校園資訊系統滲透測試及資訊安全奪旗(CTF)競賽網站架設

王德譽、楊皓宇、黃智謙、羅一誠
朝陽科技大學資訊工程系
413 台中市霧峰區吉峰東路 168 號
Tel:(04)23323000 ext. 4538
Email: dywang@csie.cyut.edu.tw

摘要

網路時代人們做甚麼事情幾乎都會經過網路，沒有資安觀念的設計師所設計的網站，往往因沒有對資料進行防護，讓駭客有機會入侵，導致使用者的資料外洩。本研究爲了增進校內資訊系統的安全與展示基本駭客攻擊手法，對校內多個網站進行測試，找尋設計師所忽略漏洞，提供漏洞的數據給設計師參考，以利設計師快速修補漏洞，防護網站資訊之安全。除了滲透測試提供防護數據供設計師參考外，本研究還架設了一個答題網站，網站內提供了許多資安方面的題目，供使用者與設計師了解一些基本的資安問題，學習到一些駭客常攻擊手法，使設計師在設計網頁時可以依據學習到的手法，在架設網頁時除了滿足功能外，兼顧基本的資訊安全問題。
關鍵詞：資訊安全、滲透測試、*CTF*

Abstract

In this era of the Internet, people almost always do something through the Internet. If there is no concept of information security designed website designers often no protection for data, and then the opportunity for hackers cause a user's data leakage, while allowing users to information circulated on the Internet. And the topic in order to enhance security and demonstrate the basic school system hacker attack techniques, on-campus testing multiple sites, looking for designers ignored vulnerabilities, the vulnerability of data provided to the designers, according to the data available quickly fix vulnerabilities designer based, can make the site reach safer. In addition to providing data to the designer, the topic also set up a website A pen, within the site provides a number of information security-related topics, for users and designers to understand some basic information security problems, hackers often learn some methods of attack so that designers can design web based learning to approach when setting up web pages can achieve security.
Keywords：Information Security, Penetration Test, CTF

1 前言

1.1 研究背景

資訊安全基本設計目標，可以分類爲保密性、完整性以及可用性。保密性的目的是防止未經許可的人或系統去存取資料或訊息；完整性的目的是確保資料是正確的或沒有遭人竄改，讓使用者使用正確的資料；可用性的目的是確保資料隨時可以使用，因爲無法使用等於沒有資料可以閱讀。

隨著電腦科技的發展，人們依賴網路的程度也是日以劇增，只要是現代人幾乎都脫離不了網路的世界，然而在網路的世界是任何人都能使用的，許多資料都會在網路間流竄，爲了保護資料的安全，人們開始想方法去保護他們自己的資料，避免遭有心人士竊取，但只要有防守必有攻擊，所以要如何讓有心人士的攻擊傷害降至最低，這就是我們目前需要深入了解的問題。

在網路的世界裡，沒有絕對的防禦機制，任何的防禦機制必有漏洞，只是漏洞的嚴重性的差異。在考慮到使用者的方便以及安全性，資訊安全是一種需要在安全與便利中進行取捨，讓使用者太過便利容易導致安全上的漏洞，但是如果防禦機制太過嚴謹又會讓使用者感到不便。如何讓使用者不會感到不便，又不會讓資料容易被竊取，但如何去設計防衛機制避免他人進行攻擊而遭入侵，首先要先了解他人是如何攻擊，以及他們的思想，才能對於駭

客的攻擊手法而加以防範。

本研究主要方向爲網頁安全與系統安全的多種漏洞原理及防禦方法，並且設計一個目前駭客流行的CTF [1]比賽，可以在裡面練習到駭客的攻擊手法，並學習防範駭客的攻擊。

1.2 研究目的

爲了讓網路世界更加安全，應該學習如何當一個資安人員，爲了讓資安人員了解網站安全漏洞，所以本研究以學校校內網站進行駭客的攻擊手法演示，以這些基本攻擊手法的演練，讓網站架設或程式設計人員了解如何架設安全網站及設計安全資訊系統，除了突顯資訊安全的重要性，更提供相關人員修補漏洞的參考。

