WHITE PAPER

Cycode

Complete ASPM Platform

應用程式安全姿勢管理完整平台

從程式碼到雲端，
統一治理應用程式安全風險

以 Risk Intelligence Graph 重新定義 ASPM · SAST · SCA · Secrets · IaC · Container · Pipeline 安全

SaaSPodium 拓維雲智資安顧問團隊
Authorized Distributor of Cycode

版本 1.0 | 2026 年 4 月

目錄

執行摘要

現代軟體開發已經進入一個前所未有的複雜階段——多雲部署、微服務架構、容器編排、GitOps、CI/CD 全面自動化、AI 輔助開發、開源組件爆炸式使用。企業為了加速產品上市,普遍採用了超過 10 種以上的開發與資安工具:SAST、SCA、Secrets Scanning、IaC Scanning、Container Scanning、DAST、Pipeline Security、SBOM 管理、License Compliance 等等。

這種「多工具、多儀表板、多告警」的架構,表面上看似完整,實則造成嚴重的資安債務:

- 資安團隊每天淹沒在數千筆來自不同工具的告警中,無法判斷哪些真正重要
- 開發人員被「工具疲勞」壓垮,對資安工具失去信任、選擇忽略告警
- 同一個漏洞可能在五個工具中重複出現,浪費修復資源
- 缺乏跨工具關聯能力,無法回答「這個 CVE 實際上會不會被攻擊者利用」的關鍵問題
- 合規報表需要手動整合多個工具的輸出,每次稽核都是耗時的工程

Cycode 以 **「Complete ASPM(應用程式安全姿勢管理)」** 的架構,從根本解決上述問題。Cycode 不只是把多個工具堆疊在同一個平台上,而是以 **Risk Intelligence Graph** 為核心,跨 SCM(程式碼)、CI/CD Pipeline、Artifact(Container / Package)、Runtime(Cloud / K8s)建立完整的資產與風險關聯圖譜,讓企業第一次真正掌握「哪些應用程式風險最重要、應該優先修復」。

本白皮書將深入剖析:

- 當前應用程式安全市場的結構性問題與 ASPM 的崛起
- Cycode 平台的完整架構、七大核心能力與差異化價值
- Risk Intelligence Graph 如何重新定義應用程式風險優先級
- 與 Checkmarx、Snyk、Veracode、SonarQube 的深度比較
- 與雲端工作負載安全產品 Twistlock(Prisma Cloud)、Sysdig 的定位差異
- 典型企業導入案例、商業價值量化與 TCO 分析

> *「導入 Cycode 之前,我們有 7 個應用程式安全工具,每月產生超過 40,000 筆告警。導入 Cycode 之後,我們把真正需要處理的風險收斂到每月不到 200 筆,修復效率提升 8 倍。」*
> *— 某全球金融集團 CISO*

一、應用程式安全的結構性困境

要理解 Cycode 與 ASPM 為什麼是應用程式安全的新典範,首先需要理解當前企業 AppSec(應用程式資安)所面臨的深層問題。

1.1 工具爆炸與告警疲勞

根據 ESG 研究,大型企業平均使用 12-18 種應用程式安全工具:

- SAST(靜態程式碼掃描):Checkmarx、Veracode、Fortify、SonarQube
- SCA(軟體組成分析):Snyk、Black Duck、Mend(WhiteSource)
- Secrets Scanning:GitGuardian、TruffleHog
- IaC Scanning:Checkov、Terrascan、Bridgecrew
- Container Scanning:Twistlock、Aqua、Sysdig
- DAST:Burp Suite、Invicti、HCL AppScan
- SBOM / License:Anchore、FOSSA
- Pipeline Security:StepSecurity、Legit Security

這些工具每天產生大量告警。某《財富》500 強企業的實際數據:SAST 每月 12,000 筆告警、SCA 8,500 筆、Secrets 450 筆、IaC 3,200 筆、Container 15,000 筆——合計每月約 39,000 筆告警。其中真正需要立即處理的關鍵風險通常不到 1%。

1.2 傳統工具的五大根本缺陷

缺陷一:缺乏跨工具關聯(Correlation)

SAST 告訴你「這段程式碼有 SQL Injection 風險」,SCA 告訴你「這個套件有 CVE」,Container Scanning 告訴你「這個映像檔有高風險漏洞」——但沒有任何工具能告訴你「這個 CVE 在這個應用程式中是否真的可被攻擊者觸達」。缺乏關聯讓企業無法優先處理真正重要的風險。

缺陷二:無法理解業務情境(Business Context)

一個 CVSS 9.8 的「Critical」漏洞,如果存在於一個只有開發測試用的內部服務,可能實際風險遠低於一個 CVSS 7.5 的「High」漏洞存在於面對網際網路的支付系統。傳統工具無法納入業務情境,導致資源錯配。

