IBM Security AppScan Source for Analysis
版本 9.0.3.7

使用手册

IBM

IBM Security AppScan Source for Analysis
版本 9.0.3.7

使用手册

IBM

本程式包含：Jacorb 2.3.0（Copyright 1997-2006 JacORB 專案）及 XOM1.0d22 (Copyright 2003 Elliotte Rusty Harold)，每一個都依 「GNU 程式庫通用公用授權 (Gnu Library General Public License, LGPL)」來提供，您可以在本程式隨附的「注意事項」檔案中找到此 LGPL 的副本。

目錄

第 1 章 AppScan Source for Analysis 簡介

本節說明 AppScan® Source for Analysis 如何適用於 AppScan Source 總體解決方案，並提供您瞭解軟體安全工作流程的基礎。

IBM Security AppScan Source 簡介

IBM® Security AppScan Source 可為您組織內每一個在軟體安全上擔任相關職務的使用者，帶來最大價值。不論是安全分析師、品保專業人員、開發人員或高階主管，AppScan Source 產品都能直接在您桌面上提供您需要的功能、彈性和權限。

此產品組包含：

- **AppScan Source for Analysis**：這個工作台可用來配置應用程式和專案、掃描程式碼、進行分析、分類，以及處理必須優先解決的漏洞。
- **AppScan Source for Automation**：可讓您在軟體開發生命週期內，將 AppScan Source 工作流程的關鍵作業自動化並整合安全與建置環境。
- **AppScan Source for Development**：開發人員外掛程式將許多 AppScan Source for Analysis 特性整合到 Microsoft Visual Studio、Eclipse 工作台和 Rational® Application Developer for WebSphere® 軟體 (RAD) 中。讓軟體開發人員能夠在開發程序期間發現漏洞並採取行動。Eclipse 外掛程式可讓您掃描原始碼來尋找安全漏洞 - 您可以使用 Eclipse 外掛程式來掃描 IBM MobileFirst Platform 專案。

為了強化 AppScan Source 在您組織內的價值，產品包含下列元件：

- **AppScan Source 安全知識庫**：每一漏洞的環境定義相關知識，提供關於主要原因、風險嚴重性和可行的補救建議的精闢說明。
- **AppScan Enterprise Server**：大部分 AppScan Source 產品和元件必須與 AppScan Enterprise Server 通訊。如果沒有的話，您可以在本端模式下使用 AppScan Source for Development - 但將無法使用某些特性，例如自訂規則、共用的掃描配置及共用的過濾器。

  伺服器提供集中式使用者管理功能，以及提供透過 AppScan Source 資料庫來共用評量的機制。伺服器包含選用的 Enterprise Console 元件。如果您的管理者安裝這個元件，您可以將評量從 AppScan Source for Analysis、AppScan Source for Automation 及 AppScan Source 指令行介面 (CLI) 發佈到這個元件。 Enterprise Console 提供各種可處理評量的工具，例如：報告特性、問題管理、趨勢分析和儀表板。

  重要：針對某些版本的 AppScan Source和 AppScan Enterprise，兩個產品的版本和版次層次必須相符，才能從 AppScan Source 連接到 AppScan Enterprise Server。請參閱http://www.ibm.com/support/docview.wss?uid=swg21975211以瞭解哪些版本的 AppScan Source 和 AppScan Enterprise 相容。

  註：

  – 在 macOS 中，不支援 AppScan Enterprise Server。
  – 如果您具備基本伺服器授權，AppScan 產品最多只能透過 10 條並行連線來存取伺服器。如果具備頂級伺服器授權，連線數量就不受限。

重要：掃描時，AppScan Enterprise Server 和 AppScan Source 用戶端（AppScan Source for Development 除外）都需要直接連線到 AppScan Source 資料庫（solidDB® 或 Oracle）。

此「軟體供應項目」不使用 Cookie 或其他技術來收集個人識別資訊。

翻譯的國家語言

AppScan Source 使用者介面提供了以下語言：

- 英文
- 巴西葡萄牙文
- 簡體中文
- 繁體中文
- 德文
- 西班牙文
- 法文
- 義大利文
- 日文
- 韓文
- 俄文

遵守美國政府法規

遵守美國政府安全和資訊技術法規，有助於消除銷售障礙和路障。另外，它還提供了 IBM 致力於產生行業內最安全產品之世界性前景的一個證明點。本主題列出 AppScan Source 支援的標準和準則。

- 『網際網路通訊協定第 6 版 (IPv6)』
- 『美國聯邦資訊處理標準 (FIPS)』
- 第 3 頁的『國家標準與技術機構 (NIST) 特殊出版品 (SP) 800-131a』
- 第 3 頁的『已配置成採用「美國政府配置基準 (United States Government Configuration Baseline, USGCB)」的 Windows 7 機器』

網際網路通訊協定第 6 版 (IPv6)

AppScan Source 已針對 IPv6 啟用，但下列情況例外：

- 不支援輸入 IPv6 數值位址，必須改為輸入主機名稱。支援輸入 IPv4 數值位址。
- 連接至 Rational Team Concert™ 時不支援 IPv6。

美國聯邦資訊處理標準 (FIPS)

在 AppScan Source 支援的 Windows 和 Linux 平台上，AppScan Source 利用 FIPS 140-2 驗證過的加密模組和經過檢驗的演算法，支援 FIPS 出版品 140-2。在 AppScan Source 支援的 macOS 平台上，於 FIPS 140-2 模式下操作需要手動步驟。

如果要瞭解 AppScan Source FIPS 標準的相關背景資訊 - 以及瞭解如何啟用及停用 AppScan Source FIPS 140-2 模式，請參閱這些 TechNotes：

- 在 macOS 的 FIPS 140-2 模式下操作 AppScan Source 8.7 版或更新版本

- 如何在 AppScan Source 中啟用/停用/驗證 FIPS 140-2 模式（Linux 和 Windows）
- AppScan Source 8.7 版或更新版本 FIPS 140-2 支援的相關背景資訊

國家標準與技術機構 (NIST) 特殊出版品 (SP) 800-131a

NIST SP 800-131A 準則提供加密金鑰管理指引。這些準則包括：

- 金鑰管理程序。
- 如何使用加密演算法。
- 要使用的演算法及其強度下限。
- 安全通訊的金鑰長度。

政府機構和金融機構使用 NIST SP 800-131A 準則，以確保產品符合指定的安全需求。

唯有當 AppScan Source 是在 FIPS 140-2 模式下操作時，才支援 NIST SP 800-131A。如果要瞭解如何啟用及停用 AppScan Source FIPS 140-2 模式，請參閱第 2 頁的『美國聯邦資訊處理標準 (FIPS)』。

重要：如果您要連接的 AppScan Enterprise Server 是為了符合 NIST 800-131a 標準而啟用，您必須設定 AppScan Source 強制執行「傳輸層安全 (TLS) 1.2 版」。如果未強制執行「傳輸層安全 (TLS) 1.2 版」，則伺服器連線會失敗。

- 如果您**未**安裝 AppScan Source 資料庫（例如，您只安裝用戶端元件），您可以修改 `<data_dir>\config\ounce.ozsettings`（其中 `<data_dir>` 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述），來強制執行「傳輸層安全 (TLS) 1.2 版」。在這個檔案中，請尋找這項設定：

  ```
  <Setting
        name="tls_protocol_version"
        read_only="false"
        default_value="0"
        value="0"
        description="Minor Version of the TLS Connection Protocol"
        type="text"
        display_name="TLS Protocol Version"
        display_name_id=""
        available_values="0:1:2"
        hidden="false"
        force_upgrade="false"
   />
  ```

  在這項設定中，將 `value="0"` 變更為 `value="2"`，然後儲存檔案。
- 如果您要安裝 AppScan Source 資料庫，則在同時安裝了 AppScan Source 和 Enterprise Server 之後，在 IBM Security AppScan Enterprise Server 資料庫配置工具中強制執行「傳輸層安全 (TLS) 1.2 版」。

已配置成採用「美國政府配置基準 (United States Government Configuration Baseline, USGCB)」的 Windows 7 機器

AppScan Source 支援在以 USGCB 規格所配置的 Windows 7 機器上掃描應用程式。

註：在以 USGCB 規格所配置的機器上，AppScan Source 不支援整合問題報告追蹤系統與 HP Quality Center 或 Rational ClearQuest®。

AppScan Source 的新增功能

探索這些已新增至 AppScan Source 的新增特性，並指出在這個版本中已淘汰的任何這些與功能。

AppScan Source 9.0.3.7 版新增功能

加強和新的掃描支援

- 支援 Red Hat Enterprise Linux (RHEL) 7.3 版作業系統。
- 支援將 AppScan Source for Development Visual Studio 外掛程式 套用至 Visual Studio 2015。

AppScan Source 9.0.3.7 版中不再支援的功能及特性

從 AppScan Source 9.0.3.7 版開始：

- OS X 10.10 版不再是支援的作業系統。
- 不再支援 Xcode 6.3 版。不再支援使用這個 Xcode 版本來掃描 Objective-C 專案。
- 不再支援 Tomcat 第 5 版和第 6 版。

AppScan Source 9.0.3.6 版新增功能

加強和新的掃描支援

- 適用於 Objective-C 的 Xcode 8.1 和 8.2（僅適用於 iOS 應用程式）現在是 macOS 上支援的編譯器。對於這些版本的 Xcode 的支援追溯到 AppScan Source 9.0.3.5 版。

AppScan Source 9.0.3.6 版中不再支援的功能及特性

從 AppScan Source 9.0.3.6 版起：

- Red Hat Enterprise Linux 第 5 版不再是支援的作業系統。
- Oracle WebLogic 伺服器 第 8、9 和 10 版不再是支援的編譯器。

AppScan Source 9.0.3.5 版新增功能

- 『加強和新的掃描支援』
- 『支援漸進式掃描 Java 原始碼和位元組碼，重新掃描更有效率、更快速』

加強和新的掃描支援

- 現在，macOS 10.12 版是受支援的作業系統。對於 macOS 10.12 版的支援已追溯到 AppScan Source 9.0.3.4 版。
- 適用於 Objective-C 的 Xcode 8.0、8.1 和 8.2（僅適用於 iOS 應用程式）現在是 macOS 上支援的編譯器。

支援漸進式掃描 Java 原始碼和位元組碼，重新掃描更有效率、更快速

從 9.0.3.5 版開始，您可以在 Windows 和 Linux 上啟用 Java 漸進式掃描支援。當啟用漸進式分析時，AppScan Source 會快取處理分析資料。當您重新掃描專案或應用程式時，AppScan Source 會使用此資料來判斷程式碼變更，而且只重新分析受到變更所影響的程式碼部分。最終結果是完整的程式碼分析 - 但時間縮短了。

使用 IBM Security AppScan Source for Analysis、AppScan Source for Development Eclipse 外掛程式、IBM Security AppScan Source for Automation 或 IBM Security AppScan Source 指令行介面 (CLI) 時，支援此特性。

AppScan Source 9.0.3.4 版新增功能

- 『加強和新的掃描支援』
- 『當使用「通用存取卡 (CAC)」進行鑑別時，現在支援將評量發佈至 AppScan Enterprise Console』
- 第 6 頁的『「付款卡產業資料安全標準 (PCI DSS)」3.2 版報告支援』
- 第 6 頁的『AppScan Source for Analysis 產品說明文件』
- 第 7 頁的『能夠使用 AppScan Source for Analysis 中的掃描配置來移除任何排除過濾器的發現項目』
- 第 7 頁的『改善在 AppScan Source for Automation 和 AppScan Source 指令行介面 (CLI) 中掃描 WAR 和 EAR 檔時的程式庫處理』
- 第 7 頁的『提交 AppScan Source 評量至 Cloud 進行分析』
- 第 7 頁的『AppScan Source 9.0.3.4 版中不再支援的功能及特性』

加強和新的掃描支援

現在 PHP 7.0 版可以在 Windows 和 Linux 上的 IBM Security AppScan Source for Analysis、IBM Security AppScan Source for Automation 和 IBM Security AppScan Source 指令行介面 (CLI) 中掃描。

當使用「通用存取卡 (CAC)」進行鑑別時，現在支援將評量發佈至 AppScan Enterprise Console

如果您使用 CAC 鑑別來連接至 AppScan Enterprise Server，現在您可以從 AppScan Source 使用者介面、AppScan Source 指令行介面 (CLI) 和 AppScan Source for Automation 將評量發佈至 AppScan Enterprise Console。

「付款卡產業資料安全標準 (PCI DSS)」3.2 版報告支援

AppScan Source 現在支援「付款卡產業資料安全標準 (PCI DSS) 3.2 版報告。

AppScan Source for Analysis 產品說明文件

從 9.0.3.4 版起，當您使用AppScan Source for Analysis中的**說明** > **說明內容**功能表項目時，會開啟 IBM Knowledge Center 中的 AppScan Source 線上說明（若為 9.0.3.4 版，會開啟 IBM Security AppScan Source 9.0.3.4 版說明文件 的說明）。同樣地，當您遵循「AppScan Source for Analysis 歡迎使用」視圖的鏈結時，則會在 IBM Knowledge Center 中開啟它們。

AppScan Source for Analysis 也提供了許多視圖、喜好設定頁面和對話框的上下文相關說明。用於上下文相關說明的鍵盤快速鍵在 Windows 上是 F1，在 Linux 上是 Shift+F1，而在 macOS 上是 command+F1。在 9.0.3.4 版中，此上下文相關說明也會開啟至 IBM Knowledge Center 中的 AppScan Source。

如果您在使用產品時沒有連接網際網路，會在本端提供說明，如下所示：

- 位於 AppScan Source 安裝目錄的 `readme.html` 檔中，會提供「IBM Security AppScan Source Readme 和版本注意事項」。
- 這些 PDF 使用手冊安裝在 AppScan Source 安裝目錄的 `doc/<lang>` 或 `doc\<lang>` 目錄（其中 `<lang>` 是您 AppScan Source 安裝架構的國家語言）：
  - 僅適用於 Windows 和 Linux：*IBM Security AppScan Source for Analysis* 使用手冊 (`Security_AppScan_Source_Analysis.pdf`)
  - 僅適用於 Windows 和 Linux：*IBM Security AppScan Source Utilities* 使用手冊 (`Security_AppScan_Source_Utilities.pdf`)
  - 僅適用於 macOS：*IBM Security AppScan Source for Analysis* 使用手冊（適用於 *macOS*） (`Security_AppScan_Source_Analysis_OSX.pdf`)
  - 僅適用於 macOS：*IBM Security AppScan Source Utilities* 使用手冊（適用於 *macOS*） (`Security_AppScan_Source_Utilities_OSX.pdf`)
  - *IBM Security AppScan Source* 安裝與管理手冊 (`Security_AppScan_Source_Installation_and_Administration.pdf`)

  您必須具備 Adobe Acrobat Reader 才能閱讀這些檔案。如果您沒有 Acrobat Reader 的副本，您可以從 http://www.adobe.com/ 進行下載。
- 某些 AppScan Source for Analysis 特性的 Javadoc 位於 AppScan Source 安裝目錄中的 `doc/Javadoc` 或 `doc\Javadoc` 目錄。從 9.0.3.4 版開始，會提供下列特性的 Javadoc：
  - 應用程式伺服器匯入架構 API 類別和方法的 Javadoc 可在 `doc/Javadoc/appserverimporter` 或 `doc\Javadoc\appserverimporter` 找到。
  - Framework for Frameworks API 類別和方法的 Javadoc 可在 `doc/Javadoc/frameworks` 或 `doc\Javadoc\frameworks` 找到。

  在這些資料夾中，開啟 `index.html` 檔。

能夠使用 AppScan Source for Analysis 中的掃描配置來移除任何排除過濾器的發現項目

排除過濾器含有一些規則，用來將漏洞類型、應用程式設計介面 (API)、檔案、目錄、專案或追蹤規則從發現項目移除。如果您在掃描配置中包含多個排除過濾器，彼此可能發生衝突，而影響發現項目。例如，假設有下列兩個過濾器：

- 過濾器 1 會移除漏洞類型 Validation.EncodingRequired 的所有發現項目。這不會反轉，因此會將這些發現項目從評量排除。
- 過濾器 2 會移除漏洞類型 Validation.Required 的所有發現項目。這不會反轉，因此會將這些發現項目從評量排除。

如果使用掃描配置套用了這兩個過濾器，依預設，它們會支配對方。過濾器 1 會排除 Validation.EncodingRequired 發現項目 - 但它會包含 Validation.Required 發現項目。過濾器 2 會排除 Validation.Required 發現項目 - 但它會包含 Validation.EncodingRequired 發現項目。最終結果是 Validation.EncodingRequired 和 Validation.Required 發現項目都包含在內。

從 9.0.3.4 版起，在您建立掃描配置時，您可以選取**符合任何未反轉的排除過濾器**，來移除指定之任何排除過濾器的發現項目。這個勾選框位於「掃描配置」視圖之**一般**標籤的**過濾器資訊**區段中。以上述範例來說，如果選取這個勾選框，就會將所有 Validation.EncodingRequired 和 Validation.Required 發現項目從評量排除。

改善在 AppScan Source for Automation 和 AppScan Source 指令行介面 (CLI) 中掃描 *WAR* 和 *EAR* 檔時的程式庫處理

在掃描 WAR 檔時，現在提供下列設定：

- -include_all_lib_jars：使用此設定可在掃描期間併入 WAR 檔中的所有程式庫。
- -include_lib_jars：使用此設定可指定要在掃描期間加入的 WAR 檔中的程式庫。

在匯入 EAR 檔時，會自動建立專案來儲存共用程式庫。如果沒有共用程式庫，會建立專案，但專案是空的。現在提供 -no_ear_project 設定，如果使用，則不會對 EAR 檔建立專案。

提交 AppScan Source 評量至 Cloud 進行分析

如果您在 IBM Cloud Marketplace 訂閱 IBM Application Security on Cloud，或訂閱 Application Security on Cloud for Bluemix，您可以在該處提交 AppScan Source 評量以進行分析。支援來自 AppScan Source 9.0 版或更新版本的評量 - 您可以提交的掃描數目視您的 Application Security on Cloud 訂閱而定。請參閱 http://www.ibm.com/support/knowledgecenter/SSYJJF_1.0.0/ApplicationSecurityonCloud/src_managing_assessments_cloud.html，以取得詳細資訊。

AppScan Source 9.0.3.4 版中不再支援的功能及特性

從 AppScan Source 9.0.3.4 版開始：

- OS X 10.9 版不再是支援的作業系統。
- 不再支援 Xcode 5.x 版、6.0 版和 6.2 版。不再支援使用這些 Xcode 版本來掃描 Objective-C 專案。
- 掃描 PHP 5.3 和 5.4 的支援已淘汰。

AppScan Source 9.0.3.3 版新增功能

新的平台和整合解決方案支援

從 AppScan Source 9.0.3.3 版開始：

- 現在，Microsoft Windows 10 是受支援的作業系統。這包括 Windows 10 Education、Enterprise 及 Pro 版本。

  註：

  – 在 Windows 10 中，AppScan Source 安裝程式（AppScanSrc_Installer.exe 檔）必須在 Windows 7 相容模式下執行。在 Windows 10 中，您還必須在解除安裝 AppScan Source 之前，先將 AppScan_Uninstaller.exe 檔設定為在 Windows 7 相容模式下執行。這個檔案位於 <install_dir>\Uninstall_AppScan\AppScan_Uninstaller.exe（其中 <install_dir> 是 AppScan Source 安裝的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）。如需相關資訊，請參閱 http://www.ibm.com/support/docview.wss?uid=swg21696098。

  – Windows 10 支援受 http://www.ibm.com/support/docview.wss?uid=swg21689814 中所述的問題所影響。

- 如果要連接到 AppScan Enterprise Server 9.0.3.1 版或更高版本，IBM Security AppScan Source 資料庫 可以安裝到 Oracle 12c 資料庫。

  重要：如果您有利用 Oracle 11g 資料庫的現有 AppScan Source 安裝，並想要升級至 Oracle 12c，則必須在升級 Oracle 資料庫之前升級 AppScan Source。

- AppScan Source 的安裝架構現在包含 **Tomcat 8**。
- Visual Studio 2015 解決方案和專案檔現在可以在 AppScan Source for Analysis、AppScan Source for Automation 和 AppScan Source 指令行介面 中掃描。如果在 Visual Studio 2015 中建立過 .sln 或 .vcproj 檔，當您在 Windows 上使用 AppScan Source for Analysis、AppScan Source for Automation 或 AppScan Source 指令行介面 時，可以匯入及掃描這些檔案。

  重要：

  – 不支援將 AppScan Source for Development Visual Studio 外掛程式 套用至 Visual Studio 2015。

  – 支援受管理的 C++ 專案。如果是使用 Visual Studio 2013 或更舊版本的 Platform Toolset（Platform Toolset V120 或更舊版本）所建置的未受管理 C++ 專案也可支援。

- 適用於 Objective-C 的 Xcode 7.3（限 iOS 應用程式）現在是 macOS 上支援的編譯器（Xcode 7.3 的支援可溯及 AppScan Source 9.0.3.2 版）。

加強和新的掃描支援

- PHP 5.5 和 5.6 版現在可以在 Windows 與 Linux 上，於 IBM Security AppScan Source for Analysis、IBM Security AppScan Source for Automation 和 IBM Security AppScan Source 指令行介面 (CLI) 中掃描。
- 現在，使用 AppScan Source 掃描 Java™ 時，支援 `@ValidatorMethod`、`@CallbackMethod` 和 `@SuppressSecurityTrace` 方法層次註釋。

Windows 的新檔案名稱

在 Windows 中，安裝檔案名稱已從 `setup.exe` 變更為 `AppScanSrc_Installer.exe`。

Windows 上的共用存取卡 (CAC) 支援

共用存取卡 (http://www.cac.mil) 是針對美國境內的現役一致性服務人員、特選後備軍人、DoD 軍屬員工及合格契約人員的標準識別。它用來啟用大廈和管制空間的實體存取權，並提供 DoD 電腦網路與系統的存取權。 CAC 可用來存取附有各種智慧卡讀取器的電腦和網路。當插入至讀取器時，裝置會詢問使用者 PIN。

如果您正在 Windows 中執行 AppScan Source，並連接至已啟用進行「共用存取卡 (CAC)」鑑別的 AppScan Enterprise Server 9.0.3.1 版 iFix-001 或更高版本，則 AppScan Source 現在支援 CAC 鑑別。

DISA 應用程式安全及開發 STIG 3.10 版報告支援

AppScan Source 現在支援「國防資訊系統局 (DISA) 應用程式安全及開發安全技術實作手冊 (STIG) 3.10 版」報告。

AppScan Source 9.0.3.2 版新增功能

AppScan Source 和 AppScan Enterprise 版本相容性

當連接至 AppScan Enterprise Server，或發佈到 AppScan Enterprise Console 時，某些版本的 AppScan Source 不再需要 AppScan Source 和 AppScan Enterprise 版本與版次層次相符。請參閱http://www.ibm.com/support/docview.wss?uid=swg21975211，以瞭解哪些版本的 AppScan Source 和 AppScan Enterprise 相容。

這項變更可溯及某些舊版的 AppScan Source，如http://www.ibm.com/support/docview.wss?uid=swg21975211中所述。

AppScan Source 9.0.3.1 版新增功能

- 『新的整合解決方案支援』
- 第 10 頁的『在 AppScan Source for Automation 和 AppScan Source 指令行介面 (CLI) 中掃描 WAR 和 EAR 檔』

新的整合解決方案支援

從 AppScan Source 9.0.3.1 版開始：

- 現在支援 Tomcat 8 來編譯 Java 和 JSP。

  註：作業系統支援會視個別編譯器支援的作業系統而定。

- 適用於 Objective-C 的 Xcode 7.0、7.1 和 7.2（限 iOS 應用程式）現在是 macOS 上支援的編譯器。

在 AppScan Source for Automation 和 AppScan Source 指令行介面 (CLI) 中掃描 *WAR* 和 *EAR* 檔

CLI 中的 `openapplication (oa)` 指令現在可以用來開啟 `WAR` 和 `EAR` 檔。此外，還可以利用 `ScanApplication` 指令在 AppScan Source for Automation 中掃描這些檔案。

AppScan Source 9.0.3 版新增功能

新的平台和整合解決方案支援

從 AppScan Source 9.0.3 版開始，即支援這些作業系統：

- Red Hat Enterprise Linux 第 6 版更新項目 6 與 7
- OS X 10.11 版。OS X 10.11 版的支援可溯及 AppScan Source 9.0.2 版，但有 http://www.ibm.com/support/docview.wss?uid=swg21968948 中所說明的限制（此限制僅影響 AppScan Source 9.0.2 版）。

此外：

- 適用於 Objective-C 的 Xcode 6.3 和 6.4（限 iOS 應用程式）現在是 OS X 上支援的編譯器（Xcode 6.3 和 6.4 的支援可溯及 AppScan Source 9.0.2 版）。請注意，Xcode 6.3 和 6.4 支援存在一些限制。請參閱http://www.ibm.com/support/docview.wss?uid=swg21962208以取得詳細資訊。這些限制不適用於 AppScan Source 9.0.3.1 版和更新版本。
- AppScan Source for Development Eclipse 外掛程式 現在與 IBM MobileFirst Platform Foundation 7.1 版整合。現在您可以掃描 IBM MobileFirst Platform 7.1 版的專案、應用程式、環境以及 AppScan Source 產品中的 HTML 檔案。
- 可掃描 Rational Application Developer for WebSphere 軟體 (RAD) 9.1.1 版的專案檔和工作區 - 而且 AppScan Source for Development（Eclipse 外掛程式）可套用至 RAD 9.1.1 版。
- 可以掃描 Eclipse 4.5 版專案檔及工作區（限 Java 和 IBM MobileFirst Platform）- 且 AppScan Source for Development（Eclipse 外掛程式） 可以套用至 Eclipse 4.5 版。
- 現在支援 IBM WebSphere Application Server 8.5.5 版來編譯 Java 和 JSP。

  註：作業系統支援會視個別編譯器支援的作業系統而定。

掃描配置加強功能

「掃描配置」視圖已重新設計，現在提供這些主要特性：

- 指定過濾器的能力。
- 設定在掃描期間要執行的分析類型。這包括污染流分析和基於型樣的分析。

AppScan Source 現在包括以下內建掃描配置：Web 預覽掃描、Web 快速掃描、Web 平衡掃描，以及 Web 深度掃描

新的規則屬性可讓您更正確地識別高嚴重性的明確安全發現項目

這個版本的 AppScan Source 引進了 `Attribute.Likelihood.High` 與 `Attribute.Likelihood.Low` 屬性。這些屬性已新增內建規則，建立自訂規則時也可使用它們。

在 AppScan Source 中，可能性代表可惡意探索安全發現項目的機率或機會。AppScan Source 使用 https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology#Step_2:_Factors_for_Estimating_Likelihood 所呈現的可能性定義，並根據追蹤內容判斷可能性來精簡它。假設有一組追蹤內容 - 例如，來源 API 名稱、來源 API 類型、來源技術或來源機制 - AppScan Source 判斷未來可利用特定漏洞惡意探索追蹤的可能性。

可能性關聯於追蹤的來源元素。來源是程式的輸入，例如檔案、Servlet 要求、主控台輸入或 Socket。對大部分輸入來源而言，傳回的資料在內容和長度方面並無限制。當輸入不受檢查時，即被認為是污染的來源。

可能性範例包括：

- 假設追蹤含有 HTTP 來源（例如，`Request.getQueryString`）及跨網站 Scripting 接收槽（例如，`Response.write`），則判斷具有高可能性，而增加對發現項目的信賴度。
- 假設追蹤含有系統內容來源（例如，`getProperty`）及跨網站 Scripting 接收槽（例如，`Response.write`），則判斷具有低可能性，而減少對發現項目的信賴度。

可能性是用來識別具有高優先順序的可操作發現項目，必須立即處理或修正。它關聯於一個可任意惡意探索的污染來源，並可提供您更精細的方法將這些發現項目分類。可能性是以一個關聯於污染來源的屬性，儲存在 AppScan Source 漏洞資料庫中。此特性立即可用。

我們進行了廣泛的研究，以判斷來源的可能性因素。您可以使用「自訂規則精靈」，把可能性資訊新增至您加入規則庫的新污染來源。這會改進從掃描產生的發現項目的分類，進而提高整體分類工作流程的效率。

在「自訂規則精靈」中，您可以對**可能性**內容設定兩個值（**高**和**低**）。**高**值表示該來源很可能遭到污染。換句話說，污染進入系統的屏障極低，使得攻擊者很容易透過手動或自動化方式提交惡意資料。**低**值表示透過此來源輸入惡意資料的屏障極高。這可能表示為了將污染引進來源中，攻擊者必須對系統非常瞭解，而且要有許可權可對受害者網路進行操作。

註：由於這些規則屬性，如果您已在舊版的 AppScan Source 中產生評量，您可能會發現在以 9.0.3 版掃描時，相同原始碼的發現項目分類已變更。如需相關資訊，以及如果要瞭解如何停用這些規則屬性，請參閱關於這些變更的移轉考量。

遺失的接收槽自動解析可產生更好的掃描結果

AppScan Source 現在嘗試透過自動推斷遺失的接收槽方法（如 getter、setter）以及傳回布林值的方法的標記，來解析在追蹤中遺失的接收槽。如此可對您的程式碼提供更完整的分析，並改善遺失的接收槽解析結果。

註：由於這項特性，如果您已在舊版的 AppScan Source 中產生評量，您可能會注意到未解析的遺失接收槽的發現項目結果有些變更。如需相關資訊，以及如果要瞭解如何停用自動產生標記，請參閱關於這些變更的移轉考量。

加強和新的掃描支援

- PHP 5.4 版現在可以在 Windows 與 Linux 上，於 IBM Security AppScan Source for Analysis、IBM Security AppScan Source for Automation 和 IBM Security AppScan Source 指令行介面 (CLI) 中掃描。
- AppScan Source 現在包括 Spring MVC 4 架構的內建支援。
- **Java 掃描最佳化：**
  - 在掃描「JavaServer 頁面」時，現在可以選擇掃描經過前置編譯的類別檔，而非在掃描期間才編譯。如果要在 AppScan Source for Development Eclipse 外掛程式 中掃描經過前置編譯的類別檔，請配置專案進行安全掃描（選取**安全分析** > **配置掃描** > **配置安全專案**），然後選取**經過前置編譯的類別**勾選框。如果要在 IBM Security AppScan Source for Analysis 中掃描經過前置編譯的類別檔，請在下列其中一個位置選取**經過前置編譯的類別**勾選框：
    - 專案內容中的「專案相依關係」標籤。
    - 建立新專案或應用程式時的「Java 專案相依關係」頁面。
  - 掃描 Java 時，AppScan Source 現在會掃描含有遺失的相依關係或編譯錯誤的 Java 檔和 Java 位元組碼。如果有遺失的相依關係或編譯錯誤，相關資訊會寫入日誌檔中。接著您可以利用這項資訊將相依關係新增到專案內容中，重新掃描，使得掃描結果具有完整涵蓋面。
- 自 AppScan Source 9.0.3 版起，在匯入及掃描 Xcode 專案時，會更正確地判定標頭位置和配置選項。這項變更引用了 `xcodebuild -dry-run` 來取得每一個檔案的建置配置，在開始掃描時，AppScan Source 會先判斷檔案配置再繼續進行，因此可能會稍微暫停。

AppScan Source 9.0.3 版中不再支援的功能及特性

從 AppScan Source 9.0.3 版開始：

- OS X 10.8 版不再是支援的作業系統。
- 不再支援 Xcode 4.6 版。不再支援使用這個 Xcode 版本來掃描 Objective-C 專案。
- 不再支援 Eclipse 3.6 版與 3.7 版的專案檔和工作區 - 而且 AppScan Source for Development（Eclipse 外掛程式） 不可再套用於 Eclipse 3.6 版與 3.7 版。

- 不再支援 Rational Application Developer for WebSphere 軟體 (RAD) 8.0.x 版的專案檔和工作區 - 而且 IBM Security AppScan Source for Development 外掛程式（適用於 IBM Rational Application Developer for WebSphere 軟體 (RAD)）不可再套用於 RAD 8.0.x 版。
- IBM Rational Team Concert 3.0 版與 3.0.1 版不再是受支援的問題追蹤系統。
- WebSphere Application Server 6.1 版不再是受支援的應用程式伺服器。
- 掃描 PHP 4.x 版至 5.2 的支援已淘汰。

移轉至 AppScan Source 現行版本

本主題包含的移轉資訊是針對此 AppScan Source 版本已發生的變更。如果是從舊版的 AppScan Source 升級，請注意您目前所升級的 AppScan Source 版本及截至此現行版本為止的所有版本所發生的變更。

- 『從 9.0.2 版移轉』
- 第 14 頁的『從 9.0 版移轉』
- 第 14 頁的『從 8.7 版移轉』

從 9.0.2 版移轉

- 『新的規則屬性可能會導致現有掃描中的發現項目分類變更』
- 第 14 頁的『自動產生遺失的接收槽』

新的規則屬性可能會導致現有掃描中的發現項目分類變更

9.0.2 版之後引進了 `Attribute.Likelihood.High` 和 `Attribute.Likelihood.Low` 規則屬性。使用這些屬性時，AppScan Source 可以更正確地判斷發現項目是明確的及/或可疑的。因此，如果您在 AppScan Source 9.0.2 版或更舊版本中掃描原始碼，您可能會發現當相同的原始碼在 9.0.2 之後的產品版本中掃描時，某些發現項目分類會變更。這對與可高度開發的 Web 原始碼相關的發現項目，或是較不可開發的內容或環境原始碼，最為明顯。

依預設會使用這些規則屬性。您可以依下列方式將其停用：

1. 在文字編輯器（其中 `<data_dir>` 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）中開啟 `<data_dir>\config\ipva.ozsettings`。在檔案中找出 `allow_likelihood` 設定。這項設定看起來如下：

   ```
   <Setting
     name="allow_likelihood"
     value="true"
     default_value="true"
     description="Allow the processing of the Likelihood
       attributes to help determine trace confidence based
       on the source API"
     display_name="Allow Likelihood"
     type="bool"
   />
   ```

   在這項設定中，修改 `value` 屬性。如果該屬性設為 `true`，這項設定即會開啟。如果設為 `false`，AppScan Source 就不會在掃描期間使用這些規則屬性。
2. 修改好這項設定之後儲存檔案，再啟動或重新啟動 AppScan Source。

自動產生遺失的接收槽

在 9.0.2 版之後，為結尾為 getters/setters 的追蹤資料以及傳回布林值的方法建立了遺失的接收槽自動解析。這是透過自動為這些應用程式設計介面 (API) 推斷標記來完成。因此，如果您在 AppScan Source 9.0.2 版或更舊版本中掃描原始碼，您可能會注意到當相同的原始碼在 9.0.2 之後的產品版本中掃描時，含有未解析的遺失接收槽的發現項目結果有些變更。

依預設會開啟自動產生標記。如果您想要使用遺失的接收槽解析的其他方法，例如自訂規則，您可以停用它，如下所示：

1. 在文字編輯器（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）中開啟 <data_dir>\config\ipva.ozsettings。在檔案中找出 automatic_lost_sink_resolution 設定。這項設定看起來如下：

```
<name="automatic_lost_sink_resolution"
  value="true"
  default_value="true"
  description="This setting tries to perform automatic
    lost sink resolution by assuming taint propagation
    for getters, setters and APIs which return boolean
    with no arguments."
  display_name="Auto Lost Sink Resolution"
  type="bool"
/>
```

   在這項設定中，修改 value 屬性。如果該屬性設為 true，這項設定即會開啟。如果設為 false，AppScan Source 就不會自動產生這些方法的標記。
2. 修改好這項設定之後儲存檔案，再啟動或重新啟動 AppScan Source。

從 9.0 版移轉

AppScan Enterprise Server 鑑別：使用 IBM WebSphere Liberty 取代 IBM Rational Jazz™ 使用者鑑別元件時的移轉考量

- **從只有本端 Jazz 使用者的 Enterprise Server 移轉：** 在此升級實務中，先前的 Jazz 使用者在 AppScan Source 資料庫中會顯示為 AppScan Enterprise Server 使用者，但是將會無效。這些使用者可以從資料庫中移除 - 或轉換成 AppScan Source 使用者（如果您遵循 http://www.ibm.com/support/docview.wss?uid=swg21686347 中的指示來啟用該轉換）。
- **從已配置 LDAP 的 Enterprise Server 移轉：**在 Enterprise Server 升級期間，您可以選擇重新以 LDAP 來配置 Enterprise Server。如果這樣做，則現有的使用者在 AppScan Source 中仍然有效。
- **從已配置 Windows 鑑別的 Enterprise Server 移轉：** 如果 Enterprise Server 已配置 Windows 鑑別，則只要新的 Enterprise Server Liberty 配置為使用 Windows 鑑別，現有的使用者在 AppScan Source 中仍然有效。

從 8.7 版移轉

- 第 15 頁的『發現項目分類的變更』
- 第 16 頁的『將改進掃描涵蓋面的預設值變更』
- 第 16 頁的『從舊版還原 AppScan Source 預先定義的過濾器』

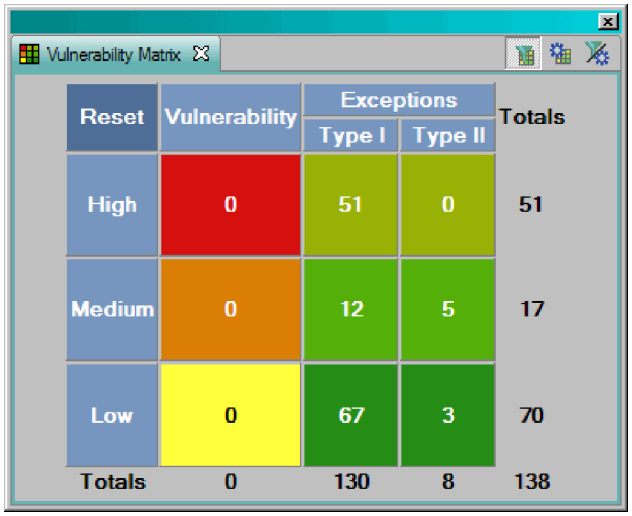
發現項目分類的變更

8.7 版之後，發現項目分類已變更。本表列出對映到新分類的舊分類：

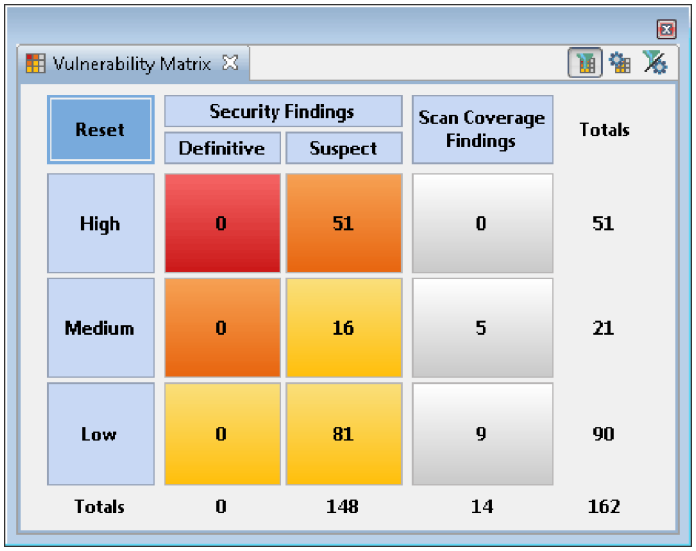
表 1. 發現項目分類變更

在 AppScan Source 8.8 版之前的發現項目分類	從 AppScan Source 8.8 版開始的分類
漏洞	最後安全發現項目
類型 I 異常狀況	可疑安全發現項目
類型 II 異常狀況	掃描涵蓋面發現項目

可在「漏洞矩陣」視圖中看到這些變更的範例。



從 8.8 版開始，此視圖的外觀如下：



將改進掃描涵蓋面的預設值變更

從 AppScan Source 8.8 版開始：

- `scan.ozsettings` 中的 `show_informational_findings` 的預設值已從 `true` 變更為 `false`。
- `ipva.ozsettings` 中的 `wafl_globals_tracking` 的預設值已從 `false` 變更為 `true`。這項設定可讓 AppScan Source 在架構型應用程式的不同元件之間尋找資料流（例如，從控制器到視圖的資料流）。

`show_informational_findings` 的變更，依預設將導致評量不包括其嚴重性層次為**參考資訊**的發現項目。

註：如果您在 8.8 版之前建立的掃描配置還有尚未明確設定這些設定的值，則掃描配置現在會使用其新的預設值。

從舊版還原 AppScan Source 預先定義的過濾器

在 AppScan Source 8.8 版，預先定義的過濾器已改良，可提供更好的掃描結果。如果您需要繼續使用 AppScan Source 舊版本中預先定義的過濾器（保存的過濾器是列在 第 133 頁的『AppScan Source 預先定義的過濾器（8.7.x 版及更舊版本）』中），請遵循第 134 頁的『還原保存的預先定義過濾器』中的指示。

AppScan Source for Analysis 概觀

AppScan Source for Analysis 是一個程式碼分析工具，提供關於重要系統中原始碼漏洞的特定資訊。 AppScan Source for Analysis 可讓您集中管理軟體風險，這些軟體可能位於多重應用程式，甚至是您的整個組合內。您可以掃描原始碼、進行分類，然後消除漏洞，以免它們對組織造成不利的情況。

AppScan Source for Analysis 提供若干工具，供審核及品質保證團隊掃描原始碼、分類結果，以及將缺失提交給問題追蹤系統。

在從 AppScan Source 安全知識庫取得環境定義相關知識之後，分析師、審核員、管理員和開發人員即可進行下列動作：

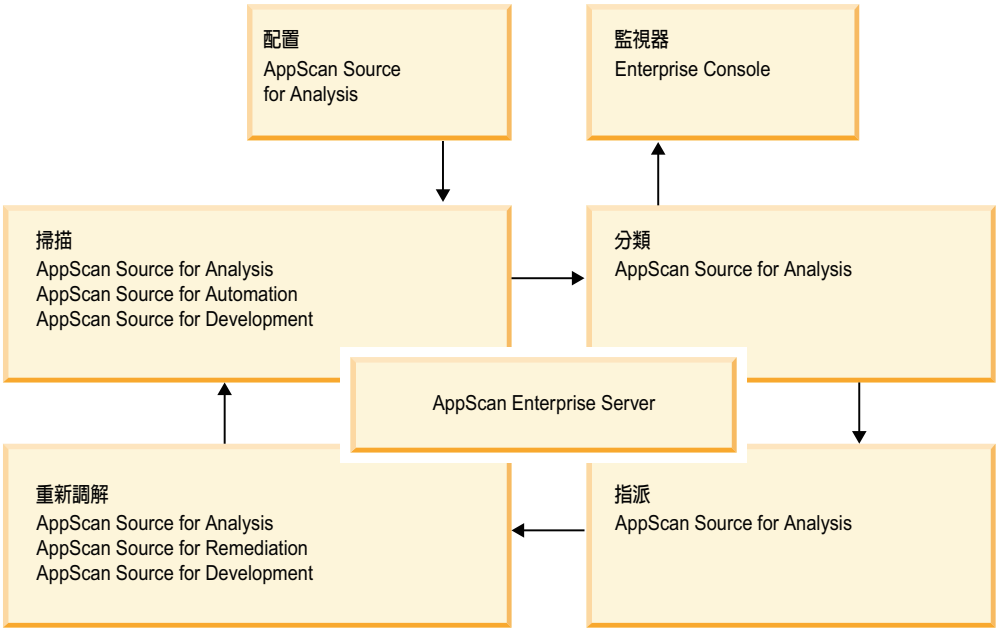
- 隨需掃描所選的原始碼來尋找重要的漏洞
- 取得精確的補救建議，直接從分析中，呼叫偏好的開發環境和程式碼編輯器
- 透過精確的互動式呼叫圖，從輸入到輸出來追蹤污染的資料
- 施行編碼原則，透過 AppScan Source 追蹤來驗證已核准的輸入驗證和編碼常式
- 在軟體開發期間，學習及實作安全的程式設計最佳實務

工作流程

在安裝、部署及使用者管理之後，AppScan Source 工作流程由下列基本步驟組成。

1. **設定安全需求**：管理員或安全專家會定義漏洞，而且也會定義判定嚴重性的方式。
2. **配置應用程式**：組織應用程式和專案。
3. **掃描**：對目標應用程式執行分析，以識別漏洞。

4. **分類和分析結果**：工作人員帶著安全意識來研究結果，設定補救工作流程的優先順序，區分實際的漏洞和潛在漏洞，以便立即開始重要問題的分類。找出您需要先解決的問題。
5. **自訂 知識庫**：自訂 AppScan Source 安全知識庫來處理內部原則。
6. **發佈掃描結果**：新增掃描結果到 AppScan Source 資料庫中，或將它們發佈到 AppScan Enterprise Console。
7. **指派補救作業**：指派問題報告給開發團隊，來解決漏洞。
8. **解決問題**：改寫程式碼、移除缺失或新增安全函數，來刪除漏洞。
9. **驗證修正**：重新掃描程式碼，確定已消除漏洞。



重要概念

在開始使用或管理 AppScan Source 之前，您應該先熟悉基本的 AppScan Source 概念。本節說明基本 AppScan Source 術語和概念。後續的章節會重複這些定義，來協助您瞭解它們在 AppScan Source for Analysis 中的環境定義。

AppScan Source for Analysis 會掃描原始碼的漏洞，並列出發現項目。發現項目是掃描期間所識別的漏洞，掃描結果是一項評量。組合是儲存在應用程式中的個別發現項目的具名集合。

應用程式、應用程式屬性和專案是在 AppScan Source for Analysis 中建立及組織：

- **應用程式**：應用程式包含一或多個專案及其相關屬性。
- **專案**：專案由一組檔案（包含原始碼）及其相關資訊（例如配置資料）所組成。專案一律是應用程式的一部分。
- **屬性**：屬性是應用程式的一種性質，協助將掃描結果組織成有意義的分組，例如按照部門或專案領導人分組。您在 AppScan Source for Analysis 中定義屬性。

AppScan Source for Analysis 的主要活動是掃描原始碼以及分析漏洞。評量是用來分析原始碼中的漏洞，其中包含：

- **嚴重性**：高、中或低，指出風險層次
- **漏洞類型**：漏洞種類，如「SQL 注入」或「緩衝區溢位」
- **檔案**：含有發現項目的程式檔
- **API/原始碼**：有漏洞的呼叫，顯示 API 及傳給它的引數
- **方法**：發出有漏洞的呼叫的函數或方法
- **位置**：程式檔中有漏洞的 API 所在的行號和直欄號碼
- **分類**：安全發現項目或掃描涵蓋面發現項目。如需相關資訊，請參閱『分類』。

分類

發現項目是由 AppScan Source 分類，指出它們是安全或掃描涵蓋面的發現項目。安全發現項目代表實際或可能的安全漏洞 - 而掃描涵蓋面的發現項目則代表可改進配置以提供更佳掃描涵蓋面的領域。

每一個發現項目屬於下列其中一個分類：

- **最後**安全發現項目：包含最後設計、實作或原則違規的發現項目，它使攻擊者有機會導致應用程式以非預期的方式操作。

  此攻擊可能導致未獲授權的存取、竊取或毀損資料、系統或資源。每一個最後安全發現項目都完整表達，並已知有漏洞狀況的特定基礎型樣，且加以說明。
- **可疑**安全發現項目：指出可疑及可能有漏洞狀況的發現項目，需要其他資訊或調查。在誤用時會產生漏洞的程式碼元素或結構。

  可疑發現項目與最後發現項目不同，因為有某種不明狀況阻止漏洞的最後判定。此種不確定性的範例包含使用的動態元素或程式庫函數，其原始碼無法使用。因此需要進一步研究，以確認或否決可疑發現項目為最後發現項目。
- **掃描涵蓋面**發現項目：代表可改進配置以提供更佳掃描涵蓋面的領域之發現項目（例如，遺失的接收槽發現項目）。

註：在某些情況下，**無**這項分類可用來表示既非安全發現項目也非掃描涵蓋面發現項目的分類。

從 AppScan Source 產品登入 AppScan Enterprise Server

大部分 AppScan Source 產品和元件都需要有 AppScan Enterprise Server 的連線。伺服器提供集中式使用者管理功能，以及提供透過 AppScan Source 資料庫來共用評量的機制。

當您啟動 AppScan Source for Analysis 時，會提示您向 AppScan Enterprise Server 鑑別。如果您在伺服器模式下執行 AppScan Source for Development，當您第一次起始的動作需要存取伺服器時，將會提示您向 AppScan Enterprise Server 鑑別，例如啟動掃描或檢視掃描配置。

- 第 19 頁的『從 AppScan Source for Analysis 和 AppScan Source for Development 使用 AppScan Enterprise Server 使用者 ID 和密碼登入』
- 第 19 頁的『使用「通用存取卡 (CAC)」鑑別從 AppScan Source for Analysis 和 AppScan Source for Development 登入』

- 第 20 頁的『從 AppScan Source for Automation 和 AppScan Source 指令行介面 (CLI) 登入』
- 第 20 頁的『AppScan Enterprise Server SSL 憑證』
- 第 20 頁的『解決 AppScan Enterprise Server 憑證錯誤』

從 AppScan Source for Analysis 和 AppScan Source for Development 使用 AppScan Enterprise Server 使用者 ID 和密碼登入

在 AppScan Source for Analysis 中，登入時會提示您提供：

- **使用者 ID**：指定您的使用者 ID（視您帳戶的設定方式而定，這是同時存在於 AppScan Enterprise Server 和 AppScan Source 資料庫的使用者 ID，或是僅存在於 AppScan Source 資料庫的使用者 ID）。
  - 如果 AppScan Enterprise Server 配置為使用 Windows 鑑別，請輸入您用來連接到 Enterprise Console 的網域和使用者名稱（以 \ 區隔網域和使用者名稱 - 例如，`my_domain\my_username`）。
  - 如果 AppScan Enterprise Server 已配置 LDAP，請輸入您用來連接到 Enterprise Console 的使用者名稱。
- **密碼**：指定您使用者 ID 的密碼。
- **AppScan Enterprise Server**：指定 AppScan Enterprise Server 實例的 URL。這個 URL 的格式是 `http(s)://<hostname>:<port>/ase`，其中 `<hostname>` 是 AppScan Enterprise Server 安裝所在的機器名稱， `<port>` 是伺服器執行所在的埠。舉例來說，這個 URL 可以是 `https://myhost.mydomain.ibm.com:9443/ase`。

在 AppScan Source for Development 中，登入時會提示您提供：

- **伺服器 URL**：指定 AppScan Enterprise Server 實例的 URL。這個 URL 的格式是 `http(s)://<hostname>:<port>/ase`，其中 `<hostname>` 是 AppScan Enterprise Server 安裝所在的機器名稱， `<port>` 是伺服器執行所在的埠。舉例來說，這個 URL 可以是 `https://myhost.mydomain.ibm.com:9443/ase`。
- **使用者 ID**：指定您的使用者 ID（視您帳戶的設定方式而定，這是同時存在於 AppScan Enterprise Server 和 AppScan Source 資料庫的使用者 ID，或是僅存在於 AppScan Source 資料庫的使用者 ID）。
  - 如果 AppScan Enterprise Server 配置為使用 Windows 鑑別，請輸入您用來連接到 Enterprise Console 的網域和使用者名稱（以 \ 區隔網域和使用者名稱 - 例如，`my_domain\my_username`）。
  - 如果 AppScan Enterprise Server 已配置 LDAP，請輸入您用來連接到 Enterprise Console 的使用者名稱。
- **密碼**：指定您使用者 ID 的密碼。

使用「通用存取卡 (CAC)」鑑別從 AppScan Source for Analysis 和 AppScan Source for Development 登入

在 Windows 中，您可以使用 CAC 鑑別 (http://www.cac.mil) 連接至 AppScan Enterprise Server。在執行此動作之前，您必須設定 AppScan Enterprise Server 和 AppScan Source 進行「共用存取卡 (CAC)」鑑別。如果已設定 Enterprise Server 進行 CAC 鑑別，則無法使用 Enterprise Server 使用者 ID 和密碼進行登入。

在 AppScan Source for Analysis 中，登入時會提示您提供：

- **使用者**：從清單中選取您的 CAC 通用名稱。
- **AppScan Enterprise Server**: 指定 AppScan Enterprise Server 實例的 URL。這個 URL 的格式是 `http(s)://<hostname>:<port>/ase`，其中 `<hostname>` 是 AppScan Enterprise Server 安裝所在的機器名稱， `<port>` 是伺服器執行所在的埠。舉例來說，這個 URL 可以是 `https://myhost.mydomain.ibm.com:9443/ase`。

在 AppScan Source for Development 中，登入時會提示您提供：

- **伺服器 URL**：指定 AppScan Enterprise Server 實例的 URL。這個 URL 的格式是 `http(s)://<hostname>:<port>/ase`，其中 `<hostname>` 是 AppScan Enterprise Server 安裝所在的機器名稱， `<port>` 是伺服器執行所在的埠。舉例來說，這個 URL 可以是 `https://myhost.mydomain.ibm.com:9443/ase`。
- **使用者**：從清單中選取您的 CAC 通用名稱。

按一下**確定**之後，「Windows 安全」對話框會提示您輸入 CAC 卡 PIN 碼

提示：

- 如果登入失敗，請確保 AppScan Enterprise Server 的設定正確且您的憑證有效。請查明您是否可以透過瀏覽器存取 AppScan Enterprise Server。如果是，則您應該能夠選取憑證及登入。
- 如果登入對話框的**使用者**欄位沒有列出可用的憑證，請確保已依照『啟用「共用存取卡 (CAC)」鑑別』中的說明修改 JRE 中的 `java.security` 檔。
- 如果「Windows 安全」對話框未提示您輸入 CAC 卡 PIN 碼，請確保 Microsoft Smart Card Resource Manager 服務在執行中。請注意，對於某些遠端桌面連線類型，此服務可能不會執行。

從 AppScan Source for Automation 和 AppScan Source 指令行介面 (CLI) 登入

執行 AppScan Source for Automation 或 AppScan Source 指令行介面 (CLI) 時也需要登入動作。如需相關資訊，請參閱《*IBM Security AppScan Source Utilities* 使用手冊》。

AppScan Enterprise Server SSL 憑證

如果要瞭解 AppScan Enterprise Server SSL 憑證，請參閱 第 23 頁的『AppScan Enterprise Server SSL 憑證』。

解決 AppScan Enterprise Server 憑證錯誤

如果您要使用不明的憑證管理中心登入 Enterprise Server，則在登入後就可能會立即收到憑證異常狀況或錯誤。AppScan Source 包含一個小型公用程式，可幫助您更正此狀況。此工具為 `<install_dir>\bin\certificatetool.bat`（其中 `<install_dir>` 是 AppScan Source 安裝的位置），而在 Linux 及 macOS 上則為 `<install_dir>/bin/certificatetool.sh`。

啟用「共用存取卡 (CAC)」鑑別

本主題在幫助您將 AppScan Source 設定為容許連線到已啟用進行「共用存取卡 (CAC)」鑑別的 AppScan Enterprise Server。

開始之前

CAC 鑑別僅在 Windows 中受支援連線至 AppScan Enterprise Server 9.0.3.1 版 iFix-001 以及更高版本。

程序

1. 確保尚未設定 AppScan Enterprise Server 進行 CAC 鑑別。
2. 以 AppScan Source 管理者身分登入 AppScan Source for Analysis 或 AppScan Source 指令行介面 (CLI)。
3. 遵循《*IBM Security AppScan Source* 安裝與管理手冊》中的指示，將所有的 AppScan Enterprise Server 使用者設定為擁有所有權限。這會將 AppScan Enterprise Server 使用者的起始預設權限設定為所有管理存取權，不過，在 CAC 設定完成之後，您將能夠將預設權限變更為符合您組織的需求。
4. 結束或關閉所有的 AppScan Source 用戶端應用程式。
5. 設定 AppScan Enterprise Server 容許 CAC 鑑別
6. 遵循《*IBM Security AppScan Source* 安裝與管理手冊》中的指示，向已啟用進行「共用存取卡 (CAC)」鑑別的 AppScan Enterprise Server 登錄 AppScan Source 資料庫。
7. 開啟 <data_dir>\config\ounce.ozsettings（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）。在這個檔案中，請尋找這項設定：

   ```
   <Setting
      name="client_cert_auth"
      value="false"
      default_value="false"
      description="Uses client certificate authentication"
      display_name="Uses client certificate authentication"
      type="boolean"
      read_only="true"
      hidden="true"
    />
   ```

8. 在這項設定中，將 value="false" 變更為 value="true"，然後儲存檔案。
9. 您將會從 AppScan Source for Analysis 或 AppScan Source for Development Eclipse 外掛程式 登入 AppScan Enterprise Server：
   a. 在 Java 安裝目錄中，尋找 jre/lib/security/java.security。如果是 AppScan Source for Analysis，則 jre 資料夾位於 AppScan Source 安裝目錄。建立此檔案的備份副本。
   b. 編輯 java.security。
   c. 在提供者以及其喜好設定順序清單中，新增 com.ibm.security.capi.IBMCAC 作為第一個安全提供者。比方說，如果您正在針對 AppScan Source for Analysis 使用情形編輯 java.security，請將這項設定：

      ```
      security.provider.1=com.ibm.crypto.fips.provider.IBMJCEFIPS
      security.provider.2=com.ibm.jsse2.IBMJSSEProvider2
      security.provider.3=com.ibm.crypto.provider.IBMJCE
      security.provider.4=com.ibm.security.cert.IBMCertPath
      security.provider.5=sun.security.provider.Sun
      ```

      變更為這項設定：

```
security.provider.1=com.ibm.security.capi.IBMCAC
security.provider.2=com.ibm.crypto.fips.provider.IBMJCEFIPS
security.provider.3=com.ibm.jsse2.IBMJSSEProvider2
security.provider.4=com.ibm.crypto.provider.IBMJCE
security.provider.5=com.ibm.security.cert.IBMCertPath
security.provider.6=sun.security.provider.Sun
```

d. 儲存並關閉 `java.security` 檔。

10. 使用 CAC 鑑別，以 AppScan Source 管理者身分登入 AppScan Source for Analysis 或 AppScan Source 指令行介面 (CLI)。
11. 變更 AppScan Enterprise Server 使用者的預設權限以符合您組織的需求。

下一步

如果您要施行「美國聯邦資訊處理標準 (FIPS)」模式，則憑證不可為 SHA-1。施行 FIPS 模式的方法為使用 SHA-2 憑證以及執行《*IBM Security AppScan Source* 安裝與管理手冊》中所說明的 `appscanserverdbmgr_cac_fips.bat` 工具。在手冊中，找出向已啟用進行「共用存取卡 (CAC)」鑑別的 AppScan Enterprise Server 登錄 AppScan Source 資料庫 的說明。

如果要判定您擁有什麼憑證，請執行下列動作：

1. 開啟 Windows Certificate Manager：在 Windows 的「開始」功能表的「搜尋」框中，輸入 `certmgr.msc`，然後按 Enter 鍵。如果系統提示您輸入管理者密碼或進行確認，請輸入密碼或進行確認。
2. 按兩下或使用者介面**開啟**動作來開啟憑證。
3. 在憑證中選取「詳細資料」標籤。
4. 找出**簽章雜湊演算法**欄位。此欄位的值指出憑證的類型。

變更 AppScan Source 使用者密碼

如果要變更 AppScan Source 使用者密碼，您必須有**管理使用者**許可權，且必須在 AppScan Source for Analysis 中進行變更。如果您沒有此許可權，請依照本主題的指示，請管理者為您變更密碼。如果您的 AppScan Enterprise Server 是配置成使用 LDAP 鑑別或 Windows 鑑別，則本主題不適用。

程序

1. 在 AppScan Source for Analysis 中，從主工作台功能表中，選取**管理** > **管理使用者**。
2. 「管理使用者」對話框會列出現有的 AppScan Source 使用者。如果要變更其中一個使用者的密碼，請完成下列其中一項作業來編輯使用者資訊：
   - 按兩下使用者。
   - 用滑鼠右鍵按一下使用者，並選擇**編輯使用者**。
   - 選取使用者，並按一下**編輯使用者**按鈕。

   註：您無法從 AppScan Source 中變更 AppScan Enterprise Server 使用者的密碼。
3. 在「編輯使用者」對話框中，輸入新密碼，然後在**確認密碼**欄位中再次輸入密碼。
4. 按一下**確定**來變更密碼。

AppScan Enterprise Server SSL 憑證

安裝 AppScan Enterprise Server 時，其應該配置成使用有效的 SSL 憑證。如果未執行這項作業，當您從 AppScan Source for Analysis 或 AppScan Source 指令行介面 (CLI)，或 Windows 和 Linux 上的 AppScan Source for Development 登入伺服器時，會收到一則非授信連線的訊息。

SSL 憑證儲存體位置

永久接受的憑證會儲存在 <data_dir>\config\cacertspersonal 和 <data_dir>\config\cacertspersonal.pem（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述） 中。如果您不再要永久儲存憑證，請移除這兩個檔案。

AppScan Source for Automation 和 SSL 憑證驗證

當您使用 AppScan Source for Automation 時，依預設會自動接受憑證。這個行為取決於自動化伺服器配置檔 (<data_dir>\config\ounceautod.ozsettings（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）) 中的 ounceautod_accept_ssl 設定。如果編輯這項設定將 value="true" 設為 value="false"，則會嘗試進行 SSL 驗證；如果發現無效的憑證，則登入或發佈至 AppScan Enterprise Console 會失敗並產生錯誤。

AppScan Source 指令行介面 (CLI) 和 SSL 憑證驗證

依預設，當使用 CLI login 指令時，會嘗試 SSL 驗證，如果發現無效憑證，登入或發佈至 AppScan Enterprise Console 將會失敗並發生錯誤（如果您沒有在透過另一個 AppScan Source 用戶端產品來登入時永久接受憑證的話）。您可以在發出 login 指令時，利用選項 -acceptssl 參數來修改這個行為。當使用這個參數時，會自動接受 SSL 憑證。

AppScan Source 和協助工具

協助工具會影響有身體障礙（像是行動不便或視力不佳）的使用者。協助工具的問題會妨礙順利使用軟體產品的能力。這個主題描述已知的 AppScan Source 協助工具問題和其暫行解決方法。

對 AppScan Source 安裝程式使用「JAWS 螢幕閱讀軟體」

在執行 AppScan Source 安裝程式時，如果要使用 Freedom Scientific JAWS (http://www.freedomscientific.com/products/fs/jaws-product-page.asp)，就必須在 AppScan Source JVM 中安裝 Java Access Bridge。這樣可以讓 JAWS 適當地讀出安裝程式畫面中的標籤及控制項。

- 在 http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136191.html 可以找到 Java Access Bridge 的相關資訊（包括下載鏈結和安裝指示）。
- 在 http://kb.flexerasoftware.com/selfservice/documentLink.do?externalID=Q200311 可以找到安裝 Java Access Bridge 之 InstallAnywhere 需求的相關資訊。

在具有說明文字的使用者介面中使用「JAWS 螢幕閱讀軟體」

AppScan Source 使用者介面的許多部分都含有說明文字。在大部分情況下，您必須使用 JAWS Insert+B 按鍵才能讀取這項說明文字。

注意事項

本資訊係針對 IBM 在美國所提供之產品與服務所開發；而在其他國家中，IBM 不見得有提供本文件所提及之各項產品、服務或功能。請洽詢當地的 IBM 業務代表，以取得當地目前提供的產品和服務之相關資訊。本文件在提及 IBM 的產品、程式或服務時，不表示或暗示只能使用 IBM 的產品、程式或服務。只要未侵犯 IBM 之智慧財產權，任何功能相當之產品、程式或服務皆可取代 IBM 之產品、程式或服務。不過，任何非 IBM 之產品、程式或服務，使用者必須自行負責作業之評估和驗證責任。

本文件所說明之主題內容，IBM 可能擁有其專利或專利申請案。本文件使用者並不享有前述專利之任何授權。您可以書面提出授權查詢，來函請寄到：

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

如果是有關雙位元組字集 (DBCS) 資訊的授權查詢，請洽詢所在國家/地區的 IBM 智慧財產部門，或書面提出授權查詢，來函請寄到：

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

下列段落若與該國之法律條款抵觸，即視為不適用：

INTERNATIONAL BUSINESS MACHINES CORPORATION 僅以現狀提供本書，而不提供任何明示或默示之保證（包括但不限於未侵害他人之智慧財產權、可售性或符合特定效用的保證）。

若有些地區在某些交易上並不允許排除上述保證，則該排除無效。

本資訊中可能會包含技術上或排版印刷上的訛誤。因此，IBM 會定期修訂；並將修訂後的內容納入新版中。IBM 隨時會改進及/或變更本出版品所提及的產品及/或程式，不另行通知。

這項資訊中任何對非 IBM 網站的敘述僅供參考，IBM 對該網站並不提供保證。這些網站所提供的資料不是本 IBM 產品的資料內容，如果要使用這些網站的資料，您必須自行承擔風險。

IBM 得以各種 IBM 認為適當的方式使用或散布 貴客戶提供的任何資訊，而無需對 貴客戶負責。

如果本程式之獲授權人為了 (i) 在個別建立的程式和其他程式（包含本程式）之間交換資訊，以及 (ii) 相互使用所交換的資訊，因而需要相關的資訊，請洽詢：

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

上述資料之取得有其特殊要件，在某些情況下必須付費方得使用。

IBM 基於雙方之「IBM 客戶合約」、「IBM 國際程式授權合約」或任何同等合約之條款，提供本文件中所說的授權程式與其所有適用的授權資料。

這裡包含的效能資料是在控制環境下所決定。因此，在其他作業環境中獲得的結果可能有明顯的差異。在開發層次的系統上可能已進行一些測量，但不保證這些測量在一般可用系統上會有相同的結果。再者，部分測量可能是經由推斷來預估。實際結果可能不同。本文件的使用者應驗證適用於其特定環境的資料。

本文件所提及的非 IBM 產品資訊，取自產品的供應商、其公佈聲明或其他公開管道。IBM 並未測試這些產品，因此無法確認其效能的準確性、相容性以及其他任何與非 IBM 產品相關的要求。有關非 IBM 產品的性能問題，應直接洽詢產品供應商。

有關 IBM 未來動向的任何陳述，僅代表 IBM 的目標，並可能於未事先聲明的情況下有所變更或撤回。

顯示的所有 IBM 價格皆為 IBM 建議的零售價格，且為目前的價格，得隨時變更，不另行通知。經銷商價格可能有所不同。

本資訊僅限於規劃用途。在所述之產品上市之前，此處的資訊可能隨時更動。

這項資訊含有日常商業運作所用之資料和報告範例。為求儘可能地完整說明，範例包括了個人、公司、品牌和產品的名稱。所有這些名稱全為虛構，任何與實際商場企業使用的名稱及地址類似之處，純屬巧合。

著作權：

本資訊含有原始語言之範例應用程式，用以說明各作業平台中之程式設計技術。 貴客戶可以為了研發、使用、銷售或散佈符合範例應用式所適用的作業平台之應用程式介面的應用程式，以任何形式複製、修改及散佈這些範例程式，不必向 IBM 付費。這些範例尚未經過所有情況的完整測試。因此，IBM 不保證或暗示這些程式的可靠性、有用性或功能。貴客戶可以為了研發、使用、銷售或散布符合 IBM 應用程式介面的應用程式，以任何形式複製、修改及散布這些範例程式，不必向 IBM 付費。

這些範例程式或任何衍生成果的每份複本或任何部分，都必須依照下列方式併入著作權聲明：

©（貴公司名稱）（年份）。本程式的若干部分係衍生自 IBM 公司的範例程式。 © Copyright IBM Corp. _enter the year or years_. All rights reserved.

若 貴客戶正在閱讀本項資訊的電子檔，可能不會有照片和彩色說明。

商標

IBM、IBM 標誌及 ibm.com® 是 International Business Machines Corp 在全球許多司法管轄區註冊的商標或註冊商標。其他產品和服務名稱可能是 IBM 或其他公司的商標。如需目前的 IBM 商標清單，請參閱網路上的「著作權與商標資訊」，網址是：www.ibm.com/legal/copytrade.shtml。

Adobe、PostScript 及所有以 Adobe 為基礎的商標都是 Adobe Systems Incorporated 在美國及（或）其他國家或地區的註冊商標或商標。

IT Infrastructure Library 是 Central Computer and Telecommunications Agency（現已納入 Office of Government Commerce）的註冊商標。

Intel、Intel 標誌、Intel Inside、Intel Inside 標誌、Intel Centrino、Intel Centrino 標誌、Celeron、Intel Xeon、Intel SpeedStep、Itanium 及 Pentium 是 Intel Corporation 或其子公司在美國及（或）其他國家或地區的商標或註冊商標。

Linux 是 Linus Torvalds 在美國及（或）其他國家或地區的商標。

Microsoft、Windows、Windows NT 及 Windows 標誌是 Microsoft Corporation 在美國及（或）其他國家或地區的商標。

ITIL 是 Office of Government Commerce 的註冊商標和註冊社群商標，且是在美國專利及商標局註冊的商標。

UNIX 是 The Open Group 在美國及其他國家或地區的註冊商標。

Java 和所有以 Java 為基礎的商標和標誌是 Oracle 及/或其子公司的商標或註冊商標。

Cell Broadband Engine 是 Sony Computer Entertainment, Inc. 在美國及（或）其他國家或地區的商標，並獲其授權使用。

Linear Tape-Open、LTO、LTO 標誌、Ultrium 及 Ultrium 標誌是 HP、IBM Corp. 和 Quantum 在美國及其他國家或地區的商標。

著作權

IBM、IBM 標誌、ibm.com Rational、AppScan、Rational Team Concert、WebSphere 及 ClearQuest 是 IBM 公司在全世界許多適用範圍註冊的商標或註冊商標。其他產品和服務名稱可能是 IBM 或其他公司的商標。如需目前的 IBM 商標清單，請參閱網路上的「著作權與商標資訊」，網址是：http://www.ibm.com/legal/copytrade.shtml。Linux 是 Linus Torvalds 在美國及/或其他國家或地區的註冊商標。Microsoft、Windows、Windows NT 和 Windows 標誌是 Microsoft Corporation 在美國及/或其他國家或地區的商標。Unix 是 The Open Group 在美國及其他國家或地區的註冊商標。Java 和所有以 Java 為基礎的商標和標誌是 Oracle 及/或其子公司的商標或註冊商標。

本程式包含：Jacorb 2.3.0（Copyright 1997-2006 JacORB 專案）及 XOM1.0d22 (Copyright 2003 Elliotte Rusty Harold)，每一個都依 「GNU 程式庫通用公用授權 (Gnu Library General Public License, LGPL)」來提供，您可以在本程式隨附的「注意事項」檔案中找到此 LGPL 的副本。

第 2 章 配置應用程式和專案

掃描之前，您必須先配置應用程式和專案。本節說明「應用程式探索助理」、「新建應用程式」精靈以及「新建專案精靈」。您將學習如何配置 AppScan Source for Analysis 的屬性。另外，這個區段也教導您如何新增現有的應用程式和專案來進行掃描，以及如何新增檔案到專案中。

AppScan Source for Analysis 配置包括建立應用程式、配置原始碼及配置屬性。您先配置及掃描，然後進行分類。您可以在「內容」視圖或利用「新建專案精靈」，來配置您的原始碼。本章帶您逐步完成這個精靈。請參閱第 248 頁的『「內容」視圖』，以取得應用程式和專案內容的概觀。

AppScan Source for Analysis 所使用的應用程式/專案模型，會直接匯入先前利用 AppScan Source 公用程式所建立的 Microsoft Visual Studio、Eclipse、Rational Application Developer for WebSphere 軟體 (RAD) 或 AppScan Source 專案（請參閱《*IBM Security AppScan Source Utilities* 使用手冊》，以取得進一步的詳細資料）。

您可以新增及配置類型各不相同且包含各種語言的專案 - 指定從目標程式碼庫及其建置程序收集來的各項設定。在配置期間，您可以指定要從掃描中排除的目錄和檔案。

掃描之前，您必須先配置應用程式和專案。應用程式是一個專案儲存器；專案是一組要掃描的檔案及所用的設定（配置）。

AppScan Source 應用程式和專案檔

AppScan Source 應用程式和專案有對應的檔案，可維護掃描及分類自訂作業所需的配置資訊。建議您將這些檔案與原始碼放在同一個目錄下，因為建置專案所需的配置資訊（相依關係、編譯器選項等），與 AppScan Source 順利掃描這些檔案所需的配置資訊非常類似。最佳實務包含使用來源控制系統管理這些檔案。

在 AppScan Source for Analysis 中建立的應用程式和專案其副檔名分別為 `.paf` 和 `.ppf`。當您在 AppScan Source for Analysis、AppScan Source for Automation 和 AppScan Source 指令行介面 中手動建立及配置應用程式或專案時，就會產生這些檔案。

在 Windows 上，當您將 Visual Studio 解決方案和專案匯入 AppScan Source for Analysis、AppScan Source for Automation 和 AppScan Source 指令行介面 時，即會針對它們建立副檔名為 `.sln.gaf` 和 `.vcproj.gpf` 的檔案。

在 macOS 上，當您匯入 Xcode 目錄和專案時，會針對它們建立副檔名為 `.xcodeproj.gaf` 和 `.xcodeproj.gpf` 的檔案。同樣地，當您匯入 Xcode 工作區時，就會建立副檔名為 `.xcworkspace.gaf` 的檔案。

註：當 Eclipse 匯入器在 Eclipse 或 Rational Application Developer for WebSphere 軟體 (RAD)工作區執行時，AppScan Source 會建立副檔名為 `.ewf` 和 `.epf` 的中間檔。最初匯入到 AppScan Source for Analysis 及以後掃描時會用到這些檔案。

重要：如果您使用的 AppScan Source 專案在開發環境中（例如 IBM MobileFirst Platform 專案）有相依關係，請務必先在開發環境中建置專案，然後再匯入它。匯入專案之後，如果您修改其中的檔案，請務必先在開發環境中重建它，再於 AppScan Source 中掃描（如果不這樣做，AppScan Source 會忽略您對檔案所做的修改）。

表 2. *AppScan Source* 檔案

AppScan Source 副檔名	說明
ppf	• AppScan Source 專案檔 • 在您以 AppScan Source for Analysis 或支援的 AppScan Source 公用程式建立專案時產生 • User-named
paf	• AppScan Source 應用程式檔案 • 在您以 AppScan Source for Analysis 或支援的 AppScan Source 公用程式建立應用程式時產生 • User-named

sln.gaf	• 當您匯入 Visual Studio 解決方案時所產生的 AppScan Source 應用程式檔案
	• 用來保留自訂應用程式資訊，例如排除和組合
	• 採用匯入的工作區名稱或解決方案名稱。例如：
	`d:\my_apps\myapp.sln`
	`d:\my_apps\myapp.sln.gaf`

vcproj.gpf	• 當您匯入 Visual Studio 專案時所產生的 AppScan Source 專案檔
	• 用來保留自訂專案資訊，例如型樣和排除
	• 採用匯入專案的名稱。例如：
	`d:\my_projects\myproject.vcproj`
	`d:\my_projects\myproject.vcproj.gpf`

xcodeproj.gaf	• 當您匯入 Xcode 目錄時所產生的 AppScan Source 應用程式檔
	• 用來保留自訂應用程式資訊，例如排除和組合
	• 採用匯入的工作區名稱或解決方案名稱。例如：
	`/Users/myUser/myProject.xcodeproj`
	`/Users/myUser/myProject.xcodeproj.gaf`

表 2. *AppScan Source* 檔案 (繼續)

AppScan Source 副檔名	**說明**
xcodeproj.gpf	• 當您匯入 Xcode 專案時所產生的 AppScan Source 專案檔 • 用來保留自訂專案資訊，例如型樣和排除 • 採用匯入的專案名稱，例如： /Users/myUser/myProject.xcodeproj /Users/myUser/myProject.xcodeproj.gpf
xcworkspace.gaf	• 當您匯入 Xcode 工作區時所產生的 AppScan Source 應用程式檔案 • 用來保留自訂應用程式資訊，例如排除和組合 • 採用匯入的工作區名稱。例如： /Users/myUser/myProj.xcworkspace.gaf
ewf	• Eclipse 工作區檔案 • 當您將 Eclipse 工作區匯入到 AppScan Source 時產生 • Eclipse 匯出器會根據 Eclipse 工作區中的資訊來建立檔案，然後 AppScan Source 即匯入檔案
epf	• Eclipse 專案檔 • 當您將 Eclipse 專案匯入到 AppScan Source 時產生 • Eclipse 匯出器會根據 Eclipse 專案中的資訊來建立檔案，然後 AppScan Source 即匯入檔案

提示：當您使用受支援的建置整合工具（例如，Ounce/Ant 或 Ounce/Maven）來產生 AppScan Source 應用程式和專案檔時，建議您在建置自動化的過程中更新來源控制中的這些檔案，以便促進在整個開發團隊之間共用這些檔案。當開發人員在來源控制中更新檔案的本端視圖時，AppScan Source 應用程式和專案檔也會一起更新。這可確保整個團隊使用一組一致的檔案。

註：如果要瞭解 AppScan Source for Analysis、AppScan Source for Automation 和 AppScan Source 指令行介面 支援哪些版本的匯入檔案，請參閱http://www.ibm.com/support/docview.wss?uid=swg27027486。在此頁面上，選取您使用的 AppScan Source 版本的標籤 - 然後選取您使用的 AppScan Source 元件。如果 AppScan Source 支援開啟及掃描來自其他開發環境的檔案，該支援會列在**支援的軟體**標籤的**編譯器與語言**區段中。

配置應用程式

您可以使用「新建應用程式」精靈或「應用程式探索助理」，來建立應用程式。「應用程式探索助理」會自動為您設定應用程式，而「新建應用程式」精靈可讓您新增應用程式，並引導您完成配置程序。這個精靈會協助您手動建立專案，或新增現有的專案至應用程式中。本節說明這兩種用來新增應用程式的方法和一些基本的配置作業。

註：「應用程式探索助理」可針對 Java 原始碼和 Microsoft Visual Studio 解決方案（或是含有 Java 專案的 Eclipse 或 IBM Rational Application Developer for WebSphere 軟體 (RAD) 工作區），快速建立及配置應用程式和專案。如果要針對其他任何支援的語言建立應用程式，請使用「新建應用程式」精靈 - 或是將支援的應用程式匯入 AppScan Source for Analysis。

在新增專案之前，您必須建立一個新的應用程式（請參閱第 33 頁的『使用「新建應用程式」精靈來建立新的應用程式』或第 33 頁的『使用應用程式探索助理來建立應用程式和專案』），或新增現有的應用程式（請參閱第 36 頁的『新增現有的應用程式』）。如果您使用 Microsoft Visual Studio，您已在專案中排列您的原始檔。AppScan Source for Analysis 可讓您匯入解決方案，將它們當作 AppScan Source 應用程式來處理。

下表列出您可以利用 AppScan Source for Analysis 來開啟及掃描的應用程式檔案類型。

表 3. 支援的應用程式檔案類型

應用程式	檔案類型
Microsoft Visual Studio **註：**如果要瞭解 AppScan Source for Analysis、AppScan Source for Automation 和 AppScan Source 指令行介面 支援哪些版本的匯入檔案，請參閱http://www.ibm.com/support/docview.wss?uid=swg27027486。在此頁面上，選取您使用的 AppScan Source 版本的標籤 - 然後選取您使用的 AppScan Source 元件。如果 AppScan Source 支援開啟及掃描來自其他開發環境的檔案，該支援會列在**支援的軟體**標籤的**編譯器與語言**區段中。	`.sln`（解決方案）
• Eclipse 工作區（僅限 Java） • RAD 工作區（僅限 Java） 請參閱AppScan Source系統需求，以瞭解支援哪些版本的 Eclipse 和 RAD 來進行工作區掃描。	`<workspace directory>` 或 `.ewf` 工作區目錄還包含另一個目錄 `.metadata`。
AppScan Source 應用程式檔案	`.paf`

提示： 在「瀏覽器」視圖中，會出現一個表示匯入之應用程式的圖示（請參閱第 77 頁的『應用程式和專案指示器』）。

註：當利用「新建應用程式精靈」和「新建專案」精靈來建立應用程式和專案時，會根據在精靈中輸入的**名稱**，自動指派它們的檔名（比方說，如果是建立專案，且在**名稱**欄位中輸入 **MyProject**，專案的檔名就是 MyProject.ppf)。應用程式和專案名稱可利用「內容」視圖來重新命名。

使用「新建應用程式」精靈來建立新的應用程式

程序

1. 完成下列動作之一：
   - 從主功能表列中，選取**檔案** > **新增應用程式** > **建立新的應用程式**。
   - 在「瀏覽器」視圖工具列中，按一下**新增應用程式**功能表向下箭頭按鈕，然後從功能表中選取**建立新的應用程式**。
   - 在「瀏覽器」視圖中，用滑鼠右鍵按一下**所有應用程式**，然後從功能表中選取**新增應用程式** > **建立新的應用程式**。
2. 輸入應用程式的**名稱**。
3. 瀏覽至用來儲存應用程式的**工作目錄**。新應用程式檔案的副檔名是 .paf。
4. 按**下一步**來配置組成應用程式的專案，或按一下**完成**來新增應用程式，不配置任何專案。本節之後會提供配置和新增專案的說明。

使用應用程式探索助理來建立應用程式和專案

AppScan Source 包含功能強大的應用程式探索助理，可讓您為 Java 原始碼和 Microsoft Visual Studio 解決方案快速建立及配置應用程式和專案。「應用程式探索助理」也可讓您尋找包含 Java 專案的 Eclipse 或 Rational Application Developer for WebSphere 軟體 (RAD) 工作區。應用程式探索助理可讓您指向來源、解決方案或工作區目錄，然後交由 AppScan Source 處理其餘工作。

關於這項作業

您可以使用應用程式探索助理來搜尋包含 Java 原始碼、Microsoft Visual Studio 解決方案及/或 Eclipse 工作區組合的位置。應用程式探索助理的最終畫面可讓您指定僅限 Java 的應用程式/專案結構喜好設定。這個畫面與 Microsoft Visual Studio 解決方案或 Eclipse 工作區的應用程式及專案檔位置沒有關聯，應用程式檔案會自動放在解決方案或工作區的根目錄，專案檔會自動放在個別解決方案或工作區專案的根目錄。

程序

1. 完成下列其中一個動作來啟動應用程式探索助理：
   - 從主功能表列中，選取**檔案** > **新增應用程式** > **探索應用程式**。
   - 在「瀏覽器」視圖的**快速入門**區段中，選取**探索應用程式**。
   - 在「瀏覽器」視圖工具列中，按一下**新增應用程式**功能表向下箭頭按鈕，然後從功能表中選取**探索應用程式**。
   - 在「瀏覽器」視圖中，用滑鼠右鍵按一下**所有應用程式**，然後從功能表中選取**新增應用程式** > **探索應用程式**。
2. 在「搜尋位置」畫面中，指定包含您要掃描的原始碼、解決方案或工作區的位置。此外，您也可以設定完成應用程式探索之後立即開始掃描。

