以 URL 資訊為基礎之網路釣魚偵測系統

曾黎明, 黃克仲, 陳天豪
國立中央大學資訊工程學系
tsenglm@csie.ncu.edu.tw, concon@dslab.csie.ncu.edu.tw, eavoch@dslab.csie.ncu.edu.tw

摘要

由於網路上提供的服務越來越多樣化，使得使用者資訊變得相對地更加有價值。而釣魚攻擊便因此而產生了，加上設立釣魚網站並不會太困難，也因而造成釣魚網站如雨後春筍般越來越多，相對的受害者卻常因為一時不察而掉入陷阱，並將自己的個人資訊洩漏出去。本文提出以 URL 為基礎資訊的釣魚偵測系統，可以在不危害使用者隱私權的情況下，達到防止釣魚攻擊，保護一般使用者免於受騙。另外結合自動填表功能來偵測釣魚網站的轉向行為模式，使得偵測的面向更加多樣化，實驗結果證實自動偵測若能加上有效的填表功能，會使得整個系統的功能性更加的強化。由於本文提出的系統只針對 URL 資訊做起始的偵測基礎，因此本系統不論是設置在伺服端點或是客戶端點都是適用的。

關鍵詞：釣魚，網路詐騙，網路安全。

1. 前言

1.1 研究背景

由 MillerSmiles.co.uk! [1]網站可以得知相關知名網站如 eBay、PayPal、YAHOO!等，都曾經遭受釣魚攻擊的傷害，也造成被害網站的信譽遭受到極大的損傷。而根據 APWG (Anti-Phishing Working Group)五月的報告[2]指出過去這一年來每個月可以收到使用者回報的釣魚網站平均數目約 26882 個，其中有一項數據更為有趣，就是釣魚網站的平均壽命約為 3.8 天，由此可知釣魚網站為了規避偵測而加快汰換的速率，以至於名單式的釣魚防禦工具亦漸趨不敷使用。

釣魚網站究竟造成了多少損失，由於並沒有相關的公司提供資訊，因此確切的數字無法得知，不過，根據研究估計一年損失可能介於十億美金[3]到二十八億美金[4]，由此可知釣魚網站對於網路商業的傷害之大是非常可觀的。

多數的釣魚網站是讓使用者誤以為正在瀏覽正當網站的網頁，以複製或側錄的方式使得使用者所輸入的資訊被記錄下來。因此使用者常在不知不覺間受害，等到自己的權益受損後才驚覺都已太遲了，因此不管是杜絕釣魚網站的設立或是防止使用者上當，在現時已經是一項重要的研究項目。

1.2 研究動機

釣魚網站與釣魚信件這兩者的關係可以說是密不可分，釣魚者通常藉由類似垃圾郵件一樣發送大量的釣魚信件，使用者接收到此類信件後，有機會因信件宣稱的一些訊息，而透過信件內的連結到釣魚網站，並根據網頁提示輸入個人資訊，使整個釣魚攻擊完成。

因此想要防禦釣魚攻擊可從兩點兩方面下手，兩點分別是伺服端點與客戶端點，而兩方面則是 Mail 與 Web。

若從 Web Server 端防禦雖可擁有主動權，但卻因為攻擊者可規避 Server 的防禦手段，而使得在 Server 端的防禦變得較為困難，因此如何提高此端點的防禦手段隱密性，是較為重要的議題。相對的，若從 Web Client 端做偵測防禦則較為被動，而且釣魚網站的欺騙手法較為多變，但卻可以依使用者量身訂做防禦工具，使得防禦的效率可以較為提升，但也由於貼近使用者的緣故，因而有隱私權的問題。

換另一個防禦面來說，如果是從 Mail Server 來做防禦，可從信件傳送的上游處就將 Phishing Mail 阻擋，不會讓使用者接收到此類信件，而且因為在 Server 端可以蒐集到較多的 Mail Pattern，使得偵測工具的參考資訊較為完整，藉以提升偵測正確性，但也會因此而牽涉到使用者的隱私權問題。若是從 Mail Client 端做防禦的動作，雖是處於信件傳送的下游且樣本會較少，但卻可以依使用者習性量身訂做防禦機制，同時卻也牽涉到使用者隱私權的問題。

