Communications of the CCISA
Vol. 29 No. 4 Nov. 2023

網路安全實作練習舞弊偵測機制之研究

魏銷志[1]、陳韋堯[2*]、陳建宏[3]
[1]國立臺北科技大學 資訊與財金管理系、[2]國立臺北科技大學 資訊與財金管理系、[3]國立臺北科技大學 創新學院資訊安全學位學程
[1]vickrey@ntut.edu.tw、[2]t111ab8027@ntut.org.tw、[3]t112c72101@ntut.org.tw

摘要

網路安全技能的提升需要不斷的實踐與磨練。然而，傳統的教學方法，僅透過作業和測驗，存在著一些限制。這些方式難以精確地呈現學習過程，也難以有效協助面臨困難的學生。同時，這些方式可能導致一些不當的舞弊行為，損害評分的公平性，進而影響整體教學效果並提高評量的難度。為克服這些問題，本研究將搶旗比賽的練習模式與舞弊偵測防制機制相結合，應用於實際的網路安全課程中。透過實際案例的驗證，我們評估了舞弊偵測防制機制的效益，同時運用數據建立機器模型，期望能夠提供更深入的教學指導，以促進未來的教育發展。

關鍵詞：網路安全、教育評測、CTF、舞弊偵測、舞弊防制

* 通訊作者 (Corresponding author.)

Regular Paper

Communications of the CCISA
Vol. 29 No. 4 Nov. 2023

The Study of Cheating Detection Mechanisms in Cybersecurity Practical Exercises

Yu-Chih Wei[1*], Wei-Yao Chen[2], Chien Hung Chen[3]
[1]Department of Information and Finance Management, National Taipei University of Technology, [2] Department of Information and Finance Management, National Taipei University of Technology, [3] Frontier Institute of Research for Science and Technology, National Taipei University of Technology
[1]vickrey@ntut.edu.tw, [2]t111ab8027@ntut.org.tw, [3]t112c72101@ntut.org.tw

Abstract

Achieving proficiency in network security requires thorough practice. However, traditional assignment and assessment methods in teaching have limitations, as they cannot accurately portray the learning process or assist struggling students. This can lead to inappropriate cheating behaviors, diminishing overall teaching effectiveness, and increasing assessment complexity. This study integrates CTF (Capture The Flag) practice models with anti-cheating mechanisms and applies them to network security courses. Through real-world cases, the efficacy of these anti-cheating measures is validated, and data is utilized to create a machine model, aiming to provide more insightful guidance for future teaching endeavors.

Keywords: Network security, Educational assessment, CTF, Cheating detection, Cheating prevention

壹、 前言

科技日新月異，此變革不僅帶來許多的機遇，同時各種組織也將面臨著更多不同的新興資訊安全威脅。這些威脅也將不斷地演進，使得專業資訊安全專責人員的需求變得更加迫切與重要。因此各界人士期望學校能夠成為培育資訊安全人員之搖籃，提供學生全面性的資訊安全領域之技術教育，從而讓學生們建立札實的資訊安全基本知識，以補足學與用之間的落差，讓學生就業時即可成為具備實力的資安專責人員之一。

尤其在 110 年 12 月 28 號由金融監督管理委員會所發布之「公開發行公司建立內部控制制度處理準則第 9 條之 1 規定之令(金管證審字第 11003656544 號)」內寫到，「上市櫃公司應配置適當人力資源及設備負責資訊安全制度之規劃、監控及執行資訊安全維護作業。」身處在不同的符合條件之下的上市櫃公司，需配置符合條件之專門負責資訊安全相關工作或職務人員，如配置資訊安全專責單位、資訊安全長、資訊安全專責主管及資訊安全專責人員。此法令帶動起資訊安全專責人力的需求，然而目前的現狀是，資訊安全人才的供應與需求之間存在著明顯的缺口。

在資訊安全相關課程之中，理論知識與實際應用的結合是非常重要的。首先透過教育資訊安全理論建構學生資安基礎觀念，再搭配上模擬真實環境的實際上機練習操作，讓學生可以在實作練習中去驗證理論，能夠更深入地理解資訊安全的概念，並學會如何應對不同的情境。在此同時嘗試激發學生們在學習資訊安全的熱忱與興趣，在課中安排具有有趣且具有挑戰性的實作練習與課後作業，我們期望學生在解決問題的過程中體會到學習的樂趣，培養學生資訊安全技術的基底，讓學生可以在資訊安全領域中從理論往實務發展。奪旗(Capture The Flag, CTF)競賽自 1996 年起至少已在資安界被用作測試實作攻擊性安全技能的方式[2]，透過題目的設定可以讓學習者在設計的題目裡進行解題，以提升學習時的樂趣，亦可設過題目的設計模擬實際環境的練習方式可有效為學生理解實際的資安威脅情況，進而去實施攻擊與防護練習。相較傳統考試需監考，CTF 線上練習因使用資訊系統及網路，學生可上網搜尋資源或工具用法，人工監考範圍較為狹隘，導致易引發舞弊，減少學習成效，且學生普遍也認為在此情境下更易舞弊[3]。

