兵臨城下：持續演變的軟體供應鏈風險

加強軟體供應鏈安全的三個步驟

目錄

概述

開源風險已超越授權合規性問題和漏洞的機會性利用問題，演變為蓄意的惡意攻擊。

本報告將探討軟體供應鏈安全格局的變化，重點討論了為什麼傳統的軟體組成分析(SCA)工具需要增強檢測功能，以提供從開源、第三方程式碼以及 AI 生成程式碼所繼承軟體風險的完整視圖。

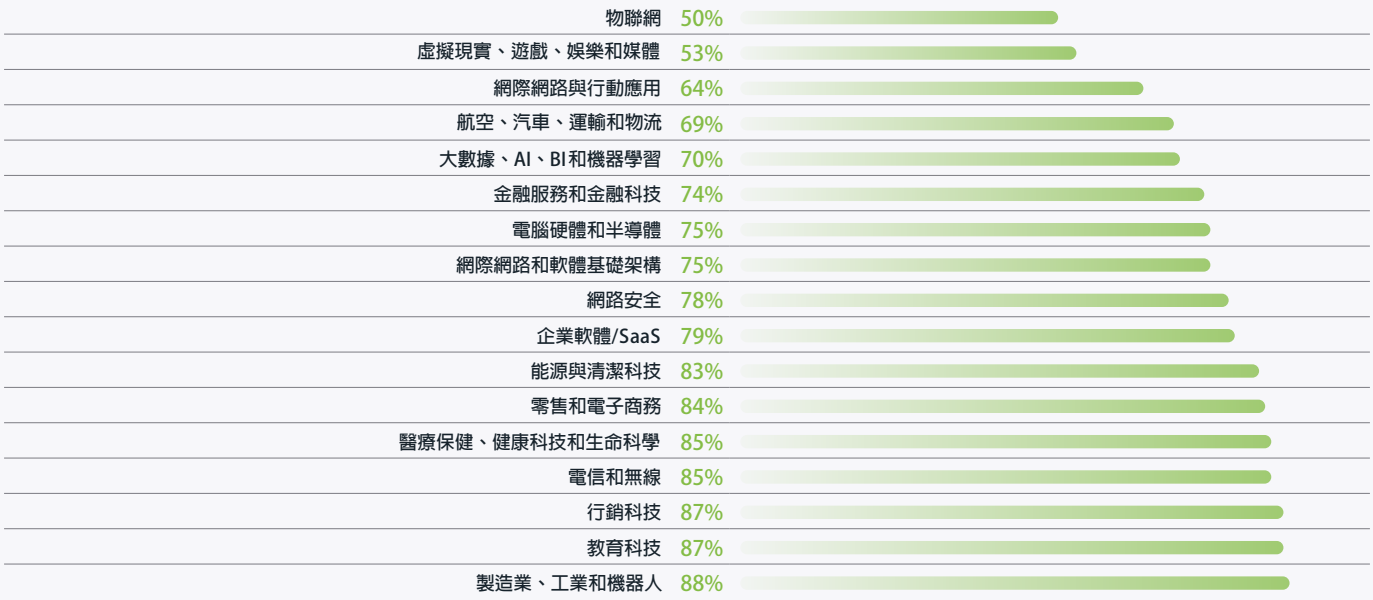
隨著開放原始碼的使用率越來越高，開源風險也在不斷演變，從授權合規性問題和漏洞的機會性利用，演變為蓄意的惡意攻擊。

現在幾乎所有組織機構都在其軟體中使用開放原始碼。為了保護軟體供應鏈安全，他們需要主動識別和管理所有這些威脅。

簡介

開源在當今的軟體中無所不在。作為一份關於開源安全的年度研究報告《開源安全和風險分析(OSSRA)》指出超過90%的商業應用中包含開放原始碼。開放原始碼和其他第三方程式碼通常透過“依賴項”— 由package manager分發，供其他軟體使用的程式碼庫—導入應用程式。由於這些開源庫很實用，它們幾乎被用於每個行業的每個應用中。事實上OSSRA報告指出，任何給定的應用程式中，平均有多達526個開源組件 — 依賴項和傳遞依賴項。

