網頁防火牆

屏東大學資管系教授

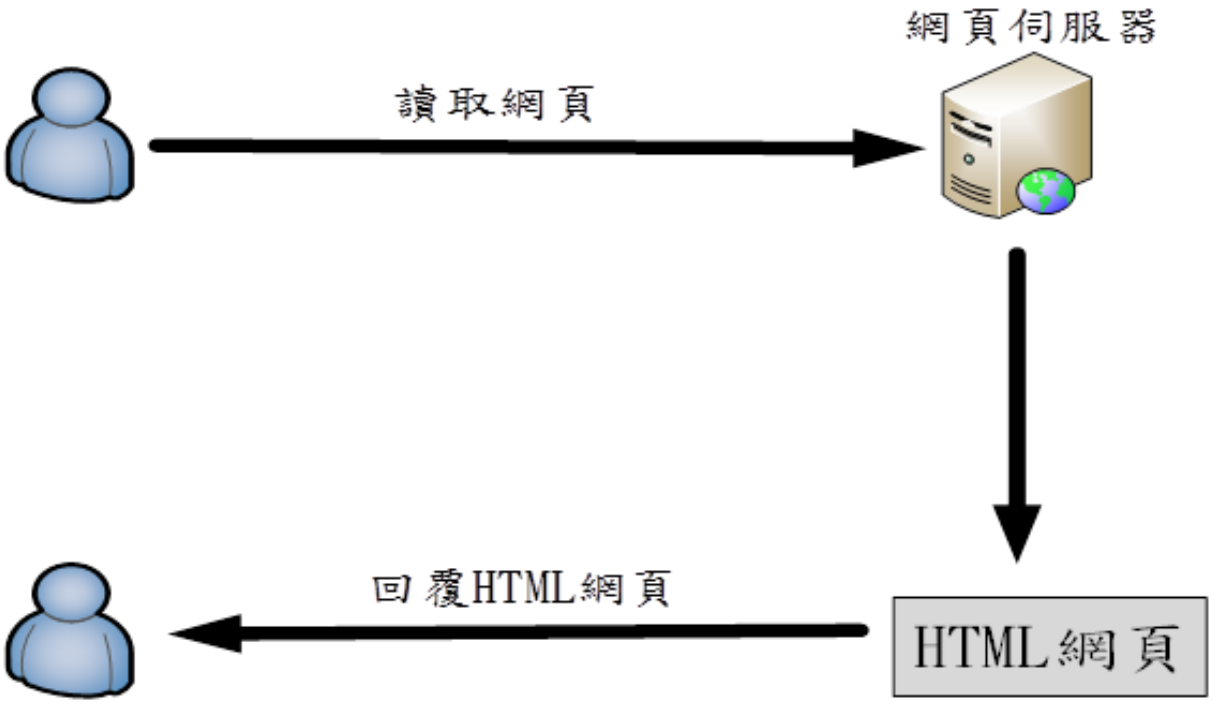
陳俊麟

前言

網頁程式帶動了便利，同樣的也帶來了安全的疑慮，網頁程式的安全疑慮通常為設計師的疏失抑或是經驗不足所致，而其解決方法通常需要將網頁程式重新一一檢視並修正，但這種方式是沒有效率的，因此網路應用程式防火牆(Application Firewall, WAF)應運而生，它的功能和防火牆一樣，但主要是用於網頁程式的保護上。