在於網路安全攻防演練中，即使是同一題目也可以使用不同的方式進行攻擊嘗試取得 Flag，沒有唯一的解法，學生固然可以透過嘗試不同的工具，或與他人不同的方式，取得 Flag 並獲得分數。然而，網路工具及攻擊方式繁多，不可能去限制學生們攻擊思維，所以會開放使用網路，供學生們在上網學習攻擊方式與工具使用方法。期盼學生能親自了解題目並熟悉操作網路工具，對於在課堂中實際演練而言，同學之間的代替操作、互相分享答案與互相分享做法是最容易讓學生忽視過程取得答案的幾種方式。不過上網查詢題目相關的資訊與同儕間互相分享做法之間有異曲同工之妙，難以判別其真實情況。目前鮮少研究針對 CTF 舞弊檢測，尤其線上測驗舞弊的文獻也有限。在線上考試中，舞弊風險增加，舞弊機制相對困難。

本研究沿續[1, 4]所建置的舞弊防制與偵測機制及機器學習方式，改進偵測與資料處

理方式以進行模型訓練，以提升在網路安全課程舞弊偵測的評估機制，以便更好的偵測發現舞弊行為。

貳、文獻探討

為了防止於 CTF 中 Flag 互相分享並提升學生自行解題之意願，於 J. Burket 等人[5]的研究中提出自動產生 CTF 題目(Automatic Problem Generation，APG)之研究，其研究最初是讓每個使用者都會收到相同題目，但每個團隊都會有一個獨立的 Flag，而後是可以自動產生難度接近的題目，以確保所有使用者的題目難度一致，主要目的都是為了在使用者之間不會發生分享 Flag 之情形。Liu 等人[6]透過使用攻擊行動圖（AAG）模型，協助在 CTF 中判斷受訓者的行為。透過建立 AAG，清晰呈現可預測和不可預測的攻擊方式，並建立攻擊方式與後果之間的聯繫。可快速查看 AAG 協助訓練者辨別受訓者的攻擊方式與階段，並發現意想不到的攻擊方式和其作弊行為。為了在 Web 應用程式 CTF 挑戰中檢測作弊行為，Chetwyn 和 Erddi [7]提出了威脅狩獵技術，用以分析玩家的行為並辨識異常情況。他們從目標服務中蒐集日誌，將其傳送至 Elasticsearch 伺服器，然後運用威脅搜尋方法來搜尋妥協指標 (IOC)。每個挑戰都需要完成特定的步驟，缺少這些步驟的前提，提交正確的 Flag 將被視為分享 Flag 的作弊行為。Minagar 和 Sakzad[8]提出了一個框架，用於在 IT 取證課程的 CTF 式評估中自動產生問題（APG）和標記。我們的主要焦點將放在虛擬硬碟（VHD）和資料包擷取（PCAP）檔案的隨機化上，這樣每位學生在評估中都會得到不同的取證材料，以防止作弊行為的發生。與自動產生題目類似，在 J.Vykopal 等人[9]的研究提出用於逆向工程的 CTF，為每個學生提供了一個獨特的二進制分析。另外，P. Matias 等人[10]提出一個非交互式零知識的 CTF 比賽平台，稱為 NIZKCTF，在該平台中不會儲存任何 Flag，因此當解完一題題目後並不需要提交 Flag，而是提出一個公開的零知識證明，讓平台方及其他解題使用者都可以查核及確認是否確實解題正確。

與傳統的人工監考相似的研究中，Y. Atoum 等人的研究[3]提出帶有視覺及聲音傳感器的全自動線上考試監考系統 OEP，需要使用兩個攝影機及一個麥克風，第一個攝影機需要面向考生；另一個需要由考生配戴至眼鏡上，進而捕捉到考生的視野；麥克風則捕捉空間內的任何聲音。使用以上設備進行檢測：從教科書、筆記及各種書籍中舞弊；與其他人聯繫；使用電腦或手機的網際網路；代考等舞弊情形。他們使用兩階段多媒體分析方式，第一階段側重於聲音及影像串流中提取特徵，主要包含六個基本組件：考生驗證、文本檢測、語音識別、活動視窗檢測、注視檢測及電話檢測，第二階段使用第一階段的 OEP 組件提取出時間特徵來執行所有組件的聯合決策，進而達到檢測舞弊行為。並且可以透過[11]提出基於臉部驗證的連續驗證考生方法，通過使用移動學習 m-learning 的線上課程中獲取圖像作為訓練資料集實施增量訓練，用以提高對姿勢及光照