由於隱私權的問題，因此本論文的架構是基於網頁部份做偵測的動作，網頁部分的資訊包含了 URL、連結、表單…等等資料，對於偵測釣魚網站亦有相當大的價值存在。

1.3 本文架構

本文架構如下：第二章討論相關研究，先描述現行的網路釣魚方式及手法，再討論防禦與追蹤策略；第三章說明本系統的設計與方法；第四章描述系統實作；第五章利用實驗來驗證本系統之可行性；第六章為結論及未來可能的研究方向。

2. 相關研究

2.1 什麼是釣魚－網路詐騙

釣魚(Phishing)又稱為網路詐騙(Web Spoofing)，最早是起源於駭客利用電話線實行犯罪，因此將 Phone 與 Fishing 結合在一起創造出 Phishing 一詞[5]。

釣魚的定義是“釣魚者將使用者導向到一個偽造的釣魚網站，讓使用者誤以為是正當網站，並在網站中洩露個人資訊”，於 APWG[6]的首頁有更詳細的定義。

釣魚這種行為，最終目的就是想得到有價值的使用者個人資訊，因此除了使用一些社會工程的技巧誤導使用者外，亦需要有偽造或複製的能力，甚至於對一些正當網站的防禦技術亦要有能力可以破解，最常見的釣魚網站設立技術可以概分為兩種：

(1)複製正當網站，並修改有關個人登錄資訊的網頁紀錄表單；

(2)使用 script 或是類似手法，在視窗內載入正當網站，並側錄使用者鍵盤輸入。

2.2 為何釣魚能成功

本節討論為何釣魚攻擊可以成功的原因。根據 Dhamija 等人提出的研究[7]可知有以下三個原因：

(1) 知識的缺乏；(2)被視覺欺騙；(3)不佳的注意力

2.3 教育使用者

Anti-Phishing Phil [8]是一個遊戲，它教導使用者如何從瀏覽器相關資訊辨識出釣魚網站，以及如何使用搜尋引擎找出正當網站。

2.4 Prevention

此節介紹一些如何預防釣魚網站的機制，相關介紹於[9]。Content Verification Certificates (CVC)[10]使用 PKI (Public Key Infrastructure)對網頁上的物件做運算後，得到與 IP/URL 相對應的電子簽章，因此若此類物件被釣魚者拷貝並貼於釣魚網站上時，只要使用者採用驗證引擎 (Verification Engine)，一種 IE Plug-in 的工具來做驗證，會使釣魚網站上的物件全部都被遮罩，而正當網站的此類物件並不會被遮罩，因此使用者可以很輕易的判別出釣魚網站。

TrustLogo[11]是由 IDAuthority™ 對於可信任的網站所即時給予的線上“信任標籤”，讓使用者可以安心的在網站上面進行交易。

另外，由於目前的寄信協定 SMTP (Simple Mail Transfer Protocol)對於寄件者的規範並不是很嚴格，因此要偽造寄件者欄位是非常簡單，所以垃圾郵件一直以來都釣魚信件的大宗來源。故加強信件的認證機制對於防止釣魚信件及垃圾信件是有非常大的助益的。

2.5 使用者介面

Web Wallet[12]採用瀏覽器的 Sidebar 形式，在使用者啟用 Web Wallet 後，Web Wallet 會以“網頁輸入資訊錢包”的形式監控使用者輸入個人資訊。其他尚有許多使用者介面工具以 toolbar 形式防範釣魚攻擊，於此文不加贅述。

2.6 偵測

本節將做除了一般常見的黑名單及白名單外的一些相關研究介紹。CANTINA[13]採用 TF-IDF (term frequency-inverse document frequency)演算法將網頁上面出現的字詞做紀錄，並製作 term 的出現頻率表，將出現率最高的 M 個 term 送至搜尋引擎，若網頁的網域與搜尋引擎前 N 筆結果的網域相同，則視為正當網站。