缺陷三:開發者體驗糟糕

多數傳統 AppSec 工具設計於 AppSec 工程師視角,不是開發者視角。開發者被迫登入多個平台、看晦澀的告警描述、無修復建議、在 IDE 中看不到即時回饋,最終演變成「資安團隊產出清單、開發團隊選擇性忽略」的對立關係。

缺陷四:無法涵蓋完整 SDLC

即使使用 10+ 種工具,仍然有大量盲區:CI/CD Pipeline 本身的安全性(Pipeline Injection、惡意 Runner)、SCM 本身的安全設定(分支保護、Code Review 強制)、第三方 Actions 的供應鏈風險。這些盲區正是近年供應鏈攻擊(SolarWinds、npm 供應鏈攻擊、GitHub Actions 攻擊)的主要入口。

缺陷五:TCO 失控

使用 10+ 種工具,代表 10+ 份合約、10+ 套授權費、10+ 個維運負擔、10+ 種工具培訓。以中大型企業來說,AppSec 工具總年費往往超過 USD 1M,且年年成長。

1.3 供應鏈攻擊的崛起改變了一切

2020 年的 SolarWinds 事件、2021 年的 Log4Shell、2023 年的 3CX、2024 年的 XZ Utils Backdoor、2024-2025 年的持續 npm 供應鏈攻擊——軟體供應鏈(Software Supply Chain)已經成為當代最嚴重的資安戰場。

傳統 AppSec 工具的視野侷限於「程式碼本身是否安全」,但攻擊者早已轉向:

- 滲透開源套件儲存庫(npm、PyPI、Maven Central)
- 攻擊 CI/CD 系統(GitHub Actions、Jenkins、GitLab Runners)
- 竊取硬編碼的 Secrets(API Keys、Cloud Credentials、Signing Keys)
- 在 Container Image Registry 植入惡意映像
- 濫用 IaC 範本部署後門

關鍵洞察: 現代應用程式安全不能只關注程式碼,必須涵蓋 **從程式碼到雲端(Code-to-Cloud)** 的完整軟體供應鏈。這正是 Cycode 的設計起點——Cycode 起家於 **SSCS(Software Supply Chain Security)**,後來演進為 **Complete ASPM** 平台。

二、什麼是 ASPM(應用程式安全姿勢管理)

ASPM(Application Security Posture Management,應用程式安全姿勢管理)是 Gartner 於 2023 年正式提出的新興類別,被列入 **《Gartner Hype Cycle for Application Security》** 的核心。Gartner 預測,到 2026 年,**超過 40% 的企業將採用 ASPM 平台取代多個獨立工具**,相較於 2023 年的不足 5%。

2.1 ASPM 的核心定義

ASPM 不是單一工具,而是一個涵蓋以下四大能力的統一平台:

核心能力	英文	說明
可視性	**Discovery & Inventory**	自動盤點所有應用程式資產:程式碼儲存庫、套件、容器、API、IaC、Pipeline、Runtime 服務。
評估	**Assessment**	整合多種掃描能力(SAST、SCA、Secrets、IaC、Container),對資產進行全面資安評估。
優先排序	**Prioritization**	以情境為基礎(Context-Aware)判斷風險優先級,考量可攻擊性、業務影響、已部署環境。
修復治理	**Remediation & Governance**	自動建立工單、推送至 IDE、阻擋 Pull Request、產出合規報表,驅動組織完成修復。

2.2 ASPM 與傳統 AppSec 工具的根本差異

許多廠商聲稱自己是 ASPM,但實際上只是把既有的 SAST、SCA 工具重新包裝成「平台」。真正的 ASPM 必須具備以下特徵:

- **原生多引擎(Native Multi-Engine)** — 平台內建或深度整合 SAST、SCA、Secrets、IaC、Container、Pipeline 等多種掃描能力,而非僅是儀表板聚合。
- **跨階段風險關聯(Cross-Stage Correlation)** — 能將 SCM 的程式碼漏洞、CI/CD 的 Pipeline 風險、Runtime 的部署狀態,關聯為單一應用程式的完整風險視圖。
- **情境感知優先級(Context-Aware Prioritization)** — 基於可攻擊性、業務重要性、資料敏感度等因素動態調整風險優先級,而非僅依賴 CVSS 分數。
- **開發者原生體驗(Developer-Native)** — 深度整合 IDE、SCM、Chat(Slack、Teams)、ITSM(Jira、ServiceNow),讓修復成為開發者自然工作流的一部分。
- **開放架構(Open Architecture)** — 能整合企業既有工具的結果(如 Checkmarx、Snyk、Veracode),而非強制取代。這是真正的「平台」而非「工具集合」。

2.3 為什麼 Cycode 是 ASPM 類別的領導者

Cycode 於 2019 年創立於以色列,創辦團隊來自以色列軍方 8200 部隊與頂尖資安新創。初期聚焦於 SSCS(Software Supply Chain Security),2023 年戰略擴展為 Complete ASPM,成為 Gartner 首批入選 ASPM Market Guide 的代表廠商之一。

Cycode 的幾個關鍵里程碑:

- 2019 年成立,專注於 SCM 與 CI/CD 供應鏈安全
- 2022 年獲 Insight Partners B 輪領投 USD 56M
- 2023 年推出 Complete ASPM,整合 SAST、SCA、Secrets、IaC、Container、Pipeline 多引擎
- 2024 年收購 Bearer(SAST 領導者),強化程式碼分析能力
- 2024 年獲 Gartner、IDC、Forrester 均列為 ASPM 代表廠商
- 2025 年發表 AI-Native Remediation(AI 輔助自動修復)

三、Cycode 平台架構總覽

3.1 三層式平台架構

Cycode 採用三層式架構,從資產發現、多引擎掃描、到智慧關聯,提供完整的應用程式安全治理:

【表現層】 統一管理儀表板、開發者入口(Developer Portal)、IDE 外掛、CLI、API

【核心引擎層】 Risk Intelligence Graph(核心資料模型)、七大原生掃描引擎、AI 修復引擎、合規引擎

【整合層】 SCM(GitHub、GitLab、Bitbucket、Azure DevOps)、CI/CD(Jenkins、CircleCI、GitHub Actions)、Container Registry(ECR、ACR、Harbor)、Cloud(AWS、Azure、GCP)、ITSM(Jira、ServiceNow)、Chat(Slack、Teams)

3.2 Code-to-Cloud 完整覆蓋

Cycode 的覆蓋範圍涵蓋現代軟體開發生命週期的每一個階段:

SDLC 階段	Cycode 能力	具體保護範圍
設計 / 規劃	**威脅模型**	識別高風險應用程式類型、業務關鍵度標記
程式開發	**SAST + SCA + Secrets + IDE 整合**	程式碼漏洞、第三方套件風險、Hardcoded Secrets、即時 IDE 回饋
程式碼儲存	**SCM 安全**	分支保護政策、Code Review 強制、提交簽章、組織設定稽核
建置 / 打包	**CI/CD Pipeline 安全**	Pipeline Injection、惡意 Runner、第三方 Action 風險、Build Integrity
容器化	**Container Scanning + IaC**	容器映像漏洞、Dockerfile 最佳實務、Kubernetes 設定風險
部署 / 發布	**SBOM + 合規檢查**	SBOM 產出、License 合規、發布前最終檢查
Runtime / 生產	**Cloud 關聯 + 風險優先排序**	實際部署狀態、可攻擊性驗證、Runtime 資料回饋至 Code 端

3.3 部署與整合

SaaS 部署模式

Cycode 主要以 SaaS 形式提供,企業無需管理基礎設施。所有資料處理在 Cycode 雲端完成,提供 US、EU、APAC 區域選擇,符合 GDPR、PIPL 等資料主權要求。

原生整合清單(部分)

- SCM:GitHub、GitHub Enterprise、GitLab、Bitbucket、Azure DevOps、AWS CodeCommit

- CI/CD:Jenkins、CircleCI、GitHub Actions、GitLab CI、Azure Pipelines、TeamCity、Bamboo
- Container Registry:Docker Hub、ECR、ACR、GCR、Harbor、JFrog Artifactory
- Cloud:AWS、Azure、GCP(CSPM 整合)
- Kubernetes:EKS、AKS、GKE、OpenShift
- ITSM:Jira、ServiceNow、Linear、Azure DevOps Boards
- Chat:Slack、Microsoft Teams
- IDE:VS Code、IntelliJ、Visual Studio

3.4 合規與安全認證

- SOC 2 Type II
- ISO 27001
- GDPR / CCPA 合規
- 零程式碼外傳選項(僅傳輸 metadata 與分析結果)
- RBAC、SSO(SAML、OIDC)、完整稽核軌跡

四、Cycode 七大核心能力

4.1 SAST(靜態程式碼分析)

Cycode 於 2024 年收購 Bearer(業界知名 SAST 引擎),將自家 SAST 能力提升至業界領先水準。核心能力:

- 支援 20+ 種程式語言:Java、Python、JavaScript/TypeScript、Go、Ruby、C#、PHP、Rust、Kotlin 等
- 深度資料流分析(Data Flow Analysis),識別真實可利用漏洞,大幅降低誤報
- OWASP Top 10、CWE Top 25、PCI-DSS 合規檢查
- 隱私敏感資料追蹤(PII / PHI / PCI 資料流分析)
- IDE 即時回饋,開發者在寫程式時即看到提示
- AI 自動修復建議,支援 Pull Request 自動建立

4.2 SCA(軟體組成分析)

- 完整涵蓋主流套件管理器:npm、PyPI、Maven、RubyGems、Go Modules、Cargo、NuGet
- Reachability Analysis:判斷漏洞是否真的可被應用程式觸達,大幅減少誤報
- License 合規檢查,識別 GPL、AGPL 等需注意的授權條款
- Transitive Dependencies(遞移相依)深度分析
- Malicious Package Detection:識別 npm / PyPI 上的惡意套件(供應鏈攻擊防護)