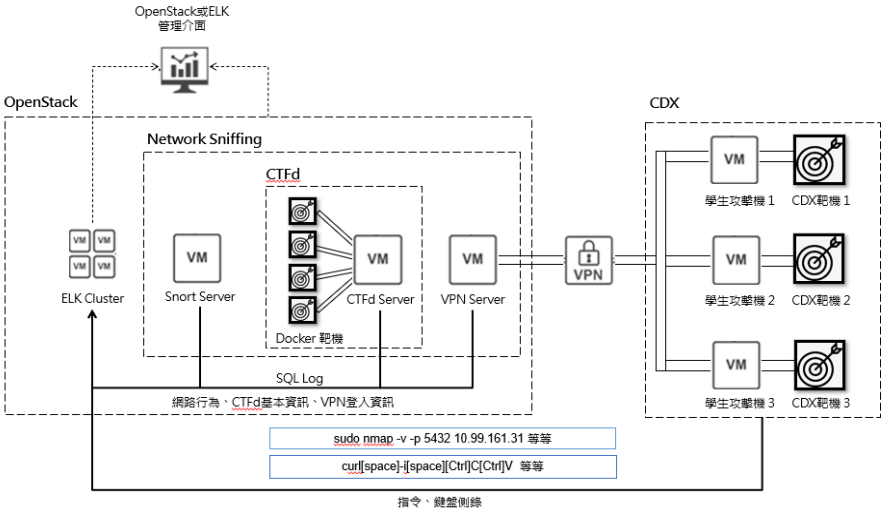
變化的強健性，進而提高需要透過攝影機執行舞弊檢測的效能。

然而，Duhaim 等人[12]認為使用攝影機記錄學生在考試的行為，侵犯學生的隱私，因此建構一個基於可靠資料集的線上考試檢測系統，首先使用每個學生的 IP 位址、花在考試上的時間及考試遲到的時間，接著使用重疊計算的相似度算法進行比對學生之間的答案，最後當學生的答案將會透過 k-means 演算法分組，最終會輸出可能舞弊的學生之回答。

與上述使用學生考試行為資料集相似，Asep 等人[13]分析虛擬學習環境 VLE 的紀錄，用以檢測學生線上考試時舞弊的證據。VLE 獲得的資料如：開始考試時間、作答完成時間、年級、考生參加測驗的 IP 位址、學生回答每題問題的時間及考試時間使用的資訊，並使用 py-Cheat 檢測可能透過合作完成考試的學生團體，因此額外考慮了學生的班級，該工具會輸出按順序參加考試的學生列表，即第二個學生在第一個學生完成前不會開始考試。其基於第二個學生完成作答時間比第一個學生來的少、第二個學生比第一個學生獲得更高的成績及他們來自同個班級，然而該工具僅適合非同步執行考試時才有意義。

參、研究方法

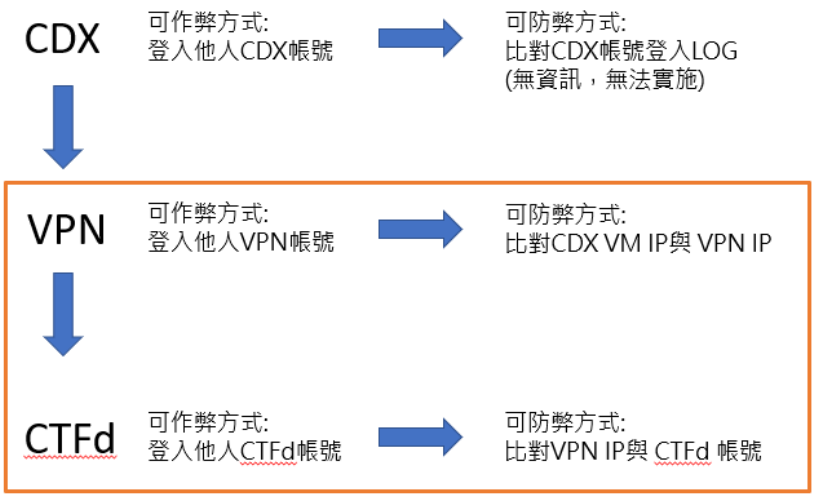
本計畫實作環境使用國家高速網路與計算中心研發之雲端資安攻防平臺(簡稱為 CDX)做為教學用戶端底層環境，使用 OpenVPN 連線至建置於 Openstack 的 CTFd 平台，並動態部署 VM 獨立靶機讓學生透過 CDX 平台連線進行攻擊，同時利用 Snort 網路流量收集工具蒐集學生網路攻擊流量與學生攻擊中的指令鍵盤記錄存放於所建置的 ELK，整體架構，如圖一。



圖一：網路安全課程練習舞弊偵測平台架構

3.1 登入流量紀錄比對

開源網路流量收集工具 Snort，通過監聽網路流量分析和檢測潛在的威脅和攻擊行為，主要用於入侵偵測和網路安全監控，並具有高度可自定義性的。本研究利用具有此特性的 Snort 蒐集 CDX 的虛擬機至 OpenStack 中 VPN Server 與 CTFd 中的所有網路流量紀錄以及參照 VPN 登入記錄，於 VPN 登入與 CTFd 登入進行即時的登入比對(如圖二所示)。隨後透過行為儀表板的查核，可以顯示學生是否存在代為操作的帳號登入行為，以便阻止進行此操作的情況，同時確保整個網路安全演練測驗的公平性。

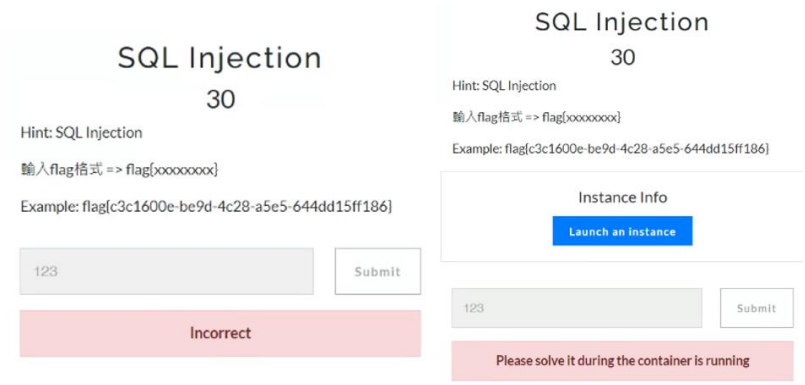


圖二：登入舞弊防制機制

3.2 基於 Docker Container 的獨立靶機環境

在許多網路安全攻防演練的題目上，希望學生可以熟悉一個作業系統的操作環境以及方式，面對這種題目，時常會額外架設一個作業系統環境，讓學生們進入此環境進行嘗試，然而共用同一環境方式容易造成環境的損壞以及效能的不堪負荷。本研究採用能提供低耗能、一致性、且個別獨立的 Docker Container 作為靶機環境，產生靶機時會對應不同的 Port，檢視指令的操作下，可以較容易看出是否有舞弊的疑慮，也避免掉共用同一環境所帶來的弊端。