以學校資訊系統做爲攻擊手法演示目標，目的是爲了提升學校校內網路及資訊安全。沒有資訊安全觀念的程式設計師所設計的網頁，出現問題時往往會造成嚴重的傷害，藉由學校校內資訊系統的滲透測試，提升校內資訊人員的資安觀念及技術。也由於校內資訊系統資訊內容與全校所有教職員及學生息息相關，更能讓校內無論是不是資訊人員都能體會到資訊系統安全之重要性及遭到攻擊的嚴重性，提升大家對資訊安全之重視。

另外，本研究也設計了一個攻擊演練網頁，以存取控制、密碼學以及安全架構等方面設計網頁與題目，攻擊演練網頁是目前駭客流行的CTF [1]網頁比賽，在網頁裡可以去模擬各式各樣的的安全性問題，讓網站架設或程式設計人員可以模擬駭客是如何進行攻擊，並展示漏洞的嚴重性，進而以這些技巧進行資訊安全漏洞的防護。

2 滲透測試

2.1 網站安全

網站安全[2]是指出於防止網站受到駭客入侵並對其網站進行掛木馬或篡改網頁等行爲而做出一系列的防禦工作。由於沒有資安觀念的網站開發者，往往只考慮滿足用户應用及如何實現業務。很少考慮網站應用開發過程中所存在的漏洞，這些漏洞在不關注安全程式設計的人員眼裡幾乎不可見的，大多數網站設計開發者、網站維護人員對網站攻防技術的了解甚少；根據OWASP TOP 10 - 2013 [3]，它包含了網站架設中最常見、最危險的十大安全隱憂。這些漏洞一旦被利用，便可讓駭客們長驅直入的進入目標主機，並不受防火牆限制。

在測試學校資訊系統時發現最多的問題是 Sql Injection [4]、任意檔案下載上傳 [5]和XSS [6]攻擊。

Sql Injection

Sql Injection 是利用注入 sql 語法查詢到不應該被查到的資訊如程式 1。

```
1 <?php
$sql = "SELECT * FROM users where
    username = '$_POST[user]' and
    password= '$_POST[passwd]'";
3 $result = mysql_query($sql);
$row = mysql_fetch_row($result);
5 ?>
```

程式 1: 潛藏 SQL Injection 漏洞的程式碼。

若將 POST 出去的資料設成 user=admin'or 1=1#&passwd=123，則 SQL 查詢語法就會是

```
SELECT * FROM users where username
    ='admin' or 1=1#' and password
     = '123'
```

因爲#在 MySQL 中是註解，所以後面的查詢語句都會被省略，故眞正查詢的語句就會變成

```
SELECT * FROM users where username
    ='admin' or 1=1
```

以這樣方法便能直接登入admin帳號且不需要密碼。

Cross-site Script

Cross-site Script (XSS) 是針對前端所做的攻擊，若網頁上存在著 XSS 漏洞，則可藉著使用者去執行進而劫持使用者的 cookie。反射型 XSS 是最常用，使用最廣的一種方式。它通過給別人發送帶有惡意腳本代碼參數的 URL，當 URL 地址被打開時，特有的惡意代碼參數被 HTML 解析、執行。它的特點是非持久化，必須用户點擊帶有特定參數的鏈接才能引起。例如：某網址爲 www.xxx.xxx/index.php?title=<script>alert(document.cookie)</script> 如圖 1，透過這樣便能讓使用者執行 JavaScript，若網站上有登入資訊並儲存在 cookie 中，駭客便可利用劫持 (hijack) 方式去取得用户的登入的 cookie 值，如圖 2。