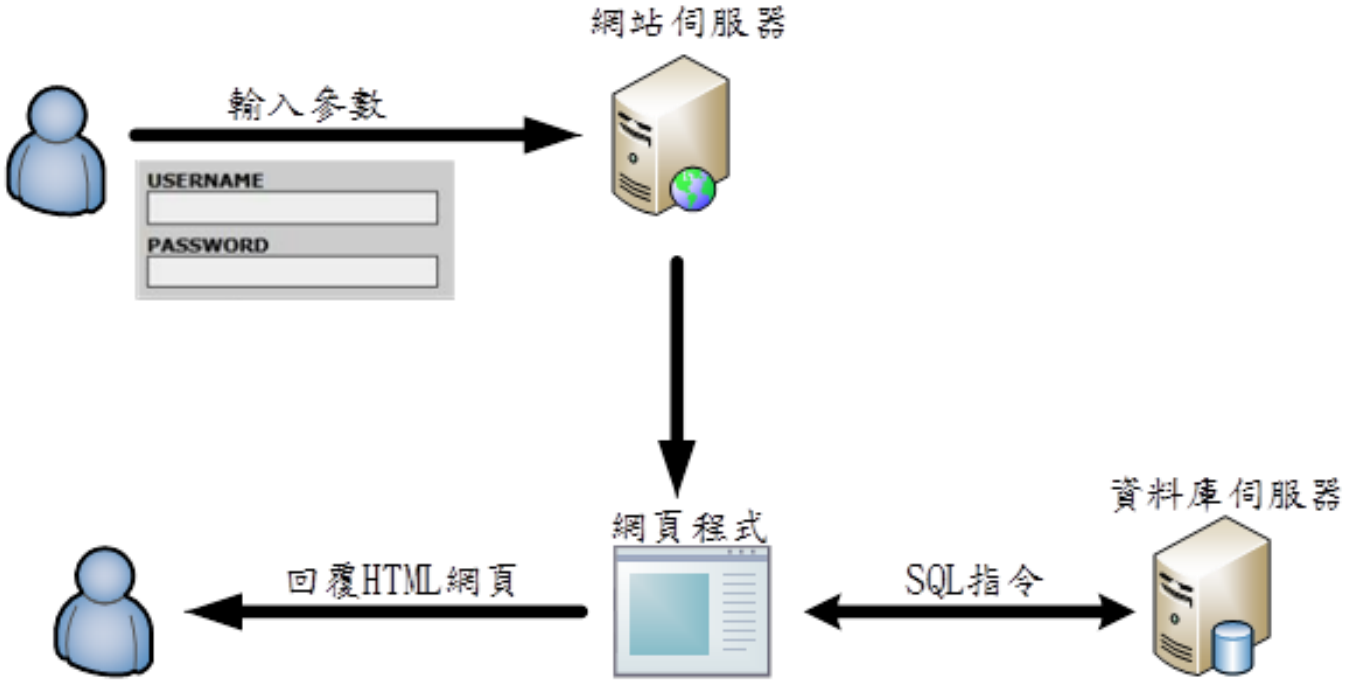
網頁類型-靜態網頁

網頁程式一般分為前端及後端，前端所使用的語言為HTML，最早是被用來統一描述文件的表示方式，下圖中的網頁伺服器非必要使用，可利用瀏覽器代替之，而這樣僅可公告訊息無法跟使用者有所互動的網站，稱之為「靜態網頁」，如下圖所示。



網頁類型-動態網頁

而目前店商所使用的方式為下述方式做資料連接，當使用者輸入參數(如下圖的帳號、密碼)與網頁程式互動，由網頁程式到資料庫伺服器抓取對應資料後，再動態組成HTML內容回傳到使用者的瀏覽器中，這樣與資訊庫功能結合的方式我們稱之為「動態網頁」。



2013 OWASP TOP 10

Injection(注入攻擊)-1

注入式攻擊肇因不是在於系統，而是在於程式設計師在撰寫網頁程式時，並未對使用者輸入的參數值進行適當的驗證或是過濾，以至於惡意使用者可利用惡意的輸入值(sql指令、script碼)，即可對系統產生危害，因為對系統而言即使輸入惡意字元依然可以正常的通訊交換，所以一般資安設備無法偵測出此類攻擊。