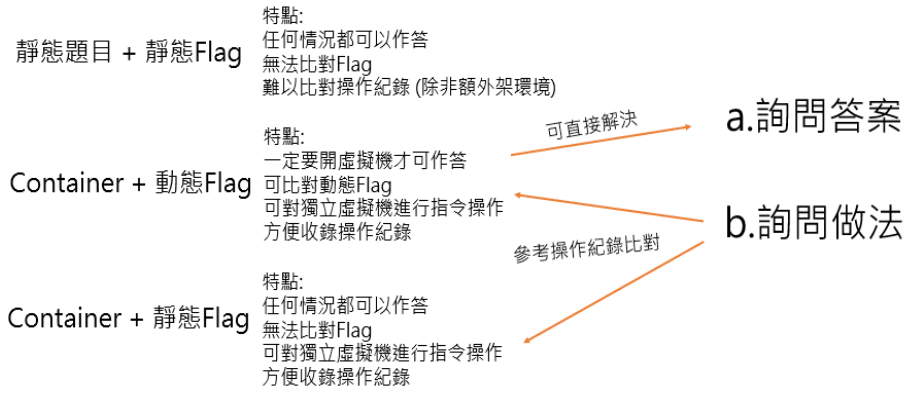
CTFd 傳統靜態展示題目的方式，讓學生們可以不斷切換查看不同的題目，並且解題自己想嘗試的題目，導致有看 A 題解 B 題的情況。在這種情況下，無法準確收錄到學生在嘗試解題哪個題目的流量資訊，學生們就完成了解題。透過 Container 的開啟與關閉的機制，可以更準確得知學生們正專注在哪些題目的解題且收錄操作紀錄。另外在設置啟用動態 Flag 的情況下，學生必須開啟 Container 才可作答，如圖三，能更準確收錄到學生操作資訊與加強舞弊。



圖三：傳統方式與Container方式比較圖

3.3 動態產生隨機 Flag

在平台建構所使用的 CTFd 中，答案的設置是固定不變的，雖然可以透過新增不同答案給與題目一些彈性，仍然是可以透過直接分享答案舞弊解題。但是在網路安全攻防演練中，最關注的是學生是否親自進行實際環境下的網路攻防練習，以提升其技能。為此透過 Docker Container 的獨立靶機機制，在每個靶機產生時也動態生成隨機的 Flag，並設置成當次開啟靶機的題目答案。這樣的設計有效地防止學生直接傳遞答案(如圖四所示)，從而阻止了舞弊行為的發生。



圖四：題目實行方式比較圖

3.4 答題過程紀錄

在網路安全攻防演練中，舞弊行為要透過學生從開啟題目開始至成功答題為止的行為特徵進行綜合評估，如作答時間、指令數量、鍵盤輸入內容等等，所以除了原本的指令紀錄以外，本研究使用網路流量蒐集工具、指令工具、鍵盤側錄工具對學生們操作時

的紀錄進行擷取，像是利用網路流量蒐集工具 Snort，側錄在攻擊機連線之 VPN 與整體練習環境中的 HTTP 網路流量，主要用以偵測網頁掃描工具的應用。也使用鍵盤側錄工具收錄學生們操作時的鍵盤輸入內容，偵測學生是否如實輸入指令或使用快捷鍵，如[Ctrl]+C、[Ctrl]+V 或[BackSpace]等按鍵，以及利用收錄學生攻擊機中的指令歷史，進而去觀察與推測學生在答題過程是否有詢問他人作法的可能性。

隨後透過虛擬機開啟與關閉的時間為基準，收錄所有答題過程記錄，快速將學生在作答時間內所有行為紀錄進行呈現與比較。方便在監考或評量成績時，可以查看學生答題的詳細情況進行判斷，以便了解學生的學習情況，也維護測驗的公平性。

3.5 使用機器學習的舞弊偵測機制

基於人為的舞弊檢測行為會耗時耗力，考慮到這一過程的複雜性和資料量的龐大，採用機器學習方法可以更加高效地處理資訊，辨識出潛在的舞弊行為。

本研究所使用的資料集是透過網路安全課程實踐教學取得的學生實際練習資料，該資料是在 2023 年 6 月 16 號早上 9 點至 2023 年 6 月 18 號下午 5 點中學生們所有於 CTFd 中實作 CTF 題目的答題紀錄，包含了網路行為紀錄、操作指令資訊、鍵盤側錄資訊以及答案提交紀錄，詳細欄位如表一至表四所示。

表一：網路行為紀錄

欄位名稱	欄位說明
cid	封包 ID
signature	種類
timestamp	時間
ip_src	IP 來源
Ip_dst	IP 目的
tcp_sport	TCP 來源 Port
tcp_dport	TCP 目的 Port
tcp_seq	TCP 序列號
tcp_ack	TCP 確認號
data_payload	TCP 封包資料