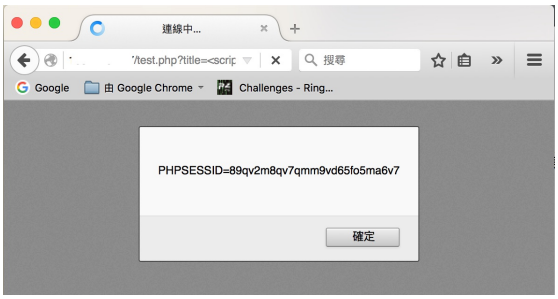

圖 1: XSS示範-1

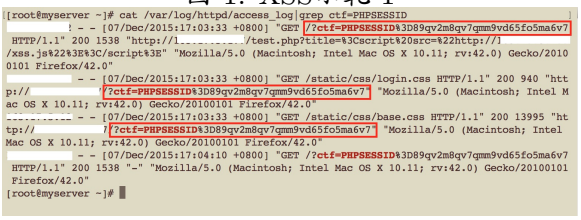

圖 2: XSS 示範-2

任意檔案上傳和下載

當網頁在下載檔案時使用 get 及 post 參數指定檔案名稱時，一旦沒有嚴謹的過濾，會造成任意檔案下載，洩漏系統敏感資料，如圖 3。網頁有上傳功能時，沒有完善的檔案過濾及限制，駭客便可以此上傳一個惡意檔案到系統中。

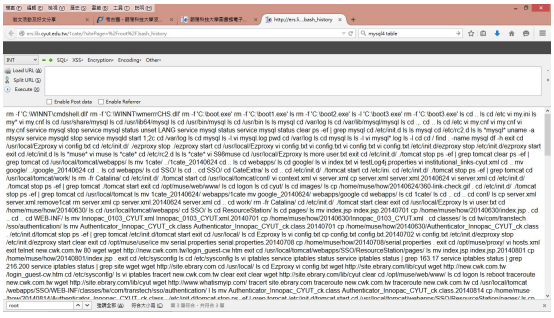

圖 3: 任意檔案下載示範

敏感訊息洩漏

敏感訊息洩漏 (Sensitive Data Exposure)[7]最常見的是應該加密的數據不進行加密。在使用加密的情況下，常見的問題是不安全的密鑰生成和管理和使用弱算法是很普遍的，特別是使用弱的哈希(hash)算法來保護密碼。加密演算法是安全防護的最後一道防線，當駭客取得了帳號密碼，可以簡單地使用一些破解軟體甚至線上服務進行破解。例如Cain & Abel，MD5 Reverse Lookup 等。例如：密碼數據庫使用未加鹽 (unsalted) 的哈希 (hash) 算法去存儲每個人的密碼。一個文件上傳漏洞使黑客能夠獲取密碼文件。所有這些未加鹽 (unsalted) 哈希 (hash) 的密碼通過彩虹表暴力破解方式破解。

2.2 程式和系統安全

程式及系統安全是指在一個系統、程式或服務在執行的期間，發生開發人員意料之外的行爲，嚴重的程度可能會被取得系統控制權、竊取機密資料、阻斷服務等。在系統上，許多管理者爲了方便，常常將權限都設定成 777，若發生問題則所有檔案都將被破壞。另外，許多伺服器的系統核心或服務過於老舊，並沒有按時更新，例如：HeartBleed(CVE-2014-0160) 或 ShellShock(CVE-2014-6271)，導致駭客可以輕鬆入侵，並透過已知的核心漏洞輕鬆拿到最高權限 root，因此資訊人員必須定期檢查更新，若有重大安全漏洞發佈，一定要立即將系統更新防止此漏洞被有心人士利用。

3 實例測試

會計系統注入問題

會計系統的查詢功能沒有過濾特殊符號，導致 Sql Injection 問題，可先利用 order by 確定資料表中有多少欄位，假設 order by 3 出現錯誤、order by 2 正確，則可以知道這資料欄位共有 2 個，再構造 union 查詢來洩漏出資料庫內容，所構造出的攻擊參數大致如下：

```
key_srh=aaa' union select null
    ,...,null,null,null#
```

並且可以挖出全校教職員的姓名、住址、身分證字號和銀行帳號等，如圖 4。

```
Input Search uid:dywang
姓名：王德譽
生日：05211**
身分證字號：R1217*****
地址：南投縣草屯鎮**********之 7 號
銀行帳號：0220**********1847
```