從這裡，您可以按**下一步**來設定其他應用程式探索助理選項（例如，外部相依關係規格、排除規則及 Java 應用程式/專案結構喜好設定），或按一下**開始**，啟動應用程式探索。如果您按一下**開始**：

- 不會設定外部相依關係位置。如果應用程式有外部相依關係，但未指定，則會對掃描結果造成不良影響。
- 將使用預設排除規則（如需預設規則清單，請參閱第 36 頁的『預設應用程式探索助理排除規則』）。
- 如果您尋找 Java 原始碼，則會建立一個專案和應用程式（單一專案包含找到的所有原始碼根目錄）。

如果您按**下一步**，會繼續到下一步。

3. 在「外部相依關係」畫面中，設定應用程式具有的每一個外部相依關係的路徑（例如，JDK 或 Web 伺服器的路徑）。請遵循下列指示來完成這個畫面：
   a. 如果要新增外部相依關係，請在表格內按一下，或按一下**新增**，然後輸入或瀏覽外部相依關係路徑。如果要透過鍵盤來接受輸入的路徑，請按鍵盤的 Enter 鍵。

      提示：在編輯相依關係路徑欄位時輸入，將會列出可供您選取的目錄。您至少必須輸入一個磁碟機代號。對於指定的路徑，將會列出其中包含的所有資料夾。
   b. 如果要移除外部相依關係路徑，請選取它並按一下**刪除**。
   c. 如果要修改外部相依關係路徑，請在路徑內按一下，然後輸入或瀏覽外部相依關係路徑。

   從這裡，您可以按**下一步**來設定其他應用程式探索助理選項，或按一下**開始**，啟動應用程式探索。如果您按一下**開始**：

   - 將使用預設排除規則（如需預設規則清單，請參閱第 36 頁的『預設應用程式探索助理排除規則』）。
   - 如果您尋找 Java 原始碼，則會建立一個專案和應用程式（單一專案包含找到的所有原始碼根目錄）。

   如果您按**下一步**，會繼續到下一步。
4. 在「排除規則」畫面中，指定濾除檔案和目錄的規則。請以 PERL、Grep、EGrep 或完全相符正規表示式來設定規則。例如，如果您要在應用程式探索搜尋時排除名為 `temp` 的目錄，您可以新增 PERL `.*[\\/]temp` 排除規則。

   依預設，會提供一組 PERL 正規表示式來排除一些共用目錄（如需完整清單，請參閱第 36 頁的『預設應用程式探索助理排除規則』）。如果要修改這份清單或建立新的規則，請遵循下列指示：
   a. 如果要修改現有的排除規則，請在規則內按一下以啟動規則編輯器。一旦您已完成編輯規則，請按一下離開，或按鍵盤 **Enter** 鍵。

      如果要修改現有規則的正規表示式類型，請在規則的**正規表示式類型**資料格內按一下，然後從功能表中選取正規表示式類型。
   b. 如果要新增排除規則，請按一下**新增**。這樣會將新的規則新增至表格，您可以遵循上方修改規則的指示來變更此規則。

c. 如果要移除排除規則，請選取它，然後按一下 **刪除**（或按一下 **全部刪除** 以移除畫面中所列出的全部排除規則）。

重要：表格中以勾號表示有效的排除規則，以紅色 X 表示無效的規則。所有規則必須有效，否則無法啟動應用程式探索或在應用程式探索助理中繼續。

從這裡：

- 如果您只是搜尋 Java 原始碼，您可以按**下一步**來設定應用程式探索助理應用程式/專案結構喜好設定，或按一下**開始**來執行助理。
- 如果您只是搜尋 Microsoft Visual Studio 解決方案或 Eclipse 工作區，按一下**開始**來執行助理。按**下一步**時，助理會進入僅適用於 Java 原始碼探索的畫面。

如果您按**下一步**，會繼續到下一步。

5. 應用程式和專案建立畫面僅適用於 Java 原始碼探索。請在其中指定將建立的應用程式和專案的結構：
   a. 如果要為所有找到的原始碼根目錄建立單一專案，請在**專案**功能表中選取**建立單一專案**。這樣選取時，您只能選擇建立單一應用程式。
   b. 如果要為找到的每一個原始碼根目錄建立個別專案，請在**專案**功能表中選取**為發現的每一個原始碼根目錄建立一個專案**。這樣選取時，您可以選擇建立一個應用程式或多個應用程式。如果要建立單一應用程式來包含所有建立的專案，請在**應用程式**功能表中選取**建立單一應用程式**。如果要為每一個建立的專案分別建立應用程式，請在**應用程式**功能表中選取**建立每個專案的應用程式**。

   此外，選擇位置來儲存應用程式及專案定義檔。

   如果您選擇**為我組織檔案**：
   - 如果您建立單一專案，則會在搜尋位置建立專案及應用程式檔案。
   - 如果您在單一應用程式中為每一個原始碼根目錄建立專案，則會在原始碼根目錄的上層目錄中建立每一個原始碼根目錄的專案檔，並且在搜尋位置中建立應用程式檔案。
   - 如果您為每一個原始碼根目錄建立專案，並為每一個專案建立應用程式，則會在原始碼根目錄的上層目錄中建立每一個原始碼根目錄的專案和應用程式檔案。

   如果您指定目錄，則會在該目錄中建立所有應用程式及專案檔。
6. 如果您要變更先前畫面中所做的任何設定，請按**上一步**。當滿意 應用程式探索 設定時，請按一下**開始**以掃描搜尋位置來尋找原始碼根目錄。

結果

當應用程式探索完成時，由應用程式探索建立的新的應用程式和專案會出現在「瀏覽器」視圖中，即備妥可進行掃描（如果您已設為完成應用程式探索之後立即開始掃描，則會開始掃描）。

如果探索期間發現問題，應用程式探索助理完成時會提供探索報告。例如，如果應用程式還有「外部相依關係」畫面中未指定的其他外部相依關係，則報告會包含警告，指出無法解析外部相依關係。在探索報告中：

- 按一下**完成**，建立應用程式和專案。如果選取**忽略警告，繼續掃描**，則會立即掃描應用程式和專案。
- 按**上一步**，變更應用程式探索助理設定或重新執行應用程式探索。
- 按一下**取消**，關閉探索報告而不建立應用程式或專案。

預設應用程式探索助理排除規則

使用應用程式探索助理時，如果未修改「排除規則」畫面，或如果您在指定搜尋目錄後啟動應用程式探索，則會使用預設排除規則。這個主題中列出預設應用程式探索排除規則。

表 *4.* 預設應用程式探索排除規則

排除規則	正規表示式類型
.*[\\/]example	PERL
.*[\\/]test	PERL
.*[\\/]demo	PERL
.*[\\/]sample	PERL

新增現有的應用程式

您可以將現有的應用程式拖放到「瀏覽器」視圖中，或使用**新增應用程式**動作，以新增現有的應用程式來進行掃描。此外，您也可以將 WAR 和 EAR 檔拖放至「瀏覽器」視圖中來新增它們。

如果要瞭解如何新增現有的應用程式，請參閱這些主題：

- 『利用使用者介面動作來新增現有的應用程式』
- 第 37 頁的『利用拖放動作來新增現有的應用程式』

利用使用者介面動作來新增現有的應用程式

程序

1. 完成下列動作之一：
   - 從主工作台功能表中，選取**檔案** > **新增應用程式** > **開啟現有的應用程式**。
   - 在「瀏覽器」視圖工具列中，按一下**新增應用程式**功能表向下箭頭按鈕，然後從功能表中選取**開啟現有的應用程式**。
   - 在「瀏覽器」視圖中，用滑鼠右鍵按一下**所有應用程式**，然後從功能表中選取**新增應用程式** > **開啟現有的應用程式**。
2. 選取包含儲存之應用程式檔案（.paf、.sln、.dsw 或 .ewf）的目錄。

   註：如果要瞭解 AppScan Source for Analysis、AppScan Source for Automation 和 AppScan Source 指令行介面 支援哪些版本的匯入檔案，請參閱http://www.ibm.com/support/docview.wss?uid=swg27027486。在此頁面上，選取您使用的 AppScan Source 版本的標籤 - 然後選取您使用的 AppScan Source 元件。如果 AppScan Source 支援開啟及掃描來自其他開發環境的檔案，該支援會列在**支援的軟體**標籤的**編譯器與語言**區段中。
3. 開啟應用程式檔案。

利用拖放動作來新增現有的應用程式

程序

1. 在工作站中，找出要新增來進行掃描的應用程式（.paf、.war、.ear、.sln、.dsw 或 .ewf）。您也可以新增含有 .war 或 .ear 檔的目錄（在一些應用程式伺服器上，其稱為放入資料夾）。

   註：您無法拖放 Eclipse 工作區目錄。

   註：如果您要新增 .war 或 .ear 檔，或是含有 .war 或 .ear 檔的目錄，這些檔案必須位於本端檔案系統上或是對映磁碟機中。

   註：如果要瞭解 AppScan Source for Analysis、AppScan Source for Automation 和 AppScan Source 指令行介面 支援哪些版本的匯入檔案，請參閱http://www.ibm.com/support/docview.wss?uid=swg27027486。在此頁面上，選取您使用的 AppScan Source 版本的標籤 - 然後選取您使用的 AppScan Source 元件。如果 AppScan Source 支援開啟及掃描來自其他開發環境的檔案，該支援會列在**支援的軟體**標籤的**編譯器與語言**區段中。
2. 選取應用程式，然後將它拖曳到「瀏覽器」視圖。
3. 將選項放在**所有應用程式**節點上，或放在它下面。
4. 如果您要新增 .war 或 .ear 檔，或是含有 .war 或 .ear 檔的目錄，會開啟一個對話框，可讓您指定這些檔案要部署到該處的應用程式伺服器。完成這個對話框之後，按一下**確定**。

新增多個應用程式

在第一次開始使用 AppScan Source for Analysis 時，您可能會想匯入多個應用程式，而不是一次只新增一個應用程式。「選取應用程式」對話框可讓您選取一個根目錄，以便從中搜尋 AppScan Source 應用程式 (.paf) 或 Visual Studio 解決方案檔案 (.sln)。您可以將多個應用程式拖放到「瀏覽器」視圖中，來新增多個要掃描的應用程式。

如果要瞭解如何新增多個應用程式，請參閱這些主題：

- 『利用使用者介面動作來新增多個應用程式』
- 第 38 頁的『利用拖放動作來新增多個應用程式』

註：如果要新增多個 WAR 和 EAR 檔，您可以拖放包含這些檔案的目錄。如需相關資訊，請參閱『利用拖放動作來新增現有的應用程式』。

利用使用者介面動作來新增多個應用程式

程序

1. 從主工作台功能表中，選取**檔案** > **新增應用程式** > **多個應用程式**。
2. 在「選取應用程式」對話框中，瀏覽至含有您要匯入之應用程式的根目錄。選取**遞迴進入子目錄**勾選框，以便在子目錄中搜尋。
3. 完成下列動作之一：
   - 按一下**完成**來匯入應用程式，將它們新增到「瀏覽器」視圖中。
   - 按**下一步**，檢視搜尋結果，並且選取要匯入的應用程式。然後按一下**完成**。

註：如果要瞭解 AppScan Source for Analysis、AppScan Source for Automation 和 AppScan Source 指令行介面 支援哪些版本的匯入檔案，請參閱http://www.ibm.com/support/docview.wss?uid=swg27026486。在此頁面上，選取您使用的 AppScan Source 版本的標籤 - 然後選取您使用的 AppScan Source 元件。如果 AppScan Source 支援開啟及掃描來自其他開發環境的檔案，該支援會列在**支援的軟體**標籤的**編譯器與語言**區段中。

利用拖放動作來新增多個應用程式

程序

1. 在工作站中，找出要新增來進行掃描的應用程式（.paf、.sln、.dsw 或 .ewf 檔）。

   註：您無法拖放 Eclipse 工作區目錄。

   註：如果要瞭解 AppScan Source for Analysis、AppScan Source for Automation 和 AppScan Source 指令行介面 支援哪些版本的匯入檔案，請參閱http://www.ibm.com/support/docview.wss?uid=swg27026486。在此頁面上，選取您使用的 AppScan Source 版本的標籤 - 然後選取您使用的 AppScan Source 元件。如果 AppScan Source 支援開啟及掃描來自其他開發環境的檔案，該支援會列在**支援的軟體**標籤的**編譯器與語言**區段中。
2. 選取一或多個應用程式，然後將它們拖曳到「瀏覽器」視圖中。
3. 將選項放在**所有應用程式**節點上，或放在它下面。

從 Apache Tomcat 及 WebSphere Application Server Liberty 設定檔應用程式伺服器，匯入現有的 Java 應用程式

如果您有已部署到支援的應用程式伺服器的現有 Java 應用程式，您可以自動將它們匯入到 AppScan Source。

開始之前

如需瞭解支援的 Apache Tomcat 及 WebSphere Application Server Liberty 設定檔版本，請參閱 AppScan Source 系統需求。在此頁面上，選取您使用的 AppScan Source 版本的標籤 - 然後選取 AppScan Source for Analysis 元件。在**支援的軟體**區段中可以找到支援的應用程式伺服器。

程序

1. 完成下列動作之一：
   - 從主工作台功能表中，選取**檔案** > **新增應用程式** > **從應用程式伺服器匯入**。
   - 在「瀏覽器」視圖工具列中，按一下**新增應用程式功能表**向下箭頭按鈕，然後從功能表中選取**從應用程式伺服器匯入**。
   - 在「瀏覽器」視圖中，用滑鼠右鍵按一下**所有應用程式**，然後從功能表中選取**新增應用程式** > **從應用程式伺服器匯入**。
2. 在「從應用程式伺服器匯入」對話框中，按一下**瀏覽**，找出並選取應用程式伺服器的安裝位置 - 或在欄位中輸入伺服器路徑及目錄，然後按一下**搜尋**，在輸入的位置中搜尋應用程式。如果該位置已辨識為支援的應用程式伺服器，則對話框的**要匯入的應用程式**區段中會列出可用的應用程式。在此區段中，選取您要匯入的應用程式，然後按一下**確定**。

3. 針對從應用程式伺服器匯入的每一個應用程式，將會建立 AppScan Source 應用程式。

結果

如果是從 WebSphere Application Server Liberty 設定檔伺服器（WebSphere Application Server 8.5 版以及更新版本）匯入，您可能會收到一則訊息指出需要手動進行 JSP 前置編譯。這是因為 Liberty 設定檔伺服器不含獨立式 JSP 編譯器。如果您收到這則訊息，請刪除由於匯入而建立的任何應用程式 - 然後遵循『針對 WebSphere Application Server Liberty 設定檔產生經過前置編譯的 JavaServer Pages』中的指示，再次從應用程式伺服器匯入。

匯入應用程式時，依預設，AppScan Source 只會掃描其 JSP 檔和 web-inf/classes 的內容。並不會掃描 web-inf/lib 的內容。如果您要掃描其他檔案，您可以使用專案內容來設定要掃描的其他副檔名（請參閱第 221 頁的『副檔名』）。例如，如果您要掃描 .jar 檔（包括web-inf/lib 中的那些檔案），請遵循第 71 頁的『修改應用程式和專案內容』中關於修改專案內容的指示。在專案的「內容」視圖中，選取第 221 頁的『副檔名』標籤。在視圖的「其他副檔名」區段中，按一下**新增副檔名**。在「新的附檔名」對話框，於**副檔名**欄位中鍵入 jar - 然後選取**掃描具有這個副檔名的檔案**，再按一下**確定**。按一下視圖右上方的**儲存**（或是從主功能表選取**檔案** > **儲存**）- 然後重新掃描專案。如果有不要掃描的檔案，您可以使用「專案」視圖的第 221 頁的『來源』標籤，將它們排除掃描。

如果伺服器上的應用程式變更，且您想要以變更的內容來重新整理 AppScan Source 應用程式，則必須再次完成上述步驟（您不需要先刪除最初建立的應用程式 - AppScan Source 會在重新匯入時自動刪除它們）。

註：如果您從伺服器匯入 .war 檔，然後從另一部伺服器匯入另一個同名的 .war 檔，則第二個 .war 檔會改寫第一個檔案。要避免這種情況，請先將第二個 .war 檔重新命名再匯入。

針對 WebSphere Application Server Liberty 設定檔產生經過前置編譯的 JavaServer Pages

如果您從 WebSphere Application Server Liberty 設定檔（WebSphere Application Server 8.5 版以及更新版本）匯入應用程式，則需要手動進行 JSP 前置編譯（Liberty 設定檔不含獨立式 JSP 編譯器）。本主題說明設定手動 JSP 前置編譯所需的步驟。

程序

1. 遵循 WebSphere Application Server Network Deployment 知識中心中，有關建立 Liberty 設定檔伺服器的指示。若為 WebSphere Application Server 8.5.5 版，請參閱使用開發人員工具來建立 Liberty 設定檔伺服器主題。
2. 在 Liberty 設定檔 server.xml 檔中，將下列程式碼新增至 server description 區段：

```
<jspEngine prepareJSPs="0"/>
<webContainer deferServletLoad="false"/>
```

例如：

```
<server description="new server">

    <!-- Enable features -->
```

```
    <featureManager>
        <feature>jsp-2.2</feature>
        <feature>localConnector-1.0</feature>
       <feature>appSecurity-2.0</feature>
        <feature>restConnector-1.0</feature>
    </featureManager>

    <!-- To access this server from a remote client
         add a host attribute to the following element,
         e.g. host="*" -->
    <httpEndpoint httpPort="9080" httpsPort="9443" id="defaultHttpEndpoint"/>

...
    <jspEngine prepareJSPs="0"/>
    <webContainer deferServletLoad="false"/>
...
</server>
```

WebSphere Application Server 中心說明 server.xml 檔案。若為 WebSphere Application Server 8.5.5 版，請參閱 Liberty 設定檔：server.xml 檔中的配置元素主題。

3. 使用下列其中一種方法，以除錯模式啟動伺服器：
   - 新增 -Dwas.debug.mode=true JVM 引數，如在 WebSphere Application Server 8.5 版 Liberty 設定檔中設定通用 JVM 引數所述。
   - 遵循 WebSphere Application Server Network Deployment 知識中心中，有關啟動及停止伺服器的指示。若為 WebSphere Application Server 8.5.5 版，請參閱使用開發人員工具來啟動及停止伺服器主題。

結果

完成這些步驟之後，遵循第 38 頁的『從 Apache Tomcat 及 WebSphere Application Server Liberty 設定檔應用程式伺服器，匯入現有的 Java 應用程式』中的步驟，從 WebSphere Application Server Liberty 設定檔匯入 Java 應用程式。

新增 Eclipse 或 Eclipse 型產品工作區

如果您有一個 Eclipse 或 Rational Application Developer for WebSphere 軟體 (RAD) 工作區含有 Java 及/或 IBM MobileFirst Platform 專案，您可以將它匯入 AppScan Source for Analysis 中。

開始之前

在新增工作區之前，請確定您已依照第 41 頁的『配置 Eclipse 和 Rational Application Developer for WebSphere 軟體 (RAD) 專案的開發環境』所說明來安裝和更新開發環境。

程序

1. 完成下列動作之一：
   - 從主工作台功能表中，選取**檔案** > **新增應用程式** > **匯入現有的 Eclipse 型工作區**。
   - 在「瀏覽器」視圖工具列中，按一下**新增應用程式功能表**向下箭頭按鈕，然後從功能表中選取**匯入現有的 Eclipse 型工作區**。
   - 在「瀏覽器」視圖中，用滑鼠右鍵按一下**所有應用程式**，然後從功能表中選取**新增應用程式** > **匯入現有的 Eclipse 型工作區**。

2. 選取**工作區類型**。
3. 瀏覽至工作區，選取目錄，然後按一下**確定**，來新增工作區。

配置 Eclipse 和 Rational Application Developer for WebSphere 軟體 (RAD) 專案的開發環境

在匯入 Eclipse 或 Rational Application Developer for WebSphere 軟體 (RAD) 專案之前，您必須適當地配置開發環境。雖然 Eclipse 是各專案類型的基礎，但對 AppScan Source 而言，不同的版本還是有所區別。

如果要瞭解 Eclipse 和 Rational Application Developer for WebSphere 軟體 (RAD) 的哪些版本受到 AppScan Source 支援，請參閱http://www.ibm.com/support/docview.wss?uid=swg27027486。

如果要進一步瞭解如何為此來配置開發環境，請參閱下列說明主題：

- 『Eclipse 或 Application Developer 更新項目』
- 『Eclipse 工作區匯入器：Eclipse 或 Rational Application Developer for WebSphere 軟體 (RAD) 喜好設定配置』

Eclipse 或 Application Developer 更新項目

對於在 AppScan Source 以外的 Eclipse 或 Application Developer 環境，您必須確定已安裝適當的軟體更新項目。這些指示說明如何取得及安裝更新項目。版本不同，程序也可能不同。

開始之前

重要： AppScan Source for Development 需要 1.5 版或更新版本的 Java 執行時期環境 (JRE)。如果您的環境指向不符合此需求的 JRE，請編輯 Eclipse 安裝目錄中的 `eclipse.ini` 檔案，使它指向符合此需求的 JRE。如需對 `eclipse.ini` 檔案進行這項變更的相關資訊，請參閱 http://wiki.eclipse.org/Eclipse.ini 中的指定 *JVM* 一節。

程序

1. 在 Eclipse 的**說明**功能表中，選取安裝新軟體的選項（功能表標籤會隨著所用的 Eclipse 版本而不同）。
2. 選取新增「本端更新網站」的選項。
3. 當提示您輸入網站位置時，請導覽至 AppScan Source 安裝目錄。
4. 新增這個更新網站，然後遵循顯示的步驟，直到提示您重新啟動 Eclipse。
5. 安裝完成之後會出現 AppScan Source 功能表。

Eclipse 工作區匯入器：Eclipse 或 Rational Application Developer for WebSphere 軟體 (RAD) 喜好設定配置

AppScan Source for Analysis 安裝架構提供預設的 Eclipse 匯入器。這個匯入器會識別 Eclipse 和 JRE 的位置。如果預設 Eclipse 匯入器無法匯入您的工作區，可能需要建立新的 Eclipse 匯入器。

開始之前

每個匯入器配置都代表一個 Eclipse 或 Rational Application Developer for WebSphere 軟體 (RAD) 安裝架構。如果要使用這些配置來將現有的工作區和專案匯入到 AppScan Source for Analysis，您可能也需要在 Eclipse 環境安裝 AppScan Source for Development 外掛程式。

新增 RAD 工作區之前，您必須先建立工作區類型的配置。

程序

1. 在 AppScan Source for Analysis 中，從主工作台功能表中，選取**編輯** > **喜好設定**。
2. 選取 **Eclipse 工作區匯入器**。
3. 按一下**建立新的配置**，然後完成「新建匯入配置」對話框，以建立新的配置：
   - **產品**：選取適當的產品。

     註：如果無法讓您選取用來建立工作區的產品，在嘗試建立工作區匯入器之前，請確定您已完成 第 41 頁的『Eclipse 或 Application Developer 更新項目』所概述的配置步驟。
   - **名稱**：匯入器名稱
   - **位置**：Eclipse 安裝架構的基本目錄路徑
   - **JRE 位置**：Java 執行時期環境 (JRE) 的根目錄路徑。請使用 `<install_dir>\JDKS`（其中 `<install_dir>` 是 AppScan Source 安裝的位置） 中的 JDK，或其他任何偏好的 JDK。
4. 按一下**確定**。
5. 如果要將匯入器識別為預設值，請選取它，然後按一下**將所選的配置設為預設值**。這時匯入器的**預設值**直欄中，會出現一個圖示。

建立應用程式的新專案

您在新增應用程式之後，新增專案到其中。可掃描的專案類型包括：Java/JSP、ASP、C/C++、COBOL、ColdFusion、.NET Assembly、Pattern Based、Perl、PHP、PL/SQL、T-SQL、Visual Basic 和 JavaScript。

關於這項作業

如果您利用 `make` 來編譯專案，建議您利用 Ounce/Make 公用程式來建立專案檔，然後新增這個專案檔。如果您利用 `ant` 來編譯專案，請利用 Ounce/Ant 來建立專案檔，然後新增專案檔。請參閱 *IBM Rational AppScan Source Edition Utilities* 使用手冊 以取得 Ounce/Make 和 Ounce/Ant 的詳細資料。

註：AppScan Source 專案的預設檔案編碼是 **ISO-8859-1**。您可以在「一般」喜好設定頁面中，變更預設檔案編碼。

註：當利用「新建應用程式精靈」和「新建專案」精靈來建立應用程式和專案時，會根據在精靈中輸入的**名稱**，自動指派它們的檔名（比方說，如果是建立專案，且在**名稱**欄位中輸入 **MyProject**，專案的檔名就是 `MyProject.ppf`)。應用程式和專案名稱可利用「內容」視圖來重新命名。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 完成「新建專案精靈」。

新增現有的專案

您可以將先前以 AppScan Source for Analysis 建立的 AppScan Source 專案（.ppf 檔），新增至 AppScan Source 應用程式。您也可以新增 Eclipse 專案檔 (.epf)、任何支援的建置整合工具（如 Ounce/Maven 或 Ounce/Ant）所建立的專案，或利用 Microsoft Visual C/C++（.vcproj 或 .dsp）、VB.NET (.vbproj) 或 C# (.csproj) 來建立的專案檔。

這份表格列出您可以利用 AppScan Source for Analysis 來開啟及掃描的專案檔類型：

表 *5.* 要開啟的專案檔類型

專案檔類型	副檔名
Microsoft Visual Studio（第 6 版）	.dsp
Microsoft Visual Studio C/C++	.vcproj
Microsoft Visual Studio C#	.csproj
Microsoft Visual Studio Visual Basic	.vbproj
AppScan Source 專案檔	.ppf
Eclipse 專案檔	.epf

如果要瞭解如何新增現有的專案，請參閱這些主題：

- 第 44 頁的『利用使用者介面動作來新增現有的專案』
- 第 44 頁的『利用拖放動作來新增現有的專案』

重要：如果您使用的 AppScan Source 專案在開發環境中（例如 IBM MobileFirst Platform 專案）有相依關係，請務必先在開發環境中建置專案，然後再匯入它。匯入專案之後，如果您修改其中的檔案，請務必先在開發環境中重建它，再於 AppScan Source 中掃描（如果不這樣做，AppScan Source 會忽略您對檔案所做的修改）。

註：當匯入現有的 .NET 專案時，您可以指定其他組合來進行掃描。請在專案之「內容」視圖的「其他組合」標籤中，新增這些組合。新增其他組合時，您可以將建置的 .NET 專案與未建置的組合（包括協力廠商組合），結合在單一掃描中。

註：您也可以將 WAR 和 EAR 檔拖放至「瀏覽器」視圖中來新增它們；不過，這些會新增為應用程式而非專案。如需相關資訊，請參閱第 37 頁的『利用拖放動作來新增現有的應用程式』。

利用使用者介面動作來新增現有的專案

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列動作之一：
   - 從主工作台功能表中，選取**檔案** > **新增專案** > **現有的專案**。
   - 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **現有的專案**。
3. 瀏覽至要新增到應用程式的專案檔。

   重要：如果您使用的 AppScan Source 專案在開發環境中（例如 IBM MobileFirst Platform 專案）有相依關係，請務必先在開發環境中建置專案，然後再匯入它。匯入專案之後，如果您修改其中的檔案，請務必先在開發環境中重建它，再於 AppScan Source 中掃描（如果不這樣做，AppScan Source 會忽略您對檔案所做的修改）。

利用拖放動作來新增現有的專案

程序

1. 在工作站上，找出要新增來進行掃描的專案（`.ppf`、`.vcproj`、`.dsp`、`.vbproj` 或 `.csproj`）。

   註：您不能拖放由任何支援的建置整合工具（例如：Ounce/Maven 或 Ounce/Ant）所建立的檔案。
2. 選取專案，然後將它拖曳到 AppScan Source for Analysis 的「瀏覽器」視圖。
3. 完成下列步驟之一：
   a. 將選項放在現有的應用程式中。
   b. 將選項放在**所有應用程式**節點上，或放在它下面。由於專案必須包含在應用程式中，但這個動作並不會將專案新增到現有的應用程式中，因此，「新建應用程式」精靈會提示您為專案建立一個新的應用程式。請輸入應用程式的**名稱**，然後瀏覽至用來儲存應用程式的**工作目錄**。請按一下**完成**來建立新的應用程式（「瀏覽器」視圖中會包含新增的專案）。

   重要：如果您使用的 AppScan Source 專案在開發環境中（例如 IBM MobileFirst Platform 專案）有相依關係，請務必先在開發環境中建置專案，然後再匯入它。匯入專案之後，如果您修改其中的檔案，請務必先在開發環境中重建它，再於 AppScan Source 中掃描（如果不這樣做，AppScan Source 會忽略您對檔案所做的修改）。

新增多個專案

當新增多個專案到應用程式中，您可以將它們拖放到「瀏覽器」視圖中；您也可以瀏覽目錄來尋找專案，將部分或全部專案匯入現行應用程式中。

如果要瞭解如何新增多個專案，請參閱這些主題：

- 第 45 頁的『利用使用者介面動作來新增多個專案』
- 第 45 頁的『利用拖放動作來新增多個專案』

重要：如果您使用的 AppScan Source 專案在開發環境中（例如 IBM MobileFirst Platform 專案）有相依關係，請務必先在開發環境中建置專案，然後再匯入它。匯入專案之後，如果您修改其中的檔案，請務必先在開發環境中重建它，再於 AppScan Source 中掃描（如果不這樣做，AppScan Source 會忽略您對檔案所做的修改）。

利用使用者介面動作來新增多個專案

您可以從目錄（包含子目錄）、Eclipse 或 Rational Application Developer for WebSphere 軟體 (RAD) 工作區中，或從 Microsoft 解決方案檔中，將多個專案新增至一個應用程式。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列動作之一：
   - 從主工作台功能表中，選取**檔案** > **新增專案** > **多個專案**。
   - 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **多個專案**。
3. 在「新增多個專案」對話框中，完成下列其中一個動作：
   - 選取**從目錄匯入**，然後瀏覽至含有您要新增之專案的根目錄。選取**遞迴進入子目錄**勾選框，以便在子目錄中搜尋。
   - 選取**從 Eclipse 型工作區匯入**。選取**工作區類型**，然後瀏覽至工作區。選取工作區目錄，然後按一下**確定**。
   - 選取**從 Microsoft 解決方案檔案匯入**。瀏覽至檔案，並且選取它，然後按一下**確定**。
4. 完成下列動作之一：
   - 按一下**完成**，將專案新增至應用程式。
   - 按**下一步**，檢視搜尋結果，並且選取要新增的專案。然後按一下**完成**。

   重要：如果您使用的 AppScan Source 專案在開發環境中（例如 IBM MobileFirst Platform 專案）有相依關係，請務必先在開發環境中建置專案，然後再匯入它。匯入專案之後，如果您修改其中的檔案，請務必先在開發環境中重建它，再於 AppScan Source 中掃描（如果不這樣做，AppScan Source 會忽略您對檔案所做的修改）。

利用拖放動作來新增多個專案

程序

1. 在工作站上，找出要新增來進行掃描的專案（`.ppf`、`.vcproj`、`.dsp`、`.vbproj` 或 `.csproj`）。

   註：您不能拖放由任何支援的建置整合工具（例如：Ounce/Maven 或 Ounce/Ant）所建立的檔案。
2. 選取一或多個專案，然後將它們拖曳到「瀏覽器」視圖中。
3. 將選項放在現有的應用程式中。

註：您也可以將選項放在**所有應用程式**節點上，或放在它下面，不過，不建議這麼做。相反地，建議您將多個專案放在現有的應用程式中，如果需要新應用程式，也可以個別放置專案。

由於專案必須包含在應用程式中，但是將專案放在**所有應用程式**節點上，或放在它下面，並不會將專案新增到現有的應用程式中，因此，針對每個要新增到視圖的專案，「新建應用程式」精靈會提示您建立一個新的應用程式。

如果要新增多個專案到尚未存在的新應用程式中，請先建立一個應用程式，然後將所選的專案拖放到其中。

重要：如果您使用的 AppScan Source 專案在開發環境中（例如 IBM MobileFirst Platform 專案）有相依關係，請務必先在開發環境中建置專案，然後再匯入它。匯入專案之後，如果您修改其中的檔案，請務必先在開發環境中重建它，再於 AppScan Source 中掃描（如果不這樣做，AppScan Source 會忽略您對檔案所做的修改）。

新增 Arxan 專案

「專案配置」精靈可協助您手動建立 Arxan 專案，並將它新增到應用程式中。

關於這項作業

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **Arxan Android** 或 **Arxan iOS** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (`.ppf`) 的位置。它也是所有相對路徑的基礎。
   b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。
5. 按一下**完成**。

新增 ASP 專案

「專案配置」精靈可協助您手動建立 ASP 專案，將它新增到應用程式中。

關於這項作業

註：僅支援在 Windows 上使用此專案類型。

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。不過，精靈中有些頁面是選用的（當**完成**按鈕啟動時，會完成必要設定）。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。當您完成「新建專案精靈」時，如果沒有完成其中的選用頁面，之後可以在「內容」視圖中，從那些頁面來變更設定。

註：如果是 PHP、VB6 和 Classic ASP，僅支援 ISO-8859-1（西歐）、UTF-8 和 UTF-16 字集。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **ASP** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (.ppf) 的位置。它也是所有相對路徑的基礎。
   b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。
5. 按**下一步**，進入精靈的下一頁。
6. 在「ASP 專案配置」頁面中，執行下列動作：
   a. 識別 ASP 內容根目錄和預設語言，來配置 ASP 專案：

      ASP 內容根目錄：對應於主要 Web 或網域 URL 的目錄

      預設語言：VB Script（預設值）或 JavaScript
   b. 新增、刪除或移動 ASP 專案相依以進行編譯的類型庫（dll、exe、ocx 或 tlb）。
7. 按一下**完成**。

新增 C/C++ 專案

關於這項作業

當新增 C/C++ 專案到應用程式時，您可以指定要掃描的原始檔集合：

- include 路徑
- 前置處理器定義
- 選項

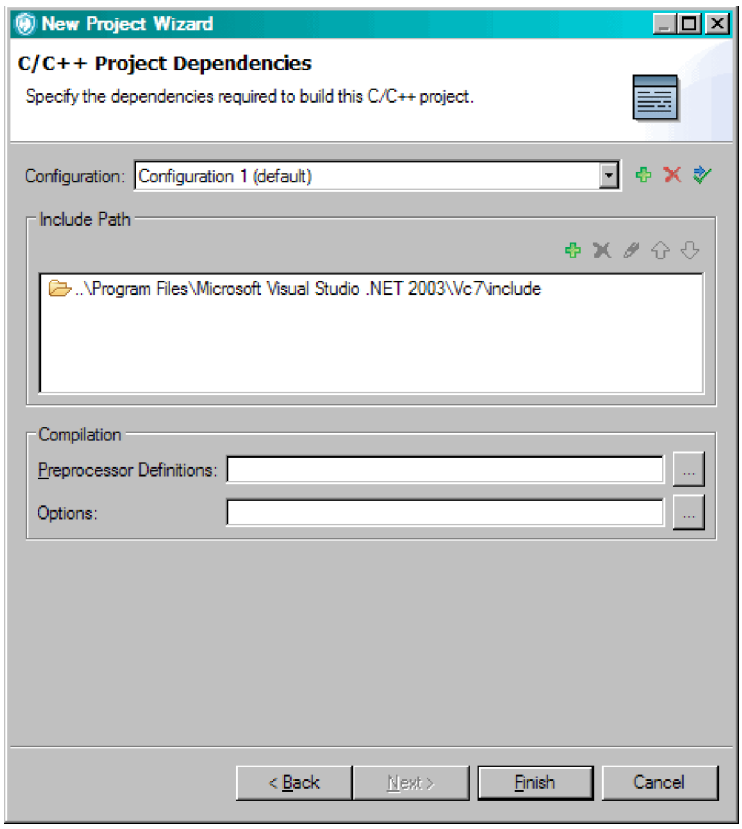
這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。不過，精靈中有些頁面是選用的（當**完成**按鈕啟動時，會完成必要設定）。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。當您完成「新建專案精靈」時，如果沒有完成其中的選用頁面，之後可以在「內容」視圖中，從那些頁面來變更設定。

重要：如果要掃描 C++ 專案，專案必須編譯及鏈結無誤。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **C/C++** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (.ppf) 的位置。它也是所有相對路徑的基礎。
   b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。
5. 按**下一步**，進入精靈的下一頁。
6. 在「C/C++ 專案相依關係」頁面中，指定專案配置和 include 路徑，來新增專案相依關係。



- **配置**：列出專案的所有可用配置。新增新的配置，或刪除現有的配置。請定義各個配置的所有其餘設定。

  您可以定義 C/C++ 專案的多重配置，例如，除錯和版本。 配置 1 是預設的專案配置名稱。
- **併入路徑**：請利用這個區段，來新增完整的路徑名稱到含有專案需要之 `#include` 檔的目錄中。
- **前置處理器定義**：請利用這個欄位，來新增定義給專案的前置處理符號。前置處理器定義專用於 C/C++ 程式碼。當您指定前置處理器定義時，請勿併入編譯器的 `-D` 選項（例如：請指定 `a=definition1` 而非 `-Da=definition1`）。當您指定多個定義時，請使用以分號區隔的清單。
- **選項**：專案配置所需要的其他編譯器參數。

7. 按一下**完成**。

新增 COBOL 專案

「專案配置」精靈可協助您手動建立 COBOL 專案，並將它新增到應用程式中。

關於這項作業

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **COBOL** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (.ppf) 的位置。它也是所有相對路徑的基礎。
   b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。
5. 按一下**完成**。

新增 ColdFusion 專案

「專案配置」精靈可協助您手動建立 ColdFusion 專案，將它新增到應用程式中。

關於這項作業

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **ColdFusion** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (.ppf) 的位置。它也是所有相對路徑的基礎。

b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。

5. 按一下**完成**。

新增 Java 或 JavaServer Pages (JSP) 專案

當新增 Java 專案到應用程式中，您先指定專案名稱，瀏覽至工作目錄，然後指定原始碼根目錄和專案相依關係。

關於這項作業

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。不過，精靈中有些頁面是選用的（當**完成**按鈕啟動時，會完成必要設定）。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。當您完成「新建專案精靈」時，如果沒有完成其中的選用頁面，之後可以在「內容」視圖中，從那些頁面來變更設定。

程序

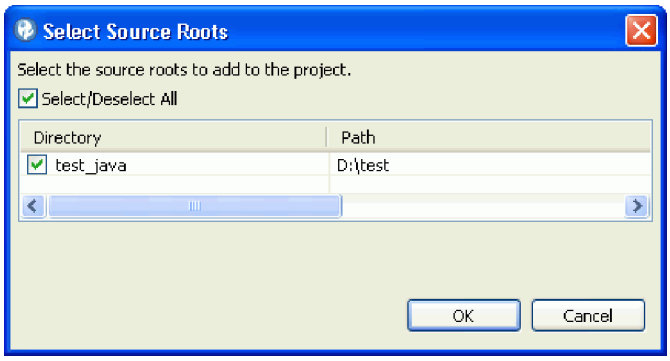
1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **Java/JSP** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源，專案來源由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要併入專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是 AppScan Source 專案檔 (`.ppf`) 的位置，且是所有相對路徑的基本目錄。
   b. 手動新增原始碼根目錄，或讓 AppScan Source for Analysis 自動尋找所有有效的原始碼根目錄。

      重要：

      - 如果要分析 Java 類別檔，您必須透過使用 `-g` 選項的 `javac` 來編譯它們。AppScan Source 分析需仰賴這個選項所產生的除錯資訊。
      - 如果專案中有包含國家語言字元的 Java 原始檔，而且您所執行的語言環境並不是原生語言環境（如 UTF-8），則掃描會失敗，主控台會出現錯誤及/或警告。
      - 如果要自動尋找原始碼根目錄，請執行下列動作：
        1) 按一下**尋找原始碼根目錄**，瀏覽至原始碼的根目錄。

2) 從所找到的所有原始碼根目錄的清單中，選取要新增到專案中的原始碼根目錄。

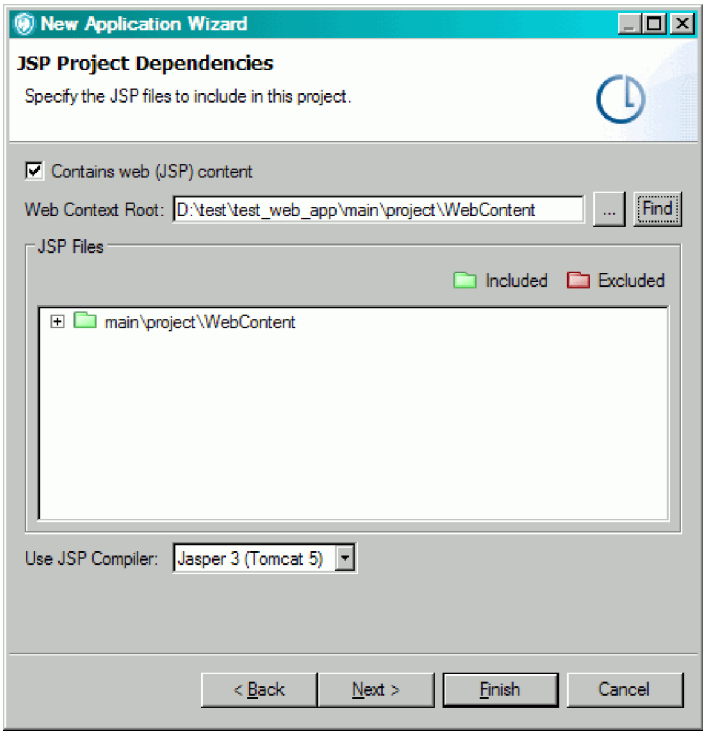


3) 按一下**確定**。要併入掃描的來源會出現在**專案來源**對話框中。

- 如果要手動尋找原始碼根目錄，請執行下列動作：
  1) 按一下**新增原始碼根目錄**。
  2) 選取原始碼根目錄或檔案。
  3) 按一下**確定**。新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。

按一下**完成**來新增專案，不設定專案相依關係；或按**下一步**來識別專案相依關係。

5. 在「JSP 專案相依關係」頁面中，執行下列動作：
   a. 識別 JavaServer Pages (JSP) 專案相依關係：如果是包含 JavaServer Pages 的 Java 專案，則識別 JSP 專案相依關係。如果專案是包含 JavaServer Pages 的 Web 應用程式，請選取**包含 Web (JSP) 內容**勾選框。



b. 手動選取 **Web 環境定義根目錄**，或按一下**尋找**來尋找它。 **Web 環境定義根目錄**是一個 WAR 檔，或含有 WEB-INF 目錄的目錄。Web 環境定義根目錄必須是有效 Web 應用程式的根目錄。

c. 選取專案的 **JSP 編譯器**。既有的 Tomcat 7 是預設 JSP 編譯器設定（您可以在 Java 和 JSP 喜好設定頁面中變更預設 JSP 編譯器）。如果要瞭解 AppScan Source 支援的編譯器，請參閱 http://www.ibm.com/support/docview.wss?uid=swg27027486。

AppScan Source 的安裝架構包含 Apache Tomcat 第 7 版和第 8 版。如果未配置 **Tomcat 7** 和 **Tomcat 8** 喜好設定頁面，AppScan Source 會利用提供的 Tomcat JSP 編譯器（目前標示為預設編譯器）來編譯 JSP 檔。如果您想要使用外部支援的 Tomcat 編譯器，請利用 Tomcat喜好設定頁面來指向您的本端 Tomcat 安裝架構。

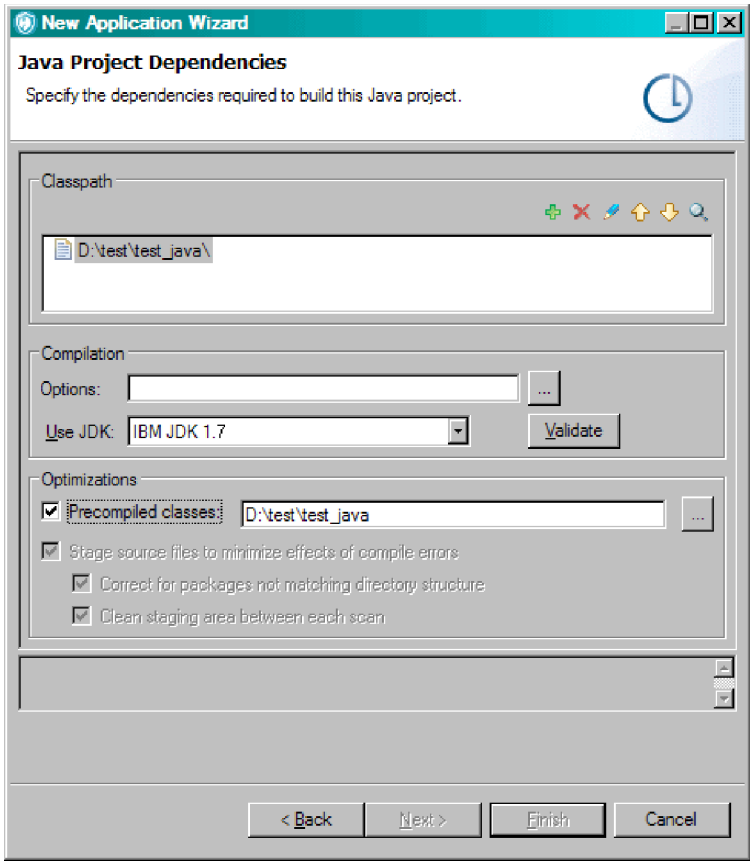
如果您使用 Oracle WebLogic 伺服器 或 WebSphere Application Server，您必須配置適用的喜好設定頁面來指向應用程式伺服器的本端安裝架構，以便在分析期間用來編譯 JSP。如果您尚未完成此配置，當您選取 JSP 編譯器時，會出現訊息來提示您這麼做。如果您在訊息中按一下**是**，就會看到適當的喜好設定頁面。如果您按一下**否**，JSP 編譯器選項旁邊會顯示警告鏈結（遵循此鏈結會開啟喜好設定頁面）。

按一下**完成**來新增含有 JSP 專案相依關係的專案；或按**下一步**來識別 Java 專案相依關係。

6. 在「Java 專案相依關係」頁面中，識別建置這個 Java 專案所需要的相依關係：

a. 手動新增 JAR 檔，或按一下**尋找**，讓 AppScan Source for Analysis 搜尋含有相依的 JAR 和類別檔的目錄。

類別路徑清單會顯示專案的相對路徑。類別路徑必須指定必要的 JAR 檔，以及專案需要的類別檔所在的目錄。



- **新增**、**移除**、**上移**和**下移**：在類別路徑中新增或移除檔案，或按次序將它們上移或下移。
- **尋找**：根據專案中的原始檔來尋找 JAR 和類別路徑項目。

重要：如果 Java 專案包含 JavaServer Pages，您也必須新增「JSP 專案相依關係」。

- 如果要手動尋找專案相依關係，請執行下列動作：
  1) 在「類別路徑」區段工具列中按一下**新增**，然後選取編譯 Java 專案所需要的 JAR 和類別檔目錄。
  2) 按一下**確定**。JAR 檔和目錄會出現在類別路徑中。請依照需要來變更順序。
- 如果要自動尋找相依關係，請執行下列動作：
  1) 按一下「類別路徑」區段工具列中的**尋找**。
  2) 指定目錄，以便在其中尋找編譯 Java 專案所需要的 JAR 和類別檔。
  3) 如果您想讓 AppScan Source for Analysis 根據來源並利用提供的搜尋路徑，來尋找必要的專案相依關係，請選取**在來源和 JAR 檔內查看**勾選框。
  4) 按**下一步**，尋找專案相依關係，並識別衝突。
- 如果要解決衝突，請執行下列動作：

1) 如果衝突存在，請在「解決衝突」對話框中，選取要解決的項目，然後按一下**解決**（或按**下一步**，自動解決衝突）。當 AppScan Source for Analysis 在滿足相依關係的目錄中，找到多個 JAR 或類別時，就會出現衝突。

   未解決的衝突左側會出現紅色圖示。解決之後，紅色圖示會變成綠色，項目會成為**已解決**。您也可以**移除**衝突。

2) 在解決或移除衝突之後，您可能會想驗證、重新排序或移除類別路徑項目。請記下找不到的匯入項目清單。任何無法分辨的匯入，都會在 AppScan Source for Analysis 掃描時導致編譯錯誤。

b. **選項**：指定專案所需要的任何其他編譯器參數。

   編譯選項是指傳遞給編譯器以便能編譯原始檔的選項。例如，-source 1.5 指定專案的來源層次。

c. **使用 JDK**：指定當掃描這個程式碼時，所使用的 Java Development Kit (JDK)。依預設，會使用 **IBM JDK 1.8**。AppScan Source 也提供了 **IBM JDK 1.7** 供選擇。如果要定義其他 JDK，或設定不同的預設 JDK，請使用 **Java 與 JSP 喜好設定**。

   註：JSP 專案既有的預設編譯器是 Tomcat 7，它需要 Java 1.6 版或更高版本。如果保留 **Tomcat 7** 作為預設值，則使用較舊的 JDK 會導致在掃描期間發生編譯錯誤。

d. **驗證**動作可確保已正確配置專案相依關係。它會檢查 Java 專案在來源和類別路徑之間的配置衝突，它也會檢查編譯錯誤。如果在原始碼根目錄中，類別路徑中的類別重複，就會發生衝突。

   如果有衝突，驗證文字區會顯示類別路徑上定義類別的 JAR 或位置，以及來源中是否有重複。請從類別路徑中移除衝突，然後重新執行檢查。

   檢查衝突之後，**驗證**會判斷專案是否進行編譯，且會報告任何編譯錯誤。

e. **經過前置編譯的類別**：這個欄位可讓您使用經過前置編譯的 Java 或 JSP 類別檔，而不在掃描期間編譯。

f. **暫置原始檔，使編譯錯誤的影響降到最低**：如果原始碼編譯正確，且在目錄中準確排列，與符合套件，請清除這個勾選框。

g. **修正不符合目錄結構的套件**：如果套件不符合目錄結構，請選取這個選項。

h. **清除各次掃描之間的暫置區**：最佳化選項。

7. 按一下**完成**。

結果

提示：如果您要掃描 Java 但 Java 專案中有遺漏的相依關係，AppScan Source 會綜合相依關係可能已提供的片段，來建立追蹤資料。此綜合作業未必能精確反映 .jar 檔中的資訊。如果要限制綜合作業，藉此改善發現項目的正確性，在此情況下，您可以依下列方式指定遺漏的相依關係：

1. 掃描之後，開啟 <data_dir>\logs\scanner_exceptions.log（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）查看 AppScan Source 是否有所報告的遺漏相依關係。

2. 修改專案內容來併入相依關係。要執行這項作業，請遵循第 71 頁的『修改應用程式和專案內容』的指示，然後在 **JSP 專案相依關係**或**專案相依關係**標籤中指定並儲存相依關係。
3. 重新掃描專案。

註：依預設，AppScan Source 會掃描含有遺漏的相依關係或編譯錯誤的 Java 檔和 Java 位元組碼。這些設定可以依下列方式加以變更：

1. 在文字編輯器中開啟 <data_dir>\config\scan.ozsettings。
2. 如果要變更編譯錯誤設定，請在檔案中找出 compile_java_sources_with_errors。這項設定看起來如下：

```
<Setting
  name="compile_java_sources_with_errors"
  value="true"
  default_value="true"
  type="bool"
  hidden="true"
  display_name="compile_java_sources_with_errors"
  description="Attempt to scan java code with compilation errors."
/>
```

3. 如果要變更遺漏的相依關係設定，請在檔案中找出 scan_java_bytecode_without_dependencies。這項設定看起來如下：

```
<Setting
  name="scan_java_bytecode_without_dependencies"
  value="true"
  default_value="true"
  type="bool"
  hidden="true"
  display_name="scan_java_bytecode_without_dependencies"
  description="Scans Java bytecode even when some of
    the dependencies are missing by artificially
    synthesizing the unresolved symbols."
/>
```

4. 在設定中，修改 value 屬性。如果該屬性設為 true，這項設定即會開啟。如果編譯錯誤設定設為 false，AppScan Source 在掃描期間會略過含有編譯錯誤的 Java 程式碼。如果遺漏的相依關係設定設為 false，當有遺漏的相依關係時，AppScan Source 就不會掃描 Java 位元組碼。
5. 修改好這項設定之後儲存檔案，再啟動或重新啟動 AppScan Source。

新增內容至 JSP 專案

JavaServer Pages (JSP) 專案包括在 JavaServer Pages 上建置的 Web 應用程式。

關於這項作業

如果要順利掃描 JSP 專案，JavaServer Pages 必須在有效的 Web 應用程式結構中。本節說明如果要順利進行掃描，在 Web 環境定義根目錄之下，所需具備的檔案結構。在配置 JSP 專案之前，您應該先熟悉 Web 應用程式結構。

部署在 Web 應用程式伺服器（如 Tomcat）的 Web 應用程式，需要標準目錄結構。已部署的應用程式可以是一組排列在目錄結構中的檔案，或是一個 WAR 檔。如果是 WAR 檔，目錄結構包含在 ZIP 檔中，以 Web 環境定義根目錄作為目錄結構的根。

在 Web 環境定義根目錄之下，您可以找到下列標準目錄：

表 *6. Web* 環境定義根目錄

<web-context-root>\	
WEB-INF\	
classes\	排列在目錄（套件）中的 Java 類別檔
lib\	新增到類別路徑中的 JAR 檔
web.xml	web.xml 說明應用程式所能使用的資源

其他目錄包含也可能存在的必要檔案。例如，您通常會見到一個內容（JSP 和 HTML 檔）以及標籤庫的目錄：

表 *7.* 其他目錄

<web-context-root>\	
jsp\	包含應用程式中的 JavaServer Pages
WEB-INF\	
tld\	包含應用程式所用的標籤庫

除了這些標準的 Web 應用程式目錄之外，Web 應用程式伺服器也可能有一些特殊目錄，所有已部署的 Web 應用程式所共用的類別檔和 JAR 檔，都預期出現在這些目錄中。例如，Tomcat 7 將這些 JAR 檔放在 common\lib 或 common\endorsed 目錄中。這些非標準目錄的位置是各個應用程式伺服器所專用。

重要：在掃描 JavaServer Pages 之前，請確認所有必要的檔案都在 Web 環境定義根目錄中。 AppScan Source for Analysis 只會掃描 Web 環境定義根目錄中的 JavaServer Pages。

程序

1. 必要的話，將檔案複製到 Web 環境定義根目錄下的適當位置。
2. 將 Web 環境定義根目錄指定為含有所有 JavaServer Pages 的目錄或 WAR 檔。
3. 確定類別路徑包含 JAR 或類別檔目錄。
4. 配置專案內容。

結果

AppScan Source for Analysis 僅會針對 JSP，將 WEB-INF\classes 目錄及 WEB-INF\lib 中的所有 JAR 檔新增到類別路徑中。您可以新增編譯 JSP 時，必須用到但卻不在 Web-INF 路徑中的項目。這些 JAR 檔類似於在應用程式伺服器一般目錄中的 weblogic.jar 或供應商 JAR 檔。

JSP 來源是在您想要掃描的 Web 環境定義根目錄之下的 JavaServer Pages。原始檔相對於 Web 環境定義根目錄。當指定 JSP 來源時，您會受限於一組在 Web 環境定義根目錄中的檔案。

JSP 專案來源由若干目錄組成，您在這些目錄中，尋找專案檔及任何要併入專案中的個別檔案。

- 指定 Web 環境定義根目錄中的 JavaServer Pages 子集。如果不這樣做，則會掃描所有檔案。

- 如果 JavaServer Pages 相依於 Java 程式碼，您必須指定這些來源。
- JSP 檔包括 `jsp` 和 `jspx` 檔。

新增 JavaScript 專案

「專案配置精靈」可以協助您手動建立 JavaScript 專案，以及將它新增到應用程式中。

關於這項作業

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **JavaScript** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (`.ppf`) 的位置。它也是所有相對路徑的基礎。
   b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。
5. 按一下**完成**。

新增 .NET 組譯碼專案

「新建專案精靈」可協助您建立「.NET 組譯碼」專案。當原始檔不可用或無法建置時，「.NET 組譯碼」專案可用來掃描編譯的 .NET 組譯碼檔。 「.NET 組譯碼」專案由工作目錄和來源清單組成，這些來源可能是目錄或個別組合檔。

關於這項作業

註：僅支援在 Windows 上使用此專案類型。

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **.NET 組譯碼**作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (.ppf) 的位置。它也是所有相對路徑的基礎。
   b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。
5. 按一下**完成**。

新增基於型樣的專案

關於這項作業

「新建專案精靈」可協助您手動建立「基於型樣」專案，並將它新增到應用程式中。「基於型樣」專案包含一個不關聯於特定語言的任何檔案的集合，可用來進行基於型樣的分析和掃描。

例如，您可能會想將 `.xml` 和 `.config` 檔以邏輯方式加以分組，然後搜尋它們來尋找某些基於型樣的表示式。AppScan Source for Analysis 會掃描檔案，並搜尋表示式（如需詳細資料，請參閱第 209 頁的『利用基於型樣的規則自訂』）。

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取**基於型樣**作為專案類型，然後按**下一步**，進入精靈的下一頁。

4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (.ppf) 的位置。它也是所有相對路徑的基礎。
   b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。
5. 按一下**完成**。

新增 Perl 專案

「新建專案精靈」可協助您手動建立 Perl 專案，將它新增到應用程式中。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **Perl** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (.ppf) 的位置。它也是所有相對路徑的基礎。
   b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。
5. 按一下**完成**。

PHP 專案配置

當新增「PHP：超文字前置處理器 (PHP)」專案到應用程式中，您先指定專案名稱，瀏覽至工作目錄，然後指定原始碼根目錄和專案相依關係。您也可以在建立好專案之後，在專案內容的「專案相依關係」標籤中，設定專案相依關係。

關於這項作業

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。不過，精靈中有些頁面是選用的（當**完成**按鈕啟動時，會完成必要設定）。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。當您完成「新建專案精靈」時，如果沒有完成其中的選用頁面，之後可以在「內容」視圖中，從那些頁面來變更設定。

註：如果是 PHP、VB6 和 Classic ASP，僅支援 ISO-8859-1（西歐）、UTF-8 和 UTF-16 字集。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **PHP** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (`.ppf`) 的位置。它也是所有相對路徑的基礎。
   b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。
5. **PHP 專案配置**：在 **PHP 文件根目錄**欄位中，輸入或瀏覽至 PHP 應用程式的根目錄。這是對映至網站基本 URL 的檔案系統目錄。如果未指定 PHP 文件根目錄，就會使用「專案來源」頁面中所指定的原始碼根目錄。
6. 選擇性的： 設定**併入路徑**。併入路徑目錄用來解析 PHP `include` 陳述式（如 `include`、`include_once`、`require`、`require_once`）中，所用檔案的相對路徑。
7. 選擇性的： 設定**類別併入路徑**。類別併入路徑目錄用來尋找包含 PHP 類別定義的檔案。
8. 按一下**完成**。
9. 選擇性的： **配置尚未解析的相依關係**：在專案內容中，移至「專案相依關係」頁面，然後遵循第 62 頁的『配置尚未解析的 PHP `include` 表示式』和第 66 頁的『配置尚未解析的 PHP 類別參照』的步驟。

範例：建立新的 PHP 專案

關於這項作業

這個範例告訴您如何利用「新建應用程式」精靈來建立 PHP 專案。

程序

1. 完成下列動作之一：
   - 從主功能表列中，選取**檔案** > **新增應用程式** > **建立新的應用程式**。
   - 在「瀏覽器」視圖工具列中，按一下**新增應用程式**功能表向下箭頭按鈕，然後從功能表中選取**建立新的應用程式**。
   - 在「瀏覽器」視圖中，用滑鼠右鍵按一下**所有應用程式**，然後從功能表中選取**新增應用程式** > **建立新的應用程式**。
2. 輸入應用程式的**名稱**。
3. 瀏覽至用來儲存應用程式的**工作目錄**。新應用程式檔案的副檔名是 `.paf`。
4. 按**下一步**來配置專案。
5. 在精靈的「選取專案類型」頁面中，選取 **PHP** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
6. 在「專案來源」頁面中，執行下列動作：
   a. 在**名稱**欄位中，輸入專案的名稱，例如 `MyProject`。
   b. 在**工作目錄**欄位中，瀏覽至要用來儲存即將建立之專案檔的位置，例如 `C:\Apps\MyProject`。
   c. 按一下**新增原始碼根目錄**，來新增應掃描之 PHP 檔所在的所有目錄。例如，在「選取檔案或目錄」對話框中，瀏覽至 `C:\Apps\MyProject\root`，然後按一下**確定**來關閉對話框。

   按**下一步**。
7. 在「PHP 專案配置」頁面中，執行下列動作：
   a. 在 **PHP 文件根目錄**欄位中，輸入或瀏覽找出 PHP 應用程式的根目錄。這是對映至網站基本 URL 的檔案系統目錄。依預設，這個欄位會預先移入「專案來源」頁面中所指定的原始碼根目錄。
   b. 選擇性的： 新增併入路徑目錄。這些用來解析 PHP Include 陳述式（如 `include`、`include_once`、`require`、`require_once`）中，所用檔案的相對路徑。
   c. 選擇性的： 新增類別路徑目錄。這些用來尋找包含 PHP 類別定義的檔案。
8. 按一下**完成**。現在，您有一個已備妥，可以開始掃描的 PHP 專案。

配置尚未解析的 PHP *include* 表示式

開始之前

在專案內容中，移至「專案相依關係」頁面。

程序

1. 按一下**配置尚未解析的 include 表示式**，來開啟「配置尚未解析的 include 表示式」對話框。
2. 對話框上半部列出尚未解析的 `include` 表示式（所有尚未解析或需要額外處理才能解析的 `include` 表示式）。提供的表示式相關資訊包括：

選項	敘述
併入的文字/更新的文字	這個直欄依照原始碼中的現狀來顯示表示式文字。您可以按一下 **+** 來展開這個直欄，接著會顯示前次掃描期間所用的更新文字。如果前次掃描期間沒有可用的更新文字，就會顯示 <empty>。展開之後，可能會有多行更新文字，原始碼中每個用到這個表示式的位置各一個。
狀態	在這個直欄中，尚未解析的表示式會有一個 X，已順利解析的表示式會有一個勾號。
解析者	這個直欄指出更新文字的產生方式。值包括： • AutoResolver：應用程式利用內部探索來尋找檔案。 • SearchReplace：在併入文字上套用一或多個搜尋和取代規則，以產生更新的文字。 • SearchReplace+AutoResolver：在併入文字上套用一或多個搜尋和取代規則，以產生更新的文字，然後套用內部探索來尋找檔案。 • SearchReplace+IncludePath：在併入文字上套用一或多個搜尋和取代規則，以產生更新的文字，然後結合 include 路徑上的目錄來尋找檔案。 • SearchReplace+RelativeDir：在併入文字上套用一或多個搜尋和取代規則，以產生更新的文字，然後發現是相對於包括 include 表示式之檔案的來源目錄。
原始檔、行、直欄	這些直欄顯示原始碼中使用表示式的位置。您可以在編輯器中查看這些位置，以瞭解應該如何解析它們。

註：有些直欄可能空白。這是因為展開之後，併入的文字可能有多行更新文字。在這些直欄中，這幾行更新的文字，每行都會有適當的文字。

3. 在對話框的下半部，「併入路徑」標籤包含「PHP 專案相依關係」頁面中所輸入的相同 include 路徑資訊。在這個對話框中，您可以更新這項資訊（在檢視尚未解析的 include 表示式時）。
4. 在對話框的下半部，「搜尋和取代」標籤用來新增規則，用以將本身是 include 檔之完整或局部檔案路徑的靜態文字，取代 include 表示式中的動態文字。在「搜尋和取代」標籤中，有三個直欄：

選項	敘述
指令	這個直欄的值決定了**搜尋文字**和**取代文字**直欄的使用方式。選項如下： • **取代文字**：這個指令用來進行簡式文字搜尋和取代。**搜尋文字**是依現狀使用，如果在併入文字中的任何位置找到它，則會將它取代為**取代文字**。 • **取代函數**：當要取代函數呼叫時，使用這個指令。**搜尋文字**應該是不含括弧的函數名稱。搜尋文字會進一步加強，以指定的名稱後面接著括弧來尋找函數呼叫，且會符合括弧的任何內容。 • **取代 Regex**：這是一個進階特性，可供針對搜尋文字指定正規表示式。
搜尋文字	這是在 include 表示式中所要搜尋的文字。您可以選取 include 表示式內的文字，將它複製到剪貼簿，再貼到這裡。請參閱上述**指令**直欄的說明，以瞭解指定搜尋文字的變異。
取代文字	這是用來取代搜尋文字的文字。這是一個本身是 include 檔的完整或局部檔案路徑的靜態文字。在取代文字中，也可以放置一些變數。您可以直接在**取代文字**資料格中輸入它們，也可以從表格上方的**取代文字變數**功能表中選取（所選變數會複製到剪貼簿）。可從**取代文字變數**功能表清單中選取的變數如下： • %ROOT_DIR%：專案指定的 PHP 文件根目錄會取代這個變數。 • %SRC_DIR%：包含 include 表示式之檔案的目錄會取代這個變數。 • %ARG_N%：只有在指令是**取代函數**時，這個變數才適用。變數中的 N 應該取代為整數（例如，%ARG_1% 或 %ARG_2%）。之後，這個變數會取代為傳入函數呼叫第 N 個參數的文字。

這些規則會循序套用。每次搜尋和取代作業順利完成之後，都會檢查新的文字，看看能不能找到檔案。如果找不到檔案，則會在更新的文字上嘗試下一個規則。

每個 PHP 專案都開始於三個搜尋和取代規則，這些規則會嘗試取代 include 表示式中一些常用的標準 PHP 常數和函數。

範例：配置尚未解析的 *include* 表示式：
開始之前

尚未解析的 include 表示式配置在專案內容的「專案相依關係」標籤頁中。進入這個頁面之後，請按一下**配置尚未解析的 include 表示式**，來開啟「配置尚未解析的 include 表示式」對話框。

這個範例假設您已新增 include 路徑到專案中（您可以在建立專案時，或在「專案相依關係」頁面中執行這個動作），且已執行掃描。掃描完成之後，請開啟「配置尚未解析的 include 表示式」對話框來查看**尚未解析的 include 表示式**清單。在這個範例中，MYPROJECT_ROOT_PATH.'/a/b/filename.php'、MYPROJECT_ROOT_PATH.'/language/'.$configInfo['language'].'/mypage.php' 和 configGet('database_inc','./includes/database.inc') 是在這份清單中的表示式。

程序

1. 遵循下列步驟，以目錄來取代前導 PHP 常數或變數：
   a. 選取 MYPROJECT_ROOT_PATH.'/a/b/filename.php' - 這會選取表示式中的文字。之後，您可以利用滑鼠或游標鍵來選取表示式的一部分。請選取 MYPROJECT_ROOT_PATH，然後按一下滑鼠右鍵，選擇**複製**。
   b. 選取「搜尋和取代」標籤。
   c. 按一下**新增所選尚未解析之項目的規則**按鈕（裝飾了綠色**加號**符號）。這會使新的搜尋和取代規則新增到清單中。
   d. 在新規則中，選取 NewSearchText，然後按一下滑鼠右鍵，從功能表中選擇**貼上**。這會使 NewSearchText 取代為 MYPROJECT_ROOT_PATH。
   e. 從**取代文字變數**功能表中，選取 %ROOT_DIR%。這會使 %ROOT_DIR% 字串複製到剪貼簿中。
   f. 在規則中，選取 NewReplacementText，然後按一下滑鼠右鍵，從功能表中選擇**貼上**。這會使 NewReplacementText 取代為 %ROOT_DIR%。

   現在，您有一個新規則會以 PHP 文件根目錄的路徑來取代常數。 PHP 連結運算子 (.) 和它後面的字串即將結合取代文字，來產生單一路徑表示式。下次掃描專案時，使用這個常數的 include 表示式應該會成功。
2. 如果要以單一值來取代動態表示式，請執行下列動作：
   a. 請選取 MYPROJECT_ROOT_PATH.'/language/'.$configInfo['language'].'/mypage.php' - 這樣可以選取表示式中的文字。之後，您可以利用滑鼠或游標鍵來選取表示式的一部分。請選取 $configInfo['language']，然後按一下滑鼠右鍵，選擇**複製**。
   b. 選取「搜尋和取代」標籤。
   c. 按一下**新增所選尚未解析之項目的規則**按鈕（裝飾了綠色**加號**符號）。這會使新的搜尋和取代規則新增到清單中。
   d. 在新規則中，選取 NewSearchText，然後按一下滑鼠右鍵，從功能表中選擇**貼上**。這會使 NewSearchText 取代為 $configInfo['language']。
   e. 在規則中，選取 NewReplacementText，然後輸入 english，以新文字來取代它。

   現在，您有一個新規則會以指定的值來取代表示式。會套用 PHP 連結運算子 (.) 來產生單一路徑表示式。下次掃描專案時，使用這個表示式的 include 表示式應該會成功。
3. 如果要將 PHP 函數呼叫取代為它的引數之一，請執行下列動作：
   a. 選取 configGet('database_inc','./includes/database.inc') - 這會選取表示式中的文字。之後，您可以利用滑鼠或游標鍵來選取表示式的一部分。請選取 configGet，然後按一下滑鼠右鍵，選擇**複製**。
   b. 選取「搜尋和取代」標籤。
   c. 在新規則中，在第一個直欄選取取代文字，從功能表中，選取取代函數。

d. 在規則中，選取 NewSearchText，然後按一下滑鼠右鍵，從功能表中選擇**貼上**。這會使 NewSearchText 取代為 configGet。

e. 從**取代文字變數**功能表中，選取 %ARG_1%。這會使變數複製到剪貼簿中。

f. 在規則中，選取 NewReplacementText，然後按一下滑鼠右鍵，從功能表中選擇**貼上**。將貼上的文字編輯為 %ARG_2%，而不是 %ARG_1%。

現在，您有一個新規則會以第二個參數的值來取代函數呼叫。下次掃描專案時，使用這個函數呼叫的 include 表示式應該會成功。

配置尚未解析的 PHP 類別參照

開始之前

在專案內容中，移至「專案相依關係」頁面。

程序

1. 按一下**配置尚未解析的類別參照**，來開啟「配置尚未解析的類別參照」對話框。
2. 對話框上半部列出尚未解析的類別參照（前次掃描期間所未解析的所有類別參照）。提供的類別參照相關資訊包括：

選項	敘述
類別名稱/產生的檔案名稱	這個直欄顯示原始碼所參照的類別名稱。您可以按一下 **+** 來展開這個直欄，接著會顯示前次掃描期間所用的產生的檔案名稱，以嘗試及尋找類別。展開之後，可能會有多個檔案名稱，原始碼中每個用到這個類別的位置各一個。
狀態	在這個直欄中，尚未解析的類別會有一個 X，已順利解析的類別會有一個勾號。
解析者	這個直欄指出產生的檔名的建立方式。值包括： • AutoResolver：應用程式利用內部探索來尋找檔案。 • SearchReplace：在類別名稱上套用一或多個搜尋和取代規則，以建立產生的檔名。
原始檔、行、直欄	這些直欄顯示原始碼中使用類別的位置。您可以在編輯器中查看這些位置，以瞭解應該如何解析它們。

註：有些直欄可能空白。這是因為展開之時，類別名稱可能有多行產生的檔名。在這些直欄中，這幾行產生的檔名，每行都會有適當的文字。

3. 在對話框的下半部，「類別併入路徑」標籤包含「PHP 專案相依關係」頁面中所輸入的相同類別 include 路徑資訊。在這個對話框中，您可以更新這項資訊（在檢視尚未解析的類別參照之時）。
4. 在對話框的下半部，「搜尋和取代」標籤用來將尚未解析之類別名稱的修改規則，新增到包含該類別定義的完整或局部檔案路徑。在「搜尋和取代」標籤中，有三個直欄：

選項	敘述
指令	這個直欄的值決定了**搜尋文字**和**取代文字**直欄的使用方式。選項如下： • **符合文字**：這個指令用來搜尋和取代文字。搜尋文字接受 * 字元，它會符合 0 或多個字元。符合文字的結果不會影響任何其他搜尋和取代規則。這通常是在產生檔名時，用來嘗試新增類別名稱的副檔名，或濾除類別名稱的字首和字尾。 • **取代文字**：這個指令用來進行簡式文字搜尋和取代。**搜尋文字**是依現狀使用，如果在類別名稱中的任何位置找到它，會將它取代為**取代文字**。這用來修改類別名稱，以便用於下列規則。 • **取代 Regex**：這是一個進階特性，可供針對搜尋文字指定正規表示式。
搜尋文字	這是在類別參照中所要搜尋的文字。您可以選取類別名稱內的文字，將它複製到剪貼簿，再貼到這裡。請參閱上述**指令**直欄的說明，以瞭解指定搜尋文字的變異。
取代文字	這是用來取代搜尋文字的文字。這是一個靜態文字，是包含類別定義之檔案的完整或局部檔案路徑。在取代文字中，也可以放置一些變數。您可以直接在**取代文字**資料格中輸入它們，也可以從表格上方的**取代文字變數**功能表中選取（所選變數會複製到剪貼簿）。可從**取代文字變數**功能表清單中選取的變數如下： • %ROOT_DIR%：專案指定的 PHP 文件根目錄會取代這個變數。 • %SRC_DIR%：包含類別參照的檔案目錄會取代這個變數。 • %MATCH_N%：只有在指令是**符合文字**時，這個變數才適用。變數中的 N 應該取代為整數（例如，%MATCH_1% 或 %MATCH_2%）。之後，這個變數會取代為符合搜尋文字中第 N 個 * 的文字。

這些規則會循序套用。每次搜尋和取代作業順利完成之後，都會檢查新的文字，看看能不能找到檔案。如果找不到檔案，除非指令是**符合文字**，否則，會在更新的文字上嘗試下一個規則。

每個 PHP 專案都開始於兩個簡式搜尋/取代規則，它們會嘗試在類別名稱上，新增一些共用的副檔名。

5. 在對話框的下半部，「找到的類別」標籤會列出在前次掃描期間找到的所有類別。這可用來更新類別 include 路徑及搜尋和取代規則。您可以在對話框的「尚未解析的類別參照」區段中，選取尚未解析的類別參照，然後按一下**尋找宣告**。如果找到宣告，它會顯示在「找到的類別」標籤清單中。

範例：配置尚未解析的類別參照：
開始之前

尚未解析的類別參照配置在專案內容的「專案相依關係」標籤頁中。進入這個頁面之後，請按一下**配置尚未解析的類別參照**，來開啟「配置尚未解析的類別參照」對話框。

這個範例假設您已新增類別 include 路徑到專案中（您可以在建立專案時，或在「專案相依關係」頁面中執行這個動作），且已執行掃描。掃描完成之後，請開啟「配置尚未解析的類別參照」對話框來查看**尚未解析的類別參照**清單。

在這些範例中，您要提供取代文字值。請注意，它們在文字中可以有多個變數，例如，%ROOT_DIR%/modules/%MATCH_1%/classes/%MATCH_1%.class.inc。

程序

1. 如果要新增 include 檔所用的另一個副檔名，請執行下列動作：
   a. 選取「搜尋和取代」標籤。
   b. 按一下**新增所選尚未解析之項目的規則**按鈕（裝飾了綠色**加號**符號）。這會將新的搜尋和取代規則新增到清單中。
   c. 在新規則中，選取**取代文字**，%MATCH_1%.php。在這個字串中，刪除 .php，在它的位置輸入 .class.inc。現在，**取代文字**應該是 %MATCH_1%.class.inc。

   現在，您有一個新規則，當解析類別時，它會嘗試將 .class.inc 字尾新增到類別名稱中。
2. 如果要移除類別名稱的字首，請執行下列動作：
   a. 選取「搜尋和取代」標籤。
   b. 按一下**新增所選尚未解析之項目的規則**按鈕（裝飾了綠色**加號**符號）。這會將新的搜尋和取代規則新增到清單中。
   c. 在新規則中，選取**取代文字**直欄中的字串 (*)，在它的位置輸入 Abc*。
   d. 請勿變更 %MATCH_1%.php 取代文字。

   現在，您有一個新規則會將 AbcHello 之類的類別名稱對映到 Hello.php。
3. 如果要移除類別名稱的字尾，請執行下列動作：
   a. 選取「搜尋和取代」標籤。
   b. 按一下**新增所選尚未解析之項目的規則**按鈕（裝飾了綠色**加號**符號）。這會將新的搜尋和取代規則新增到清單中。
   c. 在新規則中，選取**取代文字**直欄中的字串 (*)，在它的位置輸入 *Xyz。
   d. 請勿變更 %MATCH_1%.php 取代文字。

   現在，您有一個新規則會將 ByeByeXyz 之類的類別名稱對映到 ByeBye.php。
4. 您可以對映 Abc_Def_Ghi_class 之類的類別名稱，以便利用它們的字首作為進入檔案系統的相對路徑（例如，Abc/Def/Ghi/class.php）。如果要修改類別名稱文字，以用於其他規則，請執行下列動作：
   a. 選取「搜尋和取代」標籤。
   b. 按一下**新增所選尚未解析之項目的規則**按鈕（裝飾了綠色**加號**符號）。這會將新的搜尋和取代規則新增到清單中。
   c. 在新規則中，在第一個直欄選取**符合文字**，然後從功能表中，選取**取代文字**。

d. 在規則中，選取**取代文字**直欄中的字串 (*)，在它的位置輸入 _。

e. 選取**取代文字**直欄中的字串，在它的位置輸入 /。

f. 選取規則時，請按一下**上移**，將這個規則移到清單頂端。

現在，您有一個新規則會以斜線 (/) 來取代底線 (_)，後續的所有規則都會使用更新的文字。這個規則會將 Abc_Def_Ghi_class 改成 Abc/Def/Ghi/class，然後其餘的**符合文字**規則會嘗試新增 .php 和 .inc 之類的副檔名。

新增 PL/SQL 專案

「新建專案精靈」可協助您手動建立 PL/SQL 專案，將它新增到應用程式中。

關於這項作業

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **PL/SQL** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (.ppf) 的位置。它也是所有相對路徑的基礎。
   b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。
5. 按一下**完成**。

新增 T-SQL 專案

「新建專案精靈」可協助您手動建立 T-SQL 專案，將它新增到應用程式中。

關於這項作業

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **T-SQL** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：
   a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

      為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (.ppf) 的位置。它也是所有相對路徑的基礎。
   b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。
5. 按一下**完成**。

新增 Visual Basic 專案

關於這項作業

註：僅支援在 Windows 上使用此專案類型。

這個主題中的步驟會引導您完成「新建專案精靈」（如果您是在應用程式中建立專案，則為「新建應用程式」精靈）中的所有頁面。當您在所選專案的「內容」視圖中建立專案之後，可以修改精靈中的設定。

註：如果是 PHP、VB6 和 Classic ASP，僅支援 ISO-8859-1（西歐）、UTF-8 和 UTF-16 字集。

程序

1. 在「瀏覽器」視圖中，選取專案要新增到其中的應用程式（如果尚未新增應用程式，請參閱第 32 頁的『配置應用程式』）。
2. 完成下列其中一個動作，以開啟「新建專案精靈」：
   a. 從主工作台功能表中，選取**檔案** > **新增專案** > **新建專案**。
   b. 用滑鼠右鍵按一下所選的應用程式，然後從快速功能表中，選擇**新增專案** > **新建專案**。
3. 在精靈的「選取專案類型」頁面中，選取 **Visual Basic** 作為專案類型，然後按**下一步**，進入精靈的下一頁。
4. 在「專案來源」精靈頁面中，執行下列動作：

a. 識別專案來源。「專案來源」由若干目錄組成，在這些目錄中，您將尋找專案檔及任何要包含在專案中的個別檔案。

   為專案命名並指定工作目錄。**工作目錄**是將放置 AppScan Source 專案檔 (.ppf) 的位置。它也是所有相對路徑的基礎。

b. 按一下**新增原始碼根目錄**來指定原始碼根目錄，以及掃描中所要併入或排除的目錄或檔案。 新增原始碼根目錄之後，您可以從其中排除特定目錄或檔案。如果要執行這個動作，請選取原始碼根目錄中的目錄或檔案（或複選這些項目），用滑鼠右鍵按一下選項，然後從功能表中選取**排除**。如果您併入或排除檔案，檔名左側的圖示也會改變。

5. 按一下**完成**。

複製專案

AppScan Source for Analysis 可讓您複製 .NET 專案以外的所有專案類型。修改專案不會影響複製的專案；複製專案之後，原始專案與複製的專案彼此無關。當複製匯入的專案時，您會用所有配置資訊來建立 AppScan Source 專案檔 (.ppf)。

程序

1. 在「瀏覽器」視圖中，用滑鼠右鍵按一下您想要複製的專案，然後在功能表中，選取**複製專案**。
2. 在「複製專案」對話框中，執行下列動作：
   a. 指定新專案的名稱。
   b. 識別複製之專案的目的地應用程式（目的地應用程式必須是一個手動建立的 AppScan Source 應用程式，或是使用「應用程式探索助理」建立的應用程式）。
   c. 識別目的地目錄（若為新專案，則為工作目錄）。

修改應用程式和專案內容

當您在「瀏覽器」視圖中選取應用程式或專案時，現行內容會出現在「內容」視圖中，您可以在這裡進行修改。

關於這項作業

第 218 頁的『「內容」視圖：選取的應用程式』和第 219 頁的『「內容」視圖：選取的專案』提供詳細資訊，來說明當您選取應用程式或專案時，可在「內容」視圖中修改的一些設定。

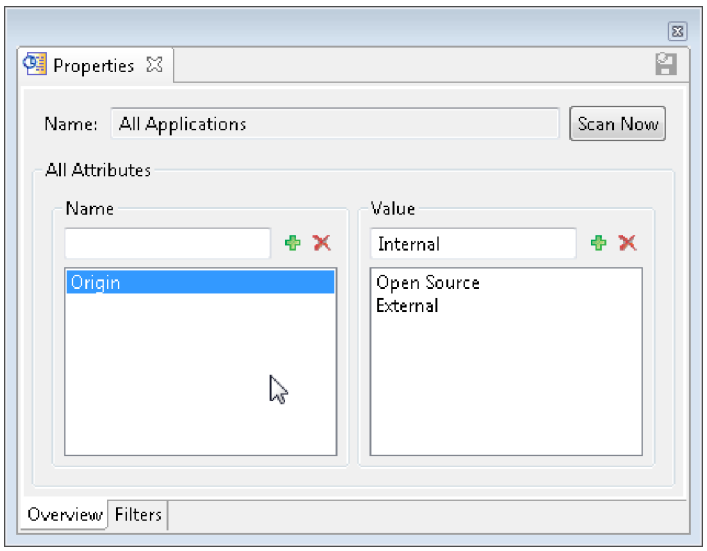
程序

1. 使用下列其中三種方法之一，開啟應用程式或專案的「內容」視圖：
   a. 在「瀏覽器」視圖中選取應用程式或專案，然後開啟「內容」視圖來顯示其內容。
   b. 在「瀏覽器」視圖中用滑鼠右鍵按一下應用程式或專案，然後選取**內容**。
2. 在「內容」視圖中檢閱內容。
3. 在適當的標籤頁上進行變更。可用的內容頁會隨著語言而不同。
4. 按一下**儲存**。