表二：鍵盤側錄紀錄

欄位名稱	欄位說明
keylog_id	鍵盤側錄紀錄 ID
eth0_ip	攻擊機 IP
vpn_ip	VPN IP
keylog_log_date	鍵盤側錄紀錄時間
keylog_log_data	鍵盤側錄紀錄資料

表三：操作指令紀錄

欄位名稱	欄位說明
cmd_id	指令紀錄 ID
eth0_ip	攻擊機 IP
vpn_ip	VPN IP
cmd_log_date	指令紀錄時間
cmd_log_data	指令紀錄資料

表四：答案提交紀錄

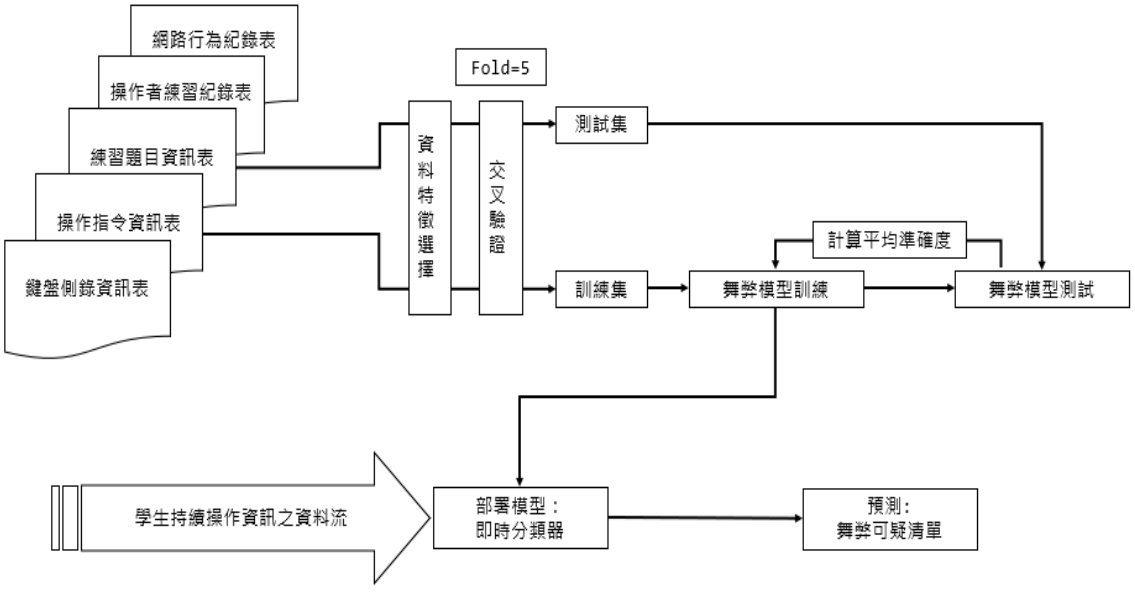
欄位名稱	欄位說明
submission_id	答案提交紀錄 ID
user	使用者帳號
challenge	題目
type	答案正確與否
provided	提交答案
date	答案提交紀錄時間

在資料前處理的部分首先說明如何將原始資料集進行標記資料、建立特徵值，標記資料的方式是由 207 筆答案提交紀錄當中共有 77 筆的成功答題資料取用紀錄，並使用上述章節提到之機制以判斷其過程是否作弊進行標記。於先前的研究[1]中，利用題目開啟的 Docker Container 虛擬機開啟至關閉的時間作為記錄之取用基準。然而，我們發現在真實的答題狀況中，作答者在成功答題之後大多不會去自主關閉虛擬機，預設自動關閉題目的時間為一小時，導致此判斷舞弊的時間區段被無故拉長收到無用的紀錄，進而影響到舞弊偵測系統判斷的真實性。因此本次研究改使用題目開啟之時間至成功答題為止的時間取用作答者的所有紀錄，其取用的相關資料筆數比較表如表五所示。

表五：不同時間區間內使用到的紀錄筆數比較

	提交紀錄	網路紀錄	指令資訊	鍵盤資訊
所有紀錄筆數	207	79,040	1,089	1,159
取至題目關閉筆數	77	28,935	474	656
取至成功答題筆數	77	26,813	453	640

本研究使用題目資訊、網路行為、指令、鍵盤側錄、作答行為等資訊去延伸出了共36個特徵值。並經過使用資訊獲益比率(Gain Ratio)作為特徵篩選主要評斷標準，使用10個特徵進行機器學習的模型訓練(如表六所示)。與先前的研究相比，本研究去除了答題總時間、成功答題記錄之外的答題時間、成功答題記錄之外的提交失敗數，三個特徵；新增輸入動態Flag與成功答題記錄之外的鍵盤側錄到Curl+V的次數，兩個特徵。期望訓練出一個舞弊模型供日後課程使用，為課程的監督和評估提供更可靠的依據。整體架構，如圖五所示。



