 帳號，則SS_1產生abnormal traffic後陸續增加相同權限的SS數量，透過WBDT流量監測模組發現，在此WiMAX環境下無論SS的數量多少，同時產生的abnormal traffic的總量皆為5Mbps，每個SS僅能均分得到各自能使用的頻寬。若是以不同權限的帳號 (10M、4M、1M各一)同時產生abnormal traffic並觀察流量，則在此WiMAX環境下，所有SS的流量總數亦為5Mbps，但分配的比例則依帳號權限而有不同。根據實驗，我們推論5Mbps則為此WiMAX BS所能提供的總上傳流量。此外，實驗結果亦指出，SS能使用的流量與帳號權限成正比，即較高權限之使用者，其獲得頻寬比例亦較低權限的使用者多。同樣

權限的使用者，其頻寬分配的比例是為均分，如四個10M的SS，可均分5Mbps的上傳頻寬，每個SS取得的頻寬為1.25Mbps。而當10M、4M、1M的SS同時上線並產生流量時，所能取得的頻寬權重以各個SS的平均流量分析，其值約為7:2:1，如表4-2所示。

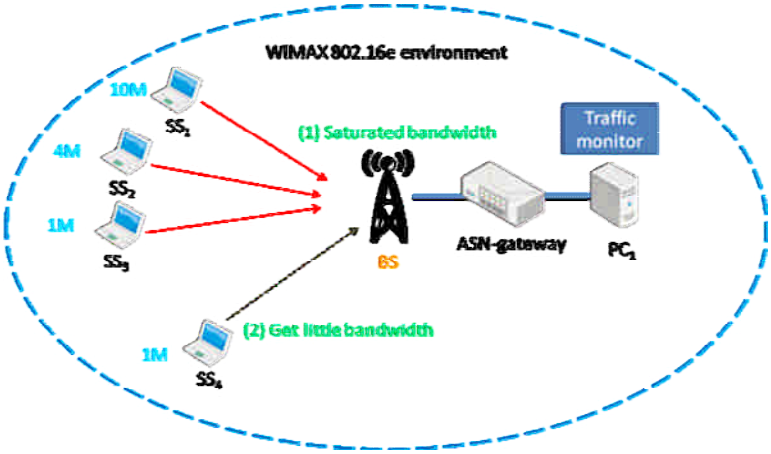


圖4-8、WiMAX測試環境

表4-1、 SS頻寬分配權限

SS account	Uplink bandwidth	Downlink bandwidth
SS_1 (10M)	10M	10M
SS_2 (4M)	4M	4M
SS_3 (1M)	1M	1M
SS_4 (1M)	1M	1M

表4-2、 帳號權限與頻寬分配比例

帳號	總頻寬分配
權限相同的帳號	所有帳號均分頻寬
10M：4M：1M	7:2:1

圖 4-9 是 WBDT 流量顯示模組呈現的 SS 流量圖。SS_1 為 10M 帳號，由於分配到較高的權重，故只由原本的總上傳流量 5Mbps 下降至 4Mbps；而 SS_2 為 4M 帳號，在進入 WiMAX 環境後只能取得約 500Kbps 的流量，上傳頻寬權重最低的 SS_4 1M 使用者，其取得的頻寬只有 100Kbps 左右。分析 WBDT 紀錄的封包流量可發現，較晚加入 WiMAX 網路的 SS 或是權限較低的 SS 無法取得依頻寬比例所該取得的頻寬，這是 BS 頻寬控管較為不足之處。

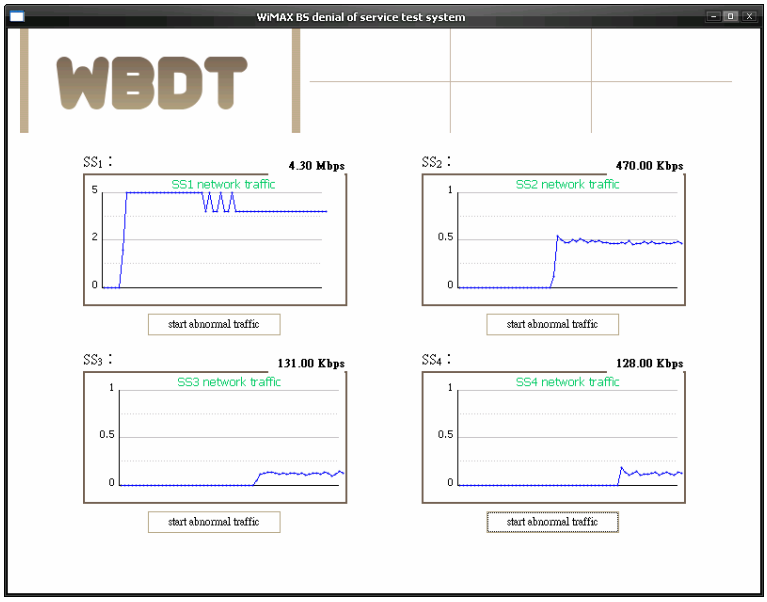


圖4-9、WBDT傳送大量流量後之流量紀錄

■ **Wireless Penetration Testing System (WiPT)**

隨著無線網路設備趨於普及，使用者自行架設無線基地台再也不是難事，尤其越來越多的商家以提供免費的無線上網吸引更多顧客上門消費，然而在使用者連接無線基地台或者是瀏覽網路的同時，就可能已經暴露在危險的無線網路環境並且遭受攻擊。

WiPT 系統著眼於攻擊者的角度進行無線網路測試。測試人員建置偽造的無線網路基地台，誘騙該區域的使用者使用該基地台。偽造的基地台提供正常的上網功能，一切使用動作幾乎和一般上網沒有什麼差別，受騙的使用者很難察覺自己連上的其實是惡意的基地台 (攻擊模擬如圖 4-10 所示)。目前出現幾種較為普遍的防衛機制如：網路提供者在架設無線基地台的同時也設定一組密碼，合法的使用者才能利用該合法無線網路基地台；網站管理者對於其網頁資訊傳輸進行加密保護，使用者在與網站連線期間都會利用 HTTPs 保護，讓一般使用者在連線期間所輸入的資訊都會以密文(cypher text)而不是明文 (plain text)進行交換。但是，上述防護機制在我們的 WiPT 系統中進行測試，都顯示該防禦方法仍有相當大的缺陷。

圖 4-10、無線釣魚測試示意圖

WiPT 滲透測試的第一步：建立惡意無線網路基地台。建置方式有二，一種是建立容易引誘使用者連線的無線基地台名稱，例如在交通大學圖書館架設惡意站台，名稱取名為 NCTU_Lib，則該名稱會有相當大的機會讓一般使用者產生興趣使用。另外一種則是建立與該區域原有無線網路基地台名稱一樣的惡意基地台，這種方式主要利用無線網路基地台掃描程式的盲點。大部分無線網路掃描程式所呈現的基地台清單，假若遇到名稱重複出現的基地台，則掃描程式會自動將訊號最強的基地台呈現在使用者的連線清單上；一旦惡意基地台的訊號強度超過正常基地台並且名稱一樣，就會出現蓋台現象，也因此，不知情的使用者很容易依循以往的習慣連上我們所建立的惡意站台 (如圖 4-11、圖 4-12 所示)。

圖 4-11、真實的基地台

圖 4-12、惡意的基地台

許多無線網路提供者架設無線網路基地台會額外增加“加密機制”提供安全防護，使用必須輸入密碼才能透過該無線網路基地台連上網路。為了完成幾可亂真的惡意基地台，並且檢測合法基地台之加密機制的安全程度，WiPT 系統亦破解了無線網路提供者所建立的防禦機制。WiPT 系統會先偽裝成外來筆記型電腦 (即沒有合法密碼的使用者)，不斷地向具有加密的合法站台發送 ARP Request 封包，根據 IEEE 的網路協定，基地台也會回覆 ARP Reply 封包，當到達一定數量時，ARP Reply 封包裡面有的特定數值產生衝突，我們依此逆推出加密規則而得知其密碼。取得合法加密基地台的密碼後，WiPT 便可對合法基地台完成進一步的測試；另一方面 WiPT 也可以利用這組密碼建置惡意無線基地台，讓使用者依照原本程序連線卻沒發現目前的連線基地台和以往有什麼不一樣。

完成第一步動作後，WiPT 系統所建立的惡意站台已經完成誘使使用者連線的工作，之後在使用者連線過程中，我們便利用中間者攻擊，竊取所有流經此基地台的網路封包。因為使用者是經由 WiPT 的惡意基地台間接上網，所以我們可以攔截所有從使用者傳出的封包以及所有輸輸給使用者的封包。完成中間者攻擊之後，WiPT 系統已經可以確實取得使用者所輸入的個人資料，儘管該這些資料是在加密協定下的傳輸，WiPT 系統也確實完成滲透工作。透過此滲透檢測的方式，可以分析以及檢驗無線網路的安全性，讓使用者瞭解其所面臨之安全威脅。

- **Wireless Security Monitor (WiMon)**

WiMon 適用於 Wi-Fi 無線區域網路，可被動掃描範圍內無線裝置上的弱點，即時監控、偵測各類型無線網路攻擊行為，包含 Authentication/Deauthentication flood attack、Beacon flood attack；並藉由監控網路封包及破解 WEP 加密達到滲透測試的目的。監控範圍內更可透過 WiMon AP 準確的定出攻擊者的位置，並產生新的防範規則保護監控中的裝置免於遭受相同方式的攻擊，是一包含主動監控、被動掃描、即時防

禦的大規模無線網路即時監控系統。

WiMon 協助網路管理者即時監控無線網路環境，包含兩大系統，其一為無線網路拓樸探索，其二為入侵預防。無線網路拓樸探索藉由 GPS 定位監控無線網路範圍內的各類型無線網路裝置，圖 4-13 即以交通大學工程三館六樓平面圖為例，此平面圖完整列出該樓層所有的無線網路裝置，且清楚的以紅色框線標示出正遭受攻擊的無線網路裝置及 WiMon AP，可協助網路管理者得知整個無線網路範圍內的無線裝置現況。

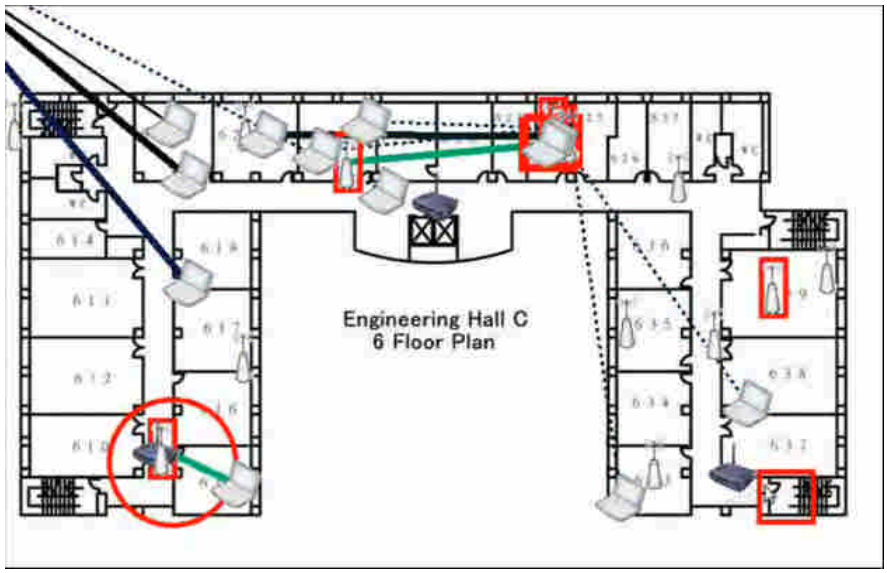


圖 4-13、交通大學工程三館六樓平面圖攻擊現況

在入侵預防方面，遭受攻擊的無線裝置將以紅色框線標示，而不同框線的形狀代表著不同種類的攻擊，包含 Authentication/Deauthentication flood attack、Beacon flood attack，如圖 4-14 所示，WiMon AP 成功的偵測到無線網路範圍內的有 Authentication/Deauthentication flood attack 的發生。接著 WiMon AP 會通知安全中心產生新的防範規則，將相對應的策略發送給裝有 WiMon 的無線裝置，若攻擊者再以相同方式攻擊將被有效被阻擋。

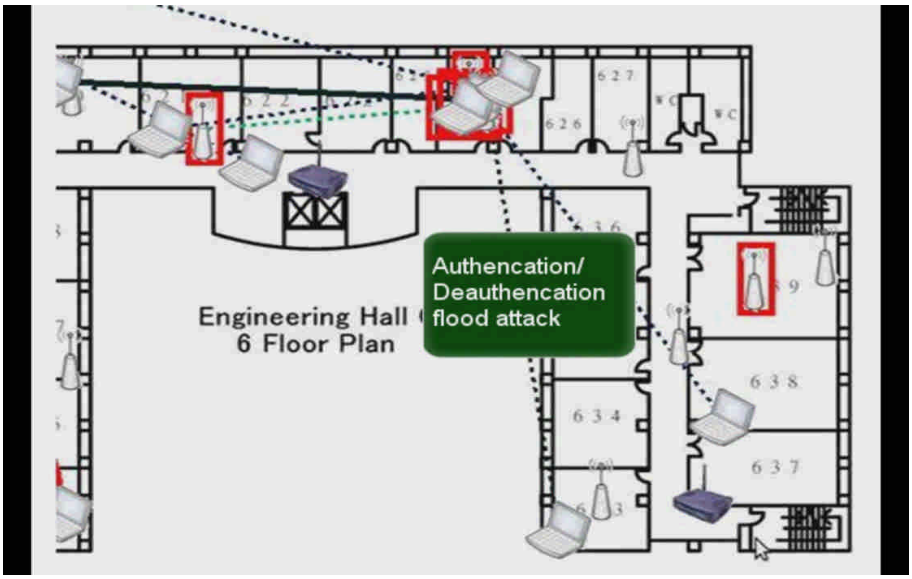


圖 4-14、成功分析攻擊類型

■ **入侵偵測系統強度評估系統 (Simple IDS Informer，簡稱 SIDSI)**

檢測一個入侵偵測系統的強度最好的方式就是直接採取「測試攻擊」，藉由模擬各式攻擊，觀察該入侵偵測系統是否能正確偵測並發揮其該有的功能。SIDSI 是基於一個以 C 語言開發的封包產生器 (packet generator)，輔以基本的 socket programming (UNIX Socket)來開發的入侵偵測系統強度評估系統。SIDSI 主要可檢測與了解入侵偵測系統 (Intrution Detection System，簡稱 IDS)的能力，以及探索是否有可能規避受測 IDS 的方法

一般網路封包的建置是從應用層開始，逐層往下傳遞、建置，直到最底層的實體層將封包送出。但 SIDSI 建置封包的程序與一般常規程序相反，是由 Layer 2 開始進行封包建置，由下往上一層一層封裝而成，SIDSI 可建置出封包內容 (payload)、IP 層表頭 (Header) 參數以及各種相關協定封包，並變造出各式封包，藉由變造異常封包或是封包異常發送來完成各式攻擊，以評估受測機器 (Device under Test，簡稱 DUT) 抵擋和防禦該攻擊之能力。

以下簡介 SIDSI 支援之功能：

- SIDSI 以 SNOT 封包產生器為基礎，並支援最新的 SNORT 入侵偵測的規則。因此 SIDSI 可發送符合最新偵測規則的攻擊封包，若是 DUT 無法阻攔該封包，則表示該 DUT 並未更新至最新版本的入侵偵測規則，而可能使得攻擊者有機可乘。此外，SNORT 本身為了能進行跨封包之規則比對，找出不同封包間的關係，它提供了 Flowbit 之選項。而 SIDSI 為了完整支援 SNORT 的規則，必須要能夠判讀 flowbit，因此 SIDSI 亦支援 flowbit，可產生相對應的測試封包。
- SIDSI 支援多種泛濫 (flooding) 攻擊測試方法，例如 TCP、UDP、ICMP 與 IGMP 等協定的泛濫攻擊。SIDSI 以如此的測試方式，來進一步得知待測系統 (Devise under test，簡稱 DUT) 的應對策略，評估該 DTU 入侵偵測系統的能力。以下為 SIDSI 內的相關攻擊函數：DoS_TcpSynFlood()、DoS_UdpFlood()、

DoS_IcmpFlood()、DoS_IgmpFlood()。

- SIDSI 支援掃描埠 (Port Scan) 攻擊，MakePacket_TCP_Scan_Unknows()為 SIDSI 提供之 port scan 的主要函式，主要利用 TCP 的 SYN-ACK 機制來達成掃描未知通訊埠之目標。
- SIDSI 可藉由編造不正常的封包來達成攻擊目的，以下為 SIDSI 中的相關函數：
    - MakePacket_TCP_LAND()
    - MakePacket_TCP_BROADCAST()
    - MakePacket_TCP_BAD_FLAG_NULL()
    - MakePacket_UDP_SMURF()
    - MakePacket_ICMP_OVER_PING_LENGTH()
    - MakePacket_IGMP_BAD_L4_SIZE()
- SIDSI 支援封包切割發送，此功能可進階測試使用 pattern matching 技巧的 IDS，瞭解其防禦的能力。許多使用 pattern matching 技巧的 IDS 雖然可以偵測出含有敏感字串之惡意封包，然而一旦將封包切割後發送，這類型的 IDS 便會失效。因此，SIDSI 將含有敏感字串的封包加工、分割後發送，測試 DUT 是否能夠成功地偵測被分割的敏感字串，評估該入侵偵測系統的進階能力。
  MakePacket_Diveded()為便是負責執行分割封包之函式。
- SIDSI 支援封包加密。考慮到真實世界中的特殊攻擊方式，例如駭客以加密封包的手法規避偵測，SIDSI 也支援封包加密的功能，用以進一步測試 DUT 抵擋此特殊攻擊手法的能力。SIDSI 提供函式以支援不同的加密方法：
    - MakePacket_TCP_Solarisx86_telnetd_TTYPROMPT_Encoded()
    - MakePacket_TCP_NetscapeAdminPasswd_Encoded()

● 行動裝置滲透檢測

在行動裝置滲透檢檢測方面，我們已開發與建置以下 8 種子系統與工具：Android 行動裝置惡意網頁檢測工具、Android 應用軟體安全漏洞檢測工具 (G-exploit)、Android 動態惡意軟體程式檢測工具、文件檔案惡意程式檢測系統 (Forenser)、動態惡意軟體分析檢測工具 (Malware Behavior Analyzer, MBA@TWISC)、大規模遠端系統滲透測試網 (Remote System Penetration Testing Network，簡稱 RSPTN)、使用者敲鍵行為辨識系統以及 Secure Wireless Overlay Observation Network (SWOON) 實驗平台。以下將分別介紹各個子系統與工具。

■ Android 行動裝置惡意網頁檢測工具

Android 行動裝置惡意網頁檢測工具可即時保護 Android 使用者在瀏覽網頁的同時，免於網頁端惡意物件的攻擊。我們針對 Android 手機瀏覽平台異於一般瀏覽器的特殊架構，利用 JavaScript 研發可掛載在伺服器端的 Android 行動裝置惡意網頁檢測工具，有效過濾來自伺服器端以外的動態資料對伺服器或其他瀏覽者的攻擊，並針對

JavaScript 本身語法上的弱點做修補，對特定不安全的或可被用來做攻擊的函式封裝成更安全的函式。

由於 JavaScript 承襲了部分 Java 語法上的物件導向特性，包含指標及類別的全面物件化，傳統在 Java 上特有的攻擊概念易被延抄到 JavaScript 上，故此類型的研究在 Java 上即有不少理論，做法上大致可分三類，其一，修改瀏覽器，使瀏覽器直譯 JavaScript 語法時最佳化，並修改 JavaScript 上的弱點；其二，建立一 Proxy 或類似於 Virtual Machine 的機器先行直譯並執行 JavaScript，若無安全上的考量，在將原始碼在客戶端的瀏覽器上直譯並執行；鑒於上述方法的負載過高，故採第三種具有低負載，且能達成上述目的的 Inlined Reference Monitor 技巧，亦即對於涉及相關安全問題的 JavaScript 函式做一 Overlay，使客戶端瀏覽器無法直接執行此類函式- Android 行動裝置惡意網頁檢測工具所開發的函式，在不改變函式名稱及參數傳遞的類型為前提下，不僅使呼叫函式的方法不需改變，該工具卻覆載了原始的函式，從新封裝成新的、安全的函式可供呼叫；亦即客戶端間接的透過該工具來傳遞參數並呼叫可信賴的函式。

圖 4-15 為 Android 行動裝置惡意網頁檢測工具所提出的模型，藉由了解 Android 系統架構與安全機制，建置惡意語法檢測工具於網頁原始碼中，當使用者利用 Android 瀏覽網頁時，針對該頁面上的所有語法都將是我們欲處理的對象。圖 4-16，Android 行動裝置惡意網頁檢測工具中的 Browser Detector 將對使用者所使用的瀏覽器類型先行做確認。圖 4-17，若經使用者確認並非使用 Android 相關的瀏覽器瀏覽網頁，則 User Confirm 將送出 No 的訊號給 Html engine & JavaScript engine，並且不提供任何保護。圖 4-18，若經使用者確認為使用 Android 瀏覽網頁，則 User Confirm 將送出 Yes 的訊號給 Html engine & JavaScript engine，以啟動專為 Android 所設計的語法規則過濾引擎，此語法過濾規則即採上述所提及的 Inlined Reference Monitor 技術做處理，可將不安全的函式經由重新封包的過程轉為安全可信賴的函式供瀏覽器使用，以達到保護使用者免於針對 Android 相關漏洞或其特殊結構而造成的攻擊行為。