SpoofGuard[14]採用網頁歷史紀錄(包含域名、URL、連結、影像)與正在連線之網頁做相似度的比對，若相似度高但卻不在同一個網域，則發出警告。另外攔截使用者輸入資訊，並比對使用者曾輸入過的相同個人資訊之網頁，若比對結果有差異時對使用者發出危險提醒。由於 SpoofGuard 會有紀錄使用者輸入之資訊，因此採用雜湊演算法將資訊做加密，以免有侵害隱私權的疑慮。

Cloudmark[15]使用自己架設的 CNFS (Cloudmark Network Feedback System)接收使用者回報資訊，加上 honeypot 系統所包含的資訊，使用 cloudmark fingerprinting 演算法，將這些資訊做成特徵，並拿來與新進的資訊相比較，藉以找出釣魚類型的攻擊。

Visual Similarity Assessment (VSA)[16]分別蒐集正當網站以及釣魚網站的網頁展示格式之特徵，再將兩種特徵做比較，若相似則為釣魚網站，見圖 2.8。

Web Bugs and Honeytokens[17]提出追蹤釣魚者的方法，首先在建立正當網站時，於網頁物件中放置一個會回報位置的小蟲，當釣魚者將正當網站連同小蟲一起放上釣魚網站時，小蟲會將位置回報給正當網站設立者，於是就可以依小蟲傳回的資訊，前往釣魚網站並於該網站留下 Honeytoken 的資料，之後若有任何存取 Honeytoken 資訊的使用者，就可判定其一定與該釣魚網站有所關聯，進而追到釣魚者。

2.7 系統比較（針對偵測部分）

由於本系統是針對偵測釣魚攻擊部分，因此只對偵測工具做比較，見表一及表二。

另外針對評估項目做概略說明：

(1)對象：依使用對象分為 Server 端及 End-User 端。

(2)方法：該項目所採用之抵禦釣魚攻擊的手段。

(3)所需資訊：該項目於運作時所需之資訊。

表一 偵測系統比較

	CANTINA	SpoofGuard	Cloudmark

對象	End-user	End-user	End-user
方法	1.Signature 2.Search Engine	1.History Check	1.User report 2.Fingerprint
所需資訊	1.Web page	1.Web page 2.URL 3.User info	1.Web page

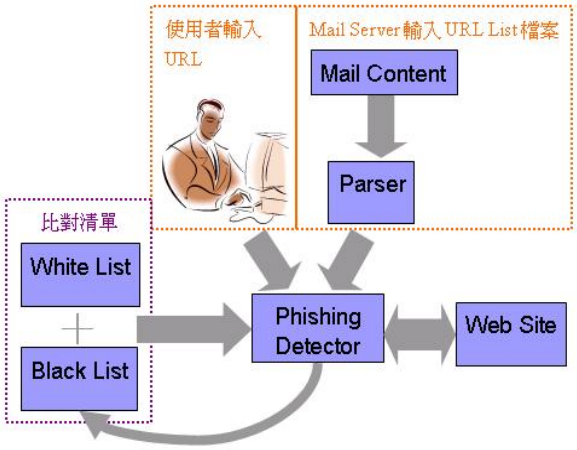
表二 偵測系統比較

	VSA	Web Bug Honeytoken	本系統
對象	End-user	Web Server	Mail Server End-user
方法	1. Layout analysis	1.Web Bug 2.Honeytoken	1.URL Check
所需資訊	1.Web page	1.Bug Report 2.Honeytoken	1.URL 2.Web page

由於目前的網路釣魚偵測工具對於使用者資訊會有隱私權的問題，因此本文提出以 URL 資訊為偵測基礎，並且對於網頁物件做簡單的來源連結分析，加上使用自動填入表單資訊來做進一步的偵測，使得 URL 轉向的資訊能夠完整的獲得並加以分析。另外，由於本系統所需資訊僅限於 URL，因此亦可設置於 Mail Server 或是 Web Client 端。