圖五：使用機器學習的舞弊偵測機制架構圖

表六：機器學習進行舞弊偵測使用的10個特徵

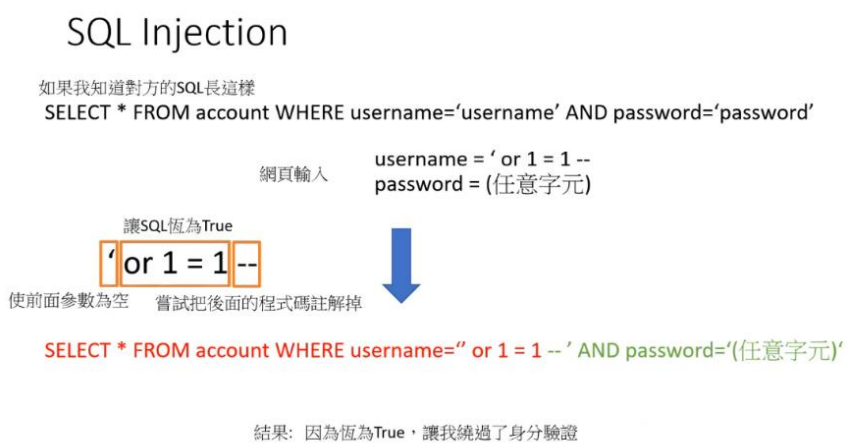
特徵	特徵說明
C_{key_cmd}	擁有關鍵指令的數量
C_{total_cmd}	指令的總數量
I_{dflag}	輸入動態 Flag
T_{suc_record}	成功答題紀錄之時間
C_{keylog_length}	鍵盤側錄字串長度
C_{keylog}	鍵盤側錄次數
$R_{challenge\_level}$	題目難度排名
C_{pre_cv}	成功答題記錄外的鍵盤側錄到 Curl+V 的次數
C_{cv}	鍵盤側錄到 Curl+V 的次數
$C_{bracket}$	使用特殊按鍵的次數

在機器學習的模型中，本研究將會使用邏輯斯回歸(Logistic Regression，LR)、k-近鄰演算法(k-NN)、支援向量機 (SVM)、貝氏分類器(Naïve Bayes，NB)、決策樹(Decision Tree，DT)、梯度提升樹(Gradient Boosting Tree，GBT)與隨機森林(Random Forests，RF)進行訓練及預測。

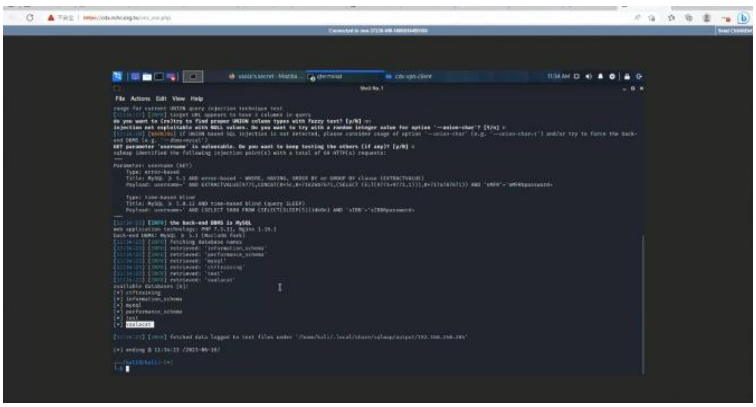
3.6 實驗設計

本研究實驗結合課程實踐教學，採取結合現場授課與課後線上測驗的方式進行，於現場先講解測驗規則以及平台操作與實施工具的教育訓練(如圖七所示)，並給予一些時間讓學生們進行測驗，已讓學生熟悉 CDX 環境(如圖六所示)，並方便觀察學生們的現場反應。

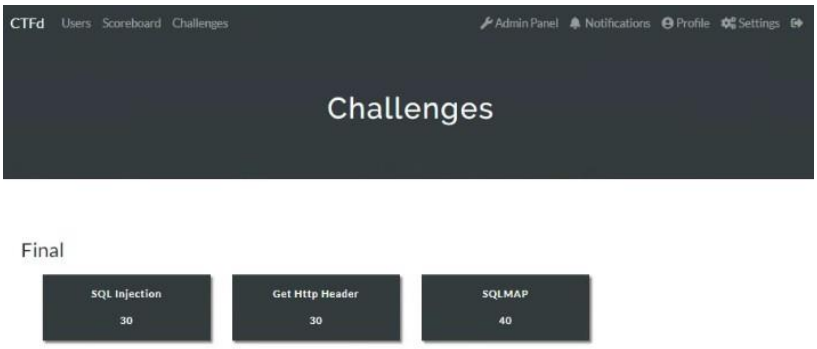
於課後規定測驗時間內，學生們可自行連線至平台進行網路安全攻防測驗(如圖八所示)。基於並無告訴學生們測驗設置著任何舞弊防制機制以及設置一個無法在課後去監控學生們的行為的測驗環境，藉由如此給予學生們舞弊以及可觀察舞弊的良好環境，期望可透過這種測驗設置，進而去驗證舞弊防制機制。



圖七：教育訓練教材



圖六：CDX練習畫面



圖八：搶旗競賽畫面

肆、研究結果

4.1 登入他人帳戶

在登入他人帳戶中的舞弊方式，由於本平台使用 VPN 連線與 CTFd 用於擴建 CDX 之雲端資安攻防平台的方式，透過 CDX 虛擬機 IP 與 VPN 連線的對應紀錄與 VPN IP 對於登入 CTFd 帳號的網路流量紀錄去做代為他人登入操作的舞弊偵測系統。在本次測驗中，並無學生刻意登入他人帳戶之代替本人操作的情況發生，不過卻有些學生有跟隨著助教的現場示範教學同步登入到助教的 VPN 的事情發生，如圖九所示，可以看到 CDX 與 VPN 登入之間的相關紀錄。

