目錄 Catalog / DevSecOps

Gartner 在 " 靜態分析產品 " 象限，Checkmarx
是唯一一家榮獲最高分的源碼檢測產品供應商
— 2014 ~ 2019 年度 SAST

Checkmarx

使應用系統安全更容易

NO. 1

Checkmarx

Checkmarx 是一款高準確率（低誤報）且靈活的源碼檢測軟體，可識別主流開發語言的數百種安全漏洞及品質缺陷，該軟體可以獨立運行，也可以有效整合到 SDLC 軟體開發生命週期過程中，去加速軟體安全漏洞及品質缺陷的發現和修復過程。該軟體可以獨立運作在企業內部，也可以利用私有雲或公有雲方式使用。

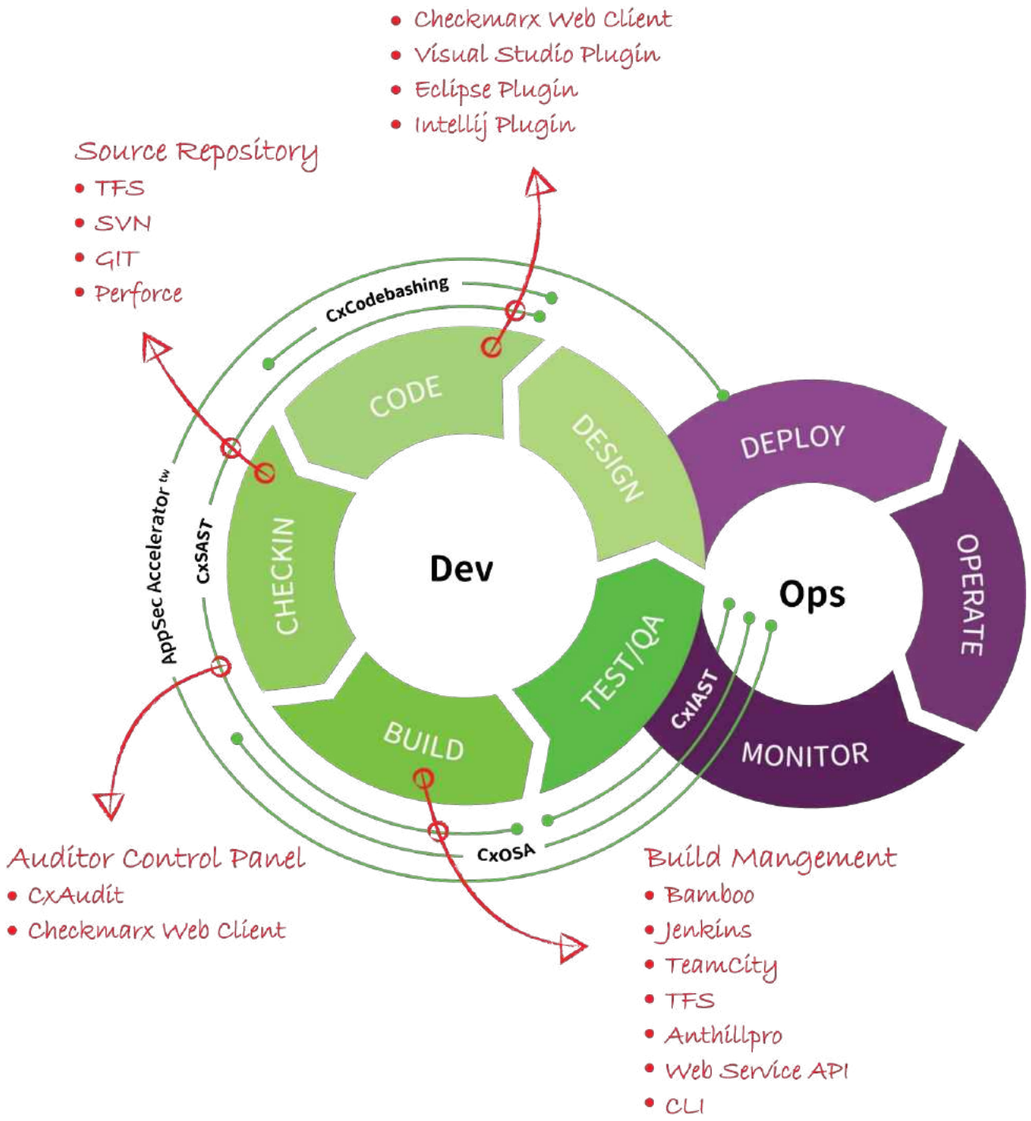
Checkmarx 作為軟體安全弱點檢測解決方案供應商，已經在業界確立了其領導地位，其 Static Application Security Testing（SAST）源碼靜態安全測試方案為眾人所知。客戶囊括了世界 10 大軟體供應商中的 4 家，數百家財富 500 大公司以及各個行業的諸多中小企業。

