CYBERSEC 2025 臺灣資安大會 | 4/15-4/17 臺北南港展覽二館

SECURE SOFTWARE & DEVSECOPS FORUM

專案執行與資安弱點掃描修正流程整合實踐優化

陳嘉賢 Patrick Chen

金融業 / 資訊安全部

資安專業人員

為什麼會說這個主題.....

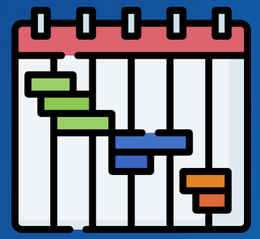
範圍 / 時間 / 成本

很痛 / 痛過.... 這樣做是不是更好

專案執行與資安弱點掃描修正流程整合實踐優化

安全防線

資安弱點若後期才修正，導致專案延期、成本上升



時程延期

疊床架屋
技術債

「同步進行」的整合方法

今天將獲得.....

1. 尚未進行 Devops 前應該注意什麼?

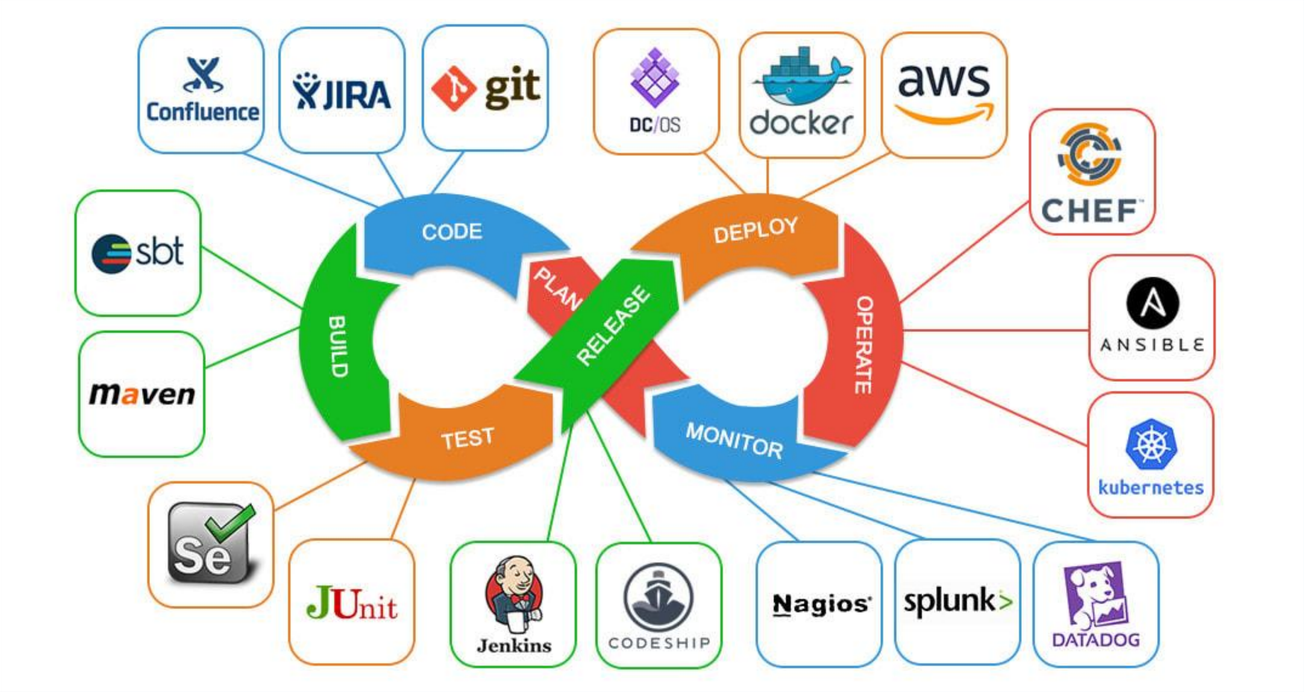
2. 修正流程怎麼配合專案時程

3.使用 Jira 進行追蹤修正進度

SDLC 階段中的資安整合

SDLC 階段中的資安整合（1）

SDLC 階段中的資安整合遇到的種種狀況



遇到問題.....