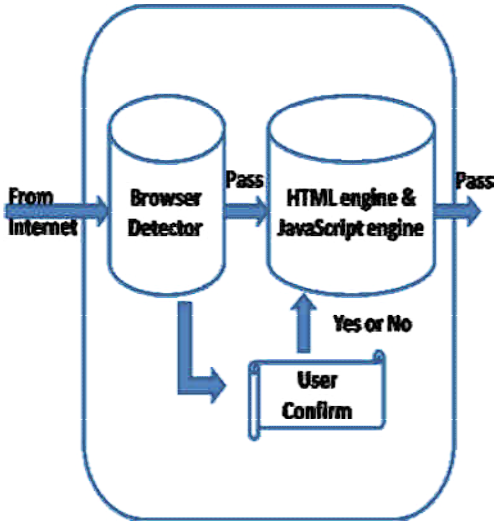


圖 4-15、Android 行動裝置惡意網頁檢測模型

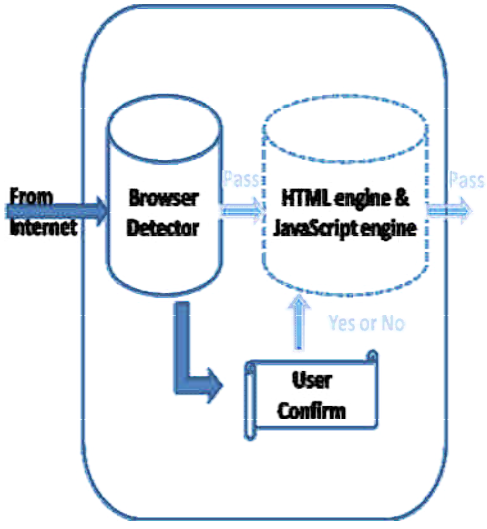


圖 4-16、確認使用 Android 瀏覽網頁

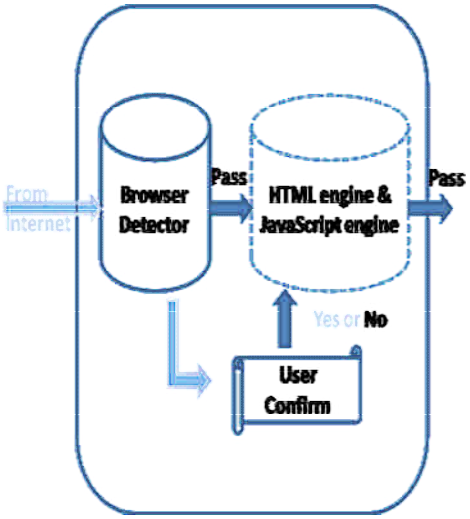


圖 4-17、非 Android 瀏覽流程

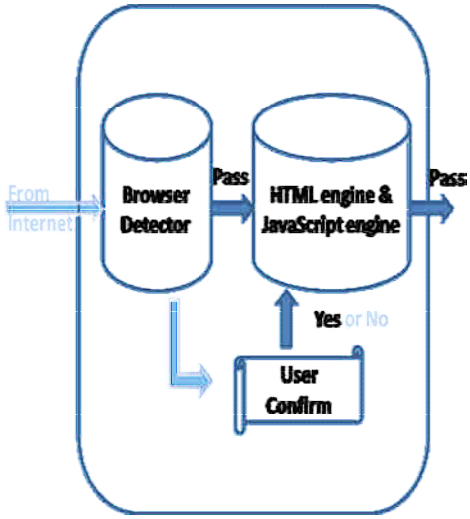


圖 4-18、Android 瀏覽流程

Android 行動裝置惡意網頁檢測工具針對 Android 手機瀏覽平台，強調即時檢測與低運算負載，具有下述優點：1. 專為 Android 特殊架構所設計，對於易造成 Android 危害的函式從新封裝。2. 不需修改瀏覽器，使客戶端不需安裝或更新瀏覽器即可受到保護。3. 呼叫函式的方法不需再被重新定義，可在不影響程式設計者的前提下提供完善的安全保護。4. 不禁止使用特定不安全的函式，使網頁的功能可以完善的被呈現。5. 易於修改及增強規則，以應變新的、未知的 Android 漏洞，達到保護使用者於使用 Android 行動裝置瀏覽網頁時的安全。

■ **Android 應用軟體安全漏洞檢測工具 (G-exploit)**

隨著科技的進步，手機裝置平台本身就已經是一個作業系統，強大的功能讓手機不再只是一個連絡工具。Google 推出了新一代的手機平台 Android，其系統核心為 Linux，並在其上面執行 java 應用程式。Android java 程式和原生的 java 程式並無不同，因此原生的 java 程式所遭遇的安全議題也可能出現在 Android 應用程式上。然而 Android 上的 java 執行環境有別於一般 java，編譯器會將 Android java 程式碼編譯成名為 DALVIK bytecode 的中介碼，此 DALVIK bytecode 會在 DALVIK 虛擬機器上執行，藉此讓程式受到保護並統一發佈格式。因此，檢測 Andriod 應用軟體之第一步便是將 DALVIK bytecode 轉回 java bytecode；轉回 java bytecode 後，便可利用 Java 平台上的軟體檢測工具進行軟體檢測。我們開發的 Android 應用軟體安全漏洞檢測工具 (G-exploit) 便是以此概念為開發基礎。

G-exploit 支援 Android 平台上的 DALVIK bytecode 與 Java bytecode 之間的轉換；轉回 java bytecode 後，G-exploit 可對 java bytecode 進行軟體安全檢測。G-exploit 並且簡化了複雜且繁瑣的檢測步驟，支援多種檔案格式，例如 Java 原始碼、jar 檔和 dex 檔等等。由於 Java 本身就是一個以安全為優先考量的程式開發語言，因此 Android 應用程式之檢測問題絕大多數皆是導因於寫作的壞習慣或是寫作規則的錯誤，極少發現可能造成重大傷害的安全性漏洞，但是由於 Android 之普遍性和高度應用性，我們絕

不能輕忽檢測 Android 應用軟體之重要性。目前 G-exploit 可以檢測到的 Android 問題包含以下三大類。

1) 正確性的問題：程式設計人員沒注意到之寫作上的錯誤，可能成為真實的程式錯誤。
2) 寫作壞習慣：程式設計人員寫作時的壞習慣，例如解構子之濫用，可能導致不必要的錯誤。
3) 異常的程式碼：不確定的程式行為，如未被確認的轉型等等，皆被視為異常的程式碼，極有可能造成安全問題。

此外，為了彌補單一檢測工具之不足，並兼具不同檢測策略之優點，G-exploit 包含了靜態分析的 Findbugs 工具與動態分析的 jcute 工具。G-exploit 首先使用靜態偵測掃描程式碼，試圖分析程式中的每一條路徑，找出程式中潛藏的安全性漏洞。接著，再用動態偵測驗證以靜態分析所找到的安全性漏洞。G-exploit 兼具靜態與動態偵測的檢測方式，大幅地降低誤判的可能性，且能精確的指出錯誤所在。圖 4-19 為 G-exploit 之架構圖。

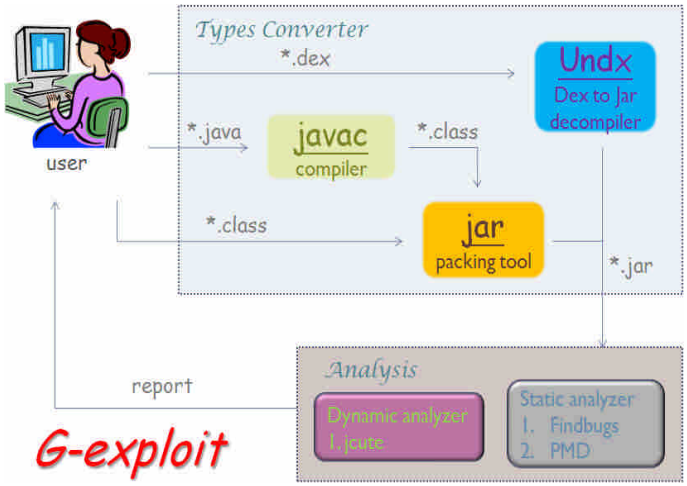


圖 4-19、G-exploit 架構圖

為協助使用者上手、提供更簡易的操作步驟，本計畫建置一個開放式的網頁平台 (見圖 4-20) 提供 G-exploit 線上檢測服務，使用者只需直接上傳 Andriod 應用程式，不必重新安裝 G-exploit 軟體，亦能得到 G-exploit 之即時檢測報告。

圖 4-20、G-exploit 之網頁平台

■ **Android 動態惡意軟體程式檢測工具**

手機發展日新月異，所能做的事情已超越過去所賦予的功能。以往手機給人的印象僅在於打電話，但是現在手機不僅能夠上網瀏覽網頁、編輯文件，甚至還可以讓使用者在手機平台上自行開發並撰寫程式。而 Google 推出的 Android 是原始碼開放的作業系統，所以在開發上有相當大的發展空間。

目前 Android 平台也開始受駭客覬覦。原本只存在電腦上的病毒，網路蠕蟲、rootkit…等惡意程式碼正快速地研究、開發至新的手機版本進而染指到手機平台上，將造成新一代手機用戶的資料受損，個人資料外洩…等嚴重問題。為了防止這些問題出現造成重大損失，所以發展手機動態惡意軟體程式檢測工具勢在必行。

在開發 Android 動態惡意軟體程式檢測工具之前，我們也做過相關的惡意程式檢測研究，研究方式最主要是利用虛擬機器 (Virtual Machine) 模擬 x86 CPU 環境。我們利用 emulator QEMU 來建置環境，以掌握所有資料流動。追蹤物件的細緻度決定了分析的準確度，分析越低層次的物件如 CPU 暫存器或記憶體位元組會得到更詳細的資料所以更能得到完整的分析。因此，我們將 Android 的 VM Dalvik 架設在 QEMU 之上，x86 的暫存器如圖 4-21 表示。其中 EAX/AX 為累積暫存器、EBX/BX 為基底暫存器等，各個暫存器都有其特定的工作。在 98 年我們主要工作是了解各暫存器的工作進而做出完整的整理。

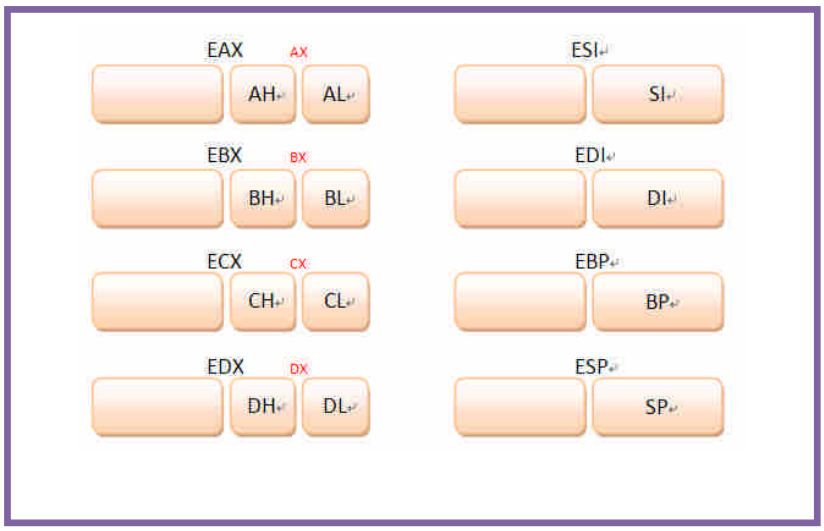


圖 4-21、x86 暫存器

x86 中的暫存器：EAX: Accumulator Register, EBX: Base Register
ECX: Count Register, EDX: Data Register
ESI: Source Index, EDI : Destination Index
EBP: Base Pointer, ESP: Stack Pointer

但 Dalvik 開發環境不是在複雜指令集 (CISC)的 x86 CPU 開發，而是在簡單指令集 (RISC)的 ARM CPU 上。複雜指令集的 CPU 的指令較簡單指令集的 CPU 多，而且指令也複雜很多，如：PUSH, POP 以及 BSWAP (交換 32-bit register 再累加)等高級指令都是 ARM 所沒有的；複雜指令集 CPU 的指令長度不定，RISC 的每個指令長度均為 32-bit。複雜指令集和簡單指令集之間還存在相當多的差異，由於以上的不同，因此我們無法將原本在 x86 的研究成果直接套用於 Dalvik。此外，Dalvik 使用自己的方法重新定義 register 的功能，而 x86 所使用的 EAX, ECX 暫存器都被刪除，所以原本 x86 能做的追蹤，在 Dalvik 上都不能成功。Dalvik 在 x86 的版本下所使用的暫存器分別為 EDX, ESI, EDI, BX, BL,以及 BH。每個暫存器功能也和 x86 的暫存器用法不一樣。例如：EDX 在 x86 是進行 I/O port 執行、運算以及一些中斷呼叫，但在 Dalvik 卻是用來翻譯程式計數器 (Interpreted program counter)取得指令。 由於 Dalvik 的暫存器使用方式和 x86 系統不一樣，並且 Dalvik 所建置在 ARM CPU 上的關係，指令及部分也和 x86 完全不同以致無法順利將之前在 x86 上的研究直接用到 Dalvik 上，所以接下來方式即是參考先前對 x86 系統的追蹤方式，建置到 ARM 系統上面，並且更改暫存器對應的方式，使其符合 Dalvik。

圖 4-21 是未來要建置的 Android 惡意軟體程式檢測流程。在 QEMU emulator 內部，我們加入了惡意行為的追蹤。它負責將一連串惡意行為記錄下來並將記錄交給 Behavior classify 做分類，再交由 Database 去做可疑項比對分析，最後比對出來的結果再通報給使用者。

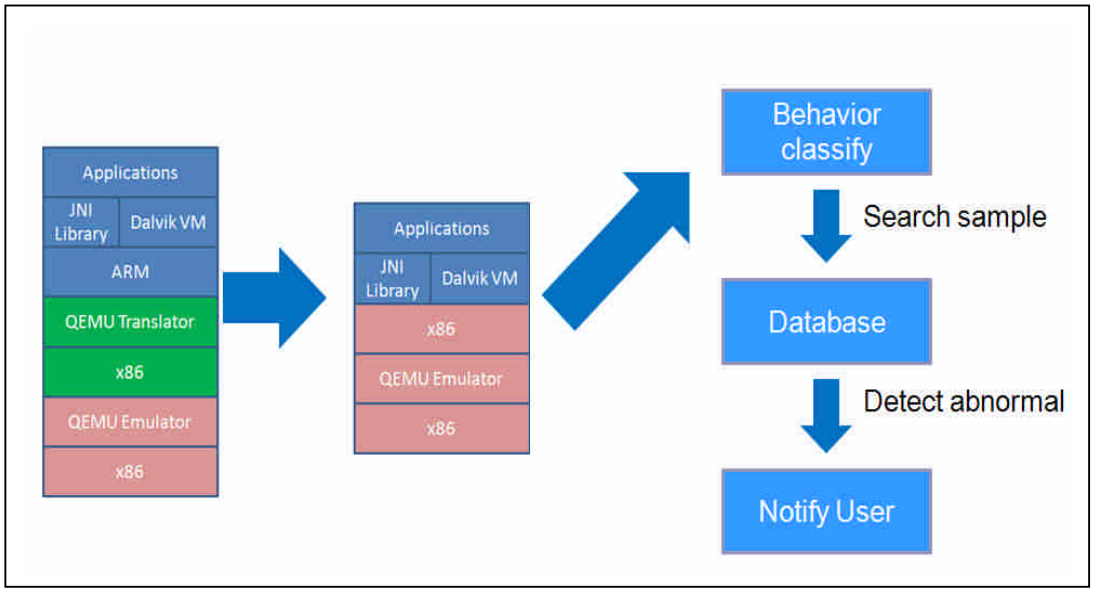


圖 4-22、惡意軟體程式檢測流程

■ **文件檔案惡意程式檢測系統 (Forenser)**

Forenser 是自動化惡意程式檢測系統，綜合多項惡意程式鑑識程序，偵測隱藏在文件檔案內的惡意程式碼，並提供完整的掃描報告利於專家判讀。Forenser 掃描檢測對象可針對單一檔案或單一資料夾，對於 Windows 執行檔及非執行檔採取不同的流程處理，過程包含檔案格式判定、加殼掃描 、PE header 解譯、可疑字串掃描、可疑 API 檢測、程式區段熵分析、已公布 OLE vulnerability 檢測、Call/Pop 掃描、PEB/TEB/SHE Loading 掃描等，其應用領域廣泛，可被整合為企業級的惡意程式掃描器。

Forenser 檢測流程以檔案格式為判別基準，分為 Windows 執行檔之檢測流程與非 Windows 執行檔之檢測流程二大項目。圖 4-23 為針對 Windows 執行檔之檢測流程，此流程包含了檔案可疑字串分析、可疑 API 檢查、程式區段熵分析及程式加殼掃瞄最後製作檢測報告 。可疑 API 檢查將判斷程式是否使用了可疑的 Win32 API；程式區段熵分析則會計算檔案各區段的資訊複雜度，若過於複雜即有加密或加殼的可能；最後再進行加殼掃瞄，即可正確判斷檔案是否已加殼。

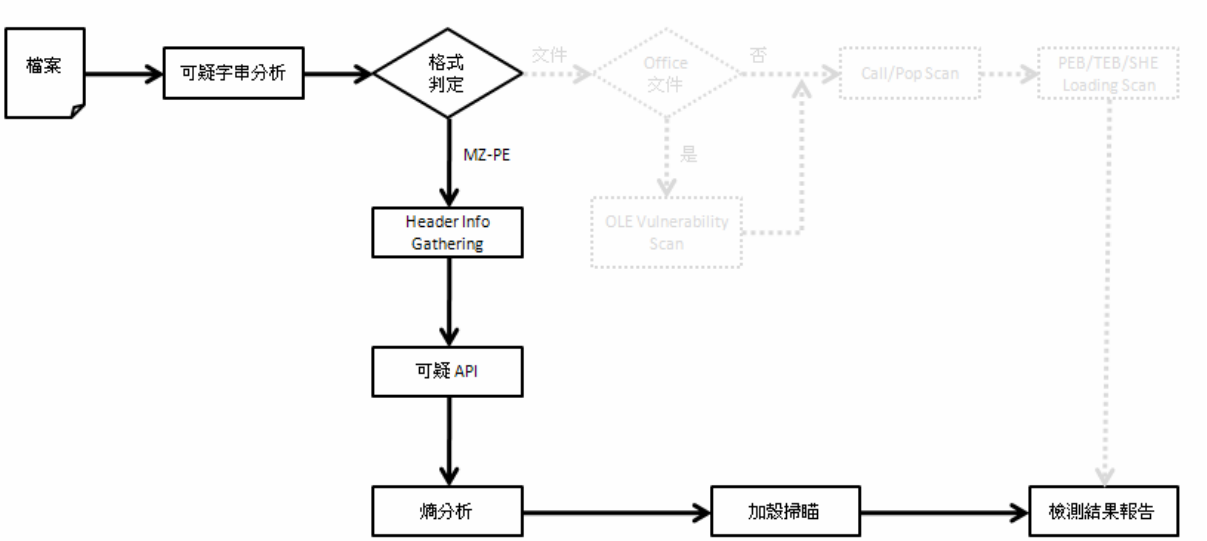


圖 4-23、Windows 執行檔檢測流程

圖 4-24 為非 Windows 執行檔之檢測流程，此流程包含了檔案可疑字串分析、Windows Office OLE 弱點掃瞄、Call/Pop sequence 掃瞄、PEB/TEB/SHE Loading 掃瞄與最後的檢測報告產生。OLE 弱點掃瞄將掃描 Microsoft Offie 文件，分析是否存在已公佈的 OLE Vulnerabilities；Call/Pop sequence 掃瞄負責檢測文件是否使用 call/pop 的方式來得知自身程式碼所在的記憶體位址，進而利用此記憶體位址執行相關的惡意程式；PEB/TEB/SHE Loading 掃瞄可得知該檢測檔案是否藉由 PEB/TEB/SHE Loading 來讀取相關的 Win32 API，並且搭配已知的自身程式碼位址，使其可在不適當的時候跳到自身程式碼位址執行。

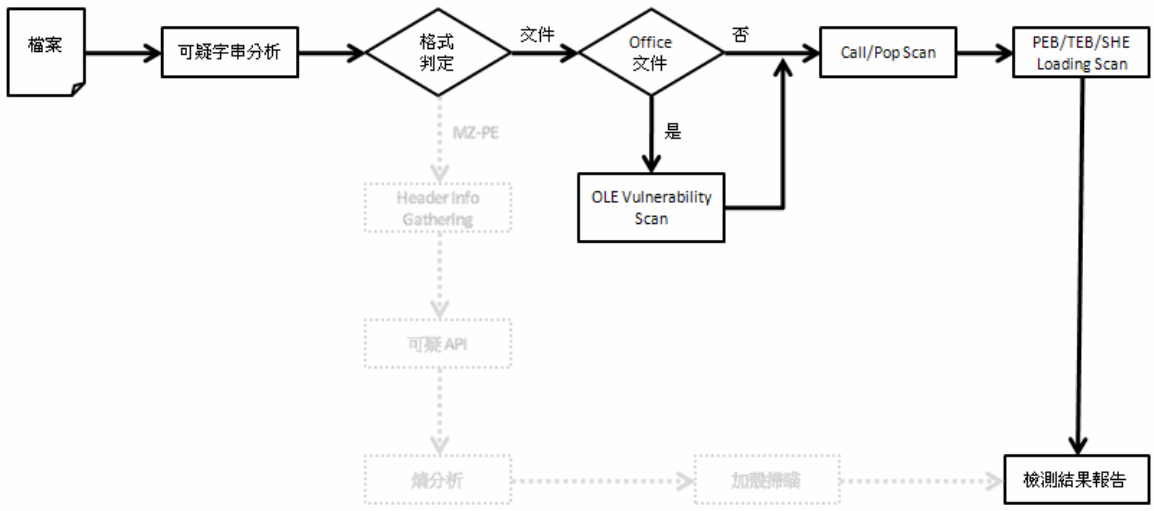


圖 4-24、非 Windows 執行檔檢測流程

Forenser 除了提供完善且複雜的檢測流程，亦提供簡易的使用者圖形化介面 (如

圖 4-25)，以協助使用者輕鬆判讀檔案的資訊，並且儲存掃描結果。Forenser 完善的檢測流程與簡易的操作方式，使其可應用於許多領域，如 Forenser 可成為一般使用者電腦裡的惡意檔案掃描器、偵測檔案是否被入侵，Forenser 可分析程式行為是否惡意，Forenser 亦可掃描電子郵件的安全性。

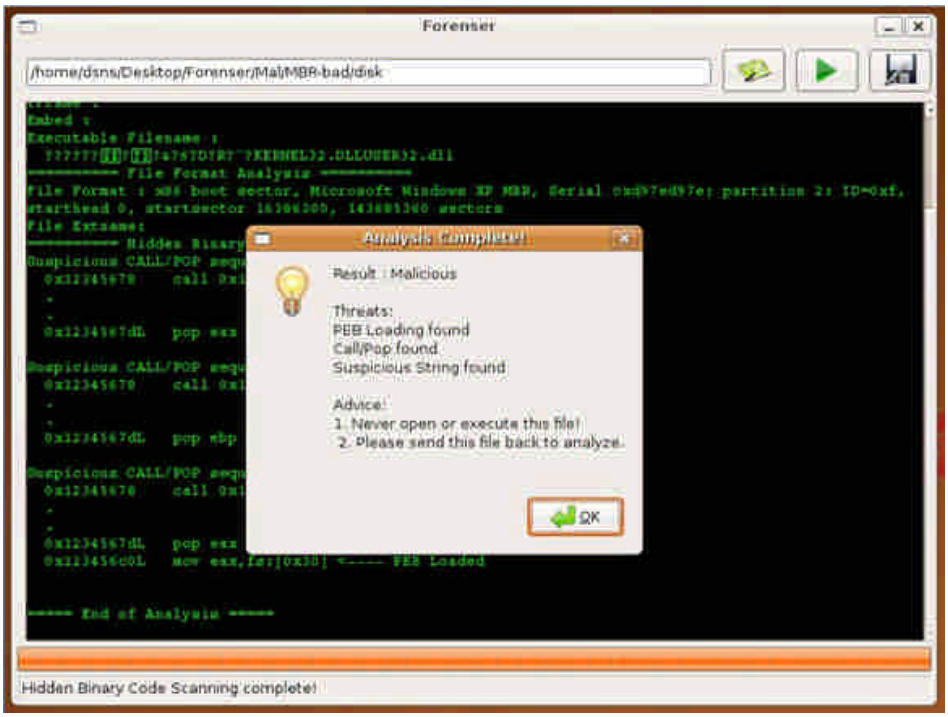


圖 4-25、Forenser 圖形化介面

■ **動態惡意軟體分析檢測工具 (Malware Behavior Analyzer, MBA@TWISC)**

MBA@TWISC (Malware Behavior Analyzer)是惡意程式行為分析工具，利用 QEMU 虛擬機器 (Virtual Machine)模擬 CPU 執行指令的過程，並記錄虛擬記憶體和虛擬 I/O 設備間的資訊流動，進而找出惡意資訊汙染源，以及完整的惡意程式行為(malware behavior)。