圖九：VPN與CTFd 登入紀錄

4.2 詢問答案

在詢問答案的舞弊方式，已透過結合 Docker Container 動態產生隨機且唯一的 Flag 的方式，防止了學生互相傳遞 Flag 成功答題的舞弊行為，也可以透過查看答案提交紀錄是否有人輸入相同的動態 Flag，而得知哪位學生有詢問答案的舞弊行為發生。如在此次研究中，從答題過程紀錄中發現有動態 Flag 相同之情況發生，進一步去調查更發現了實際有 4 位學生動態 Flag 都相同，如圖十所示。

在使用基於 Docker Container 的獨立靶機並未設置動態 Flag 的情況下，可以透過查看學生的答題過程記錄，去觀察學生們在答題上使用的指令內容，進而去判別學生的舞弊行為。在本研究中發現了幾位學生在未使用可取得 Flag 的關鍵指令的條件下(如圖十一所示)，持有著靜態 Flag 成功答題，表示他們是向其他學生問答案舞弊才成功解答的。

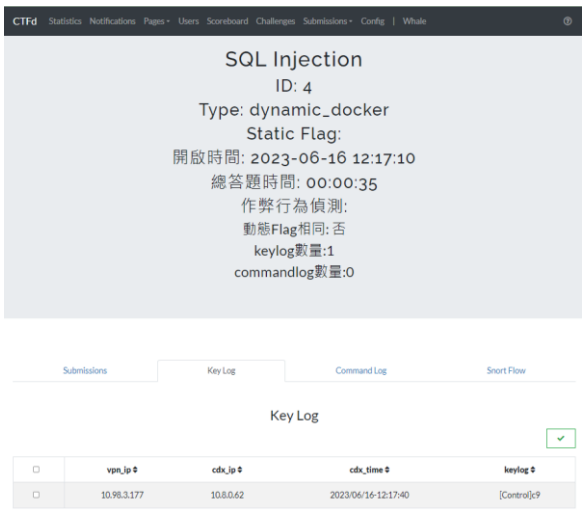
圖十：答題過程紀錄-動態Flag相同

圖十一：答題過程紀錄-指令查看

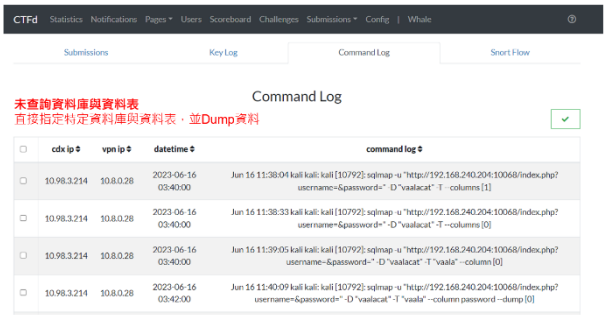
4.3 詢問做法

在於網路安全攻防演練的題目中，其解題過程有相依性，有需要實施哪些步驟後才能進行下一步的解題情況發生。從開啟題目到解題為止，至少需要有一定數量的指令與關鍵執行指令才可以解答的狀況發生。所以可以利用這種特性，觀察學生們的答題過程紀錄，查看學生們使用的指令是否有缺少某些步驟，去判別學生是否有舞弊行為。在本研究中，發現了幾個學生，直接使用了關鍵指令取得了 Flag 並成功答題，表示透過詢問了他人做法的舞弊行為，省略了許多步驟取得了資訊，直接利用了資訊與關鍵指令取得 Flag，如圖十二所示。

在其他判斷詢問做法的舞弊上，基於開放上網查詢解法的機制下，難以直接判斷學生是否舞弊，只能去透過答題紀錄觀察學生們的答題的行為，嘗試發現與正常答題過程相較之下不合理的案例去推斷是否為舞弊行為，如答題時間過短、鍵盤側錄內容大多為複製貼上、指令數量少等情況。因無法透過邏輯保證學生舞弊，故只找出舞弊的可疑名單。 如圖十三之案例所示，答題時間只有 35 秒，鍵盤側錄紀錄只有一條顯示 Ctrl+C，沒有指令紀錄，推測為向他人詢問 SQL Injection 攻擊方式，透過鍵盤 Ctrl+C 複製，並用滑鼠右鍵貼上。



圖十二：舞弊疑似案例



圖十三：答題過程紀錄-未有關鍵指令

4.4 機器學習

使用本次成功答題的過程紀錄作為機器學習模型的資料集，透過答題時間、指令和鍵盤側錄等行為作為特徵，經由 Gain Ratio 的特徵篩選出 10 個特徵(如表 1 所示)進行模型訓練，使用多種機器學習模型進行訓練，預測結果分為有舞弊與無舞弊。表現最好之機器語言模型為邏輯斯回歸，其次為 SVM，如表八。

於本次研究中表現最好的模型是使用邏輯斯回歸進行預測，先前表現最好的模型是 Naïve Bayes。兩者相較之下，在本次的研究中，主要是希望盡可能多地捕捉到所有異常情況，Recall 指標可以衡量實際正例中被模型成功預測為正例的比例。從上次研究之 Recall 結果 0.877 提升至本次 0.891，此提升可以增加我們在預測出所有舞弊案例的表現。在異常偵測中，資料集時常是保持著不平衡之情況，MCC 指標能夠處理類別不平衡的情況下提供一個相對平衡的評估分數，從上次研究之 MCC 結果 0.755 提升至本次 0.791，此提升可以看出相較於上次研究之模型，本次舞弊偵測模型的性能表現更加突出。