當前掃描政策 Checkmarx，

而專案工作說明書要求 Fortify

1. 必須 by 個案來處理
2. 很多事情都會被放大檢視

註：取自 https://de.fiverr.com/alialtaher/automate-your-software-delivery-pipeline-cicd

SDLC 階段中的資安整合（2）

SAST 工具不是屬於 Monitor 階段，而是屬於「前期防堵」的安全機制。

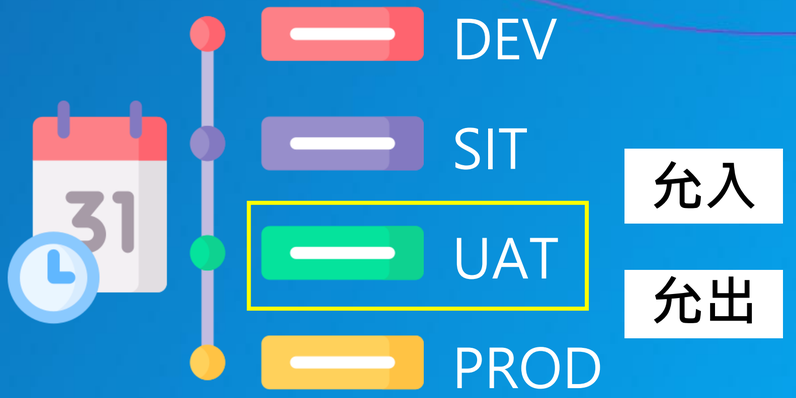
✦ Monitor 階段主要是：

Monitor 功能	常用工具與範例
系統運行監控	APM（如 Datadog、New Relic）、Prometheus、Grafana
日誌分析	ELK Stack、Splunk、SIEM 工具（如 QRadar、Sentinel）
行為異常偵測	EDR、XDR、SOC 團隊事件監控
威脅情報	Threat Intel、Threat Hunting、IDS/IPS
營運風險回報	通報通則、自動隔離、防火牆規則調整

✦ Checkmarx / Fortify 屬於「開發前期的預防性控管」

階段	功能	Checkmarx/ Fortify 所在位置
Code	程式撰寫時找出潛在安全問題	IDE Plugin / 即時掃描
Build	在建置過程中靜態程式碼分析	CI/CD 整合 SAST 掃描
Test	若整合到安全測試流程	可與測試同步跑

✦ Shift Left Security（安全左移）的典型例子



Shift Left Security（安全左移）

資安檢查「前移」到開發初期（如開發、建置階段），提早發現並修正漏洞，避免在部署或運行階段才發現重大風險。

項目	傳統流程（安全右移）	Shift Left（安全左移）
資安檢查時間點	系統快上線時才掃描	開發、建置就開始掃
問題發現時機	太晚，改起來成本高	越早發現，成本越低
負責人角色	資安部門最後才接手	開發/PM 就要開始顧安全
工具使用	傾向用 DAST / WAF	用 SAST / SCA / IDE Plugin

- ✅ 安全要 像品質一樣內建在流程裡，而不是事後補救
- ✅ 每一次 Commit、Merge 時都要被自動檢查安全
- ✅ 「安全左移」＝
  提早防堵風險 + 降低修復成本 + 提高開發效率

✅ 範例情境：「開發者小明寫了一個登入功能」

網頁登入

SQL Injection（SQL注入漏洞）

沒有左移	安全左移
1. 時效性 2. 回頭找開發者重寫邏輯 3. 又要過一次測試 4. 還會延誤功能改版	1. 剛寫完就跳出警告 2. 安全上線

安全左移就像寫完作業時，
老師馬上旁邊幫你改錯字，
不是等到交出去才被扣分

弱點修正流程重點

弱點修正流程重點（1）

我們內部使用的原始碼檢測工具

檢測項目	檢測工具	檢測標準	檢測流程
原始碼檢測	1. Fortify SCA Engine Version: Fortify SCA 20.2.2.0003 2. Fortify Security Content Version (Rulepacks): 2021 Q1	OWASP Top 10	1. 確認標的範圍 2. 檢測環境搭建 3. 原始碼檢測作業 ■工具掃描 4. 原始碼檢測報告產製及交付
網頁弱點掃描	HCL AppScan Standard	OWASP Top 10	1. 確認標的範圍 2. 檢測環境搭建 3. 網頁弱掃檢測作業 ■工具掃描 4. 網頁弱掃產製及交付
滲透測試	1. 手動。 2. 自動化工具：HCL AppScan Standard、Tenable Nessus	1. OSSTMM Open Source Security Testing Methodology Manual 第 3.0版 2. OWASP Open Web Application Security Project 網站暨其測試指南第 4.0版	1. 確認標的範圍 2. 資料蒐尋 3. 滲透規劃 ■資訊洩露與弱點測試 ■建立滲透途徑 ■安全漏洞滲透 ■信任關係測試 ■權限跳脫與提昇 4. 滲透測試報告產製及交付

