換個角度看資安
以滲透測試進行安全檢測

什麼是安全檢測

安全檢測是資訊安全中
用於保證安全性的做法

保證

人生不能重來！電腦可以！

砍掉重練才是王道！

砍掉重練才是王道！(大誤

保證的目的不是保證不出事
而是提高安全的確信度

安全檢測的做法可以有

安全檢測的做法可以有
安全稽核

安全檢測的做法可以有

安全稽核

弱點掃描

安全檢測的做法可以有

安全稽核

弱點掃描

滲透測試

安全稽核是由管理面進行查核

稽核的依據是**要求**與**準則**

要求具有強制性，所以必須遵守，否則就是缺失

準則屬非強制性，因此為參考性質，通常僅是建議

管理稽核通常只會

管理稽核通常只會
用眼睛看

管理稽核通常只會

用眼睛看

用嘴巴問

管理稽核通常只會

用眼睛看

用嘴巴問

用耳朵聽

管理稽核通常只會

用眼睛看

用嘴巴問

用耳朵聽

不會動手...

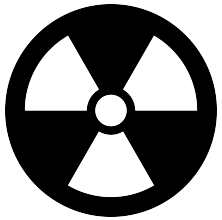
安全檢測的做法可以有

安全稽核

弱點掃描

滲透測試

弱點掃描用於找出已知漏洞



什麼是漏洞？

設定不當

```
127     add_auth_entry("\x43\x46\x4F\x4B\x4C", "\x43\x46\x4F\x4B\x4C", 7);                                  // admin    admin
128     add_auth_entry("\x50\x4D\x4D\x56", "\x1A\x1A\x1A\x1A\x1A\x1A", 6);                                  // root     888888
129     add_auth_entry("\x50\x4D\x4D\x56", "\x5A\x4F\x4A\x46\x4B\x52\x41", 5);                              // root     xmhdipc
130     add_auth_entry("\x50\x4D\x4D\x56", "\x46\x47\x44\x43\x57\x4E\x56", 5);                              // root     default
131     add_auth_entry("\x50\x4D\x4D\x56", "\x48\x57\x43\x4C\x56\x47\x41\x4A", 5);                          // root     juantech
132     add_auth_entry("\x50\x4D\x4D\x56", "\x13\x10\x11\x16\x17\x14", 5);                                  // root     123456
133     add_auth_entry("\x50\x4D\x4D\x56", "\x17\x16\x11\x10\x13", 5);                                      // root     54321
134     add_auth_entry("\x51\x57\x52\x52\x4D\x50\x56", "\x51\x57\x52\x52\x4D\x50\x56", 5);                  // support  support
135     add_auth_entry("\x50\x4D\x4D\x56", "", 4);                                                          // root     (none)
136     add_auth_entry("\x43\x46\x4F\x4B\x4C", "\x52\x43\x51\x51\x55\x4D\x50\x46", 4);                      // admin    password
137     add_auth_entry("\x50\x4D\x4D\x56", "\x50\x4D\x4D\x56", 4);                                          // root     root
138     add_auth_entry("\x50\x4D\x4D\x56", "\x13\x10\x11\x16\x17", 4);                                      // root     12345
139     add_auth_entry("\x57\x51\x47\x50", "\x57\x51\x47\x50", 3);                                          // user     user
140     add_auth_entry("\x43\x46\x4F\x4B\x4C", "", 3);                                                      // admin    (none)
141     add_auth_entry("\x50\x4D\x4D\x56", "\x52\x43\x51\x51", 3);                                          // root     pass
142     add_auth_entry("\x43\x46\x4F\x4B\x4C", "\x43\x46\x4F\x4B\x4C\x13\x10\x11\x16", 3);                  // admin    admin1234
143     add_auth_entry("\x50\x4D\x4D\x56", "\x13\x13\x13\x13", 3);                                          // root     1111
144     add_auth_entry("\x43\x46\x4F\x4B\x4C", "\x51\x4F\x41\x43\x46\x4F\x4B\x4C", 3);                      // admin    smcadmin
145     add_auth_entry("\x43\x46\x4F\x4B\x4C", "\x13\x13\x13\x13", 2);                                      // admin    1111
146     add_auth_entry("\x50\x4D\x4D\x56", "\x14\x14\x14\x14\x14\x14", 2);                                  // root     666666
147     add_auth_entry("\x50\x4D\x4D\x56", "\x52\x43\x51\x51\x55\x4D\x50\x46", 2);                          // root     password
148     add_auth_entry("\x50\x4D\x4D\x56", "\x13\x10\x11\x16", 2);                                          // root     1234
149     add_auth_entry("\x50\x4D\x4D\x56", "\x49\x4E\x54\x13\x10\x11", 1);                                  // root     klv123
150     add_auth_entry("\x63\x46\x4F\x4B\x4C\x4B\x51\x56\x50\x43\x56\x4D\x50", "\x4F\x47\x4B\x4C\x51\x4F", 1); // Administrator adm
151     add_auth_entry("\x51\x47\x50\x54\x4B\x41\x47", "\x51\x47\x50\x54\x4B\x41\x47", 1);                  // service  service
152     add_auth_entry("\x51\x57\x52\x47\x50\x54\x4B\x51\x4D\x50", "\x51\x57\x52\x47\x50\x54\x4B\x51\x4D\x50", 1); // supervisor su
153     add_auth_entry("\x45\x57\x47\x51\x56", "\x45\x57\x47\x51\x56", 1);                                  // guest    guest
154     add_auth_entry("\x45\x57\x47\x51\x56", "\x13\x10\x11\x16\x17", 1);                                  // guest    12345
```