程式碼庫中的開放原始碼佔比



軟體供應鏈攻擊現已司空見慣，惡意行為者不再侷限於對軟體漏洞或其他程式碼弱點進行簡單利用，而是開始竄改程式碼和開發管道並傳播惡意軟體。

在其2024年度的《軟體供應鏈安全狀況報告》(The State of Software Supply Chain Security)中，ReversingLabs 指出過去三年中，來自開源套件包儲存庫的威脅增幅超過1,300%，包括攻擊者直接透過軟體套件推送惡意軟體的事件。兩家常用的開源軟體平台報告稱僅去年一年，惡意PyPI套件(用於Python程式設計語言的第三方軟體）就增加了400%。

除了惡意軟體套件包有所增加，ReversingLabs的報告還顯示，透過開源軟體平台洩漏開發者“機密”的事件顯著增加，其中包括洩漏敏感訊息，例如在Slack、AWS、Google、GitHub、Azure等常用平台上使用的API、加密金鑰和密碼。

請注意，根據ReversingLabs報告稱，導致機密洩漏的主因竟然是ChatGPT人工智能平台背後的OpenAI公司。據該報告稱，這反映了OpenAI平台(以及整個人工智能)的快速普及。用於與ChatGPT和DALL-E等OpenAI服務交互的API密鑰在該公司的客戶中廣泛分發，已成為最常見的洩密形式。

隨著針對第三方系統和軟體的供應鏈攻擊日益增加，對組織機構而言，充分瞭解程式碼組成變得至關重要。尤其是從公共儲存庫下載的開源依賴項，從供應商處購買的商業軟體包，由AI編碼助手產生的程式碼，以及容器和IT基礎架構中用於部署應用程式的程式碼。

有效地識別、追蹤和管理第三方程式碼對於軟體安全專案取得成功至關重要，也是加強軟體供應鏈安全的關鍵。

本報告中使用的資料

除非另有說明，否則本報告中引用的資料均來自2024年度《開源安全和風險分析(OSSRA) 報告》以及2024年度《軟體供應鏈安全狀況報告》。

在所有程式碼庫中，96%包含開放原始碼

OSSRA報告由Synopsys Software Integrity Group編製，旨在深入解析開源安全、授權型態和程式碼品質風險的現狀。2024年度的OSSRA報告使用了17個行業1,067個匿名商業程式碼庫的資料。

程式碼庫的所有程式碼中77%是開放原始碼

ReversingLabs的《軟體供應鏈安全狀況報告》分析了透過其Titanium和軟體供應鏈安全(SSCS)產品編製的公開和非公開的供應鏈資料。基於這些資料，該報告重點講述了軟體供應鏈安全的主要趨勢，以及組織機構保護軟體供應鏈安全所必備的關鍵知識。

持續演變的風險威脅著軟體供應鏈安全

《開源安全和風險分析報告》和《軟體供應鏈安全狀況報告》都著重強調了使用未加管理的開放原始碼風險。之所以強調“未加管理”，是因為開放原始碼本身的危險程度並不比專有或商業程式碼更高。但是儘管使用開放原始碼能夠帶來諸多好處，例如提高開發人員的工作效率，但也會帶來風險，組織機構必須準備好主動管理這些風險。

授權合規性風險

當開放原始碼開始普遍使用時，大多數組織都認為使用它的主要風險是授權合規性，尤其是商業程式碼與開源“著佐權”授權 (copyleft，也稱為 “病毒” 授權) 的合規性。“著佐權”授權通常包括互惠義務，

> AI助手可能會在不提示程式碼授權條款的情況下建議程式碼

聲明程式碼的修改版本將在與原始版本相同的條款和條件下發布，並且必須根據請求提供包含更改內容的原始碼。這些條款使得商業實體對在其軟體使用受“著佐權”授權約束的開放原始碼持謹慎態度，因為這可能會使整個程式碼的智慧財產權受到質疑。