弱點修正流程重點（2）

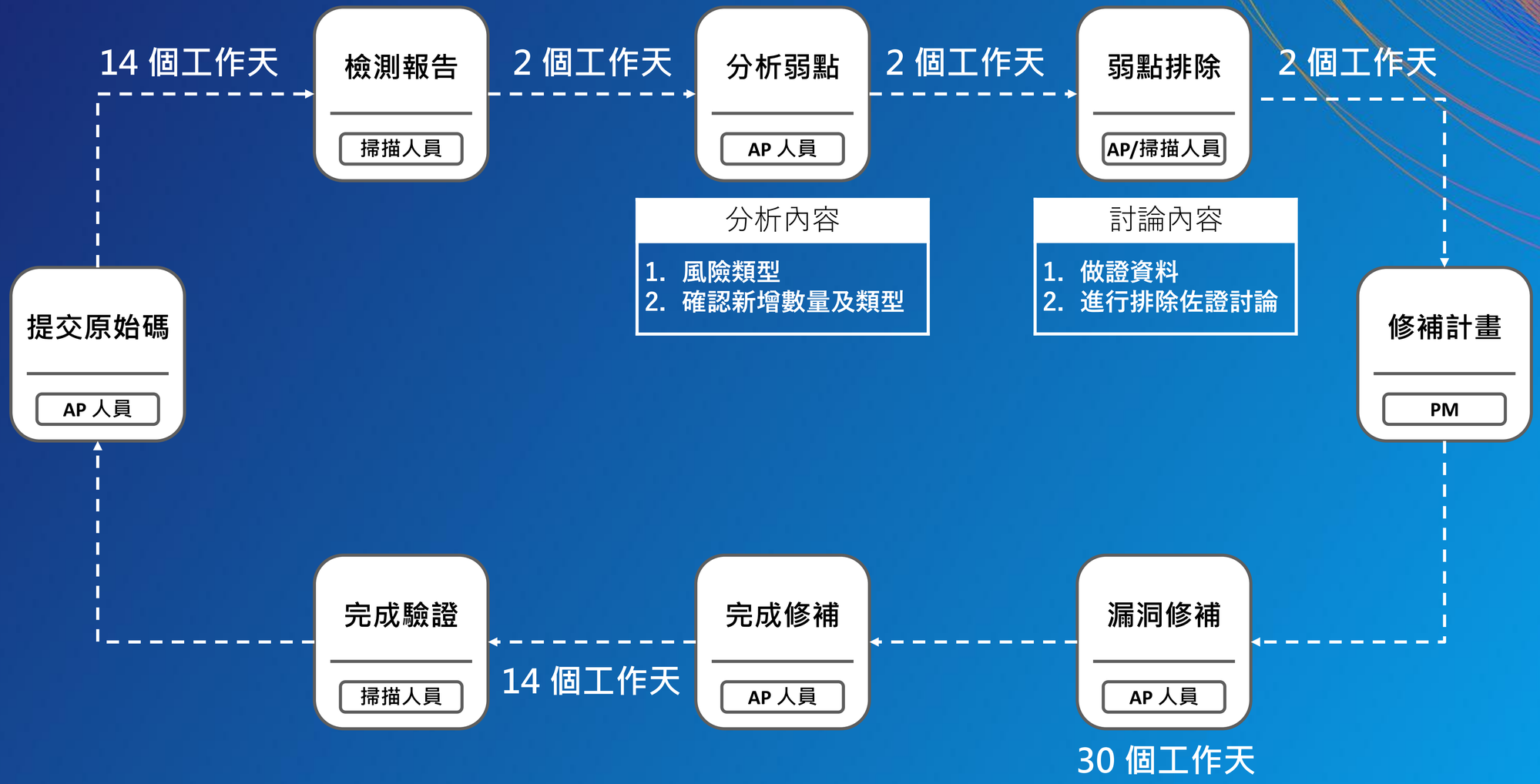
後續將安排資安掃描計畫

項次	檢測頻率	網頁弱點	滲透測試	源碼檢測	檢測與否	修補階段	預計覆核會議	備註
1	yyyy/5/19	•	•	•	完成			
2	yyyy/6/10	•	•	•	完成			
3	yyyy/6/30			•	完成			
4	yyyy/7/21			•	完成			
5	yyyy/8/11			•	完成			
6	yyyy/9/1			•	完成			
7	yyyy/9/22			•	完成	第一次 (現況)	yyyy-09-11	弱點修補
8	yyyy/11/10	•	•	•		第二次 （UAT允入）	YYYY/12月中旬	

