Micro Focus

Fortify 軟體安全性內容

2020 更新 3

2020 年 9 月 25 日

關於 Micro Focus Fortify Software Security Research

Fortify Software Security Research 團隊將尖端研究成果轉變為可為 Fortify 產品組合 (包括 Fortify Static Code Analyzer (SCA)、Fortify WebInspect 和 Fortify Application Defender) 增添動能的安全情報。現在，Micro Focus Fortify 軟體安全性內容能夠跨 27 種程式設計語言支援 1,032 個弱點類別，且涵蓋 100 多萬個單獨 API。

瞭解詳情：https://software.microfocus.com/en-us/software/security-research

Fortify Software Security Research (SSR) 欣然宣布，現已推出以下產品的更新：Fortify Secure Coding Rulepacks (英文，2020.2.0 版)、Fortify WebInspect SecureBase (可透過 SmartUpdate 取得)，以及 Fortify Premium Content。

Micro Focus Fortify Secure Coding Rulepacks [SCA]

隨著此發佈，Fortify Secure Coding Rulepacks 能夠跨 27 種程式設計語言偵測 815 種獨特的弱點類別，且涵蓋 100 多萬個單獨 API。概括地說，此發佈包含下列項目：

.NET System.Text.Json 支援 (3.1 版)[1]

System.Text.Json 命名空間為 Newtonsoft 的 Json.NET 套件提供了一種高速、內建且符合標準的替代方案。此外，System.Text.Json 命名空間還提供用於讀取和寫入編碼為 UTF-8 的 JSON 文字的類型。支援包括透過各種序列化/還原序列化方法的資料流，並增加以下類別的涵蓋範圍：

- Cross-Site Scripting: Reflected
- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation
- Privacy Violation
- System Information Leak: External
- System Information Leak: Internal

Kotlin 協同程式和標準程式庫支援 [2]

擴大了 Kotlin 標準程式庫的涵蓋範圍，已加入對協同程式的支援。協同程式提供豐富靈活的 API，可用於開發並行和非同步程式碼。此更新在使用協同程式的應用程式中新增了改進的資料流分析。

Kotlin Java 互通性和 Android 改進功能 [3]

針對既包含 Java 也包含 Kotlin 原始程式碼的專案以及使用 Kotlin 開發的 Android 應用程式，在現有弱點類別中改進了語言支援。

Java 14 改進功能 [4]

在現有弱點類別中進行改進，以支援 Java 14 以下版本的變更。

ECMAScript 版本更新 [5]

ECMAScript 版本的規則涵蓋範圍已升級，現在可支援 ECMAScript 2020 之前的所有版本。

Go Logrus (1.6.0 版)

Logrus 是 Go 的結構化記錄器，API 與標準程式庫記錄器完全相容。Logrus 是一種常見的記錄套件，可能對企業應用程式帶來記錄相關風險。支援的類別包括：

---

[1] System.Text.Json 命名空間因設計之故，不容易遭受 Newtonsoft Json.NET 套件中所報告的 JSON Injection 問題影響。
[2] 此更新不支援實驗性協同程式 API，例如使用 select 運算式的實驗性協同程式 API。
[3] 需要 SCA 20.2 或更新版本。
[4] 需要 SCA 20.2 或更新版本。
[5] 完整支援需要 SCA 20.2 或更新版本。

- Poor Logging Practice: Use of a System Output Stream
- Privacy Violation
- System Information Leak

Dockerfile 支援 [6]

Docker 允許在沙箱環境 (也稱為容器) 中封裝程式碼及其相依項目，以便在任何運算環境中執行。Dockerfile 是容器映像的配置。此更新解決了 Dockerfile 中最常見的弱點。
初步支援涵蓋現有的「Password Management: Password in Configuration File」類別和下列新類別：

- Dockerfile Misconfiguration: Default User Privilege
- Dockerfile Misconfiguration: Privileged Container
- Dockerfile Misconfiguration: Privileged Port
- Dockerfile Misconfiguration: Sensitive Host Discovery
- Dockerfile Misconfiguration: SSH Service

DISA STIG 4.11

為了在合規領域支援我們的聯盟客戶，已新增 Micro Focus Fortify Taxonomy 與 Defense Information Systems Agency (DISA) Application Security and Development STIG 4.11 版之間的關聯性。

其他勘誤

在此發佈中，我們持續盡可能投入一切資源，來確保我們可以降低誤報問題數，並提升客戶稽核問題的能力。客戶還會看到與下列各項相關回報問題的變更：

17.x 以前的 SCA 版本廢止通知：
致我們使用者的通知，這將是支援 SCA 17.x 之前版本的最後 Rulepack 版本。在下一個版本中，SCA 17.x 之前版本將不會載入 Rulepack。此時將會要求降級 Rulepack 或升級 SCA 版本。
在未來的版本中，我們將繼續支援 SCA 的最後四個主要版本。