更為嚴重的是，開發人員經常會在軟體中使用“程式碼片段”(從較大的程式碼段中提取的小片段)。雖然程式碼中可能只包含一小片開放原始碼，但軟體用戶仍必須遵守與該片段相關的所有授權要求。在程式碼生成中日益使用 AI 助手則進一步加劇了這一問題，因為 AI 程式碼編寫助手可能會在不提示程式碼任何授權義務的情況下建議使用程式碼片段。

比較典型的是Stack Overflow程式碼庫，它針對可在 Creative Commons ShareAlike (CC-SA)下公開訪問的所有開放原始碼提供訪問授權 — 包括程式碼片段。

某些情況下，CC-SA授權可以理解為要求任何源自“著佐權”授權的作品都必須根據相同的“著佐權”條款獲得認證，從法律角度看，這可能是一個問題。2024年度OSSRA報告中的資料顯示，CC-SA授權是是導致授權衝突的首要原因，僅 CC-SA 3.0 和 4.0 版本就產生了 33% 的衝突。

即使軟體中只有一個不符合條件的授權，也可能導致法律審查、併購交易凍結、知識產權損失、耗時的補救工作以及產品上市延遲。針對這些問題而開發的自動掃描工具，可以幫助組織機構識別出授權條款而不適合在其軟體中使用的開放原始碼。

儘管授權合規風險不像開源安全漏洞那樣備受關注，但致力於保護軟體供應鏈安全的組織機構也應對其給予高度重視。例如，在本文撰寫時，軟體自由保護協會(SFC)對加州電視製造商 Vizio 的訴訟仍在進行中。SFC 聲稱 Vizio在其設備中使用了開源的“著佐權”軟體，但沒有滿足使用該程式碼所附帶的要求。訴訟要求Vizio提供原始碼，遭到了Vizio的拒絕，這也是可以理解的。

2024年度的OSSRA報告指出，超過一半(53%)的受測應用程式中包含具有授權衝突的開放原始碼，導致這些應用程式的所有者可能面臨IP所有權問題。

開源漏洞風險

儘管開源授權衝突仍然是以軟體為主要IP的開發組織的關注重點，但開源風險遠不止授權合規問題。隨著開放原始碼的普及，組織機構需要解決開源漏洞的問題 — 這些漏洞很可能成為攻擊者利用整個軟體堆棧的攻擊向量。

隨著已知漏洞的資料庫呈指數級增長，已知漏洞數量已經無法手工編排。事實上OSSRA報告指出，2023年，74%的受測應用程式中包含高風險漏洞，比前幾年顯著增加。Oualys的研究顯示，2023年共披露了超過2.6萬個漏洞(包括影響開源和其他第三方程式碼的漏洞)，比上一年增加了1,500個。

2024年度的OSSRA報告指出74%的受測應用程式中包含高風險漏洞。

高風險漏洞是指已被主動利用、已有POC、或已被歸類為遠端程式碼執行的漏洞。例如Apache Log4j 2 Java庫中的嚴重漏洞Log4Shell、OpenSSL加密軟體庫中的漏洞Heartbleed、以及Apache Struts 遠端代碼執行漏洞（2017年導致Equifax 洩露1.479億筆美國居民私人紀錄的主要原因），都是一些影響開放原始碼的知名高風險漏洞。