圖 4: Sql Injection 資料洩漏

MyCyut 安全配置及敏感資料洩漏

MyCyut 系統權限爲 root 且在下載頁面發現漏洞，將參數僞造後如下：

```
filename=shadow&path=/etc/
```

這樣便可下載任意檔案，並且在某個 log 檔中存在著明文帳號密碼，如圖 5。

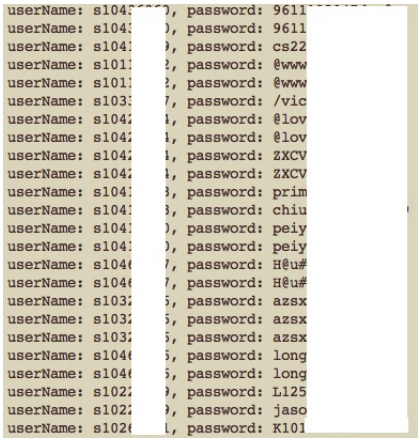

圖 5: MyCyut 敏感資料洩漏

MyCyut XSS 攻擊

MyCyut 系統中，待辦事項存在 XSS 漏洞，所構造出的攻擊參數大致如下：

```
... <script> ... src="http://xxx.xxx?xss="+document.cookies;</script> ...
```

若有用户連進這頁面，cookie 則會被駭客盜取，如圖 6。

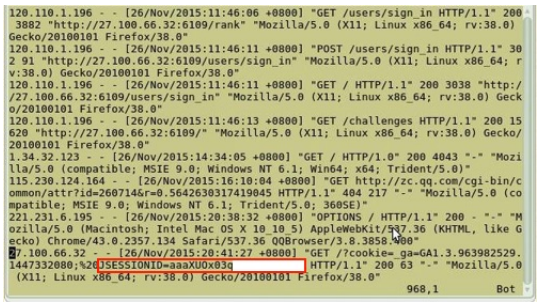

圖 6: MyCyut XSS 攻擊

學生和教職員資訊系統

這兩個系統皆有使用不需密碼的弱驗證登入方式，導致有其資料的使用者皆可登入他人資訊系統，如圖 7。

圖 7: 未授權登入系統

資通系網

資通系系網在全部公告處 mes_id 參數沒有做過濾，故能取得所需的資訊。先使用 order by 確定資料有多少欄位，再透過 MySQL 的函數 user() 得知 MySQL 的使用者，構造出的攻擊參數如下：

```
mes_id=-1 union select null,...,null,null,null#
```

因資料庫使用 root 來做連線，root 在資料庫中擁有所有權限，所以可以使用 MySql 中的 into outfile 寫入後門。由於此伺服器並沒有將目錄鎖住，因此可以瀏覽尋找可上傳目錄寫入後門，最後構造出的攻擊參數如下：執行後便能在目錄下創建 WebShell，如圖 8。

```
mes_id=-1 union select null,'<?php system($_POST[cmd]); ?>',... into outfile '...'#
```

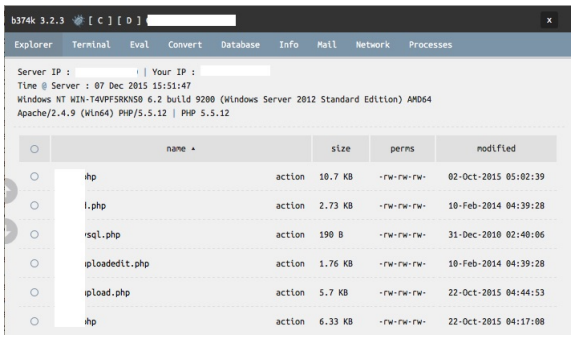

圖 8: 資通系 Sql Injection

資管系網

在資管系網頁下，發現一個能觸發 SQL Injection 的子網頁，洩漏出有用資料後拿到了使用者帳號和未加密的密碼，所構造出的參數大致如下：

```
Pname=a'union select null,...,user,password,null from ...--
```