由於過去的防毒軟體藉由病毒特徵碼的方式進行偵測，對於具有變種能力的病毒，如加殼 (Packing)：將原本的程式加密、壓縮後使原始程式碼特徵大不相同，並在程式標頭檔上加一前導的解壓縮程式，使程式散佈過程中不易被偵測，卻又能自行解殼直行。又如多行 (Polymorphism)：可說是加殼的進階技術，利用不同加密金鑰改變每次散佈出去的特徵，以增加病毒碼特徵收集的難度。變形 (Metamorphism)：藉由 x86 提供複雜的 CPU 指令集，建立自動化的變體引擎，可將同樣目的的程式碼改寫成不同 CPU 指令集的組合病毒碼特徵收集的難度。以及 P2P botnet 可利用 P2P 來達到病毒之間的聯繫即更新，當將使過去的靜態偵測方式喪失了即時捕捉新型病毒的能力；同時面臨網路傳播的迅速，新型病毒傳播的速度遠高於病毒特徵更新的速度，防毒軟體的確更顯無力。綜合以上觀點，我們認為被動的收集病毒碼特徵已難面對如此複雜的病毒發展技術，而是必須以行為分析的技術找出病毒碼的行為，為達此目

的，藉由模擬 CPU 的執行，動態找出資訊在記憶體或暫存器甚或是 I/O 設備間的資訊流動情形，可對程式執行時期的行為進行鉅細靡遺的分析，則能進一步預測出該惡意程式的行為降低惡意程式對系統所產生的威脅。

動態惡意軟體分析檢測工具的目的在於預先執行程式，並在程式執行期間，即時捕獲資訊在記憶體、暫存器、I/O 設備間的流動過程。針對這些資訊流動我們可依需要訂定相應的監視規則。通常在實做上，均將網路卡自網路接收來的封包視為汙染源，即將網路卡的緩衝區的資料做為動態偵測的起點，並在執行過程中監視某重要的系統呼叫 (例如：execve)是否接收到此資訊流，如此便可檢查是否有外部網路的資訊進入到系統呼叫中。同時，如果我們監視 CPU 的 Program Counter 是否接收到此資訊流，便可準確的偵測出記憶體覆寫的攻擊。

目前許多關於動態分析程式的研究，僅針對為單一程序提供模擬分析環境，然而我們所面臨到大多數的惡意程式皆以入侵作業系統核心的手法來隱藏，為了即時捕獲系統層次的資訊流動，我們需要模擬整體系統的運作，包括 CPU、記憶體、各項周邊設備如硬碟、網路卡、鍵盤、滑鼠等等，工程十分繁複。MBA 奠基於目前已發展成熟的模擬軟體 QEMU，於其上加入動態分析程式的功能，QEMU 是一模擬 X86 電腦系統的虛擬機器，包含對 X86 CPU 的模擬、虛擬記憶體的映射，以及各項以軟體虛擬出的周邊設備；簡言之，使用者可利用 QEMU 模擬出一台虛擬主機，於其上安裝作業系統與各種軟體，類似常見的 VMWare 系統。我們修改 QEMU 中的 CPU 模擬器、虛擬記憶體映射功能以及各項虛擬周邊，以追蹤整體主機資訊的流動過程。

虛擬機器分為 Virtualization 和 Emulation，均是利用軟體來模擬電腦的虛擬機器，亦完整模擬記憶體 (即虛擬記憶體)和 I/O 設備 (即虛擬 I/O)，MBA@TWISC 採取能模擬程式指令在 CPU 執行的 Emulation 虛擬機器 (例如：QEMU)，而非將指令轉交給真實硬體執行的 Virtualization，故能順利掌握程式碼在 CPU 執行期間資訊流動的情形，以及資訊在虛擬記憶體及虛擬 I/O 間的流動情形；再藉由修改 QEMU 原始碼，使其在模擬 CPU 指令執行時能動態偵測程式執行過程，並完整的紀錄虛擬記憶體和虛擬 I/O 設備 (已支援含硬碟、鍵盤、滑鼠、網卡等)，藉此收集和分析各種惡意程式的行為，可達到預測惡意程式攻擊路徑的目的，降低惡意程式對系統所產生的威脅。

有了虛擬機器的輔助，在紀錄資訊流動上，我們必須為系統中每一個儲存資料的暫存器與記憶體位元組加上一個標記欄位，以記錄該位元組目前是被哪個汙染源所影響。此外，必須修改 CPU 模擬的過程，為 X86 指令集中所有的指令加上記錄資訊流動的動作。如圖 4-26 所示：

程式碼	資訊流動過程
mov ebx, eax	ebx.tainted_by = eax.tainted_by
add [mem1], eax	[mem1].tainted_by.append(eax.tainted_by)
ret	pc.tainted_by = [esp].tainted_by
jmp [mem1]	pc.tainted_by = [mem1].tainted_by

圖 4-26、記錄資訊流動範例

同時，針對不同的應用，我們依照設計不同的監視規則分做三項分別是記憶體覆寫攻擊偵測、資料偷取偵測還有 Shell Command 攻擊。首先是記憶體覆寫攻擊偵測，當攻擊者透過覆寫記憶體以執行其植入的程式碼時，最終必將令 CPU 的 Program Counter 指向該段記憶體，因此我們可將所有透過經由網路卡進到系統內的資訊標記成汙染源，一旦 Program Counter 指向了某處被該資訊流汙染到的記憶體區段，即代表有外來的程式碼即將被執行。由於此現象十分特異，且僅在攻擊者藉由網路進行攻擊時才發生，因此十分適合用動態惡意軟體分析檢測工具來偵測此類型攻擊。接下來是資料偷取偵測，許多木馬程式會偷取主機內部一些敏感的資料後，再透過網路傳回至駭客的手上。為了偵測此情形，我們將硬碟中許多敏感的密碼檔作為汙染源，並隨時監視是否有被汙染的資料被放置在網路卡的傳送佇列中，如此一來即可偵測敏感資料的外洩。值得注意的是，由於動態惡意軟體分析檢測工具會追蹤系統整體的資訊流動，因此即使木馬程式在偷取敏感資料後，將資訊加密後再傳至網路上，依然會被我們偵測出來。Shell Command 則是關於攻擊許多網路應用軟體提供使用者執行外部程式的功能，如 MSSQL 當中即有 exec 的關鍵字讓 SQL 的使用者可執行外部的程式，如此一來，如果網頁設計中藏有 SQL Injection 的漏洞，即可讓攻擊者遠端執行主機中的程式，但透過動態惡意軟體分析檢測工具，我們可將來自網路的資訊標記為汙染源，並監視是否該資訊流向至，用以執行程式的系統呼叫如 execve 中，如此便可偵測此類型攻擊。

- **大規模遠端系統滲透測試網 (Remote System Penetration Testing Network，簡稱 RSPTN)**

我們在 98 年已建置了一個網頁型的滲透檢測網路平台，簡稱 RSPTN，使用者可輕易地透過 RSPTN 對該電腦進行系統滲透檢測。RSPTN 除了提供易讀的檢測結果之外，對於滲透成功的項目也會提供給修正建議，讓使用者能夠提升其電腦的安全性。RSPTN 是以 nmap 與 metasploit 等網路工具為基礎，提供進一步的滲透檢測服務；因此在建置 RSPTN 的過程中，除了必須熟悉 nmap 和 metasploit 等工具外，我們還必須將這些工具與網頁功能互相結合，同時深入研究各種弱點的原因以及尋求可能的解決方法。這對於日後提升滲透檢測網路平台的完整性和功能性將有所助益，而我們也將針對不同的設備進行不同程度或不同類別的滲透檢測。

在介紹本計畫建置的 RSPTN 遠端系統滲透測試平台的眾項功能前，先得介紹此測試網路平台的應用核心－Metasploit Framework (MSF)。MSF 是 2003 年以開放原始碼方式發布、可自由取用的開發框架，這環境為滲透測試、shellcode 編寫和漏洞研究提供了一個可靠的平台。2.x 框架主要是由面向對象的 Perl 編程語言編寫的，最新版本 3.0 採用了 Ruby 開發。它集成了各平台上常見的溢出漏洞和流行的 shellcode，並且不斷更新，最新版本的 MSF 包含了 176 種針對當前流行的操作系統和應用軟件的 exploit，以及 104 個 shellcode。而做為安全工具，它在安全測試中起發的作用也不容

忽視。合理的利用 MSF，將為漏洞自動化偵測以至及時修補系統漏洞提供強而有力的保障。同時作為開發，也大大降低了 Exploit 的開發週期和對開發者背景知識的要求。

MSF 的強大不僅僅表現在 Exploit 自動生成和開發的簡捷性上，另一個引人注意的是它的 exploit 集成性。你可以隨時對 MSF 進行擴展，加入自己的 exploit 或者 shellcode。這意味著，任何一個新的 exploit 的誕生都能快速的集成到 MSF 中，而且基於模塊化的 MSF，讓動態的 exploit 開髮變的非常簡單。這種框架化的 exploit 開發也必將成為趨勢。

因此，架構在 MSF 之上的 RSPTN 遠端系統弱點測試網路，配合上 Nmap (Network exploration tool and security scanner) 指令的輔助，先經由程式內部自行定義的幾個通訊埠對應的指紋資料，對測試端查詢各通訊埠的服務為何，再啟用 MSF 的弱點攻擊測試，對各個已開啟的通訊埠進行封包遞送測試。RSPTN 集結各種網路安全測試的功能，提供使用者在 RSPTN 的測試平台上，檢視自身電腦的系統服務狀態及各連接埠的運作狀況，並對各開啟的連接埠進行安全強度的分析與回報。

使用者在進行 RSPTN 的系統弱點測試後，系統將立即顯示測試結果，詳列各已開啟連結埠的系統服務項目，並表列各連結埠對不同攻擊封包的反應。一旦發現測試端電腦存在潛在攻擊的弱點，RSPTN 的測試結果即會顯示該弱點的詳細資訊及相對應的修補方法，以利使用者能清楚而快速地瞭解弱點所在，並能立刻對該弱點進行相對應的修補，如提供更新程式下載、關閉系統服務等解決之道。

此外，RSPTN 提供註冊使用者能紀錄不同時間系統弱點測試的結果，藉由各項紀錄的分析，配合圖表顯示的統計與對照，能讓使用者迅速比較系統在各狀態下的安全強度，並對照弱點修補前後的安全指數。而對於擁有大量電腦的企業或團體用戶，RSPTN 也將繼續加強大規模的系統弱點測試功能，讓網管人員能簡便地同時對眾多電腦進行檢測，並回報測試結果分析，甚至能做自動化的 Schedule 的檢測安排，定時定期地對電腦進行固定檢測，以維持電腦都能擁有最新的弱點抵禦能力。

■ **使用者敲鍵行為辨識系統**

傳統密碼認證系統僅需要使用者輸入正確的密碼組即可通過認證，因為認證機制的單純，密碼容易遭到破解，導致許多使用者資料遭到竊取或盜用。為了有效加強認證系統的安全性，目前已有許多研究利用生物特徵來輔助傳統密碼的認證，如語音辨識、指紋辨識、虹膜辨識等等，雖然大多生物特徵辨識系統都可提升認證系統安全性，但大多生物特徵認證方式都需要另行採購或是加裝硬體設備，使得安裝此認證系統的成本大幅上升，讓系統實用性降低。漸漸的，有部份學者將研究方向轉往生物行為的認證，許多國內外學者研究利用使用者敲鍵行為來辯證使用者身分，輔助傳統密碼認證系統，阻擋行為模式不符的使用者，經過多年研究，目前已證實可有效加強系統安全。此外，由於敲鍵行為辨識採取生物特徵的設備是經由每台電腦皆需安裝的鍵盤，無需再加裝其他硬體設備，不僅降低使用成本，也增加實用性。

早期有關於敲鍵行為變的研究大多是利用數學統計的方式，利用以往統計的數據建構生物行為特徵，進而判斷當前的使用者是否合法。如 Gaines et al 的研究是請他的

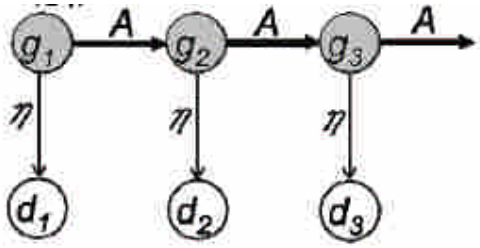
七位秘書輸入一篇文章，利用 t-test 統計方法來建立敲鍵行為的特徵；Joyce and Gupta 則採用較為直觀之方法，量測使用者輸入字元時間間隔，將此時間間隔跟以往的數據做比較，進而判斷使用者；Monrose and Rubin 利用分群法根據打字速度以及其他三種分類法 (Euclidean distance measure、non-weighted probability 以及 weighted probability measure)將使用者分類；Magalhaes et al 利用先前 Revett 及 Khan 所發表有關於敲鍵行為辨識的方法，再導入 lightweight 演算法進行辯證；Guven et al 則是根據鍵盤上按鍵之間的距離建立向量，再利用此向量與以往類似向量做比較；Hocquet et al 合併三種分類法 (classical method、measurement of the disorder 以及 discretization of the time)將使用者分類；以上諸多研究都偏向於利用統計學方式來建立敲鍵認證模型，然而他們使用的測試密碼長度大部分都超過了 14 個字元，但對誤率卻超過 5%，

直到近十年，許多研究開始針對此問題進行解決，大多採用 Machine Learning Method 來建立使用者敲鍵行為模型，提升判斷準確率，1997 年時部份學者甚至導入神經網路分析(Neutral Network Analysis)方法。Ru et al and Araujo et al 利用 Fuzzy Logic 演算法針對敲鍵時間間隔、按鍵距離以及個按鍵的組合建立模型。漸漸的，SVM(Support Vector Machine)以及 PCA (Principle Component Analysis) 也被導入了敲鍵行為模型。接著 Haidar et al 整合了神經網路分析、Fuzzy Logic、數學統計方法以及幾項方法的結合發展出另一個敲鍵模型。然而，以上研究雖然解決了以往的問題，卻往往會使認證系統變的不穩定。以 SVM 來說，通常會花很多的時間進行 model 的 training，耗費龐大系統資源，因為沒有辦法及時判斷使用者的合法性，導致這些研究方法沒有辦法實際被應用在認證系統。而我們選擇 HMM (Hidden Markov Chain)來建立敲鍵模型，HMM 在應用上有著幾項優勢，首先，根據每個使用者的敲鍵特徵，都會擁有自己的 HMM 模型，即使有人註冊了新帳號，我們也僅需要建立另一個 HMM 模型，並不會造成系統的不穩定；第二，HMM 模型很容易被建立，不需要龐大的系統資源；第三，HMM 運算的複雜度較低，一般用來建立使用者行為的模型時間複雜度為n^2，而 HMM 則只有 n。

儘管如此，眾多研究仍然存在一共同問題，敲鍵行為特徵與語音、虹膜、指紋等生物特徵不同，生物行為較容易改變。於是近幾年開始有人利用 adaptation mechanism 來更新重建模型。如 Bleha et al 使用最小的按鍵距離以及 Bayes Classifier 來進行使用者變證，每周再利用最近 30 筆登入行為資料來重建辯證模型；Monrose et al 則利用登入時的行為模式對使用者資料加密，並利用最近幾筆資料來跟新使用者行為模型；Araujo et al 將登入成功且跟以往平均值差不多的樣本加入模型，並且刪去最舊的樣本；Hosseinzadeh et al 應用高斯模型 (Gaussian Mixture Models) 進行辯證，且每次登入都將使用者模型進行更新。以上研究都使用最近幾筆資料來預測使用者行為，然而，要使用幾筆資卻是最大的問題，假如引用過多筆，那最近的行為模式變會不明顯，假如太少筆，系統則會變的不穩定，無法建立生物特徵。而在本系統裡則是透過實驗方式，採取上千個樣本，找尋最適合的樣本數目。

本系統則為避免以往問題所開發而成，以下我們將解說本系統運作基本原理，我

們整合了高斯模型 (Gaussian Model)、自回歸預測模型 (Autoregressive Predictive Model) 、隱藏式馬可鍊模型 (Hidden Markov Chain)以及數學統計方法來建立使用者的敲鍵行為辯證模型。首先，我們必須對使用者行為模式進行採樣，我們在這裡採取的為使用者輸入字元與字元間隔。利用採取到的字元問隔建立生物行為特徵。利用高斯模型，我們可利用採取的樣本建立高斯分布，因此我們可以算出使用者為合法的機率。我們再利用自回歸模型 (AR model)來預測使用者下一次輸入時的字元間隔。最後我們建立 HMM 模型，如下圖所示。



當我們將各參數運算完成，經由使用者自行設立的門檻，便可進行使用者行為的判斷，整體而言，本系統擁有以下優勢：1. 穩定的系統，對於每個新使用者，我們僅需要建立一個新的 HMM 模型即可以辯證，並且無需消耗大量的系統資源。 2. 耗費較小的系統資源，可以做即時的判斷。 3. 將使用者行為改變的趨勢視為一種生物特徵，加入判斷，避免使用者行為改變時產生誤判。除了擁有以上三項優勢外，本系統在我們測試後，其錯誤率也降低到 2.19%，已達可應用的層級。而我們目前正與 Windows 登入系統進行整合，在未來可望應用在相關登入系統。

■ **實驗平台 - Secure Wireless Overlay Observation Network (SWOON)**

無線與有線網路的差異，使得現有的有線網路安全技術無法套用於無線網路之上。有鑑於此，專家學者們紛紛提出適用於無線網路的安全防禦機制；然而該些研究卻缺乏一安全的、具彈性的實驗平台，以佐證新方法於真實世界中的可行性；同時也缺乏穩定的測試環境以重製實驗，確認新方法的穩定性與強韌度。網路模擬軟體雖可協助解決實驗環境欠缺之窘境，然對許多實驗而言，僅使用模擬軟體測試新方法仍嫌不足，例如模擬軟體粹取部分系統屬性的方式就無法模擬因硬體設備所造成的效能瓶頸。SWOON (Secure Wireless Overlay Observation Networks) 便是為了解決無線網路實驗環境欠缺所開發的無線疊蓋網路測試平台 (emulated testbed)。

圖 4-27 為 SWOON 的架構示意圖。SWOON 建構於美國加州大學柏克萊分校開發的 Defense Technology Experimental Research (DETER) 平台與美國猶他大學的 Emulab 之上。Emulab 是一有線網路實驗平台，提供實驗節點隔離機制，支援可重覆配置的有線網路安全實驗；DETER 為一用以研究網路安全議題之有線網路實驗平台，其建構與 Emulab 之上。SWOON 是一以 DETER 與 Emulab 為基礎，在有線網路實驗平台與有線網路安全平台之上所開發的無線網路安全觀測平台。SWOON 仿真模

擬 (emulate) 無線網路的特性，將仿真後的無線封包傳送至底層的 DETER 與 Emulab，DETER 與 Emulab 則負責將封包經由乙太網路傳送至其他的實驗節點，完成節點間的溝通。SWOON 的架構能支援多種無線網路技術如 WiFi 與 WiMAX，亦能支援網路安全應用與安全攻防實驗，具有高度彈性與可擴充性。

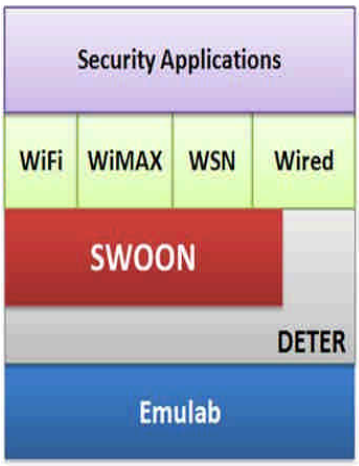


圖 4-27、SWOON 架構示意圖

圖 4-28 是 SWOON 的軟體架構圖，此架構核心為虛擬無線界面 (virtual wireless interface)，此虛擬無線界面又可細分為二部分：虛擬天線 (virtual antenna) 與 虛擬驅動程式 (WiFi driver 與 WiMAX driver)。在 SWOON「虛擬天線 – 虛擬驅動程式」之設計中，虛擬驅動程式負責與作業系統核心溝通、管理網路控制層(Media Access Control Layer，簡稱 MAC layer)以及資料封包的處理工作；虛擬天線主要負責模擬無線網路底層的傳送介質，將來自 DETER 的有線網路封包仿真模擬成為無線網路封包，並上傳給虛擬驅動程式。SWOON 目前支援 WiFi 與 WiMAX 的虛擬驅動程式，以下將介紹驅動程式和虛擬天線的實作細節。

– 虛擬驅動程式

1.WiFi (IEEE 802.11)：

我們修改業界實作 IEEE 802.11 協定最有權威的 Atheros 公司所生產網卡的 Linux 驅動程式。Atheros 公司的網路卡驅動程式由兩部分組成：一是公開源始碼，與作業系統核心接軌；另一部份則是礙於電信技術法規的限制而不能公開源始碼，其稱之為硬體抽象層 (Hardware Abstraction Layer，簡稱為 HAL)。HAL 可直接和無線網路卡硬體溝通的程式，並且提供上層一個統一的界面。由於 SWOON 模擬網路沒有存在真實的 Atheros 網路卡，因此 HAL 無法與真實的硬體進行溝通，所以 SWOON 採用自行實作的硬體抽象層，修改驅動程式中需與硬體溝通的程式碼，使其轉向虛擬天線進行資料的收送還有實體層資訊的取回。

2.WiMAX (IEEE 802.16)：

我們於 Linux 之上按照 IEEE 802.16 的標準文件，自行發展一個作業系統核心的模組，負責建立 WiMAX 的虛擬界面，以進行與應用程式間資料的傳輸，以

及處理相對應的控制訊息。由於 WiMAX 的虛擬驅動程式完全由 SWOON 開發團隊自行開發，因此與 WiFi 虛擬驅動程式不同，不需要進行 HAL 的模擬，便可直接與虛擬天線溝通，進行資料的傳送和控制訊息的處理。

– 虛擬天線

如前所述，虛擬天線主要負責仿真模擬 (emulate) 無線網路底層的傳送介質和實際機器交換資訊的工作，並提供基本的資訊統計功能。虛擬天線採用「封裝 – 解封裝」 (encapsulation – de-encapsulation) 的技術以實現無線網路的仿真模擬 (emulation)，其作法為：當虛擬天線接收到來自虛擬驅動程式欲送出的 IEEE 802.11 或是 IEEE 802.16 封包後，虛擬天線會將封包封裝成乙太網路封包，使 DETER/Emulab 能將封包送出。當虛擬天線收到來自底層 DETER/Emulab 的乙太封包後，虛擬天線首先解封裝封包，再依據 Coverage Table 紀錄的訊號衰減情形，決定是否丟棄或留下該封包。留下的封包會被傳送至虛擬驅動程式進行網路控制層的處理。