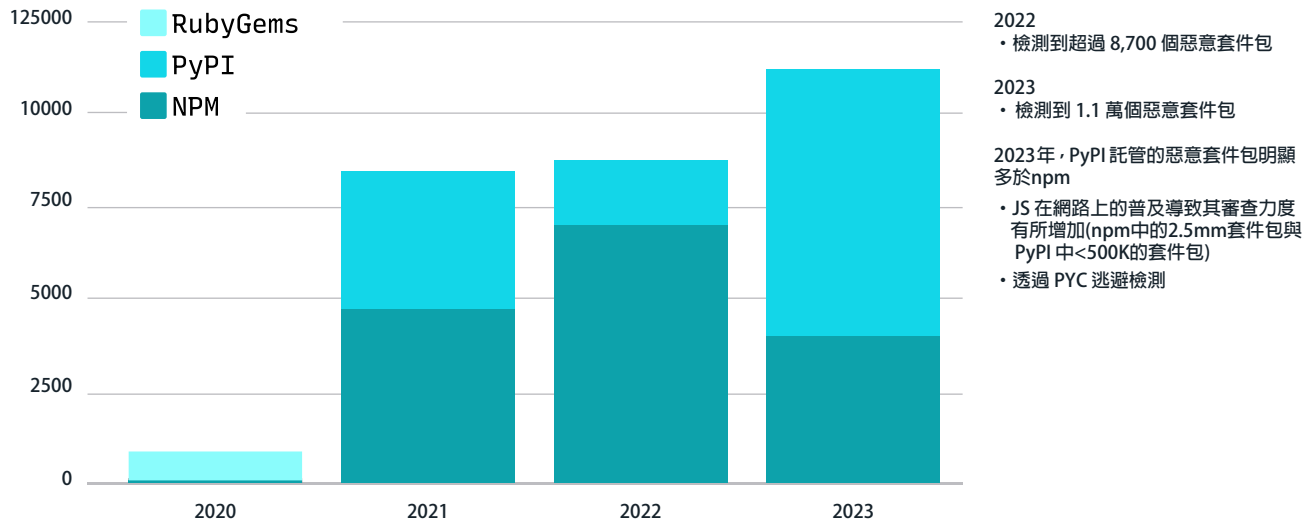
隨著人們開始日益深入地了解可利用軟體漏洞構成的威脅，市場也做出了回應：增強了自動漏洞掃描工具，使其具備軟體組成分析(SCA)功能，兼備漏洞識別和安全管理與授權辨識能力。現在的許多SCA解決方案都能將其分析結果彙整成軟體物料清單(SBOM)，對應用程式的開源依賴項以及這些依賴項的授權、版本和補丁狀態進行了分類。

從歷史上看，SBOM嚴重依賴於來自各種公共資料庫的漏洞資料，如國家漏洞資料庫(NVD)。然而，隨著已知漏洞數量的增長，以及NVD等公共平台難以跟上新漏洞的披露速度，導致許多SCA工具開始依賴於Black Duck® KnowledgeBase™ 等專有資料庫，利用詳細、即時的資料來補充公共資料，這有助於組織機構更快地識別安全漏洞並獲得可行的建議。

軟體供應鏈風險

除程式碼漏洞利用外，惡意行為者透過竄改程式碼和開發管道、傳播受污染的惡意軟體和惡意元件包等方式對軟體供應鏈發動蓄意攻擊的情況也很常見。

惡意開源套件包的發展趨勢



ReversingLabs在其關於軟體供應鏈安全性的2024年度報告中指出，過去三年中，來自開源軟體包儲存庫的威脅增幅超過了1,300%，包括攻擊者直接透過開源軟體包推送惡意軟體的事件。據2家常用的開源軟體平台報告稱，僅在過去一年，惡意軟體PyPI套件包(面向Python程式語言的第三方軟體)就增加了400%。

由於第三方程式碼是現代應用開發的重要組成部分，威脅行為者逐漸將其作為攻擊媒介，發動主要針對最終用戶組織及其客戶的供應鏈攻擊。例如，對IT管理公司SolarWinds開發的Orion軟體平台造成了嚴重影響的重大供應鏈攻擊。2019年，攻擊者成功地將惡意程式碼插入SolarWinds的Orion軟體更新中。

Orion擁有超過3.3萬客戶，其中包括許多大型私人企業和政府機構。據稱，這次攻擊影響了一半以上的客戶。財政部、國土安全部、國務院以及國防部等美國政府機構都受到了影響。Microsoft、Cisco和FireEye等私人公司也是攻擊目標。在漏洞被發現之前，攻擊者已經監控這些組織機構的電子郵件帳號和網路長達數月之久。