3. 系統設計

如圖一所示，整個系統可分為三部份來描述：(1)使用者選擇單筆 URL 輸入或是經由 Parser 輸出之 URL 清單檔案輸入，Phishing Detector 會根據輸入來源開始執行偵測。(2)Phishing Detector 會將比對清單讀入後與使用者輸入的 URL 做比對。若 URL 與白名單相符則為正當網站，反之，若符合黑名單則為釣魚網站。(3)當 Phishing Detector 在做清單比對沒有結果後，便開始執行網頁偵測的 component。



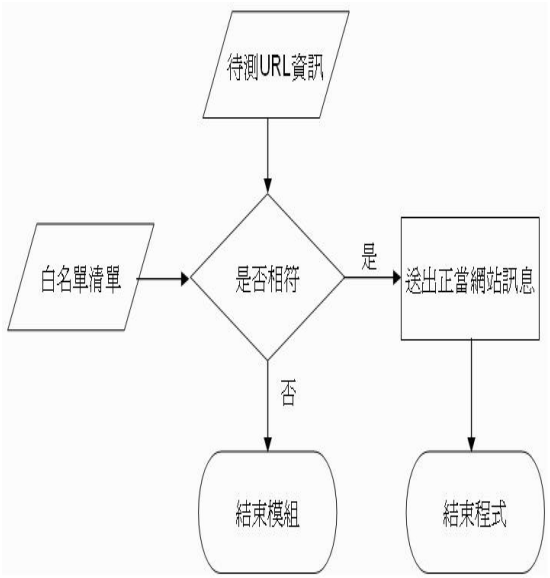
圖一 URL-Based Phishing Detecting System

Phishing Detector 包含四個模組，模組說明包含設計依據緣由及動作：

(1)White List Checker：(流程如圖二)

1.White List 可依據使用者瀏覽習慣量身訂做。

2.比對使用者輸入之 URL 與白名單是否有相符合，若有則為正當網站並離開程式，反之離開模組。

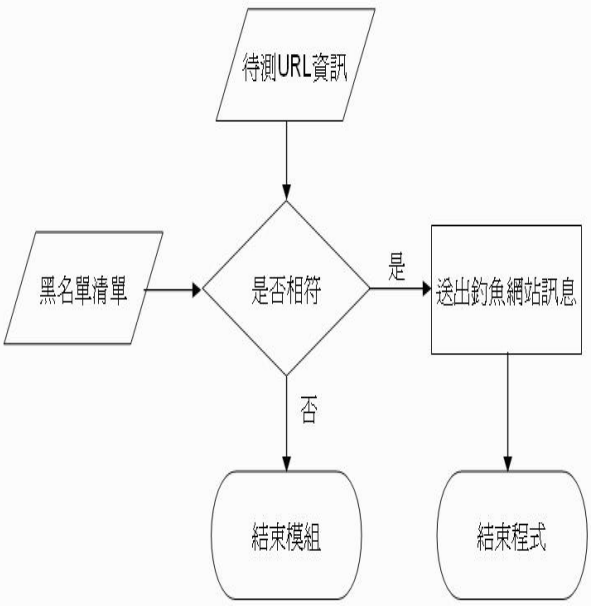


圖二 白名單模組流程圖

(2)Black List Checker：(流程如圖三)