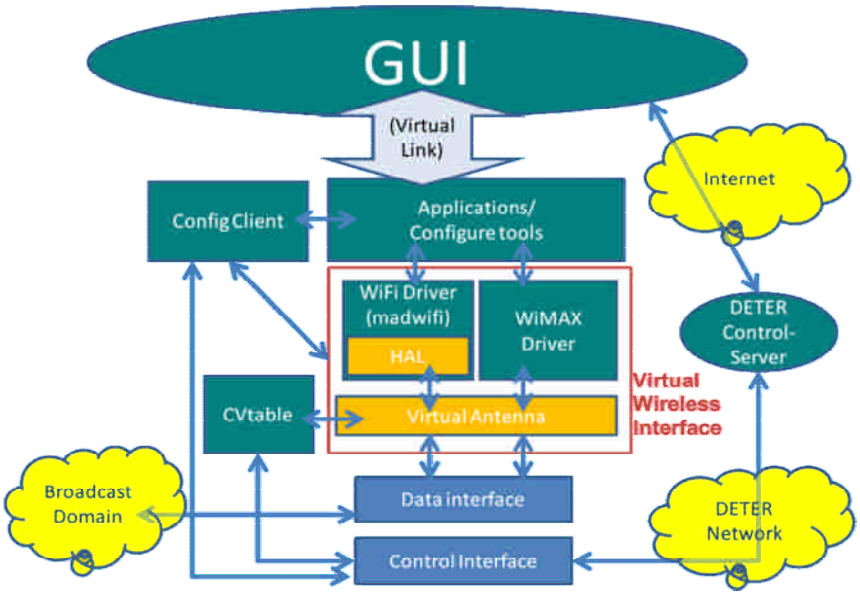


圖 4-28、SWOON 軟體架構圖

SWOON 目前雖僅提供 WiFi 與 WiMAX 的虛擬驅動程式，但此「虛擬天線 – 虛擬驅動程式」之設計使得使用者能有彈性地在虛擬天線上安裝多種無線網路界面的驅動程式，易於擴充平台支援的網路技術，並進行許多新興無線網路之安全實驗。

五、 系統效能評估成果

以下將介紹本計畫開發系統之效能評估成果。

● 異質無線多網與核心網路檢測

■ MoPT: 3.5G 滲透檢測系統

在 MoPT 無線行動網路滲透檢測系統中，我們已經完成網路拓樸探索工具 jtracert 之設計與開發。jtracert 是 MoPT 系統中所提供之網路拓樸的探索工具，可讓使用者任意指定兩個以上的網路節點，透過各種網路協定，找出網路節點之間的路由路徑。本工具可提供電信業者或網路管理者分析評估其網路拓樸與路由機制之弱點，並從而改善、提昇整體網路之安全度。本工具之特色包括：

- 圖形化使用者介面：可一覽無遺整個通訊網路中的弱鏈結與可能成為攻擊目標的受害節點。
- 快速追蹤法：藉由改善路由路徑追蹤方法可以比一般路徑追蹤法更快得到相關節點之間的路由資訊。
- 多協定支援：本工具支援多種網路協定，包括 ICMP、TCP 等。使用者可以依據當時的網路設定與狀況選擇適用的網路協定來收集路由資訊。
- 網路區域拓樸圖：本工具可提供單一路由路徑之圖形化顯示介面，亦可整併多筆路由路徑，建構出涵蓋所指定之多個來源/目的節點的區域拓樸圖。
- 拓樸分析：透過路由路徑的合併與內/外分支度的分析，本工具可找出該區域拓樸之中最可能成為攻擊目標的受害節點。

本工具的快速追蹤、多協定支援、網路區域拓樸圖與拓樸分析等都是新設計開發的方法與功能，這些都是傳統的 tracert (trace route 公用程式)中所尚未提供的。

■ **WBDT：WiMAX BS Denial of service (Dos) Testing System**

WBDT 為一個能分析 WiMAX BS 頻寬分配能力的工具。利用表 4-2 所列出的 WiMAX 頻寬權重分析，WBDT 可以計算出在不影響網路品質的狀況下，在 WiMAX 網路下的一個 BS 所能容納之 SS 個數 。以下我們將介紹如何使用 WBDT 檢測網路流量來瞭解 WiMAX 網路遭受 DoS 攻擊之風險。我們以使用 Skype 語音視訊為例，若欲使 Skype 語音、視訊通話順暢，所需之頻寬至少需為 100kbps。圖 5-1 為沒有遭受大量非正常封包干擾之 Skype 視訊截圖，此時封包流量約 200kbps，視訊畫面清晰。

圖 5-1、無 abnormal traffic 之 Skype 視訊畫面

根據先前的對於 WBDT 的介紹可以得知，10M 權限的 SS (用 SS_{10M} 表示) 所能得

到的頻寬為 5Mbps*0.7=3.5Mbps，並且所有的 SS_{10M} 會均分此 3.5Mbps，因此我們可以預期若是有 35 台 SS_{10M} 進入同一 WiMAX 網路，則每個 SS_{10M} 所能取得的頻寬將會小於 100kbps (如圖 5-2 所示)，這會造成 Skype 語音、視訊延遲的情況，大幅降低通話品質。圖 5-3 顯示了在頻寬小於 100kbps 的情況下的 Skype 視訊狀況，我們可以看到視訊品質非常不好。

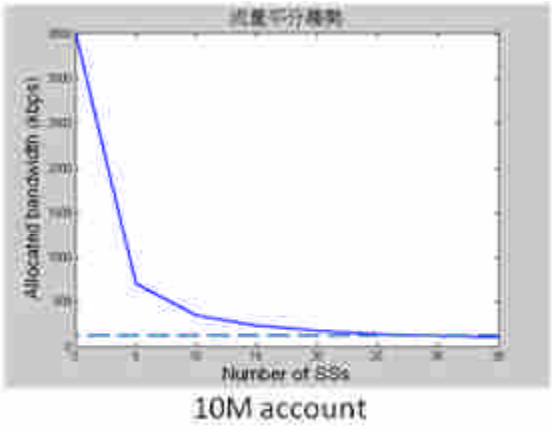


圖 5-2、SS_{10M} 的流量平分趨勢圖

圖 5-3、有 Abnormal traffic 的環境下，skype 的視訊畫面

若是以同樣的方式繼續分析具 4M 權限的 SS (用 SS_{4M} 表示) 與具 1M 權限的 SS (用 SS_{1M} 表示)。當 SS_{4M} 個數超過 10 台，SS_{1M} 個數超過 5 台，每個 SS 將獲得少於 100kbps 的頻寬，分別如圖 5-4、5-5 表示。若在此 WiMAX 網路下，產生 abnormal traffic 的 SS 個數持續增加，使得數量超過 WBDT 所分析的上限，對此網路的影響，輕則降低網路品質，重則可能癱瘓 WiMAX 網路。

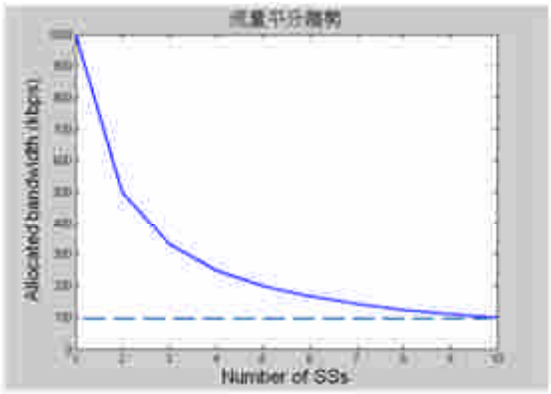


圖 5-4、SS_{10M} 的流量平分趨勢圖

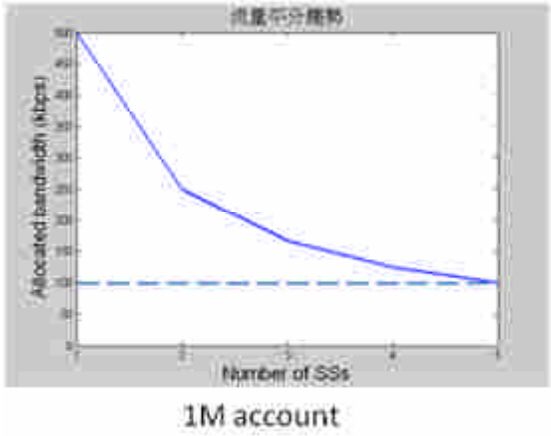


圖 5-5、SS_{10M} 的流量平分趨勢圖

■ **Wireless Penetration Testing System (WiPT)**

WiPT 是一套利用攻擊的方式來對無線網路進行滲透測試的工具。以下我們利用實驗分別顯示 WiPT 能夠成功破解以 WEP 加密的無線網路基地台以及偽裝無線網路基地台來得到使用者傳送的資訊。

傳統破解 WEP 方式是利用被動蒐集封包的方式，當初始向量蒐集到一定程度之後便會發生重複的現象進而發現 WEP 的加密規則。相反地，WiPT 採取主動的方式，它利用筆記型電腦的使用者身分不斷發出 ARP Request 的封包，迫使基地台回應 ARP Reply 封包，因此，WiPT 可以在短時間內蒐集到足夠的初始向量，瓦解利用 WEP 所設的密碼。通常利用這種方式破解 WEP 金鑰的時間不用超過三十分鐘。圖 5-6 顯示 WiPT 能夠成功地破解我們建置的一台以 WEP 加密的無線基地台(即 Test_AP)，得到該基地台所設立的密碼 “hello”。

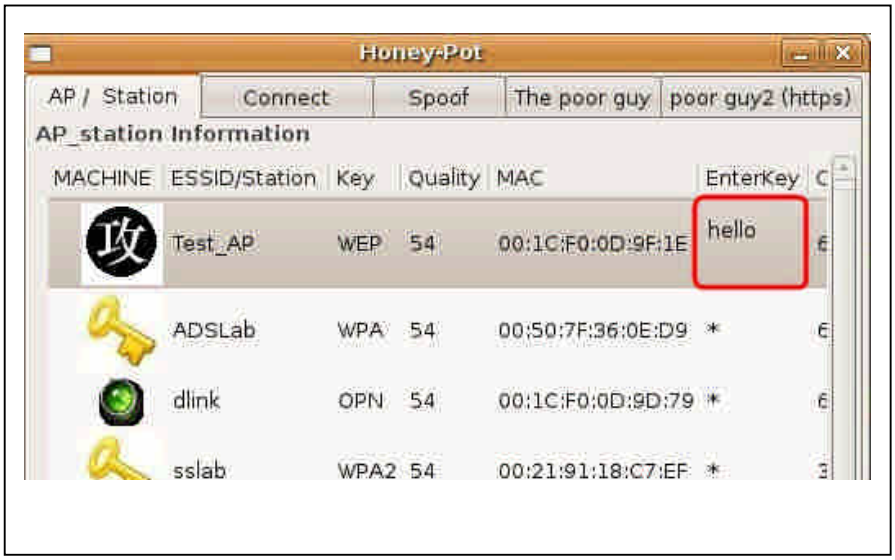


圖 5-6、基地台密碼破解

接著我們顯示 WiPT 能夠偽裝無線網路基地台來獲得使用者傳送的資訊。首先，WiPT 偽裝成一台無線網路基地台來吸引使用者連線，在該基地台的訊號範圍內的使用者可得知該基地台的存在。若使用者利用該基地台上網，他們所傳送的資料將被 WiPT 所截取並記錄，即使是經由 SSL 加密的 HTTPs 封包可會被 WiPT 得知。圖 5-7 顯示使用者登入的網頁是個利用 HTTPs 傳輸的網頁，該網頁有輸入身份證字號以及使用者代號的欄位。使用者填完這些資訊並按下"確認登入"按鍵的同時，我們所建置的偽造基地台將馬上得到該帳號密碼 (如圖 5-8 所示)，因此 HTTPs 的安全性將完全失效。

目前我們已經完成在無線網路平台上的建置，並且也確實證明麥當勞與中華電信合作的無線網路上網、Taipei WiFly 皆會遭受此類釣魚攻擊。同時我們也發現目前某些國內銀行業者也會因為此類攻擊，導致銀行用戶的重要資訊外洩，使得攻擊者可以使用受害者的帳號來進行各種行為，如銀行轉帳交易。

圖 5-7、使用者登入重要網路的畫面

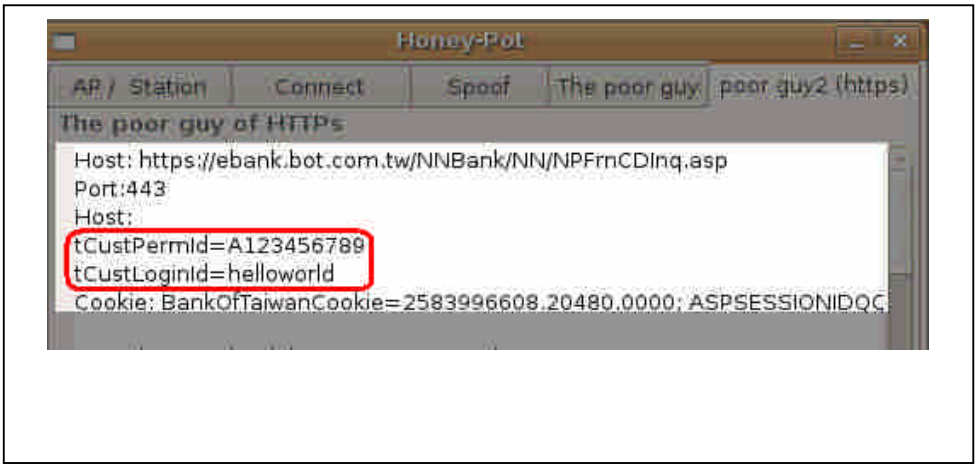


圖 5-8、偽造基地台蒐集到的資料

■ **Wireless Security Monitor (WiMon)**

WiMon 可即時監控並且偵測各種無線網路攻擊行為。以下我們將利用實際攻擊來展示 WiMon 的偵測能力。我們將許多無線網路裝置佈建於交通大學工程三館中(見圖 5-9)，並實際產生 Authentication flood attack 來測試 WiMon 的偵測能力。在 Authentication flood attack 中，攻擊者利用大量發送身分認證的要求給 AP，來填滿 AP 的關聯表，使其無法再提供建立連線的服務。

圖 5-10 顯示 WiMon 監控著交通大學工程三館六樓中的無線網路裝置，它藉由裝有 WiMon AP 搭配 GPS 定位系統，準確定出各無線裝置的實際位置。在圖 5-11 中，綠色虛線所連接的即是攻擊者電腦及受害的 AP。在攻擊者發出 Authentication flood attack 後，圖 5-12 中最左下角及上方的 WiMon AP 同時偵測出該攻擊行為，並成功找出其攻擊來源為綠色線條所連接的攻擊者電腦。此偵測結果送到安全中心進行進一步的分析。接著 WiMon 判定該攻擊為 Authentication/Deauthentication flood attck (見圖 5-13)，此結果與我們所發動的攻擊相同。接著，如圖 5-14 所示，安全中心將傳送相對應的防範措施到所有裝有 WiMon 的行動裝置上。當所有 WiMon APs 接收到新的防範規則後，該範圍內的行動裝置將不會再遭受到相同類型的攻擊。

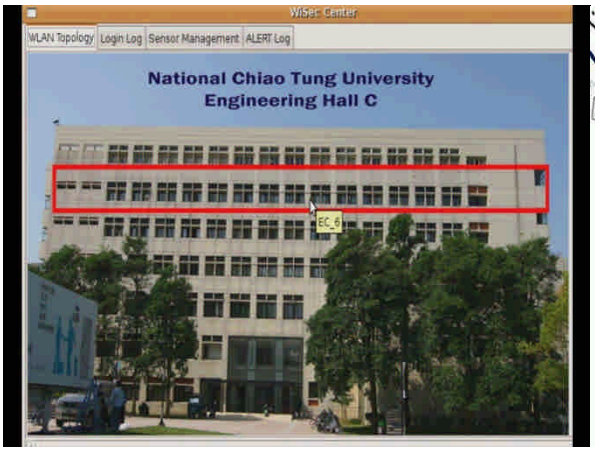


圖 5-9、交通大學工程三館

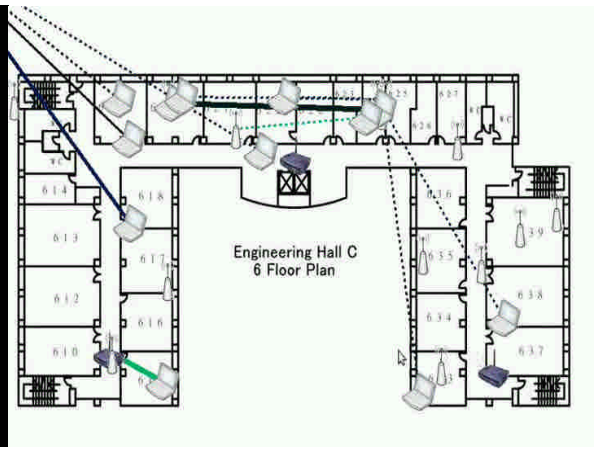


圖 5-10、交通大學工程三館 6F 平面圖

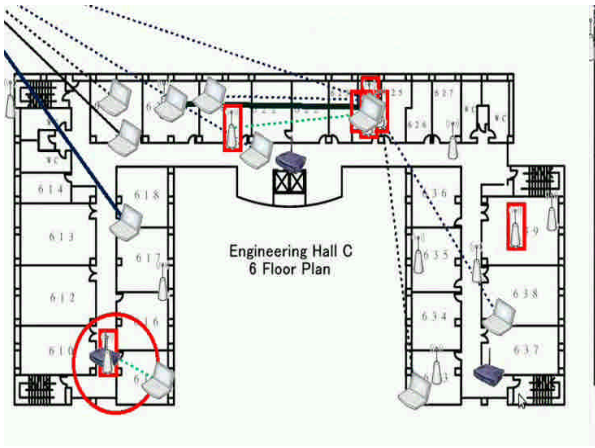


圖 5-11、攻擊示意圖

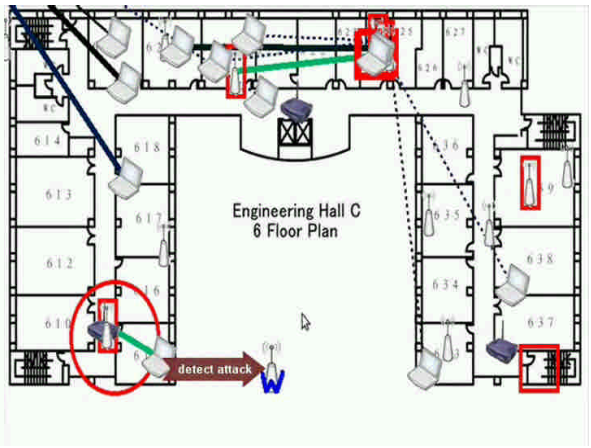


圖 5-12、WiMon APs 偵測到攻擊

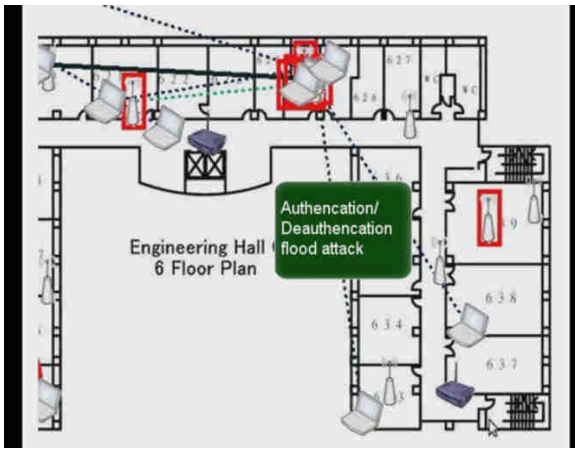


圖 5-13、成功分析攻擊類型

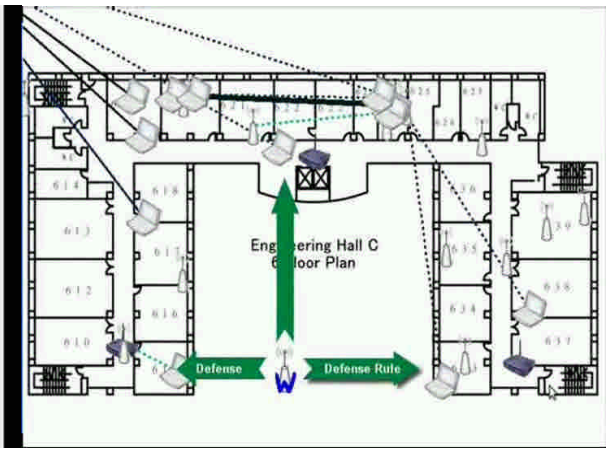


圖 5-14、傳送對應的防範規則到 WiMon APs

■ **入侵偵測系統強度評估系統 (Simple IDS Informer，簡稱 SIDSI)**

SIDSI 是一套可檢測、評估入侵偵測系統強度的系統。以下我們將透過實驗來進行 SIDSI 的效能評估。攻擊測試通常可依據攻擊類型 (Attack types) 和攻擊路徑 (Attack paths) 進行分類。以下我們列出 3 種攻擊類型和 8 種攻擊路徑來進行 SIDSI 效能評估的實驗。

一 攻擊類型

1) Shellcode
2) PostScan
3) Flooding

一 攻擊路徑

對一台具有入侵偵測系統 (IDS) 之無線網路裝置而言，其周遭網路可分為 LAN、WAN 與 WLAN 三個象限。若欲測試一台受測機器 (Devise under test，簡稱 DUT)，由上述三個象限可衍伸出下列八條攻擊路徑：

1) WLAN – WAN
2) DUT – LAN
3) DUT – WLAN
4) DUT – WAN
5) LAN – LAN
6) LAN – WLAN
7) LAN – WAN
8) WLAN – WLAN

經過我們的實驗，一般的無線 DUT 在設計上經常在以下路徑上有安全性的漏洞，這些路徑分別是 LAN – LAN、LAN – WLAN 以及 WLAN – WLAN (以下我們用 critical paths 表示這三條路徑)。圖 5-15 顯示此實驗之架構，其中紅線表示 critical paths。SIDSI 支援 flowbit 之功能，因此它可以比對跨封包之 SNOT 規則，如圖 5-16 所示。