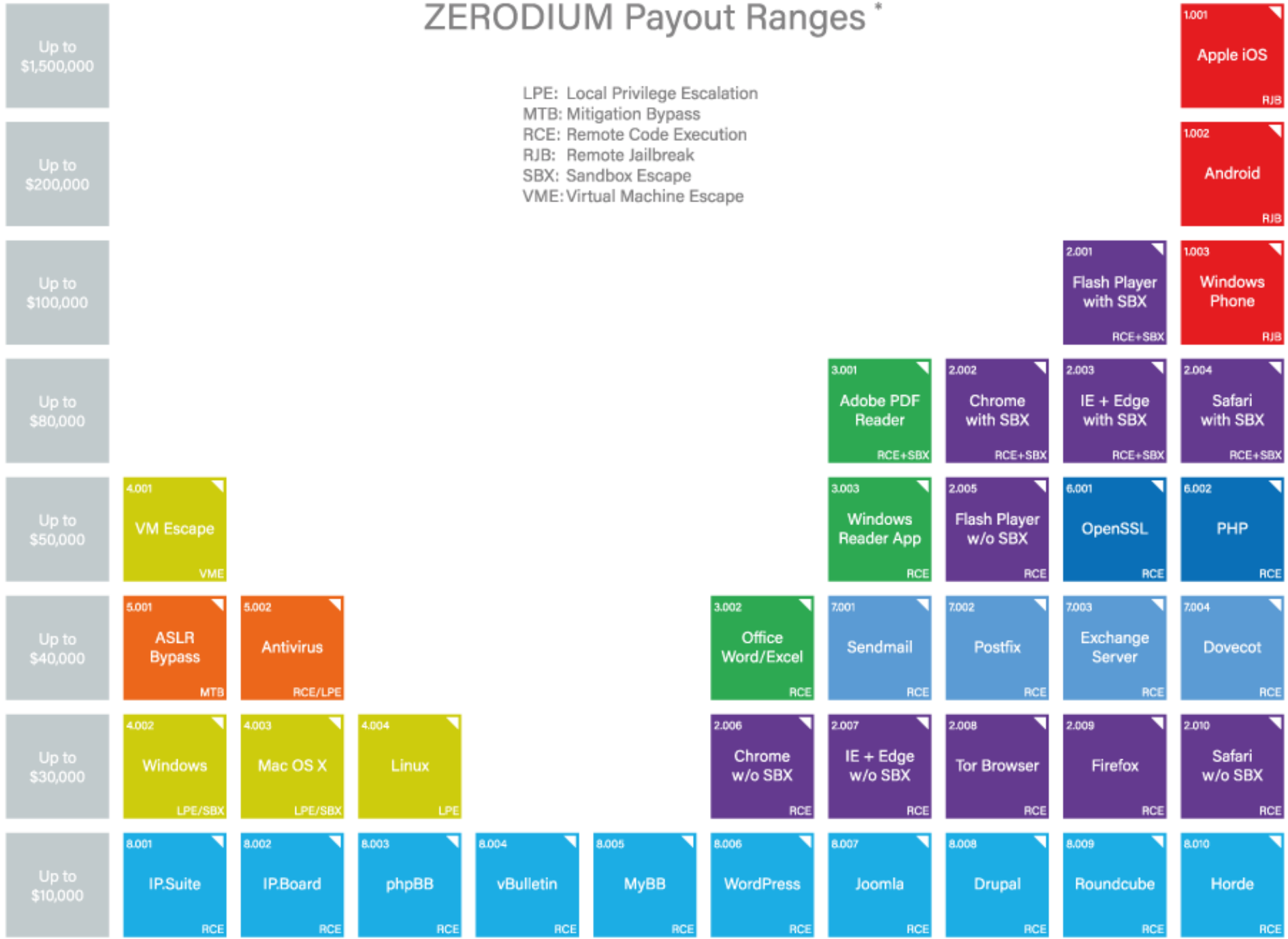
安全性錯誤

2017-08-02		-		Entrepreneur B2B Script - 'pid' Parameter SQL Injection	PHP	Meisam Monsef
2017-08-02		-		Joomla! Component SIMGenealogy 2.1.5 - SQL Injection	PHP	Ihsan Sencan
2017-08-02		-		Joomla! Component PHP-Bridge 1.2.3 - SQL Injection	PHP	Ihsan Sencan
2017-08-02		-		Joomla! Component LMS King Professional 3.2.4.0 - SQL Injection	PHP	Ihsan Sencan
2017-08-02		-		Joomla! Component Event Registration Pro Calendar 4.1.3 - SQL Injection	PHP	Ihsan Sencan
2017-08-02		-		Joomla! Component Ultimate Property Listing 1.0.2 - SQL Injection	PHP	Ihsan Sencan
2017-08-02		-		Premium Servers List Tracker 1.0 - SQL Injection	PHP	Kaan KAMIS
2017-08-02		-		EDUMOD Pro 1.3 - SQL Injection	PHP	Kaan KAMIS
2017-08-02		-		Muviko 1.0 - 'q' Parameter SQL Injection	PHP	Kaan KAMIS
2017-08-01		-		Advantech SUSIAccess <= 3.0 - Directory Traversal / Information Disclosure (Metasploit)	JSP	James Fitts
2017-08-01		-		Advantech SUSIAccess <= 3.0 - 'RecoveryMgmt' File Upload	JSP	James Fitts
