IT智能化最佳夥伴

METAGE 邁達特

GitHub Enterprise × Entra ID：14 天企業現代化導入

現狀痛點與治理危機

- 份管理斷裂 (Identity Silo)：
  - 離職人員 GitHub 權限回收依賴人工，存在平均 48 小時以上的權限空窗期。
  - 企業無法統一控管開發者是否啟動 MFA，造成身份盜用風險。
- 憑證管理的噩夢 (Secret Sprawl)：
  - Service Principal Secrets 散落在數百個 Repo 中，無人知道誰有權限、何時過期。
  - 一旦發生外洩，難以在第一時間鎖定受影響範圍並進行大規模重置。
- 合規稽核黑盒 (Audit Black Hole)：
  - 開發行為（PR/Merge/Push）無法即時對應到企業員工 ID。
  - 缺乏統一的資安門檻（Quality Gates），導致漏洞修補全憑開發者自律。

METAAGE

方案核心價值：GitHub 與 Entra ID 的實際整合方式

- Identity Lifecycle (SCIM)：

  由 Entra ID 主控帳號生命週期，自動佈建與停用 GitHub 使用者，避免人為遺漏。

- 企業級登入控管 (SSO / MFA)：

  強制使用 Entra ID SAML SSO，完整沿用既有 MFA 與條件式存取 (Conditional Access) 政策。

- 無密鑰部署 (OIDC)：

  GitHub Actions 透過 OIDC 直接向 Azure 取得短效權限，不再保存任何長效密鑰。

- 組織與權限一致性：

  Entra ID Group 自動對映 GitHub Teams，確保 Repo 權限與企業組織同步。

METAGE

14 天實施流程

天數	階段名稱	關鍵任務：以 Entra ID 為治理核心
Day 01–04	評估與規範規劃	盤點 Entra ID 結構，設計 GitHub Teams 與 Group 對映原則與命名規範。
Day 05–06	環境與身份整合	完成 SAML SSO 與 SCIM 設定，建立 GitHub 與 Azure 的 OIDC 信任關係。
Day 07	【教育訓練 Day 1】	**基礎協作工作坊**：Entra ID 登入、PR 協作規範、企業分支策略實務。
Day 08	【教育訓練 Day 2】	**進階安全工作坊**：OIDC 無密鑰部署實作、GAS 安全門檻與稽核回溯。
Day 09–12	自動化與治理落地	建立 Reusable Workflows 模板、Repo Templates 與治理自動化腳本。
Day 13–14	驗收與導入確認	驗證離職回收即時性、驗收無密鑰部署流程、交付維運 Runbook。

METAGE

教育訓練：全方位賦能工作坊(Day 07–08)

- 上午：現代化 Git 協作心法 (Standardization)
  - 標準化起手式：實作使用 Enterprise Repository Templates，確保新專案之目錄結構、.gitignore 與內部文檔（README）從第一天就符合企業架構。
  - 命名美學與規範：嚴格執行 Branch（分支）、Tag（標籤）與 Commit（提交）的命名規範，導入 Conventional Commits 確保變更歷史可追蹤且具備自動化潛力。
  - 文件驅動開發：強化 .gitignore 配置以避免垃圾檔案入庫，並建立標準化技術文件模板。
- 下午：Pull Request 協作與品質防線 (Quality Control)
  - 標準協作流實踐：深度演練 GitHub Flow 分支模型（main / feature），建立標準的程式碼生命週期觀念。
  - PR 高效評論 (Peer Review)：學習如何進行具建設性的 Code Review，運用 Suggestions (即時修改建議) 縮短溝通往返，建立同儕審核文化。
  - 合併管控與保護：解析 Squash Merge 對保持主幹簡潔的價值，並實作分支保護規則 (Branch Protection Rules) 以確保未經審核之代碼絕不進入主幹。

METAGE

教育訓練：進階功能與 DevSecOps 安全實戰

- **上午：GitHub Actions 自動化加速 (Efficiency & OIDC)**
  - CI/CD 基礎實務：自動化建構（Build）、單元測試（Test）與靜態品質檢查，確保代碼交付之一致性。
  - 去密鑰化部署 (Secretless)：配置 Azure Workload Identity，演練如何在不儲存任何 Secret（如 Service Principal Key）的情況下，透過 OIDC 進行跨雲安全部署。
  - 效能優化策略：利用 Actions Cache 技術縮短依賴項下載時間，極大化開發團隊的作業效能。
- **下午：GitHub Advanced Security (GAS) 安全內建 (Defense)**
  - 靜態掃描 (CodeQL)：學習在 PR 合併前自動觸發語義分析掃描，及早發現潛在邏輯漏洞與資安風險。
  - 密鑰防線演練 (Push Protection)：親自體驗 Push Protection 攔截機制，防止密鑰、Token 等敏感資訊在提交階段意外流出。
  - 軟體供應鏈安全管理：自動產出 SBOM (軟體物料清單) 並監控第三方依賴套件風險 (Dependabot)，確保企業軟體資產之安全性與透明度。

METAGE

預期專案效益

- 營運效能提升：

  新專案啟動時程縮短 80% (從數天縮短至數小時)。

- 安全合規達成：

  部署流程 100% 移除長效密鑰；100% 程式碼自動通過資安掃描。

- 人力成本降低：

  SCIM 整合讓權限異動人工作業減少 90%。

- 知識沉澱：

  標準化模板讓研發經驗 100% 留在企業內部，而非隨個人離職而消失。

METAGE

驗收成果與交付項目

一、 核心平台整合技術文件

- Entra ID × GitHub 整合架構與設定手冊：詳細記錄 SAML SSO 配置、Enterprise Managed Users (EMU) 階層設計、以及組織與 Tenant 的關聯設定。
- SCIM 帳號生命週期自動化規範：定義 Entra ID Group 到 GitHub Teams 的對映矩陣，含帳號自動佈建 (Provisioning) 與即時回收機制設定。

二、 現代化 DevSecOps 標準範本庫

- Azure OIDC 無密鑰部署模板 (Best Practices)：
- Web App：包含服務環境變數配置與插槽切換流程。
- AKS (Kubernetes)：包含 Helm Chart 整合與 Workload Identity 權限對應設定。
- Storage：包含靜態網站佈署與 Blob 資料傳輸安全範本。
- Reusable Workflows (企業級共享流水線)：交付符合資安規範的建置 (Build)、測試 (Test) 與掃描 (Scanning) 標準 YAML 模板。

METAGE

驗收成果與交付項目

三、 平台營運與維運維護手冊 (Runbook)

- 系統管理維運手冊 (Admin Runbook)：
  - 權限異動與角色申請流程：詳列從 Entra ID 到 GitHub 端的權限生命週期管理規範。
  - Audit Log 稽核日誌導出說明：建立與企業 SIEM 系統對接或定期匯出的標準作業程序。
  - 緊急存取 (Break-glass) 機制：定義在極端情況下的緊急權限授權流程，確保業務連續性。
- 開發者標準操作手冊 (Developer Playbook)：
  - 新專案啟動 (Scaffolding) 指引：引導開發者透過 Repo Templates 快速建立符合企業規範的新專案。
  - OIDC 無密鑰憑證授權流程：說明如何正確配置 Azure Workload Identity，捨棄傳統 Secret 進行安全部署。
  - 安全漏洞與品質修補指引：提供開發者處理自動化檢查回饋、分支衝突及代碼修正的實務手冊。

METAGE

METAGE 邁達特

Thank You

IT智能化最佳夥伴