4.3 Secrets Detection(機敏資料偵測)

- 200+ 種 Secrets 類型偵測:AWS Keys、GCP Service Account、Slack Token、SSH Keys、資料庫密碼等
- 歷史提交(Git History)完整掃描,識別過去已暴露但未處理的 Secrets
- Validated Secrets 驗證功能,即時確認該憑證是否仍有效,優先處理高風險項目
- 自動 Rotation 建議與雲端平台整合
- Pre-Commit Hook,阻止 Secrets 進入程式庫

4.4 IaC Security(基礎設施即程式碼安全)

- 支援 Terraform、CloudFormation、Kubernetes Manifest、Helm Charts、Ansible、ARM Templates
- CIS Benchmarks、NIST、PCI-DSS、HIPAA 合規對齊
- Policy-as-Code 自訂規則,符合企業特定資安標準
- 自動修復 Pull Request 建立,提供安全的 IaC 設定

4.5 Container Security(容器安全)

- 容器映像漏洞掃描(OS、應用程式相依、已知 CVE)
- Dockerfile 最佳實務檢查(Root 使用、未 Pin 版本、暴露 Port)
- Kubernetes Manifest 安全檢查(Privileged、HostNetwork、RBAC)
- Container Registry 整合,自動掃描新推送的映像

- SBOM 自動產出(SPDX、CycloneDX 格式)

4.6 Pipeline Security(CI/CD 安全)

這是 Cycode 的起家能力,也是業界最完整的 CI/CD 供應鏈安全方案:

- GitHub Actions、GitLab CI、Jenkins Pipeline 完整掃描
- 第三方 Actions 供應鏈風險評估(Typosquatting、Suspicious Maintainers)
- Pipeline Injection 偵測(Shell Injection via Untrusted Input)
- Runner 環境安全(Self-Hosted Runner 隔離、權限最小化)
- Build Integrity(SLSA Framework 符合性)
- 組織 SCM 設定稽核(分支保護、Secret 範圍、管理員權限濫用偵測)

4.7 ASPM Orchestration(應用程式安全編排)

這是 Cycode 作為 Complete ASPM 平台的核心價值——不只是提供多個工具,而是將所有工具的輸出(包括第三方工具如 Checkmarx、Snyk、Veracode)整合為統一的風險視圖:

- 整合第三方 AppSec 工具輸出,企業可保留現有投資
- 跨工具去重(Deduplication),同一漏洞不重複告警
- Risk Intelligence Graph 統一優先級計算
- 統一儀表板、統一報表、統一合規視圖
- 自動工單派發、SLA 追蹤、MTTR 指標

五、Risk Intelligence Graph:Cycode 的差異化引擎

如果說 Cycode 與其他「聲稱自己是 ASPM」的廠商有一個根本差異,那就是 **Risk Intelligence Graph(RIG)**。這是 Cycode 自行研發的圖形化資料模型,以圖資料庫為核心,將企業所有應用程式相關資產(Repositories、Packages、Containers、Pipelines、Deployments)及其關聯關係完整建模。

5.1 為什麼需要圖模型

傳統 AppSec 工具使用「表格式」資料模型——一張表記錄漏洞,另一張表記錄套件,再一張表記錄容器。要回答「這個 CVE 影響哪些生產環境應用程式?是否已對網際網路暴露?資料是否流經敏感路徑?」這類問題,需要跨多個工具手動拼接。

圖模型則不同。在 RIG 中:

- 一個 Repository 節點與它的 Commit、Author、Branch、Protection Rules 關聯
- 一個 Package 節點與使用它的 Repository、它的 CVE、它的 License、其 Transitive Dependencies 關聯
- 一個 Container 節點與建置它的 Pipeline、使用的 Base Image、部署到的 Cluster 關聯
- 一個 CVE 節點與受影響的 Package、已知 Exploit、對應的修復版本關聯

這讓 Cycode 能在毫秒級回答傳統工具難以回答的問題:

- 「這個 Log4Shell CVE 實際影響我的哪些生產環境服務?」
- 「這個開發者提交的 Commit 進入了哪些 Container Image?部署到哪些 Cluster?」
- 「這個被洩漏的 AWS Key 關聯到哪些 IaC、哪些雲端資源、有哪些權限?」

5.2 五維度風險優先級評分

Cycode 以 RIG 為基礎,對每個風險計算五個維度的評分,得出真正的「業務可執行優先級」:

- **嚴重性(Severity)** — CVE 原始分數與修復可行性
- **可攻擊性(Exploitability)** — 是否有已知 Exploit、是否在 CISA KEV 清單、Reachability 驗證
- **暴露度(Exposure)** — 是否對網際網路暴露、是否在公開 API、是否在生產環境
- **業務衝擊(Business Impact)** — 所在應用程式的業務關鍵度標記、處理的資料敏感度
- **資產關聯(Blast Radius)** — 受影響的應用程式、服務、使用者、資料範圍