軟體開發生命週期

Checkmarx 可以協助企業整合源碼靜態安全測試 Static Application Security Testing（SAST）到軟體開發生命週期 SDLC（Software Development Life Cycle）中。我們整合了最常見的建置管理工具、問題追蹤工具、以及 IDEs 整合開發環境。

內容

此平台由集中的控制介面所組成，用以設定與監控應用系統安全測試，並提供儀表板顯示測試結果與風險指標，讓整個軟體開發生命週期都可以管理注意安全策略。



源碼靜態安全檢測（C x SAST）

C x SAST 是一種靈活、準確的企業級靜態分析解決方案，能夠識別自行開發程式碼中的數百個安全漏洞和弱點；支援 20 種以上的程式碼、框架與腳本語言。

安全軟體開發生命週期

Checkmarx 能夠幫助組織整合源碼靜態安全測試 Static Application Security Testing（SAST）到軟體開發生命週期。比如，整合到最常用的版本控管工具、建置管理工具、問題追蹤工具、以及 IDEs 整合開發環境。如果 Checkmarx 無法立即整合到軟體開發生命週期的某一組件，只需透過 API 便可輕鬆解決該問題。全面的 SAST 模型優勢在於：

- 資安團隊可以專注於資安政策，使用 Checkmarx 達到自動化執行資安政策。
- 對近期新增程式片段進行快速安全測試，在開發人員記憶猶新時對所有新發現弱點進行修補。此舉不僅大幅地降低成本，還省去了在產品上線日逼近時需修復大量安全隱憂的煩惱。

支援業界標準

MISRA · Mitre CWE · HIPAA · OWASP Top 10 · BSIMM · SANS 25 · PCI DSS · OWASP Mobile Top 10 · FISMA

C x SAST 能找出數百種弱點，包括常見弱點

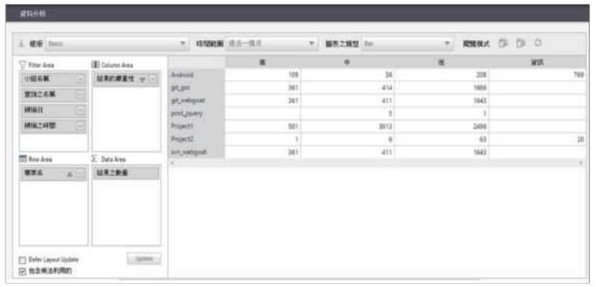
- SQL Injection
- Cross-site scripting
- Code injection
- Buffer Overflow
- DoS
- Parameter tampering
- Cross-site request forgery
- HTTP splitting
- Session Fixation
- Session poisoning
- Unhandled exceptions
- Log forgery
- Unreleased resources
- Unvalidated input
- URL redirection attack
- Dangerous Files Upload
- Hardcoded password

C x SAST　Web 介面

資安人員及開發人員必須審查已列舉出來的弱點並決定最佳的修復辦法。Checkmarx Web 介面為能提供最佳的用戶體驗，它能呈現攻擊流向及資料從輸入到執行的流程，點擊每個節點都能顯示出相關的方法和弱點處。

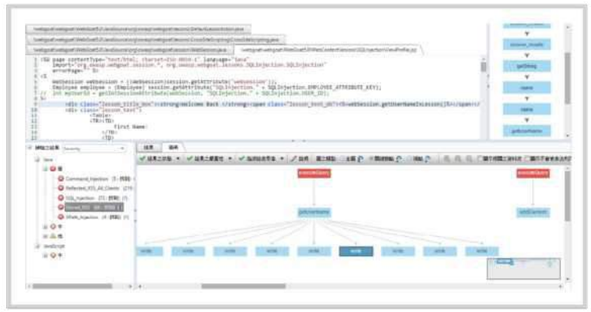
儀表板（Dashboard）與報表

使用 Checkmarx 可輕鬆分析資料並輸出報表。您可以使用預設的資料分析報表，或是透過樞紐分析的拖曳介面來設定所需的參數產生圖形化的資料，隨後可匯出 PDF 或 Excel 格式。