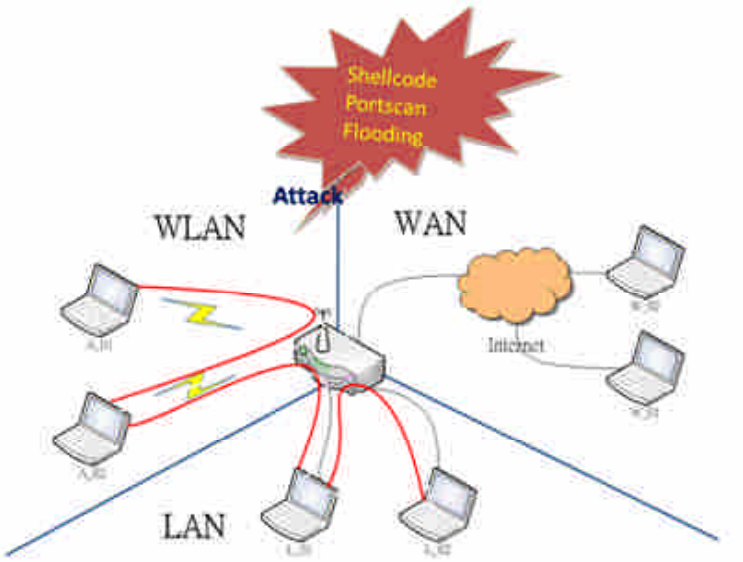


圖 5-15、SIDSI 實驗架構圖

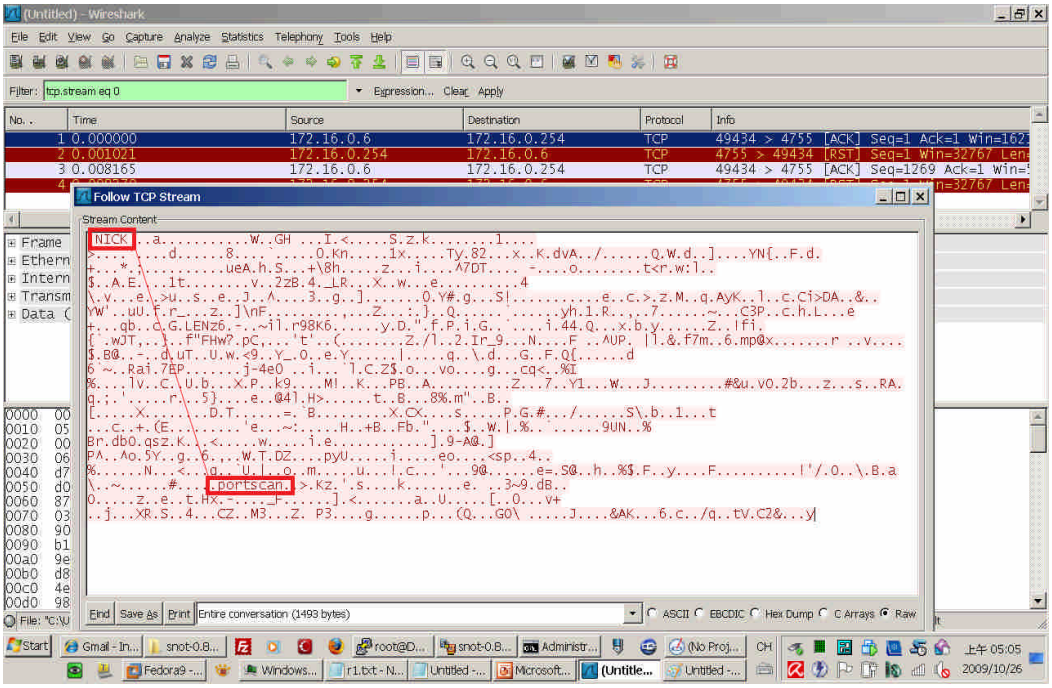


圖 5-16、SIDSI 支援跨封包比對 SNOT 規則

圖 5-17 顯示未切割之一般封包含有違規存取字串 get /admin-serv/config/admpw (以紅框標示)，經實驗測試後，該 DUT 可正確地偵測該封包有越權存取的危險性。接著我們再透過 SIDSI 送出切割後的封包，將違規字串拆散在數個封包內發送，藉以測試 DUT 之進階防禦能力。

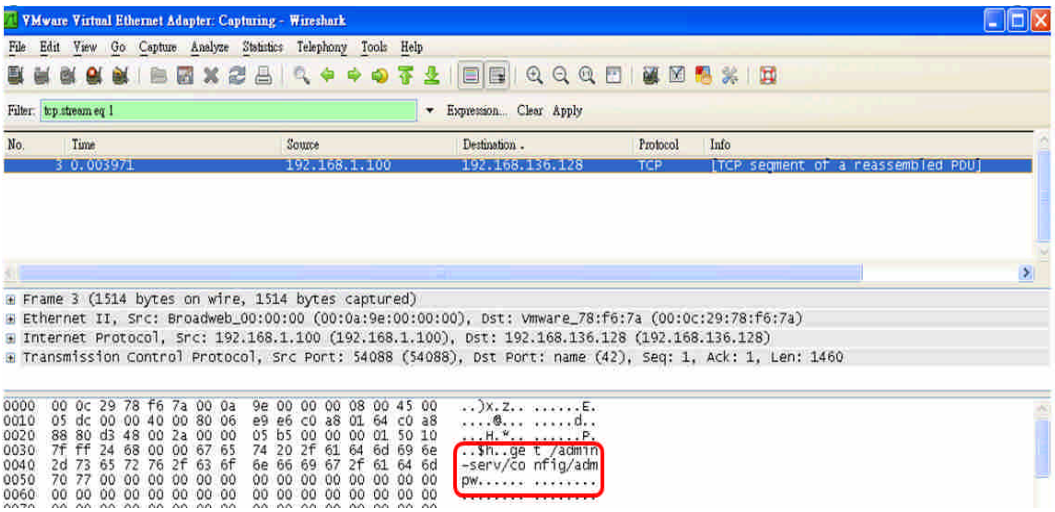


圖 5-17、含有違規存取字串之未分割封包範例

圖 5-18 顯示經由 SIDSI 分割並送出的封包，我們透過 SIDSI 的 MakePacket_Divided()函式指定封包字串長度為 3。因此，分割封包中的第一個封包的

字串內容為 get，下一個封包 payload 為/ad。

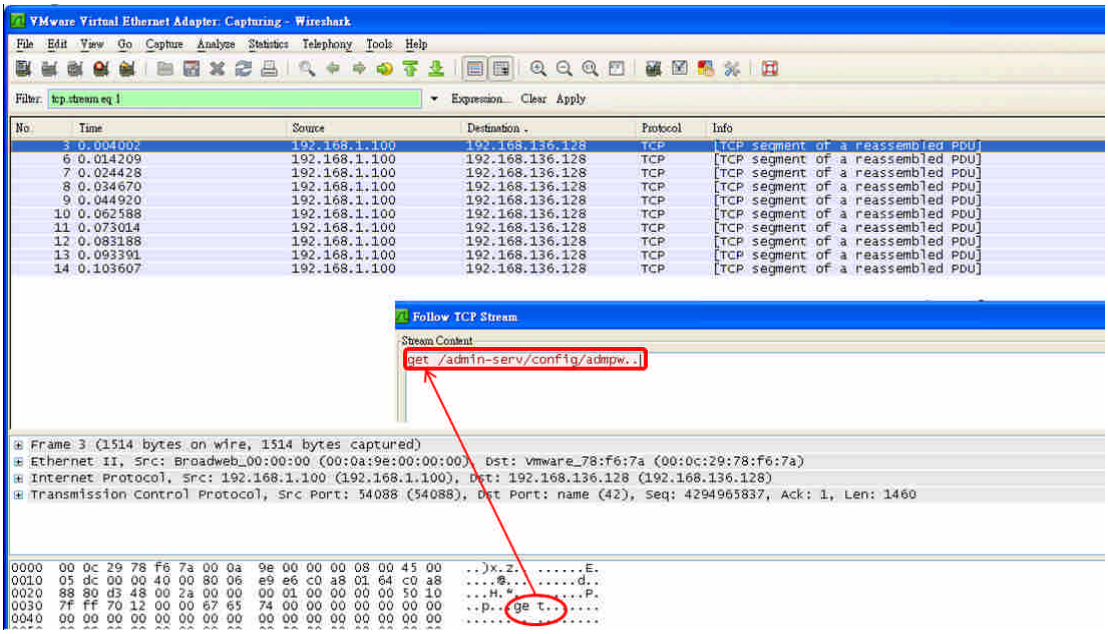


圖 5-18、SIDSI 支援分割封包之功能

以上實驗說明了 SIDSI 之實用性，SIDSI 除了提供基本的發送攻擊封包之測試功能外，它亦支援分割封包等進階功能，可用多種方式來測試 DUT 之強度與防禦能力，進而提供 DUT 強度改進的建議。

● 行動裝置滲透檢測

■ Android 行動裝置惡意網頁檢測工具

以下我們利用實驗來說明 Android 行動裝置惡意網頁檢測工具能夠攔截具有惡意攻擊的語法的網頁。在使用者瀏覽網頁時，第一個動作便是下載該網頁的資料。在下載網頁資料的同時，Android 行動裝置惡意網頁檢測工具會將該網頁資料中有安全疑慮的 javascript 進行 overlay 的動作。

圖 5-19 顯示使用者對於 google 進行連線要求。當下載完 google 網頁後，Android 行動裝置惡意網頁檢測工具也完成 javascript 的轉換並且顯示出 ”DEBUG: Finish Loading!”。

圖 5-19、下載網頁

接著 Android 行動裝置惡意網頁檢測工具將利用建立好的 policy 來判定轉換完的 javascript 函數是否具有威脅性。若是通過 policy 判定，瀏覽器將呈現 google 的網頁(見圖 5-20)。

圖 5-20、網頁未發現威脅

接著我們展示 Android 行動裝置惡意網頁檢測工具如何偵測出具有惡意程式碼的網頁。首先，我們先建立一個網頁，其網址為 http://140.113.216.158。該網頁被駭客惡意植入了 cross-site attack 攻擊的程式碼，因此所有連上該網頁的使用者的 Cookie 都會被輕易地被駭客盜用。同樣地，Android 行動裝置惡意網頁檢測工具先連線到該網頁，修改具有安全性問題的 javascript 語句 (如圖 5-21 所示)。接著 Android 行動裝置惡意網頁檢測工具發現該網頁含有惡意攻擊的語法，它會攔截該攻擊並且顯示如圖 5-22 所示的訊息。

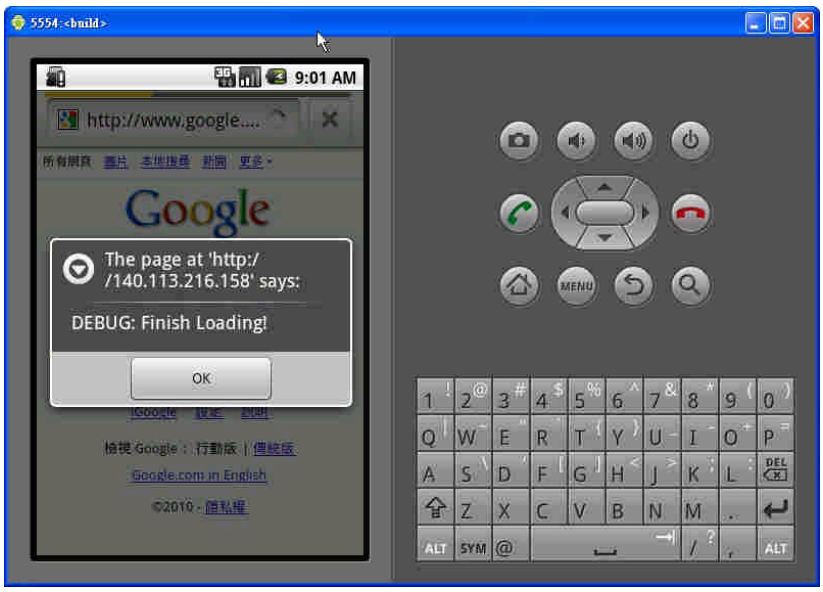


圖 5-21、惡意網頁下載

圖 5-22、成功偵測到 cross-site attack

■ **Android 應用軟體安全漏洞檢測工具 (G-exploit)**

G-exploit 旨在檢測 Android 上應用程式之漏洞，幫助程式開發人員彌補程式開發中不經意發生的錯誤或是修補隱藏的安全性漏洞。由於 java 本身就是一個以安全為優先考量的程式開發語言，因此常見的錯誤都是寫作壞習慣或是寫作規則的錯誤，較少發現可能造成重大傷害的安全性漏洞，但是仍隨著 Android 之普遍和日趨強大的功能，Android 之應用軟體安全也益發重要。

目前 G-exploit 可以檢測到的 java 問題包含:

1) 正確性的問題(correctness)：檢測程式設計人員沒注意到之寫作上的錯誤，避免此類寫作錯誤成為真實程式的臭蟲 (bug)。

2) 寫作壞習慣(bad practice)：檢測程式設計人員寫作時的壞習慣，例如解構子的濫用，避免不必要的錯誤。

3) 異常的程式碼(dodgy)：檢測不確定的程式行為，如未被確認的轉型等等。

我們以 Snake 這個 Android 應用程式為例，介紹 G-exploit 之操作流程並驗證 G-exploit 之可行性。首先，我們將 Snake.dex 檔上傳至 G-exploit 網頁平台，並選擇使用 Findbugs 軟體來進行靜態偵測 (見圖 5-23)。

圖 5-23、G-exploit 線上檢測平台

接著，G-exploit 將呈現檢測報告，如圖 5-24 所示。由檢測報告可以看出，此應用程式大小為 689 kbs，含有 75 個臭蟲。此檢測報告亦會提供安全漏洞的危險等級，其中 Bugs p1 之危險等級最高，Bugs p2 其次，依此類推。而報告中的 Ratio 則表示安全性漏洞的存在比例。此次實驗偵測到的安全性漏洞為 MS (見圖 5-25)，這表示在 Snake 程式中存在一塊可被惡意程式或是經由人為的不小心而被更動的靜態區塊；MS 安全漏洞將可能造成 Snake 程式被惡意程式所控制而執行非預期之惡意行為，其解決辦法為將 Snake 程式封裝成套裝軟體 (package)。

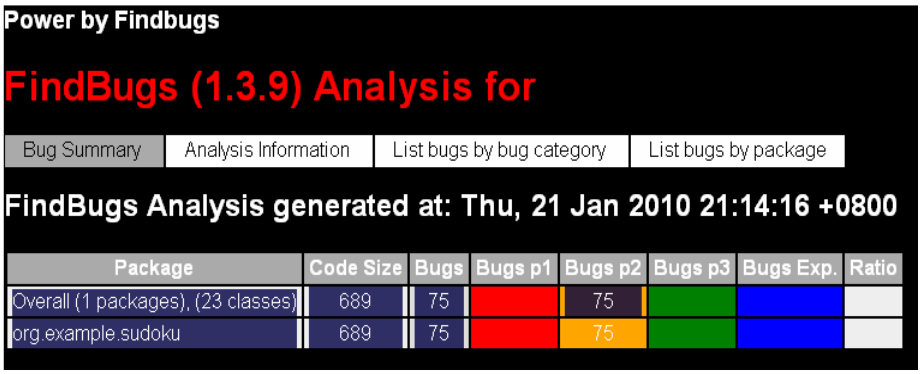


圖 5-24、G-exploit 之檢測報告

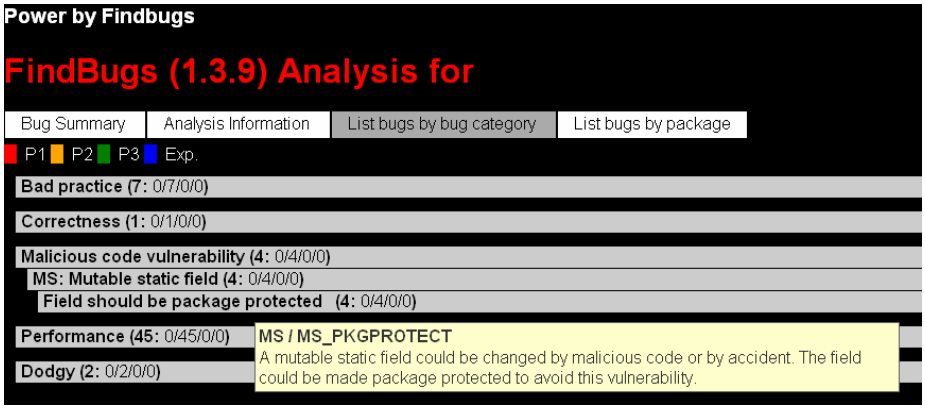


圖 5-25、檢測結果之詳細說明

■ **Android 動態惡意軟體程式檢測工具**

藉由 QEMU Translator 的幫助，我們目前已經可以成功將 Android 建置在 x86 主機模擬系統的上層，並以此進行觀測。圖 5-24 是 Android 架設在 x86 系統模擬流程。首先，利用其 VM Dalvik 將 Android 的應用程式編譯成屬於簡單指令集的 ARM 指令。接著，我們利用 QEMU Translator，將 ARM 指令轉換成複雜指令集的 x86 指令集。由於 ARM 指令已經被轉至成 x86 指令，所以它可以架設在 x86 主機模擬系統(QEMU)。因為經由 Dalvik 所編譯出來的 ARM 指令再轉成 x86 兩段手續後將會造成指令無法確實對應，因此我們還無法進行惡意軟體分析與判斷，我們需要進一步研究來解決這個問題。

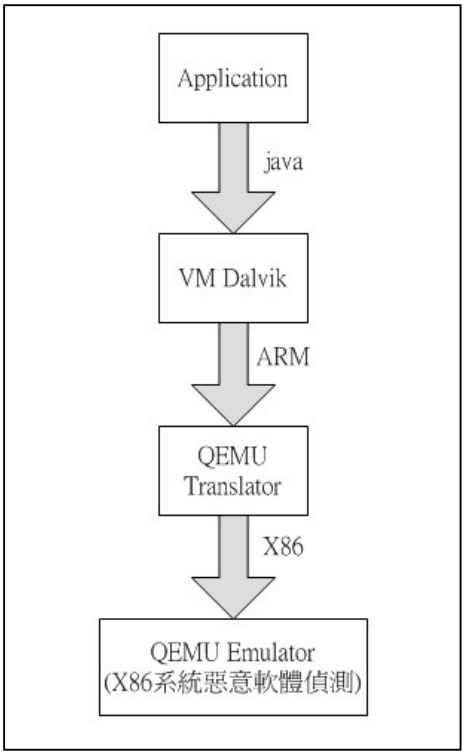


圖 5-26、Android 架設在 x86 系統上的模擬流程

圖 5-25 是我們接下來預計完成的流程圖。在未來兩年，我們將繼續研究 ARM 指令的操作碼 (OP-code)，了解每個指令是要做什麼事情，並且了解在這操作碼之後所接續的暫存器與與我們在 x86 系統動態惡意軟體程式所作記錄的暫存器相對應關係。得知這些對應關係之後，我們就可以將這些 ARM 指令轉換成 x86 指令，然後就可利如同 x86 系統動態惡意軟體行為偵測系統來進行 Android 平台上惡意軟體的檢測。

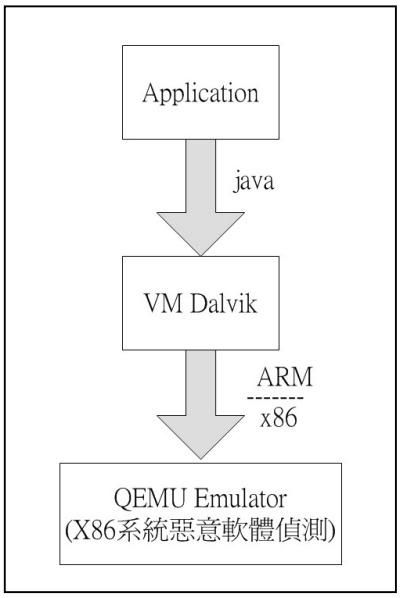


圖 5-27、預計工作流程圖

- **文件檔案惡意程式檢測系統 (Forenser)**

Forenser 為一自動化惡意程式檢測系統，它可偵測文件檔案中的惡意程式碼。以下我們將介紹 Forenser 針對非 Windows 執行檔所做的檢測流程。圖 5-28 顯示 Forenser 針對非 Windows 執行檔做檢測的流程。Forenser 會依序將掃描的過程及結果一一呈現給使用者(見圖 5-29)。當檢測完成時會會提供檢測報告並提供操作上的建議 (見圖 5-30)，檢測報告結果亦可用右上方的第三個按鍵儲存。

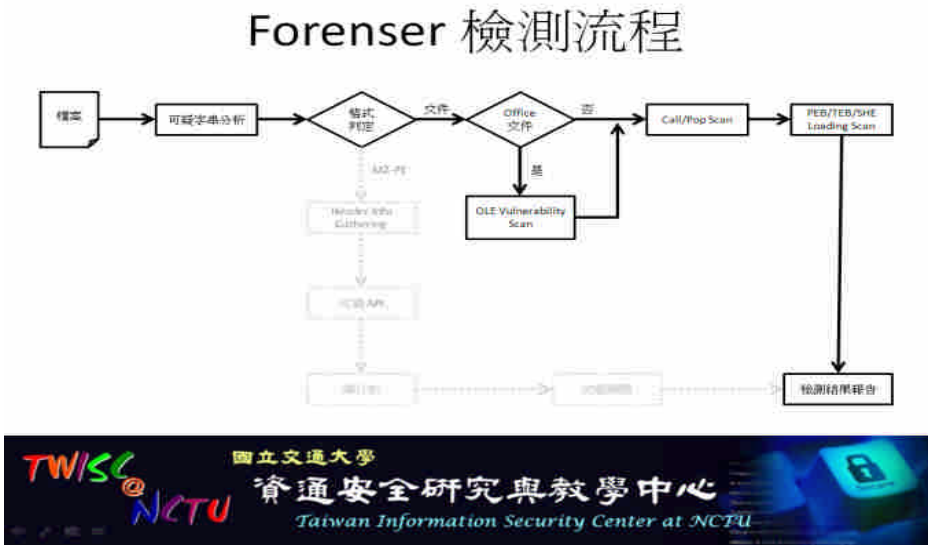


圖 5-28、針對非 Windows 檔案檢測流程

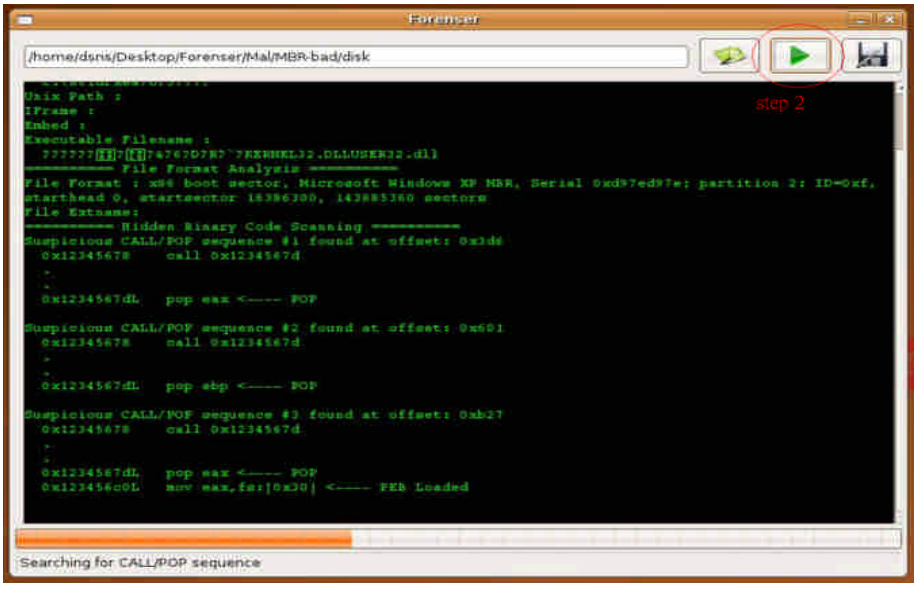


圖 5-29、檢測中提供詳細的資訊給使用者

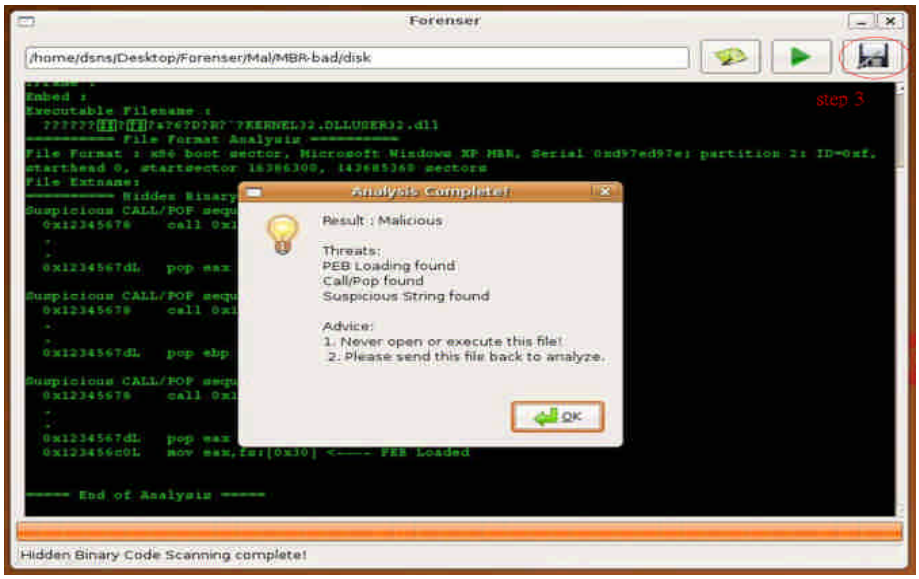


圖 5-30、檢測完成並給予使用建議

■ **動態惡意軟體分析檢測工具 (Malware Behavior Analyzer, MBA@TWISC)**

為了追蹤 X86 主機的資訊流動，我們對 X86 指令集中的每個指令進行分析，了解每個指令的汙染擴散行為。例如 mov op1, op2 這組 X86 指令中，假若 op2 是我們認定的汙染資料，藉由 mov 的搬動，op1 也隨之受到汙染，又或者 jmp [mem]會將記憶體位址 mem 汙染擴散到 CPU 的程式計數器(Program Counter)…等。在確認為需要記錄為可疑的指令後，我們將搭配接下來將會提到的”汙染資料記錄資料結構設計”做記錄。