2017-08-01				VehicleWorkshop - Authentication Bypass	PHP	Touhid M.Sh...
2017-08-01				VehicleWorkshop - Arbitrary File Upload	PHP	Touhid M.Sh...
2017-08-01		-		SOL.Connect ISET-mpp meter 1.2.4.2 - SQL Injection	Hardware	Andy Tan
2017-07-28				VehicleWorkshop - SQL Injection	PHP	Shahab Shamsi
2017-07-28		-		FortiOS < 5.6.0 - Cross-Site Scripting	Hardware	patryk_bogdan
2017-07-27		-		Joomla! Component CCNewsLetter 2.1.9 - 'sbid' Parameter SQL Injection	PHP	Shahab Shamsi
2017-07-26				Friends in War Make or Break 1.7 - Cross-Site Request Forgery (Change Admin Password)	PHP	shinnai
2017-07-26				Friends in War Make or Break 1.7 - SQL Injection	PHP	Ihsan Sencan

那未知的漏洞呢？

ZERODIUM Payout Ranges *

LPE: Local Privilege Escalation
MTB: Mitigation Bypass
RCE: Remote Code Execution
RJB: Remote Jailbreak
SBX: Sandbox Escape
VME: Virtual Machine Escape



** All payout amounts are chosen at the discretion of ZERODIUM and are subject to change or cancellation without notice.*

更何況

況且還有些漏洞還可能被低估

安全檢測的做法可以有

安全稽核

弱點掃描

滲透測試

什麼是滲透測試？

滲透測試是一種模擬駭客攻擊以驗證安全性的方法

稽核、弱掃與滲透測試的差別在於...