加速弱點修復工作

Checkmarx 不僅能識別弱點所在。除了表列的結果外，透過圖形演算法結合攻擊向量，直指多處攻擊流程的共同節點（最佳修復位置）。利用圖表（Graph View）讓開發人員可得知需修復最少的節點，即可達成全弱點修復。



Checkmarx 產品獨特之處

無需編譯、開發初期即可檢測

我們能夠檢測未經編譯的原始碼，意味著在開發周期的初期即能檢測，而此時恰是偵測安全漏洞的最佳時機，也意味著您不必擔心程式需經過編譯、完成編譯後才能檢測，只需於產品中放入程式片段即可。

檢測規則透明且可客製化

Checkmarx 的產品公開查詢規則，意味著您可以清楚的看到 Checkmarx 的掃描內容與掃描方式，同時，您也可根據特定的環境快速做出修復，並添加自行訂定的過濾方法，從而將誤報率和漏報率減少至可忽略不計的水準。進階的使用者往往會添加自己的查詢規則，利用 Checkmarx 輔助達成最佳撰寫實務、合規性及更多其他功能。

加速漏洞修復

Checkmarx 能做的不只是偵測識別原始碼漏洞。透過應用程式的整體資料流程，能偵測出漏洞關聯所在，利用「最佳修復點」您可一次修復大量漏洞，實現軟體修復最佳化。

源碼未變動則無須重複掃描

如果僅有數行程式有變動，通過 Checkmarx 獨一無二的差異掃描（incremental scan），就無需重複掃描整個專案。我們會分析自上次掃描後有變動的部分及其相依的文件，然後僅對這些進行掃描，如此便可快速得出結果，對於高速的敏捷開發環境尤為有用。

整合至現有軟體開發流程

Checkmarx 能非常靈活地整合至現有的軟體開發生命周期中，因此，您可以決定所需的安全政策，並且以自動化的方式實施。我們支援常用的版本控管工具、建置管理工具、問題追蹤工具以及 IDE 整合開發環境，使您能加速安全測試並確保最高效率地完成任務。

涵蓋主流的程式語言

Checkmarx 設計架構可以容易、快速的支援新的程式言及構架。目前支援超過 20 個程式語言、腳本語言及通用框架（Framework），每年大約新增 2~3 個種語言。

成功客戶

ATLASSIAN

沒有比 Checkmarx 更簡單易用的工具了。重要的是，你無需整合到建置管理工具，只需要把程式碼丟給它就可以了。在技術支援方面，我們團隊極為滿意。儘管時差不同，技術支援服務仍非常專業、及時。

salesforce

salesforce.com 網站選擇 Checkmarx 的靜態源碼分析工具作為官方 Force.com 的軟體安全檢測工具，至今檢測超過 13 億行 source code 。 Checkmarx 確保了 AppExchange 所有的應用程式安全性都達到最高標準。

台灣某政府單位

支援語言較多具能培養工程師定期使用源碼檢測工具的習慣，比起專案尾聲再執行檢測工具更有效率，同時避免相同的弱點產生，讓工程師開發時更嚴謹，也能節省事後修復程式問題的時間與人力。

中央研究院 ACADEMIA SINICA

Checkmarx 涵蓋率及準確率較高、支援行動裝置 APP 且不需受限於開發環境版本的特性，最重要的是簡易上手、不用高複雜度的建置環境，使用單位學習快速。

支援程式語言

Java · Kotlin · C++ · JS · PL/SQL
ASP Active Server Pages · HTML 5 · Visual Basic · GO · Swift
Scala · php · Perl · TypeScript · python
iOS · Android · COBOL · Groovy · Ruby

獎項

CYBERSECURITY VENTURES

評比 2018 年度前五百名最熱門及最具創新性公司， Checkmarx 榮獲 No.37

CYBER DEFENSE MAGAZINE WINNER 2019

榮獲 2019 年度資訊安全評比應用安全類先鋒

Gartner

榮獲 2019 年應用安全類魔力象限領導者

DEVELOPERWEEK

線上安全學習平台 Codebashing 榮獲 Devies Awards " 最佳行動開發創新 " 獎項

CYBERSECURITY BREAKTHROUGH AWARDS

2019 年度最佳應用安全解決方案

BLACK UNICORN AWARDS 2019 WINNER

軟體安全市場領導地位和願景榮獲「 Black Unicorn 」獎

常見問答

Checkmarx能輸出哪些類型的報表?