為了提升資料擷取的效率以及減少儲存汙染資料所造成龐大的空間負擔，我們為系統設計一了個資料結構以利後續分析。X86 指令間彼此互動頻繁，在一連串的指令傳遞過程中，很有可能都是受到同一分汙染資料的干預因而造成汙染資料檔必須記錄所有指令以及暫存器的內容，造成其中許多不相關的資料也跟著被記錄。依照這種方式記錄會造成資料紀錄檔過於龐大，不僅浪費記錄資料的空間，這種沒有效率的紀錄會照成往後提取資料困難而嚴重影響分析汙染資料的效能。

如果想要模擬一 512MB 的 X86 主機運作，所需要花費的記憶體就達 2.5GB 之多。為了解決這個問題，我們透過如記憶體管理單元將記憶體分層的概念，以避免未受汙染的區塊也同樣占據記憶體空間。如以記憶體前 24bit 作為索引，便可節省未受到任何汙染的 256 個記憶體區所需的汙染資料結構。如圖 5-31 所示：

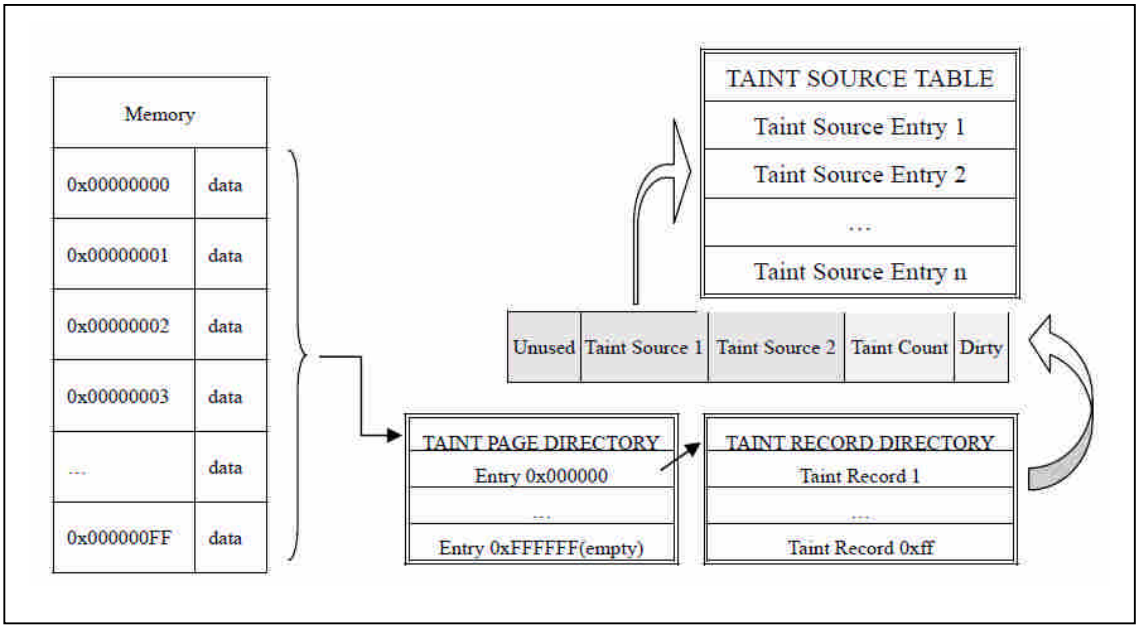


圖 5-31、利用記憶體管理方式改良記錄資料結構

汙染記錄資料結構設計完成後，接下來要為 X86 系統模擬器(QEMU)的每個模擬儲存單元作修改，所謂的模擬儲存單元包括：CPU 的暫存器、記憶體、硬碟、鍵盤輸入緩衝區、顯示卡的顯示緩衝區以及網路卡的傳送/接收緩衝區等等。每個模擬儲存單元都有不同的儲存格式，所以在資料結構上也為了各個單元做了調整，讓資料記錄更有彈性，資料分析也更有效率。

本系統已經完成資訊流動的追蹤，但是卻沒有設定資訊安全的功能。本系統提供了技術人員可以自行標定特定儲存單元或是特定資料當作汙染源的標的，以進行分析。此功能僅提供基本規則的定義方式，不足以定義和作業系統相關的資訊安全規則。由於模擬與分析工作可以彼此獨立進行，因此本系統利用管線化處理的概念，提高整體模擬運行的 throughput。我們將整個系統的工作切割成三段獨立的子單元，如圖 5-32 所示。藉由(A)模擬 CPU 運行指令、(B)依該指令之行為更新汙染記錄資料結構以及(C)檢查是否違反定義的資訊流程安全規則的獨立以及管線化處理，達成提升整體分析效能。

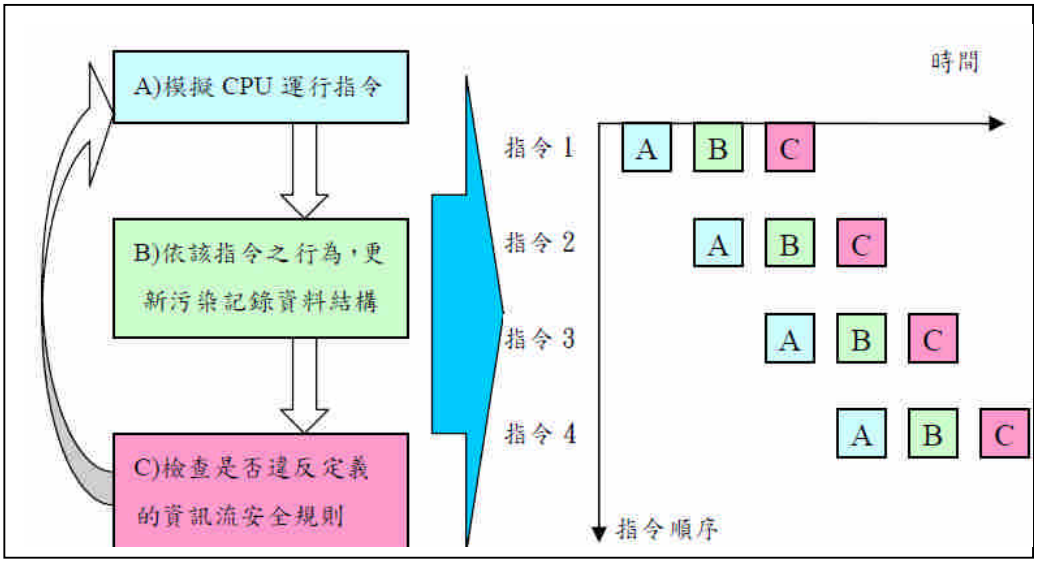


圖 5-32、管線化設計

■ **大規模遠端系統滲透測試網 (RSPTN)**

本計畫已經於 98 年度完成大規模遠端系統滲透測試網之初步建置，以下將以一範例實驗說明 RSPTN 之使用方式與支援之功能。此滲透測試網之系統規格與實驗受測電腦規格如下：

— 滲透測試網伺服器

CPU：Intel Core 2 Duo E6300

Ram：DDR II-667 1G*2

OS：LINUX 2.6.29.4

— 受測電腦

CPU：Intel Pentium-M 1.5 GHz

Ram：768 MB

OS：Windows XP Pro SP3

當受測電腦連上 RSPTN 後，若是同意 RSPTN 進行滲透測試，則 RSPTN 便可開始弱點偵測與分析。以此次測試實驗為例，經過約 90 秒，RSPTN 可產生滲透測試報告，此報告正確地分析受測電腦之作業系統 (Windows XP)，且 RSPTN 可偵測出 87 個弱點，並能成功滲透其中之一 (見圖 5-33)。

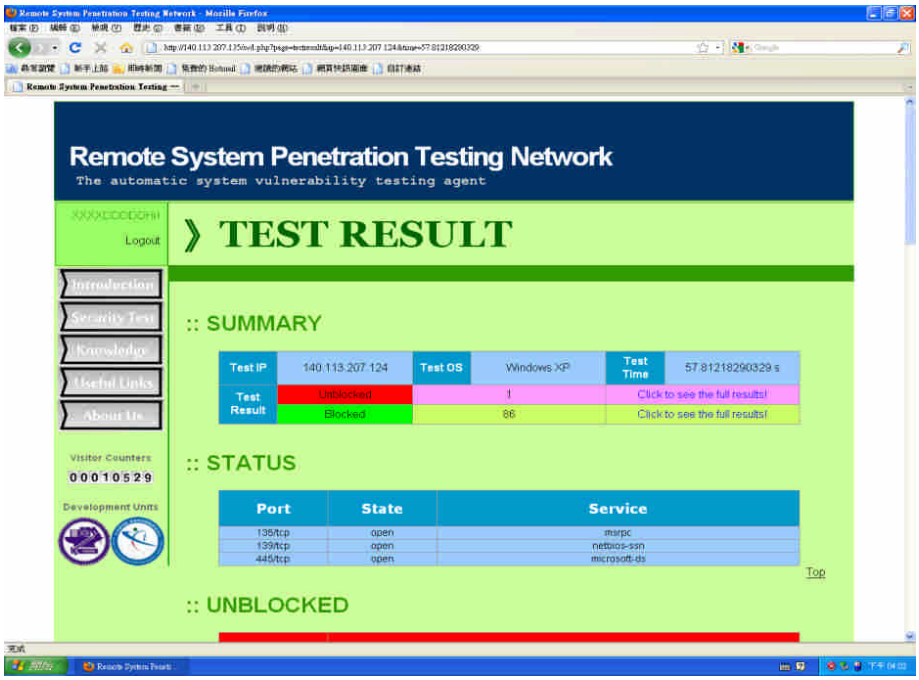


圖 5-33、RSPTN 之滲透測試範例實驗

■ **使用者敲鍵行為辨識系統**

使用者敲鍵行為辨識系統利用使用者的敲鍵行為來進行使用者身分的識別。為了有效評估系統效能，我們首先建構了一個網頁進行樣本的收集，該網頁是由 JavaScript 所寫成，採取樣本的時間單位為微秒。收集對象有 53 位志願者，這些志願者每天提供 10~20 個樣本，樣本採取為期兩個月，同時我們也請 103 位匿名使用者試著去輸入他們的帳號密碼，用來測試系統對非法使用者的拒絕率，每個帳號會遭到 50~200 次的攻擊，總共為 3126 次。此外，我們應用 AR Model 時則分別取 order 1 至 5 進行比較，AR Model 的參數則由 Burg's Algorithm 算出。最終我們會列出各個 EER (equal error rate)進行比較。我們將 EER 的比較結果列於表 5-1。Digraph 以及 Trigraph 則分別為不同的馬可夫練的建立方式，我們可發現在 AR model = 1 的時候，系統的 EER 可降低至 2.19%。

表 5-1、COMPARATIVE RESULTS OF EER IN EXPERIMENT

	Analysis with Digraph	Analysis with Trigraph
AR(1)	2.19%	2.93%
AR(2)	2.37%	2.81%
AR(3)	2.37%	2.68%
AR(4)	2.49%	3.08%
AR(5)	2.64%	3.08%

我們亦分析將生物行為改變趨勢作為生物特徵所增進的效益，發現有超過一半的

使用者的準確率上升 (見表 5-2)。而平均增進效益列於表 5-3。

表 5-2、THE RATIOS OF USERS HAVING IMPROVED EER

	Analysis with Digraph	Analysis with Trigraph
AR(1)	41.18%	50.00%
AR(2)	50.00%	55.88%
AR(3)	44.12%	44.12%
AR(4)	55.88%	52.94%
AR(5)	52.94%	55.88%
Average	***48.82%***	***51.76%***

表 5-3、THE AVERAGE PROMOTION OF EER WITH DIFFERENT ORDER OF AR MODEL IN THE EXPERIMENT

	Analysis with Digraph	Analysis with Trigraph
AR(1)	6.21%	5.79%
AR(2)	5.17%	5.69%
AR(3)	5.84%	5.49%
AR(4)	5.73%	6.76%
AR(5)	4.27%	6.64%
Average	***5.44%***	***6.07%***

圖 5-34 列出我們的系統與其他各研究的比較。在我們的實驗結果中可以發現本系統產生的 EER 範圍在 2.19%~3.08%，就目前而言要比其他研究要來的準確，之前研究出現最佳的 EER 是 2.54%，這些研究大多未將生物行為改變列入考慮。

雖然在本次實驗當中我們只增進了大約 50%使用者的準確率，但如果我們進行更長時間的樣本採取，生物行為改變將更為明顯，也會明顯提升辨識的準確率，而本次實驗針也對 digraph 以及 trigraph 做比較，發現大多時候 digraph 的表現要比 trigraph 要的好，固本系統採取的便是 digraph。

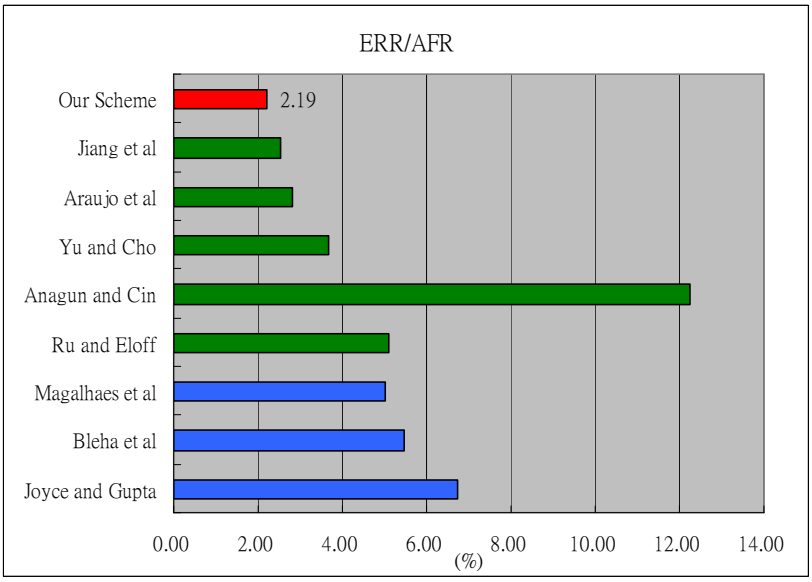


圖 5-34、EPR/AFR 比較

綜合以上所述，我們可以發現利用生物行為易改變的模式進行辨識，才是有效辨識的方法。我們結合高斯模型、自回歸模型、馬可夫鍊模型以及數學統計方法，在每次使用者合法登入時微調舊模型，使之一直可以維持良好的準確率。

■ **實驗平台 - Secure Wireless Overlay Observation Network (SWOON)**

以下將介紹二項實驗：無線竊聽攻擊實驗與分散式阻絕服務 (Distributed Denial of Service，簡稱 DDoS)攻擊實驗。此二項實驗可說明 SWOON 的可行性和應用性。

– 無線竊聽攻擊：

相較於有線網路，無線網路以空氣為傳送介質的特性使其更易於遭受攻擊，竊聽攻擊是無線網路中常見的攻擊之一。SWOON 開發團隊實作了 WiMAX 的虛擬驅動程式，該驅動程式完全符合 IEEE 802.16 標準，因此本無線竊聽攻擊實驗除了呈現 SWOON 對於網路攻防實驗的支援度之外，也可檢驗 WiMAX 虛擬驅動程式之正確性與 SWOON 監控功能的有效性。此外，SWOON 開發團隊亦修改了 Wireshark 網路封包檢視軟體，使其支援 WiMAX 封包的格式判斷，增加 SWOON 的應用性。圖 5-35 呈現此竊聽實驗擷取的 WiMAX 封包訊息。

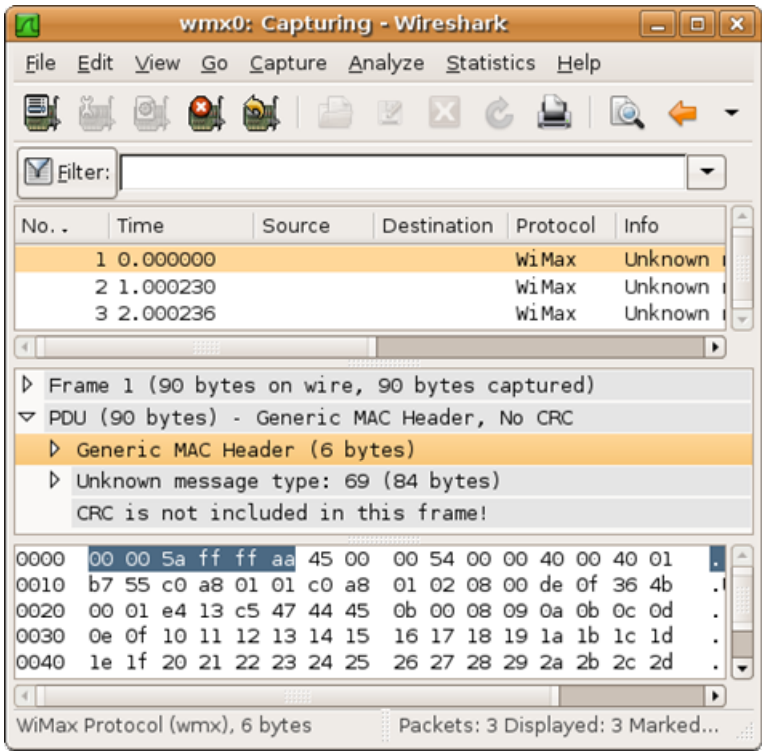


圖 5-35、以 WiMAX 為例之無線竊聽攻擊實驗

– DDoS 攻擊實驗：

DDoS 攻擊實驗除了能測試 SWOON 對於網路攻防實驗之支援度之外，亦可用於測試 SWOON 之穩定性。圖 5-36 為利用 SWOON 架設的模擬網路環境，在此環境中，攻擊者 (Attacker)透過其他機器 (Zombie1、Zombie2)對受害者 (Victim)進行的 DDoS 攻擊；在 DDoS 攻擊之下，受害者電腦的 CPU 使用率與網路接收封包量將劇增。SWOON 的 DDoS 偵測模組則在 DDoS 攻擊啟動後偵測到此一攻擊行為，當封包數量劇增超過定義的臨界值時，如圖 5-37 所示，偵測模組則會發出紅色警示。如果攻擊者停止攻擊，則 SWOON 系統中的 DDoS 偵測模組將會得到新的系統資訊。明顯地，受害端的系統 CPU 使用率將回到正常狀態(見圖 5-38)。

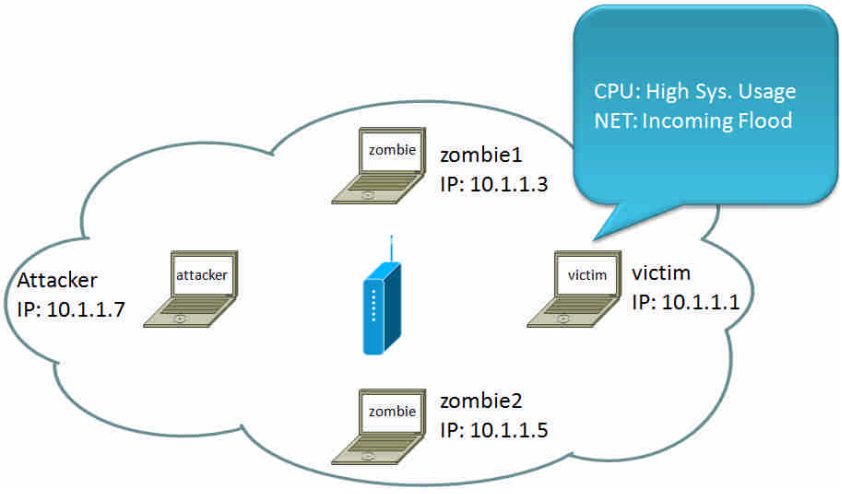


圖 5-36、模擬網路環境

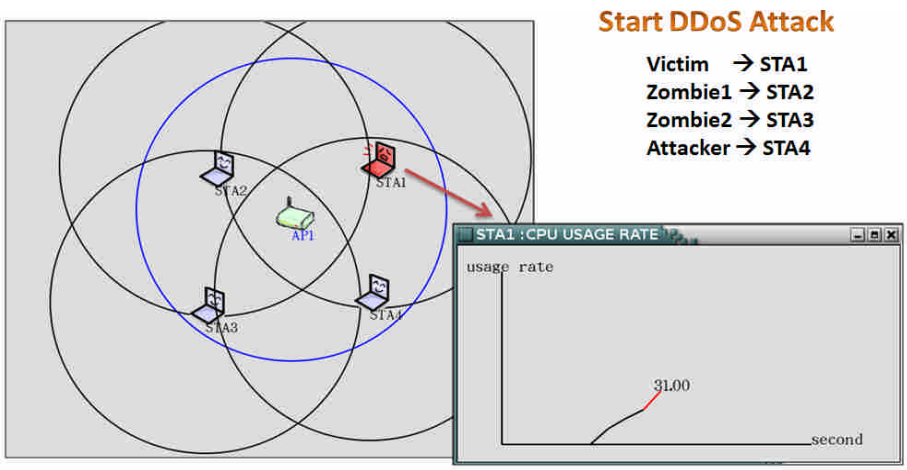


圖 5-37、SWOON 偵測到 DDoS 攻擊

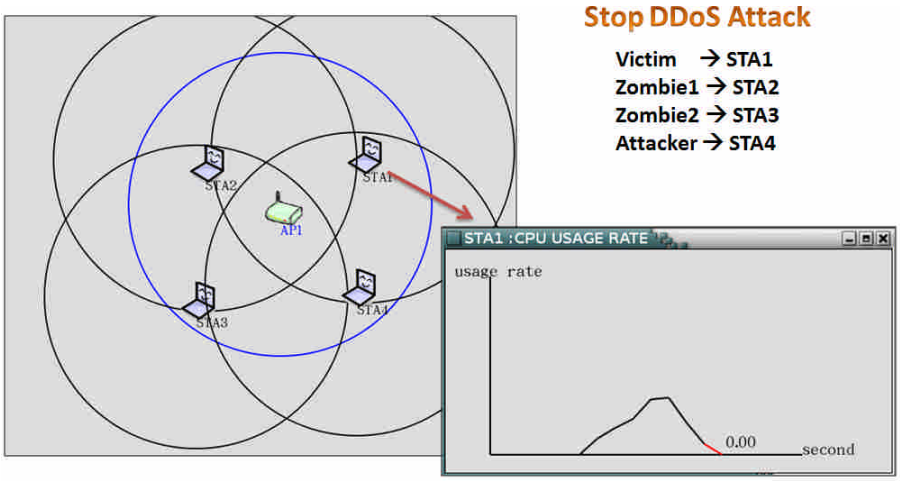


圖 5-38、DDoS 攻擊停止

除了以上二個攻擊實驗，表 5-4 亦條列了其他 SWOON 所能支援的網路攻擊實驗，並且比較與 DETER 之間的差異。以駕駛攻擊 (war driving) 為例，該實驗需能夠仿真可搜尋無線網路且蒐集封包之行動代理人 (mobile agent)，然而 DETER 是專為有線網路設計的安全實驗平台，無法支援專屬於無線網路的駕駛攻擊。而 SWOON 的「虛擬天線 – 虛擬驅動程式」之設計能仿真模擬無線訊號之衰減，並且 SWOON 監測模組能解析收到的無線封包，因此 SWOON 能有效地支援駕駛攻擊實驗。