廣域屬性

必須先定義廣域屬性，它們才能與個別的應用程式相關聯。請在「瀏覽器」視圖中選取**所有應用程式**，以便在「內容」視圖中定義廣域屬性。

關於這項作業



如果要刪除屬性或它的值，請選取名稱或值，然後按一下**刪除屬性**（ ）。刪除屬性不會影響歷程結果。

如果要建立屬性，並提供給任何應用程式使用，請執行下列動作：

程序

1. 在「瀏覽器」視圖中，選取**所有應用程式**。
2. 在「內容」視圖中，開啟**概觀**標籤。
3. 輸入屬性的名稱，並且按一下**新增屬性**（ ）；或是按一下**新增屬性**，但不先指定名稱（之後將以對話框提示您輸入屬性的名稱）。
4. 輸入屬性的**值**，並且按一下**新增屬性值**；或是按一下**新增屬性值**，但不先指定值（之後將以對話框提示您新增一值）。
5. 重複這些步驟來新增多個屬性值。

應用程式屬性

應用程式屬性適用於目前選取的應用程式，會隨著先前建立的廣域屬性而不同。

程序

1. 在「瀏覽器」視圖中選取應用程式。
2. 在「內容」視圖中，開啟**概觀**標籤。
3. 按一下**新增屬性**。這時會出現**廣域屬性**對話框，其中有一份先前建立的屬性清單（關於建立廣域屬性的指示，可在『廣域屬性』中找到）。
4. 按兩下您想新增的屬性，或是選取它，然後按一下**確定**。這時會將屬性新增至「內容」視圖的「應用程式屬性」區段。

5. 按一下**值**直欄，從清單中選取這個應用程式的值（如果所建立的廣域屬性有多值，則會提供多個值）。您可以讓應用程式具有多個相關聯的屬性。

移除應用程式和專案

如果應用程式和專案沒有登錄，您可以將它們從 AppScan Source for Analysis 中移除。

程序

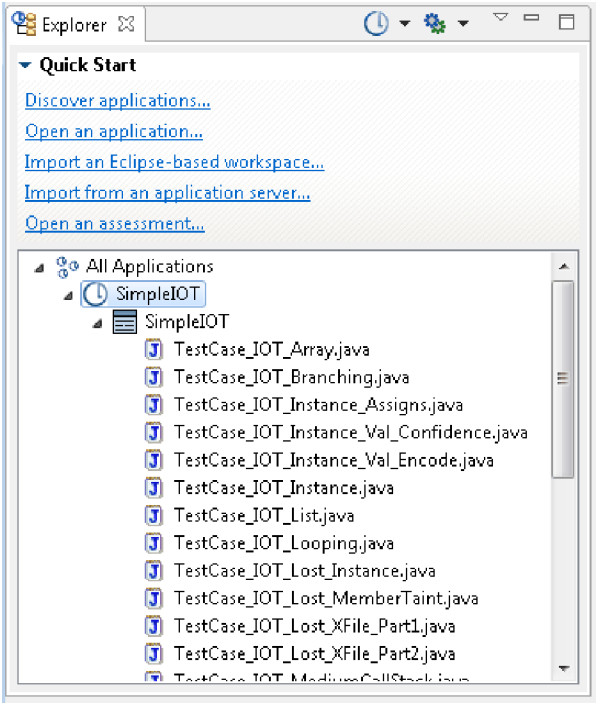
1. 選取您想要移除的應用程式或專案。您可以選擇移除多個應用程式和多個專案，不過，無法選擇混合移除應用程式和專案。
2. 完成下列動作之一：
   - 用滑鼠右鍵按一下選項，然後從功能表中，選擇**移除應用程式**或**移除專案**。
   - 按下鍵盤的 Delete 鍵。
   - 從主工作台功能表中，選取**編輯** > **移除**。

「瀏覽器」視圖

「瀏覽器」視圖的頂端有一個**快速入門**區段，底端則有一個瀏覽器區段，包含一個節點：**所有應用程式**。**快速入門**區段包含數個可啟動一般動作的實用鏈結。瀏覽器區段由一個樹狀結構窗格組成，提供資源的階層式視圖，包括：應用程式、專案、目錄和專案檔，而**所有應用程式**是它的根目錄。這些資源的導覽方式，很像檔案瀏覽器。當導覽視圖時，樹狀結構的選取狀態決定了「內容」視圖中所能使用的標籤。

- 第 74 頁的『一般資訊』
- 第 74 頁的『「快速入門」區段』
- 第 75 頁的『工具列按鈕』
- 第 75 頁的『右鍵功能表選項』
- 第 77 頁的『應用程式和專案指示器』

一般資訊



在「瀏覽器」視圖中，您可以使用工具列按鈕、**快速入門**區段中的鏈結，或瀏覽器區段中的右鍵功能表指令，以新增應用程式和專案，並掃描程式碼。新增應用程式之後，瀏覽器區段會提供應用程式和專案的視覺化指示器及各自的狀態。

提示：在「瀏覽器」視圖中，浮動說明可用來指示應用程式、專案及檔案的檔名和路徑。另外，浮動說明也會指出應用程式或專案是否已登錄。

「快速入門」區段

快速入門區段提供下列鏈結來啟動一般作業：

- **探索應用程式**：這會啟動「應用程式探索助理」，供您快速建立及配置 Java 和 Microsoft Visual Studio 原始碼的應用程式和專案。
- **開啟應用程式**：這會啟動「開啟」對話框，供您瀏覽以找出現有的應用程式並將其新增到應用程式集中。可以新增的檔案或目錄類型包括 `.paf`、`.sln`、`.dsw` 和 `.ewf`。
- **匯入 Eclipse 型工作區**：這會啟動「新增工作區」對話框，供您新增包含 Java 專案的現有 Eclipse 或 IBM Rational Application Developer for WebSphere 軟體 (RAD) 工作區。匯入工作區之後，您就可以掃描其中所包含的任何 Java 專案。

  註：在匯入工作區之前，請確定您已依照第 41 頁的『配置 Eclipse 和 Rational Application Developer for WebSphere 軟體 (RAD) 專案的開發環境』所說明來安裝和更新開發環境。
- **從應用程式伺服器匯入**：從 Apache Tomcat 或 WebSphere Application Server Liberty 應用程式伺服器匯入現有的 Java 應用程式。
- **開啟評量**：這會啟動「開啟」對話框，供您瀏覽到 AppScan Source 評量檔。可開啟的檔案類型包括 `.ozasmt` 和 `.xml`。

工具列按鈕

表 *8.* 工具列按鈕

動作	圖示	說明
新增應用程式功能表		按一下**新增應用程式功能表**按鈕的向下箭頭，可讓您選取動作來建立新的應用程式、開啟現有的應用程式、匯入工作區，或啟動「應用程式探索助理」。
掃描選項		**掃描選項**按鈕可讓您掃描瀏覽器區段中選取的物件。掃描時會使用預設掃描配置。如果要選擇不同掃描配置來用於掃描，請按一下**掃描選項**按鈕的向下箭頭。 選取您要使用的掃描配置，或者，選擇**編輯配置**動作，將另一個掃描配置設為預設值（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。
檢視功能表		**檢視功能表**按鈕會開啟功能表，讓您重新整理瀏覽器區段及隱藏已登錄的項目。

右鍵功能表選項

右鍵功能表選項的可用性取決於瀏覽器區段中所選取的項目。

- 當在瀏覽器區段中選取**所有應用程式**時，可用的右鍵功能表選項如下：
  - **掃描所有應用程式**：掃描所有應用程式。將以預設掃描配置來執行掃描。
  - **掃描所有應用程式的方式**：選取您要使用的掃描配置，或者，選擇**編輯配置**動作，將另一個掃描配置設為預設值（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。
  - **新增應用程式**
    - **建立新的應用程式**：新增應用程式到應用程式集中。這個動作會啟動「新建應用程式」精靈。
    - **開啟現有的應用程式**：這會啟動「開啟」對話框，供您瀏覽以找出現有的應用程式並將其新增到應用程式集中。可以新增的檔案或目錄類型包括 `.paf`、`.sln`、`.dsw` 和 `.ewf`。
    - **匯入現有的 Eclipse 型工作區**：這會啟動「新增工作區」對話框，供您新增包含 Java 專案的現有 Eclipse 或 IBM Rational Application Developer for WebSphere 軟體 (RAD) 工作區。匯入工作區之後，您就可以掃描其中所包含的任何 Java 專案。

      註：在匯入工作區之前，請確定您已依照第 41 頁的『配置 Eclipse 和 Rational Application Developer for WebSphere 軟體 (RAD) 專案的開發環境』所說明來安裝和更新開發環境。

- **探索應用程式**：這會啟動「應用程式探索助理」，供您快速建立及配置 Java 和 Microsoft Visual Studio 原始碼的應用程式和專案。

– **全部展開**

– **全部收合**

– **內容**：選取這個選項會開啟所選項目的「內容」視圖。

• 當在瀏覽器區段中選取應用程式時，可用的右鍵功能表選項如下：

– **掃描應用程式**：掃描所選的應用程式、專案或檔案。將以預設掃描配置來執行掃描。

– **掃描應用程式的方式**：選取您要使用的掃描配置，或者，選擇**編輯配置**動作，將另一個掃描配置設為預設值（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。

– **新增專案**

- **新建專案**：如果在「瀏覽器」視圖中選取應用程式，會啟用這個動作，選擇這個動作可讓您將新專案加到應用程式中。這個動作會啟動「新建專案精靈」。

- **現有的專案**：如果在「瀏覽器」視圖中選取應用程式，會啟用這個動作，選擇這個動作可讓您將現有的專案加到應用程式中。這個動作會啟動一個對話框，供您瀏覽要開啟的 `.ppf`、`.vcproj`、`.vcxproj`、`.csproj`、`.vbproj`、`.dsp`，或 `.epf` 檔。

- **多個專案**：新增多個專案到「瀏覽器」視圖所選的應用程式中。這個動作會啟動一個對話框，供您完成下列作業之一：

  • 指定要在其中搜尋專案的目錄。

  • 指定要在其中搜尋專案的工作區。

  • 指定要在其中搜尋專案的 Microsoft 解決方案檔案。

  在搜尋結果中，您可以選取一或多個要新增的專案。

– **移除應用程式**：如果在「瀏覽器」視圖中選取應用程式，會啟用這個動作，選擇這個動作會移除所選的應用程式。

– **新增自訂發現項目**：這個動作會啟動「建立自訂發現項目」對話框，供您建立所選應用程式的自訂發現項目。

– **重新整理**：重新整理所選的應用程式、專案或視圖的內容。

– 登錄/取消登錄：

- **登錄應用程式**：向 AppScan Source 登錄所選的應用程式或專案。您必須先登錄應用程式和專案，然後它們才能發佈到 AppScan Source 資料庫。

- **應用程式登錄為...**：選取這個選項會以新名稱來登錄應用程式。

- **取消登錄應用程式**：取消登錄所選的應用程式或專案。

- **尋找**：選取這個選項會將本端應用程式或專案，關聯於另一位 AppScan Source 使用者已登錄的應用程式或專案。

– **全部展開**

– **全部收合**

– **內容**：選取這個選項會開啟所選項目的「內容」視圖。

• 當在瀏覽器區段中選取專案時，可用的右鍵功能表選項如下：

- **掃描專案**：掃描所選的應用程式、專案或檔案。將以預設掃描配置來執行掃描。
- **掃描專案的方式**：選取您要使用的掃描配置，或者，選擇**編輯配置**動作，將另一個掃描配置設為預設值（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。
- **複製專案**：如果在「瀏覽器」視圖中選取專案，會啟用這個動作，選擇這個動作會開啟一個對話框，供您將專案複製到另一個應用程式中，或在目前包含專案的應用程式中建立專案的副本。
- **移除專案**：移除所選的物件。
- 登錄/取消登錄：
  - **登錄專案**：向 AppScan Source 登錄所選的應用程式或專案。您必須先登錄應用程式和專案，然後它們才能發佈到 AppScan Source 資料庫。
  - **取消登錄專案**：取消登錄所選的應用程式或專案。
  - **尋找**：選取這個選項會將本端應用程式或專案，關聯於另一位 AppScan Source 使用者已登錄的應用程式或專案。
- **全部展開**
- **全部收合**
- **內容**：選取這個選項會開啟所選項目的「內容」視圖。

- 當在瀏覽器區段中選取檔案時，可用的右鍵功能表選項如下：
  - **掃描檔案**：掃描所選的應用程式、專案或檔案。將以預設掃描配置來執行掃描。
  - **掃描檔案的方式**：選取您要使用的掃描配置，或者，選擇**編輯配置**動作，將另一個掃描配置設為預設值（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。
  - **從掃描中排除**：從掃描中移除所選的檔案。
  - **在內部編輯器中開啟**：在 AppScan Source 編輯器（位於「分析」視景）中，開啟所選的檔案。
  - **在外部編輯器中開啟**：選擇一個用來開啟所選檔案的外部編輯器。
  - **內容**：選取這個選項會開啟所選項目的「內容」視圖。

應用程式和專案指示器

這份表格指出「瀏覽器」視圖中的應用程式和專案圖示。

表 *9.* 應用程式和專案圖示

應用程式或專案類型	未登錄	已登錄	遺漏/找不到
已匯入的應用程式			
手動建立或利用「應用程式探索助理」來建立的應用程式			
已匯入的專案			
手動建立或利用「應用程式探索助理」來建立的專案			

「瀏覽器」視圖會顯示本端應用程式和專案，以及登錄在伺服器的應用程式和專案（登錄在伺服器而未儲存在本端的應用程式和專案會變成灰色，例如，其他使用者登錄的應用程式和專案）。如果您按一下工具列**檢視功能表**按鈕，並且將**隱藏登錄在伺服器的項目**功能表項目切換成未選取，則可以檢視現有的伺服器應用程式和專案。如果專案變成灰色，您可以按一下滑鼠右鍵，然後在功能表中選擇**尋找**。

第 3 章 喜好設定

喜好設定是關於 AppScan Source for Analysis 的外觀與操作的個人選項。

如果要開啟喜好設定頁面，請從主工作台主功能表中，選取**編輯** > **喜好設定**。您可以查看左窗格內的所有標題來瀏覽喜好設定，或利用左窗格頂端的過濾器欄位來搜尋較少量的一組標題。過濾器傳回的結果會同時符合喜好設定頁面標題和關鍵字（例如 JSP 或電子郵件）。

右窗格右上方的箭頭控制項，可讓您導覽先前檢視過的頁面。如果要在檢視若干頁之後，返回某一頁，請按向下箭頭，顯示最近檢視的喜好設定頁面清單。

一般喜好設定

一般喜好設定可讓您調整某些 AppScan Source for Analysis 預設值，以符合您的個人喜好設定。

選取語言

AppScan Source for Analysis 使用者介面可以顯示不同的國家語言。如果要變更顯示的國家語言，請從**選取語言**清單中選取它，然後在喜好設定對話框中，按一下**確定**。您必須以手動方式重新啟動工作台，變更才會生效。

註：如果要使用這個特性，在安裝程序中，至少需要安裝一個語言套件。如果只安裝英文，在使用這個喜好設定及重新啟動工作台之後，產品會顯示英文。在這個情況下，如果您想要顯示非英文，請重新執行安裝精靈，選擇新增一或多個語言套件來修復安裝。

檔案編碼

您必須設定專案中各檔案的字元編碼，AppScan Source 才能適當讀取檔案並且（舉例來說）在程式碼視圖中正確顯示它們。請在這個區段中，選取預設的字元編碼。

記載層次

請變更記載層次來提供錯誤日誌中所要包含的資訊層次。選擇**追蹤**、**除錯**、**參考資訊**、**警告**、**錯誤**或**嚴重**，其中**追蹤**提供最高雜訊層次的記載，**嚴重**只會記載重大事件，其餘設定依序提供層次逐漸提高的記載。

結束時儲存所有過濾器

選取之後，您會停用提示，當結束 AppScan Source 時，會自動儲存所有新建立或編輯的過濾器。

錯誤時取消掃描

選取之後，如果發生錯誤，就會取消掃描，以免掃描不完整。

為每個發現項目各建立一個標記

選取之後，如果您開啟了某項評量，在編輯器所開啟的掃描來源中，含有發現項目的位置會有標記。

依預設，會啟用建立標記。

建立標記有可能使掃描變慢。如果您的專案包括許多原始檔（或原始檔很大），停止建立標記，效能會比較好。

掃描完成之後

預設值是在掃描結束時出現提示，問您是否要自動開啟評量。如果您不想見到提示，請選取**一律開啟新評量**或**絕不開啟**。

掃描完成之後，相同目標有尚未儲存的掃描。

依預設，會提示您儲存或捨棄未儲存的現有掃描。您可以修改這項喜好設定，以便一律自動刪除重複的掃描，或者絕不自動刪除。當設定這項喜好設定時，請注意，當刪除重複的掃描時，會減少記憶體用量。

當發佈或匯出含絕對路徑的評量時

依預設，在發佈時，會提示您定義絕對路徑變數。這個區段中的設定可讓您停用這項預設提示，或自動開啟對話框，讓您定義變數（如果絕對路徑存在的話）。

在起始發佈之時，自動登錄應用程式

依預設，當您發佈未登錄之應用程式或專案的評量時，系統會提示您登錄它們。您可以選取發佈應用程式和專案時，一律登錄應用程式和專案，或永不登錄。

重要：您必須具備**登錄**許可權，才能登錄應用程式和專案。

當應用程式名稱衝突時

在 AppScan Source for Analysis 中，可能出現同名的應用程式，但是如果使用它們，卻可能出現難以管理的現象。依預設，如果您嘗試提供與現有應用程式相同的名稱給某個應用程式（或是在匯入應用程式時，使用與現有應用程式相同的名稱），將會提示您警告。警告訊息可讓您為應用程式產生唯一名稱、保留衝突的名稱或是取消。

如果您希望在發生衝突時，AppScan Source for Analysis 會自動產生唯一的應用程式名稱，請在喜好設定頁面中選取**產生唯一名稱**。如果您希望自動接受衝突的應用程式名稱，請選取**保留現有的名稱**。

註：這時會用檔名 <應用程式名稱>.paf 來儲存應用程式。如果您選擇**保留現有的名稱**，您所設定的工作目錄，不能與同名的現有應用程式相同。在此情況下，會提示您改寫現有的檔名，但因為現有的應用程式已在 AppScan Source for Analysis 中開啟，改寫將會失敗。

當專案名稱衝突時

只有在您試圖建立或匯入的專案，與相同應用程式中的現有專案同名時，才會套用這項設定。在此情況下，專案名稱不能衝突。如果您試圖這樣做，依預設，會提示您產

生唯一名稱，或是取消該動作。如果您希望在發生衝突時，AppScan Source for Analysis 會自動產生唯一的專案名稱，請在喜好設定頁面中選取**產生唯一名稱**。

啟動時，「已發佈的評量」和「我的評量」所顯示的評量數目

設定在「已發佈的評量」視圖或「我的評量」視圖中，要檢視的評量數目上限。

「歡迎使用視圖」中顯示的 *RSS* 資訊來源

依預設，「歡迎使用」視圖顯示 X-Force® RSS 資訊來源內容。如果要顯示替代內容，請輸入您要在**「歡迎使用視圖」中顯示的 RSS 資訊來源**欄位中顯示的 URL。

高度網路延遲最佳化

如果要快取用戶端相關資訊，將伺服器呼叫減到最少，請選取這個勾選框。

重新載入系統配置

載入最新的系統設定。如果您已在產品執行時，在產品之外變更設定（例如，透過修改 `<data_dir>\config`（其中 `<data_dir>` 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述） 中的 `.ozsettings` 檔來變更），選取這個按鈕會在產品中重新整理這些設定。

AppScan Enterprise Console 喜好設定

如果 AppScan Enterprise Server 已安裝 AppScan Enterprise Console 選項，您可以將評量發佈到那裡。 Enterprise Console 提供各種可處理評量的工具，例如：報告特性、問題管理、趨勢分析和儀表板。

如果要啟用這項特性，請完成 AppScan Enterprise Console 喜好設定頁面。這個頁面的所有欄位都必須完成有效項目，才能啟用 Enterprise Console 發佈：

- **使用者 ID** 欄位：輸入您的 AppScan Enterprise Server 使用者 ID（代表您的 AppScan Source 使用者來發佈時所建立的使用者 ID）。
  - 如果 AppScan Enterprise Server 配置為使用 Windows 鑑別，請輸入您用來連接到 Enterprise Console 的網域和使用者名稱（以 \ 區隔網域和使用者名稱 - 例如，`my_domain\my_username`）。
  - 如果 AppScan Enterprise Server 已配置 LDAP，請輸入您用來連接到 Enterprise Console 的使用者名稱。
  - 在 Windows 上，如果已在 AppScan Enterprise Server 中啟用「通用存取卡 (CAC)」鑑別，請從清單中選取「CAC 通用名稱」。

  最起碼，您必須是 QuickScan 使用者。如果您連接的是舊於 9.0.3 版的 AppScan Enterprise Server，您在 Enterprise Server 上必須有自己的使用者資料夾。
- **密碼**欄位：只有在您的 AppScan Enterprise Server 鑑別方法是使用者 ID 與密碼時，這個欄位才會是可用的。請輸入用來登入 Enterprise Console 的密碼（所輸入之使用者名稱的密碼）。
- **Enterprise Console URL** 欄位：輸入用來存取 Enterprise Console Web 應用程式的 URL。

  這個 URL 的格式如下：

`http(s)://<hostname>:<port>/ase`

其中，<hostname> 是 Enterprise Console 安裝所在的機器名稱，<port> 是主控台執行所在的埠（預設 <port> 是 9443）。舉例來說，這個 URL 可以是 `https://myhost.mydomain.ibm.com:9443/ase`。

註：

- 如果已經設定 **Enterprise Console URL**，則不需要修改此欄位。
- 您必須以**管理 AppScan Enterprise 設定**許可權登入 AppScan Source，才能夠設定 **Enterprise Console URL** 欄位。如需使用者帳戶和許可權的相關資訊，請參閱產品資訊中心的管理小節，或《*IBM Security AppScan Source* 安裝與管理手冊》的『管理*AppScan Source*』小節。
- **使用者 ID** 和**密碼**儲存在執行 AppScan Source 用戶端的機器（例如，AppScan Source for Analysis），而 **Enterprise Console URL** 儲存在 Enterprise Server（可能位於遠端機器上）。您不能從遠端機器存取使用者名稱和密碼資訊（例如，從該處發出 `getaseinfo` 指令）。
- AppScan Source 不支援發佈至已配置為使用 Proxy 設定的 AppScan Enterprise Console 實例。試圖發佈至使用 Proxy 設定的實例將會導致錯誤。

完成各項設定之後，強烈建議您按一下**測試連線**，確認 Enterprise Console 伺服器的連線有效。

提示：如果連線測試失敗，請檢查 Enterprise Console 伺服器在執行中，且您能夠在瀏覽器中存取它的控制中心 URL（請使用您在上方指定的同一個 **Enterprise Console URL**）。

JavaServer Pages 編譯的應用程式伺服器喜好設定

如果您掃描包含 JavaServer Pages (JSP) 的應用程式，AppScan Source 分析引擎必須能夠編譯 JSP 程式碼，以便加以分析。當建立 JSP 專案時，您必須指定 AppScan Source 應該使用的 JSP 編譯器（或接受 Java 和 JSP 喜好設定頁面中所能設定的預設編譯器）。如果 AppScan Source 無法編譯您的 JSP 檔，請利用應用程式伺服器喜好設定頁面，配置您的應用程式所使用的 JSP 編譯器。

AppScan Source 的安裝架構包含 Apache Tomcat 第 7 版和第 8 版。如果未配置 **Tomcat 7** 和 **Tomcat 8** 喜好設定頁面，AppScan Source 會利用提供的 Tomcat JSP 編譯器（目前標示為預設編譯器）來編譯 JSP 檔。如果您想要使用外部支援的 Tomcat 編譯器，請利用 Tomcat喜好設定頁面來指向您的本端 Tomcat 安裝架構。

如果您使用 Oracle WebLogic 伺服器 或 WebSphere Application Server，您必須配置適用的喜好設定頁面來指向應用程式伺服器的本端安裝架構，以便在分析期間用來編譯 JSP（如果您建立 JSP 專案時未先配置應用程式伺服器，屆時會提示您配置應用程式伺服器）。

Tomcat

這個主題說明配置 AppScan Source 來參照 Apache Tomcat 應用程式伺服器（不是 AppScan Source 所提供）時，所需設定的喜好設定。

AppScan Source 的安裝架構包含 Apache Tomcat 第 7 版和第 8 版。如果未配置 **Tomcat 7** 和 **Tomcat 8** 喜好設定頁面，AppScan Source 會利用提供的 Tomcat JSP 編譯器（目前標示為預設編譯器）來編譯 JSP 檔。如果您想要使用外部支援的 Tomcat 編譯器，請利用 Tomcat喜好設定頁面來指向您的本端 Tomcat 安裝架構。

如果您使用外部支援的 Tomcat 編譯器，請移至適當的喜好設定頁面，然後設定應用程式伺服器安裝目錄。指定安裝目錄可讓 AppScan Source 在配置專案時，自動尋找所有應用程式伺服器的相依關係。

WebLogic 11 和 12

這個主題說明配置 AppScan Source 來參照 Oracle WebLogic 伺服器 時所需設定的喜好設定。

在 WebLogic 喜好設定頁面中，您可以指定伺服器安裝目錄，且可以設定進階配置選項。指定安裝目錄可讓 AppScan Source 在配置專案時，自動尋找所有應用程式伺服器的相依關係。

請配置 AppScan Source 來參照 Weblogic 安裝目錄、Weblogic JAR 檔及 JavaServer Pages (JSP) 編譯器選項。

如果您需要變更預設 WebLogic JSP 編譯器選項或尋找 `weblogic.jar` 檔，請只選取**啟用進階配置選項**勾選框。預設 WebLogic JSP 編譯器選項為：

```
%JSP_JVM_OPTIONS%
 -Dcom.sun.xml.namespace.QName.useCompatibleSerialVersionUID=1.0
 -classpath
%JSP_COMPILER_CLASSPATH% weblogic.jspc
%JSP_OPTIONS% -verboseJspc -package
%PACKAGE_NAME% -linenumbers -g -debug -keepgenerated -compiler
%JAVAC_PATH% -webapp
%WEB_CONTEXT_ROOT_PATH% -d
%OUTPUT_PATH%
```

WebSphere Application Server

這個主題說明配置 AppScan Source 來參照 WebSphere Application Server 以進行 JSP 編譯時，所需設定的喜好設定。

如果要瞭解 AppScan Source 支援 WebSphere Application Server 的哪些版本，請參閱http://www.ibm.com/support/docview.wss?uid=swg27027486。

在 WebSphere Application Server 喜好設定頁面中，您可以指定伺服器安裝目錄，且可以設定進階配置選項。指定安裝目錄可讓 AppScan Source 尋找並使用 WebSphere Application Server JSP 編譯器。

配置 AppScan Source 來參照 WebSphere Application Server 安裝目錄。另外，進階配置選項也可讓您設定 WebSphere Application Server JSP 編譯器指令行和類別路徑。

要自訂 WebSphere Application Server JSP 指令行，或指定預設 WebSphere Application Server 類別路徑以外的類別路徑（如果您要在所有 WebSphere Application Server 應用程式所使用的類別路徑中併入其他的 JAR，請變更這項設定），請只選取**啟用進階配置選項**勾選框。

預設 WebSphere Application Server JSP 編譯器指令行選項為：

```
%CMD_EXE% %CMD_ARGS%
'%FILE(%%JSP_COMPILER_INSTALL_DIR%/bin/JspBatchCompiler%BAT%%)%'
-response.file
'%TMP_FILE(%-keepgenerated=true -recurse=true -useFullPackageNames=true
-verbose=false -createDebugClassfiles=true -jsp.file.extensions=%WEB_EXTS%
-javaEncoding=%ENCODING%
%JSP_OPTIONS% %QUOTE%-war.path=%WEB_CONTEXT_ROOT_PATH%%QUOTE%
%QUOTE%-filename=%RELATIVE_FILENAME_NO_QUOTE% %QUOTE%  %)%'
```

定義變數

當儲存評量或組合時，或是發佈評量時，AppScan Source for Analysis 可能會建議您建立變數，來取代絕對路徑（如果沒有變數，AppScan Source for Analysis 會將絕對路徑寫入評量檔中，以參照原始檔之類的項目）。配置絕對路徑的變數，有助於多部電腦共用評量。建議您在共用評量時使用變數。

關於這項作業

在起始儲存或發佈動作之前，您可以遵循這個主題中的指示來建立變數，或者在起始儲存或發佈動作之後，遵循第 120 頁的『在發佈和儲存時定義變數』中的步驟，來建立變數。

舉例來說，若想瞭解當共用評量時變數有何功效，請參閱第 120 頁的『範例：定義變數』。

程序

1. 從主功能表選取**編輯** > **喜好設定**。在「喜好設定」對話框中選擇**變更變數**。
2. 在「變更變數」喜好設定頁面中，按一下**新增變數**按鈕。
3. 輸入變數的名稱，且瀏覽至將以變數取代的檔案位置（在建立變數之後，AppScan Source for Analysis 會在周圍插入百分比符號 %）。
4. 針對評量中的其他任何參照項目，重複上述步驟（例如，如果評量參照來源位於多個位置，請針對每一個位置，各新增一個變數）。
5. 使用喜好設定頁面，用**修改變數**和**刪除變數**按鈕來編輯和移除變數。
6. 當完成定義變數時，請按一下**確定**。

利用喜好設定來啟用問題追蹤

「問題追蹤系統」喜好設定可讓您提交發現項目給問題追蹤系統，並且決定如何提交問題報告。

「問題追蹤系統」喜好設定頁面中的「一般」標籤可供您在 AppScan Source 中，啟用或停用「問題追蹤系統」整合特性。如果已選取**啟用問題追蹤系統整合**勾選框，評量發現項目就會有**提交問題報告**快速功能表動作可用。另外，「一般」標籤也提供了離散控制，控制在提交問題報告時將提供哪個「問題追蹤系統」。

如果要瞭解支援的問題追蹤系統所能設定的喜好設定，請參閱下列說明主題：

- 第 85 頁的『Rational ClearQuest 喜好設定』
- 第 85 頁的『Quality Center 喜好設定』
- 第 87 頁的『Rational Team Concert 喜好設定』

- 第 88 頁的『Team Foundation Server 喜好設定』

Rational ClearQuest 喜好設定

如果要完成 Rational ClearQuest 喜好設定，Rational ClearQuest 管理者必須向您提供必要的 Rational ClearQuest 設定。這些是您的 Rational ClearQuest 環境專用的設定。

註：當整合 Rational ClearQuest 8.0 版時，Rational ClearQuest 網目必須包含 **DefectTracking** 預定網目中的可用欄位。

資料庫集

一或多個問題報告資料庫的集合。

Linux 預設值 = 連線名稱，Windows 預設值 = 資料庫集

資料庫名稱

提交問題報告的目標資料庫名稱。

資料庫使用者名稱

預設 Rational ClearQuest 資料庫使用者名稱。

CQPerl 執行檔的位置

Rational ClearQuest CQPerl 執行檔在本端電腦的位置。提供的預設位置對映於預設的 Rational ClearQuest 安裝位置。

問題報告記錄的實體

Rational ClearQuest 安裝所配置用於問題報告物件的實體（資料庫物件）。

預設實體是**問題報告**。

記錄的說明欄位

預設說明是**說明**。

記錄的標題欄位

預設標題是**標題**。

每個發現項目單一問題報告

請將發現項目群組提交成單一問題報告或多份問題報告。當建立問題報告時，您可以變更提交方法。

Quality Center 喜好設定

您必須先在「一般問題追蹤系統」喜好設定中啟用 HP Quality Center，然後在 Quality Center 標籤中，設定個別喜好設定。

伺服器 URL

Quality Center Server URL - 例如，http://<主機名稱>:<埠>/qcbin/ 或 https://<主機名稱>:<埠>/qcbin/。

使用者名稱（選用）

登入 Quality Center 的使用者名稱

密碼（選用）

如果您輸入使用者名稱，請為其輸入密碼。

網域

要連接的「Quality Center 網域」。

專案

要連接的「Quality Center 專案」

自動登入

如果是 true，當提交發現項目時，AppScan Source 不會提示您輸入登入資訊，而會以「喜好設定」所指定的預設認證來登入。如果是 false，您每次向「品質中心」提交發現項目時，都必須登入。

自動提交

如果是 true，當提交發現項目時，不會出現用來提交新問題報告的對話框。AppScan Source for Analysis 會使用「喜好設定」所指定的**預設問題報告內容**。如果是 false，當提交發現項目時，會在提示中要求您輸入問題報告資訊（嚴重性、優先順序、問題報告類型、狀態，等等）。

重新提交先前提交的發現項目

提交給 Quality Center 的發現項目會標示一些 Quality Center 問題報告資訊（問題報告 ID、提交使用者及提交日期）。依預設，AppScan Source 不會重新提交相同的發現項目超過一次。如此您便可以將多個發現項目分派給 Quality Center，只需要在 Quality Center 資料庫中輸入新的發現項目。如果選取的話 (true)，可以將先前提交的發現項目重新提交給 Quality Center。

將每個發現項目當作個別錯誤來提交

當在單一作業中提交多個發現項目時，您可以用一份 Quality Center 問題報告來提交所有發現項目，也可以針對每個 AppScan Source 發現項目各提交一份 Quality Center 問題報告。選取這個勾選框，會將這個旗標設成 true，為每一個別發現項目都分別建立一份 Quality Center 問題報告。將這個旗標設為 false，會為所有在大量提交中提交的發現項目建立一份 Quality Center 問題報告。

自動產生錯誤摘要

如果是 true，AppScan Source 會自動產生 Quality Center 的問題報告提交摘要。摘要會指出問題報告所包含的發現項目數，以及發現項目的類型，例如 `Validation.Required`。

如果是 false，當提交問題報告時，在建立新問題報告而開啟的對話框中，會顯示有待填寫的「摘要」欄位。

自動載入錯誤欄位

預設值是 true。當選取這個勾選框時，AppScan Source 會根據 Quality Center 的現行使用者和群組設定，自動載入 Quality Center 資料庫中的問題報告欄位定義。如果是 false，在建立新問題報告而開啟的對話框中，AppScan Source 不會顯示 Quality Center 的問題報告欄位。

預設問題報告內容

如果要設定不同 Quality Center 問題報告屬性的預設值，請在 Quality Center 喜好設定標籤中，按一下**預設問題報告內容**。預設值有可能在提交之時，預先移入**新建問題報告**對話框，如果選取**自動提交**喜好設定，就會以無聲自動的方式傳送到 Quality Center。

註：如果選取**自動載入錯誤欄位**，每次出現**問題報告內容**對話框時，都會從 Quality Center 動態取出問題報告內容及可用的值。因此，任何新增到 Quality Center 資料庫的新欄位和值，都會自動出現在 AppScan Source for Analysis 中。必須提供有效的伺服器、登入和連線資訊之後，才能開啟**問題報告內容**對話框，將 Quality Center 資訊移入其中。

自訂 Quality Center 問題報告欄位

您可以在「新建問題報告」對話框中，利用配置檔來自訂欄位，以及這些欄位之間的互動。您可以在 `<data_dir>\config\qc.dts`（其中 `<data_dir>` 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述） 中找到範例配置檔，其中含有自訂範例及其他文件。這些自訂可讓您直接在「新建問題報告」對話框中，建立「Quality Center 工作流程」的 Script 邏輯模型。

可用的自訂包括：

- 顯示自訂的欄位、遺漏的欄位，或兩者
- 強制一律顯示欄位（置換 Quality Center 設定）
- 根據其他欄位的選項來更新所需要的欄位狀態
- 根據另一個欄位中的清單框選項，動態更新欄位的清單框選項

Rational Team Concert 喜好設定

Rational Team Concert 喜好設定標籤可讓您配置 Rational Team Concert 伺服器連線，以及配置工作項目的屬性值。

輸入連線資訊且順利登入之後，您可以選擇連接到一或多個專案區域。每個專案區域都可以配置它自己的屬性預設值。

註：當您連接到 Rational Team Concert（藉由配置喜好設定或提交問題報告），系統可能會提示您接受 SSL 憑證。如需相關資訊，請參閱 『Rational Team Concert SSL 憑證』。

如果要配置給定專案區域的屬性值，請選取專案區域，然後選擇**配置**。在配置對話框中，您可以將屬性值設為寫入程式的值，在某些情況下，也可設為參照所選發現項目的變數。例如，在屬性值中使用 {Finding.fileName}，在提交期間，會取代為發現項目實際的原始碼檔名。對支援這些變數的屬性值提供「內容輔助」(<Alt>+</>)。團隊最好利用 Rational Team Concert 喜好設定主頁面中的**匯入**和**匯出**按鈕來共用這些配置。

Rational Team Concert SSL 憑證

當安裝 Rational Team Concert 伺服器時，應該配置它來使用有效的 SSL 憑證。如果沒有執行這個步驟，當登入伺服器時，會收到未授信連線訊息（配置喜好設定或提交問題報告之時）。這個主題概述 Rational Team Concert SSL 憑證考量。

SSL 憑證儲存體位置

已永久接受的憑證儲存在 <user_home>/.jazzcerts（其中 <user_home> 是作業系統起始目錄（例如，在 Windows 上，此目錄可能是 C:\Documents and Settings\Administrator\）） 中。移除 <user_home>/.jazzcerts 會刪除 AppScan Source 和 Rational Team Concert 用戶端所有已儲存的憑證。

與 Rational Team Concert 用戶端共用的 SSL 憑證

AppScan Source 會與 Rational Team Concert 用戶端共用它的 SSL 憑證儲存庫。如果您利用 Rational Team Concert 用戶端來永久接受某個憑證，AppScan Source 會重複使用它（在 AppScan Source 中，系統不會提示您接受憑證）。同樣地，如果您在 AppScan Source 中永久接受某個憑證，Rational Team Concert 用戶端也會重複使用這個憑證。

Rational Team Concert 伺服器重新命名的考量，使用時機為當問題追蹤來源是 AppScan Source for Analysis

如果您已啟用 Rational Team Concert 在 AppScan Source for Analysis 的問題追蹤，並且 Rational Team Concert 伺服器已重新命名，則在 AppScan Source for Analysis 中的全部現有專案區域在伺服器上的配置將不再可用。您需要透過新的儲存庫 URI 連接到伺服器，並在「問題追蹤系統」喜好設定中重建配置。

Team Foundation Server 喜好設定

Team Foundation Server 喜好設定標籤可讓您配置連接到 Microsoft Team Foundation Server 的連線，以及配置工作項目欄位的值。

輸入連線資訊且順利登入之後，您可以選擇連接到一或多個專案。

註：在配置 Team Foundation Server 2010 的登入時，「伺服器 URL」必須包含要連接的「團隊專案集合」。例如：http://myserver:8080/tfs/*DefaultCollection*。

每個專案都可以配置它自己的欄位預設值。

如果要配置給定專案的欄位值，請選取專案，然後選擇**配置**。在配置對話框中，您可以將欄位值設為寫入程式的值，在某些情況下，也可設為參照所選發現項目的變數。

例如，在欄位值中使用 {Finding.fileName}，在提交期間，會取代為發現項目實際的原始碼檔名。會對支援這些變數的欄位提供「內容輔助」(<Alt>+</>)。

團隊最好利用 Team Foundation Server 喜好設定主頁面中的**匯入**和**匯出**按鈕，來共用這些配置。

Eclipse 工作區匯入器：Eclipse 或 Rational Application Developer for WebSphere 軟體 (RAD) 喜好設定配置

AppScan Source for Analysis 安裝架構提供預設的 Eclipse 匯入器。這個匯入器會識別 Eclipse 和 JRE 的位置。如果預設 Eclipse 匯入器無法匯入您的工作區，可能需要建立新的 Eclipse 匯入器。

開始之前

每個匯入器配置都代表一個 Eclipse 或 Rational Application Developer for WebSphere 軟體 (RAD) 安裝架構。如果要使用這些配置來將現有的工作區和專案匯入到 AppScan Source for Analysis，您可能也需要在 Eclipse 環境安裝 AppScan Source for Development 外掛程式。

新增 RAD 工作區之前，您必須先建立工作區類型的配置。

程序

1. 在 AppScan Source for Analysis 中，從主工作台功能表中，選取**編輯** > **喜好設定**。
2. 選取 **Eclipse 工作區匯入器**。
3. 按一下**建立新的配置**，然後完成「新建匯入配置」對話框，以建立新的配置：
   - **產品**：選取適當的產品。

     註：如果無法讓您選取用來建立工作區的產品，在嘗試建立工作區匯入器之前，請確定您已完成 第 41 頁的『Eclipse 或 Application Developer 更新項目』所概述的配置步驟。
   - **名稱**：匯入器名稱
   - **位置**：Eclipse 安裝架構的基本目錄路徑
   - **JRE 位置**：Java 執行時期環境 (JRE) 的根目錄路徑。請使用 <install_dir>\JDKS（其中 <install_dir> 是 AppScan Source 安裝的位置） 中的 JDK，或其他任何偏好的 JDK。
4. 按一下**確定**。
5. 如果要將匯入器識別為預設值，請選取它，然後按一下**將所選的配置設為預設值**。這時匯入器的**預設值**直欄中，會出現一個圖示。

電子郵件

請配置電子郵件設定，其用來傳送視為問題報告的發現項目。

- **收件者位址**：收件者的電子郵件位址。依預設，「以電子郵件傳送發現項目」對話框中的**郵件收件者**欄位會移入這個電子郵件位址，不過，準備電子郵件時，很容易改變它。
- **來源位址**：傳送者的電子郵件位址。

註：建議使用有效的電子郵件位址，以免收件人的郵件用戶端將該電子郵件視為垃圾郵件。

- **郵件伺服器**：配置為 mail.myexample.com 的 SMTP 郵件伺服器。

  重要：請洽詢您的系統管理者，確定您的郵件伺服器資訊正確。

Java 和 JavaServer Pages

請使用這個喜好設定頁面來新增、修改或刪除用於掃描的「Java 開發套件 (JDK)」（並設定預設 JDK）。此外，請使用此頁面來設定預設的 JavaServer Page (JSP) 編譯器。

預設

識別掃描所用的 JDK 位置。當專案未指定明確的 JDK 時，掃描會使用預設 JDK 路徑。如果要將 JDK 設為預設值，請用滑鼠右鍵按一下識別的 JDK 名稱，然後按一下**設定預設 JDK**。預設圖示會出現在表格中，識別目前的預設 JDK。

註：JSP 專案既有的預設編譯器是 Tomcat 7，它需要 Java 1.6 版或更高版本。如果保留 **Tomcat 7** 作為預設值，則使用較舊的 JDK 會導致在掃描期間發生編譯錯誤。

JDK 名稱和路徑

識別 JDK 的名稱和位置。

JSP 專案的預設編譯器

既有的 Tomcat 7 即是預設 JSP 編譯器設定。如果要瞭解 AppScan Source for Analysis 支援的編譯器，請參閱 http://www.ibm.com/support/docview.wss?uid=swg27027486。

知識庫文章

請使用「知識庫文章」喜好設定頁面來設定包含 AppScan Source 安全知識庫文章的位置。

此頁面列出含有文章的目錄。如果要新增目錄，請按一下**新增內容目錄**，然後瀏覽至文章位置。如果要移除目錄，請選取它，然後按一下**移除**。

專案副檔名

配置或新增各專案類型的有效廣域副檔名，變更要包含在掃描中的副檔名、從掃描中排除，以及將副檔名指定為 Web 檔。

每個可用的語言或專案類型都會有一個標籤頁：
Java、JavaScript、ASP、Perl、PHP、ColdFusion、PBSA（用於基於型樣的專案類型）、COBOL、PL/SQL、T-SQL、VB.NET、.Net 組譯碼、VB、C/C++、ASP .NET 1.x、ASP .NET 2.x、WSDL 和 C#。當新增副檔名時，請識別是否可以掃描使用新副檔名的檔案、將它們視為 Web 檔或加以排除。

這個頁面中的設定是廣域的。如果要為個別的專案設定副檔名，請針對選取的專案，使用「內容」視圖的第 221 頁的『副檔名』標籤。

副檔名設定

表 *10.* 副檔名設定

設定	說明	用法範例
掃描或**評量**	將使用所指副檔名的檔案包含在完整分析中。	• 如果為 Java 專案建立 .xxx 副檔名，並標示為**掃描**或**評量**，則會編譯和掃描使用該副檔名的檔案。 • 如果不應編譯和掃描檔案，檔案可以是專案的一部分，但不要標示為**掃描**或**評量**（例如 C++ 標頭檔）。這些檔案會包含在專案中，並在執行基於型樣的分析期間進行搜尋。
Web 檔	標示使用所指副檔名的檔案，以進行 JSP 編譯。這項設定容許 AppScan Source 將 Web 原始檔與非 Web 原始檔加以區隔。	如果為 Java 專案建立 .yyy 副檔名，並標示為 **Web 檔**，則會將使用該副檔名的檔案安排成專案中的 Web 原始檔。當 AppScan Source 準備分析時，會將這些檔案前置編譯成要分析的類別。
排除	不在專案中為使用所指副檔名的檔案建立原始檔。將不會掃描使用這個副檔名的檔案。	為您專案所需的檔案建立 .zzz 副檔名，以進行編譯，但不需要包含在分析中。

第 4 章 掃描原始碼及管理評量

本節說明如何掃描原始碼及管理評量。

配置應用程式和專案之後，或使用應用程式探索助理來建立應用程式和專案之後，您就可以開始掃描原始碼。掃描結果是一項評量，您可以將它儲存起來，或進行發佈。已儲存的評量是儲存在本端的掃描結果檔，您可在之後加以發佈及開啟來進一步分類，您也可以在 AppScan Source for Development 中開啟它。已發佈的評量由儲存在 AppScan Enterprise Server 的掃描結果組成。

您利用兩個視圖來管理評量：

- 我的評量
- 已發佈的評量

註：當儲存、發佈或開啟一項評量時，狀態列會顯示進度。

掃描原始碼

這個作業說明啟動掃描的各種方法。

關於這項作業

您可以進行各種層次的掃描（所有應用程式、一或多個應用程式、一或多個專案，或一或多個檔案）。如果已完成掃描，只要還開啟評量，您就可以再次掃描它。

- 『掃描所有應用程式』
- 第 94 頁的『掃描一或多個應用程式』
- 第 94 頁的『掃描一或多個專案』
- 第 94 頁的『掃描一或多個檔案』
- 第 95 頁的『重新掃描程式碼』

請參閱第 95 頁的『掃描考量』，以瞭解作業系統特定考量、語言特定考量，或可能影響您掃描的其他限制。

掃描時一律會使用掃描配置。如果您設定預設掃描配置，然後刪除它，則掃描時將自動使用**正常掃描**內建掃描配置。 如果要進一步瞭解掃描配置，請參閱第 96 頁的『管理掃描配置』及下列的掃描選項說明。

掃描所有應用程式

程序

完成下列動作之一：

1. 在主工作台功能表中，選取**掃描** > **掃描全部**。將以預設掃描配置來執行掃描。
2. 在「瀏覽器」視圖中：
   - 用滑鼠右鍵按一下**所有應用程式**，然後從功能表中，選取**掃描所有應用程式**。將以預設掃描配置來執行掃描。

- 如果要使用不同掃描配置來進行掃描，請用滑鼠右鍵按一下**所有應用程式**，然後從功能表中，選取**掃描所有應用程式的方式**。選取您要使用的掃描配置，或者，如果要設定不同的預設掃描配置，請選擇**編輯配置**動作（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。

掃描一或多個應用程式

程序

1. 在「瀏覽器」視圖中，選取一或多個應用程式。
2. 完成下列動作之一：
   a. 從主工作台功能表中，選取**掃描** > **掃描選項**。將以預設掃描配置來執行掃描。
   b. 在「瀏覽器」視圖中：
      - 用滑鼠右鍵按一下選項，然後從功能表中，選擇**掃描應用程式**。將以預設掃描配置來執行掃描。
      - 如果要使用不同掃描配置來進行掃描，請用滑鼠右鍵按一下選項，然後從功能表中，選取**掃描應用程式的方式**。選取您要使用的掃描配置，或者，如果要設定不同的預設掃描配置，請選擇**編輯配置**動作（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。

掃描一或多個專案

程序

1. 在「瀏覽器」視圖中，選取一或多個專案。
2. 完成下列動作之一：
   a. 從主工作台功能表中，選取**掃描** > **掃描選項**。將以預設掃描配置來執行掃描。
   b. 在「瀏覽器」視圖中：
      - 用滑鼠右鍵按一下選項，然後從功能表中，選擇**掃描專案**。將以預設掃描配置來執行掃描。
      - 如果要使用不同掃描配置來進行掃描，請用滑鼠右鍵按一下選項，然後從功能表中，選取**掃描專案的方式**。選取您要使用的掃描配置，或者，如果要設定不同的預設掃描配置，請選擇**編輯配置**動作（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。

掃描一或多個檔案

程序

1. 在「瀏覽器」視圖中，選取一或多個檔案。
2. 完成下列動作之一：
   a. 從主工作台功能表中，選取**掃描** > **掃描選項**。將以預設掃描配置來執行掃描。
   b. 在「瀏覽器」視圖中：
      - 用滑鼠右鍵按一下選項，然後從功能表中，選擇**掃描檔案**。將以預設掃描配置來執行掃描。
      - 如果要使用不同掃描配置來進行掃描，請用滑鼠右鍵按一下選項，然後從功能表中，選取**掃描檔案的方式**。選取您要使用的掃描配置，或者，如果要設定不同的預設掃描配置，請選擇**編輯配置**動作（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。

重新掃描程式碼

程序

如果要重新掃描現行目標，請從主功能表中選取**掃描** > **重新掃描**。這次掃描會使用前次用來掃描項目（或選取的項目）的掃描配置：

- 如果前一次掃描使用預設掃描配置，但已經設定新的預設掃描，則會使用新的預設掃描配置來執行掃描。
- 如果前一次掃描使用非預設的掃描配置，則會用該配置來執行這一次掃描。自前一次掃描之後，如果已修改並儲存掃描配置，則會使用已修改的掃描配置。

掃描考量

本主題說明可能影響您掃描的限制和考量。

- 『一般』
- 『Windows』
- 『Linux』
- 『Java』

一般

限制：當您掃描多個應用程式或專案時，在「我的評量」視圖中，會建立一個母節點來包含每個掃描項目的評量。在這種情況下，無法管理個別子項評量（例如，無法個別移除或發佈子項評量）。當同時掃描多個應用程式或專案時，您只能以群組（母節點）的方式來管理各個評量。

重要：如果您使用的 AppScan Source 專案在開發環境中（例如 IBM MobileFirst Platform 專案）有相依關係，請務必先在開發環境中建置專案，然後再匯入它。匯入專案之後，如果您修改其中的檔案，請務必先在開發環境中重建它，再於 AppScan Source 中掃描（如果不這樣做，AppScan Source 會忽略您對檔案所做的修改）。

Windows

掃描個別或多個選取的檔案的功能，在 Microsoft .NET 檔案中無法使用（例如，`.cs` and `.vbnet`）。

Linux

AppScan Source for Analysis 用戶端是以 Eclipse 為建置基礎。在 Linux 上，Eclipse 需要安裝協力廠商元件，才能呈現瀏覽器型的內容。若無此元件，AppScan Source for Analysis 可能出現症狀，例如登入之後當機或產品使用期間失敗。請參閱 第 106 頁的『在 Linux 上啟用 AppScan Source for Analysis 的瀏覽器型內容』，以取得詳細資訊。

Java

提示：如果您要掃描 Java 但 Java 專案中有遺漏的相依關係，AppScan Source 會綜合相依關係可能已提供的片段，來建立追蹤資料。此綜合作業未必能精確反映 `.jar` 檔中的資訊。如果要限制綜合作業，藉此改善發現項目的正確性，在此情況下，您可以依下列方式指定遺漏的相依關係：

1. 掃描之後，開啟 <data_dir>\logs\scanner_exceptions.log（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）查看 AppScan Source 是否有所報告的遺漏相依關係。
2. 修改專案內容來併入相依關係。要執行這項作業，請遵循第 71 頁的『修改應用程式和專案內容』的指示，然後在 **JSP 專案相依關係**或**專案相依關係**標籤中指定並儲存相依關係。
3. 重新掃描專案。

註：依預設，AppScan Source 會掃描含有遺漏的相依關係或編譯錯誤的 Java 檔和 Java 位元組碼。這些設定可以依下列方式加以變更：

1. 在文字編輯器中開啟 <data_dir>\config\scan.ozsettings。
2. 如果要變更編譯錯誤設定，請在檔案中找出 compile_java_sources_with_errors。這項設定看起來如下：

```
<Setting
  name="compile_java_sources_with_errors"
  value="true"
  default_value="true"
  type="bool"
  hidden="true"
  display_name="compile_java_sources_with_errors"
  description="Attempt to scan java code with compilation errors."
/>
```

3. 如果要變更遺漏的相依關係設定，請在檔案中找出 scan_java_bytecode_without_dependencies。這項設定看起來如下：

```
<Setting
  name="scan_java_bytecode_without_dependencies"
  value="true"
  default_value="true"
  type="bool"
  hidden="true"
  display_name="scan_java_bytecode_without_dependencies"
  description="Scans Java bytecode even when some of
    the dependencies are missing by artificially
    synthesizing the unresolved symbols."
/>
```

4. 在設定中，修改 value 屬性。如果該屬性設為 true，這項設定即會開啟。如果編譯錯誤設定設為 false，AppScan Source 在掃描期間會略過含有編譯錯誤的 Java 程式碼。如果遺漏的相依關係設定設為 false，當有遺漏的相依關係時，AppScan Source 就不會掃描 Java 位元組碼。
5. 修改好這項設定之後儲存檔案，再啟動或重新啟動 AppScan Source。

管理掃描配置

啟動掃描時會使用掃描配置。在掃描配置中，您可以指定掃描期間要使用的來源規則。掃描配置中的設定通常可以獲得較佳的掃描結果，而儲存這些設定的能力更可讓掃描變得輕鬆又有效率。

關於這項作業

這個作業說明管理掃描配置所包含的步驟。

- 第 97 頁的『建立掃描配置』
- 第 99 頁的『修改掃描配置』

- 第 100 頁的『移除掃描配置』
- 第 100 頁的『共用掃描配置及使用共用的配置』
- 第 100 頁的『將掃描配置設為預設值』
- 第 101 頁的『內建掃描配置』

您可以在第 101 頁的『掃描配置視圖』中管理掃描配置。您可以從主功能表列選取**檢視** > **掃描配置**，或在「瀏覽器」視圖中選取**編輯配置**動作，以開啟這個視圖。

一旦您已經備妥掃描配置，當在 AppScan Source for Analysis 啟動掃描時，您可以使用他們（如需相關資訊，請參閱第 93 頁的『掃描原始碼』）。當在 AppScan Source for Automation、AppScan Source for Development 和 AppScan Source 指令行介面 (CLI)啟動掃描時，您也可以使用掃描配置。

建立掃描配置

程序

1. 完成下列動作之一：
   a. 按一下「掃描配置」視圖的**新建**按鈕。
   b. 請從清單選取現有配置，然後按一下**複製**。這樣會根據原始掃描配置的設定來建立掃描配置，此配置可以經過修改並另存為新的配置。
2. 在**一般**標籤 - **基本資訊**區段中：
   a. 在**名稱**欄位中輸入配置的唯一名稱。請注意，在掃描配置中，指定唯一名稱是唯一的必要設定，所有其他設定都是選用的。
   b. 選擇性的： 輸入掃描配置的說明。
3. 選擇性的： 使用**一般**標籤 - **過濾器資訊**區段，來設定掃描的過濾器。如果要瞭解過濾器，請參閱第 126 頁的『利用過濾器分類』。 在此區段中，您可以選擇一個以上的過濾器，每當使用掃描配置時即將其套用至掃描。選取過濾器時，您可以選擇 AppScan Source 選擇的過濾器，或共用的過濾器，或您自己建立的過濾器。 在此區段中：
   a. 按一下**新增**，然後在「選取過濾器」對話框中，選擇您想要新增的過濾器。選取過濾器時，其特質會以唯讀形式出現在對話框的右邊。按一下**確定**，將過濾器新增至掃描配置中。

      註：

      - 如果要將過濾器的反轉套用至掃描配置中，在按一下**確定**之前，請先選取**反轉過濾器**。
      - 在「選取過濾器」對話框中，您可以複選多個過濾器來新增。當您這麼做時，如果已選取**反轉過濾器**勾選框，所選取過濾器的反轉將新增至掃描配置中。

      結束「選取過濾器」對話框之後，過濾器將出現在清單中，且**反轉**直欄指出是否反轉過濾器。
   b. 如果要移除所新增的過濾器，請選取或複選過濾器，然後按一下**移除**。
   c. 排除過濾器含有一些規則，用來將漏洞類型、應用程式設計介面 (API)、檔案、目錄、專案或追蹤規則從發現項目移除。如果您在掃描配置中包含多個排除過濾器，彼此可能發生衝突，而影響發現項目。例如，假設有下列兩個過濾器：

- 過濾器 1 會移除漏洞類型 Validation.EncodingRequired 的所有發現項目。這不會反轉，因此會將這些發現項目從評量排除。
- 過濾器 2 會移除漏洞類型 Validation.Required 的所有發現項目。這不會反轉，因此會將這些發現項目從評量排除。

如果使用掃描配置套用了這兩個過濾器，依預設，它們會支配對方。過濾器 1 會排除 Validation.EncodingRequired 發現項目 - 但它會包含 Validation.Required 發現項目。過濾器 2 會排除 Validation.Required 發現項目 - 但它會包含 Validation.EncodingRequired 發現項目。最終結果是 Validation.EncodingRequired 和 Validation.Required 發現項目都包含在內。

如果要移除指定之任何排除過濾器的發現項目，請選取**符合任何未反轉的排除過濾器**。以上述範例來說，如果選取這個勾選框，就會將所有 Validation.EncodingRequired 和 Validation.Required 發現項目從評量排除。

4. 選擇性的： 使用**污染流分析**標籤 - **污染流分析**區段，來啟用污染流分析。依預設，會選取**污染流分析**，它是 AppScan Source 所執行的主要分析類型。當您啟動掃描時，污染流分析會產生資料流追蹤，讓您更精確判定漏洞。您可以設定分析的**範圍**如下：
   - **應用程式**範圍：將在應用程式內的各專案間以及在專案內的檔案間執行污染流分析。
   - **專案**範圍：將在專案內的檔案間執行污染流分析。
   - **檔案**範圍：將個別對每一個檔案執行污染流分析。

   註：掃描 JavaScript、ColdFusion、Perl、Cobol、PL/SQL 或 T-SQL 時，**污染流分析**標籤中的設定不適用。
5. 選擇性的： **污染流分析**標籤 - **掃描規則**區段可讓您指定掃描時採用的來源規則（如需相關資訊，請參閱第 102 頁的『「污染流分析」標籤』）。在這個區段中，您可以選擇使用選取的來源規則集來掃描，也可以選取要用於掃描的個別規則內容：
   a. 依預設，這個區段可讓您選擇要套用的規則集。請選取一或多個可用的規則集勾選框。
   b. 如果要選擇個別的規則內容，而不是規則集，請按一下**捨棄選取的規則集，讓我選取個別規則內容**。這時會開啟「選取規則內容」對話框，讓您選擇個別的規則內容。如果完成這個對話框，則會捨棄任何已選取的規則集。掃描時，將會使用含有所選規則內容的掃描規則。

   如果您為掃描選擇個別規則內容，後來想要改為選取規則集，請按一下**捨棄選取的規則內容，讓我依規則集來選取**。這樣會捨棄「選取規則內容」對話框中已選取的任何規則內容，讓您改為選取規則集。

   註：
   - 選取個別的規則內容時，選取的項目會套用至來源的內容，而不是接收槽的內容。這表示您將攻擊面縮小到只限於具有選定內容的來源。您在結果中可能會看到不符合選定內容的漏洞，因為漏洞類型是根據接收槽，而不是來源。
   - 掃描 JavaScript、ColdFusion、Perl、Cobol、PL/SQL 或 T-SQL 時，**污染流分析**標籤中的設定不適用。
6. 選擇性的： **污染流分析**標籤 - **進階設定**區段僅適用於進階使用者。它包含各種可改進掃描結果的設定。這個區段中的每一個設定都有浮動說明。

註：掃描 JavaScript、ColdFusion、Perl、Cobol、PL/SQL 或 T-SQL 時，**污染流分析**標籤中的設定不適用。

7. 選擇性的： **型樣分析**標籤中的設定可讓您針對基於型樣的掃描啟用及設定規則。基於型樣的掃描是以自訂搜尋準則為基礎的原始碼分析。如需相關資訊，請參閱 第 209 頁的『利用基於型樣的規則自訂』。 如果要啟用基於型樣的掃描，請選取**型樣分析**勾選框。當您這麼做時，**型樣規則集**和**型樣規則**區段會變成已啟用：
   a. 如果要新增規則集，請按一下**型樣規則集**區段中的**新增**。這時會開啟「新增型樣規則集」對話框，讓您選擇一個以上的規則集。當您選取規則集時，它包含的規則會顯示在對話框的右邊，而套用規則集的專案類型則列在**專案類型**欄位中。按一下**確定**，即可新增所選取的規則集。
   b. 如果要新增規則，請按一下**型樣規則**區段中的**新增**。這時會開啟「新增型樣規則」對話框，讓您選擇一個以上的規則。您也可以按一下**建立新規則**來建立新規則（請參閱第 213 頁的『建立型樣規則』）。如果您建立新規則，它將新增至清單中且已選取。在選取或建立規則之後，請按一下**確定**，將它們新增至掃描配置。

      提示：在「新增型樣規則」對話框中，工具提示說明會指出對每一個規則所使用的表示式。

      註：
      - 當您新增規則集時，會從「新增型樣規則」對話框中過濾該規則集的規則。
      - 如果您新增了規則，然後又新增也包含該規則的規則集，則「型樣規則」區段將列出該規則，並指出它包含在規則集內。因為此規則已包含在規則集內，所以，如果您試圖移除個別規則，只會從「型樣規則」區段中移除它，而不會從掃描配置中移除。如果要從掃描配置中移除該規則，請移除規則集，或修改它，使它不包含該規則。

   c. 可使用**移除**按鈕，或按一下滑鼠右鍵選取**移除**，來移除您所新增的任何規則集或規則。使用此動作時，您也可以複選規則和規則集。

      註：如果掃描配置包括的規則或規則集後來從漏洞資料庫中移除，下次您開啟掃描配置時，會出現訊息指出規則或規則集不存在。但是，**移除**動作無法使用於這些規則或規則集，下次您儲存掃描配置時，將自動移除它們。
8. 在掃描配置中完成所有設定之後，按一下**儲存**。

修改掃描配置

程序

1. 在「掃描配置」視圖中，請選取您要修改的掃描配置。

   註：如果要共用掃描配置 - 或修改或刪除共用的掃描配置 - 您必須具有**管理共用配置**許可權。如果要瞭解設定許可權的相關資訊，請參閱《*IBM Security AppScan Source* 安裝與管理手冊》。

   註：您無法修改第 101 頁的『內建掃描配置』。
2. 修改掃描配置之後，按一下**儲存**。

移除掃描配置

程序

1. 在「掃描配置」視圖中，選取您要移除的掃描配置。

   註：您無法移除第 101 頁的『內建掃描配置』。
2. 按一下**刪除**。

共用掃描配置及使用共用的配置

關於這項作業

掃描配置可以儲存到 AppScan Source 資料庫，以便與其他人共用。如果要與其他人共用掃描配置，請按一下**共用**。

註：如果要共用掃描配置 - 或修改或刪除共用的掃描配置 - 您必須具有**管理共用配置**許可權。如果要瞭解設定許可權的相關資訊，請參閱《*IBM Security AppScan Source* 安裝與管理手冊》。

已經與其他人共用的掃描配置會出現在掃描配置清單中。

註：

- 一旦已經共用掃描配置，您無法移除共用。您可以改完成下列其中一項作業：
  - 刪除共用掃描配置。這將會在伺服器上將其刪除。
  - 複製共用掃描配置然後將其刪除。複製掃描配置將為其建立相同本端副本。
- 如果您要共用的掃描配置包含已共用的過濾器，則共用動作將完成且不出現提示。但是，如果您要共用的掃描配置包含您在本端建立的過濾器，則會出現提示，讓您知道也會共用該過濾器。如果您不想要共用本端過濾器，您可以取消掃描配置共用動作。
- 您無法透過新增本端過濾器來修改及儲存共用掃描配置。若要將這些過濾器新增至共用掃描配置，請共用過濾器，然後將它們新增至共用掃描配置。
- 如果您具有**管理共用配置**許可權，但沒有**管理共用過濾器**許可權，您無法共用包含本端過濾器的掃描配置。

將掃描配置設為預設值

關於這項作業

您可以將全部掃描配置設定為預設值 - 不論其為本端、內建、或共用。如果您將共用掃描配置設為預設值，則只會在本端進行這項設定，而不會影響其他使用者。掃描時一律會使用掃描配置。如果您設定預設掃描配置，然後刪除它，則掃描時將自動使用**正常掃描**內建掃描配置。

如果要瞭解如何使用預設掃描配置，請參閱第 93 頁的『掃描原始碼』。

程序

1. 在「掃描配置」視圖中，請選取您要設為預設值的掃描配置。
2. 按一下**選取為預設值**。

內建掃描配置

關於這項作業

AppScan Source 提供內建掃描配置。無法修改或移除這些配置。您可以在清單中選取它們來複製或檢視其設定。

掃描配置視圖

「掃描配置」視圖可讓您建立啟動掃描時可用的配置。您也可以使用該視圖來設定預設掃描配置。在掃描配置中，您可以指定掃描期間要使用的來源規則，也可以包含許多掃描設定。掃描配置中的設定通常可以獲得較佳的掃描結果，而儲存這些設定的能力更可讓掃描變得輕鬆又有效率。

「掃描配置」視圖有這些主要區段：

- 『掃描配置管理』
- 『「一般」標籤』
- 第 102 頁的『「污染流分析」標籤』
- 第 103 頁的『「型樣分析」標籤』

掃描配置管理

使用這個區段來選取、新增、移除、儲存及共用掃描配置，以及將掃描配置設為預設值。

- 如果要建立新的掃描配置，請按一下**新建**。完成掃描配置設定之後，按一下**儲存**，儲存變更。如果要將掃描配置設為預設值，請在儲存之後按一下**選取為預設值**。如果要瞭解如何使用預設掃描配置，請參閱第 93 頁的『掃描原始碼』。
- 如果要使用現有的掃描配置，請從清單中選取它：
  - 如果您要修改掃描配置設定，請按一下**儲存**以儲存變更（可以切換至不同掃描配置來捨棄不要的變更，然後按一下**捨棄**）。
  - 如果要移除選取的掃描配置，請按一下**刪除**。
  - 如果要複製掃描配置，請按一下**複製**。這樣會根據原始掃描配置的設定來建立新的掃描配置。
  - 如果要將掃描配置設為預設值，請按一下**選取為預設值**。如果要瞭解如何使用預設掃描配置，請參閱第 93 頁的『掃描原始碼』。
  - 如果要與其他人共用掃描配置，請按一下**共用**。這樣會將掃描配置儲存到 AppScan Source 資料庫。

    註：如果要共用掃描配置 - 或修改或刪除共用的掃描配置 - 您必須具有**管理共用配置**許可權。如果要瞭解設定許可權的相關資訊，請參閱《*IBM Security AppScan Source* 安裝與管理手冊》。

  註：AppScan Source 提供內建掃描配置。無法修改或移除這些配置。您可以在清單中選取它們來複製或檢視其設定。

「一般」標籤

基本資訊

這個區段可讓您命名掃描配置並提供說明。

過濾器

在此區段中，您可以選擇一個以上的過濾器，每當使用掃描配置時即將其套用至掃描。選取過濾器時，您可以選擇 AppScan Source 選擇的過濾器，或共用的過濾器，或您自己建立的過濾器。如需詳細資料，請參閱第 96 頁的『管理掃描配置』。

「污染流分析」標籤

污染流分析

啟用及設定污染流分析的範圍。

掃描規則

使用這個區段來決定掃描時採用的來源規則。

來源是對程式的輸入，例如：檔案、Servlet 要求、主控台輸入或 Socket。藉由排除部分來源規則，您可以加速掃描，並且避免不想要的輸入產生的偵測漏洞。

規則會標示著規則內容，指出與規則相關的特定漏洞、機制、屬性或技術。這些內容分組至規則集，對應於一般相關的規則集。藉由指定規則集或個別規則內容，您可以限制包含在掃描的來源規則。

- 選取掃描時要包含的一或多個漏洞類型（在規則集中依類型來組織）：
  - **全部**：如果選取此選項，全部支援來源產生的漏洞，都會被偵測到。
  - **使用者輸入**：如果選取此選項，將會偵測到一般使用者輸入產生的漏洞。
  - **Web 應用程式**：如果選取此選項，會偵測到 Web 應用程式風險產生的漏洞。
  - **錯誤處理和記載**：如果選取此選項，會偵測到錯誤處理和記載機制產生的漏洞。
  - **環境**：如果選取此選項，會偵測到配置檔、系統環境檔案和內容檔產生的漏洞。
  - **外部系統**：如果選取此選項，會偵測到外部實體產生的漏洞。
  - **資料儲存庫**：如果選取此選項，會偵測到資料儲存庫（例如資料庫和快取）產生的漏洞。
  - **不尋常事物**：如果選取此選項，會偵測到常式出現的漏洞，其通常不屬於正式作業的一部分。
  - **檔案系統**：如果選取此選項，會偵測到來自檔案系統的漏洞。
  - **機密資料**：如果選取此選項，會偵測到來自機密資料的漏洞。

  這個區段中的每一個規則集都有浮動說明。
- 選取要併入掃描中的個別掃描規則內容：按一下**捨棄選取的規則集，讓我選取個別規則內容**。這時會開啟「選取規則內容」對話框，讓您選擇個別的規則內容。如果完成這個對話框，則會捨棄任何已選取的規則集。掃描時，將會使用含有所選規則內容的掃描規則。

進階設定

這個區段僅適用於進階使用者。它包含各種可改進掃描結果的設定。這個區段中的每一個設定都有浮動說明。

「型樣分析」標籤

型樣分析

使用掃描配置時，可使用此區段來啟用基於型樣的掃描。基於型樣的掃描是以自訂搜尋準則為基礎的原始碼分析。

型樣規則集和型樣規則

使用這些區段來新增型樣分析期間要使用的規則和規則集。如需相關資訊，請參閱第 209 頁的『利用基於型樣的規則自訂』和第 96 頁的『管理掃描配置』。

Java 的漸進式分析

當啟用漸進式分析時，AppScan Source 會快取處理分析資料。之後當您重新掃描專案或應用程式時，AppScan Source 會使用資料來判斷程式碼變更，而且只重新分析變更所影響的程式碼部分。最終結果是完整的程式碼分析 - 但時間縮短了。

關於這項作業

漸進式分析在 Windows 和 Linux 上受到支援。如果啟用它，會在 AppScan Source 專案或應用程式或是 Eclipse 專案或工作區執行漸進式分析。在您啟用漸進式分析之後，您在專案、應用程式或工作區執行的第一個掃描一定是完整掃描（漏洞分析快取只有在完整掃描的期間才會更新）。這可讓 AppScan Source 快取後續掃描的資料。之後，您的專案、應用程式或工作區掃描即為漸進式掃描 - 只要不清除漏洞分析快取，而且變更的檔案數不超過您可以決定的臨界值設定。

如果要啟用和使用漸進式分析，請遵循下列步驟：

程序

1. 在文字編輯器中開啟 <data_dir>\config\scan.ozsettings（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）。在檔案中找出 incremental_analysis 設定。這項設定看起來如下：

   ```
   <Setting
     name="incremental_analysis"
        read_only="false"
     default_value="false"
     description="Attempt to scan only changed files,
       instead of re-scanning everything."
     type="bool"
     value="false"
     display_name="Incremental Analysis"
     hidden="true"
   />
   ```

   在這項設定中，修改 value 屬性。如果該屬性設為 true，這項設定即會開啟。如果設為 false，在掃描時，AppScan Source 不會執行漸進式分析。

2. 在 <data_dir>\config\scan.ozsettings 中，找到 percentage_of_files_changed 設定：

   ```
   <Setting
     name="percentage_of_files_changed"
        read_only="false"
     default_value="50"
     description="In incremental scanning, if percentage of files
     being changed since last scan exceeds the threshold, full
   ```

```
  scan will be initiated. The percentage ranges from 0 to 100.
  Default threshold is 50, which represents 50%."
  type="int"
  value="50"
  display_name="Percentage of files being changed"
  hidden="true"
/>
```

這個設定可讓您指定變更的檔案百分比，超過此百分比時將會起始完整掃描。依預設，此臨界值百分比為 50% - 這表示，如果在專案、應用程式或工作區中變更 50% 以上的檔案之後重新掃描，將會起始完整掃描，而不是漸進式分析掃描。在這個設定中，依需要將 `value` 屬性變更為您偏好的臨界值百分比。

3. 修改所有相關的設定之後，儲存 `<data_dir>\config\scan.ozsettings`，然後啟動或重新啟動支援漸進式分析的 AppScan Source 產品。例如，重新啟動 AppScan Source for Analysis、AppScan Source for Development Eclipse 外掛程式 或 AppScan Source 指令行介面 (CLI) - 或重新啟動 AppScan Source for Automation 服務。
4. 現在，當您重新掃描具有相同掃描配置的 Java 應用程式或專案時，如果變更的數目未超過臨界值而且未清除漏洞分析快取，則會執行漸進式分析。
5. **清除漏洞分析快取：**如果漸進式掃描有問題，或是在啟用漸進式分析時您想要執行完整分析，請先清除漏洞快取再重新掃描：
   - AppScan Source for Analysis:
     a. 開啟 AppScan Source 專案的「內容」視圖。如果是掃描應用程式，請開啟任何子專案的內容視圖（刪除專案的快取也會刪除其應用程式的快取）。
     b. 在「概觀」標籤中，按一下**清除快取**。
   - AppScan Source for Development Eclipse 外掛程式：刪除 `<data_dir>\temp\<workspace>\<project>`，其中：
     – `<data_dir>` 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述.
     – `<workspace>` 是掃描所在的 Eclipse 工作區名稱。如果要刪除整個工作區的快取，請刪除整個 `<data_dir>\temp\<workspace>` 目錄。
     – `<project>` 是掃描的 Eclipse 專案名稱。如果要刪除專案的快取，請刪除 `<data_dir>\temp\<workspace>\<project>` 目錄。
   - AppScan Source 指令行介面 (CLI)：使用 `clearcache` 指令，如 *IBM Security AppScan Source Utilities* 使用手冊 中的說明。
   - AppScan Source for Automation：使用 `ScanApplication` 指令 `-clearcache` 引數，如 *IBM Security AppScan Source Utilities* 使用手冊 中的說明。

結果

在 AppScan Source for Analysis 中掃描之後，您可以使用**評量差異**特性來比較程式碼變更前和變更後的評量。

提示：

- 如果要強制執行完整分析掃描，請停用漸進式分析或清除漏洞分析快取。
- 執行漸進式分析時，應在進行下列任何修改之後執行完整分析掃描：
  – 適用於專案或應用程式的安全規則變更或自訂規則變更。
  – 掃描配置變更。

  - 影響掃描的 .ozsettings 檔案變更。
  - 應用程式或專案內容的變更。例如，您在「所有應用程式」或選取的應用程式或專案的 AppScan Source for Analysis「內容」視圖中所做的變更。
  - 將新專案新增至應用程式或刪除現有的專案。
  - 從掃描中排除檔案。例如，在 AppScan Source for Analysis 中，您可以選擇從掃描中排除檔案，方法為在「瀏覽器」視圖中用滑鼠右鍵按一下它，然後選擇**從掃描中排除**。
- 您可以在 http://www.ibm.com/support/docview.wss?uid=swg21994390 找到關於漸進式分析的現行資訊。

註：

- 在漸進式掃描之後，編輯器中的發現項目標記可能再也不會在正確的位置。
- 沒有記錄的重新修補發現項目可能會出現在漸進式分析結果中。
- 在漸進式分析期間，您無法同時開啟多個 AppScan Source 產品或元件。此外，當您正在掃描時，同時間在同一機器上的另一個使用者無法掃描相同的應用程式或專案。

從掃描中排除檔案

開始之前

註：如果您的應用程式是一個 Eclipse 工作區，則無法從掃描中排除檔案。

程序

1. 在「瀏覽器」視圖中，選取要從掃描中排除的檔案。
2. 用滑鼠右鍵按一下選項，然後從功能表中，選擇**從掃描中排除**。

結果

當內含檔案的專案其「內容」視圖開啟時，視圖的**來源**標籤會列出專案當中的檔案，包括已排除的檔案。

原始碼根目錄圖示下會列出專案檔。遭排除在掃描之外的檔案有紅色檔案圖示（如果用滑鼠右鍵按一下遭排除的檔案，則其功能表已停用**排除**且已啟用**併入**）。如果要排除已併入的檔案，請用滑鼠右鍵按一下它，然後在功能表中選擇**排除**。如果要併入已排除的檔案，請用滑鼠右鍵按一下它，然後在功能表中選擇**併入**。

取消或停止掃描

雖然您可以取消進行中的掃描，但取消掃描會使這項掃描的資料全部消失。或者，您也可以停止掃描來中止它，並以到目前為止所發現的結果來產生評量。

在 AppScan Source for Analysis 中取消或停止掃描

如果要取消目前進行中的掃描，請從主功能表中選取**掃描** > **取消掃描**或**掃描** > **停止掃描**。

取消掃描會終止掃描，不產生任何結果。**停止掃描**會中止掃描，並以目前為止所找到的結果來產生評量。

在 AppScan Source for Development（Eclipse 外掛程式）中取消或停止掃描

當掃描在執行中：

- 如果要取消掃描，請從主功能表中選擇**安全分析** > **掃描** > **取消掃描**。這時會終止掃描，且不產生任何結果，在 Eclipse 主控台中會出現取消診斷訊息。
- 如果要停止掃描，請從主功能表中選擇**安全分析** > **掃描** > **停止掃描**。這時掃描會終止，並產生到起始停止動作為止所收集之結果的評量。

註：AppScan Source for Development（Eclipse 外掛程式）只在 Windows 和 Linux 上受到支援。

在 AppScan Source for Development（Microsoft Visual Studio 外掛程式）中取消掃描

當掃描在執行中，請從主功能表中，選擇 **IBM Security AppScan Source** > **掃描** > **取消掃描**。這時會終止掃描，且不產生任何結果。

註：AppScan Source for Development Microsoft Visual Studio 外掛程式只在 Windows 上受到支援。

在 Linux 上必備的 AppScan Source for Analysis 和 AppScan Source for Development（Eclipse 外掛程式）元件

在 Linux 上，Eclipse 需要安裝協力廠商元件，才能呈現瀏覽器型的內容。如果沒有這個元件，AppScan Source for Analysis 和 AppScan Source for Development Eclipse 外掛程式可能會出現一些症狀，例如在登入之後出現懸置或在產品使用期間出現失敗。

這個必要條件的相關資訊，位置如下：http://www.eclipse.org/swt/faq.php#browserwebkitgtk。

- 『在 Linux 上啟用 AppScan Source for Analysis 的瀏覽器型內容』
- 第 107 頁的『在 Linux 上啟用安裝於 Eclipse 3.7 版或更新版本的 AppScan Source for Development 的瀏覽器型內容』

在 Linux 上啟用 AppScan Source for Analysis 的瀏覽器型內容

AppScan Source for Analysis 以 Eclipse 為建置基礎，因此會受這個問題影響。

更正這個情況的建議方式是，確定已安裝 32 位元或 i686 版本的 WebKitGTK 1.2.0 或更新的版本。您應該向系統管理者洽詢安裝套件的適當方式，但在某些系統上，有可能如同發出 `yum install webkitgtk.i686` 一樣簡單。

如果您無法安裝 WebKitGTK，可以選擇安裝 32 位元版本的 Mozilla XULRunner 1.8。當使用這個選項時，您也可能需要對環境變數進行這些更新：

- 將 `MOZILLA_FIVE_HOME` 設為 XULRunner 安裝位置。
- 更新 `LD_LIBRARY_PATH`，將 `$MOZILLA_FIVE_HOME` 附加到後面（或前面）

在 Linux 上啟用安裝於 Eclipse 3.7 版或更新版本的 AppScan Source for Development 的瀏覽器型內容

更正這個情況的建議方式是，確定已安裝 32 位元或 i686 版本的 WebKitGTK 1.2.0 或更新的版本。您應該向系統管理者洽詢安裝套件的適當方式，但在某些系統上，有可能如同發出 `yum install webkitgtk.i686` 一樣簡單。

如果您無法安裝 WebKitGTK，可以選擇安裝 32 位元版本的 Mozilla XULRunner 1.8。當使用這個選項時，您也可能需要對環境變數進行這些更新：

- 將 `MOZILLA_FIVE_HOME` 設為 XULRunner 安裝位置。
- 更新 `LD_LIBRARY_PATH`，將 `$MOZILLA_FIVE_HOME` 附加到後面（或前面）

管理我的評量

「我的評量」視圖包含一份評量清單（目前開啟的評量，以及您已儲存的任何評量）。在這個視圖中，您可以開啟、刪除、儲存、重新命名或比較評量。掃描完成之後，或當您開啟儲存的評量時，評量會出現在「我的評量」視圖中。「我的評量」會顯示一份表格，列出已開啟或儲存的評量，且會識別已發佈或修改的評量。從這個視圖中移除某項評量（且未儲存或發佈它），就會永久刪除這個評量。

如需「我的評量」視圖的相關資訊，請參閱第 262 頁的『「我的評量」視圖』。

限制：當您掃描多個應用程式或專案時，在「我的評量」視圖中，會建立一個母節點來包含每個掃描項目的評量。在這種情況下，無法管理個別子項評量（例如，無法個別移除或發佈子項評量）。當同時掃描多個應用程式或專案時，您只能以群組（母節點）的方式來管理各個評量。

提示：您每次只能開啟與單一應用程式相關的掃描結果。如果要檢視多重應用程式或多重專案的掃描結果，您必須在「我的評量」視圖中展開樹狀結構，然後按兩下您想要開啟的評量。

提交 AppScan Source 評量至 Cloud 進行分析

如果您在 IBM Cloud Marketplace 訂閱 IBM Application Security on Cloud，或訂閱 Application Security on Cloud for Bluemix，您可以在該處提交 AppScan Source 評量以進行分析。支援來自 AppScan Source 9.0 版或更新版本的評量 - 您可以提交的掃描數目視您的 Application Security on Cloud 訂閱而定。

關於這項作業

當您使用 Application Security on Cloud 服務的 靜態分析特性時，您可以產生使用「智慧型發現項目分析 (IFA)」的安全分析報告。IFA 是功能強大的機器學習技術，此技術會過濾誤判，以及修正一個程式碼位置就能進行補救的方式來分組發現項目，而替您執行許多的分類工作。如果要進一步瞭解 IFA，請參閱這篇文章。.