Injection(注入攻擊)-2

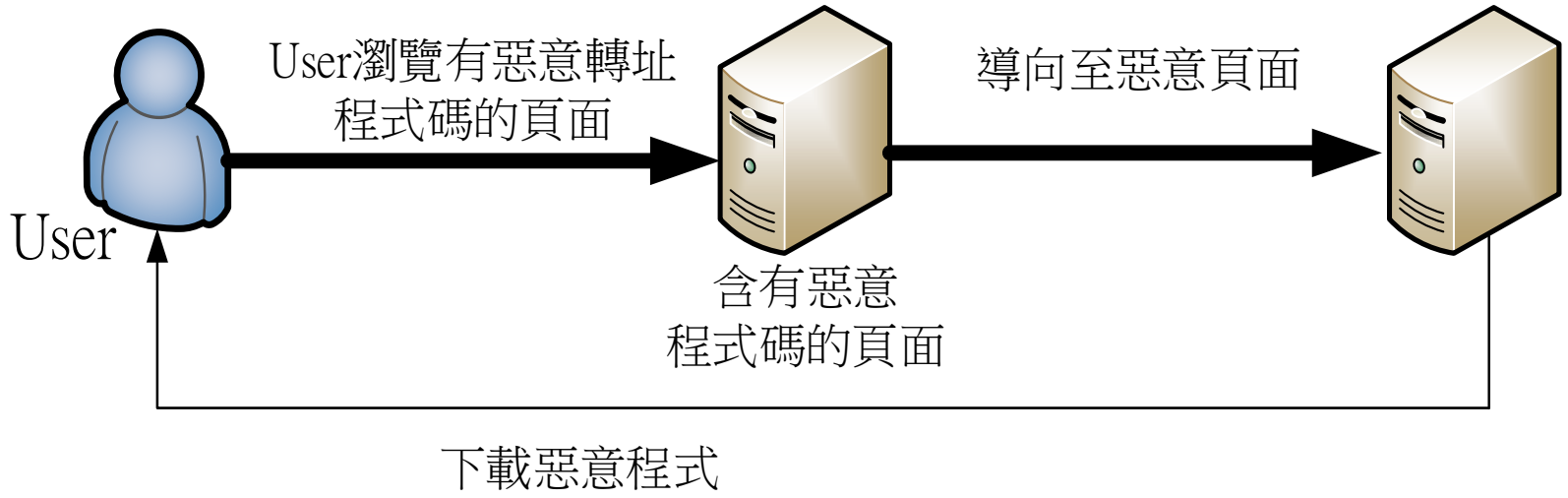
SQL injection(又稱為隱碼攻擊)、command injection(命令注入攻擊)等被稱為Injection的代表手法，其中SQL injection最具危害性，以下說明SQL injection的手法: 一個有登入功能的網站，都會需要輸入使用者的帳號與密碼來進行驗證，而後端程式在接收使用者所傳入的帳號密碼做處理時它會忠實的處理使用者輸入的資料，要是這時，使用者在帳號欄位中輸入有特殊字元的帳號: ' or 1=1 /* ，密碼:任意值，的話此時SQL語法就會變成select * from account where account='' or 1=1 /*' and password=''，而在MySQL中/*為註解的意思，所以在/*後面的字串通通沒有執行，而判斷式1=1永遠成立，此時駭客就能通過身分驗證登入成功。*MsSQL註解為--

Broken authentication and session management(無效身分認證及連線管理漏洞)

許多應用程式都需要經常需要處理身分認證及Session管理，若是Session並未設定適當的離線時間，以至於當前使用者登入使用後使用身分會一直停留於前一位使用者，後續的使用者可以一直延續使用前一位使用者的資料，以及cookies檔案若未加以適當的加密處理、Session導入方式不正確反而也會使得駭客容易取得密碼、Session令牌等私人資訊，除上述之外密碼強度不足、登入時沒有進行加密保護也會發生此項漏洞。

Cross-site scripting (XSS，跨網站腳本攻擊)-1

和SQL injection發生的原因相同，都是因為程式開發者未嚴格限制使用者輸入與未過濾特殊字串，讓惡意的Script得以在使用者的瀏覽器上執行，但SQL injection主要是造成資料庫的危害，而XSS攻擊主要是在造成使用者瀏覽網站上的危害。



Cross-site scripting (XSS，跨網站腳本攻擊)-2

以下為XSS惡意轉址Script舉例:

1.<ifram name=”test1” width=0 height=0 src=”惡意攻擊所在位置” ></iframe>

(iframe的長寬均設為0，所以攻擊者並不會於網站上看到iframe的存在因此會讓使用者對其失去警戒心)

2.使用<script src src=”http://3b3.org/c.js></script> 於瀏覽器執行的同時也執行這串惡意程式碼，這兩種方法藉由加入惡意的程式碼讓使用者啟動後，會在使用者不知情的情況下下載惡意程式碼造成電腦的損害。

Insecure Direct Object References (不安全的物件參考)

此攻擊是起因於未對使用者所輸入之參數值進行驗證進而造成的網路安全漏洞。以下舉例說明:假設攻擊者的帳號為1234，登入後的網址為https://www.cdewebsite.com/user?account=1234，若此網站沒有對帳號權限進行驗證，攻擊者可修改網址中 ?account=4567就能輕易窺探他人資料。

Security Misconfiguration (不安全的安全組態設定)

意指某些網站、裝置、作業系統等的預設帳戶、密碼、路徑，針對使用者在安裝時忽略安裝、使用手冊的提醒，未將欲設值重新設定，攻擊者利用此弱點進行安全攻擊。

Sensitive Data Exposure (敏感資訊外洩)

敏感性資料像是帳號、密碼、個資等，當這些資訊沒有透過適當的加密保護，就會增加再傳輸過程中或在系統內部被無適當權限的使用這存取，一般常見的例子為網頁程式採用http通訊協定傳送資料，在資料傳輸時未使用加密(明碼)方式傳送，惡意攻擊者可以在任何節點中，利用Sniffer(竊聽)方式來取得傳輸的資料。

Missing Function Level Access Control (不安全的功能控管)

網頁程式具有超連結的特性，所以惡意攻擊者想要控制整個系統的方式就更多元，也不如應用程式只有單一進入點，不利於防守。

Cross Site Request Forgery (跨網站冒名請求, CSRF)

可視為廣義的跨網站攻擊(X.S.S)，但CSRF通常為使用者登入的情況下。例如:當你在瀏覽一篇文章時，惡意攻擊者已經將惡意程式放在的按鈕中，當你點擊按鈕就會觸發程式自動登出你所使用的帳號，更嚴重者將會洩漏密碼、個人資料或金錢等。

using Known vulnerable components (使用有已知安全漏洞的模組或元件)

隨著軟體開發的進步，大多數的程式功能都能做成元件或函數，以便開發者在撰寫時可以參考使用，當這些元件具有某些安全漏洞，一旦開發者使用此元件，所開發出來的軟體也將繼承它的漏洞。

UnvalidatedRedirects and Forwards (未驗證的網頁重新導向)

又稱為轉址漏洞，在網站中有許多的超連結，或是利用觸發程序將使用者重新導向其他網站或前往其他頁面，如果未進行參數驗證，可能將使用者倒入惡意網站中，甚至下載到惡意攻擊的程式碼。

Mod_security實作

安裝mod_security模組所需程式

dnf install libapr*

```
已安裝:
  libapreq2-2.13-33.fc29.x86_64                libapreq2-devel-2.13-33.fc29.x86_64
  apr-devel-1.6.5-1.fc29.x86_64                apr-util-devel-1.6.1-8.fc29.x86_64
  cyrus-sasl-devel-2.1.27-0.3rc7.fc29.x86_64   expat-devel-2.2.6-1.fc29.x86_64
  httpd-devel-2.4.34-8.fc29.x86_64             libapreq2-libs-2.13-33.fc29.x86_64
  libdb-devel-5.3.28-33.fc29.x86_64            openldap-devel-2.4.46-8.fc29.x86_64

完成！
```

dnf -y install pcre*

```
已安裝:
  pcre-doc-8.43-1.fc29.noarch              pcre-devel-8.42-4.fc29.x86_64
  pcre-static-8.42-4.fc29.x86_64           pcre-tools-8.42-4.fc29.x86_64
  pcre2-devel-10.32-3.fc29.x86_64          pcre2-static-10.32-3.fc29.x86_64
  pcre2-tools-10.32-3.fc29.x86_64          pcre-cpp-8.42-4.fc29.x86_64
  pcre-utf16-8.42-4.fc29.x86_64            pcre-utf32-8.42-4.fc29.x86_64
  pcre2-utf32-10.32-3.fc29.x86_64

完成！
```