這五個維度綜合產出的風險分數,比傳統 CVSS 分數準確得多。實務上,企業採用 Cycode 後,真正需要立即修復的「Critical」事件從每月數千筆收斂到數十筆,修復團隊能專注於最重要的問題。

> *「以前我們每月有 3,000 個『Critical』漏洞,根本修不完。Cycode 的 Risk Intelligence Graph 告訴我們其中只有 47 個真正重要——這讓我們團隊第一次看到希望。」— 某電商平台 AppSec 主管*

六、競爭格局分析:Cycode vs. Checkmarx vs. Snyk vs. Veracode

在應用程式安全市場,企業通常評估的幾個主要選項包括:傳統 SAST 領導者 Checkmarx、Developer-First 代表 Snyk、企業級 SAST 老牌 Veracode、以及開源 SAST SonarQube。本章節提供完整的比較分析。

6.1 Cycode vs. Checkmarx

Checkmarx 是 SAST 市場的老牌領導者之一,在大型企業擁有穩固的市占。然而其產品架構反映了 2010 年代的設計思維,與現代 DevSecOps 需求有明顯落差:

Checkmarx 的痛點:

- **掃描時間極長** — 大型應用程式掃描常需數小時,無法整合進快速 CI/CD 流程。
- **誤報率極高** — 業界公認誤報率達 30-50%,開發者對告警失去信任。
- **模組分散** — Checkmarx One、SAST、SCA、IAST、API Security 等產品為不同模組,需分別授權,整合不佳。
- **開發者體驗差** — IDE 整合弱、無即時回饋、修復建議過於技術性,開發者抗拒使用。
- **不具備真正 ASPM 能力** — 無法跨工具去重、無 Risk Intelligence Graph、無跨階段關聯。

Cycode 相對 Checkmarx 的優勢:

- **掃描速度快 5-10 倍** — 分鐘級完成,無縫整合 CI/CD
- **誤報率降低 60% 以上** — Bearer 引擎 + Reachability Analysis 雙重降噪
- **一體化平台** — SAST、SCA、Secrets、IaC、Container、Pipeline 共用 Risk Intelligence Graph
- **開發者原生體驗** — IDE 即時回饋、AI 自動修復 PR、Slack 對話式互動
- **真正 ASPM 能力** — 可整合 Checkmarx 等第三方工具輸出,企業無需放棄現有投資

6.2 Cycode vs. Snyk

Snyk 以「Developer-First」作為差異化訴求,快速在開發者社群中獲得採用。然而,Snyk 在企業級場景面臨明顯侷限:

Snyk 的痛點:

- **SAST 引擎相對薄弱** — Snyk 強項在 SCA(套件掃描),但自家 SAST(Snyk Code)能力不如專業 SAST 廠商。
- **Pipeline / SCM 安全覆蓋不足** — 缺乏 CI/CD 供應鏈安全、SCM 組織設定稽核等現代需求。
- **授權成本飆升** — 按開發者數計價,大型企業年費常達 USD 500K-2M。
- **缺乏真正的跨階段關聯** — Snyk Code、Snyk Open Source、Snyk Container、Snyk IaC 為不同模組,內部整合有限。
- **非 ASPM 原生架構** — Snyk 近期才開始加入 ASPM 概念,但架構仍以單點工具為主。

Cycode 相對 Snyk 的優勢:

- **SAST 引擎更強** — 整合 Bearer 引擎,業界公認 SAST 領導者
- **Pipeline + SCM 全覆蓋** — Cycode 起家於 SSCS,這是最完整的 CI/CD 供應鏈安全方案

- **更透明可預測的授權模式** — 以應用程式資產數量計價,不綁定開發者人數
- **原生 ASPM 架構** — Risk Intelligence Graph 從第一天就是核心,非事後加上

6.3 Cycode vs. Veracode

Veracode 是另一家 SAST 老牌廠商,以「Binary SAST(對編譯後的二進位檔掃描)」作為差異化訴求,在金融業有穩固客戶基礎。但其架構同樣反映了傳統 AppSec 思維:

Veracode 的痛點:

- **Binary SAST 延遲高** — 需編譯後才能掃描,無法在開發階段即時回饋,不適合現代 DevOps 節奏。
- **缺乏 SCM 與 Pipeline 安全** — 與 Checkmarx 類似,視野侷限於程式碼本身,不涵蓋現代軟體供應鏈。
- **開發者體驗落後** — IDE 整合與 IDE 即時掃描能力有限。
- **授權成本高** — 企業級價格定位,對中型企業負擔沉重。

Cycode 相對 Veracode 的優勢:

- **Source-Based SAST** — 直接掃描源碼,IDE 即時回饋,不需等待編譯
- **完整供應鏈覆蓋** — SCM、Pipeline、Container、IaC 全面覆蓋
- **現代開發者體驗** — Shift-Left 真實實踐,開發者願意使用
- **更可預測的 TCO** — 透明的資產授權模式