如果您使用 AppScan Source 9.0 版或更新版本，而且有 Application Security on Cloud 訂閱，您可以將 AppScan Source 評量上傳到 Application Security on Cloud 來獲得這項技術的好處。然後，你會收到由此技術自動分類的新評量。這個評量可能是 HTML 報告或是可以在 AppScan Source 產品中開啟的評量。

如果您有 Application Security on Cloud 訂閱，可能會有每月掃描數的限制。如需掃描和並行掃描授權的相關資訊，請參閱http://www.ibm.com/support/knowledgecenter/SSYJJF_1.0.0/ApplicationSecurityonCloud/src_managing_assessments_cloud.html。

註：如果您使用免費試用的 Application Security on Cloud 來掃描 AppScan Source 評量，除了 IFA 所分類的 AppScan Source 評量檔之外，您還可以下載完整的 HTML 報告。對於所有其他掃描類型，如果是免費試用，則只能下載摘要報告。

程序

1. **如果已經在使用 Application Security on Cloud 進行 靜態分析，請跳過此步驟：**
   a. 如果沒有 Application Security on Cloud 訂閱，您可以遵循下列方式來獲得：
      - **IBM Cloud Marketplace：**移至 https://appscan.ibmcloud.com/serviceui/home 並使用 IBM ID 登入。如果沒有 IBM ID，請使用鏈結來建立 ID。然後使用位在服務的鏈結來註冊免費試用或付費訂閱。
      - **IBM Bluemix®：**移至 https://console.ng.bluemix.net/ 並使用**註冊**按鈕，在 Bluemix 完成註冊表單。然後建立 Application Security on Cloud for Bluemix 服務實例。

   b. **僅限 IBM Cloud Marketplace：**在 Application Security on Cloud 服務中，建立應用程式（請參閱 http://www.ibm.com/support/knowledgecenter/SSYJJF_1.0.0/ApplicationSecurityonCloud/ent_create_application.html），然後按一下**建立掃描**。
   c. 在**今天掃描的應用程式類型？**畫面中，選取**桌面**或 **Web** > **靜態**。
   d. 如果您先前未下載和設定 Static Analyzer 用戶端公用程式，請現在進行。如需相關資訊，請參閱 http://www.ibm.com/support/knowledgecenter/SSYJJF_1.0.0/ApplicationSecurityonCloud/src_utility_install.html。
2. 在 AppScan Source 產品或您選擇的工具中產生評量（.ozasmt 檔）。支援 9.0 版或更新版本。
3. 使用 用戶端公用程式 指令行介面 (CLI) 來產生評量的 中間表示法（IRX 或 .irx）檔（.ozasmt 檔）：
   a. 在解壓縮 用戶端公用程式 到本端磁碟機之後，將其 \bin 目錄的位置新增到 PATH 環境變數。如果您沒有這麼做，則每次發出此指令時，都需要使用 \bin 目錄來限定所有的 用戶端公用程式 CLI 指令。如需相關資訊，請參閱 http://www.ibm.com/support/knowledgecenter/SSYJJF_1.0.0/ApplicationSecurityonCloud/src_irx_gen_cli.html。
   b. 在 Windows 上發出此指令：

      ```
      appscan package -d <save_path> -f <assessment_file> -n <file_name>
      ```

      或在 Linux 上，發出此指令：

      ```
      appscan.sh package -d <save_path> -f <assessment_file> -n <file_name>
      ```

      指令引數是選用的：

      - -d：指定 -d <save_path>，其中 <save_path> 是您想要儲存 IRX 檔案的目錄。

- `-f`：指定 `-f <assessment_file>`，其中 `<assessment_file>` 是您要包裝以進行掃描的 `.ozasmt` 檔案。如果 `<assessment_file>` 檔案不在現行目錄中，請使用此選項來指定評量檔案路徑及檔名。

  註：只有在符合下列其中一個或兩個陳述式時，才需要此選項：

  – 您是從包含多個評量檔的目錄中發出指令。如果目錄只包含一個評量檔，則未使用 `-f` 選項時，將會包裝該檔案。
  – 您是從未包含評量檔的目錄中發出指令。在此情況下，必須使用 `-f` 選項來指定要包裝的評量檔的路徑及檔名。
- `-n`：指定 `-n <file_name>`，其中 `<file_name>` 是 IRX 檔案名稱。指定檔名時不一定要有 `.irx` 副檔名。如果您指定沒有副檔名的檔名，則在產生檔案時會為您自動新增副檔名。

有關 `package` 指令的其他相關資訊（包括使用範例），可以在配置指令 (Windows) 或配置指令 (Linux) 找到。

4. 使用 CLI `queue_analysis` 指令來上傳 IRX 檔案
   a. 從 CLI 登入服務。在 IBM Cloud Marketplace 和 IBM Bluemix，執行此動作的方法不同。在 CLI 對服務進行鑑別的相關詳細資訊，可以在鑑別指令 (Windows) 或鑑別指令 (Linux) 找到。
      - **IBM Cloud Marketplace:**

        在 Windows 上發出此指令：

        ```
        appscan scx_login -P <password> -u <user_name> -persist
        ```

        或在 Linux 上，發出此指令：

        ```
        appscan.sh scx_login -P <password> -u <user_name> -persist
        ```

        這些引數是必要的：

        – `-P`：指定 `-P <password>`，其中 `<password>` 是登錄 Application Security on Cloud 服務時指定的密碼。
        – `-u`：指定 `-u <user_name>`，其中 `<user_name>` 是登錄 Application Security on Cloud 服務時指定的電子郵件位址。

        此引數是選用的：

        – `-persist`：在登入記號檔到期時，自動嘗試重新接受服務的鑑別。
      - **IBM Bluemix:**

        在 Windows 上發出此指令：

        ```
        appscan login -P <password> -u <user_name> -persist
        ```

        或在 Linux 上，發出此指令：

        ```
        appscan.sh login -P <password> -u <user_name> -persist
        ```

        這些引數是必要的：

        – `-P`：指定 `-P <password>`，其中 `<password>` 是服務認證中所指定的密碼。
        – `-u`：指定 `-u <user_name>`，其中 `<user_name>` 是服務認證中所指定的連結 ID。

如果要判斷 Bluemix 服務認證，請在服務「儀表板」的左導覽窗格中選取**服務認證**。請參閱啟用外部應用程式使用 Bluemix 服務。

此引數是選用的：

– `-persist`：在登入記號檔到期時，自動嘗試重新接受服務的鑑別。

b. 使用 `queue_analysis` 指令來上傳 IRX 檔案：

- 在 Windows 上發出此指令：

```
appscan queue_analysis -a <app_id> -f <irx_file> -n <scan_name>
```

或在 Linux 上，發出此指令：

```
appscan.sh queue_analysis -a <app_id> -f <irx_file> -n <scan_name>
```

這些引數是必要的：

– `-f`：指定 `-f <irx_file>`，其中 `<irx_file>` 是您要提交以進行掃描的 IRX 檔案。如果 IRX 檔案不在現行目錄中，請使用此選項來指定 IRX 檔案路徑及檔名。

註：只有在符合下列其中一個或兩個陳述式時，才需要此選項：

- 您是從包含多個 IRX 檔案的目錄中發出指令。如果目錄只包含一個 IRX 檔案，則未使用 `-f` 選項時，將會提交該檔案。
- 您是從未包含 IRX 檔案的目錄中發出指令。在此情況下，必須使用 `-f` 選項來指定要提交的 IRX 檔案的路徑及檔名。

– `-n`：指定 `-n <scan_name>`，其中 `<scan_name>` 是在雲端進行的掃描名稱。

– `-a` **（限 IBM Cloud Marketplace）**：如果您連接至位於 IBM Cloud Marketplace 的 Application Security on Cloud 服務，您提交至雲端的 IRX 檔案必須與現有的 Application Security on Cloud 應用程式相關聯。請使用這個選項指定 `-a <app_id>`，其中 `<app_id>` 是要產生關聯之應用程式的 ID。如果要判斷 ID，請使用 `list_apps` 指令。

- 當 `queue_analysis` 指令完成時，會顯示分析工作的 ID。如果您要使用 CLI 來接收 Application Security on Cloud 分析報告，會需要在 `get_result` 指令中併入此工作 ID - 而且您應記下此 ID。如果您使用 CLI 來接收分析報告，您可以選擇接收包含 `.ozasmt` 檔的保存 (`.zip`) 檔，讓分析報告可以在 AppScan Source 中開啟。如果只有興趣查看 HTML 報告，您可以使用 CLI 或 Application Security on Cloud Web 用戶端來下載報告。

有關使用 `queue_analysis` 指令的詳細資料可以在分析指令 (Windows) 或分析指令 (Linux) 找到。

5. 如果您使用 CLI 來上傳 IRX - 或在 Application Security on Cloud Web 用戶端中選取**掃描完成時以電子郵件通知我**勾選框，當掃描完成時，您會收到電子郵件。
6. 選擇擷取分析報告的方法。您可以使用 CLI `get_result` 指令或使用 Application Security on Cloud Web 用戶端。如果您使用 CLI 來接收分析報告，您可以選擇接收包含 `.ozasmt` 檔的保存 (`.zip`) 檔，讓分析報告可以在 AppScan Source 中開啟。如果只有興趣查看 HTML 報告，您可以使用 CLI 或 Application Security on Cloud Web 用戶端來下載報告。
7. **如果您要使用 CLI `get_result` 指令來擷取分析報告，請完成此步驟：**

a. 請確定您已從 CLI 登入服務。

b. 在 Windows 上發出此指令：

```
appscan get_result -d <file_path> -i <job_id> -t <type>
```

或在 Linux 上，發出此指令：

```
appscan.sh get_result -d <file_path> -i <job_id> -t <type>
```

此引數是必要的：

- `-i`：指定 `-i <job_id>`，其中 `<job_id>` 是分析工作的 ID。

註：如果在發出 `queue_analysis` 指令時未記下 ID，您可以使用 `appscan list` 或 `appscan.sh list` 指令來查看所有分析工作的清單。如需相關資訊，請參閱分析指令 (Windows) 或分析指令 (Linux)。

這些引數是選用的：

- `-d`：指定 `-d <file_path>`，其中 `<file_path>` 是目的地檔案的完整路徑及（或）目的地檔案的檔名。如果未指定檔名，則檔名將根據掃描工作名稱。如果未指定路徑，則會將檔案儲存至現行目錄。如果未包括此選項，則會使用根據掃描工作名稱的檔名，將檔案儲存至現行目錄。
- `-t`：指定 `-t <type>`，其中 `<type>` 是 `html` 或 `zip`。結果會儲存為 HTML 檔或包含 HTML 結果的 `.zip` 檔案。如果未包括此選項，則會將結果儲存為 HTML 檔案。

  如果掃描結果適用於 `package` 指令所產生的 IRX 檔案，則指定 `-t zip` 會儲存包含新的 `.ozasmt` 檔案（可載入至 AppScan Source 9.0 版或更新版本產品中）的結果。

有關使用 `get_result` 指令的詳細資料可以在結果指令 (Windows) 或結果指令 (Linux) 找到。

8. **如果您要使用 Web 用戶端來擷取分析報告，請完成此步驟：**如果只有興趣查看 HTML 報告，您可以使用 Application Security on Cloud Web 用戶端來下載報告。

   當您登入服務時，應該會自動看到您的掃描清單（如果您已導覽至服務的另一個區段，請按一下右上方的 **X** 圖示以回到掃描清單）。在掃描清單中，找到掃描並選取**下載**圖示，然後選擇 XML 或 HTML 格式。

   如果要在 IBM Cloud Marketplace 進一步瞭解 Application Security on Cloud 掃描結果，請參閱http://www.ibm.com/support/knowledgecenter/en/SSYJJF_1.0.0/ApplicationSecurityonCloud/appseccloud_results_dashboard_cm.html。在 IBM Bluemix，請參閱 https://console.ng.bluemix.net/docs/services/ApplicationSecurityonCloud/appseccloud_results.html#results。

發佈評量

AppScan Source 提供兩個發佈選項。您可以將評量發佈到 AppScan Source 資料庫，以便儲存及共用評量。或者，如果 AppScan Enterprise Server 已安裝 Enterprise Console 選項，您可以將評量發佈到那裡。 AppScan Enterprise Console 提供各種可處理評量的工具，例如：報告特性、問題管理、趨勢分析和儀表板。

如要進一步瞭解 AppScan Source 發佈特性，請參閱『將評量發佈到 AppScan Source』和第 114 頁的『將評量發佈到 AppScan Enterprise Console』。

註：針對某些版本的 AppScan Source和 AppScan Enterprise，兩個產品的版本和版次層次必須相符，才能從 AppScan Source 發佈到 AppScan Enterprise Console。請參閱http://www.ibm.com/support/docview.wss?uid=swg21975211以瞭解在發佈評量時，哪些版本的 AppScan Source 和 AppScan Enterprise 相容。

登錄應用程式和專案，以發佈至 AppScan Source

您必須先登錄經過掃描以建立評量的應用程式或專案，才能將評量發佈至 AppScan Source 資料庫。依預設，如果您試圖發佈未登錄的應用程式或專案的評量，當時系統會提示您登錄應用程式或專案。如果**在起始發佈之時，自動登錄應用程式**這項**一般**喜好設定是**一律登錄**，AppScan Source for Analysis 會替您自動登錄。

重要：您必須具備**登錄**許可權，才能登錄應用程式和專案。

在掃描之前，如果要登錄應用程式和專案，請在「瀏覽器」視圖中選取應用程式或專案，然後從主工作台功能表中，選取**檔案** > **登錄**。在「瀏覽器」視圖中，當您用滑鼠右鍵按一下所選的項目時，也能夠使用**登錄應用程式**和**登錄專案**動作。

如果已登錄應用程式，您可以用新名稱來重新登錄它。如果要執行這個動作，請選取它，用滑鼠右鍵按一下它，然後從功能表中選擇**應用程式登錄為**。在「重新命名」對話框中，輸入已登錄之應用程式或專案的新名稱。

如果要取消登錄應用程式和專案，請在「瀏覽器」視圖中選取應用程式或專案，然後從主工作台功能表中，選取**檔案** > **取消登錄**。在「瀏覽器」視圖中，當您用滑鼠右鍵按一下所選的項目時，也能夠使用**取消登錄應用程式**和**取消登錄專案**動作。

註：將項目取消登錄不會從 AppScan Source 資料庫中移除任何已發佈的資料。

將評量發佈到 AppScan Source

您可以將評量發佈到 AppScan Source 資料庫，以便儲存及共用評量。

關於這項作業

您必須先向 AppScan Source 登錄應用程式和專案，然後才能發佈它們的評量。如需相關資訊，請參閱『登錄應用程式和專案，以發佈至 AppScan Source』。依預設，如果您試圖發佈未登錄的應用程式或專案的評量，當時系統會提示您登錄應用程式或專案（需要**登錄**許可權）。

註：因掃描個別檔案而建立的評量無法發佈。

限制：當您掃描多個應用程式或專案時，在「我的評量」視圖中，會建立一個母節點來包含每個掃描項目的評量。在這種情況下，無法管理個別子項評量（例如，無法個別移除或發佈子項評量）。當同時掃描多個應用程式或專案時，您只能以群組（母節點）的方式來管理各個評量。

程序

1. 如果要發佈「分類」視景中目前開啟的評量，請從主工作台功能表中，選取**檔案** > **將評量發佈到 AppScan Source**。
2. 如果要在「我的評量」視圖中發佈評量，請選取它，然後按一下視圖 **將評量發佈到 AppScan Source** 按鈕，或是在評量上按一下滑鼠右鍵，然後選取**將評量發佈到 AppScan Source**。

結果

當儲存評量時，AppScan Source for Analysis 會將絕對路徑寫入評量檔中，以參照原始檔之類的項目。這些絕對路徑有可能導致難以共用另一部電腦上目錄結構不同的檔案。如果要能夠建立可攜式評量檔，您應該建立一個變數（請參閱第 84 頁的『定義變數』或第 120 頁的『在發佈和儲存時定義變數』）。

在發佈之後，「我的評量」視圖中所列出評量的**已發佈**直欄中將會有一個圖示。此外，該評量將會出現在「已發佈的評量」視圖中，這是過濾器所驅動，已發佈至 AppScan Source 資料庫之評量的視圖。您可以將這個視圖設為只顯示符合過濾準則的評量。比方說，如果發佈了 1,000 個評量，但您只想檢視自己發佈的評量，您可以利用**依發佈者**作為準則，**現行使用者**或您的使用者名稱作為值，來建立一個過濾器。

在「已發佈的評量」視圖中設定過濾器

過濾器可用來限制顯示在「已發佈的評量」視圖中的評量數目。

程序

1. 在「已發佈的評量」視圖上，按一下工具列**設定過濾器**按鈕。
2. 選取您要的過濾準則的一或多個勾選框：
   - **依應用程式**：選取您要顯示其評量的應用程式。如果指定的應用程式是評量的一部分，就會出現針對多個應用程式所產生的評量。
   - **依發佈者**：設定此視圖以顯示現行使用者已發佈的評量，或指定您要顯示其已發佈評量的使用者。
   - **依日期近似性**：指定相對於現行日期的日期範圍，單位是時數、天數、週數、月數或年數。您可以選取**依日期近似性**或**依日期範圍**，但是不能同時選取兩者。
   - **依日期範圍**：指定要顯示在視圖中的一個範圍的評量日期。您可以選取**依日期近似性**或**依日期範圍**，但是不能同時選取兩者。
3. 按一下**確定**來設定過濾器。

結果

在套用過濾準則之後，請按一下**重新整理過濾器**，根據上次套用過濾器之後所新增或移除的任何評量來重新整理視圖。按一下**清除過濾器**，可移除任何現有的過濾器來顯示所有評量。

從 AppScan Source 中刪除已發佈的評量

如果您已將評量發佈到 AppScan Source，您可以利用「已發佈的評量」視圖中的動作來移除它們。

程序

1. 在「已發佈的評量」視圖中，選取您想要刪除的評量。您也可以利用鍵盤的 Ctrl 或 Shift 鍵來選取多項評量。
2. 在視圖工具列中選取**刪除評量**按鈕，或用滑鼠右鍵按一下選項，然後從功能表中選取**刪除評量**。

將評量發佈到 AppScan Enterprise Console

如果 AppScan Enterprise Server 已安裝 Enterprise Console 選項，您可以將評量發佈到那裡。 Enterprise Console 提供各種可處理評量的工具，例如：報告特性、問題管理、趨勢分析和儀表板。

關於這項作業

發佈評量到 Enterprise Console之前，您必須在 AppScan Enterprise Console 配置伺服器，設定喜好設定頁面。如需設定喜好設定的相關資訊，請參閱 第 81 頁的『AppScan Enterprise Console 喜好設定』。

註：針對某些版本的 AppScan Source和 AppScan Enterprise，兩個產品的版本和版次層次必須相符，才能從 AppScan Source 發佈到 AppScan Enterprise Console。請參閱http://www.ibm.com/support/docview.wss?uid=swg21975211以瞭解在發佈評量時，哪些版本的 AppScan Source 和 AppScan Enterprise 相容。

限制：當您掃描多個應用程式或專案時，在「我的評量」視圖中，會建立一個母節點來包含每個掃描項目的評量。在這種情況下，無法管理個別子項評量（例如，無法個別移除或發佈子項評量）。當同時掃描多個應用程式或專案時，您只能以群組（母節點）的方式來管理各個評量。

程序

1. 您可以利用下列其中一種方法，將一或多項評量發佈到 Enterprise Console：
   a. 在「我的評量」視圖中，選取一或多項評量，然後按一下**將評量發佈到 AppScan Enterprise Console**。
   b. 在「我的評量」視圖中，用滑鼠右鍵按一下評量（或您所選的多項評量），然後選取**將評量發佈到 AppScan Enterprise Console** 功能表項目。
   c. 當評量開啟時，從主功能表中，選擇**檔案** > **將評量發佈到 AppScan Enterprise Console**。
2. 在「發佈到 AppScan Enterprise Console」對話框中，執行下列動作：
   a. 指定要與評量產生關聯的 AppScan Enterprise Console 應用程式。當連接至 AppScan Enterprise Server 9.0.3 版和更新的版本時，這個選項是必要的（除非您如這裡所述的停用此需求）。當連接至舊版 AppScan Enterprise Server 時，與應用程式產生關聯是選用的。如果您連接的是 AppScan Enterprise Server 舊版，依預設，會將該應用程式設定為前一個指定要發佈的應用程式。如果先前在發佈時未指定應用程式，則依預設不會使用應用程式。如果要指定應用程式，請執行下列動作：

1) 按一下**應用程式**欄位的**選取**按鈕。
2) 這時會開啟「選取應用程式」對話框，其中顯示已存在於 AppScan Enterprise Console 中的所有應用程式。如果要在 AppScan Enterprise Console 中檢視應用程式的屬性，請按一下旁邊的**檢視設定檔**。
3) 選取要與掃描產生關聯的應用程式，或為了此目的而按一下**建立新的應用程式**，以建立新的應用程式。按一下此鏈結會開啟 AppScan Enterprise Console，可讓您建立新的應用程式。儲存新應用程式的屬性之後，「選取應用程式」對話框會自動重新整理，以納入此應用程式供您選擇（如果未自動納入新的應用程式，請按一下**重新整理**）。

   提示：在「選取應用程式」對話框中，您可以使用過濾器欄位來縮短應用程式清單。當您輸入時，過濾器會自動套用至應用程式清單。星號 (*) 和問號 (?) 字元可用來作為萬用字元。星號符合零或多個字元所組成的任意群組，問號則符合任何單一字元。

4) 選取應用程式之後，按一下**確定**。

b. 必要性的： 在**名稱**欄位中，指定在 AppScan Enterprise Console 中用來儲存評量的名稱。

c. 選擇性的： **當連接至 9.0.3 版之前的 AppScan Enterprise Server 版本時：** 使用**資料夾**欄位，來設定發佈時的目標位置。依預設，會將位置設定在前次使用發佈的目標位置。如果之前沒有已發佈的評量，則會選取您的預設 AppScan Enterprise Console 資料夾（請注意此為 AppScan Enterprise Console 喜好設定頁面中針對使用者 ID 指定的預設資料夾）。如果要選擇發佈到不同的資料夾，請按一下**資料夾**欄位的**選取**按鈕，然後選擇您要的資料夾（只會出現您有權發佈至的資料夾）。如果您要的發佈目標資料夾無法使用，請按一下**重新整理**以 依照伺服器上已作的任何變更，來更新資料夾樹狀結構。

3. 按一下**發佈**。

結果

當儲存評量時，AppScan Source for Analysis 會將絕對路徑寫入評量檔中，以參照原始檔之類的項目。這些絕對路徑有可能導致難以共用另一部電腦上目錄結構不同的檔案。如果要能夠建立可攜式評量檔，您應該建立一個變數（請參閱第 84 頁的『定義變數』或第 120 頁的『在發佈和儲存時定義變數』）。

發佈評量之後，在參考訊息中會有一個 AppScan Enterprise (Enterprise Console) 鏈結。按一下這個鏈結，您的預設外部 Web 瀏覽器會開啟入口網站頁面。

提示：如果發佈失敗，請檢查 Enterprise Console 伺服器在執行中，且您能夠在瀏覽器中存取它的控制中心 URL（請使用同一個指定的 **Enterprise Console URL**，其指定位置在 AppScan Enterprise Console 喜好設定中）。

註：

- 大型評量可能需要花較長時間，才會出現在入口網站中。如果發佈之後未收到任何錯誤訊息，且報告未出現在入口網站中，請洽詢管理者。
- 當試圖發佈評量時，只要與 Enterprise Console 目前在處理的評量同名，就會失敗。另外，如果在處理第一個評量之後，您又發佈一項通用名稱的評量，第二個評量會改寫第一個（如果事先配置的話，Enterprise Console 可以提供通用名稱報告的趨勢

分析）。如果要判定評量是否處理完成，請在 Web 瀏覽器中存取 Enterprise Console 控制中心，然後導覽至適當的使用者資料夾，檢查報告的狀態。

- AppScan Source 不支援發佈至已配置為使用 Proxy 設定的 Enterprise Console 實例。試圖發佈至使用 Proxy 設定的實例將會導致錯誤。

重要：

升級至 AppScan Source 9.0.3.4 版時，您將會注意到這些變更：

- 現在，當您將評量發佈到 AppScan Enterprise Console 時，必須將該評量與 AppScan Enterprise 中的應用程式產生關聯（如果您是在執行 AppScan Enterprise Server 9.0.3 版以及更新版本）。因此，如果自動化 Script 不包含應用程式關聯，則它們可能會失敗。在 AppScan Enterprise Server 中，如果您要利用 AppScan Enterprise Server 應用程式安全風險管理功能，必須有應用程式關聯。請參閱http://www.ibm.com/support/knowledgecenter/SSW2NF_9.0.3/com.ibm.ase.help.doc/topics/c_overview.html。
- 此外，您必須從 AppScan Enterprise URL 移除此埠。
  1. 在 AppScan Source for Analysis 中，按一下**編輯** > **喜好設定**。
  2. 在 AppScan Enterprise 主控台設定中，從**企業主控台 URL** 欄位移除此埠。
- 在發佈評量之後，它只會在 AppScan Enterprise 的「監視」視圖中提供使用（在舊版中，該評量在 AppScan Enterprise 的「掃描」視圖中提供使用）。 http://www.ibm.com/support/knowledgecenter/SSW2NF_9.0.3/com.ibm.ase.help.doc/topics/t_workflow_for_applications.html中會對移轉到此視圖加以說明。

這是在使用「共用存取卡 (CAC)」鑑別時發佈到 AppScan Enterprise Server 所需的 AppScan Source 與 AppScan Enterprise Server 之間變更的通訊協定的結果。

如果您要在已啟用 CAC 鑑別時將評量發佈到 AppScan Enterprise Server，或是您不要利用 Enterprise Server 應用程式安全風險管理功能，則可以回復到前一個通訊協定，如下：

1. 開啟 <data_dir>\config\ounce.ozsettings（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）。
2. 在這個檔案中，請尋找這項設定：

```
<Setting
  name="force_ase902_assessment_publish"
  value="false"
  default_value="false"
  description="Use ASE 9.0.2-style assessment publish"
  display_name="Use ASE 9.0.2-style assessment publish"
  type="boolean"
  read_only="true"
  hidden="true"
/>
```

3. 在這項設定中，將 value="false" 變更為 value="true"，然後儲存檔案。
4. 重新啟動您將從中發佈評量的 AppScan Source 產品。

當此設定設定為 value="true" 時：

- 如果您在發佈時將評量與 AppScan Enterprise 中的應用程式產生關聯，則該評量會在「監視」及「掃描」視圖中提供使用。

- 如果您在發佈時不將評量與應用程式產生關聯，該評量將會在「掃描」視圖中提供使用。
- 當已啟用 CAC 時，您將無法將評量發佈至 AppScan Enterprise Server。

如需進一步資訊，請參閱http://www.ibm.com/support/docview.wss?uid=swg21993010。

AppScan Enterprise Console 喜好設定

如果 AppScan Enterprise Server 已安裝 AppScan Enterprise Console 選項，您可以將評量發佈到那裡。 Enterprise Console 提供各種可處理評量的工具，例如：報告特性、問題管理、趨勢分析和儀表板。

如果要啟用這項特性，請完成 AppScan Enterprise Console 喜好設定頁面。這個頁面的所有欄位都必須完成有效項目，才能啟用 Enterprise Console 發佈：

- **使用者 ID** 欄位：輸入您的 AppScan Enterprise Server 使用者 ID（代表您的 AppScan Source 使用者來發佈時所建立的使用者 ID）。
  - 如果 AppScan Enterprise Server 配置為使用 Windows 鑑別，請輸入您用來連接到 Enterprise Console 的網域和使用者名稱（以 \ 區隔網域和使用者名稱 - 例如，my_domain\my_username）。
  - 如果 AppScan Enterprise Server 已配置 LDAP，請輸入您用來連接到 Enterprise Console 的使用者名稱。
  - 在 Windows 上，如果已在 AppScan Enterprise Server 中啟用「通用存取卡 (CAC)」鑑別，請從清單中選取「CAC 通用名稱」。

  最起碼，您必須是 QuickScan 使用者。如果您連接的是舊於 9.0.3 版的 AppScan Enterprise Server，您在 Enterprise Server 上必須有自己的使用者資料夾。
- **密碼**欄位：只有在您的 AppScan Enterprise Server 鑑別方法是使用者 ID 與密碼時，這個欄位才會是可用的。請輸入用來登入 Enterprise Console 的密碼（所輸入之使用者名稱的密碼）。
- **Enterprise Console URL** 欄位：輸入用來存取 Enterprise Console Web 應用程式的 URL。

  這個 URL 的格式如下：

  ```
  http(s)://<hostname>:<port>/ase
  ```

  其中，<hostname> 是 Enterprise Console 安裝所在的機器名稱，<port> 是主控台執行所在的埠（預設 <port> 是 9443）。舉例來說，這個 URL 可以是 https://myhost.mydomain.ibm.com:9443/ase。

  註：
  - 如果已經設定 **Enterprise Console URL**，則不需要修改此欄位。
  - 您必須以**管理 AppScan Enterprise 設定**許可權登入 AppScan Source，才能夠設定 **Enterprise Console URL** 欄位。如需使用者帳戶和許可權的相關資訊，請參閱產品資訊中心的管理小節，或《*IBM Security AppScan Source* 安裝與管理手冊》的『管理*AppScan Source*』小節。

- **使用者 ID** 和**密碼**儲存在執行 AppScan Source 用戶端的機器（例如，AppScan Source for Analysis），而 **Enterprise Console URL** 儲存在 Enterprise Server（可能位於遠端機器上）。您不能從遠端機器存取使用者名稱和密碼資訊（例如，從該處發出 `getaseinfo` 指令）。
- AppScan Source 不支援發佈至已配置為使用 Proxy 設定的 AppScan Enterprise Console 實例。試圖發佈至使用 Proxy 設定的實例將會導致錯誤。

完成各項設定之後，強烈建議您按一下**測試連線**，確認 Enterprise Console 伺服器的連線有效。

提示：如果連線測試失敗，請檢查 Enterprise Console 伺服器在執行中，且您能夠在瀏覽器中存取它的控制中心 URL（請使用您在上方指定的同一個 **Enterprise Console URL**）。

儲存評量

開始之前

重要：如果要儲存評量，您必須具備**儲存評量**許可權。如果要瞭解設定許可權的相關資訊，請參閱《*IBM Security AppScan Source* 安裝與管理手冊》。

關於這項作業

您可以將評量儲存在本端環境，之後就可以隨時重新開啟評量。依預設，會以 `.ozasmt` 副檔名，將評量儲存在作業系統的起始目錄中（例如，在 Windows 中，目錄可能是 `C:\Documents and Settings\Administrator\`）。

程序

1. 如果要儲存「分類」視景中目前開啟的評量，請從主工作台功能表中，選取**檔案** > **儲存評量**，或是**檔案** > **另存評量**。選擇**另存評量**動作可讓您指定所儲存評量的位置和檔名。
2. 如果要在「我的評量」視圖中儲存評量，請選取它，然後按一下視圖中的**儲存評量**或**另存評量**按鈕，或是用滑鼠右鍵按一下評量，然後選取**儲存評量**或**另存評量**。

結果

當儲存評量時，AppScan Source for Analysis 會將絕對路徑寫入評量檔中，以參照原始檔之類的項目。這些絕對路徑有可能導致難以共用另一部電腦上目錄結構不同的檔案。如果要能夠建立可攜式評量檔，您應該建立一個變數（請參閱第 84 頁的『定義變數』或第 120 頁的『在發佈和儲存時定義變數』）。

自動儲存評量

依預設，掃描會自動儲存到 `<data_dir>\scans`（其中 `<data_dir>` 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述），為期三天。這個行為取決於 `<data_dir>\config\scanner.ozsettings` 中的 `assessment_auto_save`、`assessment_auto_save_location` 和 `assessment_auto_save_stale_period` 設定。

- 當 `assessment_auto_save` 設定設為 `true` 時，評量完成之後，會自動儲存起來（您必須有**儲存評量**許可權）。

- assessment_auto_save_location 設定決定了評量的儲存位置。依預設，評量會儲存到 <data_dir>\scans。如果要變更這個位置，請將 value 屬性設為您選擇的目錄。比方說，如果要將位置設為 C:\myFolder，請將這個屬性設為 value="C:\myFolder"。
- assessment_auto_save_stale_period 設定決定了評量保存在 assessment_auto_save_location 的天數。您可以利用 value 屬性來變更這個設定。比方說，如果這個屬性設為 value="10"，10 天之後，會將儲存的評量從 assessment_auto_save_location 中移除。

從「我的評量」移除評量

當從「我的評量」視圖中移除評量時，並不會將它們從您的本端檔案系統中移除。如果從視圖中移除評量，您可以利用**開啟評量**動作，將它重新加入。

關於這項作業

限制：當您掃描多個應用程式或專案時，在「我的評量」視圖中，會建立一個母節點來包含每個掃描項目的評量。在這種情況下，無法管理個別子項評量（例如，無法個別移除或發佈子項評量）。當同時掃描多個應用程式或專案時，您只能以群組（母節點）的方式來管理各個評量。

程序

1. 在「我的評量」視圖中，選取您想要移除的評量。您也可以利用鍵盤的 Ctrl 或 Shift 鍵來選取多項評量。
2. 在視圖工具列中選取**從我的評量移除**按鈕，或用滑鼠右鍵按一下選項，然後從功能表中選取**從我的評量移除**。

定義變數

當儲存評量或組合時，或是發佈評量時，AppScan Source for Analysis 可能會建議您建立變數，來取代絕對路徑（如果沒有變數，AppScan Source for Analysis 會將絕對路徑寫入評量檔中，以參照原始檔之類的項目）。配置絕對路徑的變數，有助於多部電腦共用評量。建議您在共用評量時使用變數。

關於這項作業

在起始儲存或發佈動作之前，您可以遵循這個主題中的指示來建立變數，或者在起始儲存或發佈動作之後，遵循第 120 頁的『在發佈和儲存時定義變數』中的步驟，來建立變數。

舉例來說，若想瞭解當共用評量時變數有何功效，請參閱第 120 頁的『範例：定義變數』。

程序

1. 從主功能表選取**編輯** > **喜好設定**。在「喜好設定」對話框中選擇**變更變數**。
2. 在「變更變數」喜好設定頁面中，按一下**新增變數**按鈕。
3. 輸入變數的名稱，且瀏覽至將以變數取代的檔案位置（在建立變數之後，AppScan Source for Analysis 會在周圍插入百分比符號 %）。

4. 針對評量中的其他任何參照項目，重複上述步驟（例如，如果評量參照來源位於多個位置，請針對每一個位置，各新增一個變數）。
5. 使用喜好設定頁面，用**修改變數**和**刪除變數**按鈕來編輯和移除變數。
6. 當完成定義變數時，請按一下**確定**。

在發佈和儲存時定義變數

在您嘗試儲存或發佈評量時，AppScan Source for Analysis 會偵測評量中的任何絕對路徑。如果未建立絕對路徑的對應變數，會提示您建立它們。

關於這項作業

在起始儲存或發佈動作之前，您可以遵循第 84 頁的『定義變數』中的指示來建立變數，或者在起始儲存或發佈動作之後，遵循這個主題中的步驟，來建立變數。

舉例來說，若想瞭解當共用評量時變數有何功效，請參閱『範例：定義變數』。

程序

1. 起始儲存或發佈動作之後，請在「偵測絕對路徑」訊息中，按一下**是**。
2. 在「定義變數」對話框中，AppScan Source for Analysis 會建議一組封裝了資料的路徑。
3. 請選取一個目錄，然後按一下**新增變數**。
4. 針對評量中的其他任何參照項目，重複上述步驟（例如，如果評量參照來源位於多個位置，請針對每一個位置，各新增一個變數）。
5. 您也可以利用「定義變數」對話框中的**修改變數**和**刪除變數**按鈕，來編輯和移除變數。
6. 按一下**確定**，完成儲存或發佈動作。

範例：定義變數

如果要共用評量資料，您必須定義適當的變數。這個主題的範例說明變數的必要性。

使用者 Joe 在 A 電腦上進行掃描，所有原始碼都在 `C:\dev\my_code` 目錄之下。Joe 想要將它的掃描結果儲存在檔案中，與 Bill 共用。Bill 使用 B 電腦，Joe 所掃描的相同原始碼是在 `C:\code\bills_code` 目錄之下。若不使用變數，評量檔會參照所有絕對路徑起始於 `C:\dev\my_code` 的原始檔。如果 Bill 在 B 電腦上開啟這個評量檔，AppScan Source for Analysis 會找不到原始檔，因為它們是在 B 電腦的 `C:\code\bills_code` 之下。

解決方案

Joe 和 Bill 都應該建立一個變數來指向原始碼的根目錄。Joe 在 AppScan Source for Analysis 中建立一個名稱為 `SRC_ROOT` 的變數，並提供 `C:\dev\my_code` 作為它的值。這個變數在 Joe 的 AppScan Source for Analysis 安裝架構的本端。 Joe 將變數名稱 (`SRC_ROOT`) 及指向的位置告訴 Bill。之後，Bill 就在他的 AppScan Source for Analysis 中，建立一個名稱為 `SRC_ROOT` 的變數，其值為 `C:\code\bills_code`。當 Joe 儲存他的掃描時，`SRC_ROOT` 變數會取代 `C:\dev\my_code` 路徑。當 Bill 開啟來自 Joe 的評量檔時，`C:\code\bills_code` 會替代 `SRC_ROOT` 變數。

第 5 章 分類及分析

將類似的發現項目分組，可讓安全分析師或 IT 審核員進行原始碼問題的分段及分類。本節說明如何分類 AppScan Source 評量以及分析結果。

當掃描程式碼時，會出現掃描結果或發現項目。分類是指評估發現項目，以及判斷如何解決它們的程序。不過，達到這個目標所需要的步驟會隨著多種因素而不同，其中包括發現項目的總數、特定安全考量、應用程式風險評量，等等。除了決定某發現項目是否代表有效的安全問題之外，分類也包含適時修改發現項目的屬性（嚴重性、類型、分類）。

分類策略很重要，可確保您能夠依照所需要的次序和時段來完成您的目標。分類最好是以反覆運算的方式來完成，以便在各次反覆運算中，評估發現項目的子集以及判斷各子集的處置方式。您可以用許多有效的方式，來決定分類反覆運算的定義方式。其中一個方式是根據整體嚴重性，來建立高風險發現項目的子集。您可以先解決可能帶來最大風險的發現項目，再解決風險較小的發現項目。另一個方式是依「SQL 注入」或「需要驗證」等安全考量來定義子集。

一般而言，分類是由安全分析師或 IT 審核員來執行。分析師或審核員可以先將程式碼需要變更的發現項目提交給問題追蹤系統，再提交給開發人員進行補救。有時開發人員也會進行分類及解決問題。

在分類階段期間，您可以：

- 檢閱特別令人關注之漏洞類型的發現項目
- 檢視特定種類中的 API
- 比較不同評量中的發現項目
- 過濾或排除特定的發現項目
- 變更發現項目的嚴重性或漏洞類型
- 將可疑和掃描涵蓋面發現項目升級為明確的發現項目
- 標註發現項目
- 將問題報告提交問題追蹤系統，或以電子郵件將發現項目傳送給他人。

AppScan Source 提供所有必要的工具，讓您使用各種不同的分類策略來分析結果。過濾可讓您只檢視特定的分類反覆運算中所要處理的發現項目。如果您的反覆策略是依嚴重性和分類來進行，您可以過濾「漏洞矩陣」視圖中的發現項目。如果您的反覆策略是依「漏洞類型」來進行，您可以過濾「評量摘要」視圖。 另外，AppScan Source for Analysis 也提供支援複雜反覆運算方式的過濾器編輯器。

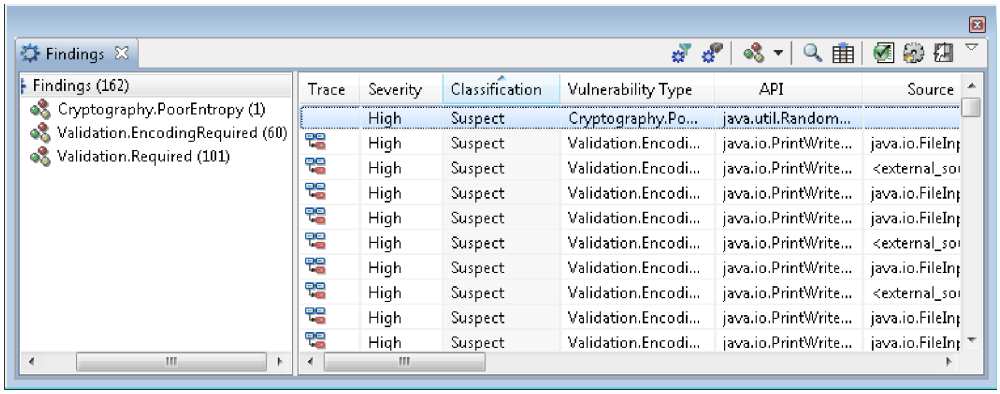
選取分類方式之後，AppScan Source for Analysis 可支援對發現項目進行處置。

- 排除個別發現項目或發現項目的集合
- 修改發現項目詳細資料（類型、嚴重性、分類）
- 建立組合（發現項目的分組機制）
- 利用「評量差異」視圖來比較評量

顯示發現項目

在「發現項目」視圖或任何含有發現項目的視圖中，每次掃描都會顯示一個「發現項目樹狀結構」（評量準則的階層式分組）和「發現項目表格」。您在發現項目樹狀結構中選取的項目，會決定呈現在表格中的發現項目。

如果選取樹狀結構根目錄，則會將所有發現項目顯示在表格中；如果選取一個分組類型，則只會顯示那些類型的發現項目。



AppScan Source for Analysis 依不同的分組來顯示發現項目，其中包括：

- **漏洞類型**
- **分類**
- **檔案**
- **來源**
- **接收槽**
- **API**
- **組合**
- **CWE**
- **表格**

註：分類和嚴重性排序，依預設，按降冪排序。所有其他直欄按升冪排序。

這些直欄出現在發現項目表格中。

表 *11.* 發現項目表格

直欄標題	說明
追蹤	這個直欄中的圖示指出遺失或已知的接收槽有追蹤資料存在。

表 *11.* 發現項目表格 *(*繼續*)*

直欄標題	說明
嚴重性	• 高：造成資料在機密性、完整性或可用性方面的風險，以及/或造成處理資源在完整性或可用性方面的風險。高度嚴重性狀況應該優先立即補救。 • 中：造成資料安全和資源完整性的風險，但狀況不是很容易遭到攻擊。中度嚴重性狀況應該儘可能檢查及補救。 • 低：造成最低的資料安全或資源完整性風險。 • 參考資訊：發現項目本身不容易產生危害。它說明程式碼中使用的技術、架構性質或安全機制。
分類	發現項目的類型：**明確**或**可疑**安全發現項目 - 或**掃描涵蓋面**發現項目。 **註：**在某些情況下，**無**這項分類可用來表示既非安全發現項目也非掃描涵蓋面發現項目的分類。
漏洞類型	漏洞種類，例如 `Validation.Required` 或 `Injection.SQL`。
API	有漏洞的呼叫，顯示 API 和傳給 API 的引數。
來源	來源是對程式的輸入，例如：檔案、Servlet 要求、主控台輸入或 Socket。對於大部分的輸入來源而言，傳回的資料內容和長度都是沒有限制的。如果沒有對輸入進行檢查，則會將其視為受到污染。
接收槽	接收槽可以是資料能夠寫出的任何外部格式。資料庫、檔案、主控台輸出和 Socket 都是接收槽的範例。未經檢查，便將資料寫入接收槽，可能是一個嚴重的安全漏洞。
目錄	掃描之檔案的完整路徑。
檔案	出現安全發現項目或掃描涵蓋面發現項目的程式檔名稱。發現項目中的檔案路徑是相對於掃描的專案工作目錄。
呼叫方法	發出有漏洞呼叫的函數（或方法）。
行	程式檔中有漏洞的 API 所在的行號。
組合	包含這個發現項目的組合。
CWE	ID and topic of the community-developed dictionar一般軟體弱點的社群開發字典 ID 和主題（一般弱點列舉 (CWE) 主題）。

註：如果 AppScan Source 找不到您選取之發現項目的原始檔，系統會顯示一個對話框，詢問當找不到原始檔時是否要提示您。如果您選取**是**，每次找不到所選取之發現項目的原始檔時，都會提示您。如果您選取**否**，即不會提示您。只要現行評量開啟，這項設定即會持續有效。每次開啟評量或是結束 AppScan Source 時，會重設此設定。

AppScan Source 分類程序

分類程序包括透過組合、過濾器及排除項目來操作發現項目，以及比較評量結果。

過濾器

過濾器是一組規則，用來定義有某些特點的發現項目。過濾器可讓您呈現這些發現項目的動態視圖，以及將類似的發現項目進行分類。

過濾器有共用過濾器和本端過濾器：

- 共用過濾器位於 AppScan 伺服器上。任何連接這部伺服器的人，都可以使用這個過濾器。
- 本端過濾器在本端電腦上。

組合

組合是儲存在應用程式中的個別發現項目的具名集合。如果要建立組合，只需選取發現項目，並將它們新增至新的或現有的組合即可。

將類似的發現項目分組到組合中，可讓安全分析師進行原始碼問題的分段及分類。您可以將組合提交給問題追蹤系統，或用電子郵件將發現項目傳給開發人員檢閱，作為分類及分析程序的一部分。

排除

排除會省略發現項目不進行掃描。 AppScan Source 有內建**已排除的組合**，其中含有您所排除的任何發現項目（例如：因為它們不需要解決）。

註：從評量結果中排除的發現項目，不會列入應用程式或專案度量的計算中。

已修改的發現項目

已修改的發現項目是漏洞類型、嚴重性或分類已變更的發現項目。如果您新增附註到發現項目中，發現項目也會視為已修改。

比較評量

在 AppScan Source for Analysis 中，請使用**差異評量**動作，來比較評量。比較兩項評量時，會將兩者之間的差異顯示在「評量差異」視圖中（類似於「我的評量」視圖和「發現項目」視圖的組合）。

註：比較評量時，會忽略過濾器和組合。

分類範例

這個範例說明安全分析師所用的 AppScan Source 分類工作流程。分類工作流程可能會隨著商業需求而不同。

Jones 先生是公司的安全分析師，想要將他的掃描結果分類。他想將類似的發現項目分組及設定優先順序，然後將它們提交給適當的開發人員解決。

首先，Jones 先生掃描應用程式的原始碼，然後在「分類」視景中開啟評量。掃描產生大約 2,000 個發現項目，他可以在「發現項目」視圖中檢閱所有這些發現項目。不過，

Jones 先生想要先取得結果的概觀，他開啟依嚴重性和發現項目類型（安全或掃描涵蓋面）顯示分析的「漏洞矩陣」視圖。掃描涵蓋面發現項目和可疑安全發現項目需要進一步調查，才能判定風險。

在「漏洞矩陣」中，Jones 先生看到 8 個高度嚴重性的明確安全發現項目。他按一下指出這 8 個明確發現項目的矩陣框，因而自動建立一個過濾器，並使「發現項目」視圖重新整理，只顯示這 8 個重要問題。Jones 先生決定將這些問題當作錯誤來處理。他選取這 8 個問題，將它們提交給問題追蹤系統。然後他從「漏洞矩陣」重設了過濾器。

之後，Jones 先生專注於「評量摘要」視圖。他發現這 2,000 個發現項目由超出 6 個的漏洞類型組成。他決定集中處理驗證問題，從「評量摘要」視圖建立另一個過濾器。在圖形上，他按一下 `Validation.EncodingRequired` 和 `Validation.Required`，將「發現項目」視圖中的發現項目數縮減到大約 500 個。

500 個發現項目還是很難分類。 Jones 先生決定進一步過濾結果。在「過濾器編輯器」視圖中，他以高度嚴重性作為需求來加強從「評量摘要」視圖建立的過濾器。現在，發現項目表格顯示 150 個項目。

當依檔名來排序時，他發現某些發現項目是來自協力廠商程式庫內的程式碼。 Jones 先生知道這個程式庫的用法是隔離的，他不想處理它的安全問題。他排除這些發現項目，立即更新「發現項目」視圖和度量。未來的掃描仍會偵測這些發現項目，但它們是隔離的，不會列入度量。

Jones 先生發現一些 `Validation.Required` 類型的高度嚴重性可疑安全發現項目。他知道有未經驗證，就直接使用資料的情況。他決定將這些發現項目從可疑升級為明確。當進行這些修改時，他又決定新增附註來說明他的變更，然後用電子郵件將這些發現項目傳給自己，以提醒他設定補救的優先順序，或在「已修改的發現項目」視圖中檢閱它們。

之後，Jones 先生又依檔名排序，發現有些發現項目是在後端伺服器，有些是在使用者介面。他選取所有後端發現項目，建立一個新的組合，標籤為後端伺服器 - 需要驗證。他選取其餘發現項目，將它們放在標籤為 `UI - 需要驗證`的組合中。分類繼續進行，焦點是高度嚴重性的 `Validation.EncodingRequired` 類型。

當天末了，Jones 先生建立了 12 個組合。在這一天裡，他利用圖形、過濾器和「漏洞矩陣」，將發現項目刪改成視圖中每次所能管理的數目。他有時將這些個別發現項目放在組合中。另一些時候，他會排除不重要的發現項目。他有時專為了特定的發現項目而建立新的組合；有時將發現項目新增到現有的組合中。

現在，Jones 先生檢閱這 12 個組合。他判定應該將後端伺服器 - 需要驗證和 `UI - 需要驗證`組合提交給問題追蹤系統，以通知開發人員關注這些區域。

Jones 先生移至「組合」視圖，然後開啟後端伺服器 - 需要驗證組合。這時開啟了標題為**後端伺服器 - 需要驗證**的新視圖，列出他放在這個組合中的發現項目。之後，他將這個組合提交給問題追蹤系統。當晚稍後，當開發人員登入 Rational ClearQuest 並見到指派給他的錯誤時，他就可以在 AppScan Source for Development 中開啟發現項目。

Jones 先生又檢閱其他組合。他將一部分提交給問題追蹤系統，用電子郵件將另一些傳給他的同事。不過，在進一步檢閱之後，部分組合中的發現項目，對他而言，不那麼重要。他將這些比較不重要的發現項目移到兩個新的組合中：依設計和無關。Jones 先生判定這些發現項目是可接受的，他不想變更程式碼。除了依設計和無關的發現項目

之外，Jones 先生理解到所有 Cryptography.PoorEntropy 發現項目，對他而言，也不重要。他知道，這些加密法呼叫的熵可能不好，雖然快速電腦能夠在一星期內破解金鑰，但這不重要，應用程式的資料在加密幾小時之後就不再有用了。Jones 先生也想將它們移除。

然後他將依設計和無關組合新增到「內容」視圖的**已排除的組合**清單中。他還打開「過濾器編輯器」，建立另一個含有漏洞類型 Cryptography.PoorEntropy 的過濾器，儲存名稱為 Crypto 的過濾器，將 Crypto 過濾器的行為設為**反轉**（在「選取過濾器」對話框中，他選擇**反轉過濾器**）。之後，他啟動一項掃描，然後回家。這些度量不會反映這些排除項目，直到下一次掃描之後。

利用過濾器分類

AppScan Source for Analysis 會報告所有潛在的安全漏洞，對於中型到大型程式碼庫，可能會產生數千個發現項目。掃描時，您可能會看到發現項目清單包含無關緊要的項目。如果要從「發現項目」視圖中移除特定發現項目，您可以選擇預先定義的過濾器，也可以建立自己的過濾器。過濾器用來指定一些準則，以決定要從視圖中移除哪些發現項目。

- 『過濾器概觀』
- 『過濾器規則』
- 第 129 頁的『過濾器範例』

過濾器概觀

過濾器會移除或限制符合過濾器規則所決定之準則的項目，並協助您在分類或產生報告期間管理掃描結果。過濾器有助於引導工作流程，使安全分析師能夠集中精力來處理發現項目子集的最重要的區域。例如，在檢查程式碼期間，分析師可以建立一個過濾器來避免檢視嚴重性低的發現項目。另外，分析師也可能偏好排除系統程式庫 include 檔案中的漏洞。過濾器可以從視圖中刪除這些項目，也可以排除個別檔案或先前調查的檔案。

可在掃描之前或之後套用過濾器：

- 如果要在掃描之前套用過濾器，請在專案或應用程式內容中設定廣域過濾器 - 或者，您可以使用包含過濾器的掃描配置來進行掃描。當您在掃描之前套用過濾器時，無法顯示未過濾的發現項目，或沒有重新掃描就移除過濾器。
- 各種視圖（尤其是「過濾器編輯器」視圖）可讓您在掃描之後套用過濾器。當這些視圖用於過濾時，所有過濾的項目都會保留在掃描結果中 - 唯有選取了**顯示過濾的發現項目**（  ）切換之後，它們才會出現在「發現項目」視圖中。

AppScan Source 包含數個預先定義的過濾器，可選取它們來過濾掃描結果。

有了過濾器，您就可以設定內容，使過濾器成為排除項目。排除項目會影響掃描，刪除所有符合過濾器的發現項目，或刪除所有不符合過濾器的發現項目。

過濾器規則

每個過濾器都由若干規則組成，這些規則定義發現項目表格中的結果要限制（併入）或移除（排除）哪些發現項目（如果是追蹤規則，您可以根據追蹤內容同時限制及移除）。

- **限於**規則（內含規則）會排除沒有指定準則的發現項目，且會從發現項目表格的可見結果中移除這些發現項目。
- **移除**規則（排除規則）會從掃描結果中，移除包含準則的發現項目。移除規則會排除任何具有指定準則的發現項目，且會從可見結果中移除這些發現項目。

過濾器規則可能包含下列特點：

- **嚴重性**：指出個別發現項目的潛在影響或風險。嚴重性規則是「僅限制」。
  - 高：造成資料在機密性、完整性或可用性方面的風險，以及/或造成處理資源在完整性或可用性方面的風險。高度嚴重性狀況應該優先立即補救。
  - 中：造成資料安全和資源完整性的風險，但狀況不是很容易遭到攻擊。中度嚴重性狀況應該儘可能檢查及補救。
  - 低：造成最低的資料安全或資源完整性風險。
  - 參考資訊：發現項目本身不容易產生危害。它說明程式碼中使用的技術、架構性質或安全機制。
- **分類**：根據這個主題所說明的分類來過濾發現項目。分類規則是「僅限制」。
- **漏洞類型**：依特定漏洞種類來過濾，例如 `BufferOverflow`。當您新增漏洞類型時，您可以從所有可能的漏洞類型中選取，也可以只從在現行評量中找到的類型中選擇。如果要從現行評量中所找到的漏洞類型中選擇，請在「選取值」對話框中，選取**只顯示已開啟評量中的值**。

  當您建立過濾器供未來的掃描使用時，從所有可能的漏洞類型中選取很有用。如果要顯示所有漏洞類型，請取消選取**只顯示已開啟評量中的值**（如果沒有已開啟的評量，依預設，會顯示所有漏洞類型，**只顯示已開啟評量中的值**勾選框無法使用）。
- **API**：過濾特定 API 的所有漏洞。
- **檔案**：過濾特定檔案中的所有漏洞。
- **目錄**：過濾特定目錄中的所有漏洞。
- **專案**：過濾特定專案中的所有漏洞。
- **追蹤**：可讓您根據追蹤內容來過濾發現項目（請參閱第 163 頁的『來源和接收槽』，以進一步瞭解追蹤內容）。過濾器可包括根據追蹤內容而同時限制及移除的追蹤規則。當您按一下任一區段（限制或移除）的**新增**時，會開啟「追蹤規則項目」對話框。在這裡，您可以指定：
  - **來源**：在「來源」區段的 **API RegEx** 欄位中，指定追蹤來源或涵蓋多重來源的正規表示式（預設項目是 **.*** - 會傳回全部的正規表示式或萬用字元）。如果您使用正規表示式，請在**正規表示式類型**欄位功能表中選取類型（預設正規表示式類型是 **PERL**）。如果沒有使用正規表示式，請在**正規表示式類型**欄位功能表中選取**完全相符**。

    如果 **API RegEx** 項目是有效的表示式，欄位旁會出現一個綠色勾號圖示。如果項目不是有效的表示式，欄位旁會出現一個紅色 X 圖示，對話框的**確定**按鈕會停用。將游標停在任一圖示上，會提供驗證結果的相關資訊。如果您建立了不是有效表示式的項目，但想要繼續使用它，請選取對話框底端的**忽略上述的驗證錯誤**勾選框。如此便能啟用對話框的**確定**按鈕（只要表示式不是空白），無效表示式旁的圖示也會變成綠色勾號，浮動說明為**已停用驗證**。

您也可以使用「來源內容」區段中的**新增 VMAT 內容**按鈕，依機制或技術來精簡過濾器（以下提供 VMAT 內容的相關資訊）- 不過，使用此特性來依漏洞限制，不會達到想要的效果，因為漏洞類型是由接收槽決定，而非來源。

– **接收槽**：在「接收槽」區段中，您可以依照指定來源的相同方式，新增接收槽作為過濾器。

  您可以精簡過濾器，將它限制於特定的漏洞類型（將追蹤規則項目的效果僅限於特定類型的漏洞、機制或技術）。如果要這麼做，請按一下「接收槽內容」區段中的**新增 VMAT 內容**按鈕，然後在「選擇內容」對話框中選取內容。內容清單可以利用**過濾器**欄位來過濾。

  VMAT 是 AppScan Source 套用於應用程式設計介面 (API) 的四個主要內容類型的分類。 VMAT 內容種類包括：

  - **漏洞**：會造成安全違規的不當使用或攻擊方向類型
  - **機制**：用來防止漏洞的安全控制
  - **屬性**：在「選擇內容」對話框中，目前無法使用這些內容
  - **技術**：API 提供之功能類型的一般說明

  過濾器範例：如果要過濾來自 HTTP（最高風險來源）的所有 SQL 注入和 XSS，請建立**限於**追蹤規則，此追蹤規則在「來源內容」區段中包含 `Technology.Communications.HTTP` 過濾器，而在「接收槽內容」區段中包含 `Vulnerability.Injection.SQL` 和 `Vulnerability.CrossSiteScripting` 規則。

– **必要的呼叫**：在「必要的呼叫」區段中，新增必須出現在來源至接收槽路徑上的特定 API 呼叫。必要的呼叫會將發現項目限制為，其追蹤資料會通過所指定之必要呼叫的發現項目。當您按一下**新增中間呼叫**時，會開啟「配置 API」對話框。在這個對話框中，請依照指定來源和接收槽的相同方式來指定呼叫。

– **禁止的呼叫**：在「禁止的呼叫」區段中，新增不能出現在來源至接收槽路徑上的特定 API 呼叫。禁止的呼叫會將發現項目限制為，其追蹤資料沒有通過所指定之禁止呼叫的發現項目。請依照新增必要的呼叫的相同方式來新增禁止的呼叫。

提示：

– 當依**漏洞類型**、**API**、**檔案**、**目錄**或**專案**來過濾時，您可以在「選取值」對話框頂端的過濾器欄位中，輸入型樣來過濾對話框中所顯示的清單。

– 在任何發現項目表格中，查看**來源**和**接收槽**直欄，體驗一下您想要濾除的來源和接收槽。

– 如果要體驗您想要過濾的來源、接收槽和呼叫內容，請查看任何發現項目表格中的**漏洞類型**直欄。

– 如果要查看您可以過濾的呼叫，請檢視任何發現項目表格中的 **API** 直欄項目。

過濾器範例

表 *12.* 過濾器範例

發現項目表格中的過濾器行為	過濾器編輯器中的「過濾器設定」視圖
發現項目表格只包含高度嚴重性可疑安全發現項目。	• 在「嚴重性」區段中，選取**高**勾選框，然後取消選取所有其他勾選框。 • 在「分類」區段中，選取**可疑**勾選框，然後取消選取所有其他勾選框。
發現項目表格包括名稱為 ProjectA 的專案中，除了資訊漏洞類型以外的所有發現項目。	• 在「漏洞類型」區段中，選取**移除**圓鈕，然後按一下**新增**。在「選取值」對話框中，選取 **Vulnerability.Info**。 • 在「專案」區段中，選取**限於**圓鈕，然後按一下**新增**。在「選取值」對話框中，選取 ProjectA。
只會顯示含有追蹤的發現項目。	在「追蹤」區段中，按一下**限於**區段中的**新增**。接受「追蹤規則項目」對話框中的預設項目，然後按一下**確定**。對話框中的預設值如下： • 來源 **API RegEx** 欄位是 **.***，正規表示式類型是 **PERL**。這會告訴 AppScan Source 過濾含有來源的任何發現項目（使用 Perl 正規表示式語法）。 • 接收槽 **API RegEx** 欄位是 **.***，正規表示式類型是 **PERL**。這會告訴 AppScan Source 過濾含有接收槽的任何發現項目（使用 Perl 正規表示式語法）。
發現項目表格會顯示沒有通過 java.lang.Integer.parseInt 的「HTTP 相關來源至 SQL 注入相關接收槽」。	在「追蹤」區段中，按一下**限於**區段中的**新增**。在「追蹤規則項目」對話框中，完成下列步驟： • 在「來源」區段中，按一下**新增 VMAT 內容**。在「選擇內容」對話框中，選取 Technology.Communications.HTTP。按一下**確定**來新增 VMAT 內容，然後返回「追蹤規則項目」對話框。 • 在「接收槽」區段中，按一下**新增 VMAT 內容**。在「選擇內容」對話框中，選取 Vulnerability.Injection.SQL。按一下**確定**來新增 VMAT 內容，然後返回「追蹤規則項目」對話框。 • 在「禁止的呼叫」區段中，按一下**新增中間呼叫**。在「配置 API」對話框的 **API RegEx** 欄位中，輸入 java.lang.Integer.parseInt.*。按一下**確定**來新增中間呼叫，然後返回「追蹤規則項目」對話框，按一下**確定**來新增追蹤規則項目。

使用 AppScan Source 預先定義的過濾器

AppScan Source 包含一組可供選取來過濾掃描結果的預先定義過濾器。這個說明主題說明這些立即可用的過濾器。

註：在 AppScan Source 8.8 版，預先定義的過濾器已改良，可提供更好的掃描結果。如果您需要繼續使用 AppScan Source 舊版本中預先定義的過濾器（保存的過濾器是列在 第 133 頁的『AppScan Source 預先定義的過濾器（8.7.x 版及更舊版本）』中），請遵循第 134 頁的『還原保存的預先定義過濾器』中的指示。

註：在 AppScan Source for Development（Visual Studio 外掛程式）中，這個視圖是「編輯過濾器」視窗的一部分。

- 『! - AppScan 關鍵少數』
- 『! - 高風險來源』
- 第 131 頁的『! - 重要類型』
- 第 131 頁的『CWE SANS Top 25 2010 漏洞』
- 第 131 頁的『外部通訊』
- 第 131 頁的『低嚴重性和參考資訊』
- 第 131 頁的『雜訊 - 品質』
- 第 132 頁的『OWASP Mobile Top 10 漏洞』
- 第 132 頁的『OWASP Top 10 2010 漏洞』
- 第 132 頁的『OWASP Top 10 2013 漏洞』
- 第 132 頁的『「PCI 資料安全標準」漏洞』
- 第 132 頁的『已鎖定的漏洞 - EncodingRequired 針對 HTTP 來源』
- 第 132 頁的『已鎖定的漏洞 - C/C++ 接收槽需要驗證』
- 第 133 頁的『授信來源』
- 第 133 頁的『無追蹤資料的漏洞』

! - AppScan 關鍵少數

這個過濾器會從某些最危險的漏洞種類之中找出符合的發現項目。結果限於高嚴重性和中嚴重性的漏洞。會從發現項目中移除具有特定來源的結果。這個過濾器所包含的特定漏洞種類如下：

```
Vulnerability.CrossSiteScripting
Vulnerability.CrossSiteScripting.Reflected
Vulnerability.CrossSiteScripting.Stored
Vulnerability.Injection.OS
Vulnerability.Injection.LDAP
Vulnerability.Injection.SQL
Vulnerability.Injection.Mail
```

! - 高風險來源

這個過濾器將發現項目的範圍限於特定的漏洞類型，以及具有下列其中一個內容的來源：

```
Technology.Communications.HTTP
Technology.Communications.IP
Technology.Communications.RCP
Technology.Communications.TCP
Technology.Communications.UDP
Technology.Communications.WebService
```

! - 重要類型

這個過濾器包含的發現項目來自含有重要漏洞種類的更廣泛範圍。以「最後」或「可疑」分類而言，發現項目限於高嚴重性和中嚴重性。這個過濾器所包含的特定種類如下：

```
Vulnerability.AppDOS
Vulnerability.Authentication.Credentials.Unprotected
Vulnerability.BufferOverflow
Vulnerability.BufferOverflow.FormatString
Vulnerability.BufferOverflow.ArrayIndexOutOfBounds
Vulnerability.BufferOverflow.BufferSizeOutOfBounds
Vulnerability.BufferOverflow.IntegerOverflow
Vulnerability.BufferOverflow.Internal
Vulnerability.CrossSiteRequestForgery
Vulnerability.CrossSiteScripting
Vulnerability.CrossSiteScripting.Reflected
Vulnerability.CrossSiteScripting.Stored
Vulnerability.FileUpload
Vulnerability.Injection
Vulnerability.Injection.LDAP
Vulnerability.Injection.OS
Vulnerability.Injection.SQL
Vulnerability.Injection.XML
Vulnerability.Injection.XPath
Vulnerability.Malicious.EasterEgg
Vulnerability.Malicious.Trigger
Vulnerability.Malicious.Trojan
Vulnerability.PathTraversal
Vulnerability.Validation.EncodingRequired
Vulnerability.Validation.EncodingRequired.Struts
```

CWE SANS Top 25 2010 漏洞

這個過濾器聚焦在與 2010 年 *CWE/SANS TOP 25* 最危險的軟體錯誤相關的漏洞類型。

如果要進一步瞭解 *2011 CWE/SANS Top 25 Most Dangerous Software Errors*（2011 CWE/SANS Top 25 最危險的軟體錯誤），請參閱 http://cwe.mitre.org/top25/。

外部通訊

這個過濾器會找出從應用程式之外，透過網路而來的發現項目。這個過濾器會找出來自任何 `Technology.Communications` 來源的發現項目。

低嚴重性和參考資訊

這個過濾器包含嚴重性為「低」和「參考資訊」的發現項目。包含所有分類（明確、可疑和掃描涵蓋面）。

雜訊 - 品質

此過濾器導致結果只包含與品質編寫實務相關的漏洞類型。

OWASP Mobile Top 10 漏洞

這個過濾器聚焦在與「開放式 Web 應用程式安全專案 (OWASP)」Mobile Top 10 Release Candidate 1.0 版清單相關的漏洞類型。

如果要瞭解 OWASP，請參閱 https://www.owasp.org/index.php/Main_Page。各種 OWASP 文件及安全風險的鏈結位置如下：https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project。

OWASP Top 10 2010 漏洞

這個過濾器聚焦在與「開放式 Web 應用程式安全專案 (OWASP)」Top 10 2010 清單相關的漏洞類型。

如果要瞭解 OWASP，請參閱 https://www.owasp.org/index.php/Main_Page。各種 OWASP 文件及安全風險的鏈結位置如下：https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project。

OWASP Top 10 2013 漏洞

這個過濾器聚焦在與「開放式 Web 應用程式安全專案 (OWASP)」Top 10 2013 清單相關的漏洞類型。

如果要瞭解 OWASP，請參閱 https://www.owasp.org/index.php/Main_Page。各種 OWASP 文件及安全風險的鏈結位置如下：https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project。

「PCI 資料安全標準」漏洞

這個過濾器聚焦在與「付款卡產業資料安全標準 (PCI DSS) 3.2 版」標準相關的漏洞類型。

如需資訊，請參閱 https://www.pcisecuritystandards.org/security_standards/index.php。

掃描涵蓋面發現項目

此過濾器導致結果只包含「掃描涵蓋面發現項目」（如需相關資訊，請參閱第 18 頁的『分類』）。

已鎖定的漏洞 - *EncodingRequired* 針對 HTTP 來源

這個過濾器聚焦在 `Validation.EncodingRequired` 和 `Validation.EncodingRequired.Struts` 漏洞種類的發現項目。只包含從 `Technology.Communications.HTTP` 來源產生的發現項目。以「明確」或「可疑」分類而言，發現項目限於高嚴重性和中嚴重性。

已鎖定的漏洞 - C/C++ 接收槽需要驗證

這個過濾器聚焦在一組已知 C 和 C++ 接收槽的 `Validation.Required` 漏洞。以「明確」或「可疑」分類而言，發現項目限於高嚴重性和中嚴重性。

授信來源

這個過濾器假設來自階段作業物件或要求屬性等特定來源的資料是安全的。

無追蹤資料的漏洞

此過濾器列出不包含追蹤資料的漏洞。

AppScan Source 預先定義的過濾器（8.7.x 版及更舊版本）

本主題列出 AppScan Source 8.7.x 版及更舊版本所包含的預先定義過濾器。

如果您需要存取這過濾器，請遵循第 134 頁的『還原保存的預先定義過濾器』中的指示。

! - 關鍵少數

這個過濾器會從某些最危險的漏洞種類之中找出符合的發現項目。只包括來自外部網路通訊來源的發現項目。這個過濾器提供雷射聚焦的高風險發現項目起點。這個過濾器所包含的特定種類如下：

```
Vulnerability.BufferOverflow
Vulnerability.BufferOverflow.FormatString
Vulnerability.PathTraversal
Vulnerability.CrossSiteScripting
Vulnerability.CrossSiteScripting.Reflected
Vulnerability.CrossSiteScripting.Stored
Vulnerability.Injection
Vulnerability.Injection.LDAP
Vulnerability.Injection.SQL
Vulnerability.Injection.OS
Vulnerability.Injection.XML
Vulnerability.Injection.XPath
```

高優先順序 - 外部通訊

這個過濾器會找出從應用程式之外，透過網路而來的發現項目。這個過濾器會找出來自任何 Technology.Communications 來源的發現項目。

高優先順序 - 重要類型

這個過濾器包含某些最危險的漏洞種類（例如 CrossSiteScripting 和 Injection.SQL）中的發現項目。這個過濾器所包含的特定種類如下：

```
Vulnerability.AppDOS
Vulnerability.Authentication.Credentials.Unprotected
Vulnerability.Authentication.Entity
Vulnerability.BufferOverflow
Vulnerability.BufferOverflow.FormatString
Vulnerability.CrossSiteScripting
Vulnerability.CrossSiteScripting.Reflected
Vulnerability.CrossSiteScripting.Stored
Vulnerability.Injection
Vulnerability.Injection.LDAP
Vulnerability.Injection.OS
Vulnerability.Injection.SQL
Vulnerability.Injection.XML
Vulnerability.Injection.XPath
Vulnerability.PathTraversal
```

低優先順序 - 測試程式碼

這個過濾器包含測試程式碼的發現項目。這個過濾器中的特定類型包括：

```
Vulnerability.Quality.TestCode
```

雜訊 - Copy-like 作業

這個過濾器包含與 Copy-like 作業相關的發現項目。如果資料來源不一定可信任，但是對資料執行的動作是可信任的，就會進行 Copy-like 作業。

會尋找以下型樣：

```
Technology.Database --> Vulnerability.Injection.SQL
Mechanism.SessionManagement --> Mechanism.SessionManagement
Technology.XML, Technology.XML.DOM, Technology.XML.Schema,
Technology.XML.XPath --> Vulnerability.AppDOS.XML,
Vulnerability.Injection.XML
```

雜訊 - 記載問題

這個過濾器包含與錯誤處理相關的發現項目。找出源自錯誤處理常式至記載機制的發現項目。符合這個型樣，如下所示：

```
Mechanism.ErrorHandling -->
Vulnerability.Logging, Vulnerability.Logging.Forge, Vulnerability.Logging.Required
```

雜訊 - 嚴重性低

這個過濾器包含嚴重性「低」的發現項目。包括所有分類。

雜訊 - 授信來源

這個過濾器包含源自授信來源的發現項目。只有將 `java.lang.System.getProperty.*` 作為其來源的發現項目，才會併入此過濾器中。

還原保存的預先定義過濾器

可遵循此作業的步驟，將 8.8 版之前的 AppScan Source 所提供的預先定義過濾器加回到產品中。當它們在單一機器上還原之後，就可以使用與您建立的過濾器的相同方式來管理它們（例如，可以讓多個用戶端共用它們）。

關於這項作業

保存的預先定義過濾器位於 `<data_dir>\archive\filters`（其中 `<data_dir>` 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）。

程序

1. 在 `<data_dir>\archive\filters` 中，尋找您想要還原的過濾器（AppScan Source 過濾器的副檔名是 `.off`）。
2. 將過濾器複製到 `<data_dir>\scanner_filters`。
3. 重新啟動 AppScan Source。

下一步

如果要瞭解如何管理過濾器（包括您還原的已保存過濾器），請參閱第 136 頁的『在「過濾器編輯器」視圖中建立及管理過濾器』。

建立及管理過濾器

AppScan Source 提供多種建立及使用過濾器的方法。「過濾器編輯器」視圖是建立過濾器的主視圖，它提供一組很健全的規則，您可以手動設定它們，然後將它們儲存在過濾器中。另外，「過濾器編輯器」視圖也提供了已建立過濾器的管理機制，您很容易修改或移除這些過濾器。另外，您也可以利用提供發現項目之圖形表示法的視圖，來過濾發現項目表格，然後在「過濾器編輯器」視圖中儲存這些過濾器。當您建立過濾器時，會更新其他視圖來反映過濾器內容。

- 『在「過濾器編輯器」視圖中建立、管理及套用過濾器』
- 『從「評量摘要」和「漏洞矩陣」視圖進行過濾』
- 『在「來源和接收槽」視圖中建立過濾器』

在「過濾器編輯器」視圖中建立、管理及套用過濾器

「過濾器編輯器」視圖可讓您指定過濾器規則來建立過濾器。您可以儲存、修改及移除在「過濾器編輯器」視圖中建立的過濾器。在此視圖中建立過濾器之後，可透過視圖中的下拉功能表套用它。請參閱第 136 頁的『在「過濾器編輯器」視圖中建立及管理過濾器』。

在 AppScan Source for Analysis 中，您可以將您已建立的過濾器分享到 AppScan Enterprise Server，也可以存取其他人分享的過濾器。在 AppScan Source for Development 中，如果您在伺服器模式下執行，則可以存取共用的過濾器。

註：在 AppScan Source for Development（Visual Studio 外掛程式）中，這個視圖是「編輯過濾器」視窗的一部分。

從「評量摘要」和「漏洞矩陣」視圖進行過濾

註：

- 「評量摘要」視圖在 macOS 上無法使用。
- 在 AppScan Source for Development（Visual Studio 外掛程式）中，這些視圖是「編輯過濾器」視窗的一部分。

「評量摘要」和「漏洞矩陣」視圖提供發現項目圖形表示法。在這些視圖中，發現項目是以不同的方式來分組。您可以選取這些群組來過濾發現項目表格，使它只顯示所選群組內的發現項目。這個方法所進行的任何過濾，都會自動反映在「過濾器編輯器」視圖中；接著您可以在這裡儲存過濾器設定。

在「來源和接收槽」視圖中建立過濾器

註：AppScan Source for Development（Visual Studio 外掛程式）中沒有「來源和接收槽」視圖。

「來源和接收槽」視圖可供您依據輸入及輸出追蹤來檢視及過濾發現項目。在這個視圖中進行的過濾，可以直接在這個視圖中儲存。建立過濾器時，您可以選擇立即將它套用至掃描結果。

請參閱第 140 頁的『在「來源和接收槽」視圖中建立過濾器』。

在「過濾器編輯器」視圖中建立及管理過濾器

在這個視圖中，您可以建立、編輯、儲存、刪除及管理過濾器。如果您使用 AppScan Source for Analysis，您可以共用過濾器及存取與其他人共用的過濾器。在 AppScan Source for Development 中，如果您使用伺服器模式且已登入 AppScan Enterprise Server，則可以存取共用的過濾器。

程序

1. 在第 278 頁的『「過濾器編輯器」視圖』工具列中，按一下**新建**。新的過濾器名稱是 `Untitled<-number>`（第一個新的未命名過濾器是 `Untitled`，下一個新的未命名過濾器是 `Untitled-1`，依此類推）。

   註：在 AppScan Source for Development（Visual Studio 外掛程式）中，這個視圖是「編輯過濾器」視窗的一部分。
2. 展開種類，然後選取過濾器需要的準則。
3. 按一下**儲存**或**另存新檔**。
4. 指定過濾器名稱，然後按一下**確定**。新的過濾器名稱會取代過濾器清單中的 `Untitled<-number>`。

下一步

如果要套用過濾器，請在「過濾器編輯器」視圖下拉功能表中選取它。

註：在「漏洞矩陣」視圖之外套用的過濾器，不會影響「漏洞矩陣」視圖。您必須選取「漏洞矩陣」視圖的**顯示過濾的發現項目計數**工具列按鈕，過濾器才會反映在「漏洞矩陣」視圖中。

您可以在清單中選取過濾器，再使用它，以便直接在「過濾器編輯器」視圖中管理過濾器；您也可以按一下**管理過濾器**來開啟「管理過濾器」對話框，它會提供一份已儲存過濾器的清單。

- **修改過濾器**：在「過濾器編輯器」視圖或「管理過濾器」對話框中，選取過濾器，然後修改它的過濾器規則並儲存變更。

  註：無法修改或刪除內建的過濾器。
- **刪除過濾器**：在「過濾器編輯器」視圖或「管理過濾器」對話框中，選取過濾器，然後按一下**刪除**。在「管理過濾器」對話框中，您可以選取多個過濾器，然後按一下**刪除**，將它們同時移除。
- **根據另一個過濾器來建立過濾器**：您可以修改過濾器，然後按一下**另存新檔**，將它儲存為新的過濾器名稱。如此一來，您便可以依照現有過濾器的設定來建置，以建立新的過濾器。您可以同時在「過濾器編輯器」視圖和「管理過濾器」對話框中，執行這個動作。

  提示：透過開啟過濾器並使用**另存新檔**動作以新名稱儲存過濾器，也可以完成相同的事項。然後，您就可以開啟新的過濾器並進行修改。選擇這個方法，您可以從其中一個內建過濾器來建立新的過濾器。
- **回復過濾器設定**：如果您修改過濾器的內容之後，想要復原這些變更，請按一下**回復**，使過濾器回到它前次儲存的設定。您可以同時在「過濾器編輯器」視圖和「管

理過濾器」對話框中，執行這個動作。在對話框中，如果多個過濾器有未儲存的變更，按一下**回復**之後，所有已選取且含有未儲存之變更的過濾器，都會回復它們已儲存的設定。

- **共用過濾器**（僅限 AppScan Source for Analysis）：如果要建立共用過濾器，請在「過濾器編輯器」中開啟一個過濾器，然後在「過濾器編輯器」視圖工具列中，按一下**共用過濾器**。

  註：如果要修改、刪除或建立共用過濾器，您必須具有**管理共用過濾器**許可權。如果要瞭解設定許可權的相關資訊，請參閱《*IBM Security AppScan Source* 安裝與管理手冊》。

從「評量摘要」視圖進行過濾

當掃描完成時，您可以在「評量摘要」視圖中查看它的發現項目（依預設，這個視圖會在「分類」視景中開啟）。在這個視圖中，您可以從長條圖建立過濾器。

關於這項作業

掃描完成之後，第 277 頁的『「評量摘要」視圖』含有發現項目的圖形長條圖表示法。您可以精簡這個視圖，依照漏洞類型、API、專案或檔案來顯示發現項目。當您在「評量摘要」視圖中選取分組的發現項目時，發現項目表格會改為只顯示「評量摘要」視圖中選取的發現項目。

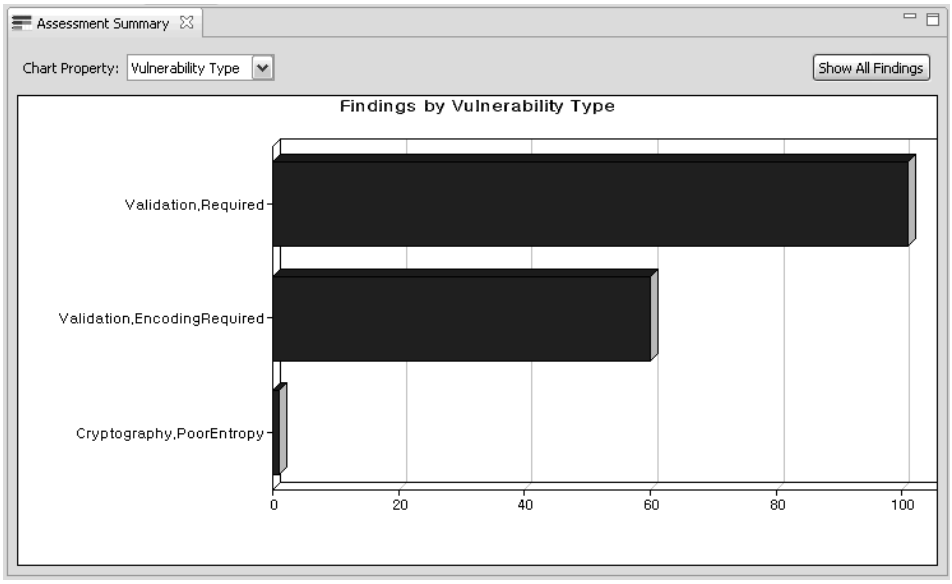
註：在「漏洞矩陣」視圖之外套用的過濾器，不會影響「漏洞矩陣」視圖。您必須選取「漏洞矩陣」視圖的**顯示過濾的發現項目計數**工具列按鈕，過濾器才會反映在「漏洞矩陣」視圖中。

註：

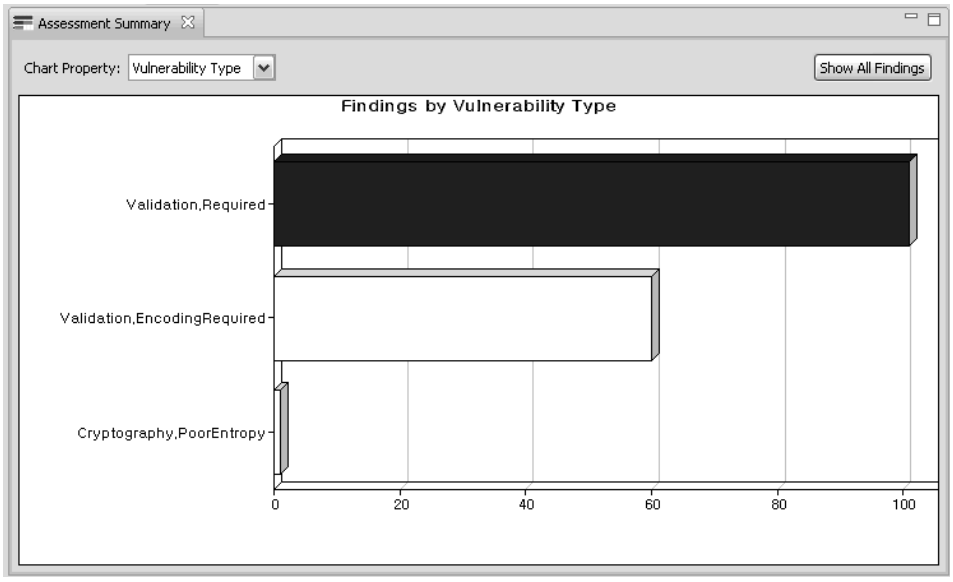
- 「評量摘要」視圖在 macOS 上無法使用。
- 在 AppScan Source for Development（Visual Studio 外掛程式）中，這個視圖是「編輯過濾器」視窗的一部分。