SolarWinds 2019年軟體供應鏈攻擊

自SolarWinds事件以來，具有同等影響的供應鏈攻擊事件再度發生，如3CX電話系統攻擊。3CX電話系統被全球60多萬家公司使用，包括汽車、製造商、醫療保健及航空等領域的公司。日用戶量超過1,200萬。2023年，攻擊者成功破壞了在建構過程中從中獲取3CX應用程式二進制文件的儲存庫，用他們自己的惡意版本替換了某些元件庫的合法版本。

儘管此次攻擊一個月內便被發現(而不是像SolarWinds漏洞那樣九個月後才被發現)，但仍對數十萬的3CX客戶產生了重大影響。

在本報告撰寫之際，美國網路安全和基礎設施安全局(CISA)警告說，已發現兩個版本的xz-Utils遭受攻擊。xz-Util是一組被廣泛使用的UNIX操作系統開源庫，包括支持世界上大多數伺服器的Linux操作系統。攻擊者竄改了xz-Utils程式碼，插入了惡意“後門”，從而可以像授權的管理員那樣對受影響的系統進行控制。

在CISA發出警告的同一天，開源軟體提供商Rad Hat發出了緊急安全警報，通知用戶其兩款Linux生態系統產品Fedora 40和Fedora Rawhide受到了xz-Utils惡意軟體的影響。該警告只是用戶”立即停止使用Fedora 40和Fedora Rawhide，直到將xz版本降級為止”。

ReversingLabs指出過去三年中，來自開源套件包儲存庫的威脅增幅超過了1,300%。

令人驚訝的是，此次攻擊似乎醞釀了三年之久。2021年，某不明人士或組織建立了一個GitHub帳戶。當最初的維護者以工作過度和精疲力盡為由放棄了對更新的主要監督義務時，該帳戶背後的實體逐漸接替了這一角色，變成了xz-Utils儲存庫新的維護者。

2024年初，這個新維護者最終連結到他們的惡意程式碼插入xz-Utils項目。曾有一位開源維護者這樣評論：“這可能是我們見過的執行最出色的供應鏈攻擊……簡直就是惡夢般存在：在廣泛使用的程式碼庫中插入了一個經過授權的、惡意的、具有卓越執行能力的上游程式碼。電腦科學家Alex Stamos表示，如果這個後門沒有被發現，“它將帶給其創造者一把可以打開全球數億台電腦的萬能鑰匙。”

正如xz-Utils駭客所證明的那樣，維護人員倦怠是影響許多開源項目的危險問題。OSSRA報告顯示在接受風險評估的900多個應用程式中，49%都包含在過去兩年未進行任何更新的開放原始碼。如果一個項目不再有人維護，那就意味著不會再有功能升級和程式碼改進，安全問題也不會再得到修復。

更糟糕的是，正如xz-Utils攻擊所證明的那樣，攻擊者可以在支持程度有限和維護力度下降的熱門項目中扮演受信任的角色。最終，他們可能有機會將受污染的程式碼插入到項目的某些版本中，然後將這些程式碼綁到下游操作系統和應用程式。

“這可能是我們見過執行最出色的供應鏈攻擊……簡直就是惡夢般存在：在廣泛使用的程式碼庫中插入了一個經過授權的、惡意的、具有卓越執行能力的上游程式碼“。

這種“殭屍”程式碼—無人維護但仍在使用的程式碼—是開源生態系統中的一個常見問題，因為開源生態系統在很大程度上依賴於無償的自願維護者來保證項目的安全和更新。據一些報告稱，在2020年維護的Java和JavaScript 開源項目中，近 20% 在2023年不再得到維護，使得這些項目很容易被利用。