1.Black List 可改善大量偵測的效率，於系統偵測到釣魚網站時，將之加入以更新黑名單。

2.比對使用者輸入之 URL 與黑名單是否有相符合，若有則為釣魚網站並離開程式，反之離開模組。

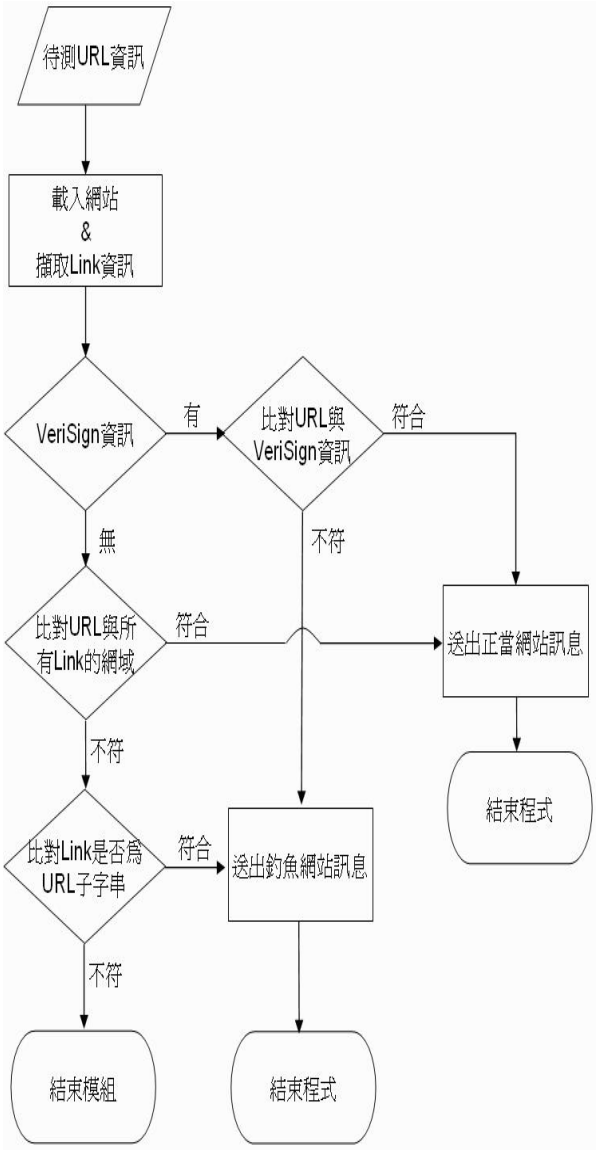


圖三 黑名單模組流程圖

(3)Relativity Checker：(流程如圖四)

1.將網頁所有連結資訊分為三類：正當網站連結、釣魚網站連結、合作網站連結，由於釣魚網站的 URL 通常會包含正當網站的部份 URL 字串，如圖五及圖六，因此針對前兩種連結做比較。

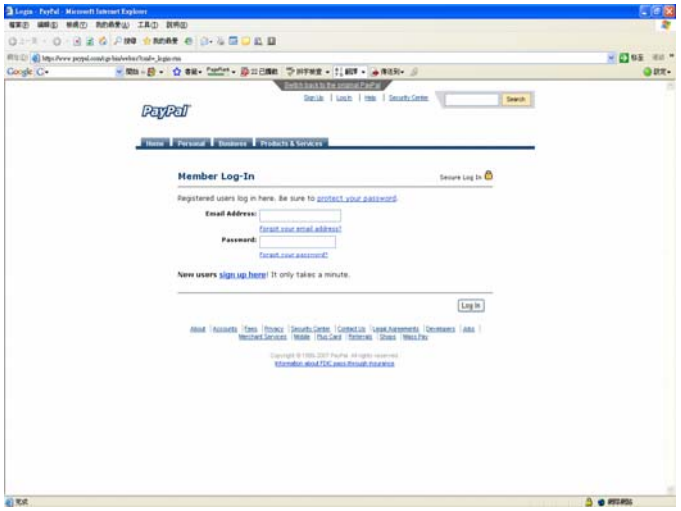
2.比對 URL 與該網頁上所有連結的網域(domain name)，若有不相符的連結，則將該連結的網域拿來與 URL 比對，看是否為 URL 的部份字串，若有則為釣魚網站，若無則跳離模組。



圖四 Relativity 模組流程圖

正當網站URL:

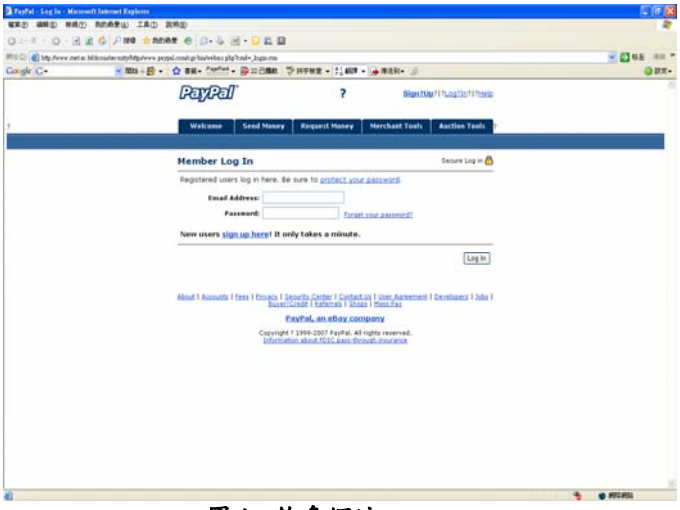
https://www.paypal.com/cgi-bin/webscr?cmd=_login-run