安裝mod_security模組所需程式

dnf -y install libxml2*

```
已安裝:
  libxml2-static-2.9.8-5.fc29.x86_64          libxml2-devel-2.9.8-4.fc29.x86_64
  cmake-filesystem-3.12.1-1.fc29.x86_64       xz-devel-5.2.4-3.fc29.x86_64
  zlib-devel-1.2.11-14.fc29.x86_64

完成！
```

dnf -y install apr*

```
已安裝:
  apr-api-docs-1.6.5-1.fc29.noarch
  apr-util-ldap-1.6.1-8.fc29.x86_64
  apr-util-mysql-1.6.1-8.fc29.x86_64
  apr-util-odbc-1.6.1-8.fc29.x86_64
  apr-util-pgsql-1.6.1-8.fc29.x86_64
  apr-util-sqlite-1.6.1-8.fc29.x86_64
  apricots-0.2.6-23.fc29.x86_64
  apron-0.9.11-22.1104.svn20180624.fc29.x86_64
  apron-devel-0.9.11-22.1104.svn20180624.fc29.x86_64
  aprsd-2.2.5-15.6.fc29.19.x86_64
  aprsdigi-3.5.1-12.fc29.x86_64
  mariadb-connector-c-3.0.9-1.fc29.x86_64
  mariadb-connector-c-config-3.0.9-1.fc29.noarch
  postgresql-libs-10.7-1.fc29.x86_64
  unixODBC-2.3.7-2.fc29.x86_64
  ax25-tools-1.0.3-8.fc29.x86_64
  freealut-1.1.0-26.fc29.x86_64
  gmp-c++-1:6.1.2-8.fc29.x86_64
  gmp-devel-1:6.1.2-8.fc29.x86_64
  libax25-1.0.5-7.fc29.x86_64
  mpfr-devel-3.1.6-2.fc29.x86_64
  openal-soft-1.18.2-6.fc29.x86_64
  ppl-1.2-8.fc29.x86_64

完成！
```

安裝mod_security模組所需程式

dnf -y install libtool

```
已安裝:
  libtool-2.4.6-27.fc29.x86_64                  autoconf-2.69-28.fc29.noarch
  automake-1.16.1-5.fc29.noarch                 m4-1.4.18-9.fc29.x86_64
  perl-Thread-Queue-3.13-1.fc29.noarch

完成！
```

dnf -y install httpd

```
[root@localhost ~]# dnf -y install httpd
上次中介資料過期檢查：0:06:08 以前，時間點為 西元2019年03月10日 (週日) 15時21分57秒。
已安裝軟體包 httpd-2.4.34-8.fc29.x86_64
依賴關係解析完畢。
無事可做。
完成！
```

安裝mod_security模組所需程式

dnf -y install php

```
已安裝:
  php-7.2.15-1.fc29.x86_64                    php-fpm-7.2.15-1.fc29.x86_64
  nginx-filesystem-1:1.14.1-2.fc29.noarch     php-cli-7.2.15-1.fc29.x86_64
  php-common-7.2.15-1.fc29.x86_64

完成！
```

dnf -y install mod_security

```
已安裝:
  mod_security-2.9.2-6.fc29.x86_64

完成！
[root@localhost ~]#
```

編輯testing.conf

cd /etc/httpd/modsecurity.d

ls

cd activated_rules

ls

vi testing.conf

```
[root@localhost ~]# cd /etc/httpd/modsecurity.d
[root@localhost modsecurity.d]# ls
activated_rules  local_rules
[root@localhost modsecurity.d]# cd activated_rules
[root@localhost activated_rules]# ls
[root@localhost activated_rules]# vi testing.conf
```