促成這一趨勢的因素有很多，例如，日益增多的安全漏洞導致負責保持程式碼更新的維護人員精疲力竭，幾乎沒有選擇的情況下，這些維護者可能會選擇放棄，而不是在看不到明確結果的情況下繼續努力。

在 OSSRA報告審查的應用程式中，有49% 包含過去兩年未進行任何更新的開放原始碼。

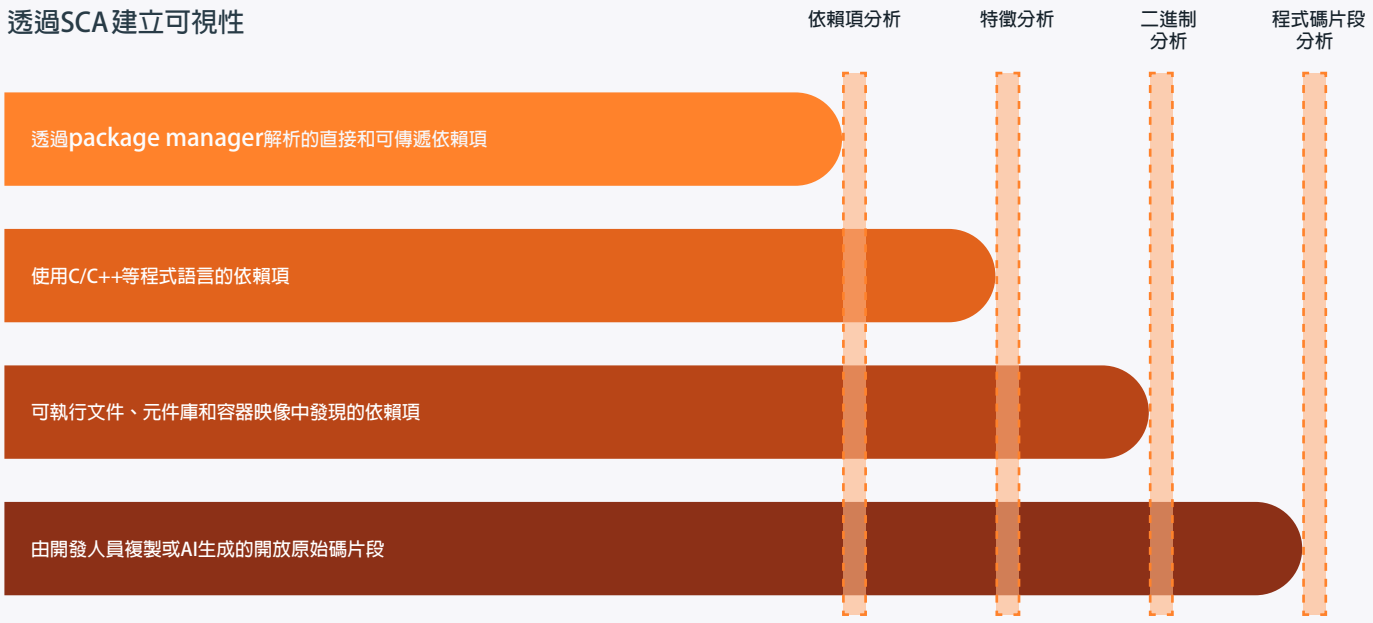
遺憾的是這種不穩定的監督和維護，再加上許多開源使用者不願意對其下載的程式碼進行安全檢查，使得開源儲存庫成為攻擊者青睞的攻擊媒介。

管控供應鏈風險的最佳實務

組織機構可以採取三步重要措施來保護其應用程式免受軟體供應鏈風險的影響。

建立對開源使用的可視性

如果96%的程式碼來自開源項目，91%的應用程式使用的開放原始碼遠遠落後於最新版本，則需要優先考慮程式碼可視性。為確保完整的可視性，您需要透過分析來識別直接的和傳遞的開源依賴項，無論語言、package manager聲明或原始碼和建構工具訪問如何。