圖五 正當網站

釣魚網站URL:

http://www.ruet.ac.bd/icons/security/https/www.paypal.com/cgi-bin/webscr.php?cmd=_login-run

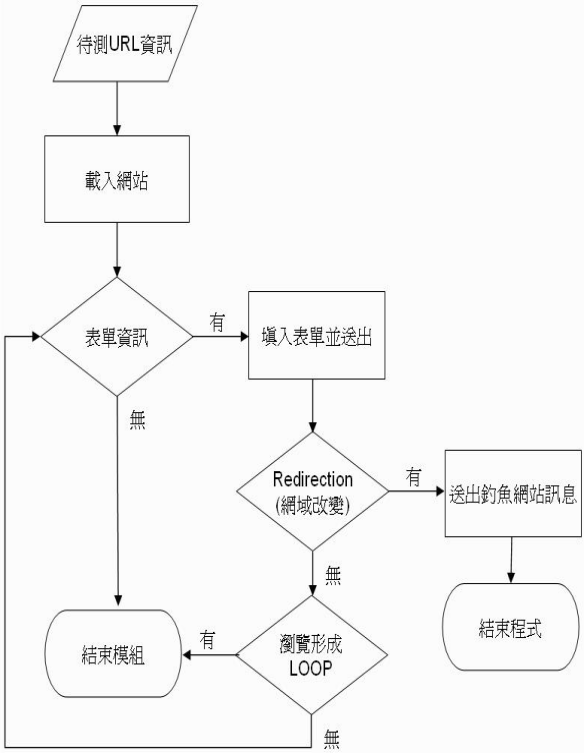


圖六 釣魚網站

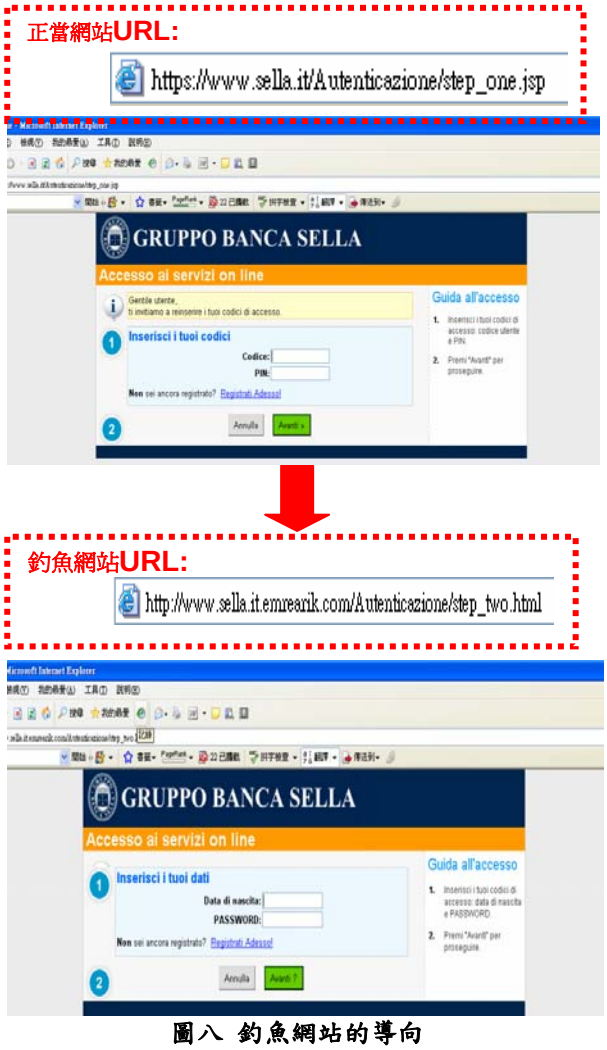
(4)Redirection Checker：(流程如圖七)