註：執行修補前，召開資安弱點覆核會議，由資安單位與開發團隊覆核資安弱點分類及分級

弱點修正流程重點（2）

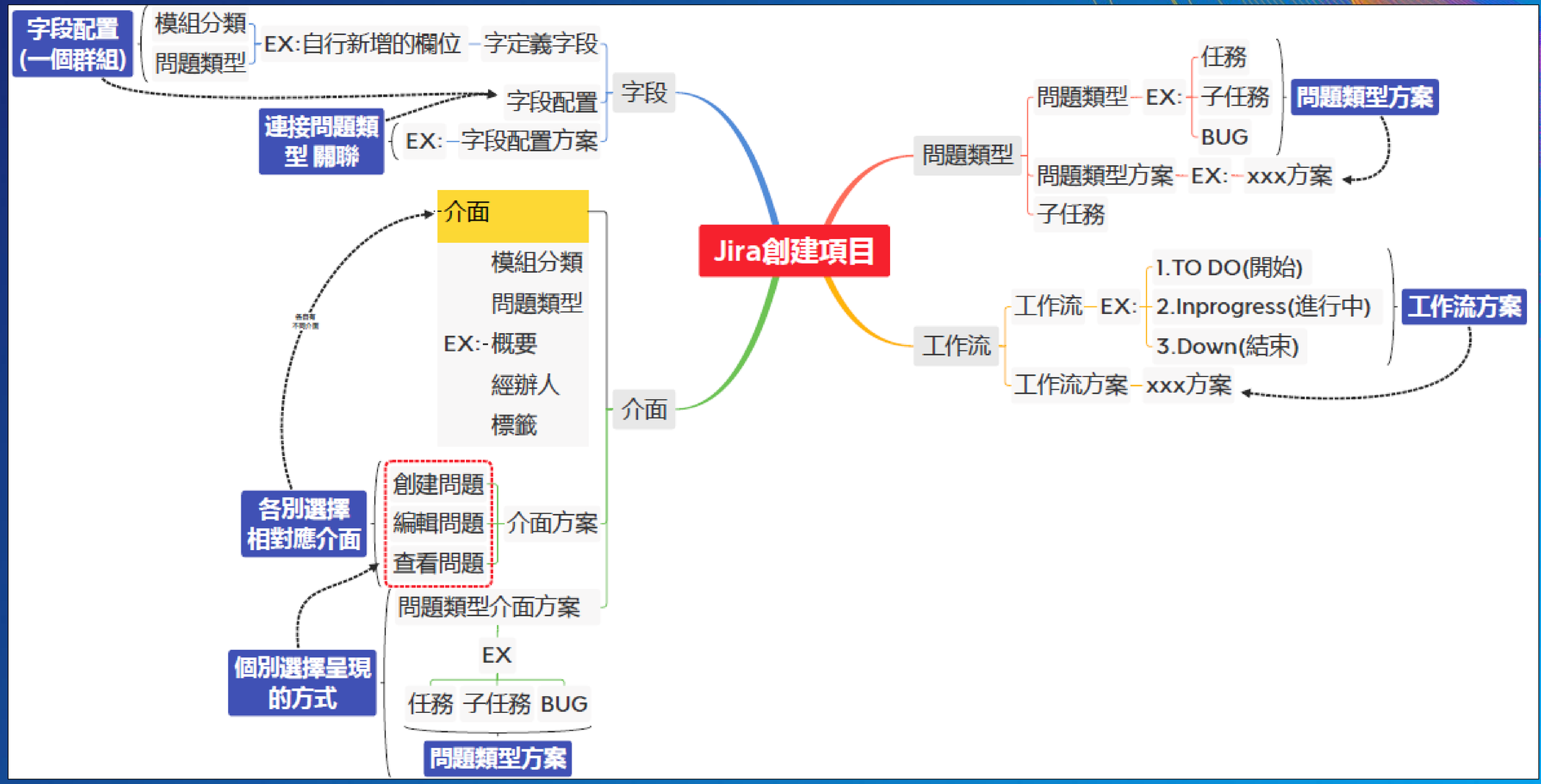
檢測後的修補作業流程



根據資安檢測報告梳理風險等級的風險類型之解決方案，作為程式開發人員的開發規範

使用 Jira 進行追蹤修正進度

Jira 結構


Jira創建項目
字段
字段配置
字定義字段
EX:自行新增的欄位
模組分類
問題類型
字段配置(一個群組)
連接問題類型 關聯
EX:
字段配置方案
介面
模組分類
問題類型
EX:-概要
經辦人
標籤
介面
介面方案
創建問題
編輯問題
查看問題
各別選擇相對應介面
問題類型介面方案
EX
任務
子任務
BUG
問題類型方案
個別選擇呈現的方式
問題類型
問題類型
EX:
任務
子任務
BUG
問題類型方案
問題類型方案
EX:
xxx方案
子任務
工作流
工作流
EX:
1.TO DO(開始)
2.Inprogress(進行中)
3.Down(結束)
工作流方案
工作流方案
xxx方案