即將過時的對應 - 安全性技術實作指南 3.x 版 (STIG 3.x)：
多年來，軟體安全性研究團隊一直持續建立從 Fortify 類別至外部標準和最佳實務的對應。因此，即使不會使用許多舊版本，我們目前所支援之各種標準的版本數量也非常可觀。
所以，我們現在使用 externalmetadata.xml 檔案中的新 XML 屬性，將 4.0 版之前的 STIG 版本標記為「已過時」。這項變更意味著從 20.2 版開始的 Fortify Software Security Center 和 SCA 應用程式將預設不顯示這些舊版本。如果需要檢查這些舊版本，則解決方法是變更對應，將 obsolete 設為「false」，或完全移除該屬性。

[6] 需要 SCA 20.2 或更新版本。

黑名單和白名單的命名：
為符合此領域的趨勢，Micro Focus Fortify 旗下的 Software Security Research 團隊已決定停止使用「黑名單」和「白名單」的術語，並改用較為中性且語義具體的術語。我們決定分別改用更具描述性的術語，也就是「拒絕清單」和「允許清單」。
這項變更也意味著以下類別名稱已更改，並可能導致移除和新問題的產生：

- 「Deserialization Bad Practice: Blacklist」現在改名為「Deserialization Bad Practice: Deny List」
- 「Django Bad Practices: Blacklisted Attributes」現在改名為「Django Bad Practices: Attributes in Deny List」
- 「Setting Manipulation: User-Controlled Whitelist」現在改名為「Setting Manipulation: User-Controlled Allow List」

弱式加密：不安全的作業模式更新
我們認為使用對稱式加密的加密區塊鏈結 (CBC) 模式不再安全。自此版本開始，針對目前支援的程式庫，只要使用 CBC 就會回報為「弱式加密：不安全的作業模式」。就 BEAST、POODLE、Zombie POODLE、GOLDENDOODLE、LUCKY13 和 Padding Oracle 攻擊而言，根據目前已知的密碼編譯研究，我們已確定 CBC 不再安全。

誤報改進功能：
我們持續傾聽客戶的意見，致力於改善誤報率。在此版本中，我們完成以下工作，以減少誤報次數：

- Cross-Site Request Forgery
  進行減少以納入其他類型的 CSRF 權杖，以及在 ASP.NET 隱藏欄位內
- Server-Side Request Forgery
  在某些 Spring 應用程式中移除
- Poor Logging Practice: Use of a System Output Stream
  當內容以指令行應用程式或公用程式為基礎時移除。
- Dead Code：未使用的欄位
  lambdas 內部
- Privacy Violation
  已移除 Java、Scala 和 Kotlin lambdas 中涉及密碼的重複問題
- 現在，將 Java 系統屬性視為內建屬性進行單獨處理，從而在許多情況下減少了誤報，並且可能發現新的系統資訊洩漏。

預覽：[7]
用於 Null Dereference 和 Redundant Null Check 的規則已重新設計，以減少誤報率。尤其是針對編寫自訂規則以處理內部 Null 檢查功能增加了相關功能。
但是，目前尚不清楚這些優點是否具有普遍性。因此，預設會在 SCA 20.2 中關閉這些可用的改進功能。如果您想測試這些改進功能，請聯絡客戶支援。

[7] 需要 SCA 20.2 或更新版本。

Micro Focus Fortify SecureBase [Fortify WebInspect]

Fortify SecureBase 將數千個弱點的檢查，與在透過 SmartUpdate 立即取得的以下更新中引導使用者的原則結合在一起：

弱點支援

弱式密碼編譯簽章：使用者控制的金鑰來源

Key Confusion 攻擊會威脅 JSON Web 權杖 (JWT) 的完整性。如果 JWT 實作未慎選金鑰來源以驗證從用戶端接收的權杖，攻擊者便能趁機偽造權杖有效承載，導致 JWT 實作容易遭受這些攻擊。此版本包括一項檢查，可用來評估使用 JWT 對 key confusion 攻擊的效果。

JSON Web Token: Missing Protection Claims

未實作保護宣告 (例如「 jti」、「exp」和「aud」) 的 JSON Web 權杖可能容易遭受模擬和重新執行攻擊。此版本包括多項檢查，可用來偵測掃描過程中遇到的權杖是否缺少這些宣告。

Access control: Authorization Bypass

JSON Web 權杖 (JWT) 提供防止資料遭到篡改的機制，因為其包含的資訊會使用 HMAC 或 RSA 演算法進行數位簽署。簽名驗證失敗的應用程式可能容易遭受未經授權的存取和模擬攻擊。此版本包括多項檢查，可用來評估 JWT 實作是否存在遺失和任意簽名攻擊。