1.一般來說，大部分的釣魚網站在獲得資訊後便會將使用者轉向導至正當網站，如圖八，以避免使用者發現自己身處釣魚網站。

2. 檢查網頁是否有表單，若有則填入資訊並觸發送出表單事件，檢查 URL 是否有轉向，若有則為釣魚網站，若無則回到檢查表單。



圖七 Redirection 模組流程



圖八 釣魚網站的導向

4. 實驗測試

本實驗資料來源來自下述三個網站 APWG[6]、PhishTank[19]以及MillerSmiles[1]。實驗資料組成為隨機抽取 10 個釣魚網站及 10 個正當網站，將 URL 資訊輸入至程式，其中樣本不取在同一網域內之網站，因若網站在同一網域則其釣魚網站製作手法大都相似。

實驗步驟如下：(1)黑白名單為靜態設定，不另做實驗測試。(2)先單獨測試 Relativity Checker 以及 Redirection Checker。(3)再將上述兩個 component 聯合測試。

實驗結果如表三至表五。

表三 Relativity Checker Only

編號	1	2	3	4	5	平均
正確率	70%	60%	70%	60%	50%	62%
誤報率	0%	0%	0%	0%	0%	0%

表四 Redirection Checker Only

編號	1	2	3	4	5	平均
正確率	40%	60%	60%	50%	50%	52%
誤報率	0%	0%	0%	0%	0%	0%

表五 Both Component

編號	1	2	3	4	5	平均
正確率	70%	80%	80%	90%	90%	82%
誤報率	0%	0%	0%	0%	0%	0%

根據 Y. Zhang etc. [18]的實驗數據，80%左右的正確率已經是非常不錯的。此外將來若能再使用更多的釣魚網頁做實驗，或許能將系統更精確的偵測率測試出來。系統漏報的原因可能是因為釣魚網站使用 script 或是其他網頁語言直接載入正當網站時，就會造成系統漏報。

另外，對於網頁資訊輸入有嚴格的檢查，亦會造成系統無法偵測。

5. 結論與討論

本論文提出以 URL 為基礎資訊的釣魚網站偵測系統，而系統所需資訊不管是從 E-mail 或是使用搜尋引擎所得到的清單都只須包含 URL，對於資訊蒐集的簡易性，以及對於使用者的隱私權保護都提供了一個解決的方法。

也由於本文所提出的偵測系統只需要 URL 資訊，且偵測過程不會造成機器的太大的負擔，因此

不管是部署在伺服器端點或是使用者端點皆有相當不錯的功用性。對於郵件伺服器管理者而言，可以將本系統與垃圾信過濾系統結合，使得使用者可以免於垃圾郵件以及釣魚郵件的侵害，提高使用者使用網路服務的意願，而對於網路管理者，可以在得知釣魚網站清單後，對管轄區域做網站的阻擋，減少使用者上當的機會。而對使用者來說，本系統可以在不記錄個人資訊的狀況下，即可幫使用者過濾掉大部分的釣魚網站，讓使用者在網路上使用網路服務時可以更加的放心。

另外本文的偵測系統加入了偵測與自動填寫表單的功能，因此使得偵測釣魚網站的功能不再侷限於剛開始連結的畫面，而能達到網站的深度偵測，讓有隨後轉向(redirection)性質的釣魚網站更加的無所遁形。惟自動填寫表單的功能尚屬剛開發階段，對於一些特殊資訊(如信用卡帳號，身分證字號…等)或是網站對於輸入字串有做較為嚴格的檢查時，會使得偵測程式無法進入網頁下一步流程而無法進行完整偵測，因而降低程式偵測率。目前採取的折衷辦法為讓使用者能自行輸入相關字串，並監視網頁流程，來達到偵測的功能，因此日後研究或許可以針對自動填寫表單功能改進，或許搭配資料庫對照方式是個不錯的選擇。