登入後發現有一個地方可以做檔案上傳，因爲這裡並沒有做上傳檔案的限制，所以可以直接丟一個WebShell上去，並且可以成功執行，如圖 9。

體育室

體育室網站內的安全配置錯誤，全部檔案皆使用 777，導致所有檔案皆可被更改，如

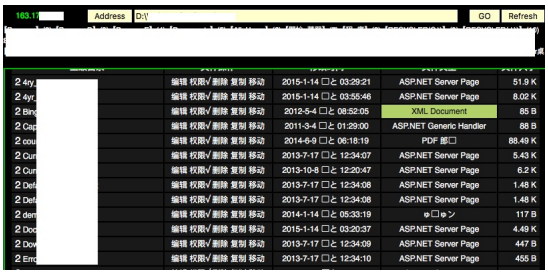

圖 9: 資管系任意上傳

圖 10。

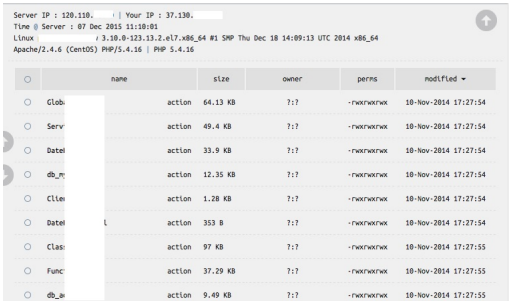

圖 10: 體育室安全配置錯誤

學生生涯系統

學生生涯系統的某頁面錯誤訊息沒有關閉，導致敏感訊息洩漏，如圖 11。

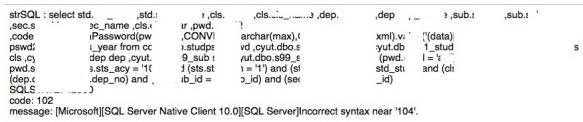

圖 11: 錯誤訊息畫面

4 資安奪旗競賽 CTF 網站架設

4.1 設計原理

CTF 網站使用 Ruby on Rails(RoR) 開發，RoR 是一個熱門的網頁開發框架，特色是能夠快速開發網站，使用 MVC(model-veiw-controller) 系統，model 負責與資料庫溝通，view 負責將界面顯示再使用者的瀏覽器，controller 則分析使用者的 HTTP Request，並轉交給 model 或 view 做下一步處理，由於邏輯跟使用者界面的分離，在程式碼的維護上也更加容易，架構如圖 12。

網站主體大致分成首頁、題目、排名、登入及註冊，如圖 13和圖 14，首頁簡單的設計一個靜態頁面說明這個網站的規則。題目頁面放置了各類型題目，一旦使用者送出答案(flag)，會發送一個 POST 請求給相同頁

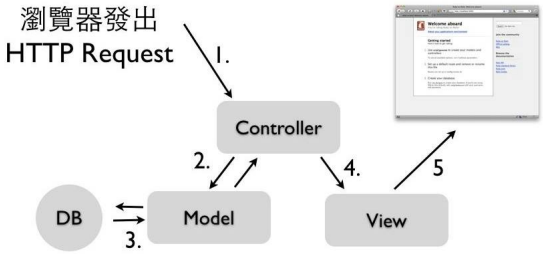

圖 12: MVC架構

面，會先經由 controller 處理，判斷所答題目是否正確及重複答題與否，回傳一個狀態值給 view，如果答對，將會更新資料庫上的資訊，包含分數及答題狀況。排名頁面將資料庫中的所有使用者資訊取出後依照分數排序並且以 10 個爲限作分頁，讓版面更簡潔。

登入及註冊頁面採用 RoR 熱門的會員管理套件 devise，預設只註冊信箱及密碼，所以需要手動加入姓名欄位，並且在註冊頁面的 controller 中加入使用者的分數及題目管理的初始化。