程序

1. 在「評量摘要」視圖中，變更圖形表示法來配合您的需求。例如，假設有一項評量包含 `Validation.Required`、`Validation.EncodingRequired` 和 `Cryptography.PoorEntropy` 漏洞類型，請將**圖表內容**設為**漏洞類型**。這時會在長條圖表示法中，依漏洞類型來顯示發現項目：

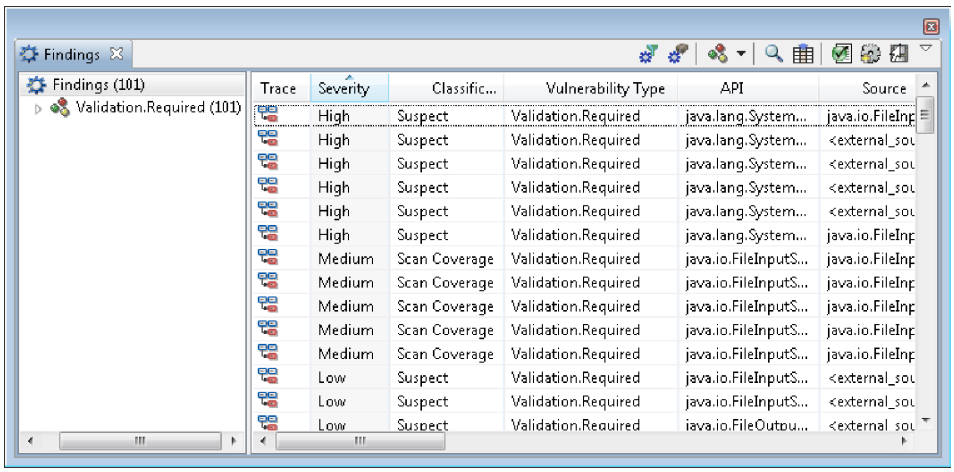


2. 如果要建立 `Validation.Required` 漏洞類型的過濾器，請按一下圖表中的 `Validation.Required` 長條圖。

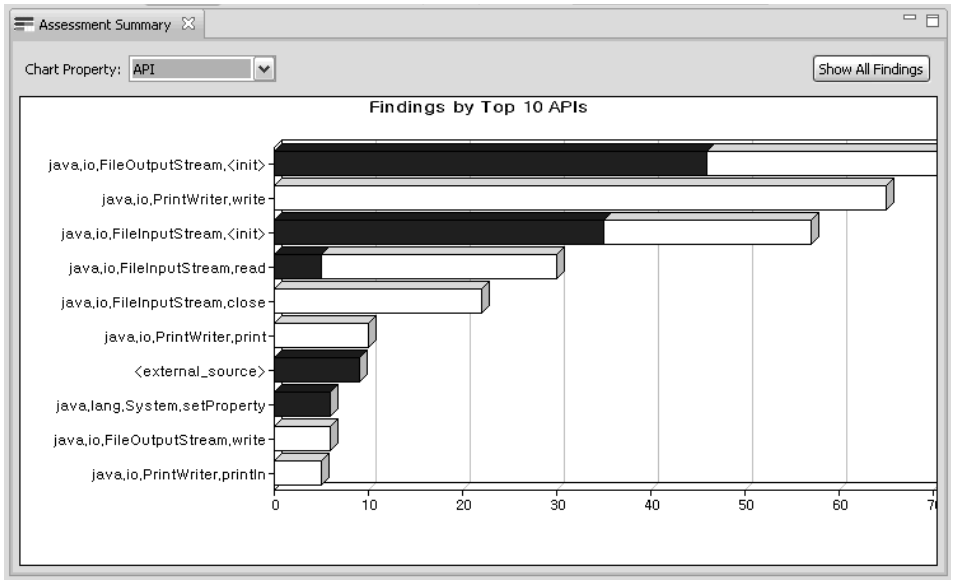


提示：請將滑鼠停在長條圖上，以查看漏洞的數目。

過濾的結果會出現在發現項目表格中：



3. 過濾動作也會將「評量摘要」視圖中選取的過濾器規則設定，移入「過濾器編輯器」視圖。這個過濾器可以儲存在「過濾器編輯器」視圖中（如果要瞭解過濾器規則設定及儲存過濾器的相關資訊，請參閱第 136 頁的『在「過濾器編輯器」視圖中建立及管理過濾器』）。
4. 如果要依 API 來檢視相同的過濾結果，請將**圖表內容**設為 **API**：



從漏洞矩陣過濾

「漏洞矩陣」視圖會顯示掃描所包含之所有應用程式的發現項目總數。在矩陣之中，這些發現項目是依嚴重性層次來分組。您可以選取這些發現項目群組來建立過濾器。

關於這項作業

當您在第 279 頁的『「漏洞矩陣」視圖』中選取分組的發現項目時，發現項目表格會改為只顯示「漏洞矩陣」中選取的發現項目。

註：在 AppScan Source for Development（Visual Studio 外掛程式）中，這個視圖是「編輯過濾器」視窗的一部分。

註：品質發現項目和以**參考資訊**嚴重性層次來分類的發現項目，不會併到「漏洞矩陣」視圖中。

程序

1. 在「漏洞矩陣」視圖中，選取您要在發現項目表格中查看的矩陣區段。比方說，如果只要在發現項目表格中查看**高**嚴重性**可疑**安全發現項目，請選取矩陣的這個區段。這會使過濾的結果出現在發現項目表格中。
2. 過濾動作也會將「漏洞矩陣」視圖中選取的過濾器規則設定，移入「過濾器編輯器」視圖。這個過濾器可以儲存在「過濾器編輯器」視圖中（如果要瞭解過濾器規則設定及儲存過濾器的相關資訊，請參閱第 136 頁的『在「過濾器編輯器」視圖中建立及管理過濾器』）。

在「來源和接收槽」視圖中建立過濾器

程序

1. 開啟或導覽至「來源和接收槽」視圖。
2. 「來源和接收槽」視圖包含三個區段。在視圖的發現項目表格區段中，會顯示您選擇在另外兩個區段中顯示的來源、接收槽和中間節點的發現項目。在第 272 頁的『「來源和接收槽」視圖』中，有相關說明。
3. 設定發現項目表格來顯示您感興趣的發現項目之後，請按一下**根據所選的來源、接收槽和中間節點來建立新的過濾器**。
4. 在「建立過濾器」對話框中，執行下列動作：
   - 在**名稱**欄位中指定過濾器名稱。
   - 選取**立即套用這個過濾器**，將這個過濾器套用於評量中的所有發現項目表格。選取這個勾選框，相當於在「過濾器編輯器」視圖中選取過濾器。它設定現行主要過濾器，會影響到所有視圖（如「漏洞矩陣」和「發現項目」視圖）。

     註：在「漏洞矩陣」視圖之外套用的過濾器，不會影響「漏洞矩陣」視圖。您必須選取「漏洞矩陣」視圖的**顯示過濾的發現項目計數**工具列按鈕，過濾器才會反映在「漏洞矩陣」視圖中。
   - 如果濾除的發現項目與現行工作無關，您可以選取**建立排除這些發現項目的應用程式過濾器**勾選框，將它們從評量中移除。選取這個勾選框會在應用程式內容中，將新過濾器新增為排除的過濾器（在應用程式的「內容」視圖中，選取「排除項目」標籤來查看已排除過濾器的清單）。在未來的應用程式掃描，符合這個過濾器的發現項目會報告在「已排除的發現項目」視圖中，而不是在「發現項目」視圖中。
5. 按一下**確定**來過濾或排除發現項目。

套用過濾器

可在掃描之前或之後套用過濾器。如果要在掃描之後套用過濾器，請使用「過濾器編輯器」或另一個可讓您套用過濾器的視圖。如果要在掃描之前套用過濾器，請使用掃描配置來設定廣域過濾器。當您在掃描之前套用過濾器時，無法顯示未過濾的發現項目，或沒有重新掃描就移除過濾器。

在掃描之前套用過濾器

請參閱『全域套用過濾器』，以瞭解如何設定廣域過濾器，或參閱第 96 頁的『管理掃描配置』，以瞭解如何在掃描配置中設定過濾器。

在掃描之後套用過濾器

當您在「過濾器編輯器」視圖中選取過濾器時，它會自動套用至發現項目的清單。其他視圖提供過濾動作，如第 135 頁的『建立及管理過濾器』所述。

全域套用過濾器

已建立的過濾器可套用至所有應用程式、個別應用程式及個別專案。廣域過濾器套用在「內容」視圖中 - 您可以在這裡指定如何套用過濾器（可直接套用過濾器 - 或套用它的反轉）。例如，若您想要設定應用程式的廣域過濾器，請在「瀏覽器」視圖中選取它，然後開啟它的「內容」視圖（使用**視圖**功能表，或用滑鼠右鍵按一下該應用程式並按一下**內容**）。

開始之前

如果您要為所有應用程式或個別專案設定過濾器，請使用「內容」視圖中的「過濾器」標籤。如果您要為個別應用程式設定過濾器，請使用「內容」視圖中的「排除和過濾」標籤。

程序

1. 在該標籤的「過濾器」區段中，按一下**新增**。
2. 在「選取過濾器」對話框中，選擇您想要全域套用的過濾器。
3. 選擇性的： 如果您想要套用過濾器的反轉（而非直接套用過濾器），請選取**反轉過濾器**。
4. 按一下**確定**，關閉「選取過濾器」對話框。
5. 當新增過濾器完成時，請在「內容」視圖中儲存變更。

判斷已套用的過濾器

過濾器可在掃描之後全域套用至應用程式和專案 - 或者，可以在掃描之後將它們套用至評量。為了讓您能夠很快判斷在評量中過濾器如何套用至發現項目，AppScan Source 在主要工作台底端提供過濾器指示器。

如果未套用過濾器，工作台底端的過濾器指示器會指出**未過濾發現項目**。

如果已套用過濾器，指示器會變成鏈結，指出**已過濾發現項目**。選取此鏈結會開啟訊息，可讓您判斷是如何套用過濾器：

- **掃描時間過濾器**是廣域過濾器，其套用至應用程式和專案：
  - 如果評量是掃描沒有配置過濾器的應用程式或專案的結果，則訊息指出未套用掃描時間過濾器。
  - 如果已掃描的應用程式或專案有配置過濾器，會依名稱列出所配置的過濾器。
  - 在某些情況下，AppScan Source 會發現已套用掃描時間過濾器，不過，評量並不包含那些過濾器的相關資訊。例如，當開啟舊評量時，便可能發生此情況。

- **現行過濾器**是在掃描之後已套用至發現項目的過濾器。訊息指出是否未套用現行過濾器 - 或是否已套用過濾器。如果是後者，**重設**鏈結便可供使用。選取此鏈結之後，會從發現項目中移除現行過濾器。

分類及排除項目

掃描之後，您可能決定某些發現項目與目前的工作無關，在分類掃描結果時，不要它們出現在發現項目表格中。這些排除項目（或已排除的發現項目）不會再出現在「發現項目」視圖中，變更結果會立即更新評量的度量。新增到配置中的過濾器和組合排除項目，只會影響到後續的掃描。

排除項目的範圍

排除項目適用於所有應用程式（廣域）、個別應用程式，或專案。

- **廣域排除項目**適用於所有的掃描。
- **應用程式排除項目**只適用於針對特定應用程式及其對應專案所執行的掃描。
- **專案排除項目**適用於存在於特定專案中的發現項目。

註：排除項目會影響評量的度量，其中包括發現項目總數（評量度量中不包含排除的發現項目）。

廣域排除項目

您可以從任何 AppScan Source for Analysis 應用程式中儲存或存取廣域排除項目，這些排除項目適用於所有掃描。只有共用過濾器會成為廣域過濾器。

應用程式和專案排除項目

組合排除項目只適用於應用程式。過濾器排除項目適用於應用程式或專案。適用於應用程式和專案的排除項目可以共用或位於本端。

指定排除項目

您可以從發現項目表格或「內容」視圖中，將發現項目標示為排除項目。排除項目可由個別發現項目、過濾器或組合組成。從發現項目表格建立的排除項目通常會立即生效。在「內容」視圖中建立的排除項目，需要再掃描一次，才會生效。

在下列程序期間，排除項目會立即套用於應用程式：

- 選取一或多個發現項目，用滑鼠右鍵按一下選項，然後從功能表中選取**排除發現項目**。
- 新增一或多個發現項目到目前已排除的組合中，其中包括已排除的組合。
- 從先前已排除的組合中（包括已排除的組合），刪除一或多個發現項目。
- 刪除已排除的組合。

在下列情況下，排除項目不會立即套用於應用程式：

- 新增組合作為排除項目。
- 新增過濾器作為排除項目。
- 修改發現項目，使它符合已排除之過濾器的準則。
- 修改發現項目，使它不再符合已排除之過濾器的準則。

在發現項目表格中，將發現項目標示為排除項目

程序

1. 請在發現項目表格中，選取您認為不重要或不想看到的發現項目（或一組發現項目）。
2. 用滑鼠右鍵按一下選項，然後從功能表中，選擇**排除發現項目**。排除項目會立即套用。已排除的發現項目不再出現在表格中，各項度量也會立即更新。

結果

如果要檢視已排除的發現項目，請開啟「已排除的發現項目」視圖。已排除的發現項目也會出現在名稱為**排除的組合**的組合中。

如果要重新併入已排除的發現項目，請遵循『重新併入已標示排除的發現項目』中的指示。

重新併入已標示排除的發現項目

排除的發現項目會出現在「已排除的發現項目」視圖中。從這個視圖中，您可以重新併入已排除的發現項目。

程序

1. 在「已排除的發現項目」視圖中，選取您想要重新併入的發現項目（或發現項目群組）。
2. 用滑鼠右鍵按一下選項，然後從功能表中，選擇**併入發現項目**。

結果

併入的發現項目重新加到評量中，這時會立即更新發現項目表格和度量來反映重新併入的發現項目。發現項目不再出現於「已排除的發現項目」視圖中。

註：在 AppScan Source for Analysis 中，您也可以重新併入從**已排除**組合視圖排除的發現項目，作法是將發現項目從組合中移除，或是將它們移至未排除的組合中。

範例：指定過濾器排除項目

過濾準則決定了過濾器要排除符合或不符合過濾器的發現項目。

這些範例說明如何建立用來排除發現項目的過濾器：

- 『範例：過濾和排除目錄』
- 第 144 頁的『範例：過濾和排除 API』

範例：過濾和排除目錄

在這個範例中，會建立過濾器，以便只顯示含有 Microsoft 併入檔的發現項目。之後，就利用這個過濾器來縮小發現項目清單的範圍（我們會排除所有符合過濾器的發現項目）。

程序

1. 在「過濾器編輯器」視圖的**目錄**區段中，新增 Microsoft 併入檔的路徑（例如 `C:\Program Files\Microsoft Visual Studio 8\VC\include`）。
2. 選取**限於**，讓這個成為一項內含規則。

3. 在「發現項目」視圖工具列中，按一下**顯示不符合過濾器的發現項目**，只查看 Microsoft 標頭檔的發現項目。這樣做可讓您瞭解，在全域套用過濾器反轉及重新掃描之後，掃描結果會是如何。
4. 以 **MS 併入項目**之類的名稱來儲存這個過濾器。
5. 回到「配置」視景，並在「瀏覽器」視圖中，選取 C/C++ 應用程式或專案。
6. 如果選取應用程式，請開啟「內容」視圖的「排除和過濾」標籤。如果選取專案，請開啟「內容」視圖的「過濾」標籤。按一下**新增**。選取 **MS 併入項目**，然後選取**反轉過濾器**。
7. 儲存您在「內容」視圖中的變更，然後重新掃描應用程式或專案。
8. 回到「分類」。排除的發現項目會出現在「已排除的發現項目」視圖中。

範例：過濾和排除 API

這是在分類程序的初期，當您想要設定發現項目的優先順序，且想要排除一些特定發現項目時，所可能出現的一般分類實務。例如，您判定有三個 API 不是威脅，您想要從後續的掃描中排除這些 API。

程序

1. 在「過濾器編輯器」的 API 區段中，按一下**新增**，然後選取三個 API。
2. 選取**限於**。
3. 儲存及命名過濾器。
4. 回到「配置」視景，並在「瀏覽器」視圖中，選取專案（或應用程式）。
5. 在「內容」視圖中，將過濾器的行為設為**反轉**（在「選取過濾器」對話框中，選取**反轉過濾器**）。
6. 重新掃描。過濾器中的 API不會再出現在發現項目中。

結果

使用相同的範例，您可以只看到包含在過濾器中的發現項目。在這個實例中，將過濾器新增到該清單時，請勿選取**反轉過濾器**。當重新掃描時，只會出現過濾器中的發現項目。

從「內容」視圖指定組合排除項目

組合排除項目會除去組合中的發現項目。您只能將組合從應用程式排除。

程序

1. 依照第 145 頁的『建立組合』所說明來建立組合。
2. 在「瀏覽器」視圖中，選取要關聯於組合的應用程式。
3. 在「內容」視圖中，選取**排除項目**標籤。
4. 按一下**新增組合**，在「選取組合」對話框中，選取包含要從應用程式中排除之發現項目的組合。
5. 按一下**確定**。
6. 重新掃描。組合的發現項目不再出現於發現項目表格中。

利用組合分類

對您的分類程序而言，具有唯一性質的組合，可能很重要。

關於這項作業

- 組合匯出到問題追蹤系統的形式，可以是單一問題報告，也可以是組合中每個發現項目各一份問題報告。
- 組合可作為產生報告的基礎。
- 組合會附加到應用程式。

重要：一個發現項目，每次只能出現在單一組合中。如果發現項目在某個組合中，將它移到另一個組合，會將它從第一個組合中移除。

這個範例概述使用組合的簡式分類：

程序

1. 掃描原始碼。
2. 建立名稱為解析 ASAP 的組合。
3. 新增一些重要的發現項目到組合中。
4. 新增附註到組合內的發現項目中。
5. 將組合或發現項目提交給問題追蹤系統，或用電子郵件將它們傳給其他開發人員。
6. 修正問題。

建立組合

建立組合是在「組合」視圖或含有發現項目表格的視圖中進行。您可以新增發現項目到現有的組合或新的組合中。

這些主題說明在「組合」和「發現項目」視圖中建立組合：

- 『在「組合」視圖中建立新的組合』
- 第 146 頁的『在「發現項目」視圖中建立新的組合』

註：如果要能夠建立評量的組合，所掃描以建立評量的應用程式必須載入 AppScan Source for Analysis 中。如果您開啟未載入之應用程式的評量，建立組合的動作將無法使用。

建立一或多個組合之後，「發現項目」視圖的**隱藏組合的發現項目**動作 ( ) 可讓您在視圖中切換顯示組合的發現項目。這個動作會隱藏您建立之所有已併入組合中的發現項目。這個設定不會影響已排除之組合中發現項目的顯示 - 這些發現項目絕不會出現在「發現項目」視圖中。

在「組合」視圖中建立新的組合

程序

1. 在「組合」視圖的工具列中，按一下**新建組合**。
2. 指定組合名稱，然後按一下**確定**。組合名稱會出現在「組合」視圖中。
3. 如果要新增發現項目至組合，請遵循第 146 頁的『新增發現項目至現有的組合中』中的指示。

在「發現項目」視圖中建立新的組合

程序

1. 在「發現項目」視圖中，選取要新增到組合的發現項目。
2. 用滑鼠右鍵按一下選項，然後從功能表中，選擇**新增至組合** > **新建**。
3. 指定組合名稱，然後按一下**確定**。

新增發現項目至現有的組合中

關於這項作業

您可以從多個視圖，將發現項目新增到組合中：

- 「發現項目」視圖
- 「已排除的發現項目」視圖
- 「修正/修改的發現項目」視圖
- 「遺漏的發現項目」視圖
- 「報告」視圖
- 發現項目詳細資料

提示：您可以利用拖放作業，將發現項目從發現項目表格移至「組合」視圖。

如果要新增發現項目至組合中，請執行下列動作：

程序

1. 選取要新增到組合中的發現項目。
2. 用滑鼠右鍵按一下選項，然後從功能表中，選擇**新增至組合** > **<bundle name>**（這份清單含有 5 個最近建立的組合）或**新增至組合** > **選取**。
3. 如果您選擇**新增至組合** > **選取**，請在「選取組合」對話框中，選取發現項目要新增到其中的組合，然後按一下**確定**。

在組合之間移動發現項目

程序

1. 在「組合」視圖中，開啟包含您要移動之一或多個發現項目的組合。
2. 選取您要移動的一或多個發現項目，然後完成下列動作之一：
   - 在視圖工具列中，按一下**移至組合**或**移至新的組合**。然後選取您要將發現項目移至的那一個組合，或是為發現項目建立新的組合。
   - 用滑鼠右鍵按一下選項，然後按一下**移至組合**。這會開啟功能表，讓您從清單或對話框中選取現有的組合，或建立新組合以成為選項的移動目標。

結果

註：移到或新增到已排除組合的發現項目，不會在現行評量中排除。如果要在現行評量中，將發現項目標示為已排除，請使用**排除發現項目**動作。

檢視組合中的發現項目

當您新增發現項目到組合中，發現項目會成為組合中的一列。如果開啟組合，您會見到組合所包含的所有發現項目。

關於這項作業

來自組合中多個專案的發現項目，可能會用不同的方式來呈現。如果在最新的掃描中找不到組合中的發現項目，則該發現項目會以綠色斜體呈現。

請看下列範例的「X 應用程式」。

程序

1. X 應用程式包含 A 專案和 B 專案。
2. 掃描 X 應用程式。
3. 建立包含 A 專案和 B 專案發現項目的組合。
4. 掃描 B 專案。在「組合」視圖中，出現 B 專案的發現項目，A 專案的發現項目以綠色斜體呈現。

結果

以綠色斜體強調顯示的發現項目是已修正/遺漏的發現項目。已修正/遺漏的發現項目是在組合中，但不在現行評量中的發現項目。將發現項目識別為已修正/遺漏，是因為已將它解決、移除，或未掃描原始檔。在「組合」視圖中，**已排除**直欄指出組合是否已排除。

將組合儲存至檔案

您可以將組合儲存成檔案，以便在 AppScan Source for Development 中開啟。組合也可讓您從 AppScan Source for Analysis 中，將發現項目的 Snapshot 匯入 AppScan Source for Remediation 中。

程序

1. 完成下列動作之一：
   a. 在「組合」視圖中，選取組合，並在工具列中按一下**將組合儲存至檔案**。
   b. 開啟組合，並按一下工具列中的**將組合儲存至檔案**。
2. 選取用來儲存組合檔的目錄。
3. 指定組合檔的名稱 (`<file_name>.ozbdl`)。

結果

如果要開啟已儲存的組合，請執行下列動作：

- 在 AppScan Source for Development（Eclipse 外掛程式）中，選取**安全分析** > **開啟** > **開啟組合**。
- 在 AppScan Source for Development（Microsoft Visual Studio 外掛程式）中，選取 **IBM Security AppScan Source** > **開啟組合**。
- 在 AppScan Source for Analysis 中，按一下「組合」視圖工具列中的**開啟組合**。

提示：在 Windows 系統中，按兩下「組合」視圖中的組合檔，以便在 AppScan Source for Analysis 或 AppScan Source for Development 中加以開啟。

提交組合進行問題追蹤，並利用電子郵件提交

組合中的發現項目可以提交給公司的問題追蹤系統，或利用電子郵件傳送。只要將發現項目放在組合中，您就可以將這些發現項目當作錯誤來提交開發人員進行補救。

程序

1. 開啟組合。
2. 按一下**提交組合進行問題追蹤**工具列按鈕下移鍵，然後選取您的問題追蹤系統。

   註：視您的問題追蹤系統而定，在提交組合之前，您可能希望修改「問題追蹤系統」喜好設定。

   或者，在「組合」工具列上，按一下**用電子郵件傳送組合**，將組合傳送給其他人（電子郵件喜好設定必須事先配置）。
3. 完成開啟的配置對話框。這些會因您選擇的問題追蹤系統而不同；請參閱說明中的 *AppScan Source for Analysis* 與問題追蹤區段。

新增附註至組合

程序

1. 在「組合」視圖中，選取要標註的組合。
2. 在「組合」工具列中，按一下**新增附註**，或用滑鼠右鍵按一下選項，然後在功能表中，選擇**新增附註**。
3. 輸入附註，然後按一下**確定**。

修改發現項目

已修改的發現項目是漏洞類型、分類、嚴重性有了改變，或擁有註釋的發現項目。「已修改的發現項目」視圖會顯示現行應用程式（因開啟其評量而在作用中的應用程式）的這些發現項目。在「我的評量」視圖（只限 AppScan Source for Analysis）中，**已修改**直欄會指出發現項目是否在現行評量中有了改變。

發現項目的修改具有即時性，它會更新度量。修改部分會與應用程式一併儲存，且會套用於應用程式未來的掃描。

您可以在「發現項目詳細資料」視圖中，或從任何含有發現項目表格的視圖中，修改發現項目。「發現項目詳細資料」視圖可讓您修改個別的發現項目，或者，您可以修改發現項目表格中的多個發現項目。

註：您必須具備**儲存評量**許可權，才能在修改評量之後儲存變更。

從發現項目表格修改

如果您要對多個檔案做相同的變更，您可能會想要透過發現項目表格來修改發現項目。如果您將修改個別的發現項目，請使用發現項目表格或「發現項目詳細資料」視圖。

- 第 149 頁的『變更漏洞類型』
- 第 149 頁的『將發現項目分類升級』
- 第 149 頁的『修改嚴重性』
- 第 158 頁的『支援的註釋和屬性』

變更漏洞類型

您可以變更個別發現項目或是一組發現項目的漏洞類型。

程序

1. 從發現項目表格中，選取要修改的發現項目或一組發現項目。
2. 用滑鼠右鍵按一下選項，然後從功能表中，選擇**設定漏洞類型**。
3. 在「選取漏洞類型」對話框中，選擇您要的漏洞類型，然後按一下**確定**。

將發現項目分類升級

其分類為可疑安全發現項目或掃描涵蓋面發現項目的發現項目，可升級為明確的發現項目。

程序

1. 從發現項目表格中，選取要修改的發現項目或一組發現項目。
2. 用滑鼠右鍵按一下選項，然後從功能表中選擇**升級為明確**。

修改嚴重性

選取新的嚴重性層次，會變更每個所選發現項目的嚴重性。例如，AppScan Source 可能會報告某個 API 的嚴重性是中度，但公司的原則將它視為更嚴重。您可以修改嚴重性來符合您的需求，但請注意，AppScan Source 的補救協助不包含這項修改。

程序

1. 從發現項目表格中，選取要修改的發現項目或一組發現項目。
2. 用滑鼠右鍵按一下選項，然後從功能表中，選擇**設定嚴重性**。
3. 選取**高**、**中**、**低**或**參考資訊**作為新的嚴重性層次。

標註發現項目

附註可做為提示，以提示您對發現項目採取進一步動作，或將發現項目的相關資訊傳給其他人。您可以新增附註到單一發現項目或一組發現項目。

程序

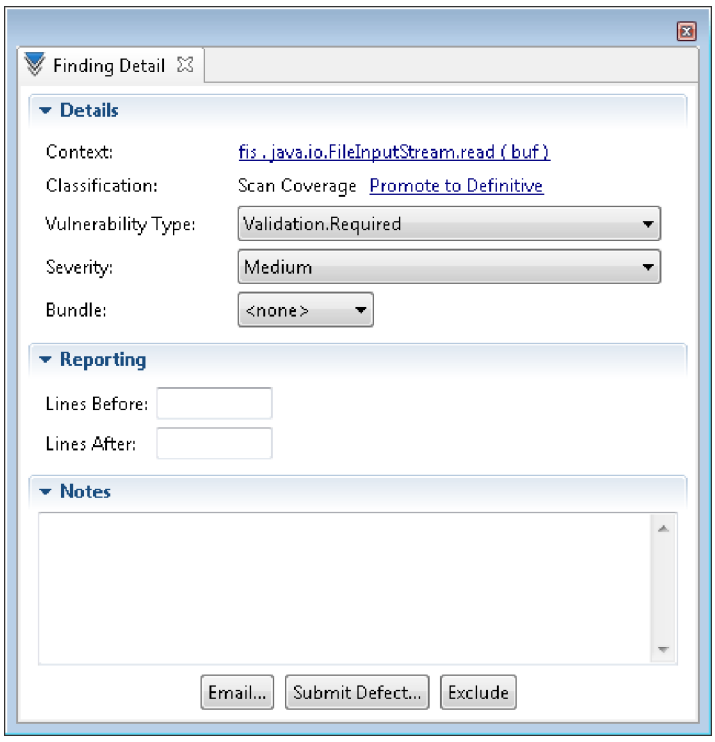
1. 從發現項目表格中，選取要修改的發現項目或一組發現項目。
2. 用滑鼠右鍵按一下選項，然後從功能表中，選擇**新增附註**。
3. 輸入附註，然後按一下**確定**。

在「發現項目詳細資料」視圖中修改發現項目

您可以在「發現項目詳細資料」視圖中，修改個別的發現項目。如果您在表格中選取某個發現項目，然後開啟「發現項目詳細資料」視圖，就會出現所選的發現項目及其性質。

「發現項目詳細資料」視圖

選取發現項目時，隨即會顯示「發現項目詳細資料」視圖，可讓您修改發現項目的內容。使用這個視圖，您可以修改個別的發現項目。



- 『「詳細資料」區段』
- 『報告區段（僅限 AppScan Source for Analysis 和 AppScan Source for Development（Eclipse 外掛程式））』
- 『附註區段』
- 第 151 頁的『「發現項目詳細資料」視圖動作』
- 第 151 頁的『自訂發現項目的「發現項目詳細資料」視圖（只限 AppScan Source for Analysis）』

「詳細資料」區段

- **環境定義**：漏洞週遭的程式碼片段
- **分類**：明確或可疑安全發現項目 - 或掃描涵蓋面發現項目 - 有可升級為**明確**發現項目或回復至原始值（如果分類已變更）的鏈結
- **漏洞類型**
- **嚴重性**：高、中、低或參考資訊。
- **組合**：包含發現項目的組合名稱（AppScan Source for Development（Visual Studio 外掛程式）中沒有這個選項）

報告區段（僅限 AppScan Source for Analysis 和 AppScan Source for Development（Eclipse 外掛程式））

指定報告中要於發現項目之前和/或之後併入的程式碼行數。

附註區段

標註發現項目。

「發現項目詳細資料」視圖動作

- **排除**：請按一下**排除**來排除（移除）發現項目表格中的發現項目。如果要檢視已排除的發現項目，請開啟「已排除的發現項目」視圖。
- 只在 AppScan Source for Analysis 中才有：
  - **電子郵件**：如果您已配置電子郵件喜好設定，您可以利用電子郵件，將發現項目組合直接傳送給開發人員，告訴他們在掃描之後所發現的可能問題報告。電子郵件包含含有發現項目的組合附件，以及說明發現項目的電子郵件文字。
    1. 如果要用電子郵件傳送「發現項目詳細資料」視圖中的現行發現項目，請按一下**電子郵件**。
    2. 在「附件檔名」對話框中，指定要附加到電子郵件的發現項目組合名稱。例如，在**附件檔名**欄位中指定 `my_finding`，會將檔名為 `my_finding.ozbdl` 的組合附加到電子郵件中。
    3. 按一下**確定**來開啟「以電子郵件傳送發現項目」對話框。依預設，「以電子郵件傳送發現項目」對話框中的**郵件收件者**欄位，會移入電子郵件喜好設定所指定的**收件者位址**，不過，準備電子郵件時，很容易改變它。請在這個對話框中，檢閱電子郵件的內容，然後按一下**確定**來傳送電子郵件。
  - **提交問題報告**：如果要將發現項目當作問題報告來提交，請按一下**提交問題報告**。這時會開啟「選取問題追蹤系統」對話框。
    - 如果您選取 **ClearQuest**，然後按一下**確定**，即會開啟「附件檔名」對話框。請在這個對話框中，指定要附加到問題報告的發現項目組合名稱，然後按一下**確定**。登入 Rational ClearQuest，提交發現項目。
    - 如果您選取 **Quality Center**，然後按一下**確定**，這時會開啟「登入」對話框，供您登入 Quality Center 來提交發現項目。
    - 如果您選取 **Team Foundation Server** 選項，這時會開啟對話框，提示您登入問題追蹤系統，並提供其他配置詳細資料。

      註：Rational Team Concert 是 macOS 上唯一支援的問題報告追蹤系統。

自訂發現項目的「發現項目詳細資料」視圖（只限 AppScan Source for Analysis）

在自訂發現項目的「發現項目詳細資料」視圖中，有其他可供編輯的資訊：

- 檔案
- 行
- 直欄
- API

此外，您編輯 第 150 頁的『「詳細資料」區段』 的方法不同於部分欄位的標準發現項目（例如，自訂發現項目出現在清單中的分類）。

移除發現項目修改部分

如果您修改了發現項目，您可以利用這個主題所說明的方法來移除修改的部分（回復原始值）。

關於這項作業

您可以利用許多方法來移除發現項目的修改部分：

- 『在「已修改的發現項目」視圖中，移除修改部分』：這個方法要求您想要移除其修改部分的應用程式，其評量必須已經開啟。當您想要回復多個修改過的發現項目時，它非常有用。
- 『在含有發現項目的其他視圖中移除修改部分』：這個方法需要一份已開啟的評量，如果某發現項目已進行多重修改，而您想要回復一部分的變更時，這尤其有用。比方說，如果您已變更發現項目的嚴重性和分類，且您既想回復原始嚴重性，又想保持已修改的分類，這個方法最合適。
- 『在「內容」視圖的「已修改的發現項目」標籤中，移除修改部分（只限 AppScan Source for Analysis）』：如果您想要移除未開啟評量之應用程式的修改部分，這個方法很有用，它可用來回復多個已修改的發現項目。

在「已修改的發現項目」視圖中，移除修改部分

程序

1. 在「已修改的發現項目」視圖中，選取您想要回復的已修改的發現項目。您可以利用 Windows 鍵盤上的 Ctrl 和 Shift 鍵，或 macOS 上的 command 鍵和 Shift 鍵來選取多個發現項目。
2. 按一下**刪除修改部分**，或用滑鼠右鍵按一下選項，然後從功能表中選擇**刪除修改部分**。

結果

這個動作會移除已對發現項目進行的所有修改部分。如果已對某發現項目進行多重修改，而您想要回復一部分的變更時，請使用『在含有發現項目的其他視圖中移除修改部分』中所說明的方法。

在含有發現項目的其他視圖中移除修改部分

關於這項作業

在任何包含發現項目表格的視圖中，您可以利用**選取直欄及進行排序**動作來選擇要顯示的直欄。您可以利用這個特性來顯示**嚴重性（原始）**、**嚴重性（自訂）**、**分類（原始）**及**分類（自訂）**直欄。這些直欄可協助您將修改部分回復其原始值（通過發現項目表格中的動作，或使用「發現項目詳細資料」視圖）。例如，某個發現項目的**嚴重性**或**嚴重性（自訂）**值是**高**，**嚴重性（原始）**值是**中** - 您可以利用各種方法，將嚴重性層次回復為**中**，例如：

- 在發現項目表格中，用滑鼠右鍵按一下發現項目，然後在功能表中選擇**設定嚴重性** > **中**。
- 選取發現項目，然後在「發現項目詳細資料」視圖中，將**嚴重性**欄位設為**中**。

在「內容」視圖的「已修改的發現項目」標籤中，移除修改部分（只限 AppScan Source for Analysis）

程序

1. 在「瀏覽器」視圖中，選取含有您想要移除修改部分的應用程式。
2. 在「已修改的發現項目」視圖中，選取您想要回復的已修改的發現項目。您可以利用鍵盤的 Ctrl 和 Shift 鍵來選取多個發現項目。
3. 按一下**刪除修改部分**，或用滑鼠右鍵按一下選項，然後從功能表中選擇**刪除修改部分**。

結果

這個動作會移除已對發現項目進行的所有修改部分。如果已對某發現項目進行多重修改，而您想要回復一部分的變更時，請使用第 152 頁的『在含有發現項目的其他視圖中移除修改部分』中所說明的方法。

比較發現項目

使用**差異評量**動作，來比較評量。比較兩項評量時，會將兩者之間的差異顯示在「評量差異」視圖中。這個視圖顯示新的、已修正/遺漏和共同的發現項目。

「評量差異」視圖中提供的控制項如下：

- **差異評量**：顯示兩個所選評量之間的差異。
- **新的發現項目**（藍色）：請利用這個工具列按鈕來切換顯示新的發現項目（藍色而非綠色標籤的評量中的發現項目）。
- **已修正/遺漏的發現項目**（綠色）：請利用這個工具列按鈕來切換顯示已修正/遺漏的發現項目（綠色而非藍色標籤的評量中的發現項目）。
- **共用**（白色）：請使用這個工具列按鈕，來切換顯示兩項評量之間共用的發現項目。
- **下一個**：移至下一個新的或已修正/遺漏的發現項目區塊。
- **上一個**：移至上一個新的或已修正/遺漏的發現項目區塊

在「評量差異」視圖中比較兩項評量

程序

1. 在左窗格中，選取兩項要比較的評量。
2. 按一下**差異評量**工具列按鈕，或用滑鼠右鍵按一下選項，並從功能表中，選擇**差異評量**。

從主功能表列來比較兩項評量

程序

1. 在主功能表列中選取**工具** > **差異評量**。
2. 在「差異評量」對話框中，選取兩項評量。
3. 按一下**確定**，在「評量差異」視圖中，開啟兩項評量的比較。

在「我的評量」和「已發佈的評量」視圖中尋找評量之間的差異

程序

1. 在兩個視圖其中之一，選取兩項評量。
2. 按一下**差異評量**工具列按鈕，或用滑鼠右鍵按一下選項，然後從功能表中，選擇**差異評量**。這會在「評量差異」視圖中，開啟兩項評量的比較。

自訂發現項目

如果要加強您的分析結果，您可以建立自訂發現項目。這些是使用者建立的發現項目，AppScan Source for Analysis 會將它們新增到目前開啟的評量中，或新增到所選的應用程式中。自訂發現項目會影響評量的度量，可以併到報告中。建立好之後，自訂發現項目會自動併入應用程式未來的掃描中。

自訂發現項目的行為相依於建立它的視圖。

當從「發現項目」視圖建立時，自訂發現項目的行為如下：

- 適用於目前開啟的評量。
- 另存成應用程式的一部分，並且會出現在應用程式內容中。
- 影響相同應用程式的現行掃描和未來的掃描。
- 立即影響評量的度量。

當從「內容」視圖建立時，或針對所選應用程式選取**新增自訂發現項目**動作來建立時，自訂發現項目的行為如下：

- 適用於所選的應用程式。
- 如果應用程式是所掃描的應用程式，就新增到現行評量中。
- 包含在這個應用程式未來的掃描中。

當從程式碼編輯器建立時：

- 如果開啟了某個評量，自訂發現項目會依照在「發現項目」視圖中建立的方式來運作。
- 如果未開啟任何評量，自訂發現項目會依照在「內容」視圖中建立的方式來運作。

建立好自訂發現項目之後，AppScan Source for Analysis 會自動儲存應用程式。不修改應用程式，就無法修改評量。不過，如果評量沒有相關的應用程式，就不會修改任何應用程式。

如果您新增自訂發現項目到應用程式中，它們會包括在該應用程式的後續掃描中，無法排除。如果要移除自訂發現項目，您必須將它從評量中排除，或從應用程式中刪除。

註： 自訂發現項目不能已修正/遺漏。

自訂發現項目由下列屬性組成：

- **漏洞類型**（必要）
- **嚴重性**（必要）
- **分類**（必要）
- **檔案**（必要）
- **環境定義**
- **行號**
- **直欄號碼**
- **API**
- **附註**
- **組合**

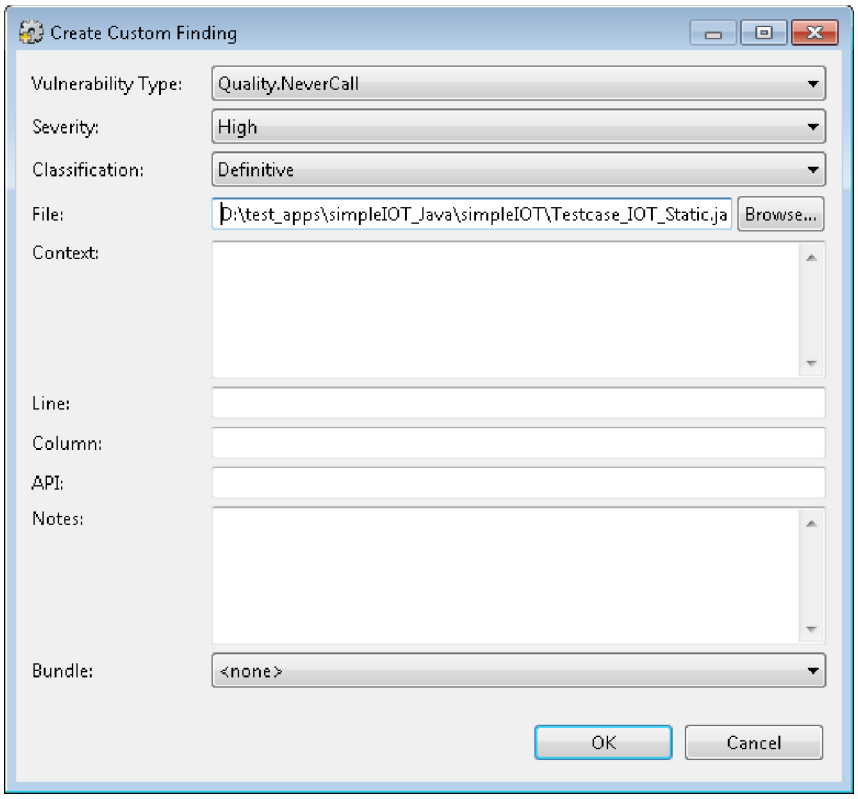
在「內容」視圖中建立自訂發現項目

如果從應用程式的「內容」視圖建立或編輯自訂發現項目，只會影響現行評量結果及未來的掃描。

程序

1. 在「瀏覽器」視圖中選取應用程式。
2. 在「內容」視圖中，選取**自訂發現項目**標籤。
3. 在工具列中，按一下**建立自訂發現項目**。
4. 在「建立自訂發現項目」對話框中，新增必要的項目：
   - **漏洞類型**
   - **嚴重性**
   - **分類**
   - **檔案**

   您可以選擇性地新增環境定義、行號、直欄、API、附註和組合指定。



5. 按一下**確定**，將自訂發現項目儲存到應用程式中。

在「內容」視圖中修改或移除自訂發現項目

如果從應用程式的「內容」視圖建立或編輯自訂發現項目，只會影響現行評量結果及未來的掃描。

程序

1. 選取發現項目。如果您要移除自訂發現項目，您可以選取要刪除的發現項目群組。

2. 如果要修改自訂發現項目，請按一下工具列上的**編輯選取的發現項目**，然後修改先前定義的發現項目資訊。
3. 如果要移除一或多個自訂發現項目，請按一下工具列上的**刪除選取的發現項目**。

在發現項目視圖中建立自訂發現項目

您可以從多個發現項目視圖中（例如：「發現項目」視圖和「自訂發現項目」視圖），建立或管理自訂發現項目。

從視圖中建立自訂發現項目，會新增發現項目到現行評量中，以及更新評量的度量。

在發現項目視圖中新增自訂發現項目時，請按一下視圖的**建立自訂發現項目**工具列按鈕。這時會開啟「建立自訂發現項目」對話框，您可以依照第 155 頁的『在「內容」視圖中建立自訂發現項目』的說明，以相同方式來完成它。

如果要移除自訂發現項目，您必須將它從評量中排除，或從應用程式中刪除 - 或遵循第 155 頁的『在「內容」視圖中修改或移除自訂發現項目』中的指示。其他發現項目視圖中不會提供這些動作。

在原始碼編輯器中建立自訂發現項目

關於這項作業

當您使用原始碼編輯器來新增自訂發現項目時，會套用下列條件：

- 如果原始碼編輯器中可見的原始檔屬於目前開啟的評量，自訂發現項目會新增到評量及相關聯的應用程式中。
- 如果自訂發現項目不屬於目前開啟的評量，自訂發現項目只會新增到包含原始檔的應用程式中。
- 如果原始檔屬於多個應用程式，或 AppScan Source for Analysis 無法判斷應用程式，您必須選取適當的應用程式。

如果您從原始碼編輯器建立自訂發現項目，在「建立自訂發現項目」對話框中，會預先移入編輯器中的資訊。

- **檔案**：目前開啟之檔案的名稱
- **環境定義**：編輯器中任何已選取的文字。如果未選取文字，環境定義就是目前游標位置所在的行。如果選取多行，所選各行會全部成為環境定義。
- **行**號和**直欄**號碼：目前的行號和直欄號碼

如果要從編輯器建立自訂發現項目，請執行下列動作：

程序

1. 選取要新增為自訂發現項目的各行程式碼。
2. 用滑鼠右鍵按一下選項，從功能表中，選取**建立自訂發現項目**。在「建立自訂發現項目」對話框中，會移入檔案、環境定義、直欄號碼和行號。
3. 選取**漏洞類型**、**嚴重性**和**分類**。您也可以選擇性地新增 API、附註或組合指定。
4. 按一下**確定**。

解決安全問題和檢視補救協助

AppScan Source 會發出安全錯誤或一般設計缺失的警示，且能夠在解決過程中提供協助。 AppScan Source 安全知識庫（以及內部或外部的程式碼編輯器）可以在這個過程中提供協助。

關於這項作業

AppScan Source 安全知識庫 提供更正發現項目的建議。本環境定義內每一漏洞的知識，提供了關於主要原因、風險嚴重性和補救建議動作的精闢說明。例如，它將「緩衝區溢位」類型 `strcpy()` 描述為高度嚴重性層次，且提供此補救協助：

> `strcpy` 很容易造成目的地緩衝區溢位，因為它不知道目的地緩衝區的長度，因此無法檢查以確定它不會改寫這個緩衝區。您應該考慮使用含有長度參數的 `strncpy`。 `strncpy` 的安全風險程度雖然較低，但還是有風險。

如果要檢視 AppScan Source 安全知識庫，請執行下列動作：

程序

- 在 AppScan Source for Analysis 中，開啟「補救協助」視圖，然後在發現項目表格中選取發現項目。這時會顯示這個特定發現項目的補救協助。或者，可以從主功能表列選取**說明** > **安全知識庫**，在瀏覽器中開啟整個 AppScan Source 安全知識庫。
- 在 AppScan Source for Development（Eclipse 外掛程式） 中，開啟「補救協助」視圖，然後在發現項目表格中選取發現項目。這時會顯示這個特定發現項目的補救協助。
- 在 AppScan Source for Development（Visual Studio 外掛程式）中，選取發現項目表格中的發現項目。從主功能表列選取 **IBM Security AppScan Source** > **知識庫說明**，或用滑鼠右鍵按一下發現項目，然後從功能表中選取**知識庫說明**。這會開啟所選發現項目的補救協助。

在編輯器中分析原始碼

AppScan Source 可讓您在內部編輯器中分析或修改原始碼，或者，您也可以從各種外部編輯器中選擇。

外部編輯器能讓您檢閱 AppScan Source for Analysis 中的結果，並且在您選擇的開發環境中進行程式碼修改。外部編輯器包含：

表 *13.* 支援的外部編輯器

編輯器	平台
Eclipse（請參閱AppScan Source系統需求，以瞭解支援哪些版本的 Eclipse）	Windows 和 Linux
記事本	Windows
vi	Linux
系統預設值	Windows 和 Linux

註： 您無法編輯 `WAR` 檔中的原始檔。

如果要在編輯器中檢視/修改原始碼，請選擇下列其中一個選項：

- 在發現項目表格中，按兩下發現項目。這時會開啟內部編輯器，顯示該行程式碼。
- 在發現項目表格中的發現項目上按一下滑鼠右鍵，並選取**在內部編輯器中開啟**或**在外部編輯器中開啟** > **<編輯器>**（其中 **<編輯器>** 是已列在上述表格中的其中一個支援的外部編輯器）。
- 選取一個追蹤節點，然後選取**在內部編輯器中開啟**或**在外部編輯器中開啟** > **<編輯器>** 工具列按鈕；或者用滑鼠右鍵按一下選項，並且從功能表中選取**在內部編輯器中開啟**或**在外部編輯器中開啟** > **<編輯器>**。

如果您已在編輯器中開啟檔案，標記會指出檔案中代表發現項目的位置。如果要遵循這些位置回到發現項目表格，請在編輯器中，用滑鼠右鍵按一下該行程式碼，然後從功能表中選取**在「發現項目」視圖中顯示**。

支援的註釋和屬性

在掃描期間，會處理一些用來裝飾程式碼的註釋或屬性。於掃描期間，當程式碼中發現支援的註釋或屬性時，會利用這項資訊，將裝飾的方法標示為污染的回呼。針對標示為污染回呼的方法，會將其所有引數都視為包含受污染的資料。這會產生更多含有追蹤資料的發現項目。這個說明主題中會列出所支援的註釋和屬性。

- 『支援的 Java 註譯』
- 『支援的 AppScan Source Java 註譯』
- 第 159 頁的『支援的 Microsoft .NET 屬性』

支援的 Java 註譯

表 *14.* 支援的 *Java* 註譯

註釋	縮寫
javax.xml.ws.WebServiceProvider	@WebServiceProvider
javax.jws.WebService	@WebService
javax.jws.WebMethod	@WebMethod

支援的 AppScan Source Java 註譯

- 『使用 AppScan Source 註釋』
- 第 159 頁的『@ValidatorMethod』
- 第 159 頁的『@SuppressSecurityTrace』
- 第 159 頁的『@CallbackMethod』

使用 AppScan Source 掃描 Java 時，支援 `@ValidatorMethod`、`@CallbackMethod` 和 `@SuppressSecurityTrace` 方法層次註釋。

使用 AppScan Source 註釋

您可以透過遵循下列步驟來使用註釋：

1. 依預設會啟用註釋支援。註釋 `.jar` 檔為 `<install_dir>\lib\SecurityAnnotations.jar`（其中 `<install_dir>` 是 AppScan Source 安裝的位置）。
2. 如果您要掃描經過前置編譯的類別 `.war` 檔或 `.jar` 檔，請找出包含加註的程式碼的 Java 專案。

3. 將 SecurityAnnotations.jar 新增至專案的類別路徑。
4. 重建專案。

您可以在掃描之前新增註釋至原始碼；或者可以在掃描之後及分類期間新增它們，以識別及刪除誤判。

系統提供註釋，讓您可以將知識以安全註釋的形式直接插入原始碼中。由於可以使用註釋來宣告程式碼安全的部分，因此應非常小心使用它們。它們不應用於應掃描是否有安全漏洞的程式碼。如果您使用註釋，則安全分析師可以選擇透過停用 <data_dir>\config\scanner.ozsettings（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述） 中的特性來忽略它們。在這個檔案中，請尋找這項設定：

```
<Setting
name="process_security_annotations"
value="true"
default_value="true"
description="When turned on, security annotations in the
     source code will be processed by AppScan Source."
display_name="Process Security Annotations"
type="bool"
/>
```

若要停用此功能，請將 value="true" 變更為 value="false"。

@ValidatorMethod

驗證器是在輸入資料上執行檢查的方法，經常會傳回布林值，以指出輸入是否有效。您可以將使用者輸入變更為可接受的格式，而不是使用驗證器來接受或拒絕輸入。這些方法稱為消毒器。

使用 @ValidatorMethod 註釋，您可以識別應用程式原始碼中的所有驗證器和消毒器方法。在 AppScan Source 掃描期間，將會使用這項資訊來移除通過這些方法的資料流程，原因是現在已將該資料視為安全。

註：目前，並沒有規定要指定註解方法的哪些參數應被視為經過驗證。在 AppScan Source 掃描期間，會假設所有輸入參數都已經過驗證。

@SuppressSecurityTrace

會移除流過以此註釋標示之方法的所有追蹤資料。當某一組追蹤資料被識別為誤判，或是比其他追蹤資料較不重要或較不引起關注時，這會非常有用。您可以使用這個註釋來過濾掉這些追蹤資料，或是隱藏它們以便減少雜亂。

@CallbackMethod

這個註釋用來識別應用程式的回呼點或進入點。所有的引數都被視為帶有污染。

支援的 Microsoft .NET 屬性

表 *15.* 支援的 *Microsoft .NET* 屬性

屬性	縮寫
System.Web.Services.WebServiceAttribute	WebService
System.Web.Services.WebMethodAttribute	WebMethod

第 6 章 AppScan Source 追蹤

當使用 AppScan Source 追蹤時，您可以驗證輸入驗證和編碼是否符合您的軟體安全原則。您可以查看產生輸入/輸出追蹤資料的發現項目，將方法標示為驗證常式和編碼常式、來源或接收槽、回呼，或污染傳播者。

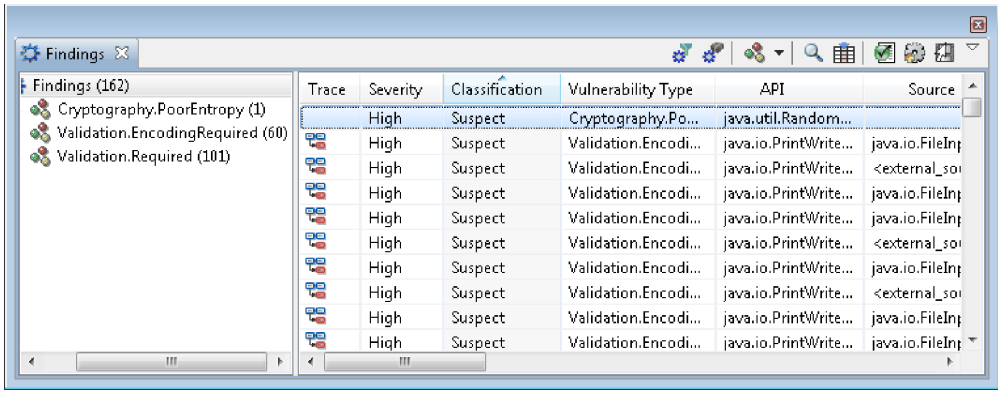
AppScan Source 會在應用程式中，跨越多個模組和語言來追蹤資料流程。它在呼叫圖中顯示可能有危險的資料路徑，指出應用程式容易出現漏洞的區域。

追蹤可協助您在應用程式中，找出欠缺經過認可的輸入驗證常式及編碼常式之處，以打敗 SQL 注入攻擊、跨網站 Scripting 攻擊，以及其他輸入驗證攻擊。您以互動方式來追蹤整個呼叫圖，在「追蹤」視圖中直接按一下，在您選擇的開發環境或程式碼編輯器中查看來源。追蹤也能夠強制執行原則，可讓您識別適當的輸入驗證及編碼所需要的已核准常式、污染的傳播，或接收槽和來源，將它們併入未來的掃描中。

當掃描產生追蹤時，您可以從「追蹤」視圖中，建立特定發現項目的輸入驗證常式或編碼常式、漏洞、接收槽、來源，或污染傳播者。比方說，如果您在 AppScan Source for Analysis 中將某個常式標示為驗證常式並將它新增到 AppScan Source 安全知識庫，後續的掃描不會再報告呼叫常式之資料路徑的 `Validation.Required` 或 `Validation.Encoding.Required` 發現項目。在「追蹤」視圖中，您也可以將漏洞定義為來源和/或接收槽，將某個方法識別為污染傳播者、污染的回呼，或不容易遭到污染。

AppScan Source 追蹤掃描結果

掃描結果可能包括 AppScan Source 追蹤所識別的追蹤資料。**追蹤**直欄中的圖示表示有呼叫圖追蹤存在。



掃描可能產生 `Validation.Required` 和 `Validation.EncodingRequired` 類型的發現項目。這些發現項目指出在原始碼之中，從外部來源讀取資料，或將資料儲存到外部接收槽的位置。對於這些情況，掃描會設定旗標，因為資料應該要驗證或編碼，以防止惡意或錯誤的資料造成傷害。

驗證和編碼

驗證是檢查輸入資料以確保它是形式完整的程序。Validation.Required 發現項目指出從來源到接收槽的給定資料路徑未進行驗證。驗證可能如同將資料限制在長度上限一樣簡單，也可能如同檢查名稱和位址的形式是否完整一樣複雜。驗證也可以檢查「SQL 注入」之類的攻擊，它會偵測出能夠啟用這些攻擊的無效字元序列。

編碼是將資料轉換成形式完整狀態的程序。Validation.EncodingRequired 發現項目指出從來源到接收槽的給定資料路徑未進行編碼。編碼可能如同字元跳出一樣簡單，也可能如同資料加密一樣複雜。編碼也可以跳出會導致「跨網站 Scripting」之類攻擊的字元，以防止這些攻擊。

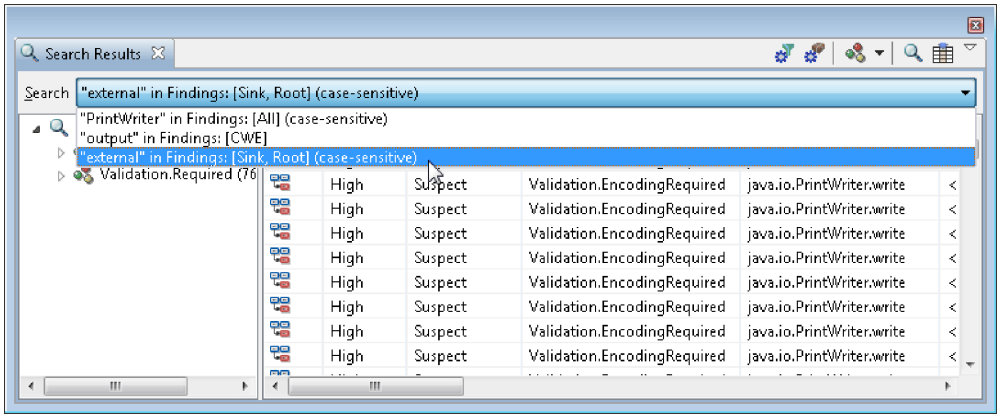
當第一次掃描時，AppScan Source可能會將某個發現項目識別為可疑安全發現項目。當建立適用於特定來源的驗證或編碼常式時，如果從來源接收資料之後，未呼叫指定的驗證或編碼常式，AppScan Source for Analysis 會將發現項目報告為明確（而不是可疑）。

評量會追蹤整個專案中之已知來源的資料。如果能夠將資料從已知來源追蹤到已知的接收槽，指定的驗證和編碼常式就可以確保不會出現以無界限的輸入資料來進行的惡意攻擊。

搜尋 AppScan Source 追蹤

如果您想要分組追蹤發現項目，您可以搜尋來源或接收槽。如此一來，追蹤發現項目會出現在「搜尋結果」視圖中。

在「追蹤」視圖中，按一下**搜尋含有相同類型常式的追蹤資料**。然後在「搜尋發現項目」對話框中，選取來源、接收槽、遺失的接收槽（包括虛擬遺失的接收槽）、虛擬遺失的接收槽或追蹤呼叫，將包含字串的追蹤結果隔離出來。累加的搜尋結果會出現在「搜尋結果」視圖中。從這個視圖中，您可以重新搜尋來精簡搜尋。



輸入/輸出追蹤

當 AppScan Source for Analysis 追蹤從已知來源到接收槽或遺失的接收槽的資料時，就會產生輸入/輸出追蹤。

輸入/輸出追蹤

如果程式碼分析能夠追蹤污染的來源到某個接收槽或遺失的接收槽，分析會產生一項輸入/輸出追蹤。追蹤的根是一個方法，它從污染產生來源取得資料，再將資料傳給一系列呼叫，並最終寫入不受保護的接收槽。

來源和接收槽

- **來源**：來源是對程式的輸入，例如：檔案、Servlet 要求、主控台輸入或 Socket。對於大部分的輸入來源而言，傳回的資料內容和長度都是沒有限制的。如果沒有對輸入進行檢查，則會將其視為受到污染。來源會列在任何發現項目表格的**來源**直欄中。
- **接收槽**：接收槽可以是資料能夠寫出的任何外部格式。資料庫、檔案、主控台輸出和 Socket 都是接收槽的範例。未經檢查，便將資料寫入接收槽，可能是一個嚴重的安全漏洞。
- **遺失的接收槽**：遺失的接收槽是指無法再追蹤的 API 方法。

  註：遺失的接收槽不適用於 JavaScript 發現項目。

使用「追蹤」視圖

關於這項作業

在「追蹤」視圖中，您可以檢視發現項目的單一輸入/輸出追蹤。這個窗格分成三個畫面：

- 輸入及輸出堆疊
- 資料流
- 圖形呼叫圖

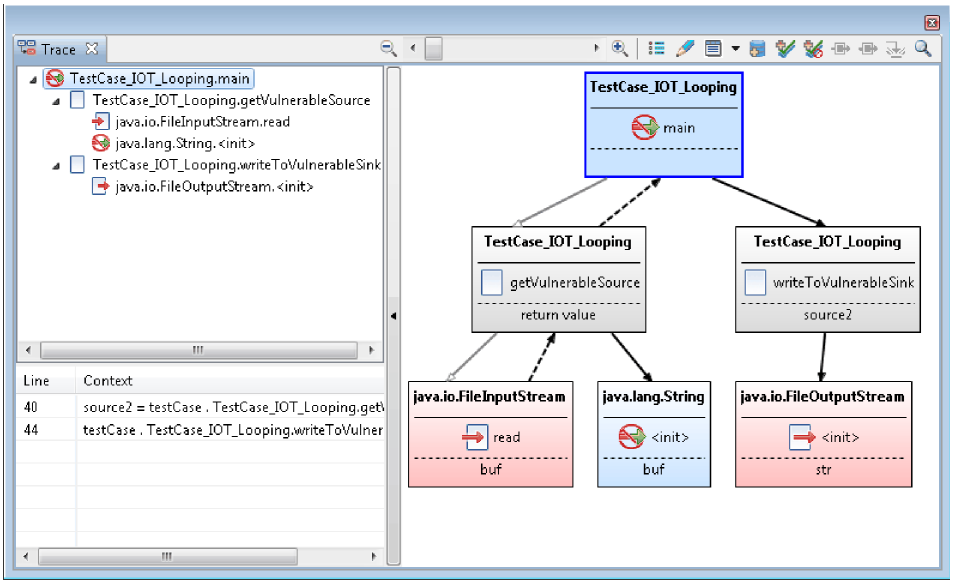
在 第 164 頁的『「追蹤」視圖中的輸入/輸出堆疊』中，有這些畫面更詳細說明。

註：在 JavaScript 追蹤中，會顯示第 166 頁的『JavaScript 陳述式圖形』，而不是「圖形呼叫圖」。

如果要檢視 AppScan Source 追蹤，請執行下列動作：

程序

1. 在「發現項目」視圖中，掃描及尋找追蹤結果。
2. 從「檢視」功能表中，開啟「追蹤」視圖。
3. 在發現項目表格中，選取顯示**追蹤**圖示的列。「追蹤」視圖會顯示追蹤詳細資料。



「追蹤」視圖中的輸入/輸出堆疊

左上畫面顯示**輸入**和**輸出**堆疊。堆疊是一個終止於來源（輸入堆疊）或接收槽（輸出堆疊）的呼叫序列。

資料流

左下畫面包含所選方法的資料流。資料可以在方法呼叫或指派中流動。「資料流」區段顯示原始碼中出現項目和環境定義的行號。

呼叫圖

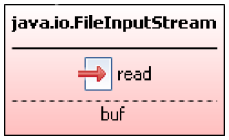
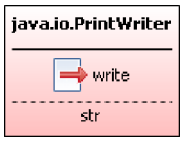
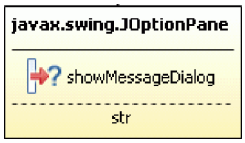
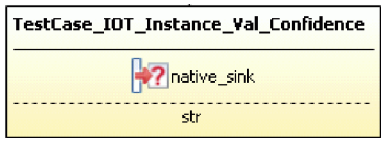
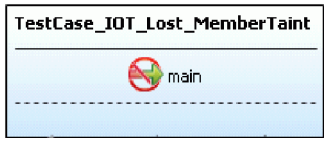
註：在 JavaScript 追蹤中，會顯示第 166 頁的『JavaScript 陳述式圖形』，而不是「圖形呼叫圖」。

圖表是呼叫圖的圖形表示法。在圖形內，每個方法呼叫都是一個矩形，顯示類別名稱和方法名稱：

- 紅色表示方法呼叫是來源、接收槽，或兩者兼俱。
- 遺失的接收槽是指無法再追蹤的 API 方法。虛擬遺失的接收槽是一個同時為虛擬函數的遺失的接收槽（一個可以有多個實作的函數）。黃色表示方法呼叫是遺失的接收槽或虛擬遺失的接收槽。
- 藍色表示方法呼叫不是虛擬/編碼常式。
- 灰色代表全部其他追蹤節點類型。

每一個方法呼叫有三個區段：類別名稱、方法名稱和汙染的引數名稱。方法呼叫的浮動說明會提供更詳細的資料。

有箭頭的線條代表方法之間的呼叫。空心箭頭表示呼叫中沒有已知污染的資料，實心箭頭代表污染的資料流。虛線箭頭表示 return 陳述式。

符號	說明
	不含已知污染資料的方法呼叫
	含有污染資料的方法呼叫
	傳回中含有污染資料
	來源（紅色）：可能為不可信資料起源的方法、函數或參數。
	接收槽（紅色）：可能容易遭到污染的資料攻擊，或是使用上可能有危險的方法或函數。
	遺失的接收槽（黃色）：對污染的資料有潛在漏洞，或可能使用上是有危險的方法/函數。
	虛擬遺失的接收槽（黃色）：是一種遺失的接收槽類型，解析為多個具體實作。
	不是驗證/編碼常式（藍色）。將 API 標示為不是驗證/編碼常式，指出這個 API 不驗證任何資料。
	污染傳播者：傳播污染至一或多個其參數、回覆值、或此指標的函數/方法。

提示：

- 在「追蹤」視圖中，將滑鼠游標移到圖形中的追蹤節點上，會提供節點的相關資訊。
- 可收合視圖中的兩個左畫面（輸入/輸出堆疊畫面和資料流畫面），使圖形呼叫曲線的檢視更容易。如果要收合這些畫面，請選取**隱藏樹狀結構視圖**方向鈕。如果要顯示這些隱藏的畫面，請選取**顯示樹狀結構視圖**方向鈕。
- 請移動捲軸，來放大及聚焦在細部，或縮小以看得更多。將滑鼠指標移至縮放捲軸可提供現行縮放比例。如果要放大到最大層次，請按一下**縮放至 200%**。如果要縮小到越遠越好，請按一下**適當縮放**。

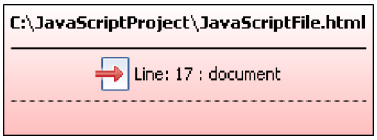
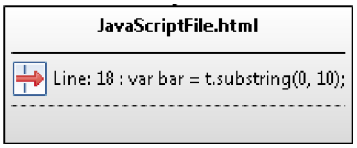
JavaScript 陳述式圖形

JavaScript 追蹤的陳述式圖形區段會顯示陳述式之間的資料流。

在圖形內，每一個陳述式都是矩形，它提供下列資訊：

- 受影響檔案的路徑和檔名。如果下一個陳述式位於相同檔案中，只會列出檔名。
- 包含該陳述式的行號。
- 感興趣的程式碼區段（若有的話）。
- 如果矩形是紅色，則陳述式是來源及/或接收槽。
- 如果矩形是灰色，則陳述式是一個污染傳播者。
- 陳述式的浮動說明會提供更詳細的資料。

有箭頭的線條代表陳述式之間的資料流。

符號	說明
	污染資料的流程
	來源（紅色）：可能為不可信資料起源的陳述式。
	接收槽（紅色）：可能容易遭到污染的資料攻擊，或是使用上可能有危險的陳述式。
	污染傳播者：將污染傳播到它的一或多個參數、它的回覆值或這個指標的陳述式。

提示：

- 在「追蹤」視圖中，將滑鼠游標移到圖形中的追蹤節點上，會提供節點的相關資訊。
- 可收合視圖中的兩個左畫面（輸入/輸出堆疊畫面和資料流畫面），使圖形呼叫曲線的檢視更容易。如果要收合這些畫面，請選取**隱藏樹狀結構視圖**方向鈕。如果要顯示這些隱藏的畫面，請選取**顯示樹狀結構視圖**方向鈕。
- 請移動捲軸，來放大及聚焦在細部，或縮小以看得更多。將滑鼠指標移至縮放捲軸可提供現行縮放比例。如果要放大到最大層次，請按一下**縮放至 200%**。如果要縮小到越遠越好，請按一下**適當縮放**。

在編輯器中分析原始碼

AppScan Source 可讓您在內部編輯器中分析或修改原始碼，或者，您也可以從各種外部編輯器中選擇。

外部編輯器能讓您檢閱 AppScan Source for Analysis 中的結果，並且在您選擇的開發環境中進行程式碼修改。外部編輯器包含：

表 *16.* 支援的外部編輯器

編輯器	平台
Eclipse（請參閱AppScan Source系統需求，以瞭解支援哪些版本的 Eclipse）	Windows 和 Linux
記事本	Windows
vi	Linux
系統預設值	Windows 和 Linux

註： 您無法編輯 WAR 檔中的原始檔。

如果要在編輯器中檢視/修改原始碼，請選擇下列其中一個選項：

- 在發現項目表格中，按兩下發現項目。這時會開啟內部編輯器，顯示該行程式碼。
- 在發現項目表格中的發現項目上按一下滑鼠右鍵，並選取**在內部編輯器中開啟**或**在外部編輯器中開啟** > **<編輯器>**（其中 **<編輯器>** 是已列在上述表格中的其中一個支援的外部編輯器）。
- 選取一個追蹤節點，然後選取**在內部編輯器中開啟**或**在外部編輯器中開啟** > **<編輯器>** 工具列按鈕；或者用滑鼠右鍵按一下選項，並且從功能表中選取**在內部編輯器中開啟**或**在外部編輯器中開啟** > **<編輯器>**。

如果您已在編輯器中開啟檔案，標記會指出檔案中代表發現項目的位置。如果要遵循這些位置回到發現項目表格，請在編輯器中，用滑鼠右鍵按一下該行程式碼，然後從功能表中選取**在「發現項目」視圖中顯示**。

驗證和編碼範圍

在「追蹤」視圖中，您可以指定自訂驗證常式和編碼常式，將它們儲存在 AppScan Source 安全知識庫之後，將資料標示為已檢查而非已污染。當使用「自訂規則精靈」時，您會根據這些常式的範圍來定義它們。

請參閱第 177 頁的『範例 4：深度驗證』，以取得建立驗證和編碼常式的程序。

驗證或編碼常式以它們的範圍為基礎，定義如下：

- 『API 特定』
- 『呼叫位置特定』

API 特定

API 特定驗證常式和編碼常式可以與單一專案或多個專案相關聯。

API 特定常式會淨化來自特定來源 API 之所有實例的任何資料。例如，您可以指定來自下列 API 之任何輸入的驗證常式：

```
javax.servlet.ServletRequest.getParameter
(java.lang.string):java.lang.string
```

API 特定常式儲存在伺服器上。專案的 API 特定常式儲存在專案中。

呼叫位置特定

呼叫位置特定常式一律與單一專案相關聯。

呼叫位置特定常式會淨化來自程式碼特定位置的資料。當建立呼叫位置特定驗證常式或編碼常式時，您會指定這個常式套用到特定的輸入呼叫位置。呼叫位置特定常式一律儲存在專案中。

註：「呼叫位置特定」會套用到驗證常式在相同方法內的任何呼叫。

從 AppScan Source 追蹤建立自訂規則

您可以從「追蹤」視圖來建立自訂規則，讓您濾除包含污染傳播者、不容易遭受污染或接收槽等追蹤的發現項目。您也可以將追蹤中的方法標記為驗證/編碼常式（或指出它們不是驗證/編碼常式）。

關於這項作業

請參閱第 172 頁的『範例 2：從「追蹤」視圖建立驗證/編碼常式』，取得建立驗證和編碼常式的原始碼、輸出和程序的範例。

表 *17.* 「追蹤」視圖節點的有效標示

選取的方法	有效標示
中介節點	• 驗證/編碼常式 • 不容易遭受污染 • 不是驗證/編碼常式
遺失的接收槽	• 污染傳播者 • 不容易遭受污染 • 接收槽

程序

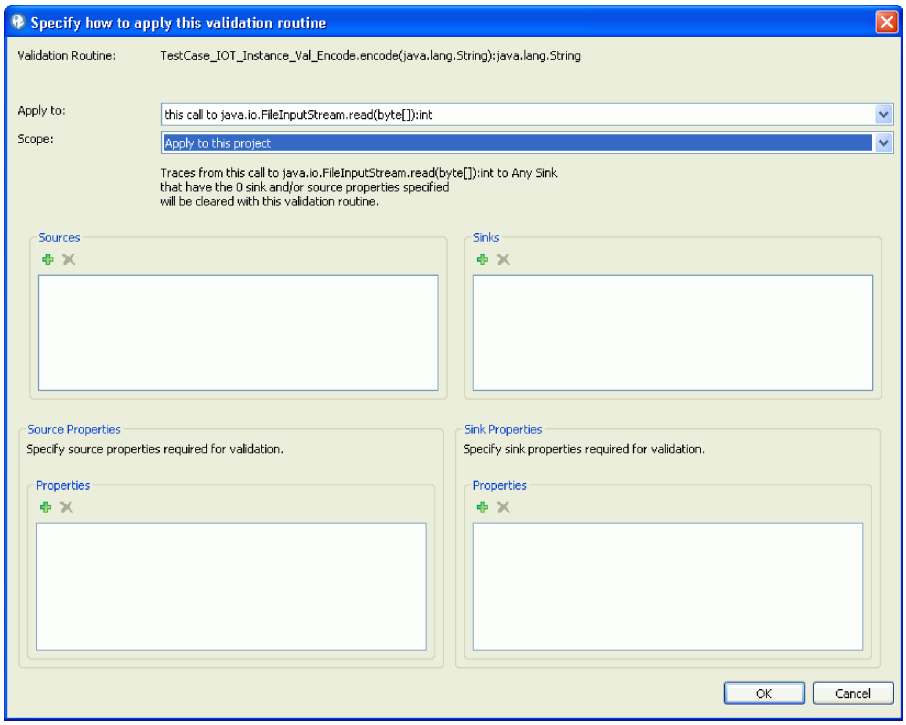
1. 在「追蹤」視圖中，用滑鼠右鍵按一下您要建立自訂規則的方法或節點，然後選擇要建立的自訂規則；或選取方法或節點，然後按一下適當的自訂規則工具列按鈕。標示常式和方法的選項如下：

選項	敘述
標示為驗證/編碼常式	
標示為不是驗證/編碼常式	
標示為污染傳播者	
標示為不容易遭受污染	
標示為接收槽	

註：如果在您要建立自訂規則之方法的「追蹤」視圖中沒有項目，請按一下**啟動自訂規則精靈來新增不在追蹤圖形中的驗證常式**。在「自訂規則精靈」中，請繼續至「選取驗證/編碼常式」頁面。選取驗證常式，然後根據下一步中的指示來指定位

置、範圍、任何來源或接收槽，或任何內容。請參閱第 175 頁的『範例 2：從「自訂規則精靈」建立驗證/編碼常式』，以取得使用此精靈建立驗證常式的詳細資料。

2. 如果您建立的自訂規則會將方法標示為接收槽或驗證/編碼常式，您可能需要進一步設定：
   a. 如果將方法標示為接收槽，請指定接收槽屬性：
      - **漏洞類型**
      - **嚴重性**
   b. 對於驗證常式，請指定位置和範圍；以及驗證常式要套用的任何來源或接收槽，或其內容。



   - **套用於**：
     - **這個 <method name> 呼叫**（呼叫位置特定）：只會套用於這個呼叫的輸入。
     - **任何 <method name> 呼叫**（API 特定）：會套用於這個方法之任何呼叫的驗證/編碼常式。
     - **不考量<方法名稱>，以下指定的所有限制項**：容許規則影響所有的來源。
   - **範圍**：
     - **套用於這個專案**：如果選取，規則會儲存在專案 (.ppf) 檔中。
     - **套用於所有專案**：以這個設定所建立的驗證規則會儲存在資料庫中。
   - **來源**：選取驗證常式要套用的輸入來源。如果要新增來源，請按一下**新增**，然後從「選擇簽章」對話框中選取來源。如果要新增多個來源，您可以在「選擇簽章」對話框中複選來源。

- **接收槽**：選取驗證常式要套用的接收槽。如果要新增接收槽，請按一下**新增**，然後從「選擇簽章」對話框中選取接收槽。如果要新增多個接收槽，您可以在「選擇簽章」對話框中複選接收槽。
- **來源內容**：如果您要規則清除在來源中以特定內容為開頭的追蹤，請按一下**新增 VMAT 內容**，然後從「選擇內容」對話框中選取內容。如果要新增多個內容，您可以在「選擇內容」對話框中複選內容。
- **接收槽內容**：如果您要規則濾除在接收槽中以特定內容為結尾的追蹤，請按一下**新增 VMAT 內容**，然後從「選擇內容」對話框中選取內容。如果要新增多個內容，您可以在「選擇內容」對話框中複選內容。

3. 在「追蹤」視圖中建立自訂規則之後，您必須重新掃描您的程式碼，以查看反映在發現項目清單和追蹤資料中的規則。您在「追蹤」視圖中建立的自訂規則，可以在「自訂規則」視圖中進行檢視和刪除。如果要在「自訂規則」視圖中檢視規則的詳細資料，請選取規則，然後按一下**自訂規則資訊**。

用來追蹤的程式碼範例

本節提供程式碼範例，其中說明從來源到接收槽的污染資料追蹤，以及如何建立驗證和編碼常式。

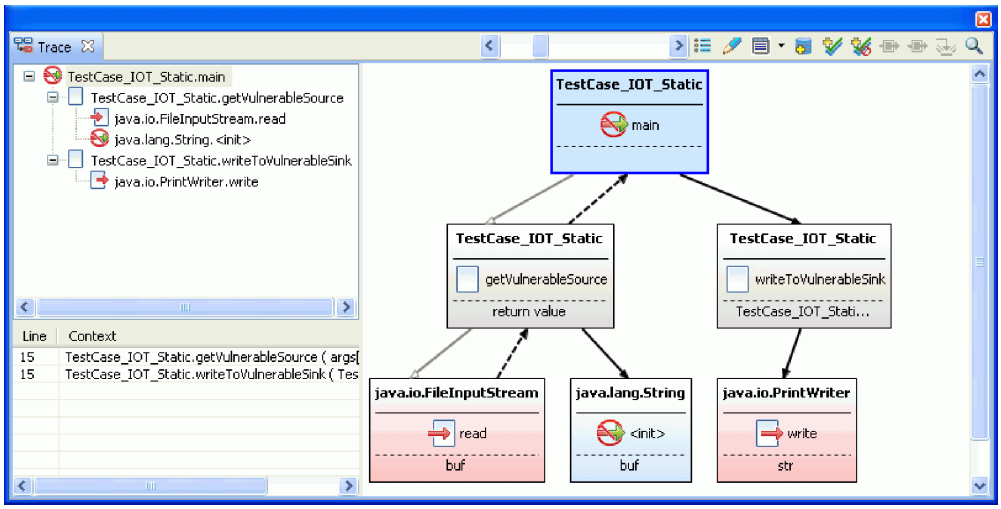
範例 1：從來源到接收槽

在下列程式碼範例中，main 方法呼叫會傳回字串的 getVulnerableSource 方法。請注意，雖然這個方法會從完全不明的檔案讀取資料，但從不檢查傳回之資料的有效性。之後，main 方法又將這個污染資料傳給 writeToVulnerableSink。writeToVulnerableSink 方法將資料寫到檔案中，從不檢查它的有效性。

```
import java.io.*;

public class TestCase_IOT_Static {
  public static void main(String[] args) {
    try {
      writeToVulnerableSink(getVulnerableSource(args[0]));
    } catch (Exception e) {
    }
  }

  public static String getVulnerableSource(String file)
    throws java.io.IOException, java.io.FileNotFoundException {
    FileInputStream fis = new FileInputStream(file);
    byte[] buf = new byte[100];
    fis.read(buf);
    String ret = new String(buf);
    fis.close();
    return ret;
  }

  public static void writeToVulnerableSink(String str)
```

```
    throws java.io.FileNotFoundException {
    FileOutputStream fos = new FileOutputStream(str);
    PrintWriter writer = new PrintWriter(fos);
    writer.write(str);
  }
}
```

程式碼範例會產生下列追蹤：



這個窗格顯示輸入堆疊（其中 main 呼叫 getVulnerableSource，後者又呼叫 FileInputStream.read），以及輸出堆疊（其中 main 呼叫 writeToVulnerableSink，後者又呼叫 PrintWriter.write）。圖形顯示資料如何從 read 方法流到 write 方法，並以 main 結合這兩個呼叫堆疊。「資料流」區段顯示 main 方法中，傳遞污染的作業中的行號。在這個範例中，這兩個方法呼叫位於方法內的同一行（第 15 行）（在上述的範例程式碼，這會轉換為第 7 行，在畫面擷取中，檔案含有 8 行註解）。

範例 2：從來源到接收槽已進行修改

範例是範例程式碼的修改。它強化了範例 1，因為它新增了一個稱為getVulnerableSource-的驗證常式，以及一個稱為 writeToVulnerableSink 的編碼常式。

```
import java.io.*;

public class TestCase_IOT_Instance_Val_Encode {
  public static void main(String[] args) {
    try {
      TestCase_IOT_Instance_Val_Encode testCase = new
        TestCase_IOT_Instance_Val_Encode();
      String file = args[0];
      String source = testCase.getVulnerableSource(file);
      source = testCase.validate(source);
      String encodedStr = testCase.encode(source);
      testCase.writeToVulnerableSink(file, encodedStr);
    } catch (Exception e) {
    }
  }

  public String getVulnerableSource(String file) throws Exception {
    FileInputStream fis = new FileInputStream(file);
    byte[] buf = new byte[100];
    fis.read(buf);
    fis.close();

    String ret = new String(buf);
```

```
    return ret;
  }

  public void writeToVulnerableSink(String file, String str)
      throws FileNotFoundException {
    FileOutputStream fos = new FileOutputStream(file);
    PrintWriter writer = new PrintWriter(fos);
    writer.write(str);
  }

  private String validate(String source) throws Exception {
    if (source.length() > 100) {
      throw new Exception("Length too long: " + source.length());
    }
  return source;
  }

  private String encode(String source) {
    return source.trim();
  }
}
```