表七：研究[1]之機器學習演算法比較結果

Model	AUC	CA	F1	Prec	Recall	MCC
Naive Bayes	0.920	0.877	0.877	0.878	0.877	0.755
SVM	0.856	0.815	0.815	0.820	0.815	0.636
Logistic Regression	0.858	0.800	0.799	0.808	0.800	0.608
Random Forest	0.829	0.785	0.785	0.785	0.785	0.569
Gradient Boosting	0.772	0.785	0.784	0.785	0.785	0.570

表八：本研究之機器學習演算法比較表

Model	AUC	CA	F1	Prec	Recall	MCC
Logistic Regression	0.896	0.891	0.890	0.900	0.891	0.791
SVM	0.876	0.875	0.875	0.876	0.875	0.751
Naive Bayes	0.882	0.828	0.828	0.828	0.828	0.657
Random Forest	0.810	0.797	0.797	0.797	0.797	0.594
K-NN	0.827	0.797	0.797	0.797	0.797	0.594

伍、結論

本研究所建置之舞弊偵測防制機制能蒐集學生作答數據、及時追蹤學生解題過程記錄，動態偵測學生答題情形與舞弊行為，以便依照學生作答情況適時調整教材及教學方式，加強學生表現較差的內容，也能更全面的評估學生的學習狀況給予分數。因此本研究致力於解決上述問題，沿用[1, 4]之架構與資料集，重新定義題目作答至結束的時間區段進而去重新抓取答題紀錄，並將此資料進行訓練，以提升了機器學習模型反應實際作答情況的準確度，用以建置一個適合不同種類的題目都能使用的舞弊偵測模型。與[1]的機器學習評估效果相比，提升了 MCC 的分數，代表能夠更好進行真實舞弊結果的之偵測。

此外，在依據[1]所建立的結構下，建立 CTF 題目的成本相對較高。因此，我們計劃在未來的研究中專注探討這個情況，期望開發一個能夠自動生成 CTF 題目的系統。這個系統將根據不同的參數和標準，在每次生成的題目中保持相同的解題邏輯，以更有條理的方式鼓勵學生僅分享解題邏輯，減少對直接分享指令的依賴，同時提供學生更多的思考機會。

[誌謝]
本研究接受國科會之研究計畫編號「NSTC 112- 2221-E-027-067-」及教育部教學實踐研究計畫 「PSK1110207」部份補助。

參考文獻

[1] 魏銪志 and 陳韋堯, "網路安全課程練習舞弊偵測機制之研究," presented at the TANET & NCS 2023 臺灣網際網路研討會暨全國計算機會議, 2023.

[2] K. Chung and J. Cohen, "Learning Obstacles in the Capture The Flag Model," presented at the 2014 USENIX Summit on Gaming, Games, and Gamification in Security

Education (3GSE 14), 2014/8, 2014. [Online]. Available: https://www.usenix.org/conference/3gse14/summit-program/presentation/chung.

[3] Y. Atoum, L. Chen, A. X. Liu, S. D. Hsu, and X. J. I. T. o. M. Liu, "Automated online exam proctoring," vol. 19, no. 7, pp. 1609-1624, 2017.

[4] 魏銪志 and 陳韋堯, "網路安全演練舞弊偵測之研究," presented at the 第三十三屆全國資訊安全會議(CISC 2023), 2023.

[5] J. Burket, P. Chapman, T. Becker, C. Ganas, and D. Brumley, "Automatic problem generation for capture-the-flag competitions," in *2015 {USENIX} Summit on Gaming, Games, and Gamification in Security Education (3GSE 15)*, 2015.

[6] Z. Liu, H. Qiu, J. Zhu, and Z. Zeng, "AAG: A Model for Attack Behavior Judgment in CTF-style Cyber Security Training," in *2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS)*, 2019: IEEE, pp. 1-5.

[7] R. A. Chetwyn, L. J. P. o. t. t. C. Erdődi, and ESAR, "Cheat Detection In Cyber Security Capture The Flag Games-An Automated Cyber Threat Hunting Approach," p. 175, 2021.

[8] S. Minagar and A. Sakzad, "Automatic Problem Generation for CTF-Style Assessments in IT Forensics Courses," in *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1*, 2023, pp. 229-235.

[9] J. Vykopal, V. Švábenský, and E.-C. Chang, "Benefits and pitfalls of using capture the flag games in university courses," in *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, 2020, pp. 752-758.

[10] P. Matias, P. Barbosa, T. N. Cardoso, D. M. Campos, D. F. J. I. S. Aranha, and Privacy, "NIZKCTF: a noninteractive zero-knowledge capture-the-flag platform," vol. 16, no. 6, pp. 42-51, 2018.

[11] H. S. Asep and Y. Bandung, "A design of continuous user verification for online exam proctoring on M-Learning," in *2019 International Conference on Electrical Engineering and Informatics (ICEEI)*, 2019: IEEE, pp. 284-289.

[12] A. M. Duhaim, S. O. Al-Mamory, and M. S. Mahdi, "Cheating Detection in Online Exams During Covid-19 Pandemic Using Data Mining Techniques," EasyChair, 2516-2314, 2021.

[13] A. Balderas and J. A. Caballero-Hernández, "Analysis of Learning Records to Detect Student Cheating on Online Exams: Case Study during COVID-19 Pandemic," in *Eighth International Conference on Technological Ecosystems for Enhancing Multiculturality*, 2020, pp. 752-757.