表 5-4、可仿真的網路攻擊實驗

網路攻擊	DETER	SWOON

駕駛攻擊 (War driving)	No	Yes
MAC 欺騙 (MAC spoofing)	No	Yes
IP 欺騙 (IP spoofing)	Yes	Yes
有線竊聽 (Wired eavesdropping)	Yes	Yes
無線竊聽 (Wireless eavesdropping)	No	Yes
中間人攻擊 (Man-in-the-Middle)	Yes	Yes
邪惡雙生 (Evil Twin)	No	Yes
分散式阻絕服務 (DDoS)	Yes	Yes

六、 整體計畫成果

本計畫主要為建置異質無線多網安全檢測平台。此平台滲透與檢測之對象可分為兩大部份: 第一部份為異質多核心網路 (Heterogeneous Multiple Core Networks)，包括 3G、WiMAX、Wi-Fi 與有線網路；另一部份為行動設備 (Mobile Devices)，包括 NB、netbook、智慧型手機，而這些行動設備可為 Windows、Linux 或者 Android 平台。以下將針對計畫績效、學術成就與專利三方面來說明計畫成果。

● 計畫績效

表 6-1 為計畫績效 KPI 總表，本計畫在 2009 年 (即今年度) 建置與開發共 13 個子系統與工具、建置 3 種智慧型手機平台、提供 5 項檢測服務 (請見表 6-2)、在技術移轉、技術服務與產學合作方面已簽約總金額達 1387.5 萬，其中包括技術移轉 5 件 (請見表 6-3)、技術服務 4 件 (請見表 6-4)以及產學合作 6 件 (請見表 6-5)，以上的產學合作和軟體授權執行期間由 2009 年跨年執行到 2010 年。

表 6-1、計畫績效 KPI 總表

	2009 (本年度成果)	2010 (預期成果)	2011 (預期成果)
建置異質無線多網安全測試平台	1	1	1
開發工具與系統	13	13	13
建置智慧型手機平台(Linux, Windows, Android)	3	3	3
建置異質網路平台(WiFi, WiMAX, 3.5G	3 種網路	3 種網路	3 種網路
檢測服務	5 項	5 項	5 項
技術移轉、技術服務與產學合作總金	1387.5 萬*	1700 萬	2000 萬

額	(技術移轉 5 件、技術服務 4 件、產學合作 6 件)		

*上表為累計數量

表 6-2、檢測服務清單

單位	服務名稱
資訊工業策進會	無線網路漫遊認證交換中心安全檢測
台北市電腦商業同業公會	線上遊戲安全檢測(戲谷、新幹線等網路遊戲業者)並訂定遊戲業者資安檢測標準
工業技術研究院	WiMAX 科學園區建置安全檢測
友訊科技 D-Link	委託 Open D-Link Routers Forum 建置、維護與測試技術與諮詢服務
工業技術研究院	MACsec 金鑰管理方法及其環境效能分析

註: 上表中檢測服務項目取自技術服務與產學合作中具有檢測工作者

表 6-3、技術移轉(屬於軟體授權類別)項目及簽訂金額

項目名稱	對象	2009 已簽訂金額(萬)
程式檔案與文件檔案安全鑑識器	國安局	50
無線網路安全弱點監控與檢測平台	國安局	75
嵌入式平台安全檢測雛形系統	工業技術研究院	30
自動化惡意程式檢測系統	中華電信	85
基於虛擬機器之惡意程式行為分析軟體系統	國安局	50
總金額		**290**

表 6-4、技術服務項目及簽訂金額

項目名稱	對象	2009 已簽訂金額(萬)
MACsec 金鑰管理方法及其環境效能分析	工業技術研究院	20
線上遊戲安全檢測	台北市電腦商業同	100

	業公會	
無線網路漫遊認證交換中心安全檢測	資訊工業策進會	35
資訊系統儲存媒體加密機制方案規劃與評估	國安局	65
Wi-Fi/WiMAX Roaming 機制研究與效能分析 (含 WiMAX 科學園區建置安全檢測)	工業技術研究院	本計畫含產學合作與安全檢測,並已列舉在產學合作表中
總金額		**220**

表 6-5、產學合作項目及簽訂金額

項目名稱	對象	2009 已簽訂金額(萬)
新型網路攻擊、風險評估與入侵追蹤的誘捕預警技術	中科院	367.5
Relay Selection for Wireless HDMI Cooperative Systems	聯發科技	55
Wi-Fi/WiMAX Roaming 機制研究與效能分析 (含 WiMAX 科學園區建置安全檢測)	工業技術研究院	50
資訊產品安全檢測技術整合型研究	中科院	255
委託 Open D-Link Routers Forum 建置、維護與測試技術與諮詢服務	友訊科技	150
使用者敲鍵行為辨識系統	微軟	已經與微軟合作開發完成，現由微軟推廣中
總金額		**877.5**

- **學術成就**

本計畫主持人為謝續平老師，共同主持人為曾文貴老師、黃能富、楊武、趙禧綠、黃世昆、黃育綸與吳育松等老師。參與學校包括國立交通大學與國立清華大學，參與實驗室共有 7 個。我們於交通大學電子資訊大樓內成立異質無線多網安全實驗室，提供跨校之師生計畫交流的平台。目前參與本計畫的碩博士生共計 30 位,其中博士生有 10 位。而在 2009 年本計畫共培育 27 位碩士生畢業，大部份的同學服務於相關企業中(例如：鴻海、華碩、中華電信、智邦、友訊等)，也有繼續留在國內或出國深造博士學位者。

參與本計畫之成員於 2009 年發表於國際重要期刊之論文數共 8 篇，而發表於國際研討會之論文數共 7 篇 (詳細列表如下方所示)，由此可知本計畫之相關成員學術研究成果相當

豐碩。在我們所發表的國際期刊中，有兩篇論文的 impact factor 高於達 2.23 與 2.181，分別是 IEEE Transactions on Wireless Communications 與 IEEE Transactions on Information Forensics and Security。其中 SCI 論文有 6 篇、EI 論文有 6 篇。另外我們也發表了二篇技術報告。

- **國際期刊論文**

1. Y. L. Huang, P. H. Lu, J. D. Tygar, A. D. Joseph, "OSNP: Secure Wireless Authentication Protocol using One-Time Key," to be published in Computers & Security, Fall 2009.
2. H. Y. Tsai, Y. L. Huang, D. Wagner, "A Graph Approach to Quantitative Analysis of Control Flow Obfuscating Transformations," IEEE Transactions on Information Forensics and Security, vol. 4, no. 2, pp. 257-267, June 2009.
3. Yu-Lun Huang, Alvaro Cardenas, Saurabh Amin, Song-Zyun Lin, Hsin-Yi Tsai and S. Shankar Sastry, "Understanding the Physical and Economic Consequences of Attacks on Control Systems," International Journal of Critical Infrastructure Protection, vol. 2, no. 3, pp. 73 – 83, Oct. 2009.
4. S.-C. Tsai, W.-G. Tzeng, Kun-Yi Zhou, "Key Establishment Schemes Against Storage-Bounded Adversaries in Wireless Sensor Networks," IEEE Transactions on Wireless Communications, vol. 8, no. 3, pp. 1218-1222, 2009.
5. Shih-I Huang, Shiuhpyng Shieh, Doug Tygar, "Secure Encrypted-Data Aggregation for Wireless Sensor Networks," accepted for publication, ACM Journal of Wireless Networks, 2009.
6. Shih-I Huang, Shiuhpyng Shieh, "Secret Search Mechanism for Wireless Sensor Networks with Passive RFIDs," accepted for publication, International Journal of Security and Networks, 2009.
7. H.-Y. Lin, W.-G. Tzeng, "Anonymous Password Based Authenticated Key Exchange with Sub-linear Communication," Journal of Information Science and Engineering, vol. 25, no. 3, pp. 907-920, 2009.
8. C.-K. Chu, W.-G. Tzeng, "Efficient Identity-Committable Signature and Group-Oriented Ring Signature Schemes," to be published in Journal of Information Science and Engineering vol. 25, no. 5, 2009.

- **國際會議論文**

1. Z. Lin, A. Cardenas, H. Y. Tsai, S. Amin, Y. L. Huang and S. Sastry, "Understanding the Physical Consequences of Attacks on Control Systems," Third Annual IFIP Working Group 11.10 International Conference on Critical Infrastructure Protection, March 2009.
2. Yu-Lung Huang, Chih-Ya Shen, Shiuhpyng Shieh, Hung-jui Wang, Cheng-Chun Lin, "Provable Secure AKA Scheme with Reliable Key Delegation in UMTS," IEEE Conference on Secure Software Integration and Reliability Improvement, 2009.
3. Nen-Fu Huang, Yen-Ming Chu, Chi-Hung Tsai, Wei-Jin Tzeng and Wei-Zen Huang, "Resource-Efficient Traffic Localization Scheme for Multiple BitTorrent," IEEE ICC, 2009.
4. Nen-Fu Huang, Hung-Shen Wu, and Guan-Hao Lin, "Identifying the Use of

Data/Voice/Video-based P2P Traffic by DNS-query Behavior," IEEE ICC,2009.

5. Yu-Ting Yu and His-Lu Chao, "Redundancy-Based Delivery Mechanism for Error-Prone Wireless Networks," IEEE VTC-Spring, pp. 1-5, 2009.
6. TakChon Lou and His-Lu Chao, "On Synchronized Channel Sensing and Accessing for Cognitive Radio Users in IEEE 802.11 Wireless Networks," to be published in IEEE PIMRC Sept. 2009.
7. Shie-Yuan Wang and His-Lu Chao, "On Multi-hop Forwarding over WBSS-based IEEE 802.11(p)/1609 Networks," to be published in IEEE PIMRC Sept. 2009.

■ **技術報告**

1. 由黃育綸教授、謝續平教授與蔡欣宜博士生發表的「SWOON: 安全無線疊蓋觀測網路」技術報告刊於 NCP 網路通訊國家型科技計畫。
2. 由謝續平教授與林佳純博士生發表的「我國異質無線網路資安技術研發方向」技術報告刊於 NCP 網路通訊國家型科技計畫。

● **專利**

以下列出目前申請的專利，其中國內專利申請共有三件、國內專利申請共有四件。

■ **國外專利**

- S.I. Huang, S.P. Shieh, and C.W. Wang, "Light-Weight Authentication and Secret Retrieval Scheme and Its Applications," USA patent pending.
- Yu-Ting Yu, Jia-Long Liou, and Hsi-Lu Chao, "A pre-scheduling based bandwidth aggregating uplink packet distribution mechanism for heterogeneous networks," USA patent pending.
- S.I. Huang and S.P. Shieh, "Method and System for Secure Data Aggregation in Wireless Sensor Networks," USA patent pending.
- S.I. Huang and S.P. Shieh, "無線感測器網路中安全資料聚合的方法和系統," 大陸專利申請中。

■ **國內專利**

1. S.I. Huang and S.P. Shieh, "無線感測器網路中安全資料聚合的方法和系統," 大陸專利申請中。
2. 黃士一、謝續平、王繼偉，"輕量網路安全認證機制及秘密資料擷取方法與其應用"， 臺灣專利申請中。
3. 黃士一、謝續平，"無線感測器網路中安全資料聚合的方法和系統"，臺灣專利申請中。
4. 尤昱婷、劉家隆、趙禧綠，"以前排程為基礎的異質網路頻寬聚集上行封包排程機制"，臺灣專利申請中。

七、 重大突破

本計畫的重大突破可就異質無線多網核心網路滲透檢測與行動裝置滲透檢測兩大方面來敘述。在異質無線多網路與核心網路滲透檢測方面，我們發現 WiMAX 與 3.5G 核心網路中的弱點；在行動裝置滲透檢測方面，本計畫開發的文件檔案惡意程式檢測系統 (Forenser)以及動態惡意軟體分析檢測工具 (Malware Behavior Analyzer, MBA@TWISC) 能對文件檔提供自動化惡意程式的檢測服務以及分析惡意程式行為，預測惡意程式的攻擊路徑。此外，本計畫制定了遊戲業者安全檢測標準。以下將詳細敘述之。

●WiMAX 網路滲透檢測

在 WiMAX 網路滲透檢測方面，經由測試與分析工研院所建置的 WiMAX 科學園區實驗網路 (從新竹市建構到竹東，涵蓋科學園區)，我們發現到 WiMAX 網路中的 Subscribers 會遭受 137 port 的 DoS attack 與 DDoS attack。另外我們也發現了不需認證即可連線的 frequency 。這些弱點與設定上的不周全將使得 WiMAX 網路的安全性受到威脅。

●3.5G 核心網路滲透檢測

在 98 年我們已對台灣數家 ISP 業者的 3.5G 網路進行拓樸探索，並透過實驗瞭解各家 ISP 業者對於防火牆的設置和設定情形。另外經由模擬、測量與分析得知若大約 11750 個被駭客所控制的行動電話同時執行 insert_call_forwarding 這個要求，會使得 3G 核心網路中的 Home Location Register (HLR)無法正常提供服務。因此這個弱點將導致 3.5G 網路失效，這項重大突破可幫助 ISP 業者知道 3.5G 核心網路的弱點，讓他們提供因應措施來解決此類合法封包的 DDoS 攻擊。

●文件檔案惡意程式檢測系統 (Forenser)

Forenser 能夠對文件檔提供自動化惡意程式的檢測服務，Forenser 可偵測來自系統內文件檔、網路下載、電子郵件附檔的網路惡意程式，以提供完整的掃瞄報告以利專家判讀。這項重大突破可有效地找出隱藏於任何文件檔案中的惡意程式碼。該系統已經經由軟體授權技術移轉給中華電信、國安局、工研院、中科院。宏碁目前也正以本系統推廣使用中。

●動態惡意軟體分析檢測工具 (Malware Behavior Analyzer, MBA@TWISC)

MBA@TWISC (Malware Behavior Analyzer)是一套動態惡意程式行為分析工具，可即時收集並分析各種惡意程式 (malware)機器碼的行為，其選擇性、平行分析技術為目前最新設計，是目前唯一速度可達即時分析的系統。此項重大突破可預測惡意程式的攻擊路徑，偵測變形的惡意程式，徹底解決目前特徵值比對技術的不足。透過適度的措施，能夠降低惡意程式對系統的威脅。

●訂定遊戲業者安全檢測標準

本計畫在與台北市電腦商業同業公會的技術服務中提供了對戲谷、新幹線等網路遊戲業者等線上遊戲的安全檢測，另外我們訂定遊戲業者資安檢測標準，此項重大突破可提供遊戲業者來遵循，使得線上遊戲的安全檢測標準化。

八、 結論與展望

隨著 WiFi 無線網路及 3.5G 行動上網的普及以及智慧型行動裝置與小筆電的市場佔有提

升，越來越多使用者習慣利用手機或是小型行動裝置上網接收訊息、收發 e-mail 或查閱資料。在行動資訊越來越普及的時代，我們需要一套安全機制或是檢測標準來衡量異質無線多網與各種行動裝置的安全性。本計畫建置並開發一套異質無線網路安全檢測平台，對於無線網路與行動裝置，提供相關的安全檢測工具與系統，提供使用者評量其行動網路使用環境的安全性。本異質無線網路安全檢測平台分為兩部分：(1)異質無線多網安全檢測，(2)行動裝置之系統與軟體安全檢測。檢測範疇涵蓋使用者行動裝置系統與行動網路環境，提供完整的行動平台安全評量。

在 98 年，本計畫有豐碩的成果，我們總共開發了 13 個子系統或工具。我們在 98 年發表於國際期刊之論文共有 8 篇，而發表於國際研討會之論文數共 7 篇。另外，我們在技術移轉與產學合作方面的成果包括技術移轉（軟體授權）共 5 件，總金額為 290 萬、技術服務共 4 件，總金額為 220 萬、產學合作共 6 件，總金額為 877.5 萬。另外我們也提供了 5 件檢測服務。藉由此平台的建置與檢測工具的開發，我們可提供政府機關、財團法人及高科技廠商無線網路安全檢測的服務，幫助上述單位發現漏洞及弱點。此外，我們所建置的工具可為產業界創造上億元以上的經濟效益、提升無線產品附加價值、節省因網路攻擊或系統弱點所消耗的產值、節省專業檢測人力並且有效減少無線網路環境的攻擊。

藉由我們所建置的異質無線多網安全檢測平台，我們可以滿足國內目前對於無線網路安全檢測上的需求。未來我們將提供更功能更為完善之安全檢測項目，以及支援更多樣的網路類型，例如：LTE 或是其他的 4G 網路技術。

參考文獻

[1] P. Montesinos, L. Ceze, and J. Torrellas, “DeLorean: Recording and Deterministically Replaying Shared-Memory Multiprocessor Execution Efficiently,” in Proceedings of the 35th International Symposium on Computer Architecture, 2008.

[2] P. Montesinos, M. Hicks, S. T. King et al., “Capo: a software-hardware interface for practical deterministic multiprocessor replay,” in Proceeding of the 14th international conference on Architectural support for programming languages and operating systems, Washington, DC, USA, 2009.

[3] S. T. King, G. W. Dunlap, and P. M. Chen, “Debugging operating systems with time-traveling virtual machines,” in Proceedings of the annual conference on USENIX Annual Technical Conference, Anaheim, CA, 2005.

[4] J. Chow, T. Garfinkel, and P. M. Chen, “Decoupling dynamic program analysis from execution in virtual environments,” in USENIX 2008 Annual Technical Conference on Annual Technical Conference, Boston, Massachusetts, 2008.

[5] D. A. S. d. Oliveira, J. R. Crandall, G. Wassermann et al., “ExecRecorder: VM-based full-system replay for attack analysis and system recovery,” in Proceedings of the 1st workshop on Architectural and system support for improving software dependability, San Jose, California, 2006.

[6] G. W. Dunlap, D. G. Lucchetti, M. A. Fetterman et al., “Execution replay of multiprocessor virtual machines,” in Proceedings of the fourth ACM SIGPLAN/SIGOPS international conference on Virtual execution environments, Seattle, WA, USA, 2008.

[7] G. W. Dunlap, S. T. King, S. Cinar et al., “ReVirt: enabling intrusion analysis through virtual-machine logging and replay,” SIGOPS Oper. Syst. Rev., vol. 36, no. SI, pp. 211--224, 2002.

[8] H. Liu, H. Jin, X. Liao et al., “XenLR: Xen-based Logging for Deterministic Replay,” in Proceedings of the 2008 Japan-China Joint Workshop on Frontier of Computer Science and Technology, 2008.

[9] F. Qin, J. Tucek, Y. Zhou et al., “Rx: Treating bugs as allergies---a safe method to survive software failures,” ACM Trans. Comput. Syst., vol. 25, no. 3, pp. 7, 2007.

[10] J. Tucek, S. Lu, C. Huang et al., “Triage: diagnosing production run failures at the user's site,” in Proceedings of twenty-first ACM SIGOPS symposium on Operating systems principles, Stevenson, Washington, USA, 2007.

[11] K. Nance, M. Bishop, and B. Hay, “Virtual Machine Introspection: Observation or Interference?,” IEEE Security and Privacy, vol. 6, no. 5, pp. 32--37, 2008.

[12] S. Yasushi, “Jockey: a user-space library for record-replay debugging,” in Proceedings of the sixth international symposium on Automated analysis-driven debugging, Monterey, California, USA, 2005.

[13] S. M. Srinivasan, S. Kandula, C. R. Andrews et al., “Flashback: a lightweight extension for rollback and deterministic replay for software debugging,” in Proceedings of the annual conference on USENIX Annual Technical Conference, Boston, MA, 2004.

[14] S. Narayanasamy, G. Pokam, and B. Calder, “BugNet: Continuously Recording Program Execution for Deterministic Replay Debugging,” in Proceedings of the 32nd annual international symposium on Computer Architecture, 2005.

[15] M. Xu, R. Bodik, and M. D. Hill, “A "flight data recorder" for enabling full-system multiprocessor deterministic replay,” in Proceedings of the 30th annual international symposium on Computer architecture, San Diego, California, 2003.

[16] A. Dinaburg, P. Royal, M. Sharif, and W. Lee, “Ether: Malware Analysis via Hardware Virtualization Extensions”, CCS’08, October 27–31, Alexandria, Virginia, USA 2008.

[17] Xu Chen, Jon Andersen, Z. Morley Mao, “Towards an Understanding of Anti-virtualization and Anti-debugging Behavior in Modern Malware,” International Conference on Dependable System & Networks, pp.177-186, 2008.

[18] Min Gyung Kang, Heng Yin, Dawn Song, “Emulating Emulation-Resistant Malware,” 2nd Workshop on Virtual Machine Security (VMSec’09), November 9, 2009.

[19] M. G. Kang, P. Poosankam, and H. Yin, “Renovo: a hidden code extractor for packed executables,” in Proceedings of the 2007 ACM workshop on Recurring malcode, Alexandria,

Virginia, USA, 2007.

[20] Ramesh Govindan and Hongsuda Tangmunarunkit, “Heuristics for Internet Map Discovery,”in Proceedings of IEEE INFOCOMM, 2000

[21] Damien Magoni and Mickael Hoerdt, “Internet Core Topology Mapping and Analysis,” *Computer Communications*, vol. 28, pp. 494 – 506, 2005

[22] Paul Barford and Azer Bestavros and John Byers and Mark Crovella, “On the Marginal Utility of Network Topology,” in Proceedings of ACM SIGCOMM Internet Measurement Workshop (IMW), 2001

[23] Neil Spring and Ratul Mahajan and David Wetherall and Thomas Anderson, “Measuring ISP Topologies with Rocketfuel,” *IEEE/ACM Transactions on Networking*, vol. 12, 2004

[24] M. Bishop, “About Penetration Testing, ” IEEE Security & Privacy, vol. 5, no. 6, pp. 84 – 87, Nov. — Dec., 2007

[25] 漢昕科技，“網路滲透測試服務” http://www.bccs.com.tw/index.asp?module=product&action=ShowContext&BID=15&ID=3&SubMenu=3

[26] Bing Duan, Yinqian Zhang, Dawu Gu, **“**An Easy-to-Deploy Penetration Testing Platform,” proceedings of the 9th International Conference for Digital Object Identifier, pp. 2314 – 2318, Nov. 18-21, 2008

[27] L. Batyuk, A.-D. Schmidt, H.-G. Schmidt, A. Camtepe, and S. Albayrak. Developing and benchmarking native linux applications on android. In MobileWireless Middleware, Operating Systems, and Applications, 2009.