第一次掃描會產生一個堆疊追蹤，類似於範例 1 的堆疊追蹤。

延伸知識庫來併入驗證和編碼常式，會減少發現項目中的雜訊，且可以確認所有呼叫圖都呼叫驗證和編碼常式。比方說，如果您在上例的 java.io.FileInputStream.read(byte[]):int 任何呼叫中指定資料，任何從 read 發出的呼叫，只要也呼叫這個驗證常式，掃描就會將它排除。另外，從 read 發出的呼叫，只要未呼叫自訂驗證方法，就會升級至明確的安全發現項目，因為未呼叫程式碼中已知的驗證方法，有可能導致惡意的攻擊。

驗證常式也可能會驗證 FileInputStream 的 read 方法的其他變異。這可能會指定為其他來源。此外，您可能也知道這個方法只驗證某些接收槽（或含某些內容的接收槽）。例如，這個常式僅受限在含 Technology.IO 內容的接收槽，例如用來消耗此範例資料的 PrintWriter.write 接收槽。

範例 2：從「追蹤」視圖建立驗證/編碼常式

關於這項作業

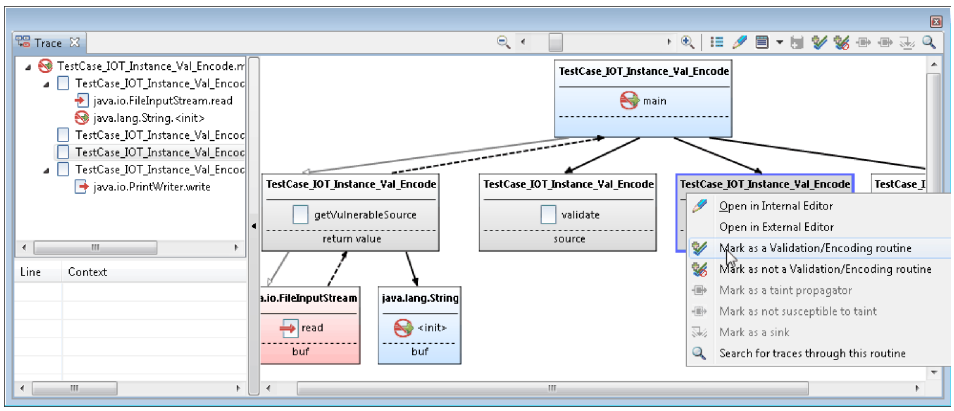
由於 AppScan Source 追蹤將 FileInputStream.read 方法視為產生污染資料的來源，您應該建立一個驗證常式或編碼常式，使未來的掃描不再有這個發現項目。

如果要建立 FileInputStream.read 的輸入驗證常式，請執行下列動作：

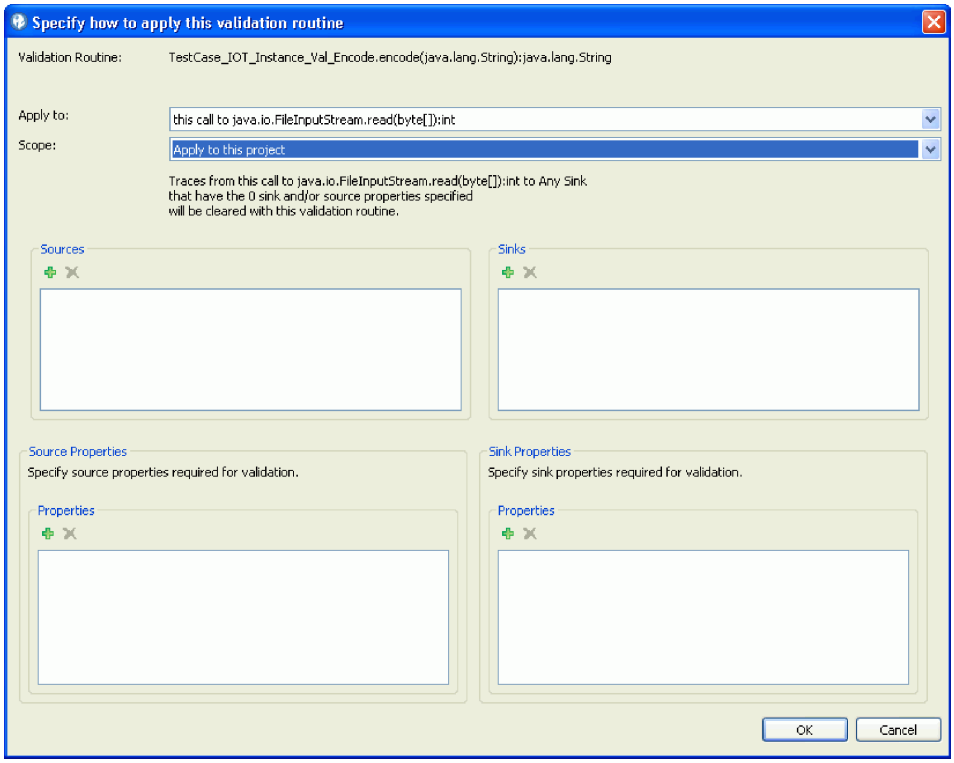
程序

1. 在「追蹤」視圖的呼叫圖中，選取且用滑鼠右鍵按一下 TestCase_IOT_Instance_Val_Encode.encode 方法。

   提示：如果您要建立的驗證/編碼常式未出現在追蹤圖形中，您可以從「追蹤」視圖啟動「自訂規則精靈」來建立常式。第 175 頁的『範例 2：從「自訂規則精靈」建立驗證/編碼常式』說明執行這項作業所包含的步驟。
2. 在功能表中，選取**標示為驗證/編碼常式**。



3. 如果 encode 常式只套用於這個呼叫 FileInputStream.read 的特定實例，請在「指定如何套用這個驗證常式」對話框中，選取**這個 java.io.FileInputStream.read 呼叫**。

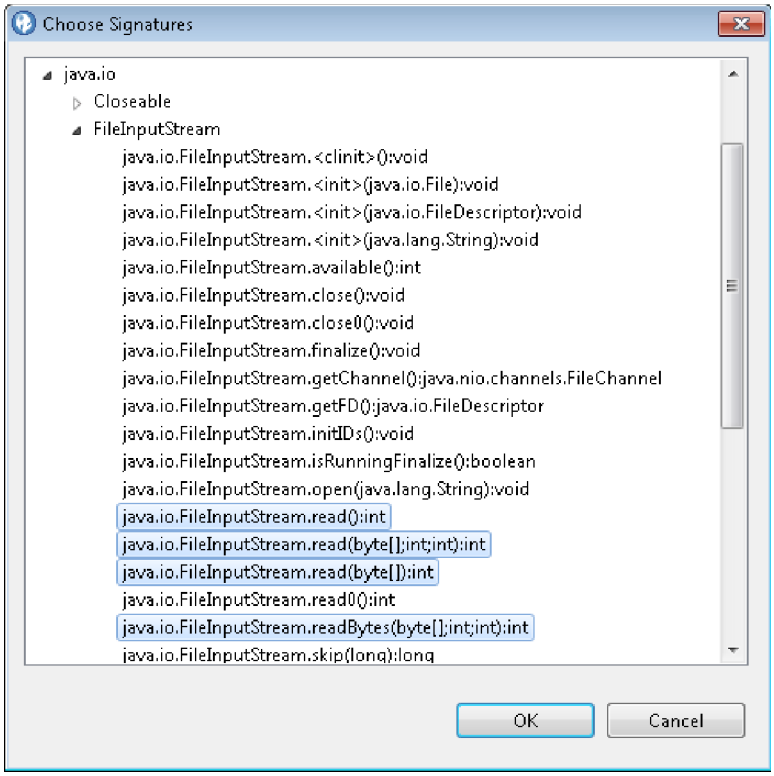


因為 validate 方法專用於這個類別且緊密關聯於程式碼，您通常會指定**這個 java.io.FileInputStream.read 呼叫**。

選取**任何 java.io.FileInputStream.read 呼叫**，將驗證常式套用於任何 read 方法呼叫。當選取這個選項時，如果只對現行專案有效，也要選取**套用至這個專案**，否則，請選取**套用至所有專案**。

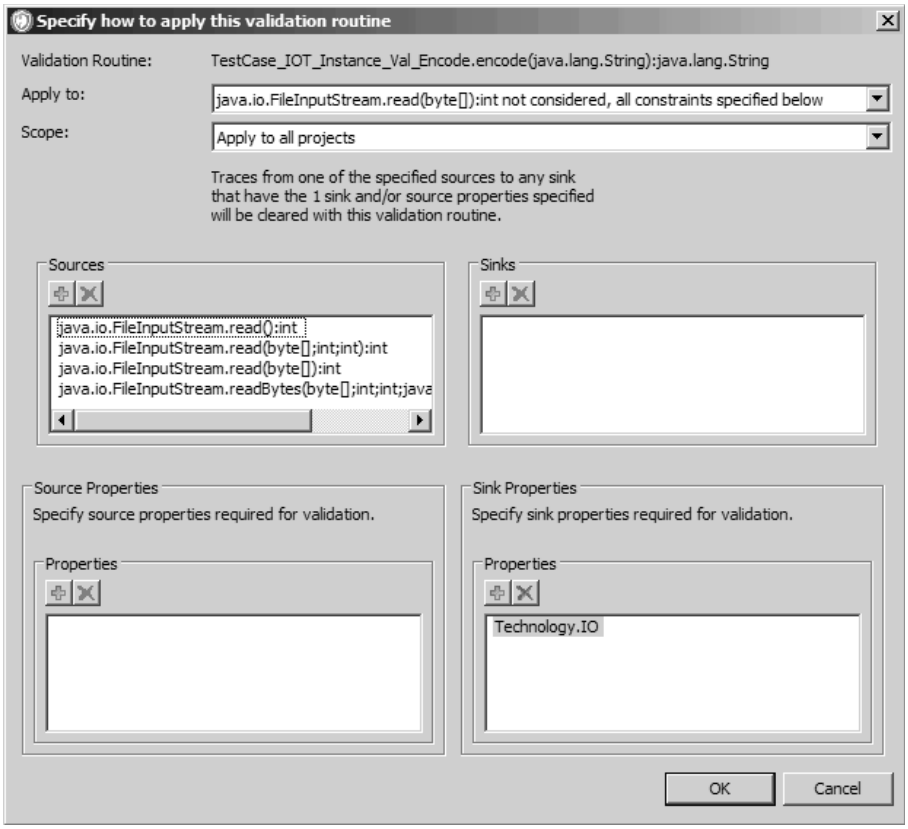
4. 設定常式以套用到 FileInputStream 類別的所有 read 方法以及具有 Technology.IO 內容的任何接收槽（如 java.io.PrintWrite.write 方法）：
   a. **新增 read 方法為來源**：雖然您可以指定**任何 java.io.FileInputStream.read (byte[]):int 呼叫**，將 java.io.FileInputStream.read(byte[]):int 新增為來

源，但我們將個別新增來源。在「指定如何套用這個驗證常式」對話框中，選取**套用於**功能表中的**不考量 java.io.FileInputStream.read(byte[]):int，以下指定的所有限制項**。然後按一下**來源**區段**新增**按鈕。在「選擇簽章」對話框中，展開 `java.io`，然後是 `FileInputStream` 區段。複選 `java.io.FileInputStream.read*` 節點，然後按一下**確定**。



b. **新增接收槽內容**：按一下**接收槽內容**區段的**新增 VMAT 內容**按鈕。在「選擇內容」對話框中，選取 `Technology.IO` 內容，然後按一下**確定**。

c. 完成所有的設定時，對話框類似下列顯示內容：



5. 按一下**確定**來新增驗證常式到資料庫中。

範例 2：從「自訂規則精靈」建立驗證/編碼常式

如果您要建立的驗證/編碼常式未出現在追蹤圖形中，您可以從「追蹤」視圖啟動「自訂規則精靈」來建立常式。

關於這項作業

本範例將建立第 172 頁的『範例 2：從「追蹤」視圖建立驗證/編碼常式』中所建立的相同驗證常式；不過，在本範例中，將使用「自訂規則精靈」來建立常式。

程序

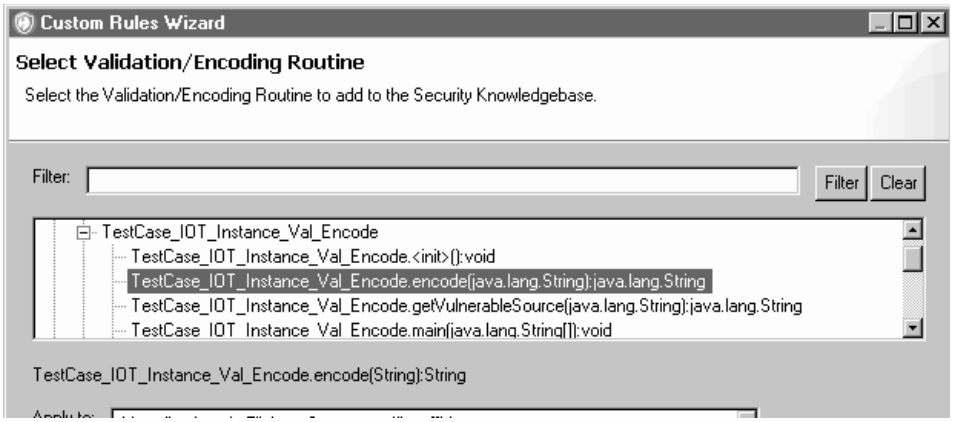
1. 在「追蹤」視圖中，按一下工具列上的**啟動自訂規則精靈來新增不在追蹤圖形中的驗證常式**。

   註：如果是從「自訂規則」視圖中啟「自訂規則精靈」，則無法從該精靈來建立驗證常式。

2. 在精靈的「選取驗證/編碼常式」頁面中，指定驗證常式的位置。

   關於這個範例，請選取下列常式：

   ```
   TestCase_IOT_Instance_Val_Encode.encode(java.lang.String):
   java.lang.String
   ```



3. 使用在第 172 頁的『範例 2：從「追蹤」視圖建立驗證/編碼常式』中的**指定如何套用這個驗證常式**對話框的相同設定來完成其餘的精靈頁面區段。
4. 按一下**完成**來新增驗證常式到資料庫。

範例 3：不同的來源和接收槽檔案

下列範例說明與接收槽不同檔案的來源。

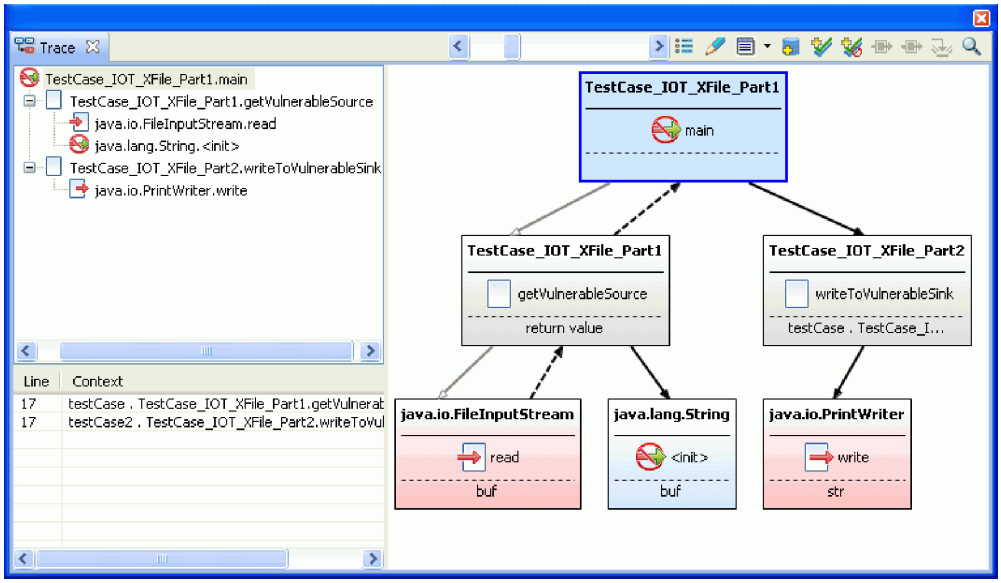
TestCase_IOT_Xfile_Part1.java：

```
public class TestCase_IOT_XFile_Part1 {
 public static void main(String[] args) {
  try {
      TestCase_IOT_XFile_Part1 testCase =
        new TestCase_IOT_XFile_Part1();
      TestCase_IOT_XFile_Part2 testCase2 =
        new TestCase_IOT_XFile_Part2();
   testCase2.writeToVulnerableSink(
        testCase.getVulnerableSource(args[0]));
  } catch (Exception e) {
  }
 }

  public String getVulnerableSource(String file)
    throws IOException, FileNotFoundException {
  FileInputStream fis = new FileInputStream(file);
  byte[] buf = new byte[100];
  fis.read(buf);
  String ret = new String(buf);
  fis.close();
  return ret;
 }
}
```

TestCase_IOT_Xfile_Part2.java：

```
public class TestCase_IOT_XFile_Part2 {
  public void writeToVulnerableSink(String str)
  throws FileNotFoundException {
  FileOutputStream fos = new FileOutputStream(str);
  PrintWriter writer = new PrintWriter(fos);
  writer.write(str);
 }
}
```

將資料從 TestCase_IOT_Xfile_Part1.java 追蹤到 TestCase_IOT_Xfile_Part2.java，可以在整個程式中，全程追蹤資料流向。堆疊追蹤呈現如下：



這個範例顯示在 main 方法中，從 TestCase_IOT_XFile_Part1 流到 TestCase_IOT_XFile_Part2 的資料流向。

範例 4：深度驗證

當掃描範例 4 程式碼時，第一個掃描會包括三項 AppScan Source 追蹤，其根目錄位在對應的追蹤常式。假設選取 trace1 中的 FileInputStream.read 方法，新增了 validate 常式。跟在範例原始碼後的一節說明驗證常式每個範圍的效果。

```
public class TestCase_IOT_UserValidation {
     ResultSet resultSet;
     FileInputStream fileInputStream;
     PrintWriter printWriter;
     byte[] buffer;

     public static void main(String[] args) throws Exception {
        TestCase_IOT_UserValidation testCase = new TestCase_IOT_UserValidation();
          testCase.trace1();

        TestCase_IOT_UserValidation testCase2 = new TestCase_IOT_UserValidation();
        testCase2.trace2();

        TestCase_IOT_UserValidation testCase3 = new TestCase_IOT_UserValidation();
        testCase3.trace3();
    }

     private void trace1() throws Exception {
          String source = getVulnerableSource1();
        source = validate(source);
          writeToVulnerableSink(source);
    }

     private void trace2() throws Exception {
          String source = getVulnerableSource2();
        source = validate(source);
          writeToVulnerableSink(source);
    }

     private void trace3() throws Exception {
          String source = getVulnerableSource3();
```

```
        source = validate(source);
          writeToVulnerableSink(source);
    }

     public String getVulnerableSource1() throws Exception {
          fileInputStream.read(buffer);
          return new String(buffer);
    }

     public String getVulnerableSource2() throws Exception {
          fileInputStream.read(buffer);
          return new String(buffer);
    }

     public String getVulnerableSource3() throws Exception {
          return resultSet.getString("x");
    }

     public void writeToVulnerableSink(String str) throws Exception {
          printWriter.write(str);
    }

    private String validate(String source) throws Exception {
          // 驗證
        return source;
    }
}
```

呼叫位置特定驗證常式 - 這個 *FileInputStream.read* 呼叫的輸入

當驗證只適用於範圍非常狹小的環境定義，或輸入方法太普通，以致於無法提供一個驗證常式時，請建立呼叫位置特定驗證常式。當您在 trace1 方法中**套用於這個 FileInputStream.read 呼叫**時，在下一次掃描之後，trace1 不會呈現為發現項目，因為它的呼叫堆疊包含 validate 方法呼叫。不過，仍會報告 trace2，即使它呼叫了 validate 也一樣，因為驗證常式的範圍是關聯於 trace1 呼叫位置。 trace3 方法也會呼叫 validate，但仍會繼續報告它，因為它以 ResultSet.getString 為來源。

API 特定驗證常式 - 任何 *FileInputStream.read* 呼叫的輸入

當驗證只套用於特定來源時，請建立 API 特定驗證常式。當您**套用於任何 FileInputStream.read 方法呼叫**時，下一次掃描，trace1 和 trace2 方法不會出現發現項目，因為它們含有 validate 方法呼叫。不過，trace3 方法仍存在，即使它呼叫 validate 也一樣，因為它以 ResultSet.getString 為來源。

第 7 章 AppScan Source for Analysis 和問題追蹤

AppScan Source for Analysis 整合了問題追蹤系統，可以將確認的軟體漏洞直接遞送到開發人員的桌面。提交到問題追蹤系統的問題報告包含錯誤的文字說明，還有一個檔案，其中只含有隨著問題報告而提交的發現項目。

您可以利用整合各種問題追蹤系統（包括 IBM Rational ClearQuest、IBM Rational Team Concert、HP Quality Center 和 Microsoft Team Foundation Server）的 AppScan Source for Analysis，追蹤您的軟體漏洞問題報告。

將發現項目提交給問題追蹤系統，或將問題報告郵寄給開發人員之前，您可能需要先配置問題追蹤系統喜好設定（請參閱第 84 頁的『利用喜好設定來啟用問題追蹤』）。

利用喜好設定來啟用問題追蹤

「問題追蹤系統」喜好設定可讓您提交發現項目給問題追蹤系統，並且決定如何提交問題報告。

「問題追蹤系統」喜好設定頁面中的「一般」標籤可供您在 AppScan Source 中，啟用或停用「問題追蹤系統」整合特性。如果已選取**啟用問題追蹤系統整合**勾選框，評量發現項目就會有**提交問題報告**快速功能表動作可用。另外，「一般」標籤也提供了離散控制，控制在提交問題報告時將提供哪個「問題追蹤系統」。

如果要瞭解支援的問題追蹤系統所能設定的喜好設定，請參閱下列說明主題：

- 第 85 頁的『Rational ClearQuest 喜好設定』
- 第 85 頁的『Quality Center 喜好設定』
- 第 87 頁的『Rational Team Concert 喜好設定』
- 第 88 頁的『Team Foundation Server 喜好設定』

Rational ClearQuest 喜好設定

如果要完成 Rational ClearQuest 喜好設定，Rational ClearQuest 管理者必須向您提供必要的 Rational ClearQuest 設定。這些是您的 Rational ClearQuest 環境專用的設定。

註：當整合 Rational ClearQuest 8.0 版時，Rational ClearQuest 綱目必須包含 **DefectTracking** 預定綱目中的可用欄位。

資料庫集

一或多個問題報告資料庫的集合。

`Linux 預設值 = 連線名稱，Windows 預設值 = 資料庫集`

資料庫名稱

提交問題報告的目標資料庫名稱。

資料庫使用者名稱

預設 Rational ClearQuest 資料庫使用者名稱。

CQPerl 執行檔的位置

Rational ClearQuest CQPerl 執行檔在本端電腦的位置。提供的預設位置對映於預設的 Rational ClearQuest 安裝位置。

問題報告記錄的實體

Rational ClearQuest 安裝所配置用於問題報告物件的實體（資料庫物件）。

預設實體是**問題報告**。

記錄的說明欄位

預設說明是**說明**。

記錄的標題欄位

預設標題是**標題**。

每個發現項目單一問題報告

請將發現項目群組提交成單一問題報告或多份問題報告。當建立問題報告時，您可以變更提交方法。

Quality Center 喜好設定

您必須先在「一般問題追蹤系統」喜好設定中啟用 HP Quality Center，然後在 Quality Center 標籤中，設定個別喜好設定。

伺服器 URL

Quality Center Server URL - 例如，`http://<主機名稱>:<埠>/qcbin/` 或 `https://<主機名稱>:<埠>/qcbin/`。

使用者名稱（選用）

登入 Quality Center 的使用者名稱

密碼（選用）

如果您輸入使用者名稱，請為其輸入密碼。

網域

要連接的「Quality Center 網域」。

專案

要連接的「Quality Center 專案」

自動登入

如果是 true，當提交發現項目時，AppScan Source 不會提示您輸入登入資訊，而會以「喜好設定」所指定的預設認證來登入。如果是 false，您每次向「品質中心」提交發現項目時，都必須登入。

自動提交

如果是 true，當提交發現項目時，不會出現用來提交新問題報告的對話框。AppScan Source for Analysis 會使用「喜好設定」所指定的**預設問題報告內容**。如果是 false，當提交發現項目時，會在提示中要求您輸入問題報告資訊（嚴重性、優先順序、問題報告類型、狀態，等等）。

重新提交先前提交的發現項目

提交給 Quality Center 的發現項目會標示一些 Quality Center 問題報告資訊（問題報告 ID、提交使用者及提交日期）。依預設，AppScan Source 不會重新提交相同的發現項目超過一次。如此您便可以將多個發現項目分派給 Quality Center，只需要在 Quality Center 資料庫中輸入新的發現項目。如果選取的話 (true)，可以將先前提交的發現項目重新提交給 Quality Center。

將每個發現項目當作個別錯誤來提交

當在單一作業中提交多個發現項目時，您可以用一份 Quality Center 問題報告來提交所有發現項目，也可以針對每個 AppScan Source 發現項目各提交一份 Quality Center 問題報告。選取這個勾選框，會將這個旗標設成 true，為每一個別發現項目都分別建立一份 Quality Center 問題報告。將這個旗標設為 false，會為所有在大量提交中提交的發現項目建立一份 Quality Center 問題報告。

自動產生錯誤摘要

如果是 true，AppScan Source 會自動產生 Quality Center 的問題報告提交摘要。摘要會指出問題報告所包含的發現項目數，以及發現項目的類型，例如 `Validation.Required`。

如果是 false，當提交問題報告時，在建立新問題報告而開啟的對話框中，會顯示有待填寫的「摘要」欄位。

自動載入錯誤欄位

預設值是 true。當選取這個勾選框時，AppScan Source 會根據 Quality Center 的現行使用者和群組設定，自動載入 Quality Center 資料庫中的問題報告欄位定義。如果是 false，在建立新問題報告而開啟的對話框中，AppScan Source 不會顯示 Quality Center 的問題報告欄位。

預設問題報告內容

如果要設定不同 Quality Center 問題報告屬性的預設值，請在 Quality Center 喜好設定標籤中，按一下**預設問題報告內容**。預設值有可能在提交之時，預先移入**新建問題報告**對話框，如果選取**自動提交**喜好設定，就會以無聲自動的方式傳送到 Quality Center。

註：如果選取**自動載入錯誤欄位**，每次出現**問題報告內容**對話框時，都會從 Quality Center 動態取出問題報告內容及可用的值。因此，任何新增到 Quality Center 資料庫的新欄位和值，都會自動出現在 AppScan Source for Analysis 中。必須提供有效的伺服器、登入和連線資訊之後，才能開啟**問題報告內容**對話框，將 Quality Center 資訊移入其中。

自訂 Quality Center 問題報告欄位

您可以在「新建問題報告」對話框中，利用配置檔來自訂欄位，以及這些欄位之間的互動。您可以在 `<data_dir>\config\qc.dts`（其中 `<data_dir>` 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述） 中找到範例配置檔，其中含有自訂範例及其他文件。這些自訂可讓您直接在「新建問題報告」對話框中，建立「Quality Center 工作流程」的 Script 邏輯模型。

可用的自訂包括：

- 顯示自訂的欄位、遺漏的欄位，或兩者
- 強制一律顯示欄位（置換 Quality Center 設定）
- 根據其他欄位的選項來更新所需要的欄位狀態
- 根據另一個欄位中的清單框選項，動態更新欄位的清單框選項

Rational Team Concert 喜好設定

Rational Team Concert 喜好設定標籤可讓您配置 Rational Team Concert 伺服器連線，以及配置工作項目的屬性值。

輸入連線資訊且順利登入之後，您可以選擇連接到一或多個專案區域。每個專案區域都可以配置它自己的屬性預設值。

註：當您連接到 Rational Team Concert（藉由配置喜好設定或提交問題報告），系統可能會提示您接受 SSL 憑證。如需相關資訊，請參閱 第 88 頁的『Rational Team Concert SSL 憑證』。

如果要配置給定專案區域的屬性值，請選取專案區域，然後選擇**配置**。在配置對話框中，您可以將屬性值設為寫入程式的值，在某些情況下，也可設為參照所選發現項目的變數。例如，在屬性值中使用 `{Finding.fileName}`，在提交期間，會取代為發現項目實際的原始碼檔名。對支援這些變數的屬性值提供「內容輔助」(`<Alt>+</>`)。團隊最好利用 Rational Team Concert 喜好設定主頁面中的**匯入**和**匯出**按鈕來共用這些配置。

Team Foundation Server 喜好設定

Team Foundation Server 喜好設定標籤可讓您配置連接到 Microsoft Team Foundation Server 的連線，以及配置工作項目欄位的值。

輸入連線資訊且順利登入之後，您可以選擇連接到一或多個專案。

註：在配置 Team Foundation Server 2010 的登入時，「伺服器 URL」必須包含要連接的「團隊專案集合」。例如：`http://myserver:8080/tfs/`*`DefaultCollection`*。

每個專案都可以配置它自己的欄位預設值。

如果要配置給定專案的欄位值，請選取專案，然後選擇**配置**。在配置對話框中，您可以將欄位值設為寫入程式的值，在某些情況下，也可設為參照所選發現項目的變數。

例如，在欄位值中使用 {Finding.fileName}，在提交期間，會取代為發現項目實際的原始碼檔名。會對支援這些變數的欄位提供「內容輔助」(<Alt>+</>)。

團隊最好利用 Team Foundation Server 喜好設定主頁面中的**匯入**和**匯出**按鈕，來共用這些配置。

整合 HP Quality Center 與 AppScan Source for Analysis

如果要整合 HP Quality Center 與 AppScan Source for Analysis，本端電腦需要安裝 Quality Center 用戶端。您第一次透過 Quality Center 瀏覽器型用戶端介面登入 Quality Center 時，Quality Center 用戶端應用程式會下載並安裝到本端電腦上。

配置 Quality Center 資訊

您可以在 Quality Center 標籤的「問題追蹤系統」喜好設定中，配置 Quality Center。在將 AppScan Source 發現項目提交成問題報告之前，您必須先啟用 Quality Center 並設定 Quality Center 喜好設定。請參閱第 85 頁的『Quality Center 喜好設定』，以取得每個喜好設定設定的說明。

註：在某些環境（例如，執行 HP Quality Center 11 版的環境）中，您可能需要安裝 HP ALM Client MSI Generator 附加程式，HP Quality Center 整合才能運作。

向 Quality Center 提交發現項目

您可以透過任何 AppScan Source for Analysis 的「發現項目」視圖，向 Quality Center 提交發現項目。

程序

1. 在表格中選取一或多個發現項目，或開啟一個組合。（如果您開啟某個組合，請選取要提交的組合發現項目。）
2. 用滑鼠右鍵按一下選項，從功能表中，選取**提交問題報告** > **分派至 Quality Center**。
3. 登入 Quality Center。

   如果喜好設定配置自動登入，就不會出現登入對話框。 AppScan Source 會以預設認證來登入。
4. 提交發現項目。

   如果喜好設定配置為自動提交，AppScan Source 會利用**預設問題報告內容**喜好設定來提交發現項目資訊。

結果

提交發現項目之後，會出現一則參考訊息，指出已順利提交的發現項目數。

追蹤提交給 Quality Center 的發現項目

提交給 Quality Center 的發現項目會標示一些提交資訊：

- Quality Center 問題報告 ID
- 提交日期

- Quality Center 使用者名稱

提交資訊出現在任何發現項目視圖之發現項目表格的**問題報告 ID**、**問題報告日期**和**問題報告使用者**直欄中。不過，預設的「發現項目表格」不包含這些直欄。您必須按一下**選取直欄及進行排序**工具列按鈕來自訂表格，將表格配置成包含這些直欄。請參閱第 266 頁的『自訂發現項目表格』，以取得新增直欄至發現項目視圖的詳細資料。

問題報告資訊會跨越掃描而持續保存，供您在分類和補救期間，追蹤 AppScan Source 發現項目的狀態。

Quality Center 中的 AppScan Source 發現項目資訊

當 AppScan Source for Analysis 在 Quality Center 資料庫中建立問題報告時，發現項目資訊會設為問題報告**說明**。這個發現項目資訊包括「嚴重性」、「類型」、API 和「分類」。

Quality Center 問題報告也能夠包含附加於問題報告的 AppScan Source 組合檔 (`.ozbdl`)。這個組合檔包含 AppScan Source 發現項目的所有相關資訊，追蹤也包括在內。之後，開發人員可以在 AppScan Source for Analysis 或開發人員外掛程式中儲存及開啟組合，以及分類問題報告。

整合 Rational ClearQuest 和 AppScan Source for Analysis

如果要整合 Rational ClearQuest 與 AppScan Source for Analysis，本端電腦需要安裝 Rational ClearQuest 用戶端。這項安裝包括 `CQPerl` 執行檔，您必須在 AppScan Source for Analysis Rational ClearQuest 喜好設定中配置執行檔的位置。

您在配置 Rational ClearQuest 整合喜好設定時，會指定問題報告資料庫綱目的相關資訊。 Rational ClearQuest 實體是指 Rational ClearQuest 資料庫物件，您必須指定 Rational ClearQuest 安裝架構用於問題報告的實體。

註：AppScan Source for Analysis 整合所需要的 CQPerl 執行檔的預設位置，即為預設 Rational ClearQuest 安裝目錄。

註：當整合 Rational ClearQuest 8.0 版時，Rational ClearQuest 綱目必須包含 **DefectTracking** 預定綱目中的可用欄位。

向 Rational ClearQuest 提交發現項目

發現項目與公司的問題追蹤系統整合起來，可供開發人員進行補救。您可以將個別發現項目傳送到您的問題追蹤系統，或者，您也可以提交含有一或多個發現項目的組合。在 AppScan Source 階段作業期間，第一次從 AppScan Source 向 Rational ClearQuest 提交發現項目時，您必須用您的使用者名稱和密碼登入。

當您向 Rational ClearQuest 提交組合時，錯誤號碼會與組合中特定的發現項目相關聯，而不是關聯於組合本身。如此可以確保在建立問題報告時，您既能進一步操作這個組合，又能保留與問題報告相關的特定發現項目。

組合可以包含許多發現項目。您可以選擇在一份問題報告中，提交所有發現項目，或每個發現項目各提交一份問題報告。如果您選取**每個發現項目單一問題報告**喜好設定，且存在多個發現項目，您可以編輯這些問題報告的「說明」。您只能編輯單一提交問題報告的「說明」。

註：登入 Rational ClearQuest 之前，您必須先設定「預設追蹤系統」喜好設定。

註：當整合 Rational ClearQuest 8.0 版時，Rational ClearQuest 網目必須包含 **DefectTracking** 預定網目中的可用欄位。

向 Rational ClearQuest 提交問題報告

程序

1. 在表格中選取一或多個發現項目，或開啟一個組合。（如果您開啟某個組合，請選取要提交的組合發現項目。）
2. 用滑鼠右鍵按一下選項，從功能表中，選取**提交問題報告** > **分派至 ClearQuest**。
3. 登入 Rational ClearQuest，提交發現項目。

結果

每份問題報告都會附加一個只含有相關檔案的評量檔。 AppScan Source for Analysis 或 AppScan Source for Development 可以開啟這個評量檔。

註：當整合 Rational ClearQuest 8.0 版時，Rational ClearQuest 網目必須包含 **DefectTracking** 預定網目中的可用欄位。

整合 Rational Team Concert 和 AppScan Source for Analysis

如果要整合 Rational Team Concert 與 AppScan Source for Analysis，電腦不需要另外安裝 Rational Team Concert 用戶端。

如果要配置 Rational Team Concert 連線，請移至「問題追蹤系統」喜好設定的 Rational Team Concert 標籤，或者您也可以提交一個問題報告，讓系統提示您登入及配置連線。

Rational Team Concert 喜好設定也可讓您配置預設欄位值，以便在提交問題報告期間使用。如此一來，您就可以設定每份問題報告所要使用的值，也可以修改 AppScan Source 所提供的預設值。

註：當您連接到 Rational Team Concert（藉由配置喜好設定或提交問題報告），系統可能會提示您接受 SSL 憑證。如需相關資訊，請參閱 第 88 頁的『Rational Team Concert SSL 憑證』。

向 Rational Team Concert 提交問題報告

您可以向 Rational Team Concert 提交含有一或多個發現項目的組合，您也可以提交個別發現項目。您第一次從 AppScan Source for Analysis 中向 Rational Team Concert 提交發現項目時，必須用您的使用者名稱和密碼登入。如果您要配置提交期間所用的預設欄位值，您可以在 Rational Team Concert 喜好設定中配置它們。

關於這項作業

當您向 Rational Team Concert 提交組合時，工作項目號碼會與組合中的特定發現項目相關聯，而不是關聯於組合本身。如此可以確保您既能進一步操作這個組合，又能保留特定發現項目與工作項目號碼的關聯。

程序

1. 在表格中選取一或多個發現項目，或開啟組合。（如果您開啟組合，請選取要提交的組合發現項目。）
2. 用滑鼠右鍵按一下選項，從功能表中，選取**提交問題報告** > **分派至 Rational Team Concert**。
3. 之後，提交對話框會帶您逐步執行這個程序，其中包括登入，必要的話，還得填寫必要的屬性。

   註：當您連接到 Rational Team Concert（藉由配置喜好設定或提交問題報告），系統可能會提示您接受 SSL 憑證。如需相關資訊，請參閱 第 88 頁的『Rational Team Concert SSL 憑證』。

結果

組合會自動新增到提交的工作項目中，稍後 AppScan Source for Analysis 或 AppScan Source for Development 使用者可以開啟這個組合。

Rational Team Concert SSL 憑證

當安裝 Rational Team Concert 伺服器時，應該配置它來使用有效的 SSL 憑證。如果沒有執行這個步驟，當登入伺服器時，會收到未授信連線訊息（配置喜好設定或提交問題報告之時）。這個主題概述 Rational Team Concert SSL 憑證考量。

SSL 憑證儲存體位置

已永久接受的憑證儲存在 `<user_home>/.jazzcerts`（其中 `<user_home>` 是作業系統起始目錄（例如，在 Windows 上，此目錄可能是 `C:\Documents and Settings\Administrator\`）） 中。移除 `<user_home>/.jazzcerts` 會刪除 AppScan Source 和 Rational Team Concert 用戶端所有已儲存的憑證。

與 Rational Team Concert 用戶端共用的 SSL 憑證

AppScan Source 會與 Rational Team Concert 用戶端共用它的 SSL 憑證儲存庫。如果您利用 Rational Team Concert 用戶端來永久接受某個憑證，AppScan Source 會重複使用它（在 AppScan Source 中，系統不會提示您接受憑證）。同樣地，如果您在 AppScan Source 中永久接受某個憑證，Rational Team Concert 用戶端也會重複使用這個憑證。

整合 Microsoft Team Foundation Server 和 AppScan Source for Analysis

如果要整合 Team Foundation Server 和 AppScan Source for Analysis，本端電腦需要安裝 Microsoft Visual Studio Team Explorer 用戶端。

如果要配置連線來通往 Team Foundation Server，請移至「問題追蹤系統」喜好設定的 Team Foundation Server 標籤，或者您也可以提交一個問題報告，讓系統提示您登入及配置連線。

Team Foundation Server 喜好設定也可讓您配置預設欄位值，以便在提交問題報告期間使用。如此一來，您就可以設定每份問題報告所要使用的值，也可以修改 AppScan Source 所提供的預設值。

向 Microsoft Team Foundation Server 提交問題報告

您可以向 Team Foundation Server 提交含有一或多個發現項目的組合，您也可以提交個別發現項目。您第一次從 AppScan Source for Analysis 中向 Team Foundation Server 提交發現項目時，必須用您的使用者名稱和密碼登入。如果您要配置提交期間所用的預設欄位值，您可以在 Team Foundation Server 喜好設定中配置它們。

關於這項作業

當您向 Team Foundation Server 提交組合時，工作項目號碼會與組合中的特定發現項目相關聯，而不是關聯於組合本身。如此可以確保您既能進一步操作這個組合，又能保留特定發現項目與工作項目號碼的關聯。

註：在配置 Team Foundation Server 2010 的登入時，「伺服器 URL」必須包含要連接的「團隊專案集合」。例如：`http://myserver:8080/tfs/DefaultCollection`。

程序

1. 在表格中選取一或多個發現項目，或開啟組合。（如果您開啟組合，請選取要提交的組合發現項目。）
2. 用滑鼠右鍵按一下選項，然後從功能表中，選取**提交問題報告 > 分派至 Team Foundation Server**。
3. 之後，提交對話框會帶您逐步執行這個程序，其中包括登入，必要的話，還得填寫必要欄位。

結果

組合會自動新增到提交的工作項目中，稍後 AppScan Source for Analysis 或 AppScan Source for Development 使用者可以開啟這個組合。

處理已提交的問題報告

如果您以個別問題報告來提交較多發現項目，當您的分類程序在繼續進行時，這個程序會在背景中執行。在提交問題報告之後，從問題報告系統收到的問題報告 ID 會附加到相關的發現項目，且會保持與這個發現項目同在。如果要處理已提交給問題追蹤系統的問題報告，請遵循這個主題中的步驟。

程序

1. 開啟您的問題追蹤系統，找出問題報告。
2. 將附件儲存為 AppScan Source 組合檔 (`.ozbdl`)。您可以在 AppScan Source for Analysis 中開啟這個檔案，或在 AppScan Source for Development 中將它當作一項組合來開啟。

提交組合進行問題追蹤，並利用電子郵件提交

組合中的發現項目可以提交給公司的問題追蹤系統，或利用電子郵件傳送。只要將發現項目放在組合中，您就可以將這些發現項目當作錯誤來提交開發人員進行補救。

程序

1. 開啟組合。
2. 按一下**提交組合進行問題追蹤**工具列按鈕下移鍵，然後選取您的問題追蹤系統。

   註：視您的問題追蹤系統而定，在提交組合之前，您可能希望修改「問題追蹤系統」喜好設定。

   或者，在「組合」工具列上，按一下**用電子郵件傳送組合**，將組合傳送給其他人（電子郵件喜好設定必須事先配置）。
3. 完成開啟的配置對話框。這些會因您選擇的問題追蹤系統而不同；請參閱說明中的 *AppScan Source for Analysis* 與問題追蹤區段。

透過電子郵件追蹤問題報告（利用電子郵件傳送發現項目）

關於這項作業

如果您已配置電子郵件喜好設定，您可以利用電子郵件，將發現項目或組合直接傳送給開發人員，向他們提出忠告，指出在掃描之後，發現潛在的問題報告。電子郵件包括含有發現項目的附件，以及說明發現項目的文字。

註：部分「簡易郵件傳送通訊協定 (SMTP)」中繼只會將郵件遞送到特定網域。在這種情況下，如果您從 `mydomain.com` 傳送，只有 `mydomain.com` 中的收件者能夠透過 AppScan Source for Analysis 接收電子郵件。

如果要以電子郵件傳送發現項目表格中的發現項目，請執行下列動作：

程序

1. 在表格中選取一或多個發現項目，或開啟一個組合。如果您開啟某個組合，請選取要用電子郵件傳送的組合發現項目。
2. 用滑鼠右鍵按一下選項，然後從功能表中，選擇**以電子郵件傳送發現項目**。
3. 電子郵件將包含含有發現項目的組合附件。在「附件檔名」對話框中，指定發現項目組合的名稱。例如，在**附件檔名**欄位中指定 `my_finding`，會將檔名為 `my_finding.ozbdl` 的組合附加到電子郵件中。 按一下**確定**來開啟「以電子郵件傳送發現項目」對話框。
4. 依預設，「以電子郵件傳送發現項目」對話框中的**郵件收件者**欄位，會移入電子郵件喜好設定所指定的**收件者位址**，不過，準備電子郵件時，很容易改變它。請在這個對話框中，檢閱電子郵件的內容，然後按一下**確定**來傳送電子郵件。

結果

範例電子郵件內容：

```
1 findings:
Name: JavaAny.test_DataInput
Type: Vulnerability.Validation.Required
Severity: Low
```

```
Classification: Suspect
File Name: C:\TestApps\java\JavaAny\src\JavaAny.java
Line / Col: 275 / 0
Context: di . java.io.DataInput.readFully ( ba )
Notes: Check into this vulnerability and report back ASAP.
```

提示：您可以從「發現項目詳細資料」視圖中，利用電子郵件來傳送個別發現項目或組合。您也可以按一下「組合」工具列上的**電子郵件組合**利用電子郵件來傳送組合。

第 8 章 發現項目報告和審核報告

安全分析師和風險管理員可以存取所選取發現項目的報告，或一系列審核報告，這些審核報告用來測量是否符合軟體安全最佳實務和規章需求。本節說明如何建立發現項目集成資料的報告。

AppScan Source for Analysis 會產生兩個報告類型：「發現項目報告」和「AppScan Source 報告」。發現項目報告是所選發現項目的報告。 *AppScan Source* 報告是以所有發現項目的種類分組為基礎，配合特定安全原則來進行調整的報告。如需 AppScan Source 報告清單，請參閱 第 193 頁的『AppScan Source 報告』。

各報告提供特定掃描期間所收集之發現項目的詳細資料，所有 AppScan Source 報告都可以包含新增到發現項目中的任何附註和追蹤資料。報告長度取決於報告所包含的發現項目數。您可以產生 PDF 檔報告，或產生「超文字標記語言 (HTML)」報告。HTML 報告會依照網頁的相同方式運作，按一下某個按鈕或鏈結，就可以跳到某一區段。之後，您可以利用 Web 瀏覽器中的瀏覽功能來導覽資訊。

報告也會列出已套用至發現項目的任何掃描時間過濾器。掃描時間過濾器會在第 141 頁的『判斷已套用的過濾器』中加以說明。

建立發現項目報告

關於這項作業

掃描之後，您可能會想要產生所識別之漏洞的相關報告。您可以產生多份發現項目報告：

- 發現項目
- 發現項目（依類型）
- 發現項目（依分類）
- 發現項目（依檔案）
- 發現項目（依 API）
- 發現項目（依組合）
- 發現項目（依 CWE）（一般弱點列舉）
- DTS 活動

註：發現項目報告會依種類來顯示詳細的發現項目，類似於發現項目表格中的結果。發現項目報告的產生可能需要大量記憶體（與 https://xmlgraphics.apache.org/fop/1.1/running.html#memory 相關），有可能需要高達 1024 MB 的額外系統記憶體。如果您在產生大型應用程式的掃描報告而注意到有記憶體問題，您可以個別掃描部分應用程式或變更掃描配置，然後重試產生報告。

發現項目報告中的 CWE ID 超鏈結會連接到 CWE 網站 (http://cwe.mitre.org/)。

如果要產生發現項目報告，請執行下列動作：

程序

1. 在含有發現項目的視圖中，選取要併入報告的發現項目。如果您沒有選取任何發現項目，報告就由作用中視圖的所有發現項目組成。

   在**工具**功能表中，按一下**產生發現項目報告**。或者，在包含發現項目的視圖中，選取一組發現項目並按一下滑鼠右鍵，然後在功能表中選取**產生發現項目報告**。

2. 在**選取發現項目報告**對話框中，選取一種報告類型。

   按一下**完成**，產生報告；或是按**下一步**，以便在「指定目的地和樣式表」頁面中，指定這些選用設定：

   - 您可以指定報告目的地和格式。您可以產生 HTML 格式的報告，使它成為含有所有 HTML 報告元件的 ZIP 檔，或產生 PDF（您必須有 Adobe Acrobat Reader，才能檢視 PDF 報告）。如果您沒有指定報告的目的地和格式（或在「選取發現項目報告」頁面中按一下**完成**），依預設，會選擇 HTML，並將報告儲存至 `<data_dir>\reports`（其中 `<data_dir>` 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）。

     註：如果您建立 PDF 格式的自訂報告（而不是發現項目報告），您可以指定報告中要包含的詳細程度：

     – **摘要**：包含每一個報告群組的計數
     – **詳細**：包含每一個 API 的每一個漏洞內容的計數
     – **綜合性**：包含由每一個 API 的每一個發現項目組成的表格
     – **標註**：包含所有發現項目，以及併入發現項目的任何附註、追蹤資料或程式碼片段

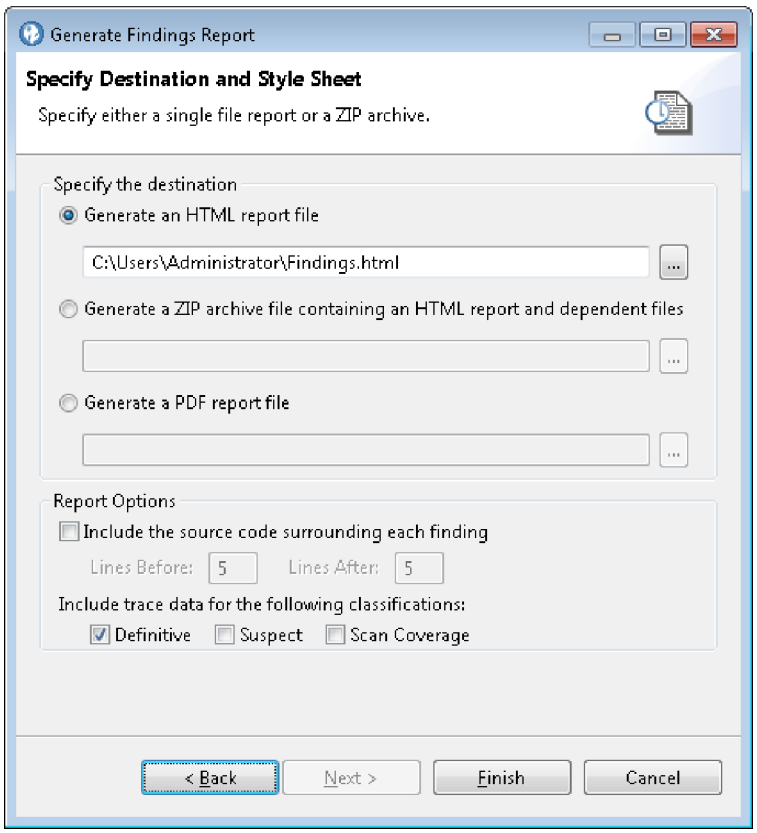
   - 如果要將程式碼片段併入報告中，請選取**併入每個發現項目周圍的原始碼**，指出在有漏洞的程式碼行前後，要併入報告的行數。

     提示：在「發現項目詳細資料」視圖的「報告」區段中，您也可以設定在報告中，要於發現項目前後併入的程式碼行數。

     產生報告之後，當您展開含有附註或程式碼片段的發現項目時，原始碼會出現在藍框內發現項目之下，或在黃色附註之下。粗體紅字強調有漏洞的程式碼行。

   - 如果要將 AppScan Source 追蹤資料併入報告中，請在**併入下列分類的追蹤資料**之下，選取一或多個分類（**明確**、**可疑**或**掃描涵蓋面**）。

   按一下**完成**來產生報告。



AppScan Source 報告

AppScan Source 報告可以協助軟體安全分析師、研發經理和風險管理審核員，測量是否符合軟體安全最佳實務和規章需求。 AppScan Source 報告有助於確保重要的應用程式會符合所設定的安全標準。

AppScan Source 利用原始碼漏洞分析結果，來推動一系列的報告，提供是否符合安全、開發或審核專業標準的詳細圖像。

AppScan Source 報告特性如下：

- 報告卡：報告卡代表每個主要種類的簡要安全狀態視圖
- 詳細審核檢閱：不符合標準的發現項目之詳細審核
- 往下探查：直接存取不符合標準的程式碼，以進一步分析，以及設定補救和指派的優先順序

AppScan Source for Analysis 會產生多種 AppScan Source 報告：

- 第 195 頁的『CWE/SANS Top 25 2011 報告』
- 第 195 頁的『DISA 應用程式安全及開發 STIG 3.10 版報告』
- 第 196 頁的『「開放式 Web 應用程式安全專案 (OWASP)」Mobile Top 10 報告』
- 第 195 頁的『「開放式 Web 應用程式安全專案 (OWASP)」Top 10 2013 報告』
- 第 196 頁的『付款卡產業資料安全標準 (PCI DSS) 3.2 版報告』

- 第 196 頁的『Software Security Profile 報告』：提供應用程式安全狀態的整體概觀，跨越每個主要的漏洞種類。

建立 AppScan Source 自訂報告

程序

1. 在**工具**功能表中，按一下**產生報告**。
2. 在「產生報告」對話框中，選取 AppScan Source 報告：
   - **CWE SANS Top 25 2011**
   - **DISA 應用程式安全及開發 STIG 3.10 版**
   - **OWASP Mobile Top 10**
   - **OWASP Top 10 2013**
   - **PCI 資料安全標準 3.2 版**
   - **Software Security Profile**

   按一下**完成**，產生報告；或是按**下一步**，以便在「指定目的地和樣式表」頁面中，指定這些選用設定：

   - 您可以指定報告目的地和格式。您可以產生 HTML 格式的報告，使它成為含有所有 HTML 報告元件的 ZIP 檔，或產生 PDF（您必須有 Adobe Acrobat Reader，才能檢視 PDF 報告）。如果您沒有指定報告的目的地和格式（或在「選取發現項目報告」頁面中按一下**完成**），依預設，會選擇 HTML，並將報告儲存至 <data_dir>\reports（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）。

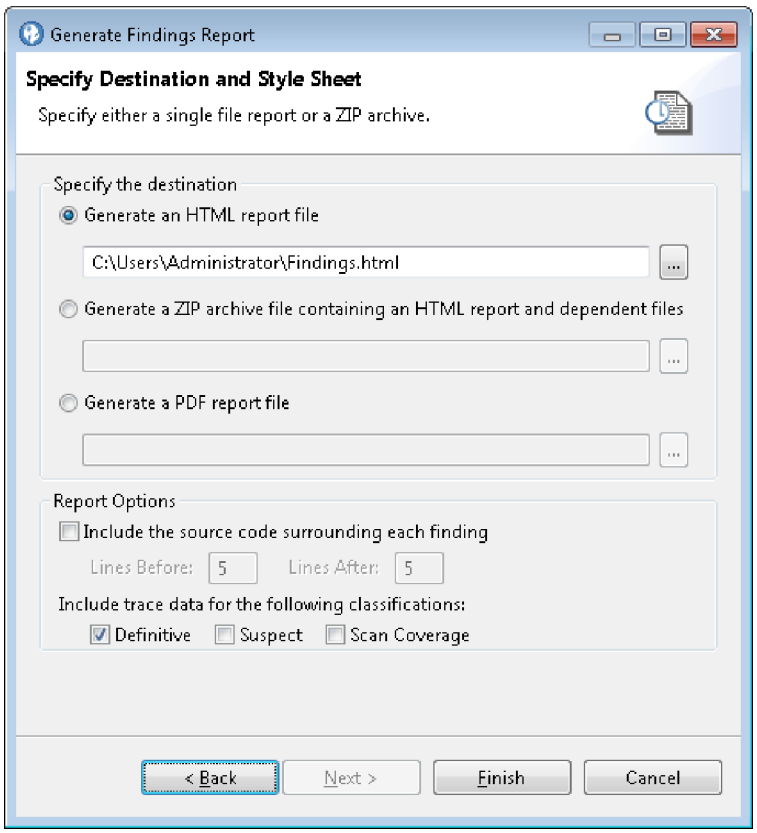
     註：如果您建立 PDF 格式的自訂報告（而不是發現項目報告），您可以指定報告中要包含的詳細程度：

     - **摘要**：包含每一個報告群組的計數
     - **詳細**：包含每一個 API 的每一個漏洞內容的計數
     - **綜合性**：包含由每一個 API 的每一個發現項目組成的表格
     - **標註**：包含所有發現項目，以及併入發現項目的任何附註、追蹤資料或程式碼片段
   - 如果要將程式碼片段併入報告中，請選取**併入每個發現項目周圍的原始碼**，指出在有漏洞的程式碼行前後，要併入報告的行數。

     提示：在「發現項目詳細資料」視圖的「報告」區段中，您也可以設定在報告中，要於發現項目前後併入的程式碼行數。

     產生報告之後，當您展開含有附註或程式碼片段的發現項目時，原始碼會出現在藍框內發現項目之下，或在黃色附註之下。粗體紅字強調有漏洞的程式碼行。
   - 如果要將 AppScan Source 追蹤資料併入報告中，請在**併入下列分類的追蹤資料**之下，選取一或多個分類（**明確**、**可疑**或**掃描涵蓋面**）。

   按一下**完成**來產生報告。



CWE/SANS Top 25 2011 報告

CWE/SANS Top 25 2011 報告是根據 *2011 CWE/SANS Top 25 Most Dangerous Software Errors*（2011 CWE/SANS Top 25 最危險的軟體錯誤）。

如果要進一步瞭解 *2011 CWE/SANS Top 25 Most Dangerous Software Errors*（2011 CWE/SANS Top 25 最危險的軟體錯誤），請參閱 http://cwe.mitre.org/top25/。

如果要瞭解 AppScan Source 支援的所有「一般弱點列舉 (CWE)」弱點，請參閱第 285 頁的第 15 章, 『CWE 支援』。

DISA 應用程式安全及開發 STIG 3.10 版報告

本主題提供「國防資訊系統局 (DISA) 應用程式安全及開發安全技術實作手冊 (STIG)」網站和指引文件的鏈結。

如果要瞭解「DISA 應用程式安全及開發 STIG」，請參閱 http://iase.disa.mil/。

「開放式 Web 應用程式安全專案 (OWASP)」Top 10 2013 報告

本主題提供「開放 Web 應用程式安全專案 (OWASP)」網站和指引文件的鏈結。

如果要瞭解 OWASP，請參閱 https://www.owasp.org/index.php/Main_Page。各種 OWASP 文件及安全風險的鏈結位置如下：https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project。

「開放式 Web 應用程式安全專案 (OWASP)」Mobile Top 10 報告

本主題提供「開放 Web 應用程式安全專案 (OWASP)」網站和指引文件的鏈結。

如果要瞭解 OWASP 行動式安全專案，請參閱 https://www.owasp.org/index.php/OWASP_Mobile_Security_Project。

付款卡產業資料安全標準 (PCI DSS) 3.2 版報告

此報告提供要確保符合「付款卡產業資料安全標準 (PCI DSS)」 所需的相關資料。

如需資訊，請參閱 https://www.pcisecuritystandards.org/security_standards/index.php。

Software Security Profile 報告

Software Security Profile 呈現與應用程式安全直接相關的應用程式性質綜合分析。它會提供特定專案軟體中重要安全特性的詳細審核。這份報告可協助您在認證軟體部署之前，進行加密、存取控制、記載和錯誤處理等需求的實作驗證。

該複合報告會識別潛在風險區，並提供降低那些風險的建議。這份報告能夠協助評量應用程式的整體安全，有助於進行相符性、原則和架構的審查。發現項目是以廣泛的原始碼靜態分析為基礎，所用的資料庫含有各種缺失、漏洞、業界專用標準以及一般的最佳實務。

Software Security Profile 會顯示下列資訊：

- 報告卡：包含彙總區段之報告詳細資料以及嚴重性指示器的鏈結。
- 概觀：彙總報告目的及說明應用程式配置。
- 度量：指出在專案的所有套件中，套件、類別、方法和程式碼行的總數。
- 詳細的發現項目（依種類）：報告所找到的每個漏洞種類，有漏洞種類名稱，以及表示漏洞嚴重性層次的圖示。

第 9 章 建立自訂報告

在「報告編輯器」中，您可以建立用來產生自訂報告的報告範本。

AppScan Source 發現項目報告或 AppScan Source 報告不一定會提供您所需要的確切資料；您的報告可能必須包含更多或更少的資訊。您可以利用 AppScan Source for Analysis 報告編輯器來建立自訂報告。

一般而言，當有下列需求時，您會建立自訂報告：

- 產生一份報告來對映至唯一安全原則，以及報告這個原則。您首先建立一份自訂報告，然後將這份報告套用於特定的評量。
- 定義及產生一份報告來強調某些唯一的發現項目和性質。
- 在現有的報告上進行修改及新增內容。

當您將報告範本儲存於 <data_dir>\reports（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述） 時，此報告可用來評量任何應用程式。當儲存於特定應用程式的目錄時，便可以利用這份報告來掃描這個應用程式或其中的任何專案。

開始建立或編輯 AppScan Source 報告之前，請先熟悉報告類型及每份報告的組成元素。當建立自訂報告時，您可以依照任何次序來對映報告元素。報告元素包括發現項目資訊、程式碼片段、追蹤資料和補救內容，以及文字和圖形元素。

報告編輯器

當使用「報告編輯器」時，您可以編輯自訂報告或範本，或建立一份新的報告。自訂報告包括發現項目報告所能使用的任何項目，例如，發現項目資訊、程式碼片段、AppScan Source 追蹤，以及補救內容，另外還有漏洞矩陣。開始設計新報告之前，建議您先在「報告編輯器」中修改現有的報告範本，以熟悉報告的建立程序。

「報告編輯器」由「報告佈置」、「種類」和「預覽」標籤組成。

- **報告佈置**：設計報告的外觀。您可在佈置中新增、移除和重新排序 AppScan Source 報告元素。
- **種類**：建立和編輯種類。種類是一個發現項目群組。種類用來識別要併入報告的發現項目、這些發現項目的分組方式，以及分組次序。
- **預覽**：在編輯之時，查看現行評量的報告。

這三個標籤共用的欄位如下：

- **檔案**：儲存之分組檔（唯讀）的路徑。在儲存檔案之前，這個欄位不會顯示任何項目。在儲存之後，分組檔是一個用來定義報告的 XML 檔。
- **名稱**：使用者定義的報告名稱。

用來儲存、開啟、建立、複製和產生自訂報告的工具列按鈕包括：

- **建立新報告**：建立新的自訂報告
- **從現有中新建報告**：從現有的報告範本中建立新的自訂報告

- **開啟儲存的報告**：開啟要編輯的分組檔
- **儲存**：將現行報告儲存在指定的檔案中
- **另存新檔**：將現行報告儲存在新檔案中
- **產生這份報告的實例**：建立目前開啟之評量的報告副本

提示：如果要檢視現有報告範例，請按一下**從現有中新建報告**，然後選擇其中一個 AppScan Source 報告範本。探索範本中的「報告佈置」和「種類」標籤，可讓您概括瞭解如何設計報告。

「報告佈置」標籤

「報告佈置」標籤含有「選用區」和「佈置」區段，以及一些可讓您指定每一頁面標頭或標底的區段。

頁面標頭和頁面標底

頁面標頭欄位可讓您指定出現在每一報告頁面頂端的文字，**頁面標底**欄位可讓您指定出現在每一頁面底端的文字。

選用區

「選用區」會顯示一份組成 AppScan Source 標準報告的元素清單。有些元素只會顯示已定義在「種類」標籤中之種類的相關資訊（請參閱表 19）。

表 *18.* 報告佈置選用區 - 不與種類相依的元素

報告元素	**說明**
文字標頭	將粗體文字區塊新增至報告佈置。
影像標頭	顯示調整到指定大小的影像（像素）。
AppScan Source 標頭	含有 AppScan Source 品牌行銷的報告標頭。
標題與日期	含有已掃描之項目名稱的報告標題，以及掃描日期和報告產生日期。
文字區塊	使用者定義的任何文字。也可以在**標籤**欄位中，新增文字區塊的標題。
漏洞矩陣	評量漏洞矩陣（顯示出現在「漏洞矩陣」視圖中的相同圖形）。
度量	識別在專案的所有套件中，套件、類別、方法，以及程式碼行數的總數。
掃描歷程	現行掃描的度量，以及相同目標掃描的歷程度量。

表 *19.* 報告佈置選用區 - 與種類相依的元素

報告元素	**說明**
報告卡	針對「種類」標籤中所定義的每一個種類，簡短分析其漏洞層次。包含彙總區段之報告詳細資料以及嚴重性指示器的鏈結。
漏洞分析	這份表格分析「種類」標籤中所定義之所有種類中的漏洞數目，並依嚴重性和分類區分

表 *19.* 報告佈置選用區 - 與種類相依的元素 *(*繼續*)*

報告元素	說明
局部報告卡	針對使用者在「種類」標籤中所指定的種類，分析其漏洞層次。
種類	列出「種類」標籤中所定義之所有已分類的發現項目資料。
種類	列出已定義在「種類」標籤中之一或多個種類中的所有發現項目。

佈置

當您從選用區新增項目時，它們會出現在「佈置」中。請使用區段工具列，來移除、修改或移動佈置中的項目。

「種類」標籤

您可以利用「種類」標籤，根據您選擇的組合、內容或發現項目，來新增含有發現項目的種類。之後，當您新增某些項目到「佈置」時，就可以使用種類。舉例來說，當您新增「漏洞分析」到「佈置」時，會在佈置中新增一份表格，其中含有所有種類中漏洞數目的分析（依嚴重性和分類區分）。「種類」標籤有兩個窗格，一個是種類的樹狀結構，另一個是用來編輯所選種類的屬性。每個種類都包含評量中能夠滿足所定義特定需求的發現項目。

可用的種類包括：

- 組合：組合種類由一份組合名稱清單組成。組合中的任何發現項目，只要名稱出現在這份清單中，就會出現在這個種類中。雖然您從現行評量中選擇組合，但由於組合是依名稱比對，您可以將組合種類套用於任何評量。
- 個別發現項目：請選擇要新增到種類中的特定發現項目。只會新增發現項目的 Snapshot 到報告中。如果發現項目新增到報告之後，您又修改它，變更不會反映在報告中。
- 「漏洞類型」、「機制」和「技術」內容：請從 AppScan Source 安全知識庫的 API 中，選擇各組內容及必要的內容。如果發現項目至少包含其中一個**內容**及所有**必要的內容**，它就會併入報告中。

這份表格識別種類窗格及組成窗格的項目。

表 *20.* 「種類」標籤屬性

屬性	說明	如何編輯
標籤	種類的簡短名稱，例如「緩衝區溢位」。標籤用來識別種類樹狀結構清單中的種類，它是自訂報告中的種類標題。	請在單行文字欄位中輸入標籤。
摘要	指出這個種類報告了多少發現項目的句子範本。在產生報告期間，實際的計數會取代 `%FindingCount%`。	輸入種類的簡要說明，按一下**新增計數**，將 `%FindingCount%` 變數放在詞組中的游標位置。
文字	簡短種類說明。	輸入文字來說明種類。

表 20. 「種類」標籤屬性 (繼續)

屬性	說明	如何編輯
內容（只限內容種類）	這個種類會報告至少有其中一個內容的發現項目。如果發現項目未完整具備所有列出的必要內容，發現項目就不會併到這個種類中。	請在工具列中，按一下**新增**，從「新增內容」對話框中選取一個內容。按一下**移除**，可以從清單中移除選取的項目。
必要的內容（只限內容種類）	在這個種類之下，含有所有必要內容及至少一個內容的發現項目會出現在報告中。	請在工具列中，按一下**新增**，從「新增內容」對話框中選取一個內容。按一下**移除**，可以從清單中移除選取的項目。
組合（只限組合種類）	指定要併入這個種類的組合名稱。	在「組合」區段中，按一下**新增組合**，從清單中選取組合。
發現項目（僅限於「發現項目」種類）	指定要併入這個種類的發現項目。	選取任何發現項目表格中的發現項目，然後按一下表格工具列上的**新增發現項目**，來新增選取的發現項目。如果有多個視圖含有選取的發現項目，會提示您選取包含您要新增之所選發現項目的視圖。 您也可以將發現項目表格中的發現項目，拖曳到「報告編輯器」視圖或「報告編輯器」中的表格，或直接拖曳到種類樹狀結構中的現有發現項目種類。

「預覽」標籤

當編輯範本時，您可以預覽 AppScan Source for Analysis 報告。請從「預覽」窗格中，按一下**預覽**來查看開啟之評量的報告。

產生自訂報告

本節主題中的程序說明如何從現有的自訂報告中，設計及產生報告。另外，您也可以建立新的報告。如果要編輯現有的報告，請開啟報告，然後遵循設計、修改和預覽程序。

- 第 201 頁的『從現有的自訂報告設計報告』
- 第 201 頁的『將種類併入報告中』
  - 第 201 頁的『新增組合至種類』
  - 第 201 頁的『新增發現項目至種類』
  - 第 202 頁的『新增內容至種類』
- 第 202 頁的『預覽報告』
- 第 202 頁的『儲存報告範本』

從現有的自訂報告設計報告

程序

1. 在「報告編輯器」視圖的工具列中，按一下**從現有中新建報告**。
2. 從現有的報告清單中，選取一個報告範本。在「佈置」窗格中，預覽報告範本。
3. 變更報告名稱、標頭和標底，或範本元素：
   a. 新增**頁面標頭**或**頁面標底**。頁面標頭和標底會出現在每個頁面中。
   b. 新增其他元素到報告中。請從**選用區**中選取您要的報告元素，然後按一下**插入**（必須分別插入每一個元素）。
   c. 刪除報告中的元素。從範本中選取要移除的元素，然後按一下**移除選取的報告元素**工具列按鈕。
4. 重新排序報告元素。請在預覽中選取一個元素，然後按一下工具列中的**上移選取的報告元素**或**下移選取的報告元素**，將報告元素上移或下移。
5. 在「佈置」窗格中，按兩下某個元素來編輯它，或選取元素，然後在工具列中，按一下**編輯選取的報告元素**。

   在產生的對話框中，進行您要的變更。比方說，如果要編輯文字區塊，請在「編輯文字區塊」對話框中，修改標籤和說明文字來進行變更。

   註：某些元素無法修改。

將種類併入報告中

定義好佈置之後，請決定要併入報告的種類。

程序

1. 在「種類」窗格中，按一下**建立新的內容種類**、**建立新的組合種類**或**建立新的發現項目種類**。
2. 輸入種類標籤（也就是可能含有計數的簡短種類摘要）及說明文字，來命名種類。

   利用箭頭工具列按鈕，將種類或子種類升級或降級。
3. 新增組合、發現項目或內容至種類中。

新增組合至種類

程序

1. 開啟包含組合的評量。如果評量尚未包含任何組合，您無法新增組合到報告中。
2. 在「組合」窗格中，按一下**新增組合**，然後指定要包含在種類中的一或多個組合。

新增發現項目至種類

程序

1. 開啟發現項目視圖，其中包含您要新增的發現項目。請選取您要的發現項目，將它們拖曳到發現項目表格中，或拖曳到「報告編輯器」中種類樹狀結構的節點。
2. 另外，您也可以在發現項目表格上方，按一下工具列上的**新增發現項目**，以併入其他視圖中所選取的發現項目。如果您在多個視圖中選取發現項目，您必須選取包含要新增到種類中之發現項目的視圖。
3. 從任何含有發現項目表格的視圖中，選取發現項目。

新增內容至種類

程序

1. 按一下**新增內容**（內容包括「漏洞」、「機制」和「技術」）。當您選取內容時，若有這個內容的知識庫說明，則會顯示這個說明。
2. 至少選擇一個內容，以及任何必要的內容。發現項目必須有**必要的內容**清單的所有內容，才能併到種類中。

   如果要建立子種類，請選取一個種類，然後按一下工具列的左移鍵或右移鍵按鈕。

預覽報告

當設計自訂報告時，您可以先預覽，再產生定案的報告。請在「預覽」窗格中，按一下**預覽**來顯示目前開啟之評量的報告。

儲存報告範本

從「報告編輯器」視圖工具列中，您可以按一下**儲存**來保留現行報告範本，或按一下**另存新檔**，將現行報告範本儲存到新檔案中。

如果您將報告範本儲存在應用程式檔案（.paf 或 .gaf）的相同目錄中，「自訂報告」精靈和「報告編輯器」視圖的選項清單就會提供這個報告範本，以用於該應用程式的後續掃描。如果您將它儲存在 <data_dir>\reports（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述） 中，就可以利用它來掃描任何應用程式。

第 10 章 自訂漏洞資料庫和型樣規則

本節說明如何自訂資料庫，以及將自訂的漏洞及其他常式整合到掃描中。

掃描程序有多個階段：

- 語言專用掃描是利用漏洞資料庫（或 AppScan Source 安全知識庫）來執行。
- 追蹤是利用漏洞資料庫來執行。
- 基於型樣的掃描是利用廣域型樣規則庫中的型樣規則來執行。

您可以利用自訂規則，調整 AppScan Source 安全知識庫來配合您的特定安全標準，並在整個企業中一致地套用這些標準。您也可以自訂型樣規則。

延伸 AppScan Source 安全知識庫

本節說明如何自訂資料庫，以及將自訂的漏洞及其他常式整合到掃描中。自訂規則可調整 AppScan Source 安全知識庫（或漏洞資料庫），以配合您的特定安全標準，並在整個企業內一致地套用這些標準。

指定自己的驗證和編碼常式，或將特定應用程式設計介面 (API) 定義為漏洞、接收槽和來源、污染傳播者或參考項目，往往會變得很重要。當建立這些規則時，您會自訂和延伸 AppScan Source 漏洞資料庫，它是 AppScan Source 安全知識庫的整合部分。新增自訂規則到資料庫之後，在掃描期間，AppScan Source for Analysis 會識別出這項規則。呼叫自訂 API 會顯示為安全發現項目或掃描涵蓋面發現項目 - 然後會報告發現項目。

例如，分析師可能會新增類型為 `BufferOverflow`、名稱為 `readBuffer( )` 的 API。在後續的掃描中，當 AppScan Source for Analysis 發現符合這個新 API 規格的漏洞時，就會參照這個 API。如需漏洞類型的詳細資料，請參閱 AppScan Source 安全知識庫（在主工作台功能表中，選取**說明** > **安全知識庫**）。

當您新增自訂驗證和編碼常式時，AppScan Source for Analysis 就不再將傳入及傳出這些常式的資料視為有漏洞。藉著新增自訂常式到知識庫，AppScan Source for Analysis 就能夠判定資料是否未經驗證或編碼，就從污染的輸入來源流動到輸出。

註：AppScan Source 安全知識庫不提供自訂記錄的線上說明，但會顯示漏洞類型的說明。

重要： 您必須有知識庫管理的許可權，才能變更 AppScan Source 安全知識庫。

建立自訂規則

您可以在「自訂規則」視圖中開啟「自訂規則精靈」，這個工具會帶您逐步建立自訂資料庫記錄。建立自訂規則之後，您可以在「自訂規則」視圖中檢視它們。表格會顯示簽章、語言和目的。

套用規則的專案必須存在於「瀏覽器」視圖**所有應用程式**之下的應用程式中，專案特定的驗證和編碼常式才會出現在「自訂規則」視圖中。

- **簽章**：簽章是完整的函數名稱。比方說，Java 簽章包括引數和傳回類型，例如，`com.test.vulnerable.VulnClass.vulnerable(java.lang.string;int):int`。
- **語言**：C/C++、Java、Visual Basic、Classic ASP 或 .NET
- **目的**：給定方法的自訂記錄類型，例如，`Validation.EncodingRequired` 常式、接收槽或來源。

提示：如果您要反覆掃描及新增自訂規則，然後在不變更原始碼的情況下重新掃描，以精簡您的程式碼庫評量，您可以設定專案內容來使用漏洞分析快取，掃描的時間會縮短許多。如果要執行這個動作，請在專案內容中，選取**啟用漏洞分析快取**勾選框。如果要瞭解如何設定專案內容，請參閱第 219 頁的『「選取的專案概觀」標籤』的使用指示。

使用「自訂規則」精靈

「自訂規則精靈」可協助您新增方法到 AppScan Source 安全知識庫中。大部分自訂規則都是廣域範圍（適用於所有專案）。自訂無追蹤發現項目、來源、接收槽和污染傳播者，一律是廣域範圍。自訂驗證/編碼常式不是廣域範圍。

註：「自訂規則精靈」不會驗證您的選項。例如，您可以定義一個自訂規則，將方法識別為污染傳播者及接收槽，這並非是有效實務。

「自訂規則精靈」會逐步指引您在知識庫中定義及新增下列項目的程序：

- 接收槽和來源
- 污染傳播者
- 不易遭到污染的應用程式設計介面 (API)
- 漏洞
- 產生不含追蹤之發現項目的 API
- 非驗證/編碼常式的 API
- 污染的回呼
- 參考資訊發現項目

來源（污染）

提供程式輸入的方法，可能形態異常或具有惡意。

接收槽（很可能遭到污染）

從程式（或程式的可見部分）外傳送資料至檔案、網路、資料庫、其他程式庫或裝置的 API，很可能遭到惡意輸入。

污染傳播者

將方法標示為污染傳播者，意謂著倘若傳給 API 的任何引數是從未經驗證的輸入資料（污染的資料）衍生而來，在呼叫之後，其他引數所參照的非固定資料，以及回覆值，也都有可能遭到污染。這些資料必須先驗證或編碼，才能傳到接收槽。發生這個狀況，通常是因為將受污染之引數的資料複製或附加到其他引數，或傳回這些資料。

不容易遭受污染

將 API 標示為不容易遭受污染（不是污染傳播者），意謂著以未經驗證的輸入資料（污染的資料）所衍生的引數來呼叫 API，不會使 API 產生不安全或惡意的行為。

如果污染的資料傳入呼叫中，但這個呼叫標示為不容易遭受污染，在追蹤方面，AppScan Source 會忽略這個呼叫。 AppScan Source 追蹤不會報告遺失的追蹤，也不會將傳送的資料視為受到污染。

註： 如果污染的資料所沾染的方法既不是驗證或編碼常式、接收槽、污染傳播者，也不容易遭受污染，則這個方法會報告為遺失的追蹤。非常數的引數和回覆值不一定會遭到污染。 AppScan Source 追蹤呼叫圖會顯示遺失的追蹤。

無追蹤發現項目

一律會顯示為發現項目，但不會產生追蹤的方法或 API。

不是驗證/編碼常式

將 API 標示為**不是驗證/編碼常式**，指出這個 API 不驗證任何資料。

污染的回呼

回呼是在程式碼之中，通常由其他程式碼來呼叫的一個常式（例如，從低階架構內呼叫）。回呼是以引數的形式傳給其他程式碼，之後，可用可能已污染的引數來呼叫它。如果您懷疑可能有污染的資料傳入回呼的引數，您可以將它標示為污染的回呼。如此一來，就能看到污染的資料在常式中的流程。

標示為污染回呼的常式將被當作呼叫圖的根目錄來進行分析（換言之，是由某個不明的外部呼叫端來呼叫），它的所有輸入引數都視為已遭污染。因此，AppScan Source 會報告含有追蹤資料的發現項目，追蹤資料是由受污染之回呼的引數開始。

如果在其他環境定義中，您的應用程式碼呼叫相同的常式，在處理它時，不會有任何特殊的污染考量。在這些環境定義中，會進行平常的分析。

參考資訊

識別為參考資訊發現項目的各行程式碼不一定有漏洞，但也應該併入安全審核中。

新增規則

這個作業主題說明利用「自訂規則精靈」來新增自訂規則的程序。

關於這項作業

註：新增或移除安全或掃描涵蓋面發現項目，以及變更嚴重性，會影響專案的「V 密度」。

程序

1. 從「自訂規則」視圖中，按一下**啟動自訂規則精靈**按鈕來開啟精靈。
2. 在**選取應用程式、專案和檔案**頁面中，選取要套用規則的**應用程式**和**專案**。請確定現行應用程式和專案與要新增到知識庫之項目的原始碼相關。如果有可用者，請選取**配置**。

3. 在**範圍**區段中，設定掃描的範圍。依掃描的語言而定，範圍選項如下：

表 *21.* 依語言提供的專案檔選項

語言	專案檔選項
.NET	• **掃描整個專案以尋找方法簽章** • **選取專案外的一或多個檔案** .NET 專案包括任何有效的組譯碼，通常是 .dll 或 .exe 檔。
Java	• **掃描整個專案以尋找方法簽章** • **選取專案中的一或多個檔案** • **選取專案外的一或多個檔案** Java 專案包括 .jar 或 .class 檔，或類別檔的目錄階層。
C/C++	• **掃描整個專案以尋找方法簽章** • **選取專案中的一或多個檔案**
Visual Basic	掃描 FRM（表單）檔、CLS（類別）檔，以及 BAS（基本）
典型 ASP	只掃描 ASP 檔

   - **掃描整個專案以尋找方法簽章**是預設掃描模式。這個模式會掃描整個專案，傳回所有可用的簽章。這個掃描模式可能很耗時間。
   - **選取專案中的一或多個檔案**選項會將包含需要自訂規則之方法的特定專案檔隔離出來。
   - **選取專案外的一或多個檔案**選項會識別在這個專案之外，但要併入掃描中的檔案。
4. 在**快取**區段中，選取重新讀取修改過的專案或程式碼的勾選框。也會清除漏洞分析快取（如果現行專案設為快取漏洞分析，下一次掃描時，會重建漏洞分析快取）。
5. **字串分析**：字串分析會監視 Java 或 Microsoft .NET 專案中的字串操作。它能夠自動偵測消毒器和驗證器常式。當進行這項偵測時，可以減少誤判 (false positive) 及假性無侵害攻擊 (false negative) 的情況。選取**啟用字串分析來尋找驗證器/消毒器功能**勾選框來啟用字串分析。**將匯入的規則套用於廣域範圍**勾選框會決定所發現的消毒器或驗證器常式，是否應該套用到單一專案或廣域層次（套用至所有專案）。

   註：套用字串分析，掃描可能會變慢。因此，建議只應在程式碼變更之後才套用，而且應於後續掃描中停用。另外，還應該將發現的常式視為建議，由審核員來進行檢查。您可以在「自訂規則」視圖中檢視這些常式。
6. 按**下一步**，進入精靈的下一頁。
7. 在**選取方法**頁面中，請執行下列動作：
   a. 選取要新增到知識庫的一或多個方法。方法是有漏洞 API 的名稱。

      您可以利用兩種方式來過濾方法清單：

- 自動過濾：在**過濾器**欄位中，輸入過濾文字。當您輸入時，過濾器會自動套用於方法清單。這是預設的過濾模式。
- 手動過濾：在**過濾器**欄位中輸入過濾文字，然後按一下**過濾器**按鈕（或按 Enter 鍵），將過濾器套用於清單。當大量方法造成自動過濾延遲時，您可能會想使用手動過濾。

在這兩種情況下，星號 (*) 和問號 (?) 字元可用來作為萬用字元。星號符合零或多個字元所組成的任意群組，問號則符合任何單一字元。

如果要變更過濾模式，請利用**過濾器**按鈕，按它兩下來進行切換，或利用鍵盤來導覽到這個按鈕，然後按空格鍵。當開啟手動過濾時，**過濾器**按鈕會顯示未按下狀態，它的浮動說明是**套用過濾器（按兩下或按空格來自動過濾）**。當開啟自動過濾時，這個按鈕會顯示按下狀態，它的浮動說明是**手動過濾**。

如果要妥善檢視方法清單，您可以使用展開及收合動作。如果要展開或收合整個樹狀結構，請按一下滑鼠右鍵，選取**全部展開**或**全部收合**。如果要展開套件或類別及其所有子項目，請用滑鼠右鍵按一下套件或類別，然後選取**展開子項**。