OAuth2: Insufficient state Parameter Entropy

若 OAuth2 通訊協定未使用具有足夠複雜度 (entropy) 的「state」參數值時，便容易遭受模擬攻擊。此版本包括一項檢查，可用來評估 state 參數是否無法滿足足夠複雜度的要求。

OAuth2: Insufficient Refresh Token Revocation

在重新發出新的存取權杖之前，未驗證用戶端 ID、用戶端密碼和更新權杖到期的授權伺服器，可能容易遭受模擬攻擊。此版本包括一項檢查，可用來偵測是否可以使用更新權杖來產生多個存取權杖，而無需用戶端 ID 和用戶端密碼。

OAuth2: Insufficient Authorization Code Expiration

若 OAuth2 授權伺服器未能及時使授權碼到期，便可能受到模擬攻擊，因為它們可能繼續為被盜的授權碼授予新的有效存取權杖。本版本包含一項檢查，可用來偵測授權碼濫用。

Cross-Site Request Forgery

無法在 OAuth2 授權流程中使用「state」參數的應用程式，很容易遭受模仿用戶端的未經授權操作。此外，將 Cookie 儲存用於 JSON Web 權杖 (JWT) 的應用程式也會面臨類似 CSRF 攻擊的風險。此版本包含多項檢查，可用於偵測容易遭受 CSRF 的 JWT 和 OAuth2 執行個體。

弱式密碼編譯簽章：金鑰大小不足

JSON Web 權杖包含數位簽署的資訊，用以證明承載的完整性。 如 HS256 之類具有弱式密碼的對稱式演算法可能會被暴力破解，進而讓攻擊者可以產生任意承載，並使用有效金鑰對其進行簽署。此版本包括一項檢查，可用來偵測是否使用弱式演算法 (如 HS256) 對權杖進行簽署。

SQL Injection[8]

SQLite 是一個 C 程式庫，提供輕量級的磁碟式資料庫，不需要單獨的伺服器程序，並且允許使用 SQL 查詢語言的非標準變體存取資料庫。此版本增強了 WebInspect SQL Injection 檢查功能，以使用 SQLite 程式庫 3.16.0 或更高版本偵測 Web 應用程式中是否存在 SQL Injection 弱點。

合規報告

DISA STIG 4.11

為了在合規領域支援我們的聯盟客戶，本版本包含 WebInspect 檢查與最新版本 Defense Information Systems Agency Application Security and Development STIG 4.11 版之間的關聯性。

原則更新

DISA STIG 4.11

在 WebInspect SecureBase 內現有的支援原則清單中，已新增為納入與 DISA STIG 4.11 相關的檢查而自訂的原則。

Web API

在 WebInspect SecureBase 內現有的支援原則清單中，已新增為納入與 API 安全性評估相關的檢查而自訂的原則。

[8] 在報表內容中，若要加入 SQLite 的相關資料庫類型資訊，將需要使用 WebInspect 20.2 或更新版本。

其他勘誤：

在此發佈中，我們持續盡可能投入一切資源，來確保我們可以降低誤報問題數，並提升客戶稽核問題的能力。客戶還會看到與下列各項相關回報問題的變更：

不安全的傳輸：弱式 SSL 加密：
識別碼 ID 為 11285 的檢查已經過修改，現在當偵測到 SSLv3 或 TLS1.0 以及 CBC 加密時，會將所有 CBC 模式加密項目回報為「弱」，而非回報 Weak SSL Protocol 內容模糊。考慮到 POODLE、GoldenPOODLE 和 ZombieDOODLE 等弱點，為了傳輸層安全性，不再建議使用 CBC 模式加密。

黑名單和白名單的命名：
為符合此領域的趨勢，Micro Focus Fortify 旗下的 Software Security Research 團隊已決定停止使用「黑名單」和「白名單」的術語，並改用較為中性且語義具體的術語。我們決定在檢查安全性內容中，分別改用更具描述性的術語，也就是「拒絕清單」和「允許清單」。

Micro Focus Fortify Premium Content

研究團隊在我們的核心安全情報產品之外建置、延伸並維護各種資源。

DISA STIG 4.11

為了呼應新的關聯性，本版本也包含支援 DISA STIG 4.11 的新 Fortify Software Security Center 報告套件，您可以從 Fortify 客戶支援入口網站的 Premium Content 下方進行下載。

Micro Focus Fortify Taxonomy：軟體安全性錯誤

Fortify Taxonomy 網站包含了新增類別支援的說明，網址為：https://vulncat.fortify.com/zh-tw。
客戶若在舊網站尋找最新的支援更新，可從 Micro Focus Fortify 支援入口網站取得該更新內容。

連絡 Fortify 技術支援
Micro Focus Fortify
https://softwaresupport.softwaregrp.com/
+1 (844) 260-7219

連絡 SSR
Alexander M. Hoole
軟體安全性研究經理
Micro Focus Fortify
hoole@microfocus.com
+1 (650) 258-5916

2020 年 9 月 25 日