[28] A.-D. Schmidt, H.-G. Schmidt, L. Batyuk,J. –H. Clausen, S. –A. Camtepe, and S. Albayrak. Smartphone Malware Evolution Revisited: Android Next Target. 2009.

[29] G. E. Suh, J. W. Lee, D. Zhang, and S. Devadas. “Secure program execution via dynamic information flow tracking”, in ACM ASPLOS-XI, 2004.

[30] F. Qin, C.Wang, Z. Li, H. Kim, Y. Zhou, and Y. Wu. “LIFT: A low-overhead practical information flow tracking system for detecting security attacks”, in MICRO-39, 2006.

行政院國家科學委員會補助國內專家學者出席國際學術會議報告

98 年 11 月 15 日

報告人姓名	黃育綸	服務機構及職稱	國立交通大學電機工程系
會議時間 會議地點	Nov 9 – 13, 2007 Chicago, Illinois, USA	本會核定補助文號	
會議名稱	(中文) 第 16 屆 ACM 電腦與通訊安全國際研討會 (英文) 16th ACM Conference on Computer and Communications Security		
發表論文題目	Security Analysis for Process Control Systems		

表 Y04

報告內容應包括下列各項：

一、參加會議經過

本次大會總共接受 59 篇論文，分爲 18 個 Sessions 發表，包括 Attack, Applied Cryptography, RFID, Cloud Computing, Security of Mobile Services, Malware & Bots 等。大會安排一場由 DOROTHY E. DENNING 教授主講的 ``Designing Secure Passwords”，講解各種密碼的組合與安全度。從資訊安全最基本的密碼到整個系統的安全度，談到過去十多年來密碼與使用者、密碼與應用程式、系統管理者與忘記密碼之使用者之間的互動與牽連關係，進行精闢之解說與剖析。大會另外安排五場 Toutorial，分別針對電力網路系統安全、電腦網路安全評估、無線網路安全、安全網頁設計等議題加以說明闡述。

二、 與會心得

此行最主要目的爲發表 ``Security Analysis for Process Control Systems”論文一篇。雖是比較偏控制領域的一種安全分析，但因爲從不同角度點出控制系統安全與電腦安全的不同，因此也獲得許多國外專家學者的迴響與建議，包括：

1. 參考訊號的產生，除了 Linear Model (Deterministic Probability) 之外，也可以考慮其他 Random Process 方法的引入。
2. 既然控制安全與電腦安全有異，可以更深入探討攻擊者的攻擊手法與攻擊點。
3. 可以考慮套用到其他 PCS，如 medical device 的攻擊與防禦策略。

另節錄大會中的幾篇重要論文：

1. Computational Soundness for Key Exchange Protocols with Symmetric Encryption

   一般我們常用正規分析討論安全協定之安全度，但是這類的安全分析是否隱喻對各種新密碼技術的保證，卻是大家需要深思的問題。這篇論文主要針對計算結果的完整度討論採用對稱式金鑰交換之安全協定的強度。作者開發一套可以自動檢查的條件，透過觀察計算相同度，來驗證金鑰交換協定是否滿足其所宣稱之安全度與目的。文中採用到的 π-calculus 是我所不熟悉的，但大會中多篇論文一再提及此計算方法，是將來研究時所可以考慮引用的一種設計基礎。

2. On Cellular Botnets: Measuring the Impact of Malicious Devices on a Cellular Network Core

   本篇論文討論惡意或已被入侵之行動裝置（手機）對整個行動通訊網路所可能造成的影響，尤其是在無線頻寬的佔用，以及行動核心網路的處理成本方面。由於過去相關文獻多著重於討論藍芽傳輸與 MMS 的惡意程式傳播機制，對無線頻寬與營運成本方面較少探討，因此，這篇論文的論述重點（如何量測惡意程式碼對行動網路的影響程度）在大會上引起很大的回響，發問者眾多。另外，作者以爲行動網路有別於網際網路，攻擊者必須非常瞭解行動網路核心結構、底層訊號傳送方式，才能真正入侵到行動網路中。然而，一旦入侵成功，至少都是以「區碼」爲範圍的感染，感染程度可能高達該區碼內 90% 以上的裝置。

3. On Lightweight Mobile Phone Application Certification

   也是一篇行動裝置相關的安全研究論文。主要著重於探討智慧型手機下載網路程式時所可能受到的威脅。作者並以 Android 爲主，設計輕量安全通訊機制，使能

透過 Security Requirements 的確認，評估手機應用程式在「功能」與「設定」等方面的安全程度與可能受到的威脅。

三、考察參觀活動(無是項活動者省略)

無

四、建議

相較於其他國家，國內資安相關研究並不熱烈，人才也需大力培養，否則落後其他國家太多。

五、攜回資料名稱及內容

資料名稱

CCS 2009 國際研討會論文集，ISBN 978-1-60558-894-0，ACM order No. 537091

六、其他

1. 遇到 Professor Virgil Gligor、Professor Adrain Perig、Professor Heng Yin 等人，並交換許多寶貴的研究經驗與資訊。
2. CCS 一直以來致力於引領並促進全球資通訊安全的技術研究。今年幾場 sessions 下來，從聽眾的反應與問題中，可以略微察覺，相較於理論密碼技術，現在各國對行動網路安全與 Malware 惡意程式碼的重視更甚於從前，也應該是國內專家學者所應努力的方向。

S

Security Analysis for Process Control Systems

Zong-Syun Lin[†], Alvaro A. Cárdenas[‡], Saurabh Amin[‡], Hsin-Yi Tsai[†], Yu-Lun Huang[†] and Shankar Sastry[‡]

[†] National Chiao Tung University, Taiwan
[‡] University of California, Berkeley

ABSTRACT

We present security analysis of process control systems (PCS) when an attacker can compromise sensor measurements that are critical for maintaining the operational goals. We present the general sensor attack model that can represent a wide variety of DoS and deception attacks. By taking example of a well studied process control system, we discuss the consequences of sensor attacks on the performance of the system and important implications for designing defense actions. We develop model-based detection methods that can be tuned to limit the false-alarm rates while detecting a large class of sensor attacks. From the attacker's viewpoint, we show that when the detection mechanisms and control system operations are understood by the attacker, it can carry stealth attacks that maximize the chance of missed detection. From the defender's viewpoint, we show that when an attack is detected, the use of model-based outputs maintains safety under compromised sensor measurements.

1. INTRODUCTION

Control systems are computer-based systems that *monitor* and *control* physical processes. These systems represent a wide variety of networked information technology (IT) systems connected to the physical world. Depending on the application, these control systems are also called Process Control Systems (PCS), Supervisory Control and Data Acquisition (SCADA) systems, or Cyber-Physical Systems (CPS). The overall objectives of these control systems are: (1) to maintain safe operational goals by limiting the probability of undesirable behavior, (2) to meet the production demands by keeping certain process values within prescribed limits, (3) to maximize production profit.

Control systems are more vulnerable today than in the past due to the increased standardization of technologies, the increased connectivity of control systems to other computer networks and the Internet, insecure connections, etc. Because of the increasing risk to computer attacks, there has been a significant effort in recent years to discuss and identify the security issues of control systems [1, 2, 4–7, 9–11, 13–15].

In this proposal we focus on attacks on the *regulatory layer*. The regulatory control layer has direct access to the sensors that measure the process variables and is responsible for nominal safety and operation of the processes in the system. Since the regulatory layer controllers are required to demonstrate faster response, they are traditionally based on the classic proportional-integral-derivative (PID) algorithms.

2. OUR APPROACH

We believe that most of the previous work in the security of control systems has three goals: (1) create awareness of security issues with control systems, (2) help control systems operators and IT security officers design a security policy, and (3) recommend basic security mechanisms for prevention (authentication, access controls, etc), detection, and response to security breaches.

While these recommendations and standards have placed significant importance in the *survivability* of control systems; we argue that they have not considered new research problems that arise when control systems are under attack. In particular, researchers have not considered how attacks affect the estimation and control algorithms -and ultimately, how attacks affect the physical world.

In this work we argue that the major distinction of control systems with respect to other IT systems is the interaction of the control system with the physical world. We propose to incorporate the physical process dynamics in the security analysis of the control system and focus on an attacker that compromises sensor readings. We have two major goals (1) to develop a threat assessment methodology, and (2) to design attack detection and response mechanisms.

3. ATTACK MODELS

In this proposal we focus on attacks on sensor networks and the effects they can have on the process control system. We consider the case when the state of the system is measured by a sensor network of p sensors that observes the measurement vector $y(k) = \{y_1(k), \ldots, y_p(k)\}$, where $y_i(k)$ denotes the measurement by sensor i at time k. All sensors have a dynamic range that defines the domain of y_i for all k. That is, all sensors have defined minimum and maximum values $\forall k, y_i(k) \in [y_i^{\min}, y_i^{\max}]$. Let $\mathcal{Y}_i = [y_i^{\min}, y_i^{\max}]$. We assume each sensor has a unique identity protected by a cryptographic key.

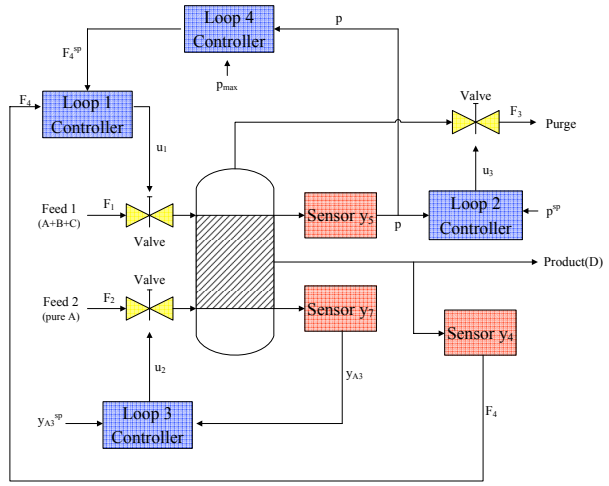


Figure 1: Architecture of the Simplified TE Plant

Let $\tilde{y}(k) \in \mathbb{R}^p$ denote the *received measurements by the controller* at time k. Based on these measurements the control system defines control actions to maintain certain operational goals. If some of the sensors are under attack, $\tilde{y}(k)$ may be different from the real measurement $y(k)$; however, we assumed that the attacked signals $\tilde{y}_i(k)$ also lie within $\mathcal{Y}_i$ (signals outside this range can be easily detected by fault-tolerant algorithms).

Let $\mathcal{K}_a = \{k_s, \ldots, k_e\}$ represent the attack duration; between the start time k_s and stop time k_e of an attack. A general model for the observed signal is the following:

$$\tilde{y}_i(k) = \begin{cases} y_i(k) & \text{for } k \notin \mathcal{K}_a \\ y_i(k) + \lambda_i(k) & \text{for } k \in \mathcal{K}_a \\ y_i^{\min} & \text{for } k \in \mathcal{K}_a,\ y_i(k) + \lambda_i(k) < y_i^{\min} \\ y_i^{\max} & \text{for } t \in \mathcal{K}_a,\ y_i(k) + \lambda_i(k) > y_i^{\max} \end{cases}$$

This general sensor attack model can be used to represent a variety of attacks such as additive injection, multiplicative scaling, replay attacks and DoS attacks.

4. PROCESS DESCRIPTION

To test our attacks, we use the Tennessee-Eastman process control system (TE-PCS) model and the associated multi-loop PI control law as proposed by Ricker [12]. The process architecture and the control loops are described in Figure 1. The *control objective* is to *regulate* F_4, the rate of production of the product D, at a set-point F_4^{sp}, while maintaining P, the operating pressure of the reactor, below the shut-down limit of 3000 kPa as dictated *safety* considerations, such that C, the *operating cost* is minimized.

There are four *input variables*, denoted as u_1, u_2, u_3 and u_4, available to achieve the above control objective. Ricker [12] suggests the input-output pairings (or *control loops*) as seen in Figure 1. The PI control law for the loop$-i$ controller for the k^{th} sampling period is given by

$$u_i(k) = u_i(k-1) + K_i \left(e_i(k) - e_i(k-1) + \frac{\Delta t}{\kappa_i} e_i(k) \right) \quad (1)$$

where $e_i(k) =$ setpoint $-$ measured value of controlled variable for loop$-i$ controller at k^{th} sampling period. The controller settings K_i and κ_i are pre-tuned and given [12]. The control input vector for k^{th} sampling period is denoted as $u(k) = (u_1(k), \ldots, u_4(k))^\top$. We also add a Gaussian disturbance to the control inputs $u(k)$ so that the system is never in a complete steady state.

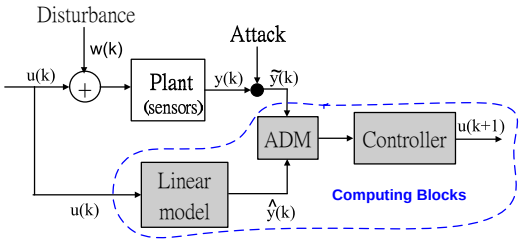


Figure 2: The proposed detection module.

5. THREAT ASSESSMENT

We study the security issues of control systems by experimenting and simulating cyber attacks on sensor signals in the TE-PCS model. Because operating the chemical reactor with a pressure larger than 3000 kPa is unsafe, it may lead to an explosion or damage of the equipment. Assume that the goal of the attacker is to raise the pressure level of the tank to a value larger than 3000 kPa, we attack a single sensor or a single controller at a given time. From the experimental results, we found that the most effective of these attacks were the max / min attacks (make the forged signals the extreme values, i.e, $y^{\max}$ or $y^{\min}$); however, not all of them were able to drive the pressure to unsafe levels. We found out that, in general, the DoS attacks do not affect the plant. We conclude that if the plant operator wants to prevent an attack from making the system operate in an unsafe state, it should prioritize the integrity of the sensors rather than their availability.

6. MODEL-BASED ATTACK DETECTION

Detecting attacks to control systems can be formulated as anomaly-based intrusion detection systems [3]. Our proposed attack detection system is presented in Figure 2. The control input sequence $u(k)$ is fed to the physical system after being perturbed by an additive Gaussian process noise sequence $w(k)$. The process noise sequence can be thought as unmodeled factors that affect the evolution of system state. The input sequence $u(k)$ is also fed to a system model that is representative of the physical system and is internal to the detection system. The internal model will produce an output sequence $\hat{y}(k)$. The anomaly detection module (ADM) will compare the two measurement sequences: the sequence $\tilde{y}(k)$ that is received from the sensor measurements and may have been influenced by the attacker with the sequence $\hat{y}(k)$ that is obtained from the internal model. The ADM raises an alert if the deviation between the two sequences is significant.

To formalize this problem, we need (1) a linear model that is representative of the physical system, and (2) an anomaly detection algorithm. We use the linear model, characterized by the matrices A, B, and C, obtained by linearizing the non-linear TE-PCS model about the steady-state operating conditions. The model dynamics that are linear in

state $x(k) \in \mathbb{R}^n$ and control input $u(k) \in \mathbb{R}^m$ are

$$x(k+1) = Ax(k) + Bu(k) \tag{2}$$

Assume that the system (2) is monitored by a *sensor network* with p sensors. We can obtain the representative measurement sequence, $\hat{y}(k) \in \mathbb{R}^p$, from the observation equations

$$\hat{y}(k) = Cx(k), \tag{3}$$

For our anomaly detection algorithm we use a change detection formulation [8]. The problem formulation is: given a time series sequence $z(1), z(2), \ldots, z(N)$, determine the minimum number of samples, N, the anomaly detection scheme should observe before making a decision d_N between two hypotheses: H_0 (normal behavior) and H_1 (attack). Let

$$z_i(k) := |\tilde{y}_i(k) - \hat{y}_i(k)| - b_i \tag{4}$$

where b_i is a small positive constant chosen such that

$$\mathbb{E}_{H_0}[|\tilde{y}_i(k) - \hat{y}_i(k)| - b_i] < 0 \tag{5}$$

The nonparametric CUSUM statistic for sensor i is

$$S_i(k) = (S_i(k-1) + z_i(k))^+, \; S_i(0) = 0 \tag{6}$$

and the corresponding decision rule is

$$d_{N,i} \equiv d_\tau(S_i(k)) = \begin{cases} H_1 & \text{if } S_i(k) > \tau_i \\ H_0 & \text{otherwise} \end{cases} \tag{7}$$

where τ_i is the threshold selected based on the false alarm rate for sensor i.

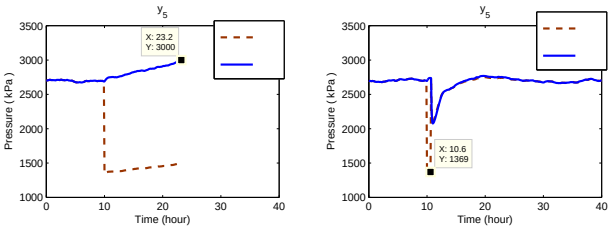
Our response strategy (shown in Fig 2) can be summarized as follows: For sensor i, if $S_i(k) > \tau_i$, the ADM replaces the sensor measurements $\tilde{y}_i(k)$ with measurements generated by the linear model $\hat{y}_i(k)$ (that is the controller will receive as input $\hat{y}_i(k)$ instead of $\tilde{y}_i(k)$). Otherwise, it treats $\tilde{y}_i(k)$ as the correct sensor signal.

7. EXPERIMENTS

In this section, we briefly discuss how our defense system works under attacks. We omit the details for determining the two parameters (b and τ) of the nonparametric CUSUM statistic. We also have to make sure that if there is a false alarm, controlling the system by using the estimated values from the linear system will not cause any safety concerns. We found that while a false response mechanism increases the pressure of the tank, it never reaches dangerous levels.

We now test the detection and response performance of the ADM for certain attacks. Because operating the chemical reactor with a pressure larger than 3000 kPa is unsafe, all our attacks attempt to raise the pressure in the tank. In order to quantify the magnitude of the attack we use multiplicative scaling attacks with parameter λ^m and attack each sensor. Our attacks start at time $T = 10$ hours. We only report attacks on y_5 here. The results for y_4 and y_7 are similar. Sensor y_5 monitors pressure of the reactor. Attacking sensor y_5 by lowering the value makes controller turn down the purge valve to increase pressure. In an unprotected system the safety of the system is compromised at time $T = 23.5$ (hr) if we set parameter of scaling attack λ^m_{y5} to 0.5. With ADM enabled, the attack can be detected at time $T = 10.7$ (hr) and the plant remains stable.

If an attacker compromises two more sensors then he can mount multiple attacks; however these attacks can also be



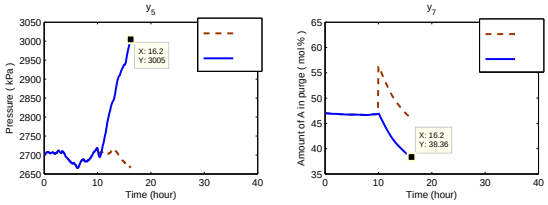
(a) Without ADM

(b) ADM detects and responds to the attack at $T = 10.7$ (hr)

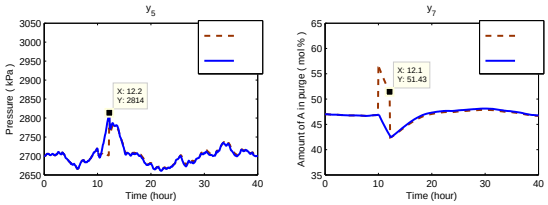
Figure 3: $\tilde{y_5} = y_5 * 0.5$

detected independently by each statistic $S_I(k)$. As an anecdote we attack y_5 with a replay attack and y_7 with a scaling attack. In the original plant system (without ADM), Fig 4 shows that plant goes to an unsafe state at time $T = 16.2$ (hr). Compared with just launching an scaling attack on y_7, the combined attack takes much less time to drive the pressure past safety levels. The reason is that the replay attack on y_5, gives an erroneous information to the controller that tries to prevent an increase in pressure.

If we have an ADM the attack is detected by $S_{y5}(k)$ at time $T = 12.2$ (hr) and independently by $S_{y7}(k)$ at time $T = 12.1$ (hr).



(a) without ADM the pressure grows past safety levels.



(b) The statistics for y_5 and y_7 independently detect the attack.

Figure 4: $\tilde{y_5}(t) = y_5(t-10)$ **&** $\tilde{y_7} = y_7 * 1.2$

8. STEALTH ATTACKS

Although the proposed ADM can detect a wide range of attacks, we consider a more powerful adversary that knows about the detection scheme. We take a conservative approach in our models by assuming a very powerful attacker with knowledge of: (1) the exact linear model that we use, the parameters of the ADM, and (3) the control command signals. Such a powerful attacker may be unrealistic in some scenarios, but one may want to test the resiliency of our system to such an attacker to guarantee safety for a wide range of attack scenarios. The goal of a stealth attacker is to raise the pressure in the tank without being detected. We define and analyze three such attacks in our work.

9. REFERENCES

[1] E. Byres. Designing Secure Networks for Process Control. *IEEE Industry Applications Magazine*, 6:33–39, 2000.

[2] E. Byres and J. Lowe. The myths and facts behind cyber security risks for industrial control systems. In *VDE Congress*, 2004.

[3] D. Denning. An intrusion-detection model. *Software Engineering, IEEE Transactions on*, SE-13(2):222–232, Feb. 1987.

[4] J. Falco, N. I. of Standards, and T. (US). *IT Security for Industrial Control Systems*. US Dept. of Commerce, Technology Administration, National Institute of Standards and Technology, 2002.

[5] GAO. Critical infrastructure protection. Multiple efforts to secure control systems are under way, but challenges remain. Technical Report GAO-07-1036, Report to Congressional Requesters, September 2007.

[6] E. Goetz and S. Shenoi. *Critical Infrastructure Protection, Proceedings of the First Annual IFIP Working Group 11.10 International Conference on Critical Infrastructure Protection*. Springer, Dartmouth College, Hanover, New Hampshire, USA, March 2007.

[7] V. Igure, S. Laughter, and R. Williams. Security issues in SCADA networks. *Computers & Security*, 25(7):498–506, 2006.

[8] T. Kailath and H. V. Poor. Detection of stochastic processes. *IEEE Transactions on Information Theory*, 44(6):2230–2258, October 1998.

[9] T. Kilpatrick, J. Gonzales, R. Chandia, M. Papa, and S. Shenoi. Forensic analysis of SCADA systems and networks. *International Journal of Security and Networks*, 2:95–102, 2008.

[10] P. Oman, E. Schweitzer, and J. Roberts. Safeguarding IEDs, Substations, and SCADA Systems Against Electronic Intrusions. In *Proceedings of the 2001 Western Power Delivery Automation Conference*, pages 9–12, 2001.

[11] M. Papa and S. Shenoi. *Critical Infrastructure Protection II, Proceedings of the Second Annual IFIP Working Group 11.10 International Conference on Critical Infrastructure Protection*. Springer, March 2008.

[12] N. Ricker. Model predictive control of a continuous, nonlinear, two-phase reactor. *JOURNAL OF PROCESS CONTROL*, 3:109–109, 1993.

[13] P. Tsang and S. Smith. YASIR: A Low-Latency, High-Integrity Security Retrofit for Legacy SCADA Systems. *Proceedings of the IFIP TC 11 23rd International Information Security Conference*, 2008.

[14] US-CERT. *Control Systems Security Program*. US Department of Homeland Security, `http://www.us-cert.gov/control_systems/index.html`, 2008.

[15] A. Wright, J. Kinast, and J. McCarty. Low-Latency Cryptographic Protection for SCADA Communications. *LECTURE NOTES IN COMPUTER SCIENCE*, 3089:263–277, 2004.