將全面的分析建立到SDLC中

許多測試解決方案完全依賴於package manager清單來識別開源組件，但這些工具可能無法顯示程式碼中包含的大部份開源內容，包括：

- 開發人員沒有在package manager清單中聲明的開放原始碼
- 使用C和C++等語言的開放原始碼，這些語言未使用套件包管理器。
- 內置於容器映像各層中的開放原始碼
- 已編譯的二進制文件和工件中超出聲明範圍的開放原始碼

使用具備以下能力的解決方案很重要：

- 與Maven和Gradle等建構工具整合，以追蹤Java和C#等語言建構的應用程式中聲明的和傳遞的開源依賴項
- 查看文件簽名和編譯好的程式碼流，以解析未聲明的依賴項和上下文依賴項
- 使用二進制分析來識別編譯好的應用程式庫和可執行文件中的開源依賴項
- 執行程式碼片段分析，找到已複製到專有程式碼中的開放原始碼摘錄

顯然，這些分析首先需要不影響推動開發人員使用開源的速度與效率。這也正是將分析直接建立到軟體開發生命週期中的主要原因，這對確保不會給開發團隊帶來摩擦非常重要。

將依賴項分析建立到SDLC中

貫穿整个SDLC，應該在何時何地進行何種測試掃描呢？例如，在將依賴項併入程式碼時或在建構依賴項時，是否應該進行依賴項分析？當儲存庫有依賴項並且可以同時進行特徵分析和二進制分析時，執行特徵分析是否更有效？對於程式碼片段分析，相對於每次建構程式碼時都執行分析而言，選擇在夜間進行分析或者以周為單位進行分析是否更有效？

基於業務需求和開發工作流，每個組織都有不同的優先順序。其目標是建立兼顧測試準確性和開發效率的良好媒介，以便開發團隊可以在不影響其主要工作 — 建構軟體 — 的情況下，編制出準確且安全的依賴項清單。

使用覆蓋範圍比清單更廣泛的SBOM

了解軟體中包含哪些依賴項對於準確全面地管理程式碼至關重要，但要發揮作用，SBOM需要清楚地顯示相關風險。

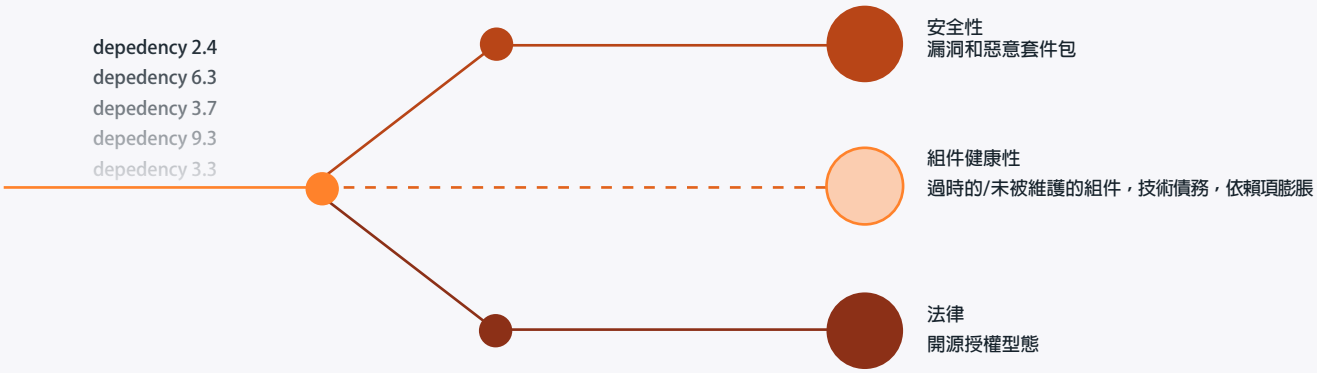
您應與提供包含詳細即時數據的SBOM供應商合作，以幫助您快速識別安全漏洞，並獲得可操作的建議。您應持續監控生成和導入的SBOM中識別的依賴項，以發現開源漏洞、機密、惡意軟體和惡意套件包。這一目標可以透過使用與SCA解決方案協同的SBOM來實現。