Jira 結構 – 工作流程

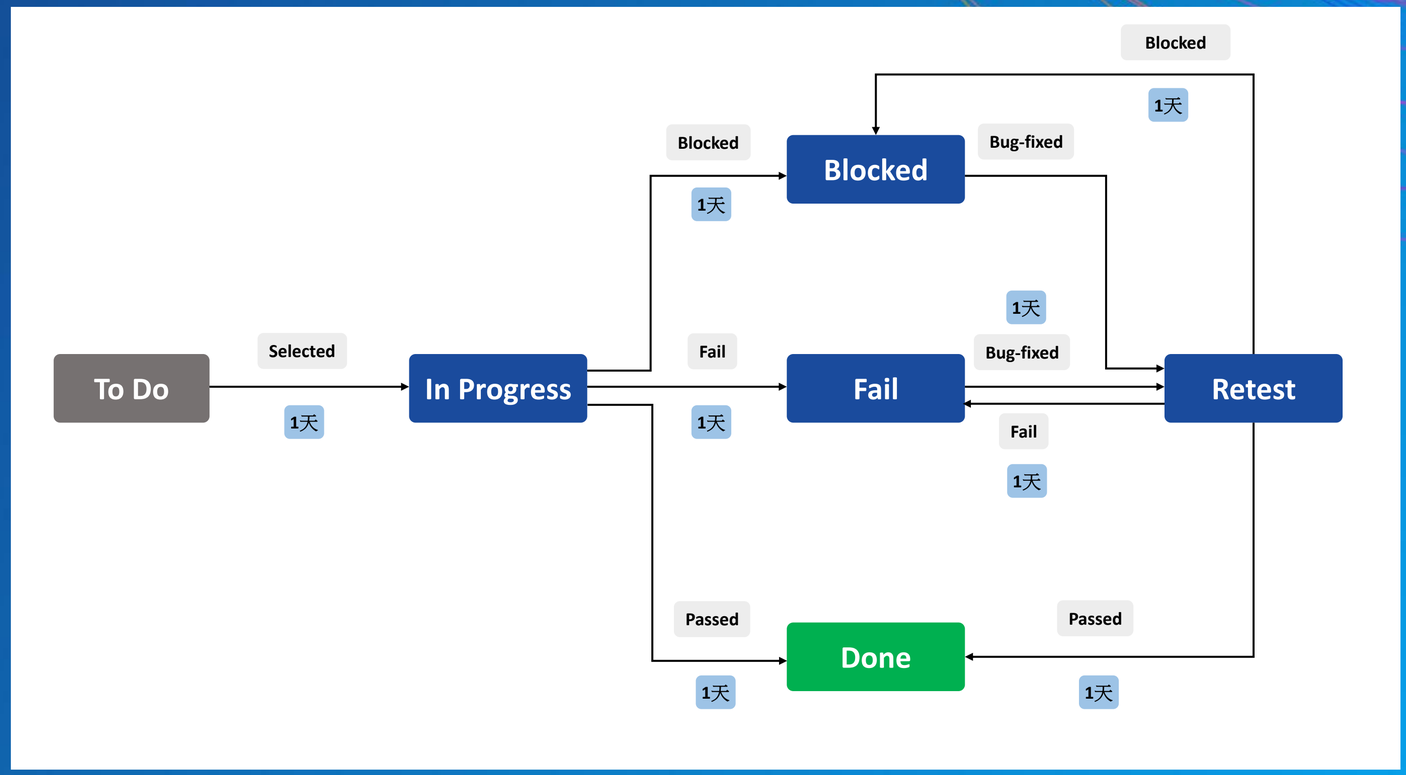
1.右圖此為目前Jira上應用的測試案例流程

2-1.狀態說明：

待辦　進行中　完成

2-2.狀態流轉細節：

轉換狀態　執行天數



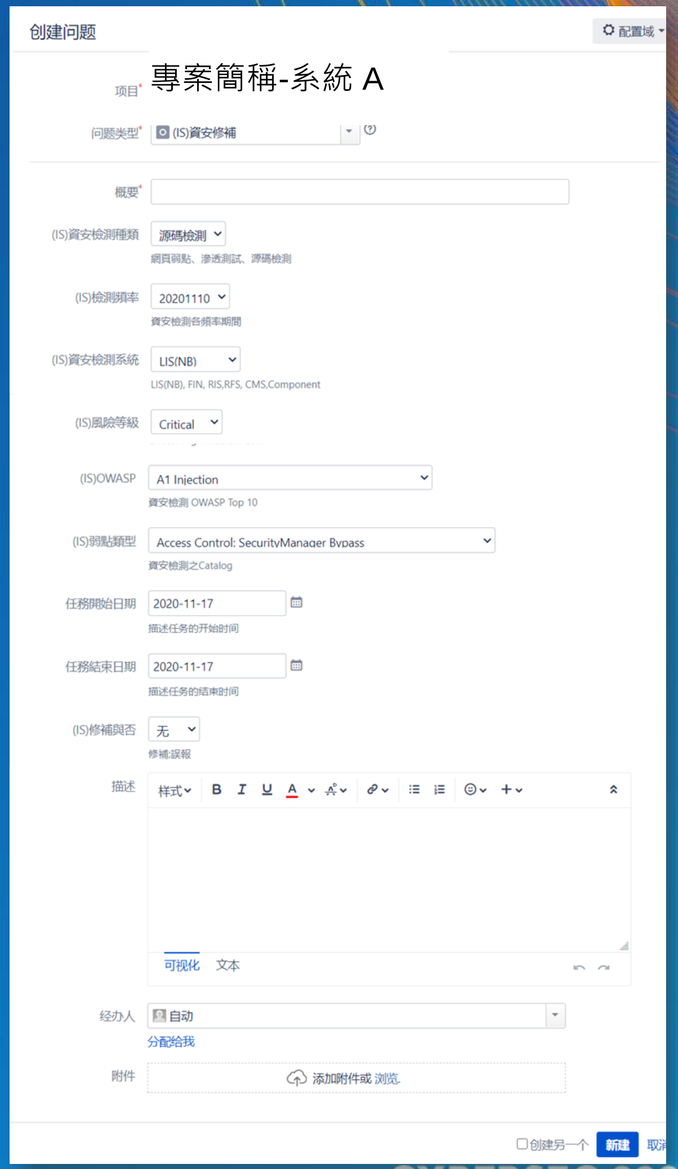
修補任務建立

- 依據資安檢測的報告產製所有等級的風險任務
- 所有中風險以上之風險修補均須要導入JIRA

編號	系統	Jira項目
1	系統 A	專案簡稱-系統 A
2	系統 B	專案簡稱-系統 B
3	系統 C	專案簡稱-系統 C
4	系統 D	專案簡稱-系統 D
5	系統 E	專案簡稱-系統 E

創建任務各欄位說明

項目	創建內容說明
問題類型*	點選(IS)資安修補
概要*	議題簡述說明 **檢測報表之Abstract**
(IS)資安檢測種類*	依任務歸屬之檢測種類點選 (如：源碼檢測、網頁弱點、滲透測試)
(IS)檢測頻率*	頻率為預計檢測日期 (如：請見 p3,p4 檢測頻率，20201110)
(IS)資安檢測系統*	修補任務屬於哪一個系統 (如：系統A , 系統 B , 系統 C , 系統 D)
(IS)風險等級*	修補任務之風險等級 (如：Critical、High、Medium、Low)
(IS)OWASP*	請選擇修補風險任務歸屬 OWASP Top 10 哪一種 (如：A1 Injection、A2 Broken Authentication …)
(IS)弱點類型*	請依檢測報告之Category 內容填入 (如：Access Contrl: Database、SQL Injection、Command Injection …)
(IS)是否修補	屬於誤報，請選擇誤報，否則請填寫修補 (如：排除弱點、修補)
經辦人	請填寫該議題的處理人員 (分派可用輸入姓名或員編搜尋)
報告人*	該任務的建立人員(系統預設)
任務開始日期*	請填寫該任務預計開始日期
任務結束日期*	請填寫該任務預計結束日期
描述	建立時可不填，若是誤報，請將誤報說明檔案放在Confluence, 並在此加入連結
附件	