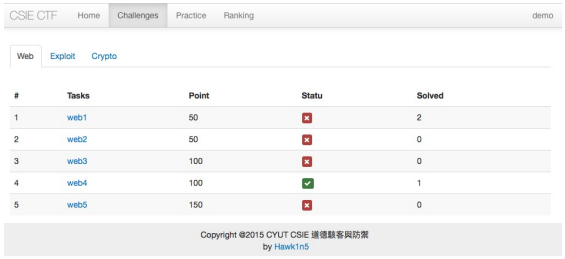

圖 13: CTF 挑戰頁面

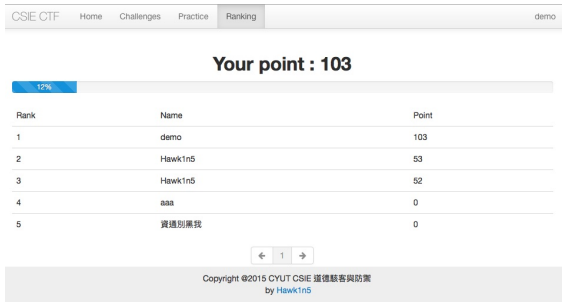

圖 14: CTF 排名頁面

4.2 題目設計

CTF 網站的題目設計以學校資訊系統常出現的漏洞爲主軸，設計相關的模擬題目，以期達到最佳的訓練效果。

不安全的函數及不安全的語法

使用 php 有問題的函數加上有問題的 if 語法判斷，來達成繞過條件。例如程

式 2。strcmp 傳遞 2 個 string 型別參數，兩者相同會得到 0 值，兩者不同得到非 0 值，但如果使用者將一陣列傳遞入 strcmp 中將會引發錯誤，而繞過條件判斷。

```
int strcmp ( string $str1 , string
    $str2 )

Returns < 0 if str1 is less than
   str2; > 0 if str1 is greater
   than str2, and 0 if they are
   equal.
```

程式 2: strcmp說明

資料庫注入(Sql injection)

資料庫注入是 OWASP TOP 10 中最嚴重的網頁應用程式安全問題，本系統設計一個登入頁面，會要求輸入帳號及密碼，並直接向資料庫做查詢的動作，由於沒有過濾使用者的輸入，因此可以插入任何的查詢語法，但只簡單的設計出需要以 Admin 的帳號登入即可過關。

Cookie 冒用

OWASP TOP 10 中排行第二的身份認證問題，也是常遇見的網頁安全之一，一般網頁在使用者登入後會使用 cookie 或 session 做身份認證，確保用户持續在網頁上活動，一旦駭客利用一些網頁上的漏洞截取到其他使用者的 cookie，便可冒用該使用者在網頁上活動。所以本系統設計一個會員管理的系統，再登入後會給予一份 cookie，但是該 cookie 只使用簡單的加密，在解開加密的方法之後便可輕易地使用 Admin 權限登入該系統。

格式化字符串 (format string)

格式化字符串在程式語言中是非常普遍的功能，主要發生原因事開發人員不當使用格式化函數，例如：printf、sprintf 等，一旦受到駭客的攻擊，嚴重可以取得主機的控制權。因此本系統設計一支程式，讓使用者輸入一個字串，直接用 printf 輸出，由於只使用一個參數，當輸入字串中帶有格式化字串，就會引發格式化字串漏洞。

命令注入 (command injection)

命令注入也是發生在網頁或軟體使用外部命令時，被惡意的加入額外的系統指令或程式碼，本系統設計一支程式執行 system 函數並由使用者輸入其參數，進而理解該攻擊的原理。

5 結論

隨著網路科技的發達，人們依賴網路的程度日與俱增，幾乎做任何事都跟網路密不可分，安全觀念越來越重要，但許多網頁卻沒有安全的觀念，使黑客有機會入侵到內部，發生資料外洩的嚴重問題。

爲了提供本校資訊系統安全，本研究主要分成兩大部分，第一部分爲以 OWASP TOP 10 中提到的安全漏洞，對學校資訊系統進行滲透測試，除突顯網站架設者及程式設計者若沒有資訊安全觀念，造成資訊外洩的嚴重性，並提供滲透測試數據供資訊人員修補漏洞參考。第二部分則將校內資訊系統經常出現的安全漏洞，設計成 CTF 網站，除供資訊人員自我檢測駭客攻擊能力，做爲架設或設計資訊系統時之參考，也可做爲資訊相關科系學生訓練用，以提升資訊人員的資安觀念與技術。

參考文獻

[1] What is ctf. https://ctftime.org/ctf-wtf/.

[2] 網站安全. http://baike.baidu.com/view/2962427.htm.

[3] Owasp top 10 - 2013. https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project#tab=OWASP_Top_10_for_2013.

[4] Sql資料隱碼攻擊. http://en.wikipedia.org/wiki/SQL_injection.

[5] Unrestricted file upload. https://www.owasp.org/index.php/Unrestricted_File_Upload.

[6] Cross-site scripting. https://www.owasp.org/index.php/Cross-site_Scripting_(XSS).

[7] Security misconfiguration. https://www.owasp.org/index.php/Top_10_2013-A5-Security_Misconfiguration.