報表提供四種格式（PDF、RTF、CSV 或 XML）其中包括詳細專案檢測結與自訂儀表版（Dashboard）。

你們支援對mobile的檢測嗎?

C x SAST 全面支援 Android（Java / Kotlin）、iOS（Objective - C / SWIFT）與 WindowsPhone（C#） Apps 檢測。

你們是如何做到這些不可思議的事情的?

Checkmarx 對原始程式碼（無須編譯）進行解析並儲存於資料庫中，透過數百種規則檢測漏洞。

Checkmarx 是提供產品還是提供服務?

Checkmarx 是產品可以獨立運作於企業內部，也可以利用私有雲或公有雲方式使用。

我可以透過C x SAST了解因程式異程而產生的弱點嗎?

可以， Checkmarx 會提供並列出檢測結果指出差異對比，並提供修復建議。

我能與建置管理系統整合嗎？

可以，我們目前已有 Jenkins、Bamboo、TeamCity、TFS、SonarQube 等。

你們多久會發佈產品更新?

每年會發佈一個新版本，每季會發佈一個服務更新，並時按需求發佈 HotFix。

你們的操作介面是什麼?

目前支援多種操作介面（Web , IDE , CLI , WebService API），其中 Web 提供中文化介面。

我是否每次都必順重複掃描整個程式？

不需要。差異檢測（Incremental Scan）選項可以自動針對變動過的程式及其呼叫的其他程式進行檢測。

可以描述你們的產品架構嗎？

Checkmarx 安裝於中央伺服器中， Web 客戶及 IDE 可透過 http 或 https 與其連接。

互動系應用系統安全檢測（C x IAST）

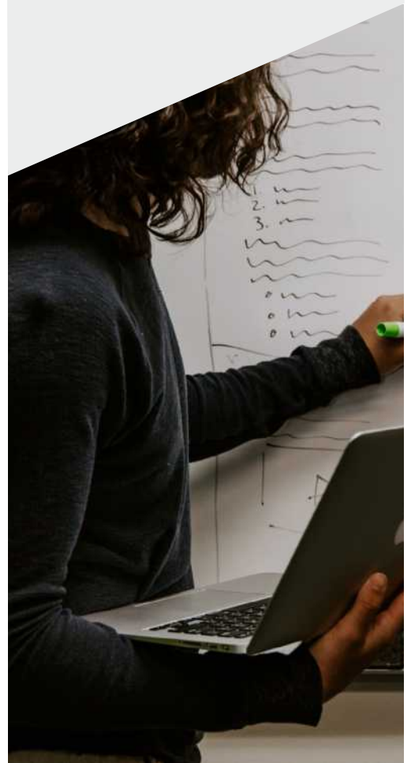
C x IAST 利用現有的功能測試（QA）來達成自動檢測正在運行應用系統的漏洞，從而填補了關鍵的動態軟件安全漏洞。 C x IAST 是業界第一個與靜態安全檢測解決方案完全整合，並提供客製化規則調整，進而可以大規模的覆蓋漏洞與提高準確性。

在當今競爭激烈的世界裡，比的就是產品上市的時間。為了贏得先機，企業組織需要不斷改進現有軟體或提供新的軟體，只求能夠快速發佈產品。這種對速度的需求，常使企業將安全性拋諸腦後，使得產品成為駭客的攻擊目標。為了避免發生上述事件，越來越多的企業組織開始尋找能夠支持這種快速頻繁發佈程式的安全測試工具。

C x IAST 是一種動態且持續的安全測試工具，它能利用開發流程中現有的 functional testing，即時檢測應用程式上的漏洞。

C x IAST 是專為敏捷開發、 DevOps 和 CI / CD 流程而設計的，與傳統的動態應用程序安全測試工具（DAST）不同。 C x IAST 可以跟著現有的軟體開發生命週期（SDLC）同步進行，因此不會拖延上市時間。