稽核是由管理面切入，缺乏實機驗證

弱掃可以找到漏洞，但漏洞可能被低估

滲透測試可以彌補上述的不足，但成本較高

但別怕！台灣最便宜的是 人！

只要有熱心的**替代役**就可以了！

滲透測試的階段可以分為前、中、後

前
決定範圍、方式、做法

中
執行滲透測試

後
提交報告、修復諮詢、結案

滲透測試做法

黑箱

XX

反黑箱

黑箱
對目標環境—無所悉下進行測試

白箱

當選後一切將

公開透明

白箱
充分掌握目標環境後進行測試

灰箱

灰箱
僅知悉部分資訊

告知 / 未告知
受測單位知悉或不知悉其將受測

目的是

目的是
將滲透測試當成鬥爭的工具！

目的是
將滲透測試當成鬥爭的工具！(大誤特誤

目的是
測試監控、應變與通報的能力

常見滲透測試方法

The Penetration Testing Execution Standard

http://www.pentest-standard.org/

Open Source Security Testing Methodology Manual

http://www.isecom.org/

OWASP Testing Guide

https://www.owasp.org/

OWASP Application Security Verification Standard

https://www.owasp.org/index.php/ASVS

Penetration Testing Framework by Kevin Orrey

http://www.vulnerabilityassessment.co.uk/

PCI Penetration Testing Guidance

https://www.pcisecuritystandards.org/

NIST Technical Guide to Information Security Testing and Assessment

http://csrc.nist.gov/publications/

滲透測試方法

滲透測試方法
掃描

滲透測試方法

掃描

列舉

滲透測試方法

掃描

列舉

弱點評估

滲透測試方法

掃描

列舉

弱點評估

攻擊

滲透測試方法

掃描

列舉

弱點評估

攻擊

後續攻擊

案例解析

ECSA 資安分析專家認證課程

EC-Council Certified Security Analyst

- 針對滲透測試所設計的安全分析課程
- 內容包含滲透測試之流程、技術與執行
- 適合對象
  - 資安 / 滲透測試工程師。
  - 完成 CEH 課程者。
  - 資安顧問 / 資安分析師。
- 建議先修課程：CEH

ECSA v9 課程內容

- 滲透測試導論、方法與流程
- 情報收集與漏洞分析
- 網路安全性測試
  - 外網、內網、無線網路、防火牆與入侵偵測
- 系統安全性測試
  - 網站、網站應用程式、資料庫、手機及雲端
- 合計核心章節 16 章、自修章節 23 章，共 39 章

ECSA 考試流程

EC-Council Certified Security Analyst

第一階段：術科

繳交滲透測試報告

第二階段：學科

參加電腦上機選擇題考試

術科 - ECSA Dashboard

- 上課首日發給 Access Code
- 需在期限內開通，逾期失效
  （效期約三個月，已加註在 Access Code 上）
- 開通後 60 天內需繳交滲透測試報告
  未繳交者視同考試失敗
- 繳交後 7 天內回覆，通過者核發 ECSA 考試券
- 審核未過者可購買新的 ECSA Dashboard，
  費用為 300 美元（含 30 天 iLab）

術科 - iLab

- ECSA Dashboard 開通後會立即收到 iLab Access Code
- 開通後 30 天到期
- 可另外加購 30 天 iLab，費用為 200 美元
- 總共有約 30 部虛擬機模擬企業網路環境
  含 Windows 用戶端與伺服器及 Linux 作業系統
- 共 14 個章節練習及 12 個挑戰

學科

- 滲透測試報告審核通過後，原廠將直接核發考試券
- 需在 90 天內完成電腦上機選擇題考試
- 考試時間 4 小時，共 150 題選擇題
- 採 ProctorU (http://www.proctoru.com) 線上監考
- 通過考試者核發 ECSA v9 證書

Thank You! 敬請指教！

SYSTEX making it happen 精誠資訊 | UCOM 恆逸資訊 教育訓練中心 Information Technology Education Center