任務建立後查看


NCS-CMPT
创建面板
问题
报告
发布版本
模块
Git提交
Gantt-Chart
插件
项目链接
資安檢測任務清單
查看所有或是篩選
某些條件清單
项目设置
專案簡稱-系統 A
NCSCMPT-108
在 TimeController.java 的 add() 方法將不可序列化的物件當作 HttpSession 屬性儲存，這樣會破壞應用程式的可靠性。
编辑
备注
分配
更多
開始修改
管理
导出
详情
类型: (IS)資安修補
状态: 开放 (查看工作流)
优先级: Medium
解决结果: 未解决
模块: 无
标签: 无
(IS)資安檢測種類: 源碼檢測
(IS)檢測頻率: 20200901
(IS)資安檢測系統: 專案簡稱-系統 A
(IS)風險等級: High
(IS)OWASP: A4 XML External Entities (XXE)
(IS)弱點類型: J2EE Bad Practices: Non-Serializable Object Stored in Session
(IS)是否修補:
建立檢測任務後
描述
点击添加描述信息
附件
人员
经办人: 經辦人
报告人: 負責人
表决: 0
管理关注列表: 0 开始关注这个问题
日期
创建日期: 2020-11-02 6:17 下午
已更新: 6天前
任務開始日期: 2020-11-01
任務結束日期: 2020-11-05
Agile
在敏捷看板中查看
Git 源代码

修補任務清單

資安檢測任務清單 保存为 详情 ★

共享 导出 工具

NCS-CMPT (IS)資安修補 状态: 全部 经办人: 全部 包含文本 更多 搜索 高级

(IS)OWASP: A1 Injection, A10 I... (IS)是否修補: 修補, 誤報 (IS)檢測頻率: 20200901, 20200... (IS)資安檢測種類: 源碼檢測 (IS)資安檢測系統: CMS, FIN, LI...

(IS)風險等級: Critical, High, Lo...

1–14 个问题，共 14 个

列

关键字	(IS)檢測頻率	(IS)資安檢測種類	(IS)資安檢測系統	(IS)風險等級	经办人	状态	概要	任務開始日期	任務結束日期	(IS)OWASP	(IS)是否修補	(IS)弱點類型
專案簡稱-系統 A	20200901	源碼檢測	FIN	High	經辦人	开放	在 TimeController.java 的 add() 方法將不可序列化的物件當作 HttpSession 屬性儲存，這樣會破壞應用程式的可靠性。	2020-11-01	2020-11-05	A4 XML External Entities (XXE)	誤報	J2EE Bad Practices: Non-Serializable Object Stored in Session
專案簡稱-系統 A	20200901	源碼檢測	FIN	High	經辦人	开放	UserAddInput.jsp 中的表單在第 78 行使用了自動完成，允許部份瀏覽器在其歷史資訊中保留敏感資訊。	2020-11-01	2020-11-05	A4 XML External Entities (XXE)	誤報	Privacy Violation: Autocomplete
專案簡稱-系統 A	20200901	源碼檢測	FIN	High	經辦人	开放	UserAddInput.jsp 中的表單在第 82 行使用了自動完成，允許部份瀏覽器在其歷史資訊中保留敏感資訊。	2020-11-01	2020-11-05	A4 XML External Entities (XXE)	誤報	Privacy Violation: Autocomplete
專案簡稱-系統 A	20200901	源碼檢測	FIN	High	經辦人	开放	在組態設定檔案中儲存純文字密碼可能會危及系統安全。	2020-11-01	2020-11-05	A1 Injection	誤報	Password Management: Password in Configuration File
專案簡稱-系統 A	20200901	源碼檢測	FIN	High	經辦人	开放	在 LogonSubmitController.java 的 logon() 方法	2020-11-01	2020-11-05	A1 Injection	修補	J2EE Bad

Eazy介面

To Do → In Progress → Done

(預計完成數)
NonZero(Count(

Cell formatting for: 偏差狀態 Custom formula

Range | Exact value | Regular expression | Top/Bottom | Heatmap

	Min	Max	Text	Background	Bold	Sample	
⋮	-0.099				☐		Delete
⋮	-0.199	-0.1			☐		Delete
⋮		-0.2			☐		Delete

3 available dimensions

Pages Nonempty
- Issue Type
- (IS)檢測頻率
- (IS)資安檢測種類
- (IS)是否修補
- (IS)處理單位

Rows Nonempty
- (IS)資安檢測系統
- (IS)風險等級

1

2 Columns: Measures

Table | Bar | Line | Pie | Scatter | Timeline | Map | Gantt | Gauge

Hide empty rows | Total row

(IS)資安修補 | 2022.08.08 | 源碼檢測 | 修補 | All (IS)處理單位s

4 5 7 6

		總數 (T)	待辦 (B)	修改中 (C)	完成 (D)	預計完成數 (E)	完成率=(D/T)	預計完成率=(E/T)	偏差率=(D-E)/E	偏差狀態	預計完成日期
LIS	嚴重風險	2			2	2	100.00%	100.00%	0.00%		2022-08-12
	高風險	11			11	11	100.00%	100.00%	0.00%		2022-08-12
	中風險	1			1	1	100.00%	100.00%	0.00%		2022-08-12
	低風險	8			8	8	100.00%	100.00%	0.00%		2022-08-12
RFS	嚴重風險	1			1	1	100.00%	100.00%	0.00%		2022-08-12
	高風險	4			4	4	100.00%	100.00%	0.00%		2022-08-12
FIN Phace II	高風險	8			8	8	100.00%	100.00%	0.00%		2022-08-05
	低風險	5			5	5	100.00%	100.00%	0.00%		2022-08-05
Total		40			40	40	100.00%	100.00%	0.00%	0.00%	2022-08-12