C x IAST 可以與 Checkmarx C x SAST 整合，關聯兩邊的掃描結果，提高掃描覆蓋率，以及更準確的漏洞修復措施。在不影響安全性的同時，縮短漏洞修復的時間，等同於提升產品上市速度。



優化您的弱點修復工作

C x IAST 是市場上唯一能夠與最佳 SAST 解決方案完全整合的 IAST 產品。相較於獨立的 IAST 產品，C x IAST 與 C x SAST 能相互關聯應用程式的檢測結果，提供更準確的補救方案。C x SAST 的原始碼洞察力與 C x IAST runtime 採集的準確度相結合，使得開發人員能夠更好地瞭解弱點的所在，分辨需要優先解決的急迫問題，因此能夠更快地進行弱點修復。

基於現有流程自動執行

C x IAST 不需額外執行專門的安全測試軟體來檢測運行中的應用程式。C x IAST 在測試環境中安裝非侵入式的 Agent，在背景持續觀察與蒐集應用程式的各種活動。安全掃描將隨著功能測試開始，也隨著功能測試結束。

快速交付的同時確保應用程式安全性

C x IAST 是專為 DevOps 所打造，可以無縫的導入 QA 自動化或 CI / CD 流程。只要將 C x IAST 部署好，Agent 會自動檢測 Server 中運行的應用程式，所以幾乎能支援所有 SDLC 流程。不論應用程式是否被刪除或重新部署；亦不論功能測試是自動還是手動，只要應用程式在 Server 中運行，C x IAST 就會自動進行安全檢測。

完善您的應用程式安全測試工具

C x IAST 擴展了 Checkmarx 軟體安全風險管理平台，為您的應用程式安全測試工具補上關鍵的一環。雖然 C x SAST 和 C x OSA 能夠掃描您所有自行開發的原始碼和第三方的開源函式庫，但仍然存在某些只能在 runtime 才能檢測到的安全缺陷。 C x IAST 可以在不影響現有 DevOps 和 CI / CD 流程下進行動態安全性檢測，強化 SDLC 流程。

C x IAST Web 介面

C x IAST 的獨特之處

唯一能夠完美整合最佳 SAST 解決方案的 IAST 掃描結果關聯性整合，可實現更快、更準確的漏洞嚴重分級，有效節省修復的時間成本。

無掃描時間直接利用現有 functionnal testing，無需單獨進行安全檢測。

使用 C x IAST 可即時回應在 functionnal testing 執行期間，一發現安全漏洞會立刻回報。

完整分析應用程式完整檢查應用程式的 runtime 數據流，包含自定義程式碼、函式庫、應用程式框架及組態檔。

容易客製能調整或客制檢測規則，進一步優化結果。用程式框架及組態檔。

即裝即用只需部署一次，就能檢測該 Server 上的應用程式，不須額外的營運成本。

靈活的部署選項能夠部署在企業內部的資料中心，或是部屬到 AWS 上託管的租用伺服器中。

支援程式語言

支援常見應用系統弱點，包含 OWASP Top 10 及其他，列舉如下

- SQL Injection
- XSS Injection
- OS Command Injection
- Trust Boundary Violation
- Cross-Site Request Forgery
- Decentralized RCE Vulnerability
- Path Traversal
- XPath Injection
- Parameter Tampering
- Open Redirect

AppSec開發培訓課程（C x CodeBashing）

Foreword 前言

Codebashing 開發培訓課程主要希望養成開發人員對於軟體安全的觀念與重視，讓開發人員能將安全掌握在自身手中並樂於執行。利用即時的弱點挑戰讓學員可設身處地的了解原因以及快速了解如何撰寫安全的程式碼。

根據美國國土安全局發布的論文（註1）指出，軟體安全基本上是軟體功能的問題之一，必須在軟體開發生命周期中有系統的管理及學習。

人們普遍認為安全來自於軟體工程師，但依據 Node.js® 和 Sqreen 的研究（註2），六成的工程師對於自己開發應用系統的安全強度沒有信心。這與 SANS 於 2016 年的應用程式安全報告（註3）的結果相符，該報告指出最核心的挑戰在於缺乏應用程式安全（AppSec）的技能、工具與方法。