如果要選取多個方法，請使用鍵盤的 Ctrl 或 Shift 鍵。

選取**顯示完整簽章**勾選框，可以在樹狀結構中顯示方法的完整簽章。例如，完整的 Java 簽章包括套件、類別、方法、引數類型，以及傳回類型，例如 `com.test.vulnerable.VulnClass.vulnerable(java.lang.string;int):int`。

b. 識別掃描是否應該將方法標示為下列項目之一：
   - 第 204 頁的『來源（污染）』
   - 第 204 頁的『接收槽（很可能遭到污染）』
   - 第 204 頁的『污染傳播者』
   - 第 205 頁的『不容易遭受污染』
   - 第 205 頁的『無追蹤發現項目』
   - 第 205 頁的『不是驗證/編碼常式』
   - 第 205 頁的『污染的回呼』
   - 第 205 頁的『參考資訊』

8. 如果您要將方法新增為第 205 頁的『不容易遭受污染』、第 205 頁的『不是驗證/編碼常式』、第 204 頁的『污染傳播者』或第 205 頁的『污染的回呼』，請按一下**完成**，將記錄新增至 AppScan Source 安全知識庫。
9. 如果您要將方法新增為第 204 頁的『來源（污染）』或第 205 頁的『參考資訊』，請執行下列動作：
   a. 按**下一步**，繼續進行**指派規則屬性**頁面。
   b. 對於已新增的每一個方法：選取一或多個要指派給方法的內容。方法的**類型**直欄會更新，以指出自訂規則將產生之發現項目的漏洞類型。

      提示：如果要將多個相同的內容新增至多個方法，請使用鍵盤 Ctrl 或 Shift 鍵選取多個方法，然後選取要指派給方法的內容。
   c. 按一下**完成**，將記錄新增到 AppScan Source 安全知識庫。

10. 如果您要將方法新增為第 204 頁的『接收槽（很可能遭到污染）』，請執行下列動作：
    a. 按**下一步**，繼續進行**指派規則屬性**頁面。
    b. 對於已新增的每一個方法，請執行下列動作：
        - 選取漏洞影響的**嚴重性**層次：**高**、**中**或**低**。
        - 選取要套用至方法的**漏洞類型**。

        提示：如果要將多個相同的內容新增至多個方法，請使用鍵盤 Ctrl 或 Shift 鍵選取多個方法，然後選取要指派給方法的內容。
    c. 按一下**完成**，將記錄新增到 AppScan Source 安全知識庫。
11. 如果您要將方法新增為第 205 頁的『無追蹤發現項目』，請執行下列動作：
    a. 按**下一步**，繼續進行**指派規則屬性**頁面。
    b. 對於已新增的每一個方法，請執行下列動作：
        - 選取漏洞影響的**嚴重性**層次：**高**、**中**或**低**。
        - 選取要指派給方法的**分類**：**明確**、**可疑**或**配置**。
        - 選取要套用至方法的**漏洞類型**。

        提示：如果要將多個相同的內容新增至多個方法，請使用鍵盤 Ctrl 或 Shift 鍵選取多個方法，然後選取要指派給方法的內容。
    c. 按一下**完成**，將記錄新增到 AppScan Source 安全知識庫。

可能性**規則屬性**

`Attribute.Likelihood.High` 和 `Attribute.Likelihood.Low` 屬性是內建規則的一部分，建立自訂規則時可使用它們。

在 AppScan Source 中，可能性代表可惡意探索安全發現項目的機率或機會。AppScan Source 使用 https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology#Step_2:_Factors_for_Estimating_Likelihood 所呈現的可能性定義，並根據追蹤內容判斷可能性來精簡它。假設有一組追蹤內容 - 例如，來源 API 名稱、來源 API 類型、來源技術或來源機制 - AppScan Source 判斷未來可利用特定漏洞惡意探索追蹤的可能性。

可能性關聯於追蹤的來源元素。來源是對程式的輸入，例如：檔案、Servlet 要求、主控台輸入或 Socket。對於大部分的輸入來源而言，傳回的資料內容和長度都是沒有限制的。當輸入不受檢查時，即被認為是污染的來源。

可能性範例包括：

- 假設追蹤含有 HTTP 來源（例如，`Request.getQueryString`）及跨網站 Scripting 接收槽（例如，`Response.write`），則判斷具有高可能性，而增加對發現項目的信賴度。
- 假設追蹤含有系統內容來源（例如，`getProperty`）及跨網站 Scripting 接收槽（例如，`Response.write`），則判斷具有低可能性，而減少對發現項目的信賴度。

可能性是用來識別具有高優先順序的可操作發現項目，必須立即處理或修正。它關聯於一個可任意惡意探索的污染來源，並可提供您更精細的方法將這些發現項目分類。可能性是以一個關聯於污染來源的屬性，儲存在 AppScan Source 漏洞資料庫中。此特性立即可用。

我們進行了廣泛的研究，以判斷來源的可能性因素。您可以使用「自訂規則精靈」，把可能性資訊新增至您加入規則庫的新污染來源。這會改進從掃描產生的發現項目的分類，進而提高整體分類工作流程的效率。

在「自訂規則精靈」中，您可以對**可能性**內容設定兩個值（**高**和**低**）。**高**值表示該來源很可能遭到污染。換句話說，污染進入系統的屏障極低，使得攻擊者很容易透過手動或自動化方式提交惡意資料。**低**值表示透過此來源輸入惡意資料的屏障極高。這可能表示為了將污染引進來源中，攻擊者必須對系統非常瞭解，而且要有許可權可對受害者網路進行操作。

透過 AppScan Source 追蹤來自訂輸入/輸出追蹤

部分應用程式（尤其是 Web 應用程式）需要輸入/輸出追蹤，以識別與 SQL 注入、指令注入、跨網站 Scripting 攻擊相關的安全漏洞。透過 AppScan Source 追蹤，您可以指定一個驗證常式，它可用來消除任何漏洞報告。如果輸入未經驗證，就會將所有其他輸出標示為漏洞。

使用者定義的驗證常式是處理輸入資料，讓資料能夠安全傳給輸出常式的常式。如果驗證常式先處理輸入資料，再將它傳給輸出常式，就不會有輸入驗證漏洞。開發人員可以指定他們自己用來搭配追蹤的輸入驗證和編碼常式。

利用基於型樣的規則自訂

基於 AppScan Source 型樣的掃描是以自訂搜尋準則為基礎的原始碼分析。基於型樣的掃描類似於 grep（grep 會在一或多個檔案中，搜尋給定的字串或型樣）。執行分類的審核員或安全分析師，有可能利用基於型樣的掃描，在特定應用式程式中，或在專案中，搜尋特定的型樣。將型樣定義為漏洞類型之後，掃描原始碼時，會將這個型樣識別為漏洞。當 AppScan Source 找到相符者時，項目會出現在發現項目表格中。立即可用的 AppScan Source 規則庫包含預先定義的規則及規則集（規則的集合）。

基於型樣的掃描會搜尋正規表示式。正規表示式（通常稱為型樣）是一個字串，用來根據特定語法規則來說明或比對一組字串。您可以透過建立規則來指定搜尋。規則類似於您在「自訂規則」視圖中，新增到 AppScan Source 安全知識庫的自訂規則。當建立規則時，您會定義嚴重性、分類、漏洞類型及其他準則。

第 248 頁的『「型樣規則庫」視圖』可讓您建立新的型樣規則和規則集 - 及修改或移除現有的規則和規則集。然後，您可以使用所選取的應用程式的「內容」視窗、所選取的專案的「內容」視圖或掃描配置，來套用型樣規則和規則集（您也可以啟動對話框，它可讓您建立這些視圖的新規則）。如果要進一步瞭解如何套用規則和規則集，請參閱第 215 頁的『套用型樣規則和規則集』。

可建立的型樣規則範例包括：

- 檔名型樣相符項
- 含有多重型樣的單一規則
- 缺席規則

註： 您必須有**管理型樣**許可權，才能建立型樣規則或規則集 - 或修改及移除自訂規則和規則集。

型樣規則集

型樣規則集是型樣規則的集合。您可以新增型樣規則集，也可以修改或移除現有的規則集。 AppScan Source 提供一系列語言專用的型樣規則集，可供您選擇來套用至專案或應用程式（例如，您可能想要將 **Java** 型樣規則集套用至 **Java/JSP** 專案）。

第 248 頁的『「型樣規則庫」視圖』可讓您建立新的型樣規則和規則集 - 及修改或移除現有的規則和規則集。然後，您可以使用所選取的應用程式的「內容」視窗、所選取的專案的「內容」視圖或掃描配置，來套用型樣規則和規則集（您也可以啟動對話框，它可讓您建立這些視圖的新規則）。如果要進一步瞭解如何套用規則和規則集，請參閱第 215 頁的『套用型樣規則和規則集』。

AppScan Source 隨附的部分型樣規則集不含任何規則。您可以將組織適用的規則新增到這些規則集。這些規則集包括：

- ColdFusion
- JQuery
- 用戶端 JavaScript
- Visual Basic 6
- MooTools

提示：在「型樣規則庫」視圖中，用滑鼠右鍵按一下某個規則集，然後選取**內容**來開啟對話框，其顯示該規則集的相關資訊。「規則集內容」對話框提供了型樣規則集內的規則數目，以及與其他規則集的上下代關係之類的資訊。它也可以讓您修改規則集的**顯示名稱**和**專案類型**。

註： 您必須有**管理型樣**許可權，才能建立型樣規則或規則集 - 或修改及移除自訂規則和規則集。

在「型樣規則庫」視圖中建立規則集

型樣規則集是型樣規則的集合。如果要瞭解如何建立規則集，請遵循這個主題的指示。

開始之前

註： 您必須有**管理型樣**許可權，才能建立型樣規則或規則集 - 或修改及移除自訂規則和規則集。

程序

1. 在「型樣規則庫」視圖中，按一下**新增規則集**。
2. 在「新增規則集」對話框的**名稱**欄位中，輸入規則集的名稱。
3. 選取一個以上將套用此規則集的專案類型。
4. 按一下**確定**。
5. 新的規則集會出現在規則集清單中。您可以利用兩種方式來移入規則集：
   a. 在型樣規則區段中，選取一個或多個規則，並將它們拖放到規則集中。
   b. 在型樣規則區段中，選取一個規則或多個規則，然後用滑鼠右鍵按一下選擇，並選取**新增至規則集**功能表項目。在「選擇規則集」對話框中，選取您要將一或多個規則新增到其中的規則集。

修改和移除規則集

您可以在「型樣規則庫」視圖中，修改和移除立即可用的型樣規則集，以及您所建立的規則集。

註： 您必須有**管理型樣**許可權，才能建立型樣規則或規則集 - 或修改及移除自訂規則和規則集。

修改規則集

可對規則集進行這些修改：

- 您可以遵循第 210 頁的『在「型樣規則庫」視圖中建立規則集』中移入新規則集的指示，將規則新增至現有的規則集。
- 如果要移除規則集的一個規則或多個規則，請選取要移除的規則，然後執行下列動作之一：
  - 按一下**從規則集移除規則**。
  - 按一下滑鼠右鍵選取**從規則集移除規則**。
- 您可以利用兩種方式，新增規則集到另一個規則集中：
  - 選取一個規則集，將它拖放到另一個規則集中。
  - 用滑鼠右鍵按一下規則集，選取**新增規則集為子項**，然後在「選擇規則集」對話框中，選取要新增為母規則集的規則集。
- 您可以修改規則集的**顯示名稱**和**專案類型**：用滑鼠右鍵按一下規則集，並選取**內容**來開啟「規則集內容」對話框。在這個對話框中，您可以編輯**顯示台稱**欄位 - 或者，您可以按一下**專案類型**欄位的**編輯**按鈕，來選取一個或多個專案類型。

移除規則集

如果要移除規則集，請選取規則集，然後執行下列動作之一：

- 按一下**移除規則集**。
- 按一下滑鼠右鍵選取**移除**。

型樣規則

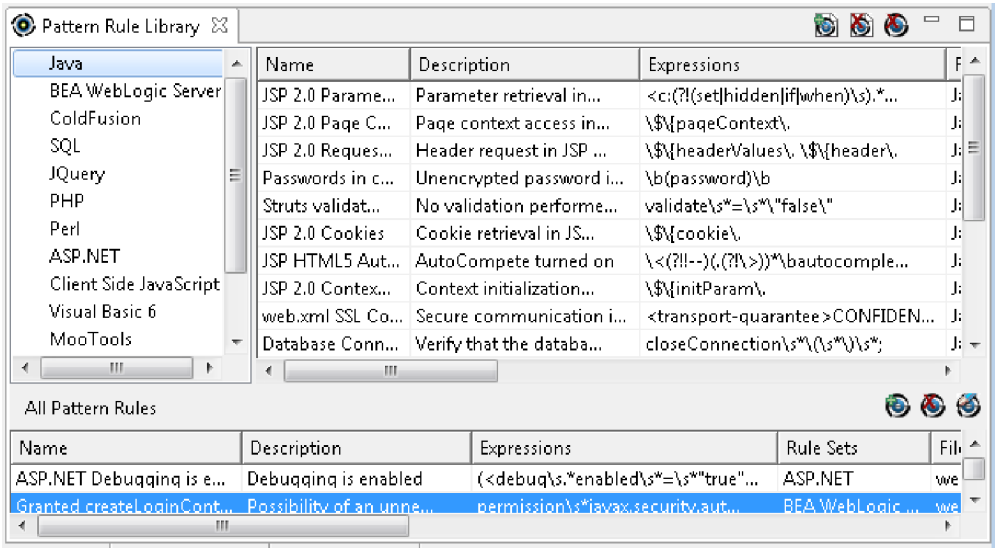
AppScan Source 文字規則可以是「延伸廣域正規表示式列印 (egrep)」、「廣域正規表示式 (grep)」或 Perl 正規表示式。這些正規表示式（字串值使用整組英數和特殊字元的表示式）符合規則。

字元	說明
^	開始於
$	結束於
\n、\t 或 \r	文字換行、定位點、返回
[xyz]	任何列出的字元
[^abx]	任何未列出的字元
[a-fA-F0-9]	任何十六進位字元
.	任何字元
|	其中之一

字元	說明
\	取消特殊字元意義 \$ \^ \\ \?

型樣規則是儲存在廣域型樣規則庫中（在 AppScan Source 資料庫 中），各專案和應用程式可以共用它們。所有使用者也能夠共用規則和規則集。規則是依參照來新增，您只要移除相關物件中的參照，不需要刪除基礎規則，就能夠將它們停用。

您可以在「型樣規則庫」視圖中、「瀏覽器」視圖的「內容」標籤中或在掃描配置中建立規則。當您安裝 AppScan Source 時，「型樣規則庫」視圖會顯示 AppScan Source 提供的規則。在這個視圖中，您可以編輯、刪除或建立規則。



重要： 您可以新增或移除搜尋準則，但每個基於型樣的規則，至少都必須有一個搜尋準則。

搜尋文字型樣

在給定的原始檔內，基於型樣的掃描會依副檔名來搜尋各檔案內的文字型樣，使搜尋能夠在原始檔、XML 配置檔及其他文字檔內進行。

例如，您可以建立型樣搜尋，以確保不會將不當的電子郵件位址寫在您的應用程式中。在這個情況下，如果您想要確定應用程式不會使用公司電子郵件位址，您可以搜尋 `.*@mycompany.com` 之類的型樣。

範例

這個型樣尋找	型樣
電子郵件位址	`[A-Za-z]\.[A-Za-z]@[A-Za-z][A-Za-]\.com`
型樣的所有實例，例如 `passWord =`	`[Pp][Aa][Ss][Ss][Ww][Oo][Rr][Dd]\W*=`
MD5 雜湊演算法的任何實例	`getInstance[[:space:]]*\([[:space:]]*"MD5`

建立型樣規則

可在「型樣規則庫」視圖中、專案或應用程式的「內容」視圖中或在掃描配置中建立規則。

開始之前

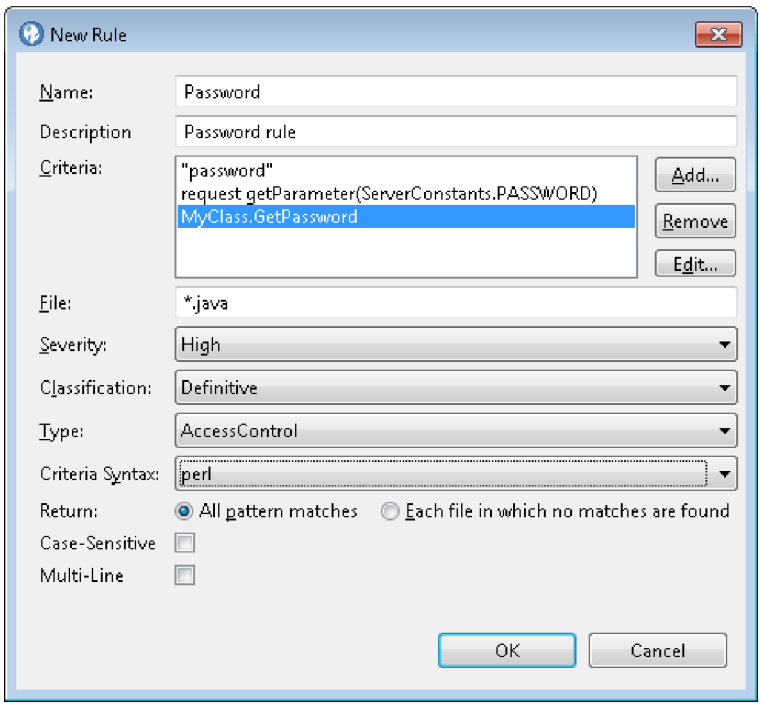
註： 您必須有**管理型樣**許可權，才能建立型樣規則或規則集 - 或修改及移除自訂規則和規則集。

規則是在「新增規則」對話框中建立：

- 如果要在「型樣規則庫」視圖中開啟此對話框，請按一下**新增規則**。
- 在掃描配置中，選取「型樣分析」標籤，然後選取**型樣分析**勾選框。在該標籤的「型樣規則」區段中，按一下**新增**，以開啟「新增型樣規則」對話框。在此對話框中，按一下**建立新規則**，以開啟「新增規則」對話框。
- 如果要從所選取的應用程式或專案的「內容」視圖中開啟此對話框，請選取「內容」視圖**規則和規則集**標籤，按一下**新增**，或在「規則」區段內按一下滑鼠右鍵，並選取**新增**。在「選擇規則」對話框中按一下**新增規則**。

程序

1. 在「新增規則」對話框中，為該規則**命名**。
2. 選擇性的： 新增規則的**說明**。
3. 新增**準則**。按一下**新增**，輸入每個規則的正規表示式。
4. 識別檔案類型，如 `*.java` 或 `*.xml`。您可以輸入任何含有或不含萬用字元的檔案類型。
5. 選擇性的： 選取**嚴重性**：
   - **高**
   - **中**
   - **低**
   - **參考資訊**
6. 選擇性的： 選取**分類**：
   - **明確**
   - **可疑**
   - **掃描涵蓋面**
7. 選擇性的： 選取掃描所要搜尋的漏洞類型。（如需漏洞類型的其他詳細資料，請參閱AppScan Source 安全知識庫）



8. 選擇性的： 選取準則語法：
   - **egrep**
   - **grep**
   - **perl**
9. 選擇性的： 確認所傳回的結果包括**所有型樣相符項**或**每個找不到相符項的檔案**。當找不到相符項時，型樣就是缺席規則。
10. 選擇性的： 如果型樣比對要區分大小寫，請選取**區分大小寫**勾選框。
11. 選擇性的： 如果規則應符合橫跨多行的型樣，請選取**多行**勾選框。
12. 按一下**確定**來確認規則中的正規表示式有效。然後，該規則就會新增至型樣規則庫中。

修改和移除型樣規則

您可以在「型樣規則庫」視圖中，修改和移除您建立的型樣規則。

註： 您必須有**管理型樣**許可權，才能建立型樣規則或規則集 - 或修改及移除自訂規則和規則集。

修改規則

如果要編輯規則，請選取它，然後執行下列動作之一：

- 按一下**編輯規則**。
- 按一下滑鼠右鍵，選取**編輯**。

這時會開啟「編輯規則」對話框，供您修改規則名稱以外的任何設定。

移除規則

請選取一或多個規則，然後執行下列動作之一：

- 按一下**移除規則**。
- 按一下滑鼠右鍵選取**移除**。

套用型樣規則和規則集

規則和規則集是在「內容」視圖的應用程式或專案層次套用 - 或在掃描配置中套用。套用規則來掃描應用程式或專案 - 或使用包含規則的掃描配置 - 之後，規則搜尋的結果會出現在包含發現項目的視圖中。

在掃描配置中套用規則和規則集

如果要啟用基於型樣的掃描，請選取**型樣分析**勾選框。當您這麼做時，**型樣規則集**和**型樣規則**區段會變成已啟用：

- 如果要新增規則集，請按一下**型樣規則集**區段中的**新增**。這時會開啟「新增型樣規則集」對話框，讓您選擇一個以上的規則集。當您選取規則集時，它包含的規則會顯示在對話框的右邊，而套用規則集的專案類型則列在**專案類型**欄位中。按一下**確定**，即可新增所選取的規則集。
- 如果要新增規則，請按一下**型樣規則**區段中的**新增**。這時會開啟「新增型樣規則」對話框，讓您選擇一個以上的規則。您也可以按一下**建立新規則**來建立新規則（請參閱第 213 頁的『建立型樣規則』）。如果您建立新規則，它將新增至清單中且已選取。在選取或建立規則之後，請按一下**確定**，將它們新增至掃描配置。

  提示：在「新增型樣規則」對話框中，工具提示說明會指出對每一個規則所使用的表示式。

在「內容」視圖中套用規則和規則集

請在「瀏覽器」視圖中選取專案或應用程式，然後在它的「內容」視圖的「型樣規則和規則集」標籤中，進行下列修改。指定要套用於應用程式或專案的規則和規則集之後，請儲存應用程式或專案內容。應用程式或專案的後續掃描，會併入這些規則。

- 如果要新增規則集，請按一下**規則集**區段中的**新增**，或用滑鼠右鍵按一下此區段，並選取**新增**。這時會開啟「選擇規則集」對話框，讓您選取要新增的規則集。
- 如果要移除規則集，以免在應用程式或專案的掃描期間使用它，請選取它並按一下**移除**，或用滑鼠右鍵按一下此規則集，並選取**移除**。
- 如果要新增規則，請按一下**規則**區段中的**新增**，或用滑鼠右鍵按一下此區段，並選取**新增**。這時會開啟「選擇規則」對話框，讓您選取要新增的規則。在這個對話框中，您也可以按一下**新增規則**來建立新規則（請參閱第 213 頁的『建立型樣規則』）。如果您建立新規則，它將新增至清單中且已選取。在選取或建立規則之後，請按一下**確定**來新增它。
- 如果要移除規則，以免在應用程式或專案的掃描期間使用它，請選取它並按一下**移除**，或用滑鼠右鍵按一下此規則，並選取**移除**。您也可以複選多個規則，然後使用這些動作來移除它們。

掃描配置視圖

「掃描配置」視圖可讓您建立啟動掃描時可用的配置。您也可以使用該視圖來設定預設掃描配置。在掃描配置中，您可以指定掃描期間要使用的來源規則，也可以包含許多掃描設定。掃描配置中的設定通常可以獲得較佳的掃描結果，而儲存這些設定的能力更可讓掃描變得輕鬆又有效率。

「掃描配置」視圖有這些主要區段：

- 第 101 頁的『掃描配置管理』
- 第 101 頁的『「一般」標籤』
- 第 102 頁的『「污染流分析」標籤』
- 第 103 頁的『「型樣分析」標籤』

掃描配置管理

使用這個區段來選取、新增、移除、儲存及共用掃描配置，以及將掃描配置設為預設值。

- 如果要建立新的掃描配置，請按一下**新建**。完成掃描配置設定之後，按一下**儲存**，儲存變更。如果要將掃描配置設為預設值，請在儲存之後按一下**選取為預設值**。如果要瞭解如何使用預設掃描配置，請參閱第 93 頁的『掃描原始碼』。
- 如果要使用現有的掃描配置，請從清單中選取它：
  - 如果您要修改掃描配置設定，請按一下**儲存**以儲存變更（可以切換至不同掃描配置來捨棄不要的變更，然後按一下**捨棄**）。
  - 如果要移除選取的掃描配置，請按一下**刪除**。
  - 如果要複製掃描配置，請按一下**複製**。這樣會根據原始掃描配置的設定來建立新的掃描配置。
  - 如果要將掃描配置設為預設值，請按一下**選取為預設值**。如果要瞭解如何使用預設掃描配置，請參閱第 93 頁的『掃描原始碼』。
  - 如果要與其他人共用掃描配置，請按一下**共用**。這樣會將掃描配置儲存到 AppScan Source 資料庫。

    註：如果要共用掃描配置 - 或修改或刪除共用的掃描配置 - 您必須具有**管理共用配置**許可權。如果要瞭解設定許可權的相關資訊，請參閱《*IBM Security AppScan Source* 安裝與管理手冊》。

註：AppScan Source 提供內建掃描配置。無法修改或移除這些配置。您可以在清單中選取它們來複製或檢視其設定。

「一般」標籤

基本資訊

這個區段可讓您命名掃描配置並提供說明。

過濾器

在此區段中，您可以選擇一個以上的過濾器，每當使用掃描配置時即將其套用至掃描。選取過濾器時，您可以選擇 AppScan Source 選擇的過濾器，或共用的過濾器，或您自己建立的過濾器。如需詳細資料，請參閱第 96 頁的『管理掃描配置』。

「污染流分析」標籤

污染流分析

啟用及設定污染流分析的範圍。

掃描規則

使用這個區段來決定掃描時採用的來源規則。

來源是對程式的輸入，例如：檔案、Servlet 要求、主控台輸入或 Socket。藉由排除部分來源規則，您可以加速掃描，並且避免不想要的輸入產生的偵測漏洞。

規則會標示著規則內容，指出與規則相關的特定漏洞、機制、屬性或技術。這些內容分組至規則集，對應於一般相關的規則集。藉由指定規則集或個別規則內容，您可以限制包含在掃描的來源規則。

- 選取掃描時要包含的一或多個漏洞類型（在規則集中依類型來組織）：
  - **全部**：如果選取此選項，全部支援來源產生的漏洞，都會被偵測到。
  - **使用者輸入**：如果選取此選項，將會偵測到一般使用者輸入產生的漏洞。
  - **Web 應用程式**：如果選取此選項，會偵測到 Web 應用程式風險產生的漏洞。
  - **錯誤處理和記載**：如果選取此選項，會偵測到錯誤處理和記載機制產生的漏洞。
  - **環境**：如果選取此選項，會偵測到配置檔、系統環境檔案和內容檔產生的漏洞。
  - **外部系統**：如果選取此選項，會偵測到外部實體產生的漏洞。
  - **資料儲存庫**：如果選取此選項，會偵測到資料儲存庫（例如資料庫和快取）產生的漏洞。
  - **不尋常事物**：如果選取此選項，會偵測到常式出現的漏洞，其通常不屬於正式作業的一部分。
  - **檔案系統**：如果選取此選項，會偵測到來自檔案系統的漏洞。
  - **機密資料**：如果選取此選項，會偵測到來自機密資料的漏洞。

  這個區段中的每一個規則集都有浮動說明。
- 選取要併入掃描中的個別掃描規則內容：按一下**捨棄選取的規則集，讓我選取個別規則內容**。這時會開啟「選取規則內容」對話框，讓您選擇個別的規則內容。如果完成這個對話框，則會捨棄任何已選取的規則集。掃描時，將會使用含有所選規則內容的掃描規則。

進階設定

這個區段僅適用於進階使用者。它包含各種可改進掃描結果的設定。這個區段中的每一個設定都有浮動說明。

「型樣分析」標籤

型樣分析

使用掃描配置時，可使用此區段來啟用基於型樣的掃描。基於型樣的掃描是以自訂搜尋準則為基礎的原始碼分析。

型樣規則集和型樣規則

使用這些區段來新增型樣分析期間要使用的規則和規則集。如需相關資訊，請參閱第 209 頁的『利用基於型樣的規則自訂』和第 96 頁的『管理掃描配置』。

「內容」視圖：選取的應用程式

在這個視圖中，您可以配置所選應用程式的屬性。應用程式屬性相依於先前建立的廣域屬性。

- 『概觀』
- 『排除和過濾』
- 『規則和規則集』
- 第 219 頁的『已修改的發現項目』
- 第 219 頁的『自訂發現項目』

概觀

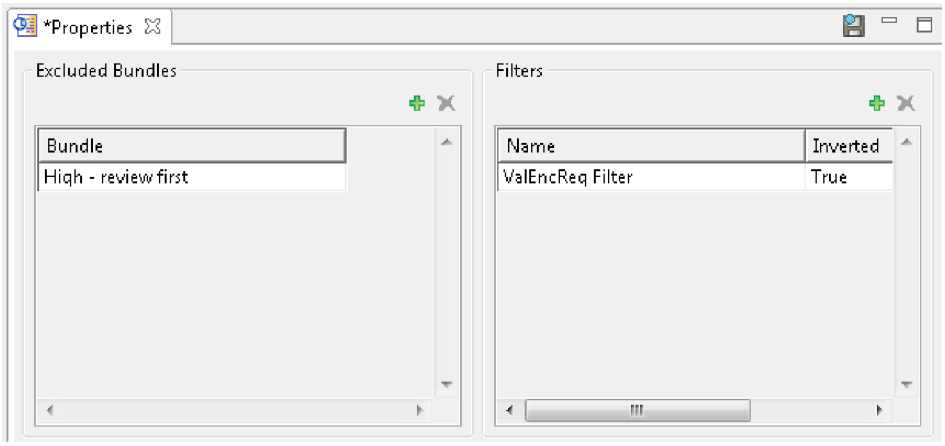
「概觀」標籤會顯示：

- 應用程式名稱。在欄位中輸入新名稱，可以將應用程式重新命名。
- 應用程式屬性

排除和過濾

此標籤可讓您指定所選取應用程式的現有過濾器，及如何套用過濾器（可直接套用過濾器 - 或套用它的反轉）。在此標籤中，您也可以管理排除掃描結果的組合。如需過濾器的相關資訊，請參閱第 121 頁的第 5 章, 『分類及分析』 - 如需全域套用它們的詳細資料，請參閱第 141 頁的『全域套用過濾器』。

已排除及過濾掉的發現項目不會出現在掃描結果中，也不會計入應用程式或專案的度量中。



規則和規則集

當在「瀏覽器」視圖中選取應用程式時，「內容」視圖中的「型樣規則和規則集」標籤可讓您新增在掃描該應用程式時將套用的型樣規則和規則集。您可以利用基於型樣的掃描，來搜尋要呈現為發現項目的文字型樣。個別規則和規則集可同時套用至應用

程式和專案。請參閱第 209 頁的『利用基於型樣的規則自訂』，以瞭解基於型樣的分析，並參閱第 215 頁的『套用型樣規則和規則集』，以瞭解如何在「內容」視圖中套用規則和規則集。

已修改的發現項目

在「已修改的發現項目」標籤中，您可以檢視、編輯或刪除任何先前修改過的發現項目，或修改現有的發現項目。已修改的發現項目是漏洞類型、嚴重性、分類有了改變，或擁有附註的發現項目。

自訂發現項目

在「自訂發現項目」標籤中，您可以檢視、新增、編輯或刪除自訂發現項目。請參閱第 154 頁的『自訂發現項目』，以取得詳細資料。

「內容」視圖：選取的專案

在「內容」視圖的這個模式之下，您可以配置所選專案的參數。專案屬性相依於先前建立的廣域屬性。內容會隨著所選的專案而不同。

- 『「選取的專案概觀」標籤』
- 第 220 頁的『過濾器 』
- 第 220 頁的『型樣規則和規則集』
- 第 221 頁的『副檔名』
- 第 221 頁的『來源』
- 第 222 頁的『JavaServer Pages (JSP) 專案相依關係』
- 第 222 頁的『專案相依關係』
- 第 222 頁的『編譯』
- 第 223 頁的『最佳化』
- 第 223 頁的『前置編譯標籤（只限 ASP.NET）』

「選取的專案概觀」標籤

「概觀」標籤會顯示：

- 專案**名稱**。在欄位中輸入新名稱，可以將專案重新命名。
- **專案檔**名稱和路徑
- **專案類型**
- 配置：這個區段顯示目標配置。如果是 .NET 和 C++ 專案，這個區段會顯示儲存在「專案相依關係」標籤中的目標配置。如果是所有其他專案類型，這個區段會顯示**預設值**。
- 過濾選項：請選取**過濾外部來源所包含的發現項目**，來濾除在所掃描專案之原始檔以外的檔案中發現的任何發現項目。這個選項會減少在編譯器產生的檔案或暫存檔中報告發現項目之專案的雜訊，例如 ASP.NET。
- 漏洞分析快取選項：如果您要反覆掃描及新增自訂規則，然後在不變更原始碼的情況下重新掃描，以精簡您的程式碼庫評量，您可以設定專案內容來使用漏洞分析快取，掃描的時間會縮短許多。如果要執行這個動作，請在專案內容中，選取**啟用漏**

洞分析快取勾選框。選取這個勾選框之後，當您第一次掃描項目時，會建立一項漏洞分析快取。以後每一次掃描這個專案，都會使用這個漏洞分析快取，掃描時間會減少。

如果要清除漏洞分析快取，以及啟用 Java 漸進式分析時建立的快取，請按一下**清除快取**。下一次掃描專案時，將會進行完整掃描，並建立新的漏洞分析快取。在下列情況下，您可以清除快取：

- 前次掃描之後，專案中的原始碼又有了改變。
- 您已進行專案配置變更，例如，新增或刪除原始檔。
- 您已變更程式碼配置選項。例如，如果您要掃描 Java，且類路徑已經改變，或您要掃描 C 或 C ++，且已變更了 `include` 路徑或前置處理器定義，您可能會想清除快取。
- 您已啟用 Java 漸進式分析而且想要執行完整掃描，或是遇到問題，可以透過清除快取來補救。如需相關資訊，請參閱 第 103 頁的『Java 的漸進式分析』。

註：您也可以在利用「自訂規則精靈」來建立自訂規則時，選取**清除快取**勾選框來清除漏洞分析快取。

- **字串分析**：字串分析會監視 Java 或 Microsoft .NET 專案中的字串操作。它能夠自動偵測消毒器和驗證器常式。當進行這項偵測時，可以減少誤判 (false positive) 及假性無侵害攻擊 (false negative) 的情況。選取**啟用字串分析來尋找驗證器/消毒器功能**勾選框來啟用字串分析。**將匯入的規則套用於廣域範圍**勾選框會決定所發現的消毒器或驗證器常式，是否應該套用到單一專案或廣域層次（套用至所有專案）。

  註：套用字串分析，掃描可能會變慢。因此，建議只應在程式碼變更之後才套用，而且應於後續掃描中停用。另外，還應該將發現的常式視為建議，由審核員來進行檢查。您可以在「自訂規則」視圖中檢視這些常式。
- **檔案編碼**：您必須設定專案中各檔案的字元編碼，AppScan Source 才能適當讀取檔案並且（舉例來說）在程式碼視圖中正確顯示它們。

  註：AppScan Source 專案的預設檔案編碼是 **ISO-8859-1**。您可以在「一般」喜好設定頁面中，變更預設檔案編碼。

過濾器

此標籤可讓您指定所選取專案的現有過濾器，及如何套用過濾器（可直接套用過濾器 - 或套用它的反轉）。如需過濾器的相關資訊，請參閱第 121 頁的第 5 章, 『分類及分析』 - 如需全域套用它們的詳細資料，請參閱第 141 頁的『全域套用過濾器』。

型樣規則和規則集

當在「瀏覽器」視圖中選取專案時，「內容」視圖中的「型樣規則和規則集」標籤可讓您新增在掃描該專案時將套用的型樣規則和規則集。您可以利用基於型樣的掃描，來搜尋要呈現為發現項目的文字型樣。個別規則和規則集可同時套用至應用程式和專案。請參閱第 209 頁的『利用基於型樣的規則自訂』，以瞭解基於型樣的分析，並參閱第 215 頁的『套用型樣規則和規則集』，以瞭解如何在「內容」視圖中套用規則和規則集。

副檔名

使用這個標籤，來配置或新增專案的有效副檔名 - 並且將檔案從掃描中排除，以及將副檔名指定為 Web 檔。

副檔名區段列出現行專案類型的第 90 頁的『專案副檔名』喜好設定頁面中已廣域設定的副檔名（您可以使用**副檔名設定**功能表，為不同的專案類型選擇副檔名）。如果要將某個副檔名從現行專案的掃描中排除，請在清單中選取它，並按一下**排除副檔名**。這會將該副檔名列在標籤的**排除的副檔名**區段中。

如果要為專案新增其他副檔名，請在**其他副檔名**區段中選取**新增副檔名**，然後輸入副檔名，並指出是否應掃描使用該副檔名的檔案、將它們視為 Web 檔或加以排除。

表 22. 副檔名設定

設定	說明	用法範例
掃描或**評量**	將使用所指副檔名的檔案包含在完整分析中。	• 如果為 Java 專案建立 .xxx 副檔名，並標示為**掃描**或**評量**，則會編譯和掃描使用該副檔名的檔案。 • 如果不應編譯和掃描檔案，檔案可以是專案的一部分，但不要標示為**掃描**或**評量**（例如 C++ 標頭檔）。這些檔案會包含在專案中，並在執行基於型樣的分析期間進行搜尋。
Web 檔	標示使用所指副檔名的檔案，以進行 JSP 編譯。這項設定容許 AppScan Source 將 Web 原始檔與非 Web 原始檔加以區隔。	如果為 Java 專案建立 .yyy 副檔名，並標示為 **Web 檔**，則會將使用該副檔名的檔案安排成專案中的 Web 原始檔。當 AppScan Source 準備分析時，會將這些檔案前置編譯成要分析的類別。
排除	不在專案中為使用所指副檔名的檔案建立原始檔。將不會掃描使用這個副檔名的檔案。	為您專案所需的檔案建立 .zzz 副檔名，以進行編譯，但不需要包含在分析中。

來源

請指定要併入掃描的來源。

- 工作目錄：AppScan Source 專案檔 (ppf) 的位置，且是所有相對路徑的基本目錄。
- **新增原始碼根目錄**和**移除原始碼根目錄**：「來源」標籤會顯示從「專案配置」精靈為專案建立的內容，或在匯入的 ppf 中所定義的內容。

  只有在選取了**原始碼根目錄**圖示時，才有**移除原始碼根目錄**可用。它用來移除原始碼根目錄。
- 尋找原始碼根目錄（只限 Java 專案）：可讓 AppScan Source for Analysis 自動尋找所有有效的原始碼根目錄。

- **原始碼根目錄**圖示下會列出專案檔。遭排除在掃描之外的檔案有紅色檔案圖示（如果用滑鼠右鍵按一下遭排除的檔案，則其功能表已停用**排除**且已啟用**併入**）。如果要排除已併入的檔案，請用滑鼠右鍵按一下它，然後在功能表中選擇**排除**。如果要併入已排除的檔案，請用滑鼠右鍵按一下它，然後在功能表中選擇**併入**。

JavaServer Pages (JSP) 專案相依關係

「JSP 專案相依關係」標籤會顯示針對指定的 JSP 專案所建立的內容。

- 包含 Web (JSP) 內容：識別專案是否為包含 JavaServer Pages 的 Web 應用程式。
- Web 環境定義根目錄：WAR 檔或包含 WEB-INF 目錄的目錄。 Web 環境定義根目錄必須是有效 Web 應用程式的根目錄。
- JSP 編譯器：既有的 Tomcat 7 是預設 JSP 編譯器設定（您可以在 Java 和 JSP 喜好設定頁面中變更預設 JSP 編譯器）。如果要瞭解 AppScan Source 支援的編譯器，請參閱 http://www.ibm.com/support/docview.wss?uid=swg27027486。

  AppScan Source 的安裝架構包含 Apache Tomcat 第 7 版和第 8 版。如果未配置 **Tomcat 7** 和 **Tomcat 8** 喜好設定頁面，AppScan Source 會利用提供的 Tomcat JSP 編譯器（目前標示為預設編譯器）來編譯 JSP 檔。如果您想要使用外部支援的 Tomcat 編譯器，請利用 Tomcat喜好設定頁面來指向您的本端 Tomcat 安裝架構。

  如果您使用 Oracle WebLogic 伺服器 或 WebSphere Application Server，您必須配置適用的喜好設定頁面來指向應用程式伺服器的本端安裝架構，以便在分析期間用來編譯 JSP。如果您尚未完成此配置，當您選取 JSP 編譯器時，會出現訊息來提示您這麼做。如果您在訊息中按一下**是**，就會看到適當的喜好設定頁面。如果您按一下**否**，JSP 編譯器選項旁邊會顯示警告鏈結（遵循此鏈結會開啟喜好設定頁面）。

專案相依關係

「專案相依關係」標籤會顯示專案內容。這個標籤的配置設定會隨著語言而改變，例如：

- **選項**可讓您選取任何其他必要的編譯器參數。
- JDK 設定專用於 Java。
- 前置處理器定義專用於 C/C++ 程式碼。當您指定前置處理器定義時，請勿併入編譯器的 -D 選項（例如：請指定 a=definition1 而非 -Da=definition1）。當您指定多個定義時，請使用以分號區隔的清單。
- 目標配置只適用於 .NET 和 C++ 專案。

編譯

- 選項：專案配置所需要的其他編譯器參數。
- 使用 JDK：依照「喜好設定」所配置，識別用來編譯專案的 JDK。請參閱第 79 頁的第 3 章, 『喜好設定』。

  Java 專案可以參照本端 Java Development Kit (JDK) 位置。當專案移到伺服器時，JDK 路徑可能不再有效。如果要將本端專案傳送到伺服器，您必須指出每個指定了具名 JDK 的專案之預設 JDK 路徑。

註：JSP 專案既有的預設編譯器是 Tomcat 7，它需要 Java 1.6 版或更高版本。如果保留 **Tomcat 7** 作為預設值，則使用較舊的 JDK 會導致在掃描期間發生編譯錯誤。

- 驗證：**驗證**可確保已正確配置專案相依關係。它會檢查 Java 專案在來源和類別路徑之間的配置衝突，它也會檢查編譯錯誤。如果在原始碼根目錄中，類別路徑中的類別重複，就會**發生衝突**。（如果發生衝突，請修改類別路徑來移除衝突的類別。）

  檢查衝突之後，**驗證**會判斷專案是否進行編譯，且會報告任何編譯錯誤。

最佳化

- **經過前置編譯的類別**：使用經過前置編譯的 Java 或 JSP 類別檔，而不在掃描期間編譯。選取之後，這個選項會停用來源暫置選項。
- 暫置原始檔，使編譯錯誤的影響降到最低：控制 AppScan Source 是否將來源複製到暫置目錄。

  修正不符合目錄的套件需要「Java 編譯」來開啟每個原始檔。

  清除掃描之間的暫置區會增加掃描之間的效能。

前置編譯標籤（只限 ASP.NET）

前置編譯是藉由向網站中的特殊頁面（依預設，是 precompile.axd）發出 HTTP 要求來完成。這個頁面由 web.config 中所指定的特殊 HTTP 處理程式來處理。這個處理程式會將整個網站（包括 client.aspx 檔）編譯到 .NET 架構目錄之下的 Temporary ASP.NET Files 目錄中，它們接著會全部在這裡進行掃描。

如果要掃描 ASP.NET 1.1，您必須設置網站來編譯及建置除錯資訊。接著，網站會編譯及建置除錯資訊，這種情況本身就是一個安全漏洞。您可以放心忽略這個漏洞，因為掃描需要它。不過，請務必不要在 web.config 中，以 debug=true 來編譯已部署的應用程式。

如果要前置編譯 ASP.NET 1.1 網站，請在 web.config 檔中，新增這個元素作為 <system.web> 元素的子項：

```
  <httpHandlers><add verb="*" path="precompile.axd"
type="System.Web.Handlers.BatchHandler"/></httpHandlers>
```

您也應該在編譯元素中設定 debug=true。例如：

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <system.web>
    <httpHandlers><add verb="*" path="precompile.axd"
    type="System.Web.Handlers.BatchHandler"/>
    </httpHandlers>
      <compilation
        defaultLanguage="c#"
        debug="true"
   />
...
```

這個元素向網站指定由特殊的 .Net System.Web.Handlers.BatchHandler 類別來處理 precompile.axd 頁面。這個類別會將網站內容前置編譯到 Temporary ASP.NET Files 目錄中。

- 網站：網站前置編譯的要求目標。預設位置是 precompile.axd。precompile.axd 是一個虛擬檔案，對映至 web.config 檔中所指定的檔案。
- 輸出目錄：前置編譯的目標目錄。AppScan Source 會在這個目錄尋找前置編譯的輸出。
- 前置編譯 ASP.NET 網站：在掃描期間，AppScan Source 會自動前置編譯及掃描經過前置編譯的輸出。
- 如果前置編譯失敗，就停止掃描：選取**前置編譯 ASP.NET 網站**和**如果前置編譯失敗，就停止掃描**，以便在前置編譯失敗時停止掃描。否則，只會在網站的主要輸出上繼續進行掃描。
- 立即編譯：在掃描之前進行測試，看看基於現行設定的前置編譯是否成功。編譯輸出會顯示在「前置編譯輸出」窗格中。
- 其他組合：對於任何 .NET 專案類型，請指定其他要掃描的組合。
- 專案參照：列出要在其中搜尋 .NET 組譯碼專案及現有的 .NET 專案中所參照之組合的目錄。

第 11 章 延伸應用程式伺服器匯入架構

AppScan Source 可讓您從 Apache Tomcat 及 WebSphere Application Server Liberty 設定檔匯入 Java 應用程式。您可以延伸應用程式伺服器匯入架構，以便從其他應用程式伺服器匯入 Java 應用程式，如本主題所述。

關於這項作業

應用程式伺服器匯入架構所隨附的 API 文件沒有提供 PDF 格式。如果您透過 Adobe PDF 存取此說明主題，則只有利用下列方式才能存取 API 文件：啟動 AppScan Source for Analysis 線上說明，並導覽至**延伸產品功能** > **延伸應用程式伺服器匯入架構** > **應用程式伺服器匯入延伸 API 類別及方法** - 或在 http://www.ibm.com/support/knowledgecenter/SSS9LM/welcome上找出該說明小節。

如果要延伸應用程式伺服器匯入架構，請完成下列步驟。這些步驟將引導您完成下列工作：

- 配置 Eclipse 整合開發環境
- 在 Eclipse 中建立新的外掛程式
- 在新建立的外掛程式中設定必要的相依關係
- 在外掛程式中定義應用程式伺服器的延伸規格
- 測試外掛程式
- 對 AppScan Source for Analysis 啟用外掛程式

程序

1. 針對 AppScan Source 應用程式伺服器匯入架構所需的相依關係，配置 Eclipse 整合開發環境：
   a. 在 Eclipse 中，從主功能表選取**視窗** > **喜好設定**。
   b. 在「喜好設定」對話框中，展開**外掛程式開發**，然後選取**目標平台**。
   c. 在「目標平台」喜好設定頁面中，按一下**新增**，建立新的目標定義。
   d. 在「目標定義」精靈頁面中，選取**無：從空的目標定義開始**，然後按**下一步**。
   e. 在「目標內容」精靈頁面的**名稱**欄位中輸入目標的名稱，然後按一下**新增**，新增您的 AppScan Source 安裝目錄（請參閱第 282 頁的『安裝和使用者資料檔位置』）。
   f. 選擇性的： 選取**顯示位置內容**，以驗證外掛程式可用。
   g. 按一下**完成**。
   h. 在「目標平台」喜好設定頁面中，選取您剛建立的目標平台，再按**套用**。然後按**確定**。
2. 在 Eclipse 中建立新的外掛程式：
   a. 從主功能表中，選取**檔案** > **新建專案**，以開啟「新建專案」精靈。
   b. 在「選取精靈」頁面中，選取**外掛程式專案**，然後按**下一步**。
   c. 在「外掛程式專案」頁面的**專案名稱**欄位中輸入外掛程式的名稱（此說明主題使用 `com.example.appserverimporter` 作為範例），然後按**下一步**。

d. 在「內容」頁面中，取消選取**產生啟動器，這是一個控制外掛程式生命週期的 Java 類別**，然後按**完成**。

3. 在您剛建立的外掛程式中，設定必要的相依關係：
   a. 開啟 META-INF\MANIFEST.MF，然後選取**相依關係**標籤。
   b. 在編輯器的**必要的外掛程式**區段中，執行下列動作：
      - 按一下**新增**，然後新增 com.ouncelabs.core.appserverimporter 及 org.eclipse.core.runtime。
      - 選取剛新增的 com.ouncelabs.core.appserverimporter 外掛程式，然後按一下**內容**。在外掛程式內容中，移除**最低版本**和**最高版本**欄位中的所有項目，然後按一下**確定**。
      - 對 org.eclipse.core.runtime 外掛程式重複上述步驟。
   c. 從主功能表中，選取**檔案** > **儲存**，以儲存對編輯器所做的所有變更。
   d. 下一步是定義應用程式伺服器的延伸規格。在該步驟中，您將繼續使用 META-INF\MANIFEST.MF 編輯器。
4. 遵循下列步驟，定義應用程式伺服器的匯入器延伸規格：
   a. 選取**延伸規格**標籤，然後按一下**新增**來新增 com.ouncelabs.appserver - 然後從主功能表中，選取**檔案** > **儲存**。
   b. 選取 **plugin.xml** 標籤。內容應該類似如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<?eclipse version="3.4"?>
<plugin>
  <extension
    point="com.ouncelabs.appserver">
  </extension>
</plugin>
```

   編輯此檔案來完成延伸規格定義。例如：

```
<?xml version="1.0" encoding="UTF-8"?>
<?eclipse version="3.4"?>
<plugin>
  <extension
    point="com.ouncelabs.appserver">
    <importer
     class="com.example.appserverimporter.MyAppServerImporter"
     id="com.example.appserverimporter.myappserver"
     name="My App Server">
   </importer> </extension>
</plugin>
```

   c. 從主功能表中，選取**檔案** > **儲存**，儲存對 **plugin.xml** 所做的變更。
5. 建立匯入器類別（此範例中為 com.example.appserverimporter.MyAppServerImporter），以定義新應用程式伺服器匯入器的行為。此類別必須延伸 BaseAppServerImporter（架構對於 AppServerImporter 介面的基本實作）。在此類別中：
   a. 實作 AppServerImporter.importAppServer(String)。這由架構用來決定要匯入的 Java EE 專案及其所在位置。一般而言，每一個專案只需要 Java EE 專案的名稱及路徑。如果建立 EAR 專案，則在 AppScan Source 使用者介面中選取專案時，包含的 Java EE 專案會隱藏。在此情況下，將匯入整個 EAR。否則會列出所有專案供個別選取。

      強烈建議盡可能使用這些方法：

- BaseAppServerImporter.processDropInsFolder(AppServerProfile, File)
- BaseAppServerImporter.processEARFile(AppServerProfile, File)

b. 實作 AppServerImporter.isValidLocation(String)。這用來偵測伺服器的類型（根據安裝目錄）。

c. 選擇性的： 置換 BaseAppServerImporter.getJSPCompilerType()。此方法會傳回要用於 AppScan Source 專案的 JSP 編譯器。如果未這樣做，則基本實作會傳回空值，且會使用產品的預設 JSP 編譯器。

6. 選擇性的： 在進階選項上，您可以自訂 JSP 編譯來使用經過前置編譯的 JSP 編譯器（將於匯入之前或期間進行 JSP 編譯）：

a. 置換 BaseAppServerImporter.getJSPCompilerType() 以傳回 JSPCompilerType.PRECOMPILED。

b. 置換 BaseAppServerImporter.getJSPCompilerType()，以呼叫 JMX、Java API、外部 Script 來編譯 JSP 檔 - 或直接將類別檔複製到 AppScan Source 專案的暫置目錄。使用 Application.getStagingDirectory(Project) 來取得暫置目錄。

c. 置換 BaseAppServerImporter.createJSPCompilerSupport() 來傳回 JSPCompilerSupport 的自訂延伸規格。這用於持續保存 JSP 檔與產生的類別檔之間的對映 - 以及在 JSP 編譯之後驗證。

d. 置換 BaseAppServerImporter.createClasspathProvider() 來傳回 AppServerClasspathProvider 的自訂實作。編譯任何在伺服器程式庫上具有相依關係的 Java 或 JSP 檔時需要此類別。此類別必須延伸 BaseAppServerClasspathProvider。請注意，呼叫 getClasspathEntries() 時，BaseAppServerClasspathProvider.installDirectory 將已設為應用程式伺服器的安裝目錄。

7. 遵循下列步驟來測試外掛程式：

a. 從主功能表中，選取**執行** > **執行配置**（或者，如果您要在除錯模式下測試，則選取**執行** > **除錯**）。

b. 建立新的 **Eclipse 應用程式**配置。

- 移至新配置的**主要**標籤。在**執行的程式**區段中，選取**執行產品**，然後設為執行 **com.ouncelabs.osa.rcp.product**。
- 移至**引數**標籤。在**工作目錄**區段中，選取**其他**，然後在欄位中輸入 AppScan Source 資料目錄（請參閱第 282 頁的『安裝和使用者資料檔位置』）。
- 在**外掛程式**標籤中，將**啟動工具**選項設為**僅以下選取的外掛程式**。展開**工作區**，確定已選取您建立的外掛程式 - 然後在**目標平台**下取消選取這些外掛程式：
  - com.ouncelabs.plugin.base
  - com.ouncelabs.plugin.base
  - com.ouncelabs.plugin.base.nl
  - com.ouncelabs.plugin.base.nl
  - com.ouncelabs.plugin.enhanced
  - com.ouncelabs.plugin.enhanced
  - com.ouncelabs.plugin.enhanced.nl
  - com.ouncelabs.plugin.enhanced.nl

c. 在「執行配置」對話框中按一下**執行**之前，請移至 AppScan Source 安裝目錄並執行 bin\OunceScanner.exe。

d. 回到「執行配置」對話框，按一下**執行**來啟動 AppScan Source for Analysis 並測試外掛程式。

8. 遵循下列步驟，對 AppScan Source for Analysis 啟用外掛程式：

a. 用滑鼠右鍵按一下專案並選取**匯出**。

b. 在「匯出」精靈的「選取」頁面中，展開**外掛程式開發**，選取**可部署的外掛程式及片段**，然後按**下一步**。

c. 在「可部署的外掛程式及片段」頁面中：

- 移至**目的地**標籤，然後瀏覽至機器上的暫存目錄來設定**目錄**。
- 移至**選項**標籤，選取**將外掛程式包裝為個別的 JAR 保存檔**及**限定元取代**。
- 按一下**完成**。

d. 找出作為外掛程式匯出目的地的暫存目錄，然後開啟其 plugins\ 資料夾。在此資料夾中，找出已建立的 .jar 檔並複製到 <install_dir>\dropins（其中 <install_dir> 是 AppScan Source 安裝的位置）。

註：

- 如果 \dropins 目錄不存在，則需要手動建立。
- 變更 AppScan Source 安裝目錄可能需要管理專用權。

e. 找出 <install_dir>\configuration\org.eclipse.equinox.simpleconfigurator\bundles.info。備份此檔案 - 然後編輯檔案，在尾端新增下列程式碼：

```
<my_plugin>,<my_plugin_version>,
dropins/<my_plugin>_<my_plugin_version>.jar,4,false
```

其中：

- <my_plugin> 是您建立的外掛程式的名稱。
- <my_plugin_version> 是您建立的外掛程式的版本號碼。

註：在此項目的開頭，<my_plugin>、<my_plugin_version> 及 dropins/ 位置以逗點 (,) 區隔。

f. 啟動 AppScan Source for Analysis。

g. 從主功能表中，選取**說明** > **關於 AppScan Source for Analysis**，然後按一下**安裝詳細資料**。選取**外掛程式**標籤，並確定其中已列出您的外掛程式。

h. 關閉「安裝詳細資料」對話框，開始使用您的應用程式伺服器匯入架構。

第 12 章 AppScan Source for Analysis 範例

AppScan Source for Analysis 包含多個範例應用程式，您可利用它們讓自己熟悉該產品。

安裝 AppScan Source for Analysis 之後，範例應用程式會位於 <data_dir>\samples（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述）。

範例 Java 應用程式：*simpleIOT*

simpleIOT 範例是小型 Java 應用程式，包含各種安全漏洞。它可以手動匯入到 AppScan Source for Analysis 工作台 - 或者，您可以匯入此範例隨附的應用程式檔 (SimpleIOT.paf) 或專案檔 (SimpleIOT.ppf)。如果要瞭解如何新增應用程式和專案，請參閱第 29 頁的第 2 章, 『配置應用程式和專案』。

將範例新增至 AppScan Source之後，您可以掃描它及瀏覽其發現項目。

學習 Framework for Frameworks 處理 API 的範例應用程式：*F4FEjbExample.zip*

此範例專案保存檔用來示範 Framework for Frameworks 處理 API。如需相關資訊，請參閱《*IBM Security AppScan Source Utilities* 使用手冊》。

第 13 章 AppScan Source for Analysis 工作環境

為了充分運用 AppScan Source，您應該瞭解 AppScan Source for Analysis 工作環境背後的基本概念，以及如何使用最適合您的工作流程的選項。

AppScan Source for Analysis 工作台

AppScan Source for Analysis 工作流程是在工作台中進行，工作台由視景、視圖和編輯器所組成，它們會隨著環境定義的不同而顯示或隱藏。

視景

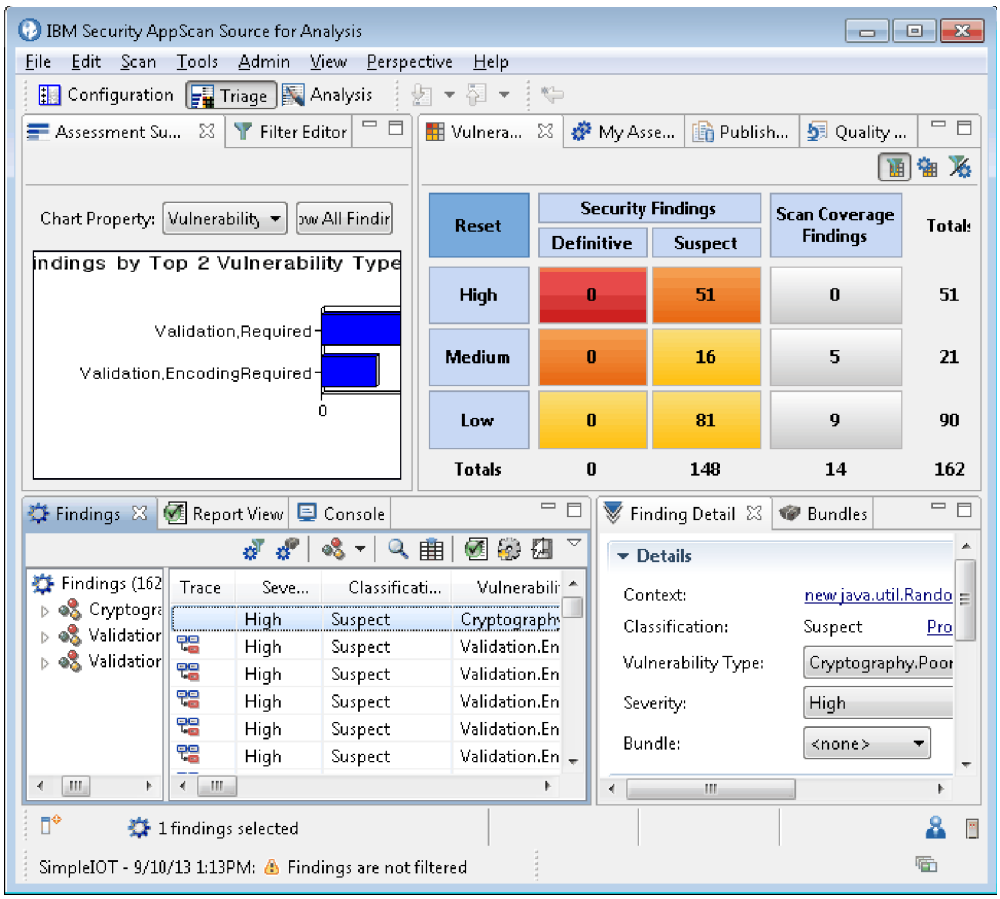
產品中的三個視景（「配置」、「分類」和「分析」）是由多個視圖組成。雖然每個視景在開啟時都會有預設的視圖，但您可以重組視圖來自訂每個視景。在說明的第 243 頁的第 14 章, 『視圖』區段中，會詳細說明這些視圖。

- 「配置」視景：建立及管理應用程式、專案和屬性。
- 「分類」視景：檢視掃描結果，將補救工作流程排列優先順序，並將實際漏洞與潛在漏洞分開。這個視景可用來釐清需要先修正的問題。
- 「分析」視景：往下探查到個別發現項目，檢閱原始碼、補救建議和 AppScan Source 追蹤資訊

工作台視窗

AppScan Source for Analysis 工作台視窗由下列元素組成：

- 主功能表：存取 AppScan Source for Analysis 功能的功能表
- 工具列：常用功能的圖示和按鈕
- 視景：視圖的集合
- 視圖：工作台中資訊的呈現以及導覽的方式



工作台底端的工具列和資訊

- **快速視圖**工具列：快速視圖是可快速開啟及關閉的隱藏視圖。它們和其他視圖的運作方式類似，只不過它們不佔用工作台視窗的空間。快速視圖是由快速視圖列上的工具列按鈕來代表，此工具列就是在工作台視窗左下角的工具列。當您按一下工具列按鈕來顯示快速視圖時，該視圖會暫時在現行視景中開啟（重疊在它上面）。一旦按一下該視圖之外，或視圖不再是焦點時，就會重新隱藏它。如果要將某視圖設為快速視圖，請按一下**將視圖顯示為快速視圖**，然後從功能表中選擇該視圖。
- 選取的發現項目：選取發現項目之後，工作台底端的指示器會顯示所選取的發現項目數。
- 原始檔資訊：當原始檔開啟時，檔案的這項相關資訊會顯示在工作台底端：
  - 檔案為可寫入或唯讀。如果您試圖編輯唯讀檔案，AppScan Source for Analysis中的提示可讓您將檔案設為可寫入。
  - 作業系統輸入模式是插入或改寫。
  - 檔案中的現行游標位置（行號和欄號）。
- 伺服器連線資訊：將滑鼠指標移至使用者圖示指出使用者目前登入 AppScan Enterprise Server - 將滑鼠指標移至伺服器圖示可讓您查看 AppScan Source for Analysis所連接的 AppScan Enterprise Server。
- 開啟評量時，工作台底端會包含此資訊：
  - 評量的名稱，及它的建立日期和時間。

– 可讓您快速判斷過濾器如何套用至評量中的發現項目的指示器。請參閱 第 141 頁的『判斷已套用的過濾器』，以取得詳細資訊。

- 進度指示器也會顯示在工作台底端，以顯示動作的進度。例如，在掃描和評量發佈期間，此指示器會出現。此外，此區段指出何時開啟評量。

主功能表

主功能表列包含可讓您執行各種動作的功能表。您的使用者專用權會控制這些功能表中可供您使用的指令。

- 『「檔案」功能表』
- 第 236 頁的『「編輯」功能表』
- 第 237 頁的『「掃描」功能表』
- 第 238 頁的『「工具」功能表』
- 第 238 頁的『「管理」功能表』
- 第 239 頁的『「檢視」功能表』
- 第 239 頁的『「視景」功能表』
- 第 239 頁的『「說明」功能表』

「檔案」功能表

檔案功能表提供應用程式、專案和評量的選項，且可讓您結束產品。有些**檔案**功能表項目與環境定義相關，會隨著作用中的視圖及該視圖中目前選取的項目而不同。

表 *23.* 「檔案」功能表

功能表項目	說明	鍵盤快速鍵
新增應用程式 > **建立新的應用程式**	新增應用程式到應用程式集中。這個動作會啟動「新建應用程式」精靈。	Ctrl+N
新增應用程式 > **開啟現有的應用程式**	這會啟動「開啟」對話框，供您瀏覽以找出現有的應用程式並將其新增到應用程式集中。可以新增的檔案或目錄類型包括 `.paf`、`.sln`、`.dsw` 和 `.ewf`。	Ctrl+O

表 23. 「檔案」功能表 (繼續)

功能表項目	說明	鍵盤快速鍵
新增應用程式 > 匯入現有的 Eclipse 型工作區	這會啟動「新增工作區」對話框，供您新增包含 Java 專案的現有 Eclipse 或 IBM Rational Application Developer for WebSphere 軟體 (RAD) 工作區。匯入工作區之後，您就可以掃描其中所包含的任何 Java 專案。 **註：**在匯入工作區之前，請確定您已依照第 41 頁的『配置 Eclipse 和 Rational Application Developer for WebSphere 軟體 (RAD) 專案的開發環境』所說明來安裝和更新開發環境。	
新增應用程式 > 從應用程式伺服器匯入	從 Apache Tomcat 或 WebSphere Application Server Liberty 應用程式伺服器匯入現有的 Java 應用程式。	
新增應用程式 > 多個應用程式	新增多個應用程式到應用程式集。這個動作會啟動一個對話框，供您指定要在其中搜尋應用程式的目錄。在搜尋結果中，您可以選取一或多個要新增的應用程式。	
新增應用程式 > 探索應用程式	這會啟動「應用程式探索助理」，供您快速建立及配置 Java 和 Microsoft Visual Studio 原始碼的應用程式和專案。	
移除應用程式	如果在「瀏覽器」視圖中選取應用程式，會啟用這個動作，選擇這個動作會移除所選的應用程式。	
新增專案 > 新建專案	如果在「瀏覽器」視圖中選取應用程式，會啟用這個動作，選擇這個動作可讓您將新專案加到應用程式中。這個動作會啟動「新建專案精靈」。	
新增專案 > 現有的專案	如果在「瀏覽器」視圖中選取應用程式，會啟用這個動作，選擇這個動作可讓您將現有的專案加到應用程式中。這個動作會啟動一個對話框，供您瀏覽要開啟的 .ppf、.vcproj、.vcxproj、.csproj、.vbproj、.dsp，或 .epf 檔。	

表 23. 「檔案」功能表 (繼續)

功能表項目	說明	鍵盤快速鍵
新增專案 > 複製專案	如果在「瀏覽器」視圖中選取專案，會啟用這個動作，選擇這個動作會開啟一個對話框，供您將專案複製到另一個應用程式中，或在目前包含專案的應用程式中建立專案的副本。	
新增專案 > 多個專案	新增多個專案到「瀏覽器」視圖所選的應用程式中。這個動作會啟動一個對話框，供您完成下列作業之一： • 指定要在其中搜尋專案的目錄。 • 指定要在其中搜尋專案的工作區。 • 指定要在其中搜尋專案的 Microsoft 解決方案檔案。 在搜尋結果中，您可以選取一或多個要新增的專案。	
登錄	向 AppScan Source 登錄所選的應用程式或專案。您必須先登錄應用程式和專案，然後它們才能發佈到 AppScan Source 資料庫。	
取消登錄	取消登錄所選的應用程式或專案。	
開啟評量	這會啟動「開啟」對話框，供您瀏覽到 AppScan Source 評量檔。可開啟的檔案類型包括 `.ozasmt` 和 `.xml`。	F7
關閉評量	關閉「分類」視景中目前開啟的評量。	
儲存評量	將開啟的評量儲存到檔案中。	Ctrl+Shift+S
另存評量	以不同的名稱來儲存評量、將它儲存在另一個目錄中，或兩者兼備。	
將評量發佈到 AppScan Source	將現行評量儲存在 AppScan Source 資料庫中。必須先登錄所掃描的應用程式（或是應用程式包含的專案或檔案），然後才能完成發佈動作。如果尚未登錄應用程式，則在選擇發佈動作時，系統將提示您登錄。	

表 23. 「檔案」功能表 (繼續)

功能表項目	說明	鍵盤快速鍵
將評量發佈到 AppScan Enterprise Console	如果 AppScan Enterprise Server 已安裝 Enterprise Console 選項，您可以將評量發佈到那裡。 您必須先在 AppScan Enterprise Console 喜好設定頁面中填入有效值，才能將評量發佈到 Enterprise Console。	
儲存	在下列情況之下，可以使用這個動作： • 在「內容」視圖中，修改了應用程式的內容。 • 在「內容」視圖中，修改了專案的內容。 • 已修改內部編輯器中所開啟的檔案。 請選取這個動作來儲存這些變更。	Ctrl+S
結束	結束 AppScan Source for Analysis。	

註：如果要瞭解 AppScan Source for Analysis、AppScan Source for Automation 和 AppScan Source 指令行介面 支援哪些版本的匯入檔案，請參閱http://www.ibm.com/support/docview.wss?uid=swg27027486。在此頁面上，選取您使用的 AppScan Source 版本的標籤 - 然後選取您使用的 AppScan Source 元件。如果 AppScan Source 支援開啟及掃描來自其他開發環境的檔案，該支援會列在**支援的軟體**標籤的**編譯器與語言**區段中。

「編輯」功能表

這個功能表提供標準的修改和搜尋/取代控制項。這個功能表也用來啟動產品喜好設定。有些**編輯**功能表項目與環境定義相關，會隨著作用中的視圖及該視圖中目前選取的項目而不同。

表 24. 「編輯」功能表

功能表項目	說明	鍵盤快速鍵
剪下	複製和移除所選的文字。請利用這個動作來處理在主控台、編輯器或各種文字欄位中選取的文字。	Ctrl+X
複製	將所選的文字複製到剪貼簿中。請利用這個動作來處理在主控台、編輯器或各種文字欄位中選取的文字。	Ctrl+C

表 *24.* 「編輯」功能表 *(*繼續*)*

功能表項目	說明	鍵盤快速鍵
貼上	貼上已複製或剪下的文字。這個動作通常用來複製資訊，以及將資訊重新產生在產品的另一部分中。	Ctrl+V
重新命名	重新命名所選的物件。可以重新命名的物件包括應用程式、專案、評量和組合。	F2
移除	移除所選的物件。	Delete
全選	選取文字整體。請利用這個動作來處理在主控台、編輯器或各種文字欄位中的文字。	Ctrl+A
重新整理	重新整理所選的應用程式、專案或視圖的內容。	F5
尋找	在主控台或編輯器中搜尋文字，或在發現項目表格中搜尋發現項目。	Ctrl+F
尋找下一個	如果在主控台或編輯器中利用尋找動作來搜尋文字，請利用這個動作來尋找文字的下一個實例。	F3
喜好設定	請選取這個選項來開啟「喜好設定」對話框。喜好設定是關於 AppScan Source for Analysis 的外觀與操作的個人選項。	

「掃描」功能表

從**掃描**功能表，您可以管理所選應用程式、專案或檔案的掃描。

表 *25.* 「掃描」功能表

功能表項目	說明	鍵盤快速鍵
掃描全部	掃描所有應用程式。將以預設掃描配置來執行掃描。	
掃描選項	掃描所選的應用程式、專案或檔案。將以預設掃描配置來執行掃描。	F4
重新掃描	重新掃描評量目標。這次掃描會使用前次用來掃描項目（或選取的項目）的掃描配置。	
取消掃描	終止掃描，不產生任何結果。	
停止掃描	中止掃描，產生局部結果。	

表 25. 「掃描」功能表 (繼續)

功能表項目	說明	鍵盤快速鍵
建置配置	配置用來定義專案建置參數，例如，前置處理器定義或併入路徑。一般而言，匯入的專案會有名稱為**版本**或**除錯**的配置。 如果這個功能表項目不適用，則會停用。	

「工具」功能表

這個功能表包括比較評量和產生報告的選項，以及在編輯器中檢閱檔案或發現項目的選項。有些**工具**功能表項目與環境定義相關，會隨著作用中的視圖及該視圖中目前選取的項目而不同。

表 26. 「工具」功能表

功能表項目	說明
差異評量	這個動作會開啟一個對話框，供您選取兩個評量來進行比較。
產生發現項目報告	產生所選發現項目或組合內容的報告。當發出這個動作時，您必須選取一個發現項目或組合視圖。如果未在視圖中選取發現項目，報告會包含視圖中的所有發現項目。
產生報告	產生一份報告，供您檢視所有基於特定相符性需求或準則的發現項目。
在內部編輯器中開啟	在內部 AppScan Source for Analysis 編輯器中，開啟一個檔案。這個動作可用於所選的發現項目，且會在編輯器中開啟與發現項目相關聯的檔案。
在外部編輯器中開啟	利用外部編輯器來開啟檔案。這個動作可用於所選的發現項目，且會在編輯器中開啟與發現項目相關聯的檔案。

「管理」功能表

管理功能表提供可用來管理使用者及啟動審核資訊的動作。

表 27. 「管理」功能表

功能表項目	說明
管理使用者	這個動作會啟動一個對話框，供您建立及編輯使用者和許可權。 您必須具備 AppScan Source 管理許可權，才能管理使用者。
審核	這個動作會啟動一個視圖，供您查看鑑別事件之類的審核資訊。

請參閱《*IBM Security AppScan Source* 安裝與管理手冊》，以取得管理作業進一步的詳細資料。

「檢視」功能表

檢視功能表用來控制每個視圖的顯示，或選取開啟的視圖。

如果要進一步瞭解 AppScan Source for Analysis 中可用的視圖，請參閱 AppScan Source for Analysis 視圖。

「視景」功能表

視景功能表用來控制 AppScan Source for Analysis 視景的顯示，它們是預先配置的視圖和選項集合。

表 *28.* 「視景」功能表

功能表項目	說明	鍵盤快速鍵
配置	這個視景可讓您建立及管理應用程式、專案和屬性。	Alt+1
分類	這個視景可讓您檢視掃描結果，設定補救工作流程的優先順序，以及區分實際的漏洞和潛在漏洞。這個視景可用來釐清需要先修正的問題。	Alt+2
分析	這個視景可讓您往下探查進入個別發現項目，以及檢閱原始碼、補救建議，以及 AppScan Source 追蹤資訊。	Alt+3
重設視景	選取這個項目，會使目前顯示的視景回到其預設視圖和佈置。	

「說明」功能表

說明功能表包括能夠開啟各種工具來協助使用產品的動作。其中包括歡迎使用產品、線上使用者協助，以及 AppScan Source 安全知識庫。

表 *29.* 「說明」功能表

功能表項目	說明
歡迎使用	選取這個選項會開啟 AppScan Source for Analysis 的「歡迎使用」視圖。這個視圖提供了各種不同說明資源的快速鏈結，其中包括 X-Force RSS 資訊來源。
說明內容	選取這個選項會開啟 AppScan Source for Analysis 的產品使用者協助。
安全知識庫	這個動作會開啟 AppScan Source 安全知識庫。知識庫提供了每一漏洞的信息，它提供關於主要原因、風險嚴重性和可行的補救建議的詳盡說明。

表 29. 「說明」功能表 (繼續)

功能表項目	說明
日誌	選取這個選項會開啟「日誌」視圖。在這個視圖內，各標籤可讓您選取要顯示的日誌檔。
AboutIBM Security AppScan Source for Analysis	選取這個選項會開啟一個對話框來提供 AppScan Source for Analysis 的相關產品資訊。

工具列

AppScan Source for Analysis 工作台中的工具列提供指令的圖形捷徑。如果要識別特定工具列圖示，請將滑鼠暫停在圖示上，直到出現浮動說明。工具列按鈕代表常用的作業（也會出現在**主要**功能表中）。工具列作業會隨著環境定義而不同。

主要工具列提供 AppScan Source for Analysis 視景的快速鏈結。此外，大部分視圖都有工具列，可供快速啟動視圖相關的一般動作。

浮動說明

浮動說明是環境定義相關說明的一種形式，當滑鼠指標停在介面某元素上，浮動說明會出現在一個小型蹦現視窗中。蹦現視窗會顯示這個介面元素的簡要說明。

除了提供按鈕和圖示的浮動說明之外，AppScan Source for Analysis 也在各種位置提供浮動說明，例如：

- 在「瀏覽器」視圖中，浮動說明可用來指示應用程式、專案及檔案的檔名和路徑。另外，浮動說明也會指出應用程式或專案是否已登錄。
- 在「追蹤」視圖中，將滑鼠游標移到圖形中的追蹤節點上，會提供節點的相關資訊。
- 在「過濾器編輯器」視圖的**追蹤**區段中，將滑鼠游標移到追蹤項目上，會提供項目的詳細資料。
- 在「掃描配置」視圖的**進階設定**區段中，每個設定都有浮動說明。
- 將滑鼠游標移到「評量摘要」視圖的直條圖上，會提供直條所代表的確切發現項目數。
- 在工作台狀態列（位於工作台底端）中，將滑鼠游標移到使用者圖示上，會啟動浮動說明來識別登入的使用者。將滑鼠游標移到伺服器圖示上，會啟動浮動說明來指出 AppScan Source for Analysis 所連接的 Enterprise Server。

狀態列

狀態列位於工作台底端，它會顯示用來識別現行動作（例如掃描）的參考訊息。

例如，在掃描期間，狀態列可能會顯示 `Scanning <Project name>`，且會提供進度指示器。此外，也會顯示掃描的現行階段，例如，`Preparing for Vulnerability Analysis: 99%`。掃描完成之後，狀態列會顯示經歷時間。

另外，狀態列也包括現行使用者和伺服器連線的相關資訊。將滑鼠游標移到使用者圖示上，會啟動浮動說明來識別登入的使用者。將滑鼠游標移到伺服器圖示上，會啟動浮動說明來指出 AppScan Source for Analysis 所連接的 Enterprise Server。

第 14 章 視圖

AppScan Source for Analysis 工作環境由多個視景和視圖組成，其中含有不同的評量或掃描資料。

AppScan Source for Analysis 視圖提供發現項目的替代呈現方式（其中一部分支援編輯程式碼），它們可讓您在工作台中導覽資訊。例如，「瀏覽器」視圖會顯示應用程式、專案及其他資源。視圖可以單獨出現，也可以在標籤記事本中，與其他視圖堆疊起來。您可以變更視景版面，方法是開啟並關閉視圖，再將它們定位在工作台視窗中的其他位置。

下列各節會更詳細說明各個視圖：

- 『配置視圖』
- 第 261 頁的『協助掃描輸出的視圖』
- 第 264 頁的『協助分類的視圖』
- 第 273 頁的『可讓您調查單一發現項目的視圖』
- 第 277 頁的『可讓您處理評量的視圖』
- 第 280 頁的『「組合」視圖』

配置視圖

這一節的視圖是用於配置 AppScan Source。

- 『「自訂規則」視圖』
- 第 73 頁的『「瀏覽器」視圖』
- 第 248 頁的『「型樣規則庫」視圖』
- 第 248 頁的『「內容」視圖』
- 第 101 頁的『掃描配置視圖』
- 第 197 頁的『報告編輯器』

「自訂規則」視圖

在「自訂規則」視圖中，您可以利用「自訂規則精靈」來建立自訂規則。新增、檢視或刪除現有的規則。

請參閱第 203 頁的『建立自訂規則』，以取得詳細資料。

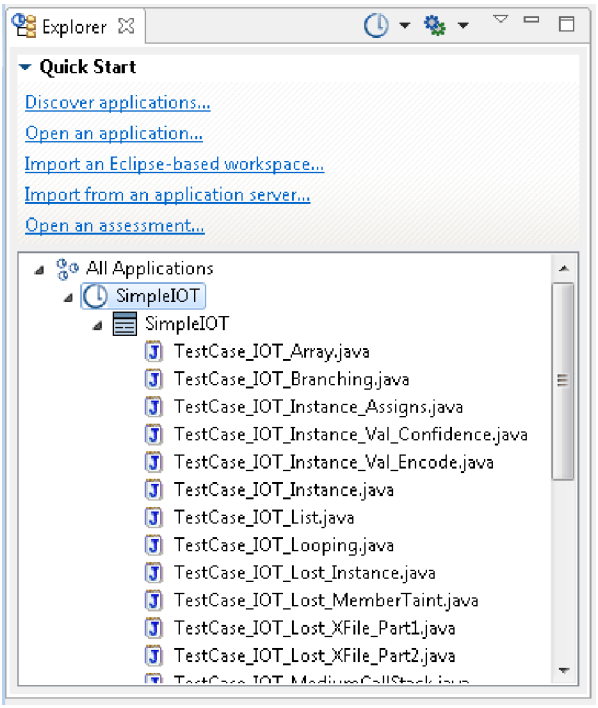
「瀏覽器」視圖

「瀏覽器」視圖的頂端有一個**快速入門**區段，底端則有一個瀏覽器區段，包含一個節點：**所有應用程式**。**快速入門**區段包含數個可啟動一般動作的實用鏈結。瀏覽器區段由一個樹狀結構窗格組成，提供資源的階層式視圖，包括：應用程式、專案、目錄和專案檔，而**所有應用程式**是它的根目錄。這些資源的導覽方式，很像檔案瀏覽器。當導覽視圖時，樹狀結構的選取狀態決定了「內容」視圖中所能使用的標籤。

- 第 74 頁的『一般資訊』
- 第 74 頁的『「快速入門」區段』

- 第 75 頁的『工具列按鈕』
- 第 75 頁的『右鍵功能表選項』
- 第 77 頁的『應用程式和專案指示器』

一般資訊



在「瀏覽器」視圖中，您可以使用工具列按鈕、**快速入門**區段中的鏈結，或瀏覽器區段中的右鍵功能表指令，以新增應用程式和專案，並掃描程式碼。新增應用程式之後，瀏覽器區段會提供應用程式和專案的視覺化指示器及各自的狀態。

提示：在「瀏覽器」視圖中，浮動說明可用來指示應用程式、專案及檔案的檔名和路徑。另外，浮動說明也會指出應用程式或專案是否已登錄。

「快速入門」區段

快速入門區段提供下列鏈結來啟動一般作業：

- **探索應用程式**：這會啟動「應用程式探索助理」，供您快速建立及配置 Java 和 Microsoft Visual Studio 原始碼的應用程式和專案。
- **開啟應用程式**：這會啟動「開啟」對話框，供您瀏覽以找出現有的應用程式並將其新增到應用程式集中。可以新增的檔案或目錄類型包括 `.paf`、`.sln`、`.dsw` 和 `.ewf`。
- **匯入 Eclipse 型工作區**：這會啟動「新增工作區」對話框，供您新增包含 Java 專案的現有 Eclipse 或 IBM Rational Application Developer for WebSphere 軟體 (RAD) 工作區。匯入工作區之後，您就可以掃描其中所包含的任何 Java 專案。

  註：在匯入工作區之前，請確定您已依照第 41 頁的『配置 Eclipse 和 Rational Application Developer for WebSphere 軟體 (RAD) 專案的開發環境』所說明來安裝和更新開發環境。

- **從應用程式伺服器匯入**：從 Apache Tomcat 或 WebSphere Application Server Liberty 應用程式伺服器匯入現有的 Java 應用程式。
- **開啟評量**：這會啟動「開啟」對話框，供您瀏覽到 AppScan Source 評量檔。可開啟的檔案類型包括 `.ozasmt` 和 `.xml`。