WorkFlow

任務結束日期

CYBERSEC 2025
臺灣資安大會

報告

報告

- 第 XX次 (XX)，檢測結果共 30 個風險數 (中級以上風險 15個，低風險 15個)，經修補後本次檢測結果如下：

修補結果	數量	備註
已完成修復	25	
未修復成功	5	5 個風險數未修復成功，皆為XX開發所產生： 2個嚴重風險，2個高風險，1個低風險
總計	30	中級以上風險 15 個，低風險 15 個

修補結果	數量	備註
第 XX 次修復後檢測未通過	5	已與 AP 溝通重新調整修復方式，將安排於 XXXXX/XX/XX 進行檢測。
新增風險	95	2022/XX/XX 檢測結果：新增 95 個風險數 (中級以上風險 20 個，低風險 75 個) • 20 個中級以上風險 (廠商A 10 個，廠商B 10 個) 來自於 10 種原有已知風險類型。 • 75 個低風險 (廠商A 15 個，廠商B 60 個) 來自於 20 種原有已知風險類型。 新增風險內容來自於 CR 所產生之風險， 請各模組開發過程中確實遵守程式開發規範，XXX於檢測前，皆會使用工具掃描程式，已降低既有風險類型發生機率。
總計	100	中級以上風險 24 個，低風險 76 個

備註：範例內容及數字皆為假資料

第XX次資安檢測風險報告 - 原始碼檢測 (分析結果)

檢測系統	風險總數	排除數	修復數
XX1	50	10	40
XX2	50	20	30
XX3	50	30	20
XX4	50	40	10
XX5	50	50	0
Total	250	150	100

原因說明

- 本次報告總共250，經分析並提交佐證說明後，正式審查程序上在進行中確認可排除共150個，餘100個需進行修復 (皆為前述產生之風險，中風險以上：24、低風險：76)。
- 已將 100 需修復數上傳 Jira 任務，預計**XXXX/XX/XX**修復完畢，預定 **XXXX/XX/XX**進行檢測。

備註：範例內容及數字皆為假資料

報告

模組	Priority	Category	XXXX/XX/XX	分析類別	說明
XXX	High	SQL Injection	10	修改中	廠商A開發CR內容相關功能程式碼所產生的風險，經分析後將排入修復計畫。
XXX	High	Path Manipulation	5	修改中	廠商A開發CR內容關功能程式碼所產生的風險，經分析後將排入修復計畫。
XXX	High	Access Control: Database	5	修改中	廠商A開發保單相關功能程式碼所產生的風險，經分析後將排入修復計畫。
XXX	High	Access Specifier Manipulation	5	修改中	廠商A契約模組開發（CR-XXX）保單退件管理-大宗掛號郵件編碼作業整合流程 相關功能程式碼，經分析後將排入修復計畫。
XXX	High	Mass Assignment: Insecure Binder Configuration	5	修改中	廠商A開發外圍入核心相關功能程式碼所產生的風險，經分析後將排入修復計畫。
XXX	High	Often Misused: Authentication	5	修改中	廠商A開發批處理相關功能程式碼所產生的風險，經分析後將排入修復計畫。
XXX	High	Password Management: Hardcoded Password	5	修改中	其中2個為廠商A開發批處理相關功能程式碼所產生的風險，經分析後將排入修復計畫。剩餘3個為誤判。
XXX	Medium	Cross-Site Scripting: External Links	5	修改中(含白名單)	打包資安檢測原始碼時，將本地測試文件誤提交上去所導致風險，經分析後將排入修復計畫。
XXX	Low	Access Control: Database	5	修改中	2個為廠商A開發保單相關功能所產生的風險，2個為廠商A開發CR內容相關功能程式碼所產生的風險，經分析後將排入修復計畫。
XXX	Low	Password Management: Password in Comment	5	修改中	廠商A開發ftp工具類相關功能程式碼所產生的風險，經分析後將排入修復計畫。
合計			100	—	—

備註：範例內容及數字皆為假資料

歡迎交流導入經驗

TEAM
CYBERSECURITY