此外，本論文目前只針對 HTML-Based 的網頁進行偵測，因此對於使用其他網頁程式攥寫的釣魚網站，或是使用 Proxy 方式載入正當網站的釣魚網站，目前並未能有效偵測。目前較常見的網頁程式還有其他如 java script、CGI、Flash 等多種語言，所以若能將系統擴展至能偵測其他程式語言構成之釣魚網站，相信必能防堵更多的釣魚攻擊，使得更多的使用者免於受害。

參考文獻

[1] MillerSmiles.co.uk!, http://www.millersmiles.co.uk/

[2] Anti-Phishing Working Group, “Phishing Attack Trends Report - May 2007”, http://www.antiphishing.org/reports/apwg_report_may_2007.pdf

[3] Gregg Keizer, “Phishing Costs Nearly $1 Billion**”,** TechWeb Technology News. http://www.techweb.com/wire/security/164902671

[4] Robert McMillan, “Gartner: Consumers to lose $2.8 billion to phishers in 2006”, NetworkWorld, 2006. http://www.networkworld.com/news/2006/110906-gartner-consumers-to-lose-28b.html

[5] APWG, “Origins of the Word "Phishing"”. http://www.antiphishing.org/word_phish.html

[6] Anti-Phishing Working Group, http://www.antiphishing.org/index.html

[7] Dhamija, R., J. D. Tygar. and M. Hearst. “Why phishing works”. CHI 2006, April 22-27, Montréal, Québec, Canada

[8] Steve Sheng, Bryant Magnien, Ponnurangam Kumaraguru,Alessandro Acquisti, Lorrie Faith Cranor, Jason Hong, Elizabeth Nunge, “Anti-Phishing Phil: The Design and Evaluation of a Game That Teaches People Not to Fall for Phish”, Symposium on Usable Privacy and Security (SOUPS) 2007, July 18-20, 2007, Pittsburgh, PA, USA.

[9] COMODO, “Anti-Phishing Portfolio”, Comodo Inc, 2005

[10] CVC (Content Verification Certificates), http://www.contentverification.com

[11] TrustLogo, http://www.trustlogo.com

[12] Min Wu, Robert C. Miller, Greg Little, “Web Wallet: Preventing Phishing Attacks by Revealing User Intentions”, Symposium On Usable Privacy and Security (SOUPS) 2006, July 12-14, 2006, Pittsburgh, PA, USA.

[13] Zhang, Y., J. Hong., and L. Cranor, “CANTINA: a Content-Based Approach to Detecting Phishing Websites”. In *Proceedings of the 16th International World Wide Web Conference (WWW2007)*, Banff, Alberta, Canada, May 8-12, 2007

[14] Chou, N., R. Ledesma, Y. Teraguchi, D. Boneh, and J.C. Mitchell. “Client-Side Defense against Web-Based Identity Theft”. In *Proceedings of The 11th Annual Network and Distributed System Security Symposium (NDSS '04)*.

[15] Vipul Ved Prakash, Christopher Abad, Jamie de Guerre. “Cloudmark's Unique Approach To Phishing”. Cloudmark, Inc. ,2006

[16] Liu Wenyin, Guanglin Huang, Liu Xiaoyue, Xiaotie Deng and Zhang Min, “Phishing Webpage Detection“. Proceedings of the 2005 Eight International Conference on Document Analysis and Recognition (ICDAR’05)

[17] Craig M. McRae, Rayford B. Vaughn, “Phighting the Phisher:Using Web Bugs and Honeytokens to Investigatethe Source of Phishing Attacks”. Proceedings of the 40th Annual Hawaii International Conference on System Sciences (HICSS'07)

[18] Yun Zhang, Serge Egelman, Lorrie Cranor, and Jason Hong, “Phinding Phish: Evaluating Anti-Phishing Tools”, In Proceedings of the 14th Annual Network and Distributed System Security Symposium (NDSS 2007), February 2007.

[19] PhishTank, http://www.phishtank.com/

[20] VeriSign, http://www.verisign.com