而為什麼會缺少 AppSec 的技能？

從 CloudPassage（註4）的研究可以發現在美國 computer science 學程中只有一間要求安全概念課程才能畢業。依據 StackOverflow 的統計 69.1% 的工程師其實是自學而來的。這告訴我們若企業希望自己的工程師交付安全的程式碼，需要提供他們一流的安全程式撰寫教育訓練（Secure Coding Education, SCE）。

進一步來說，若企業也認同，依據 SANS 的報告企業認為開發人員的訓練中比起弱點的掃描/檢測，更加需要加強的是應用程式安全（AppSec）的技能。不過現實生活中即使企業安排了各式的安全教育訓練與方法，技能的落差仍有漫長的一段路需要努力。

本指南希望能降低企業所需之安全程式與開發人員技能間的落差。將引導您了解需要知道的所有事項，以確保軟體工程師獲得最有效的 SCE 並符合實際需求。

註1 https://www.us-cert.gov/sites/default/files/publications/infosheet_SoftwareAssurance.pdf
註2 https://www.businesswire.com/news/home/20171109005210/en/NodeSource-Sqreen-Survey-Quarter-Node.js-Developers-Form
註3 https://www.sans.org/reading-room/whitepapers/analyst/2016-state-application-security-skills-configurations-components-36917
註4 https://www.cloudpassage.com/company/press-releases/cloudpassage-study-finds-u-s-universities-failing-cybersecurity-education/
註5 https://insights.stackoverflow.com/survey/2016#developer-profile-education

Understanding the Developer's Perspective 了解開發人員的想法

現今快速的開發環境中，需要快速交付並持續整合、持續交付（CI/CD），開發人員最寶貴的資源就是時間。撰寫安全的程式碼會降低開發人員的速度，這所帶來的問題可能會導致忽視安全的議題。正因如何企業必須考量與 DevOps 環境可以匹配的源碼檢測 Source Code Analysis（SCA）方案。

在解決開發人員的安全技能落差時要注意的另一個重要因素在於開發人員的主要工作為撰寫「程式碼」；很少在應徵工作時條件為「安全程式撰寫/交付」。安全程式撰寫就慢慢演變為「如果有時間再處理（Nice to have）」，但往往時間都是不夠的；而軟體工程師在評估績效時通常是透過速度與解決的 Bug 數，而不是解決安全漏洞的數量。

綜合上述所言雖然對開發人員充滿同情，但仍需要求開發人員提交安全的程式代碼。為了實現這點，需要改變的第一件事即是開發團隊的領導者需要在開發的過程中處理/修正安全弱點。而開發人員也需了解目標為交付無 Bug 且預先考量妥善防範的程式碼。此時就需要落實安全程式撰寫教育訓練以達到這個目標。

Deciphering Developer Secure Coding Education 分析開發人員安全程式撰寫教育訓練

教育訓練常見的方式為影片、定期課程與線上課程。這些活動多半被列為待辦清單或例行公事，並非將其認定為安全程式開發的工具。因此多數此類活動在進行時，開發人員的思緒都在於其他工作的執行。也因如此參與的學員也會以較不重視的態度參與。有鑑於此，如何讓安全程式撰寫教育訓練更容易推行？可能要從課程的模式改變，或許改採用遊戲化、角色伴演的方式進行，將會是更能夠引發興趣及提升學習效率的一種方式。

Gamification 遊戲化訓練

遊戲化並非指的是玩遊戲，而是仿造遊戲的設計原則與元素。雖然遊戲化在教育訓練中並非是新的議題，但多數的安全程式撰寫教育訓練並未落實。當學員處在享受的環境中學習，其成效會更好。開發人員長時間都在撰寫程式碼，依據參加過課程的數千位工程師的回饋，透過程式碼的閱讀、修改會讓開發人員更能接受這樣的課程。若需在企業中落實安全程式撰寫教育訓練，請注意以下四點，以確保學員都能有效的參與學習課程。

Make it Interactive 一定要互動式的

Chief Learning Officer 提到：「點擊的頻率並不能代表學習內容有吸引力，有可能只是學員想快點完成」。這種情況在組織強制性的課程中非常常見，學員並不理解課程的內容，只是想提早完成，儘早回去完成繁重的工作，而不是接受重要的內容，這將導致無效的課程與測驗。