編輯testing.conf

在修改testing.conf前須先#ifconfig指令找到ip位置

```
[root@localhost activated_rules]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::9083:89e5:b2:39ae  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:a9:7e:1c  txqueuelen 1000  (Ethernet)
        RX packets 342635  bytes 289374177 (275.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 152848  bytes 9680233 (9.2 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

再進入testing.conf後打 i 編輯以下文字

SecRuleEngine On

SecServerSignature "Microsoft-IIS/5.0"

輸入完成後:wq 結束編輯並儲存

修改 httpd.conf

- vi /etc/httpd/conf/httpd.conf

…

server's response header

ServerTokens OS

service httpd restart

(出現Redirecting to /bin/systemctl restart httpd.service表示成功)

```
[root@localhost activated_rules]# service httpd restart
Redirecting to /bin/systemctl restart httpd.service
```

測試mod_security功能

將系統中有httpd的過程顯示

ps aux | grep httpd

(如果有紅字表示成功)

```
[root@localhost activated_rules]# ps aux | grep httpd
root      6960  1.0  0.3  44100 16972 ?        Ss   15:49   0:00 /usr/sbin/httpd -DFORE
GROUND
apache    6966  0.0  0.1  54344  8340 ?        S    15:50   0:00 /usr/sbin/httpd -DFORE
GROUND
apache    6968  0.0  0.2 1243212 10420 ?       Sl   15:50   0:00 /usr/sbin/httpd -DFORE
GROUND
apache    6969  0.0  0.2 1112076 10424 ?       Sl   15:50   0:00 /usr/sbin/httpd -DFORE
GROUND
apache    6970  0.0  0.2 1112076 10420 ?       Sl   15:50   0:00 /usr/sbin/httpd -DFORE
GROUND
root      7208  0.0  0.0 213216   824 pts/0    S+   15:50   0:00 grep --color=auto http
d
```

檢測連接埠的服務/版本資訊

dnf -y install nmap

```
已安裝：
  nmap-2:7.70-4.fc29.x86_64

完成！
```

nmap -sV 127.0.0.1 (找到 Microsoft-IIS/5.0")

```
[root@localhost activated_rules]# nmap -sV 127.0.0.1

Starting Nmap 7.40 ( https://nmap.org ) at 2018-10-23 14:48 CST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000070s latency).
Not shown: 998 closed ports
PORT    STATE SERVICE VERSION
80/tcp  open  http    Microsoft IIS httpd 5.0
631/tcp open  ipp     CUPS 2.2
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.56 seconds
```

再進入testing.conf後打 i 編輯以下文字

SecRuleEngine On

#SecServerSignature "Microsoft-IIS/5.0"

SecRule REMOTE_ADDR "127.0.0.1" "id:60009, redirect:http://www.example.com "

輸入完成後:wq 結束編輯並儲存

service httpd restart

(出現Redirecting to /bin/systemctl restart httpd.service表示成功)

再進入testing.conf後打 i 編輯以下文字

SecRuleEngine On

#SecServerSignature "Microsoft-IIS/5.0"

#SecRule REMOTE_ADDR "127.0.0.1" "id:60009, redirect:http://www.example.com"

SecRule REMOTE_ADDR "127.0.0.1" "id:60008, deny"

輸入完成後:wq 結束編輯並儲存

service httpd restart

(出現Redirecting to /bin/systemctl restart httpd.service表示成功)

再進入testing.conf後打 i 編輯以下文字

SecRuleEngine On

#SecServerSignature "Microsoft-IIS/5.0"

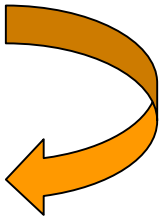
#SecRule REMOTE_ADDR "10.0.2.15" "id:60009, redirect:http://www.example.com”

#SecRule REMOTE_ADDR "10.0.2.x" "id:60008, deny”

SecContentInjection On

SecRule RESPONSE_CONTENT_TYPE ^text/html

"id:60010,phase:3,nolog,pass,prepend:'<script>alert(\"you are hacker\")</script>'"



打成一行

輸入完成後:wq 結束編輯並儲存

service httpd restart

- 開啟瀏覽器，輸入本身IP位址/隔壁IP位址