從程式碼品質的角度來看，全面的SBOM應列出所有組件的版本訊息和補丁狀態，並給出以下問題的答案：

- 這個特定的組件是否有社群支持？
- 上一次提交或更新是什麼時候？
- 是否屬於用戶未知的新項目？

從授權的角度來看，SBOM應就每個組件對相關的授權進行編排，以確保package manager、程式碼片段、AI生成的程式碼所導入的依賴項不會產生知識產權或其他法律問題。

全面的清單使風險管理成為可能



結語

由於開源和第三方程式碼的使用者基本上沒有進行風險管理，導致軟體供應鏈的安全日益受到威脅，過去三年間，除了傳統的授權和軟體漏洞風險外，來自開源儲存庫的威脅增幅超過了1,300%。攻擊者經常在PyPI、GitHub和npm等常用的儲存庫中植入惡意套件包和惡意軟體。雖然這些攻擊許多都是使用搶註錯誤網域名等策略的簡單攻擊，但近期的惡意軟體事件等備受矚目的供應鏈攻擊表示，軟體供應鏈威脅越來越複雜。

為了控制軟體供應鏈風險，您需要將傳統SCA的功能與依賴項、特徵、程式碼片段、二進制和容器分析相結合的解決方案，建立對開放原始碼使用情況的全面可視。理想情況下，該解決方案應包括以下功能：

- 多種開源檢測方法：該解決方案應提供套件包、依賴項、程式碼片段、二進制、特徵和容器分析，以跨越程式語言來識別開源組件。
- 第三方SBOM導入和分析：解決方案應能夠從第三方軟體供應商處導入，並自動對其中包含的開源、商業和訂製組件進行編排。
- 惡意軟體檢測：該解決方案應執行建構後分析，以檢測惡意軟體的存在，如可疑文件、不需要的潛在應用程式和異常文件結構。
- 風險識別與控制：該解決方案應能夠持續監控您生成的SBOM和導入的SBOM中的開源漏洞、暴露的機密、惡意軟體和惡意套件包。
- IP風險及授權合規管理：該解決方案應能夠自動識別與依賴項相關連的任何授權，並提供有關應用程式授權、部署和分發方式的義務與衝突的指導。它還應能夠對AI生成的程式碼進行分析，以識別出可能受版權或授權義務約束的開放原始碼片段。
- 業界標準的SBOM輸出：該解決方案應能夠以SPDX或CycloneDX格式交付SBOM。

Black Duck Supply Chain Edition簡介

Black Duck Supply Chain Edition 利用 Black Duck SCA 領先業界的自動惡意軟體檢測功能，針對您從開源、第三方和AI生成程式碼中繼承的軟體風險提供全面視圖。開發和資安團隊可以追蹤整個應用程式生命週期中的依賴項，以識別並解決安全漏洞、惡意套件包以及授權違規和衝突問題。

您可以在我們的網頁上了解更多資訊，並觀看有關如何透過Black Duck Supply Chain Edition來保護軟體供應鏈安全的影片。

docutek 達友科技

Synopsys 台灣代理商：達友科技股份有限公司
聯絡電話：02-2658-8970
聯絡信箱：contact@docutek.com.tw

新思科技與眾不同

新思科技提供的整合解決方案可以改變您建構和交付軟體的方式，在應對業務風險的同時加速創新。與新思科技同行，您的開發人員可以在編寫程式碼的時候快速兼顧安全。您的 開發和DevSecOps團隊可以在不影響速度的情況下在開發管道中自動進行安全測試。您的安全團隊可以主動管理風險並將補救工作聚焦在對貴組織最重要的事情上。我們無與倫比的專業知識可以幫助您規劃和執行所需安全計劃。只有新思科技能夠滿足您建構可信賴軟體的一切需求。

如想了解有關 Synopsys Software Integrity Group 的更多訊息，請訪問：www.synopsys.com/software.