為什麼需要互動式學習，課程中包含的故事與實例是吸引學員參與的重要部份。

故事中創造了一種情境，讓學員切身參與，可提高課程吸收力與印象。此外，若課程僅需單純的點擊式互動，不易增加課程的互動性。 為了提高互動性，學員需更加關注內容，提供實務學習的機會；實際執行操作相較聆聽或閱讀的效果更好。

Tell a Story 勾勒情境

故事背景介紹、角色扮演可以刺激學員的反應，當現實生活中若面臨相同情況，將需如何應對？安全程式撰寫教育訓練若以條列式的問題、答案易讓學員產生例行公式的厭煩感。而情況、人物、所面臨的問題可避免這個問題。依據故事情節（攻防實例、需解決的漏洞），有助於記住所學的內容。而故事也是許多遊戲的精髓，往往是一種有趣的催化劑，與遊戲化是切不可分的。

Keep it Short 精簡

近年來因 3C 產品的發展，人們的能夠專注在一件事上的時間越來越短(註6)。提供的資訊保持精簡是最好的方式。如同演講中的使用的 PowerPoint 一樣，提供簡短重要的內容，以減少過多不必要的資訊。而這正是我們課程的原則，以最有限的時間、資源，提供最重要的安全資訊與防範作法。

Ensure They Win 確保參與者贏

依據 Dr. Ian Robertson 著名的研究「The Winner Effect」(註7)。最容易被低評的是大腦潛力。當滿足學員參與的信心，大腦會釋放些刺激因子增加腦中的多巴胺活性，讓學員可以擁有信心，積極主動的學習。

Contextual Learning 情境體驗

準備精彩、遊戲化的教學對於開發人員實際使用與學習是最重要的。而課程的學習與投入工作的時間往往是衝突的，盡量以最短的時間提供開發人員對於弱點修改的印象，當實務上發生時即可想起對應的方法。我們發現，開發人員在編寫代碼時可以持續訪問我們的培訓課程，鼓勵他們在遇到安全漏洞時回來查詢弱點的資訊。最後需持續更新程式語言版本。安全程式撰寫在各語言間有些許差異，但持續更新是所有語言都需注意的。

In conclusion... 結論

我們得到的結論是，以短期、互動、有故事性對於教育訓練來說是重要的元素。不要忽視「贏」對於課程的重要性，用勝利作為課程的結束，如成功解決弱點。確認開發人員的培訓是有效的；同時也因讓人員感覺良好，會自願的去找尋解決方案，以探索新的弱點挑戰。

Summing Up 總結

希望您在採購安全程式撰寫教育訓練前，清楚的了解課程的行式以及對於開發人員的幫助，本課程是藉由 Checkmarx 研究團隊豐富的經驗，並以遊戲的方式進行讓課程不再是枯燥的例行公事。若想了解更新請嘗試 CodeBashing 的線上課程，學習對於開發人員的撰寫實務。經歷課程後仍對處理方式有印象且實作它們。

支援語言

Java	.NET	Ruby	PHP	Python	Scala
C/C++	Android	Apex	ASP.NET	Groovy	JavaScript
Kotlin	Perl	GO	VB6	VB.Net	持續新增...

效益總結

Checkmarx 軟體安全平台是功能強大的軟體安全測試工具，提供以下重要不可或缺的好處。

企業級的軟體管理平台

集中設定與管理

單一的風險儀表板

量身定作，優化調整規模

您知道您所使用的資安檢測工具該如何正確的使用嗎?

依據叡揚資訊多年協助客戶導入資安工具的經驗，多數的客戶在面臨資安工具時，常有以下幾個情境：

- 不了解工具如何正確操作
- 收到很多的檢測報告，但對於修復不知所措

安全性弱點修復的挑戰

對於程式開發人員來說，安全性弱點總是又多又綁手綁腳，在不理解其原理的情況下，難以說服這些是需要花時間成本進行修復；對於 ISD 而言，將針對詢問的弱點，說明其原理以及可能的危害，藉此讓相關的處理人員了解所花費的時間成本是必要的，同時也提供建議的處理辦法，以利後續降低弱點的作業。

安全性弱點管理的挑戰

對於不同系統的檢測結果，或許可能有些問題的相似的，而這之間的修改 Know-How 是值得共享的，此時可藉此平台的分享歷史諮詢記錄來達到知識的傳承；利用搜尋功能來快速找尋解決辦法，提供修改的效率。

諮詢記錄分享