6.4 Cycode vs. SonarQube

SonarQube 是開源世界最廣泛使用的 Code Quality + SAST 工具,有免費版與商業版(SonarCloud / SonarQube Enterprise)。但其核心定位仍是程式碼品質,而非企業級資安:

SonarQube 的定位與侷限:

- 核心定位是 Code Quality(可維護性、可讀性),資安只是副產品
- 不具備 SCA、Container、IaC、Pipeline Security 能力
- 無 Risk Intelligence Graph 或跨工具關聯
- 無企業級合規報表能力
- 需自行部署、維運(Community 版)或採購 Enterprise 版

SonarQube 適合純粹的程式碼品質檢查,但若企業目標是完整 ASPM,SonarQube 只能作為補充工具,不能取代 Cycode 的角色。Cycode 甚至可以整合 SonarQube 的輸出作為 ASPM 訊號源之一。

6.5 綜合功能比較表

能力	Cycode	Checkmarx	Snyk	Veracode
SAST	★★★★★	★★★★★	★★★	★★★★
SCA	★★★★★	★★★★	★★★★★	★★★
Secrets Detection	★★★★★	★★★	★★★	★★
IaC Security	★★★★★	★★★	★★★★	★★

能力	Cycode	Checkmarx	Snyk	Veracode
Container Security	★★★★	★★	★★★★	★★
Pipeline / CI/CD 安全	★★★★★	★	★★	★
SCM 組織設定稽核	★★★★★	無	無	無
Risk Intelligence Graph	原生核心	無	部分	無
跨工具關聯與去重	★★★★★	★★	★★	★★
開發者 IDE 整合	★★★★★	★★★	★★★★★	★★★
AI 自動修復	★★★★★	★★★	★★★★	★★★
ASPM 完整度	★★★★★	★★	★★★	★★
掃描速度	分鐘級	小時級	分鐘級	小時級
典型誤報率	**10-15%**	30-50%	20-30%	25-40%

七、雲端工作負載安全:Cycode vs. Twistlock vs. Sysdig

Twistlock(現為 Palo Alto Prisma Cloud Compute)、Sysdig 與 Aqua Security 是雲端工作負載保護(CWPP / Container Security)領域的代表廠商。這些工具與 Cycode 存在部分功能重疊(如 Container Scanning),但核心定位有根本差異。

7.1 定位差異:Shift-Left vs. Runtime Protection

Cycode 的定位: 以 Shift-Left 為核心的 ASPM 平台,聚焦於「防止風險進入生產環境」——在程式碼、Pipeline、Container Build 階段就攔截問題。

Twistlock / Sysdig 的定位: 以 Runtime Protection 為核心的 CWPP 平台,聚焦於「生產環境的即時防護」——監控容器、Kubernetes、雲端工作負載的運行時異常與威脅。

這兩類工具是互補關係,非競爭關係。成熟的企業同時需要 ASPM(代表 Cycode)+ CWPP(代表 Upwind、Prisma Cloud、Sysdig)來覆蓋完整的 Code-to-Cloud 安全。

7.2 Cycode vs. Twistlock(Prisma Cloud Compute)

Twistlock 於 2019 年被 Palo Alto 收購,整合進 Prisma Cloud 平台。原本定位為容器安全,後擴展為 CNAPP(Cloud-Native Application Protection Platform)。其特徵:

Twistlock / Prisma Cloud 的焦點:

- 容器映像漏洞掃描(Registry + Runtime)
- Kubernetes Runtime Protection
- 主機 OS 安全
- Serverless Security
- CSPM(Cloud Security Posture Management)

Twistlock 的侷限(相對於 ASPM):

- SAST / SCA 能力薄弱,無法取代專業 AppSec 工具
- 無 Secrets 深度掃描能力
- 無 SCM 組織設定與 Pipeline 供應鏈安全
- Risk Intelligence Graph 聚焦於 Cloud Runtime,不涵蓋 Code 階段
- 被 Palo Alto 整體生態綁定,授權費高昂

結論:

Twistlock 與 Cycode 是互補而非取代關係。Cycode 負責 Code-to-Build 階段,Twistlock 負責 Runtime 階段。完整的現代化雲端安全架構應同時涵蓋兩者。

7.3 Cycode vs. Sysdig

Sysdig 起源於開源的 Falco 專案(CNCF Runtime Security 標準),其商業產品 Sysdig Secure 專注於:

Sysdig 的焦點:

- Runtime Threat Detection(以 Falco 為基礎)
- Container Image Scanning
- Kubernetes Security Posture

- Cloud Detection & Response
- Incident Response 能力強

Sysdig 的侷限(相對於 ASPM):

- SAST 幾乎沒有(僅有極基礎的 Shift-Left 掃描)
- SCA 能力有限
- 無 Secrets、SCM 安全、Pipeline 安全等完整 ASPM 能力
- 核心在 Runtime Telemetry,非 Code 階段治理

結論:

Sysdig 是優秀的 Runtime Security 工具,特別是對於重度使用 Kubernetes 的企業。但它不是 ASPM 替代品。企業應同時部署 Cycode(ASPM / Shift-Left)+ Sysdig 或 Upwind(CWPP / Runtime)。

7.4 Cycode + Runtime 安全的完整架構

SaaSPodium 推薦的現代化雲端應用程式安全完整架構:

階段	推薦方案	涵蓋範圍
程式碼 / 建置 (Shift-Left)	**Cycode ASPM**	SAST、SCA、Secrets、IaC、Container Scan、Pipeline Security、SCM 治理
Runtime / 部署 (Shift-Right)	**Upwind Runtime CNAPP**	容器 Runtime 威脅偵測、K8s Posture、雲端工作負載保護、eBPF 即時觀測
資料管線整合	**Cribl SDPP**	將 Cycode + Upwind 告警統一路由至 SIEM,減少重複告警
網路流量偵測	**Corelight NDR**	東西向與南北向網路流量威脅偵測,補足端點盲區

這就是 SaaSPodium「Cloud-Native Security Pack」的架構主張——Cycode + Upwind + Cribl + Corelight 組成的完整現代化雲端資安堆疊。

八、開發者體驗:Shift-Left 的真實實踐

Shift-Left(左移)是過去十年 DevSecOps 的核心口號,但真正實踐成功的企業不多。主要原因是多數資安工具的設計未真正考量開發者體驗,結果造成開發團隊的抗拒。Cycode 從第一天就將「開發者原生體驗」列為核心設計原則。

8.1 開發者每天接觸 Cycode 的介面

- **IDE 即時回饋** — VS Code、IntelliJ、Visual Studio 外掛,在寫程式時即時提示風險
- **Pull Request 自動評論** — 開 PR 時 Cycode 自動留言指出風險與建議修復,非獨立平台
- **AI 自動修復 PR** — Cycode 主動建立修復 PR,開發者只需 Review 與合併
- **Slack / Teams ChatOps** — 透過對話式介面詢問、處理風險,不需切換到資安平台
- **CLI 工具** — 本地端執行掃描,整合進開發者既有工作流

8.2 實測數據:Cycode 如何改變開發團隊行為

某 200 位開發者的企業導入 Cycode 後 6 個月的實測數據:

- 告警忽略率從 65% 降至 12%(開發者願意處理告警)
- 平均修復時間(MTTR)從 18 天縮短至 3 天
- 新發漏洞在進入生產環境前被攔截比例從 28% 提升至 89%
- 開發者對資安工具滿意度從 2.1/5 提升至 4.3/5

> *「Cycode 是第一個讓我們開發者主動打開的資安工具。以前他們看到資安工單就躲,現在他們在 Slack 會主動問『Cycode 上有沒有我該處理的?』」 — 某 SaaS 企業 VP of Engineering*

九、典型應用場景

9.1 多工具整合與去重

某金融業客戶同時使用 Checkmarx(SAST)、Black Duck(SCA)、Prisma Cloud(Container),每月告警超過 50,000 筆。導入 Cycode 作為 ASPM 整合層後,透過 Risk Intelligence Graph 關聯與去重,真正需要處理的「Top Priority」風險收斂至每月 340 筆,AppSec 團隊效率提升 10 倍以上。

9.2 供應鏈攻擊防護

某軟體公司在 2024 年 npm 供應鏈事件中,透過 Cycode 的 Malicious Package Detection 與 SCM 即時監控,於惡意套件被發布後 2 小時內偵測並自動阻擋開發者安裝,未受影響。

9.3 Secrets 外洩緊急處置

某企業開發者誤將 AWS Key 提交至 public repository。Cycode 於 30 秒內偵測、自動觸發 AWS Key Rotation、通知開發者、建立修復 PR。從暴露到完成處置總時間 8 分鐘,避免潛在雲端資源濫用。

9.4 合規稽核加速

某金融客戶過去每次 ISO 27001 / PCI-DSS 稽核需耗費 AppSec 團隊 4 週準備時間。導入 Cycode 後,合規報表一鍵產出,準備時間縮短至 3 天。

9.5 從 Checkmarx 逐步遷移

某製造業客戶原本使用 Checkmarx 10 年,年費 USD 380K。導入 Cycode 後,先以並行模式運作 6 個月驗證能力,確認 Cycode 覆蓋所有原 Checkmarx 規則後完成切換,年度總成本降低 55%,且新增了 SCM / Pipeline / Container 安全覆蓋。

9.6 M&A 併購資安盡職調查

某科技集團在併購 target 前,使用 Cycode 對 target 的所有 Repository、Pipeline、Container 進行快速資安評估,在一週內產出完整的技術債與資安風險報告,協助併購估值談判。

十、導入流程與最佳實務

SaaSPodium 提供標準化的四階段 Cycode 導入方法論,典型導入週期 6-10 週:

階段	時程	主要活動
Phase 1 Discover	第 1-2 週	企業 SDLC 現況盤點、現有 AppSec 工具清單、SCM/CI/CD 架構、風險優先事項識別。
Phase 2 Connect	第 3-4 週	SCM、CI/CD、Container Registry、Cloud 整合連線,建立完整資產清單,RIG 初始化。
Phase 3 Tune	第 5-8 週	風險優先級規則客製化、開發者工作流整合、Jira/Slack 整合、誤報校準、團隊培訓。
Phase 4 Scale	第 9 週及之後	完整上線、SLA 定義、持續優化、進階能力啟用(AI 修復、Pipeline Security 深化)。

導入最佳實務

- **從 Top 10 關鍵應用程式切入** — 先覆蓋最關鍵的 Repository 與 Pipeline,不求一次全面上線
- **並行模式驗證** — 若既有使用 Checkmarx/Snyk/Veracode,先以並行運作 2-3 個月驗證 Cycode 覆蓋度
- **邀請開發團隊參與** — AppSec 與 Dev 團隊共同參與配置,確保開發者體驗符合預期
- **善用 AI 修復** — 越早啟用 AI 自動修復,團隊越能從「人工處理」轉為「審閱 AI 產出」

十一、商業價值與 TCO 分析

以下為一家典型中大型企業(500 位開發者、200 個 Repository、30 個微服務)導入 Cycode 後的商業價值分析:

指標	導入前	導入 Cycode 後
AppSec 工具總數	7-10 個	**1-2 個(Cycode + 既有深化工具)**
每月告警量	40,000+ 筆	**200-400 筆(高優先級)**
誤報率	30-45%	**10-15%**
平均修復時間(MTTR)	18-22 天	**3-5 天**
新漏洞上生產環境比例	25-30%	**10-12%**
合規稽核準備時間	4 週	**3 天**
AppSec 工具年度授權總費用	USD 1.2M	**USD 480K(-60%)**
開發者對資安工具滿意度	2.0-2.5 / 5	**4.2-4.5 / 5**
年度資安事件(與 AppSec 相關)	平均 3-5 起	**0-1 起**

綜合而言,典型企業導入 Cycode 後:

- 直接授權費用節省:USD 500K-800K / 年
- AppSec 團隊生產力提升 3-5 倍
- 開發者滿意度顯著提升
- 資安事件降低 70-85%
- ROI 達成時間通常在 6-9 個月內

十二、SaaSPodium 的服務承諾

作為 Cycode 在台灣與亞太區的授權代理商,SaaSPodium 提供完整的 Cycode 解決方案服務:

- **免費 AppSec Maturity Assessment** — 協助企業評估現有 AppSec 成熟度,量化工具重複與盲區,作為決策依據。
- **Cycode 授權諮詢** — 依 Repository 數量、開發者規模、產業特性,提供最適合的 Cycode 授權方案。
- **結構化 POC 服務** — 提供 4 週 POC,以企業實際 Repository 驗證 Cycode 效能,對比既有工具。
- **導入顧問服務** — 認證顧問團隊協助完成 SCM/CI/CD 整合、Risk Intelligence 客製化、開發者工作流整合。
- **從 Checkmarx / Veracode / Snyk 遷移服務** — 針對現有使用其他 AppSec 工具的企業,提供並行驗證、規則轉換、歷史資料遷移。
- **中文教育訓練** — 針對 AppSec 團隊、開發主管、開發者的分層培訓,建立組織內部 Cycode 能力。
- **長期維運支援** — 中文技術支援、季度健檢、新功能導入、年度 AppSec 成熟度檢視。
- **Cloud-Native Security Pack 整合方案** — Cycode + Upwind + Cribl + Corelight 完整現代化雲端資安堆疊,提供一站式架構諮詢。

結語

應用程式安全已經進入新時代。以前企業購買 SAST、SCA、Container Scanner 等獨立工具,拼湊出所謂的 AppSec 架構;現在,ASPM 取代了這種拼裝模式,提供一個統一、智慧、開發者原生的平台。

Cycode 不是在既有 SAST 市場中多一個選項,而是定義了 ASPM 這個新類別本身:

- 以 Risk Intelligence Graph 重新定義應用程式風險視圖
- 以 Complete ASPM 取代多工具拼裝架構
- 以 Developer-Native 體驗真正實現 Shift-Left
- 以 Code-to-Cloud 完整覆蓋取代片段式保護

SaaSPodium 作為 Cycode 授權代理商,協助企業從評估、POC、導入到長期優化,打造現代化的應用程式安全治理架構。我們不只銷售 Cycode,更是您 DevSecOps 轉型的長期夥伴。

下一步:重新定義您的應用程式安全

免費 AppSec Maturity Assessment | 4 週 POC | 工具遷移諮詢

SaaSPodium 拓維雲智資安顧問團隊

Authorized Distributor of Cribl, Device42, Corelight, Cycode and Upwind

| Freshworks Premier Partner |