工具列按鈕

表 *30.* 工具列按鈕

動作	圖示	說明
新增應用程式功能表		按一下**新增應用程式功能表**按鈕的向下箭頭，可讓您選取動作來建立新的應用程式、開啟現有的應用程式、匯入工作區，或啟動「應用程式探索助理」。
掃描選項		**掃描選項**按鈕可讓您掃描瀏覽器區段中選取的物件。掃描時會使用預設掃描配置。如果要選擇不同掃描配置來用於掃描，請按一下**掃描選項**按鈕的向下箭頭。 選取您要使用的掃描配置，或者，選擇**編輯配置**動作，將另一個掃描配置設為預設值（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。
檢視功能表		**檢視功能表**按鈕會開啟功能表，讓您重新整理瀏覽器區段及隱藏已登錄的項目。

右鍵功能表選項

右鍵功能表選項的可用性取決於瀏覽器區段中所選取的項目。

- 當在瀏覽器區段中選取**所有應用程式**時，可用的右鍵功能表選項如下：
  - **掃描所有應用程式**：掃描所有應用程式。將以預設掃描配置來執行掃描。
  - **掃描所有應用程式的方式**：選取您要使用的掃描配置，或者，選擇**編輯配置**動作，將另一個掃描配置設為預設值（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。
  - **新增應用程式**
    - **建立新的應用程式**：新增應用程式到應用程式集中。這個動作會啟動「新建應用程式」精靈。
    - **開啟現有的應用程式**：這會啟動「開啟」對話框，供您瀏覽以找出現有的應用程式並將其新增到應用程式集中。可以新增的檔案或目錄類型包括 `.paf`、`.sln`、`.dsw` 和 `.ewf`。

- **匯入現有的 Eclipse 型工作區**：這會啟動「新增工作區」對話框，供您新增包含 Java 專案的現有 Eclipse 或 IBM Rational Application Developer for WebSphere 軟體 (RAD) 工作區。匯入工作區之後，您就可以掃描其中所包含的任何 Java 專案。

  註：在匯入工作區之前，請確定您已依照第 41 頁的『配置 Eclipse 和 Rational Application Developer for WebSphere 軟體 (RAD) 專案的開發環境』所說明來安裝和更新開發環境。
- **探索應用程式**：這會啟動「應用程式探索助理」，供您快速建立及配置 Java 和 Microsoft Visual Studio 原始碼的應用程式和專案。

– **全部展開**

– **全部收合**

– **內容**：選取這個選項會開啟所選項目的「內容」視圖。

• 當在瀏覽器區段中選取應用程式時，可用的右鍵功能表選項如下：

– **掃描應用程式**：掃描所選的應用程式、專案或檔案。將以預設掃描配置來執行掃描。

– **掃描應用程式的方式**：選取您要使用的掃描配置，或者，選擇**編輯配置**動作，將另一個掃描配置設為預設值（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。

– **新增專案**

- **新建專案**：如果在「瀏覽器」視圖中選取應用程式，會啟用這個動作，選擇這個動作可讓您將新專案加到應用程式中。這個動作會啟動「新建專案精靈」。
- **現有的專案**：如果在「瀏覽器」視圖中選取應用程式，會啟用這個動作，選擇這個動作可讓您將現有的專案加到應用程式中。這個動作會啟動一個對話框，供您瀏覽要開啟的 .ppf、.vcproj、.vcxproj、.csproj、.vbproj、.dsp，或 .epf 檔。
- **多個專案**：新增多個專案到「瀏覽器」視圖所選的應用程式中。這個動作會啟動一個對話框，供您完成下列作業之一：
  - 指定要在其中搜尋專案的目錄。
  - 指定要在其中搜尋專案的工作區。
  - 指定要在其中搜尋專案的 Microsoft 解決方案檔案。

  在搜尋結果中，您可以選取一或多個要新增的專案。

– **移除應用程式**：如果在「瀏覽器」視圖中選取應用程式，會啟用這個動作，選擇這個動作會移除所選的應用程式。

– **新增自訂發現項目**：這個動作會啟動「建立自訂發現項目」對話框，供您建立所選應用程式的自訂發現項目。

– **重新整理**：重新整理所選的應用程式、專案或視圖的內容。

– 登錄/取消登錄：

- **登錄應用程式**：向 AppScan Source 登錄所選的應用程式或專案。您必須先登錄應用程式和專案，然後它們才能發佈到 AppScan Source 資料庫。
- **應用程式登錄為...**：選取這個選項會以新名稱來登錄應用程式。
- **取消登錄應用程式**：取消登錄所選的應用程式或專案。

    - **尋找**：選取這個選項會將本端應用程式或專案，關聯於另一位 AppScan Source 使用者已登錄的應用程式或專案。
  - **全部展開**
  - **全部收合**
  - **內容**：選取這個選項會開啟所選項目的「內容」視圖。
- 當在瀏覽器區段中選取專案時，可用的右鍵功能表選項如下：
  - **掃描專案**：掃描所選的應用程式、專案或檔案。將以預設掃描配置來執行掃描。
  - **掃描專案的方式**：選取您要使用的掃描配置，或者，選擇**編輯配置**動作，將另一個掃描配置設為預設值（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。
  - **複製專案**：如果在「瀏覽器」視圖中選取專案，會啟用這個動作，選擇這個動作會開啟一個對話框，供您將專案複製到另一個應用程式中，或在目前包含專案的應用程式中建立專案的副本。
  - **移除專案**：移除所選的物件。
  - 登錄/取消登錄：
    - **登錄專案**：向 AppScan Source 登錄所選的應用程式或專案。您必須先登錄應用程式和專案，然後它們才能發佈到 AppScan Source 資料庫。
    - **取消登錄專案**：取消登錄所選的應用程式或專案。
    - **尋找**：選取這個選項會將本端應用程式或專案，關聯於另一位 AppScan Source 使用者已登錄的應用程式或專案。
  - **全部展開**
  - **全部收合**
  - **內容**：選取這個選項會開啟所選項目的「內容」視圖。
- 當在瀏覽器區段中選取檔案時，可用的右鍵功能表選項如下：
  - **掃描檔案**：掃描所選的應用程式、專案或檔案。將以預設掃描配置來執行掃描。
  - **掃描檔案的方式**：選取您要使用的掃描配置，或者，選擇**編輯配置**動作，將另一個掃描配置設為預設值（在「掃描配置」視圖中，選取您要設為預設值的配置，然後按一下**選取為預設值**）。
  - **從掃描中排除**：從掃描中移除所選的檔案。
  - **在內部編輯器中開啟**：在 AppScan Source 編輯器（位於「分析」視景）中，開啟所選的檔案。
  - **在外部編輯器中開啟**：選擇一個用來開啟所選檔案的外部編輯器。
  - **內容**：選取這個選項會開啟所選項目的「內容」視圖。

應用程式和專案指示器

這份表格指出「瀏覽器」視圖中的應用程式和專案圖示。

表 *31.* 應用程式和專案圖示

應用程式或專案類型	未登錄	已登錄	遺漏/找不到
已匯入的應用程式			

表 31. 應用程式和專案圖示 (繼續)

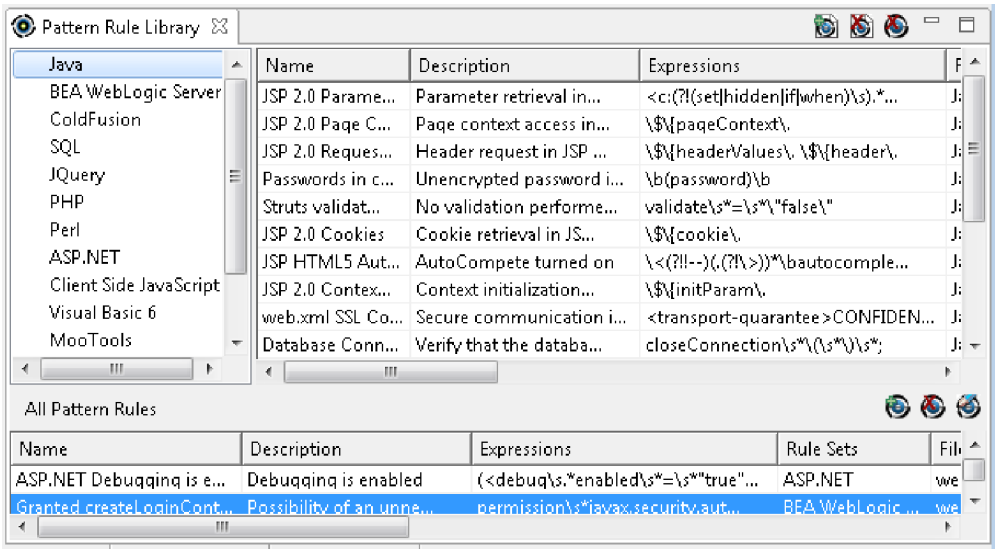
應用程式或專案類型	未登錄	已登錄	遺漏/找不到
手動建立或利用「應用程式探索助理」來建立的應用程式			
已匯入的專案			
手動建立或利用「應用程式探索助理」來建立的專案			

「瀏覽器」視圖會顯示本端應用程式和專案，以及登錄在伺服器的應用程式和專案（登錄在伺服器而未儲存在本端的應用程式和專案會變成灰色，例如，其他使用者登錄的應用程式和專案）。如果您按一下工具列**檢視功能表**按鈕，並且將**隱藏登錄在伺服器的項目**功能表項目切換成未選取，則可以檢視現有的伺服器應用程式和專案。如果專案變成灰色，您可以按一下滑鼠右鍵，然後在功能表中選擇**尋找**。

「型樣規則庫」視圖

基於型樣的掃描是以自訂搜尋準則為基礎的原始碼分析。「型樣規則庫」視圖可讓您依語言來檢視現有基於型樣的規則（包括立即可用的 AppScan Source 型樣規則庫）。此外，這個視圖還可讓您針對基於型樣的掃描，新增規則和型樣。

建置規則庫之後，您可以將型樣分析套用於特定的應用程式或專案。請參閱第 209 頁的『利用基於型樣的規則自訂』，以取得型樣搜尋的詳細資料。



「內容」視圖

「內容」視圖的內容取決於「瀏覽器」視圖中所選取的項目。內容適用於所有應用程式、個別應用程式、專案或檔案。可見的內容取決於語言或所選的專案類型。

- 第 249 頁的『「內容」視圖：所有應用程式』
- 第 218 頁的『「內容」視圖：選取的應用程式』
- 第 219 頁的『「內容」視圖：選取的專案』

- 第 256 頁的『檔案內容』

「內容」視圖：所有應用程式

如果您在「瀏覽器」視圖中選取**所有應用程式**，「內容」視圖會顯示「概觀」和「過濾器」標籤。

概觀

「概觀」標籤會顯示廣域屬性。屬性是含有類似性質的使用者定義項目的具名分組。您可以新增或刪除屬性及其值。

過濾器

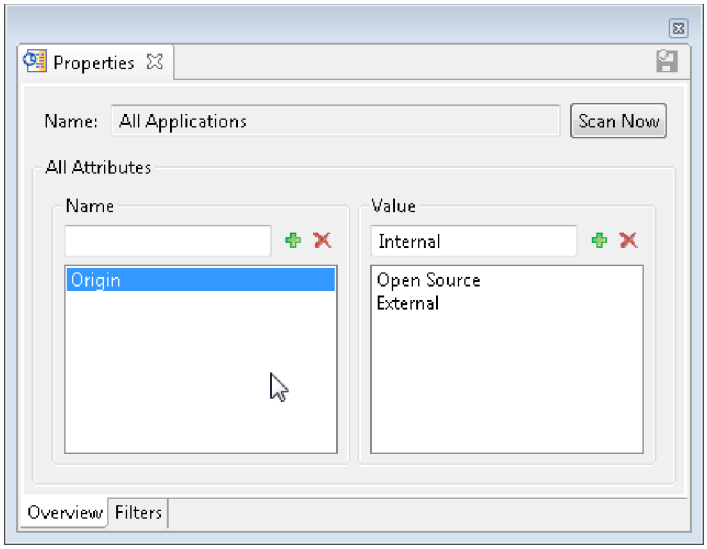
此標籤可讓您指定所有應用程式的現有過濾器，及如何套用過濾器（可直接套用過濾器 - 或套用它的反轉）。如需過濾器的相關資訊，請參閱第 121 頁的第 5 章, 『分類及分析』 - 如需全域套用它們的詳細資料，請參閱第 141 頁的『全域套用過濾器』。

過濾掉的發現項目不會出現在掃描結果中，也不會計入應用程式或專案的度量中。

新增及移除廣域屬性：

在進行應用程式的屬性分組之前，您必須先定義**所有應用程式**的屬性。

關於這項作業



如果要刪除某個廣域屬性或它的值，請選取屬性名稱或屬性值，然後按一下**移除屬性**。名稱或值不會再出現在清單中。

註：刪除屬性不會影響歷程結果。

如果要新增廣域屬性及其值，請遵循下列步驟。

程序

1. 選取**所有應用程式**。
2. 在「內容」視圖的「概觀」標籤中，鍵入屬性的名稱。

3. 按一下**新增屬性**。屬性名稱會出現在「名稱」清單中。
4. 選取指名的屬性。
5. 輸入屬性的**值**。
6. 按一下**新增值**。屬性值會出現在值清單中。

「內容」視圖：選取的應用程式

在這個視圖中，您可以配置所選應用程式的屬性。應用程式屬性相依於先前建立的廣域屬性。

- 第 218 頁的『概觀』
- 第 218 頁的『排除和過濾』
- 第 218 頁的『規則和規則集』
- 第 219 頁的『已修改的發現項目』
- 第 219 頁的『自訂發現項目』

概觀

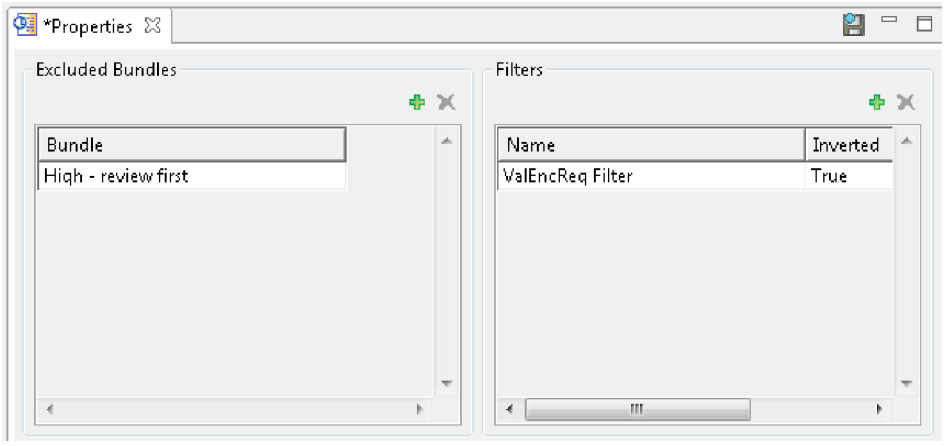
「概觀」標籤會顯示：

- 應用程式名稱。在欄位中輸入新名稱，可以將應用程式重新命名。
- 應用程式屬性

排除和過濾

此標籤可讓您指定所選取應用程式的現有過濾器，及如何套用過濾器（可直接套用過濾器 - 或套用它的反轉）。在此標籤中，您也可以管理排除掃描結果的組合。如需過濾器的相關資訊，請參閱第 121 頁的第 5 章, 『分類及分析』 - 如需全域套用它們的詳細資料，請參閱第 141 頁的『全域套用過濾器』。

已排除及過濾掉的發現項目不會出現在掃描結果中，也不會計入應用程式或專案的度量中。



規則和規則集

當在「瀏覽器」視圖中選取應用程式時，「內容」視圖中的「型樣規則和規則集」標籤可讓您新增在掃描該應用程式時將套用的型樣規則和規則集。您可以利用基於型樣

的掃描，來搜尋要呈現為發現項目的文字型樣。個別規則和規則集可同時套用至應用程式和專案。請參閱第 209 頁的『利用基於型樣的規則自訂』，以瞭解基於型樣的分析，並參閱第 215 頁的『套用型樣規則和規則集』，以瞭解如何在「內容」視圖中套用規則和規則集。

已修改的發現項目

在「已修改的發現項目」標籤中，您可以檢視、編輯或刪除任何先前修改過的發現項目，或修改現有的發現項目。已修改的發現項目是漏洞類型、嚴重性、分類有了改變，或擁有附註的發現項目。

自訂發現項目

在「自訂發現項目」標籤中，您可以檢視、新增、編輯或刪除自訂發現項目。請參閱第 154 頁的『自訂發現項目』，以取得詳細資料。

建立應用程式屬性：

程序

1. 在「概觀」標籤中，按一下**新增屬性**。
2. 在**廣域屬性**對話框中，選取要套用於應用程式的屬性名稱。
3. 按一下**值**直欄，從清單中選取屬性值。

「內容」視圖：選取的專案

在「內容」視圖的這個模式之下，您可以配置所選專案的參數。專案屬性相依於先前建立的廣域屬性。內容會隨著所選的專案而不同。

- 第 219 頁的『「選取的專案概觀」標籤』
- 第 220 頁的『過濾器 』
- 第 220 頁的『型樣規則和規則集』
- 第 221 頁的『副檔名』
- 第 221 頁的『來源』
- 第 222 頁的『JavaServer Pages (JSP) 專案相依關係』
- 第 222 頁的『專案相依關係』
- 第 222 頁的『編譯』
- 第 223 頁的『最佳化』
- 第 223 頁的『前置編譯標籤（只限 ASP.NET）』

「選取的專案概觀」標籤

「概觀」標籤會顯示：

- 專案**名稱**。在欄位中輸入新名稱，可以將專案重新命名。
- **專案檔**名稱和路徑
- **專案類型**
- 配置：這個區段顯示目標配置。如果是 .NET 和 C++ 專案，這個區段會顯示儲存在「專案相依關係」標籤中的目標配置。如果是所有其他專案類型，這個區段會顯示**預設值**。

- 過濾選項：請選取**過濾外部來源所包含的發現項目**，來濾除在所掃描專案之原始檔以外的檔案中發現的任何發現項目。這個選項會減少在編譯器產生的檔案或暫存檔中報告發現項目之專案的雜訊，例如 ASP.NET。
- 漏洞分析快取選項：如果您要反覆掃描及新增自訂規則，然後在不變更原始碼的情況下重新掃描，以精簡您的程式碼庫評量，您可以設定專案內容來使用漏洞分析快取，掃描的時間會縮短許多。如果要執行這個動作，請在專案內容中，選取**啟用漏洞分析快取**勾選框。選取這個勾選框之後，當您第一次掃描項目時，會建立一項漏洞分析快取。以後每一次掃描這個專案，都會使用這個漏洞分析快取，掃描時間會減少。

  如果要清除漏洞分析快取，以及啟用 Java 漸進式分析時建立的快取，請按一下**清除快取**。下一次掃描專案時，將會進行完整掃描，並建立新的漏洞分析快取。在下列情況下，您可以清除快取：
  - 前次掃描之後，專案中的原始碼又有了改變。
  - 您已進行專案配置變更，例如，新增或刪除原始檔。
  - 您已變更程式碼配置選項。例如，如果您要掃描 Java，且類路徑已經改變，或您要掃描 C 或 C ++，且已變更了 include 路徑或前置處理器定義，您可能會想清除快取。
  - 您已啟用 Java 漸進式分析而且想要執行完整掃描，或是遇到問題，可以透過清除快取來補救。如需相關資訊，請參閱 第 103 頁的『Java 的漸進式分析』。

  註：您也可以在利用「自訂規則精靈」來建立自訂規則時，選取**清除快取**勾選框來清除漏洞分析快取。
- **字串分析**：字串分析會監視 Java 或 Microsoft .NET 專案中的字串操作。它能夠自動偵測消毒器和驗證器常式。當進行這項偵測時，可以減少誤判 (false positive) 及假性無侵害攻擊 (false negative) 的情況。選取**啟用字串分析來尋找驗證器/消毒器功能**勾選框來啟用字串分析。**將匯入的規則套用於廣域範圍**勾選框會決定所發現的消毒器或驗證器常式，是否應該套用到單一專案或廣域層次（套用至所有專案）。

  註：套用字串分析，掃描可能會變慢。因此，建議只應在程式碼變更之後才套用，而且應於後續掃描中停用。另外，還應該將發現的常式視為建議，由審核員來進行檢查。您可以在「自訂規則」視圖中檢視這些常式。
- **檔案編碼**：您必須設定專案中各檔案的字元編碼，AppScan Source 才能適當讀取檔案並且（舉例來說）在程式碼視圖中正確顯示它們。

  註：AppScan Source 專案的預設檔案編碼是 **ISO-8859-1**。您可以在「一般」喜好設定頁面中，變更預設檔案編碼。

過濾器

此標籤可讓您指定所選取專案的現有過濾器，及如何套用過濾器（可直接套用過濾器 - 或套用它的反轉）。如需過濾器的相關資訊，請參閱第 121 頁的第 5 章, 『分類及分析』 - 如需全域套用它們的詳細資料，請參閱第 141 頁的『全域套用過濾器』。

型樣規則和規則集

當在「瀏覽器」視圖中選取專案時，「內容」視圖中的「型樣規則和規則集」標籤可讓您新增在掃描該專案時將套用的型樣規則和規則集。您可以利用基於型樣的掃描，來搜尋要呈現為發現項目的文字型樣。個別規則和規則集可同時套用至應用程式和專

案。請參閱第 209 頁的『利用基於型樣的規則自訂』，以瞭解基於型樣的分析，並參閱第 215 頁的『套用型樣規則和規則集』，以瞭解如何在「內容」視圖中套用規則和規則集。

副檔名

使用這個標籤，來配置或新增專案的有效副檔名 - 並且將檔案從掃描中排除，以及將副檔名指定為 Web 檔。

副檔名區段列出現行專案類型的第 90 頁的『專案副檔名』喜好設定頁面中已廣域設定的副檔名（您可以使用**副檔名設定**功能表，為不同的專案類型選擇副檔名）。如果要將某個副檔名從現行專案的掃描中排除，請在清單中選取它，並按一下**排除副檔名**。這會將該副檔名列在標籤的**排除的副檔名**區段中。

如果要為專案新增其他副檔名，請在**其他副檔名**區段中選取**新增副檔名**，然後輸入副檔名，並指出是否應掃描使用該副檔名的檔案、將它們視為 Web 檔或加以排除。

表 *32.* 副檔名設定

設定	說明	用法範例
掃描或評量	將使用所指副檔名的檔案包含在完整分析中。	• 如果為 Java 專案建立 .xxx 副檔名，並標示為**掃描**或**評量**，則會編譯和掃描使用該副檔名的檔案。 • 如果不應編譯和掃描檔案，檔案可以是專案的一部分，但不要標示為**掃描**或**評量**（例如 C++ 標頭檔）。這些檔案會包含在專案中，並在執行基於型樣的分析期間進行搜尋。
Web 檔	標示使用所指副檔名的檔案，以進行 JSP 編譯。這項設定容許 AppScan Source 將 Web 原始檔與非 Web 原始檔加以區隔。	如果為 Java 專案建立 .yyy 副檔名，並標示為 **Web 檔**，則會將使用該副檔名的檔案安排成專案中的 Web 原始檔。當 AppScan Source 準備分析時，會將這些檔案前置編譯成要分析的類別。
排除	不在專案中為使用所指副檔名的檔案建立原始檔。將不會掃描使用這個副檔名的檔案。	為您專案所需的檔案建立 .zzz 副檔名，以進行編譯，但不需要包含在分析中。

來源

請指定要併入掃描的來源。

- 工作目錄：AppScan Source 專案檔 (ppf) 的位置，且是所有相對路徑的基本目錄。
- **新增原始碼根目錄**和**移除原始碼根目錄**：「來源」標籤會顯示從「專案配置」精靈為專案建立的內容，或在匯入的 ppf 中所定義的內容。

  只有在選取了**原始碼根目錄**圖示時，才有**移除原始碼根目錄**可用。它用來移除原始碼根目錄。

- 尋找原始碼根目錄（只限 Java 專案）：可讓 AppScan Source for Analysis 自動尋找所有有效的原始碼根目錄。
- **原始碼根目錄**圖示下會列出專案檔。遭排除在掃描之外的檔案有紅色檔案圖示（如果用滑鼠右鍵按一下遭排除的檔案，則其功能表已停用**排除**且已啟用**併入**）。如果要排除已併入的檔案，請用滑鼠右鍵按一下它，然後在功能表中選擇**排除**。如果要併入已排除的檔案，請用滑鼠右鍵按一下它，然後在功能表中選擇**併入**。

JavaServer Pages (JSP) 專案相依關係

「JSP 專案相依關係」標籤會顯示針對指定的 JSP 專案所建立的內容。

- 包含 Web (JSP) 內容：識別專案是否為包含 JavaServer Pages 的 Web 應用程式。
- Web 環境定義根目錄：`WAR` 檔或包含 `WEB-INF` 目錄的目錄。 Web 環境定義根目錄必須是有效 Web 應用程式的根目錄。
- JSP 編譯器：既有的 Tomcat 7 是預設 JSP 編譯器設定（您可以在 Java 和 JSP 喜好設定頁面中變更預設 JSP 編譯器）。如果要瞭解 AppScan Source 支援的編譯器，請參閱 http://www.ibm.com/support/docview.wss?uid=swg27027486。

  AppScan Source 的安裝架構包含 Apache Tomcat 第 7 版和第 8 版。如果未配置 **Tomcat 7** 和 **Tomcat 8** 喜好設定頁面，AppScan Source 會利用提供的 Tomcat JSP 編譯器（目前標示為預設編譯器）來編譯 JSP 檔。如果您想要使用外部支援的 Tomcat 編譯器，請利用 Tomcat喜好設定頁面來指向您的本端 Tomcat 安裝架構。

  如果您使用 Oracle WebLogic 伺服器 或 WebSphere Application Server，您必須配置適用的喜好設定頁面來指向應用程式伺服器的本端安裝架構，以便在分析期間用來編譯 JSP。如果您尚未完成此配置，當您選取 JSP 編譯器時，會出現訊息來提示您這麼做。如果您在訊息中按一下**是**，就會看到適當的喜好設定頁面。如果您按一下**否**，JSP 編譯器選項旁邊會顯示警告鏈結（遵循此鏈結會開啟喜好設定頁面）。

專案相依關係

「專案相依關係」標籤會顯示專案內容。這個標籤的配置設定會隨著語言而改變，例如：

- **選項**可讓您選取任何其他必要的編譯器參數。
- JDK 設定專用於 Java。
- 前置處理器定義專用於 C/C++ 程式碼。當您指定前置處理器定義時，請勿併入編譯器的 `-D` 選項（例如：請指定 `a=definition1` 而非 `-Da=definition1`）。當您指定多個定義時，請使用以分號區隔的清單。
- 目標配置只適用於 .NET 和 C++ 專案。

編譯

- 選項：專案配置所需要的其他編譯器參數。
- 使用 JDK：依照「喜好設定」所配置，識別用來編譯專案的 JDK。請參閱第 79 頁的第 3 章, 『喜好設定』。

  Java 專案可以參照本端 Java Development Kit (JDK) 位置。當專案移到伺服器時，JDK 路徑可能不再有效。如果要將本端專案傳送到伺服器，您必須指出每個指定了具名 JDK 的專案之預設 JDK 路徑。

註：JSP 專案既有的預設編譯器是 Tomcat 7，它需要 Java 1.6 版或更高版本。如果保留 **Tomcat 7** 作為預設值，則使用較舊的 JDK 會導致在掃描期間發生編譯錯誤。

- 驗證：**驗證**可確保已正確配置專案相依關係。它會檢查 Java 專案在來源和類別路徑之間的配置衝突，它也會檢查編譯錯誤。如果在原始碼根目錄中，類別路徑中的類別重複，就會**發生衝突**。（如果發生衝突，請修改類別路徑來移除衝突的類別。）

  檢查衝突之後，**驗證**會判斷專案是否進行編譯，且會報告任何編譯錯誤。

最佳化

- **經過前置編譯的類別**：使用經過前置編譯的 Java 或 JSP 類別檔，而不在掃描期間編譯。選取之後，這個選項會停用來源暫置選項。
- 暫置原始檔，使編譯錯誤的影響降到最低：控制 AppScan Source 是否將來源複製到暫置目錄。

  修正不符合目錄的套件需要「Java 編譯」來開啟每個原始檔。

  清除掃描之間的暫置區會增加掃描之間的效能。

前置編譯標籤（只限 ASP.NET）

前置編譯是藉由向網站中的特殊頁面（依預設，是 precompile.axd）發出 HTTP 要求來完成。這個頁面由 web.config 中所指定的特殊 HTTP 處理程式來處理。這個處理程式會將整個網站（包括 client.aspx 檔）編譯到 .NET 架構目錄之下的 Temporary ASP.NET Files 目錄中，它們接著會全部在這裡進行掃描。

如果要掃描 ASP.NET 1.1，您必須設置網站來編譯及建置除錯資訊。接著，網站會編譯及建置除錯資訊，這種情況本身就是一個安全漏洞。您可以放心忽略這個漏洞，因為掃描需要它。不過，請務必不要在 web.config 中，以 debug=true 來編譯已部署的應用程式。

如果要前置編譯 ASP.NET 1.1 網站，請在 web.config 檔中，新增這個元素作為 <system.web> 元素的子項：

```
  <httpHandlers><add verb="*" path="precompile.axd"
type="System.Web.Handlers.BatchHandler"/></httpHandlers>
```

您也應該在編譯元素中設定 debug=true。例如：

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <system.web>
    <httpHandlers><add verb="*" path="precompile.axd"
    type="System.Web.Handlers.BatchHandler"/>
    </httpHandlers>
      <compilation
        defaultLanguage="c#"
        debug="true"
   />
...
```

這個元素向網站指定由特殊的 .Net System.Web.Handlers.BatchHandler 類別來處理 precompile.axd 頁面。這個類別會將網站內容前置編譯到 Temporary ASP.NET Files 目錄中。

- 網站：網站前置編譯的要求目標。預設位置是 precompile.axd。precompile.axd 是一個虛擬檔案，對映至 web.config 檔中所指定的檔案。
- 輸出目錄：前置編譯的目標目錄。AppScan Source 會在這個目錄尋找前置編譯的輸出。
- 前置編譯 ASP.NET 網站：在掃描期間，AppScan Source 會自動前置編譯及掃描經過前置編譯的輸出。
- 如果前置編譯失敗，就停止掃描：選取**前置編譯 ASP.NET 網站**和**如果前置編譯失敗，就停止掃描**，以便在前置編譯失敗時停止掃描。否則，只會在網站的主要輸出上繼續進行掃描。
- 立即編譯：在掃描之前進行測試，看看基於現行設定的前置編譯是否成功。編譯輸出會顯示在「前置編譯輸出」窗格中。
- 其他組合：對於任何 .NET 專案類型，請指定其他要掃描的組合。
- 專案參照：列出要在其中搜尋 .NET 組譯碼專案及現有的 .NET 專案中所參照之組合的目錄。

檔案內容

檔案內容類似於專案相依關係，通常是針對 C/C++ 應用程式來配置。

併入專案中的配置資料：併入檔案配置中的專案配置資料。之後，檔案配置由累加的專案配置和檔案配置組成。檔案配置會取代專案配置。

掃描配置視圖

「掃描配置」視圖可讓您建立啟動掃描時可用的配置。您也可以使用該視圖來設定預設掃描配置。在掃描配置中，您可以指定掃描期間要使用的來源規則，也可以包含許多掃描設定。掃描配置中的設定通常可以獲得較佳的掃描結果，而儲存這些設定的能力更可讓掃描變得輕鬆又有效率。

「掃描配置」視圖有這些主要區段：

- 第 101 頁的『掃描配置管理』
- 第 101 頁的『「一般」標籤』
- 第 102 頁的『「污染流分析」標籤』
- 第 103 頁的『「型樣分析」標籤』

掃描配置管理

使用這個區段來選取、新增、移除、儲存及共用掃描配置，以及將掃描配置設為預設值。

- 如果要建立新的掃描配置，請按一下**新建**。完成掃描配置設定之後，按一下**儲存**，儲存變更。如果要將掃描配置設為預設值，請在儲存之後按一下**選取為預設值**。如果要瞭解如何使用預設掃描配置，請參閱第 93 頁的『掃描原始碼』。
- 如果要使用現有的掃描配置，請從清單中選取它：
  - 如果您要修改掃描配置設定，請按一下**儲存**以儲存變更（可以切換至不同掃描配置來捨棄不要的變更，然後按一下**捨棄**）。
  - 如果要移除選取的掃描配置，請按一下**刪除**。
  - 如果要複製掃描配置，請按一下**複製**。這樣會根據原始掃描配置的設定來建立新的掃描配置。

- 如果要將掃描配置設為預設值，請按一下**選取為預設值**。如果要瞭解如何使用預設掃描配置，請參閱第 93 頁的『掃描原始碼』。
- 如果要與其他人共用掃描配置，請按一下**共用**。這樣會將掃描配置儲存到 AppScan Source 資料庫。

  註：如果要共用掃描配置 - 或修改或刪除共用的掃描配置 - 您必須具有**管理共用配置**許可權。如果要瞭解設定許可權的相關資訊，請參閱《*IBM Security AppScan Source* 安裝與管理手冊》。

註：AppScan Source 提供內建掃描配置。無法修改或移除這些配置。您可以在清單中選取它們來複製或檢視其設定。

「一般」標籤

基本資訊

這個區段可讓您命名掃描配置並提供說明。

過濾器

在此區段中，您可以選擇一個以上的過濾器，每當使用掃描配置時即將其套用至掃描。選取過濾器時，您可以選擇 AppScan Source 選擇的過濾器，或共用的過濾器，或您自己建立的過濾器。如需詳細資料，請參閱第 96 頁的『管理掃描配置』。

「污染流分析」標籤

污染流分析

啟用及設定污染流分析的範圍。

掃描規則

使用這個區段來決定掃描時採用的來源規則。

來源是對程式的輸入，例如：檔案、Servlet 要求、主控台輸入或 Socket。藉由排除部分來源規則，您可以加速掃描，並且避免不想要的輸入產生的偵測漏洞。

規則會標示著規則內容，指出與規則相關的特定漏洞、機制、屬性或技術。這些內容分組至規則集，對應於一般相關的規則集。藉由指定規則集或個別規則內容，您可以限制包含在掃描的來源規則。

- 選取掃描時要包含的一或多個漏洞類型（在規則集中依類型來組織）：
  - **全部**：如果選取此選項，全部支援來源產生的漏洞，都會被偵測到。
  - **使用者輸入**：如果選取此選項，將會偵測到一般使用者輸入產生的漏洞。
  - **Web 應用程式**：如果選取此選項，會偵測到 Web 應用程式風險產生的漏洞。
  - **錯誤處理和記載**：如果選取此選項，會偵測到錯誤處理和記載機制產生的漏洞。
  - **環境**：如果選取此選項，會偵測到配置檔、系統環境檔案和內容檔產生的漏洞。
  - **外部系統**：如果選取此選項，會偵測到外部實體產生的漏洞。

- **資料儲存庫**：如果選取此選項，會偵測到資料儲存庫（例如資料庫和快取）產生的漏洞。
- **不尋常事物**：如果選取此選項，會偵測到常式出現的漏洞，其通常不屬於正式作業的一部分。
- **檔案系統**：如果選取此選項，會偵測到來自檔案系統的漏洞。
- **機密資料**：如果選取此選項，會偵測到來自機密資料的漏洞。

這個區段中的每一個規則集都有浮動說明。

- 選取要併入掃描中的個別掃描規則內容：按一下**捨棄選取的規則集，讓我選取個別規則內容**。這時會開啟「選取規則內容」對話框，讓您選擇個別的規則內容。如果完成這個對話框，則會捨棄任何已選取的規則集。掃描時，將會使用含有所選規則內容的掃描規則。

進階設定

這個區段僅適用於進階使用者。它包含各種可改進掃描結果的設定。這個區段中的每一個設定都有浮動說明。

「型樣分析」標籤

型樣分析

使用掃描配置時，可使用此區段來啟用基於型樣的掃描。基於型樣的掃描是以自訂搜尋準則為基礎的原始碼分析。

型樣規則集和**型樣規則**

使用這些區段來新增型樣分析期間要使用的規則和規則集。如需相關資訊，請參閱第 209 頁的『利用基於型樣的規則自訂』和第 96 頁的『管理掃描配置』。

報告編輯器

當使用「報告編輯器」時，您可以編輯自訂報告或範本，或建立一份新的報告。自訂報告包括發現項目報告所能使用的任何項目，例如，發現項目資訊、程式碼片段、AppScan Source 追蹤，以及補救內容，另外還有漏洞矩陣。開始設計新報告之前，建議您先在「報告編輯器」中修改現有的報告範本，以熟悉報告的建立程序。

「報告編輯器」由「報告佈置」、「種類」和「預覽」標籤組成。

- **報告佈置**：設計報告的外觀。您可在佈置中新增、移除和重新排序 AppScan Source 報告元素。
- **種類**：建立和編輯種類。種類是一個發現項目群組。種類用來識別要併入報告的發現項目、這些發現項目的分組方式，以及分組次序。
- **預覽**：在編輯之時，查看現行評量的報告。

這三個標籤共用的欄位如下：

- **檔案**：儲存之分組檔（唯讀）的路徑。在儲存檔案之前，這個欄位不會顯示任何項目。在儲存之後，分組檔是一個用來定義報告的 XML 檔。
- **名稱**：使用者定義的報告名稱。

用來儲存、開啟、建立、複製和產生自訂報告的工具列按鈕包括：

- **建立新報告**：建立新的自訂報告
- **從現有中新建報告**：從現有的報告範本中建立新的自訂報告
- **開啟儲存的報告**：開啟要編輯的分組檔
- **儲存**：將現行報告儲存在指定的檔案中
- **另存新檔**：將現行報告儲存在新檔案中
- **產生這份報告的實例**：建立目前開啟之評量的報告副本

提示：如果要檢視現有報告範例，請按一下**從現有中新建報告**，然後選擇其中一個 AppScan Source 報告範本。探索範本中的「報告佈置」和「種類」標籤，可讓您概括瞭解如何設計報告。

「報告佈置」標籤

「報告佈置」標籤含有「選用區」和「佈置」區段，以及一些可讓您指定每一頁面標頭或標底的區段。

頁面標頭和頁面標底

頁面標頭欄位可讓您指定出現在每一報告頁面頂端的文字，**頁面標底**欄位可讓您指定出現在每一頁面底端的文字。

選用區

「選用區」會顯示一份組成 AppScan Source 標準報告的元素清單。有些元素只會顯示已定義在「種類」標籤中之種類的相關資訊（請參閱第 198 頁的表 19）。

表 *33.* 報告佈置選用區 - 不與種類相依的元素

報告元素	**說明**
文字標頭	將粗體文字區塊新增至報告佈置。
影像標頭	顯示調整到指定大小的影像（像素）。
AppScan Source 標頭	含有 AppScan Source 品牌行銷的報告標頭。
標題與日期	含有已掃描之項目名稱的報告標題，以及掃描日期和報告產生日期。
文字區塊	使用者定義的任何文字。也可以在**標籤**欄位中，新增文字區塊的標題。
漏洞矩陣	評量漏洞矩陣（顯示出現在「漏洞矩陣」視圖中的相同圖形）。
度量	識別在專案的所有套件中，套件、類別、方法，以及程式碼行數的總數。
掃描歷程	現行掃描的度量，以及相同目標掃描的歷程度量。

表 *34.* 報告佈置選用區 - 與種類相依的元素

報告元素	**說明**
報告卡	針對「種類」標籤中所定義的每一個種類，簡短分析其漏洞層次。包含彙總區段之報告詳細資料以及嚴重性指示器的鏈結。

表 *34.* 報告佈置選用區 - 與種類相依的元素 *(*繼續*)*

報告元素	說明
漏洞分析	這份表格分析「種類」標籤中所定義之所有種類中的漏洞數目，並依嚴重性和分類區分
局部報告卡	針對使用者在「種類」標籤中所指定的種類，分析其漏洞層次。
種類	列出「種類」標籤中所定義之所有已分類的發現項目資料。
種類	列出已定義在「種類」標籤中之一或多個種類中的所有發現項目。

佈置

當您從選用區新增項目時，它們會出現在「佈置」中。請使用區段工具列，來移除、修改或移動佈置中的項目。

「種類」標籤

您可以利用「種類」標籤，根據您選擇的組合、內容或發現項目，來新增含有發現項目的種類。之後，當您新增某些項目到「佈置」時，就可以使用種類。舉例來說，當您新增「漏洞分析」到「佈置」時，會在佈置中新增一份表格，其中含有所有種類中漏洞數目的分析（依嚴重性和分類區分）。「種類」標籤有兩個窗格，一個是種類的樹狀結構，另一個是用來編輯所選種類的屬性。每個種類都包含評量中能夠滿足所定義特定需求的發現項目。

可用的種類包括：

- 組合：組合種類由一份組合名稱清單組成。組合中的任何發現項目，只要名稱出現在這份清單中，就會出現在這個種類中。雖然您從現行評量中選擇組合，但由於組合是依名稱比對，您可以將組合種類套用於任何評量。
- 個別發現項目：請選擇要新增到種類中的特定發現項目。只會新增發現項目的 Snapshot 到報告中。如果發現項目新增到報告之後，您又修改它，變更不會反映在報告中。
- 「漏洞類型」、「機制」和「技術」內容：請從 AppScan Source 安全知識庫的 API 中，選擇各組內容及必要的內容。如果發現項目至少包含其中一個**內容**及所有**必要的內容**，它就會併入報告中。

這份表格識別種類窗格及組成窗格的項目。

表 *35.* 「種類」標籤屬性

屬性	說明	如何編輯
標籤	種類的簡短名稱，例如「緩衝區溢位」。標籤用來識別種類樹狀結構清單中的種類，它是自訂報告中的種類標題。	請在單行文字欄位中輸入標籤。
摘要	指出這個種類報告了多少發現項目的句子範本。在產生報告期間，實際的計數會取代 `%FindingCount%`。	輸入種類的簡要說明，按一下**新增計數**，將 `%FindingCount%` 變數放在詞組中的游標位置。

表 35. 「種類」標籤屬性 (繼續)

屬性	說明	如何編輯
文字	簡短種類說明。	輸入文字來說明種類。
內容（只限內容種類）	這個種類會報告至少有其中一個內容的發現項目。如果發現項目未完整具備所有列出的必要內容，發現項目就不會併到這個種類中。	請在工具列中，按一下**新增**，從「新增內容」對話框中選取一個內容。按一下**移除**，可以從清單中移除選取的項目。
必要的內容（只限內容種類）	在這個種類之下，含有所有必要內容及至少一個內容的發現項目會出現在報告中。	請在工具列中，按一下**新增**，從「新增內容」對話框中選取一個內容。按一下**移除**，可以從清單中移除選取的項目。
組合（只限組合種類）	指定要併入這個種類的組合名稱。	在「組合」區段中，按一下**新增組合**，從清單中選取組合。
發現項目（僅限於「發現項目」種類）	指定要併入這個種類的發現項目。	選取任何發現項目表格中的發現項目，然後按一下表格工具列上的**新增發現項目**，來新增選取的發現項目。如果有多個視圖含有選取的發現項目，會提示您選取包含您要新增之所選發現項目的視圖。 您也可以將發現項目表格中的發現項目，拖曳到「報告編輯器」視圖或「報告編輯器」中的表格，或直接拖曳到種類樹狀結構中的現有發現項目種類。

「預覽」標籤

當編輯範本時，您可以預覽 AppScan Source for Analysis 報告。請從「預覽」窗格中，按一下**預覽**來查看開啟之評量的報告。

協助掃描輸出的視圖

這一節的視圖用於檢視和管理掃描輸出。

- 『「主控台」視圖』
- 第 262 頁的『「度量」視圖』
- 第 262 頁的『「我的評量」視圖』
- 第 263 頁的『「已發佈的評量」視圖』

「主控台」視圖

「主控台」視圖會顯示現行掃描的輸出，其中包括狀態資訊、輸出文字和錯誤訊息。這個視圖可以顯示兩個主控台，一個是目前執行中的掃描，另一個是已完成的掃描。

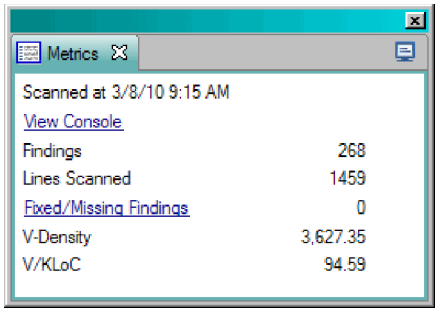
輸出主控台會顯示完整的掃描輸出，其中包括：掃描的檔案、掃描的檔案總數、發現的漏洞總數、掃描時間，以及漏洞密度。

工具列按鈕用來操作主控台輸出。

錯誤主控台會顯示輸出錯誤訊息，以及掃描中的錯誤數目。在掃描期間，會更新錯誤值。

「度量」視圖

「度量」視圖提供基於每個評量的統計資料，包括掃描的程式碼行數、發現項目總數、V 密度，以及 V/KLoC。



檢視主控台

開啟「主控台」視圖來查看現行掃描輸出的超鏈結。

發現項目

掃描所識別的發現項目數。

掃描行數

掃描的程式碼行數。

已修正/遺漏的發現項目

包含在應用程式組合中，但在這項掃描中找不到的項目數。

V 密度

這是可供採用一致的方式來評估應用程式漏洞的數值表示式。將發現項目的數目及嚴重性關聯到目前分析的應用程式或專案的大小，可以計算出「V 密度」。

V/KLoC

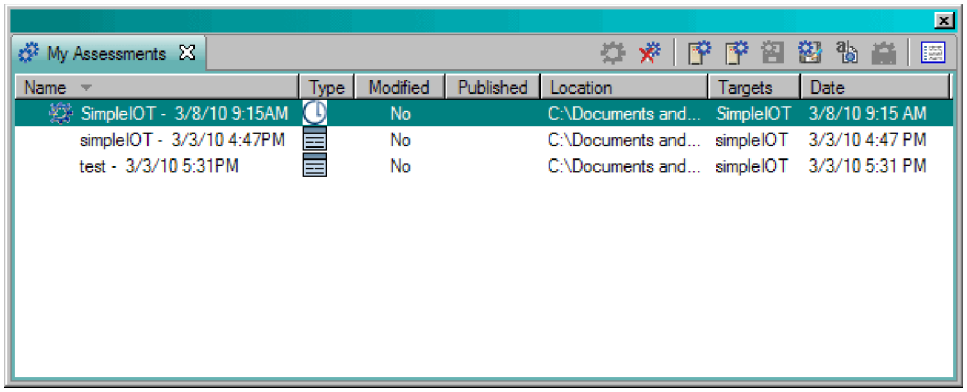
每千行程式碼所找到的漏洞數

「我的評量」視圖

「我的評量」視圖包含一份評量清單（目前開啟的評量，以及您已儲存的任何評量）。如果修改了評量的現行工作集（比方說，您增加新的評量或修改評量），視圖標題旁的星號表示工作集中有未儲存的變更。

- **名稱**：評量名稱。
- **類型**：此圖示指出掃描涵蓋的應用程式 ( )、專案 ( ) 或檔案 ( )。評量名稱旁的星號表示評量目前已開啟。

- **掃描配置**：用於掃描的掃描配置。
- **已修改**：**是**或**否**指出評量的已修改狀態。
- **已發佈**：指出評量已發佈到 AppScan Source 資料庫的指示器。
- **位置**：評量檔的路徑 (<file_name>.ozasmt)。
- **目標**：掃描的應用程式、專案或檔案。
- **日期**：掃描完成日期。



掃描完成之後，它們會自動出現在「我的評量」視圖中。在這個視圖中，可見的評量包含這部電腦所進行的掃描，或是您新增的掃描。

在這個視圖中，您可以開啟、新增、移除、發佈、儲存、重新命名或比較評量。如果您從這個視圖中移除評量，且未儲存或發佈它，它就會永久刪除。請注意，每個已儲存的評量都包含所有結果、輸出和錯誤日誌。請參閱第 118 頁的『儲存評量』，以取得儲存和發佈評量的詳細資料。

請參閱第 153 頁的『在「評量差異」視圖中比較兩項評量』，以取得比較評量的詳細資料。

提示：一般喜好設定會設定「已發佈的評量」和「我的評量」視圖所顯示的評量數目上限。

「已發佈的評量」視圖

「已發佈的評量」視圖會列出已發佈到 AppScan Source 資料庫的評量。

- **名稱**：評量名稱。
- **類型**：此圖示指出掃描涵蓋的應用程式 ( )、專案 ( ) 或檔案 ( )。評量名稱旁的星號表示評量目前已開啟。
- **掃描配置**：用於掃描的掃描配置。
- **發佈者**：評量發佈人員的使用者名稱
- **目標**：掃描的應用程式、專案或檔案。
- **日期**：掃描完成日期。

在「已發佈的評量」視圖中，您可以執行下列動作：

- 新增評量至「我的評量」視圖
- 過濾評量
- 開啟和刪除評量

- 關閉評量
- 比較評量
- 儲存評量
- 重新命名評量
- 檢視度量

提示：一般喜好設定會設定「已發佈的評量」和「我的評量」視圖所顯示的評量數目上限。

協助分類的視圖

這一節的視圖用於檢視和管理細部掃描輸出。

- 『「評量差異」視圖』
- 第 265 頁的『「自訂發現項目」視圖』
- 第 269 頁的『「已排除的發現項目」視圖』
- 第 267 頁的『「發現項目」視圖』
- 第 269 頁的『「已修正/遺漏的發現項目」視圖』
- 第 269 頁的『「已修改的發現項目」視圖』
- 第 269 頁的『「搜尋結果」視圖』
- 第 270 頁的『「報告」視圖』
- 第 272 頁的『「來源和接收槽」視圖』

「評量差異」視圖

「評量差異」視圖代表「我的評量」視圖和「發現項目」視圖的組合。當您選取兩個要比較的評量時，會顯示兩個評量之間的差異。

在這個視圖中，您會見到新的、已修正/遺漏及共同發現項目的總數。

- 共同發現項目會出現在兩個評量中
- 新發現項目是在這兩個評量之中，只出現在最新者之中的發現項目（藍色強調）
- 已修正/遺漏的發現項目是只出現在較舊評量中的發現項目（綠色斜體強調）

右窗格顯示發現項目。請用滑鼠右鍵按一下表格中的發現項目，來執行下列動作：

- 產生發現項目報告
- 將發現項目當作問題報告來提交
- 在外部編輯器中開啟
- 在內部編輯器中開啟

左窗格列出要比較的評量。

註：「評量差異」視圖會忽略過濾器。

「自訂發現項目」視圖

「自訂發現項目」視圖會顯示目前開啟的評量中，現有的使用者定義發現項目或自訂發現項目。在這個視圖中，您可以建立、刪除和修改現行評量的自訂發現項目。當在「自訂發現項目」視圖中建立自訂發現項目時，新的發現項目會新增到現行評量中，且會更新評量度量。

過濾器和組合不會影響「自訂發現項目」視圖中的發現項目。在這個視圖中，您無法檢視自訂報告結果，或儲存所選的發現項目。

含有發現項目的視圖

許多 AppScan Source for Analysis 視圖都包含發現項目：

- 「發現項目」視圖
- 「已修改的發現項目」視圖
- 「自訂發現項目」視圖
- 「已排除的發現項目」視圖
- 「組合」視圖
- 「已修正/遺漏的發現項目」視圖
- 「報告」視圖
- 「搜尋結果」視圖
- 「評量差異」視圖

發現項目表格

這份表格說明發現項目表格中可用的直欄。如果無法使用某個直欄，很可能是表格將它隱藏了。如果要選取直欄來檢視（或在表格中執行任何其他自訂作業），請遵循第 266 頁的『自訂發現項目表格』中的指示。

表 *36.* 發現項目表格

直欄標題	**說明**
追蹤	這個直欄中的圖示指出遺失或已知的接收槽有追蹤資料存在。
嚴重性	• 高：造成資料在機密性、完整性或可用性方面的風險，以及/或造成處理資源在完整性或可用性方面的風險。高度嚴重性狀況應該優先立即補救。 • 中：造成資料安全和資源完整性的風險，但狀況不是很容易遭到攻擊。中度嚴重性狀況應該儘可能檢查及補救。 • 低：造成最低的資料安全或資源完整性風險。 • 參考資訊：發現項目本身不容易產生危害。它說明程式碼中使用的技術、架構性質或安全機制。

表 *36.* 發現項目表格 *(*繼續*)*

直欄標題	說明
分類	發現項目的類型：**明確**或**可疑**安全發現項目 - 或**掃描涵蓋面**發現項目。 **註：**在某些情況下，**無**這項分類可用來表示既非安全發現項目也非掃描涵蓋面發現項目的分類。
漏洞類型	漏洞種類，例如 `Validation.Required` 或 `Injection.SQL`。
API	有漏洞的呼叫，顯示 API 和傳給 API 的引數。
來源	來源是對程式的輸入，例如：檔案、Servlet 要求、主控台輸入或 Socket。對於大部分的輸入來源而言，傳回的資料內容和長度都是沒有限制的。如果沒有對輸入進行檢查，則會將其視為受到污染。
接收槽	接收槽可以是資料能夠寫出的任何外部格式。資料庫、檔案、主控台輸出和 Socket 都是接收槽的範例。未經檢查，便將資料寫入接收槽，可能是一個嚴重的安全漏洞。
目錄	掃描之檔案的完整路徑。
檔案	出現安全發現項目或掃描涵蓋面發現項目的程式檔名稱。發現項目中的檔案路徑是相對於掃描的專案工作目錄。
呼叫方法	發出有漏洞呼叫的函數（或方法）。
行	程式檔中有漏洞的 API 所在的行號。
組合	包含這個發現項目的組合。
CWE	ID and topic of the community-developed dictionar一般軟體弱點的社群開發字典 ID 和主題（一般弱點列舉 (CWE) 主題）。

自訂發現項目表格：

除了 AppScan Source for Analysis 中的「評量差異」視圖之外，在所有含有發現項目的視圖中，您可以只將您想要看到的直欄及直欄次序識別出來，以自訂發現項目表格。每個視圖可以各有不同設定，您也可以將選項套用於所有視圖。如果要自訂直欄次序，請遵循這個作業主題中的步驟。

關於這項作業

如果要瞭解發現項目表格中的直欄，請參閱第 265 頁的『發現項目表格』。

程序

1. 按一下**選取直欄及進行排序**工具列按鈕。

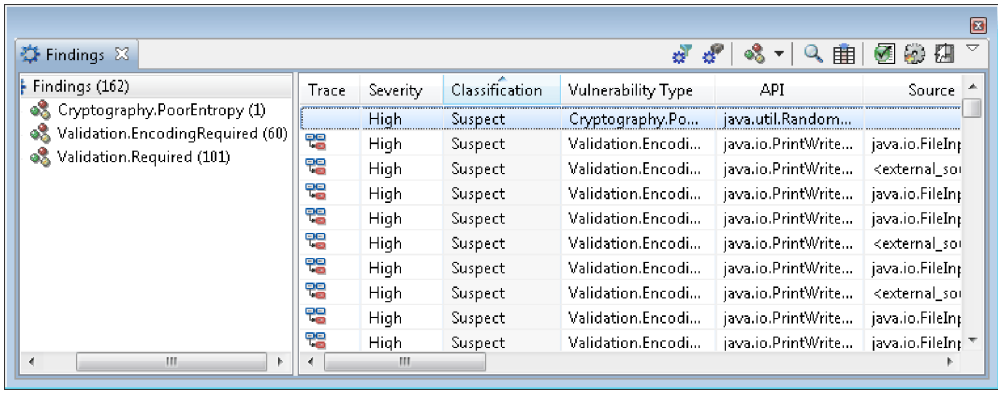
   註：在 AppScan Source for Development（Visual Studio 外掛程式）中，按一下**選取表格直欄及進行排列**工具列按鈕。

2. 在**選取直欄及進行排序**對話框中，選取直欄名稱，然後按**上移鍵**或**下移鍵**來移動直欄位置。
3. 按一下**新增直欄**按鈕，將直欄新增到視圖中。另外，您也可以按一下**刪除直欄**按鈕來移除視圖中的直欄。

   註：在 AppScan Source for Development（Visual Studio 外掛程式）中，這些按鈕的標籤分別是**插入**和**移除**。
4. 按一下**還原預設值**來重設預設的直欄和直欄次序。
5. 按一下**確定**來儲存設定。

「發現項目」視圖

「發現項目」視圖包含評量中各發現項目的資料。您可以利用這個主題所列的參數來將這些發現項目分組。



記住：在 AppScan Source for Development（Eclipse 外掛程式）和 AppScan Source for Analysis 的使用者介面中，這些稱為視圖。在 AppScan Source for Development（Visual Studio 外掛程式）的使用者介面中，這些稱為視窗。在這份說明文件中，視圖一詞通常用來表示視圖和視窗。

發現項目表格參數分組

在「發現項目」視圖中，請選擇**選取樹狀結構階層**工具列按鈕向下箭頭，然後選擇用來將發現項目分組的參數。

表 37. 發現項目表格參數分組

模式	**分組**
漏洞類型	類型、嚴重性、分類
分類	分類、嚴重性、類型
檔案	專案、目錄、檔案、方法
API	API、類型
組合	組合、類型、API
CWE	CWE
表格	不分組

工具列按鈕

表 *38.* 工具列按鈕

動作	圖示	說明
顯示不符合過濾器的發現項目		這個按鈕可讓您切換顯示「發現項目」視圖中已過濾的發現項目。
顯示連結的發現項目		這個按鈕可讓您切換顯示「發現項目」視圖中已組合的發現項目。這個動作會隱藏您建立之所有已併入組合中的發現項目。這個設定不會影響已排除之組合中發現項目的顯示 - 這些發現項目絕不會出現在「發現項目」視圖中。
選取樹狀結構階層	這會隨著所選的分組而不同。	請參閱第 267 頁的『發現項目表格參數分組』。
搜尋		這個按鈕會開啟一個對話框，供您搜尋發現項目。這個對話框有各種可用的搜尋選項。進行搜尋之後，結果會出現在「搜尋結果」視圖中。
選取直欄及進行排序		這個按鈕會開啟「選取直欄及進行排序」對話框，供您新增或移除直欄，或修改現有的直欄。
報告視圖		這個按鈕會開啟「報告」視圖，其中會根據測量是否符合軟體安全最佳實務和規章需求的綜合性審核報告來顯示發現項目。
建立自訂發現項目		只有在 AppScan Source for Analysis 中，才能夠使用這個按鈕。選取它會開啟「建立自訂發現項目」對話框，供您新增自訂發現項目到現行評量中。
儲存所選的發現項目		如果選取一或多個發現項目，這個按鈕會開啟「儲存所選的發現項目」對話框，供您將所選的發現項目儲存到新的評量檔中。
檢視功能表		這個功能表可用來快速存取所有工具列按鈕動作。

在「發現項目」視圖中，您可以：

- 在程式碼編輯器中開啟發現項目
- 建立排除項目

- 修改發現項目
- 以不同的分組來檢視發現項目
- 搜尋發現項目中的特定項目

在 AppScan Source for Analysis 中使用這個視圖時，您也可以執行下列動作：
- 將發現項目移到組合中
- 將問題報告提交至問題追蹤系統
- 建立自訂發現項目
- 產生發現項目報告
- 以電子郵件傳送發現項目或組合

「已排除的發現項目」視圖

在「已排除的發現項目」視圖中，只包含已排除的發現項目。已排除的發現項目是您省略不掃描的發現項目。在這個視圖中，您可以搜尋特定的發現項目。這個視圖中的直欄與「發現項目」視圖中的直欄相同。

如果要重新併入已排除的發現項目，請遵循第 143 頁的『重新併入已標示排除的發現項目』中的指示。

「已修改的發現項目」視圖

「已修改的發現項目」視圖包含現行應用程式所有已變更的發現項目。已修改的發現項目是漏洞類型、嚴重性、分類有了改變，或擁有附註的發現項目。遺失的發現項目（不在目前所開啟之評量中的發現項目）顯示成綠色斜體，無法修改。

在這個視圖中，您可以：
- 搜尋特定發現項目。
- 進行其他修改

在 AppScan Source for Analysis 中，您也可以在這個視圖中執行這些動作：
- 新增發現項目至組合中
- 將問題報告提交至問題追蹤系統
- 以電子郵件傳送發現項目（問題報告）
- 產生發現項目報告

「已修正/遺漏的發現項目」視圖

「已修正/遺漏的發現項目」視圖會識別在組合之中，但不在現行評量中的發現項目。將發現項目識別為修正/遺漏的發現項目，是因為已將它解決、移除，或來源檔案未掃描。

「搜尋結果」視圖

當您搜尋發現項目時，結果會出現在「搜尋結果」視圖中。

在這個視圖中，您可以
- 排列發現項目
- 在內部或外部編輯器中，編輯程式碼
- 設定漏洞類型

- 將可疑和掃描涵蓋面發現項目升級為明確的發現項目
- 設定嚴重性層次
- 標註發現項目
- 排除特定發現項目
- 執行後續搜尋

在 AppScan Source for Analysis 中使用這個視圖時，您也可以執行下列動作：

- 新增發現項目至組合中
- 將問題報告提交至問題追蹤系統，或以電子郵件傳送發現項目
- 產生發現項目報告

「搜尋結果」視圖只包含符合搜尋準則的項目，最多維護 5 個搜尋結果。比方說，如果您在「搜尋結果」視圖中，搜尋「緩衝區溢位」漏洞類型，然後搜尋「明確的」分類，則搜尋結果會是這兩種搜尋的交集。

搜尋準則出現在「搜尋」欄位中，而搜尋的表示法為 "<keyword>" in <originating_view>: <fields searched>，例如 "shutdown" in Findings [Context, API, Method]。如果關閉現行評量，所有搜尋結果都會捨棄，「搜尋」欄位會顯示下列文字：No Current Search。

「報告」視圖

「報告」視圖可讓您根據各種測量是否符合軟體安全最佳實務和規章需求的審核報告，來組織掃描結果。

視圖會根據這些報告來顯示發現項目：

- 第 195 頁的『CWE/SANS Top 25 2011 報告』
- 第 195 頁的『DISA 應用程式安全及開發 STIG 3.10 版報告』
- 第 196 頁的『「開放式 Web 應用程式安全專案 (OWASP)」Mobile Top 10 報告』
- 第 195 頁的『「開放式 Web 應用程式安全專案 (OWASP)」Top 10 2013 報告』
- 第 196 頁的『付款卡產業資料安全標準 (PCI DSS) 3.2 版報告』
- 第 196 頁的『Software Security Profile 報告』

如果您利用 AppScan Source for Analysis 來建立儲存於 <data_dir>\reports\profile（其中 <data_dir> 是 AppScan Source 程式資料的位置，如第 282 頁的『安裝和使用者資料檔位置』所述） 的自訂報告，您也可以利用「報告」視圖，依自訂報告來顯示發現項目。

「報告」視圖中的直欄與第 267 頁的『「發現項目」視圖』相同。

搜尋發現項目

在包含發現項目的多重視圖中，您可以搜尋特定的發現項目。搜尋準則包含組合、程式碼、檔案、專案或漏洞類型。搜尋結果會出現在「搜尋結果」視圖中。

當搜尋程式碼時，可以搜尋多個項目，也可以搜尋所有項目，其中包含：

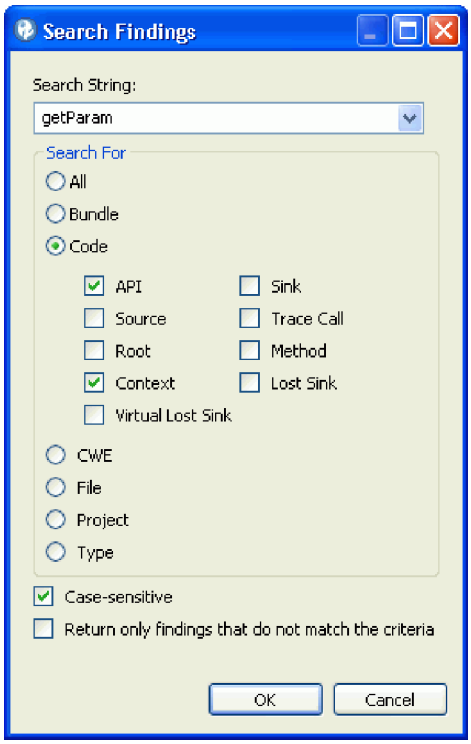
- API
- 環境定義
- 方法

- 來源
- 接收槽
- 遺失的接收槽
- 根
- 追蹤呼叫

在所有發現項目中，搜尋單一項目每次出現之處：

程序

1. 選取要進行搜尋的視圖。
2. 從主功能表中，選取**編輯** > **尋找**（在 AppScan Source for Development（Eclipse 外掛程式） 中，選取**編輯** > **尋找/取代**，或在 AppScan Source for Development（Visual Studio 外掛程式）中，在含有發現項目的視圖中，按一下**搜尋**按鈕）。



3. 在**搜尋發現項目**對話框中，輸入搜尋字串。
4. 在**組合**、**程式碼**、**CWE**、**檔案**、**專案**、**類型**或**全部**中，搜尋字串。相符的發現項目會出現在「搜尋結果」視圖中。

   選取**區分大小寫**來搜尋區分大小寫的文字。

   如果您使用 AppScan Source for Analysis 或 AppScan Source for Development（Eclipse 外掛程式），請選取**只傳回不符合準則的發現項目**，以傳回未對應於搜尋準則的發現項目。

在發現項目表格中搜尋發現項目：

程序

1. 在工具列上，按一下**搜尋**。
2. 識別搜尋的性質，然後按一下**確定**。

在發現項目樹狀結構中搜尋：

程序

1. 在工具列上，按一下**搜尋**。
2. 識別搜尋的性質，然後按一下**確定**。

結果

在發現項目視圖中，您也可以在可見的發現項目子集內搜尋。例如，您可能會想搜尋特定子集（如「漏洞類型」）內的發現項目。

「來源和接收槽」視圖

「來源和接收槽」視圖可用來檢視基於輸入及輸出追蹤的發現項目。

「來源和接收槽」視圖分成三個區段：

- **來源和接收槽**：在左畫面中，有三個最上層節點：
  - **來源**：來源是對程式的輸入，例如：檔案、Servlet 要求、主控台輸入或 Socket。對於大部分的輸入來源而言，傳回的資料內容和長度都是沒有限制的。如果沒有對輸入進行檢查，則會將其視為受到污染。來源會列在任何發現項目表格的**來源**直欄中。
  - **接收槽**：接收槽可以是資料能夠寫出的任何外部格式。資料庫、檔案、主控台輸出和 Socket 都是接收槽的範例。未經檢查，便將資料寫入接收槽，可能是一個嚴重的安全漏洞。
  - **遺失的接收槽**：遺失的接收槽是指無法再追蹤的 API 方法。

  每個節點都可以展開，以顯示受影響的套件。接著，套件又可以展開，以顯示受影響的類別，然後再顯示方法。之後，可以展開這些方法來顯示追蹤內相反端的套件、類別和方法。比方說，如果您關心特定接收槽，您可以往下探查到**接收槽**根之下的方法。到了這裡，這個方法之下的樹狀結構會顯示返回的路徑，以便回到先前走向這個接收槽的所有來源：

```
- Sources
 - packageA
  - classA
   - methodA
    - packageB
     - classB
           - methodB (at opposite end of trace)
- Sinks
 - packageB
  - classB
   - methodB
    - packageA
     - classA
      - methodA
- Lost Sinks
```

在這個樹狀結構視圖中所做的選擇，會決定視圖內另外兩個區段中所顯示的內容。

- **中間節點**：在這個視圖區段中，會顯示追蹤內所有中間節點的聯集，這些中間節點會套用至「來源和接收槽」區段的選項中。它可讓您精簡發現項目表格的內容。

  依預設，會隱藏這個區段。按一下**顯示/隱藏中間呼叫表**，就能夠顯示它（或再次隱藏）。

  如果只要顯示套件、類別或方法的發現項目，請選取它在**必要**直欄中的勾選框。如果要濾除套件、類別或方法的發現項目，請選取它在**移除**直欄中的勾選框。這個區段的過濾器設定可用來建立新的過濾器。

  用法範例：「來源和接收槽」區段的樹狀結構節點如下：

  ```
  - Sources
   - java.util
    - Properties
     - getProperty
  ```

  當選取 getProperty 時，發現項目表格只會顯示其中含有以 getProperty 為來源之追蹤的發現項目。這時中間節點區段會顯示來源為 getProperty 之所有追蹤的所有中間節點（指追蹤內除了來源和接收槽以外的所有節點）。不過，如果追蹤通過特定的 API，則您就可以不管它。比方說，您可能會有一個驗證常式，能夠確保來自 getProperty 的資料有效，因而不想查看通過這個驗證常式的追蹤。中間節點區段會包括這個驗證常式，因為它是追蹤的一個中間節點。您可以在中間節點區段中，瀏覽到這個驗證常式，按一下它的**移除**勾選框。這會從發現項目表格中，移除其追蹤通過這個中間節點的所有發現項目。

- **發現項目**：這個區段包含一份與第 267 頁的『「發現項目」視圖』及其他含有發現項目之視圖相同的第 265 頁的『發現項目表格』（以及相關的動作）。它會顯示您選擇在視圖內另外兩個區段中顯示的來源、接收槽和中間節點的發現項目。

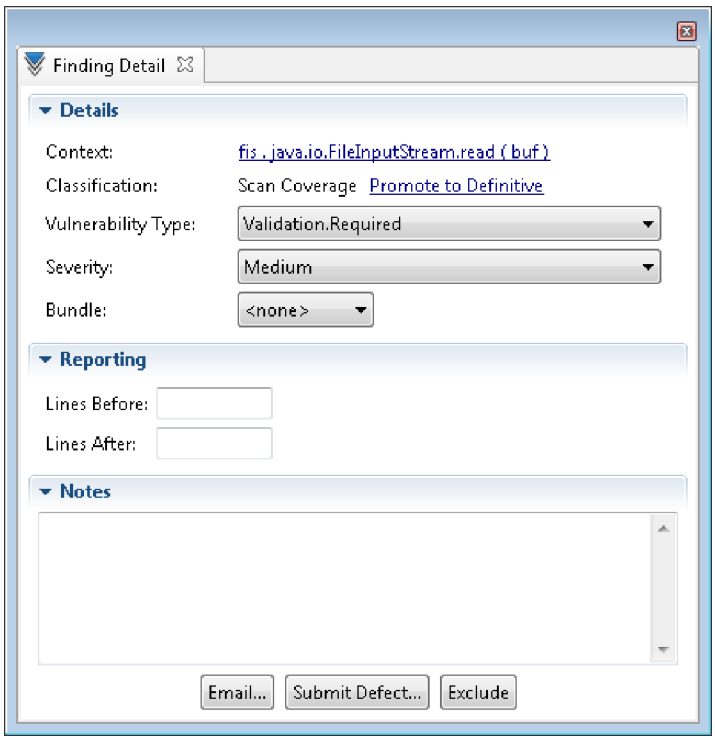
可讓您調查單一發現項目的視圖

這一節的視圖是用於調查單一發現項目。

- 第 149 頁的『「發現項目詳細資料」視圖』
- 第 275 頁的『「補救協助」視圖』
- 第 276 頁的『「追蹤」視圖』

「發現項目詳細資料」視圖

選取發現項目時，隨即會顯示「發現項目詳細資料」視圖，可讓您修改發現項目的內容。使用這個視圖，您可以修改個別的發現項目。



- 第 150 頁的『「詳細資料」區段』
- 第 150 頁的『報告區段（僅限 AppScan Source for Analysis 和 AppScan Source for Development（Eclipse 外掛程式））』
- 第 150 頁的『附註區段』
- 第 151 頁的『「發現項目詳細資料」視圖動作』
- 第 151 頁的『自訂發現項目的「發現項目詳細資料」視圖（只限 AppScan Source for Analysis）』

「詳細資料」區段

- **環境定義**：漏洞週遭的程式碼片段
- **分類**：明確或可疑安全發現項目 - 或掃描涵蓋面發現項目 - 有可升級為**明確**發現項目或回復至原始值（如果分類已變更）的鏈結
- **漏洞類型**
- **嚴重性**：高、中、低或參考資訊。
- **組合**：包含發現項目的組合名稱（AppScan Source for Development（Visual Studio 外掛程式）中沒有這個選項）

報告區段（僅限 AppScan Source for Analysis 和 AppScan Source for Development（Eclipse 外掛程式））

指定報告中要於發現項目之前和/或之後併入的程式碼行數。

附註區段

標註發現項目。

「發現項目詳細資料」視圖動作

- **排除**：請按一下**排除**來排除（移除）發現項目表格中的發現項目。如果要檢視已排除的發現項目，請開啟「已排除的發現項目」視圖。
- 只在 AppScan Source for Analysis 中才有：
  - **電子郵件**：如果您已配置電子郵件喜好設定，您可以利用電子郵件，將發現項目組合直接傳送給開發人員，告訴他們在掃描之後所發現的可能問題報告。電子郵件包含含有發現項目的組合附件，以及說明發現項目的電子郵件文字。
    1. 如果要用電子郵件傳送「發現項目詳細資料」視圖中的現行發現項目，請按一下**電子郵件**。
    2. 在「附件檔名」對話框中，指定要附加到電子郵件的發現項目組合名稱。例如，在**附件檔名**欄位中指定 `my_finding`，會將檔名為 `my_finding.ozbdl` 的組合附加到電子郵件中。
    3. 按一下**確定**來開啟「以電子郵件傳送發現項目」對話框。依預設，「以電子郵件傳送發現項目」對話框中的**郵件收件者**欄位，會移入電子郵件喜好設定所指定的**收件者位址**，不過，準備電子郵件時，很容易改變它。請在這個對話框中，檢閱電子郵件的內容，然後按一下**確定**來傳送電子郵件。
  - **提交問題報告**：如果要將發現項目當作問題報告來提交，請按一下**提交問題報告**。這時會開啟「選取問題追蹤系統」對話框。
    - 如果您選取 **ClearQuest**，然後按一下**確定**，即會開啟「附件檔名」對話框。請在這個對話框中，指定要附加到問題報告的發現項目組合名稱，然後按一下**確定**。登入 Rational ClearQuest，提交發現項目。
    - 如果您選取 **Quality Center**，然後按一下**確定**，這時會開啟「登入」對話框，供您登入 Quality Center 來提交發現項目。
    - 如果您選取 **Team Foundation Server** 選項，這時會開啟對話框，提示您登入問題追蹤系統，並提供其他配置詳細資料。

      註：Rational Team Concert 是 macOS 上唯一支援的問題報告追蹤系統。

自訂發現項目的「發現項目詳細資料」視圖（只限 AppScan Source for Analysis）

在自訂發現項目的「發現項目詳細資料」視圖中，有其他可供編輯的資訊：

- 檔案
- 行
- 直欄
- API

此外，您編輯 第 150 頁的『「詳細資料」區段』 的方法不同於部分欄位的標準發現項目（例如，自訂發現項目出現在清單中的分類）。

「補救協助」視圖

AppScan Source 安全知識庫提供關於各漏洞特定環境定義的知識。知識庫告訴您漏洞是什麼、為什麼不安全、如何修正它，以及將來如何加以避免。掃描原始碼之後，知識庫會提供從關鍵應用程式中消除風險所需要的特定資訊。知識庫補救建議出現在「補救協助」視圖中。掃描之後，知識庫會提供從關鍵應用程式中消除風險所需要的特定資訊。

如果要檢視知識庫及取得補救建議，請執行下列動作：

- 在發現項目表格中選取一個發現項目，然後開啟「知識庫說明」或「補救協助」視圖。
- 在 AppScan Source for Analysis 中，您也可以從功能表中，選取**說明** > **安全 知識庫**，來查看整個知識庫。

資料庫中的特定 API 會列出嚴重性層次和嚴重性類型。例如，strcpy() API 是一個「緩衝區溢位」類型，嚴重性層次是「高」。說明指出 strcpy() 很容易造成目的地緩衝區溢位，因為它不知道目的地緩衝區的長度，因此無法檢查以確定它不會改寫這個緩衝區。請利用 strncpy () 來修正這個問題，它有一個長度參數。

如果發現項目有相關聯的「一般弱點列舉 (CWE)」ID，在「補救協助」視圖中，會有一個超鏈結通往 CWE 主題 (CWE: <id>)，位置如下：http://cwe.mitre.org/data/definitions/<CWE_ID>.html。

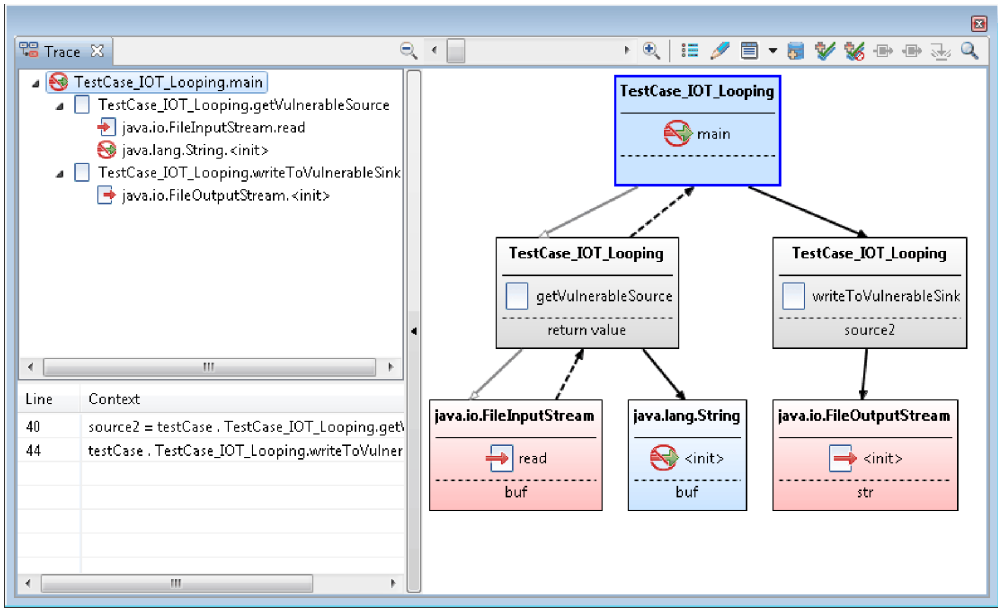
「追蹤」視圖

AppScan Source 會執行輸入/輸出分析，且會識別及顯示這些漏洞。在發現項目清單中，有一個圖示用來識別包含 AppScan Source 追蹤圖的列。

在「追蹤」視圖中，您會看到輸入及輸出堆疊會合的根節點。輸入堆疊是一系列的呼叫，通往已知會提供受污染資料的來源。輸出堆疊是通往接收槽的一系列呼叫。當分析的程式碼能夠追蹤使用來自未受保護的來源到未受保護的接收槽的情形時，會產生一項 AppScan Source 追蹤。

- **來源**：來源是對程式的輸入，例如：檔案、Servlet 要求、主控台輸入或 Socket。對於大部分的輸入來源而言，傳回的資料內容和長度都是沒有限制的。如果沒有對輸入進行檢查，則會將其視為受到污染。來源會列在任何發現項目表格的**來源**直欄中。
- **接收槽**：接收槽可以是資料能夠寫出的任何外部格式。資料庫、檔案、主控台輸出和 Socket 都是接收槽的範例。未經檢查，便將資料寫入接收槽，可能是一個嚴重的安全漏洞。
- **遺失的接收槽**：遺失的接收槽是指無法再追蹤的 API 方法。

此圖說明從根到輸入堆疊和輸出堆疊的呼叫序列。



在這個圖中：

- 空心箭頭所顯示的呼叫，沒有已知受污染的資料流。
- 實心箭頭表示有受污染的資料。虛線是返回路徑。
- 實線表示方法呼叫。

提示：

- 在「追蹤」視圖中，將滑鼠游標移到圖形中的追蹤節點上，會提供節點的相關資訊。
- 可收合視圖中的兩個左畫面（輸入/輸出堆疊畫面和資料流畫面），使圖形呼叫曲線的檢視更容易。如果要收合這些畫面，請選取**隱藏樹狀結構視圖**方向鈕。如果要顯示這些隱藏的畫面，請選取**顯示樹狀結構視圖**方向鈕。
- 請移動捲軸，來放大及聚焦在細部，或縮小以看得更多。將滑鼠指標移至縮放捲軸可提供現行縮放比例。如果要放大到最大層次，請按一下**縮放至 200%**。如果要縮小到越遠越好，請按一下**適當縮放**。

可讓您處理評量的視圖

這一節的視圖是用於以高階方式處理評量。

- 『「評量摘要」視圖』
- 第 278 頁的『「過濾器編輯器」視圖』
- 第 279 頁的『「漏洞矩陣」視圖』

「評量摘要」視圖

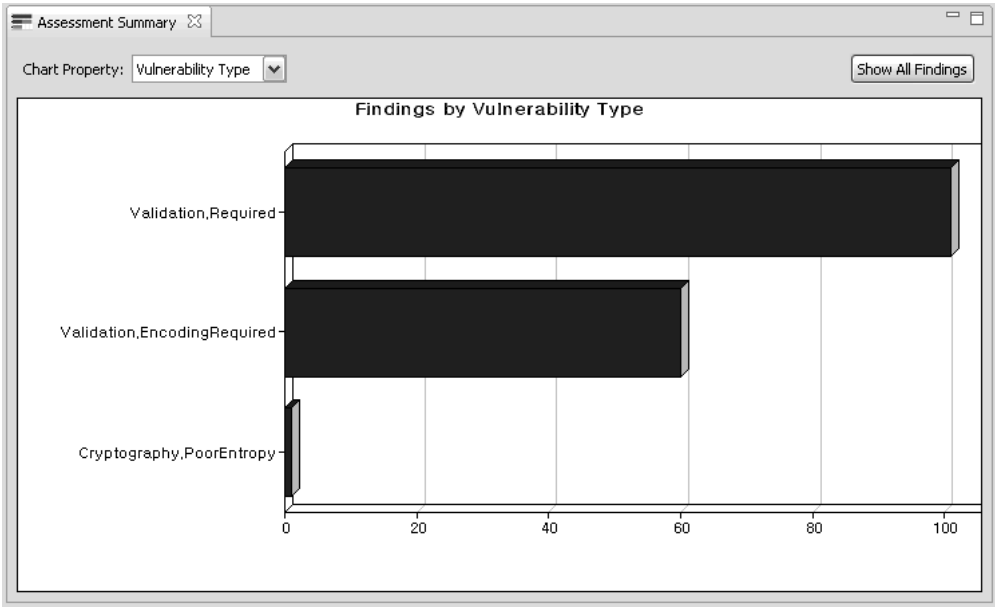
「評量摘要」視圖是已開啟之評量的長條圖圖形視圖，會顯示所選發現項目的資訊。

註：

- 「評量摘要」視圖在 macOS 上無法使用。
- 在 AppScan Source for Development（Visual Studio 外掛程式）中，這個視圖是「編輯過濾器」視窗的一部分。

您可以依圖表內容來檢視：

- **漏洞類型**：指漏洞類型，如：`Validation.Encoding` 或 `Injection.SQL`
- **API**：出現漏洞的 API 名稱
- **專案**：依專案分類的發現項目（若有多個專案）
- **檔案**：出現漏洞的個別檔案



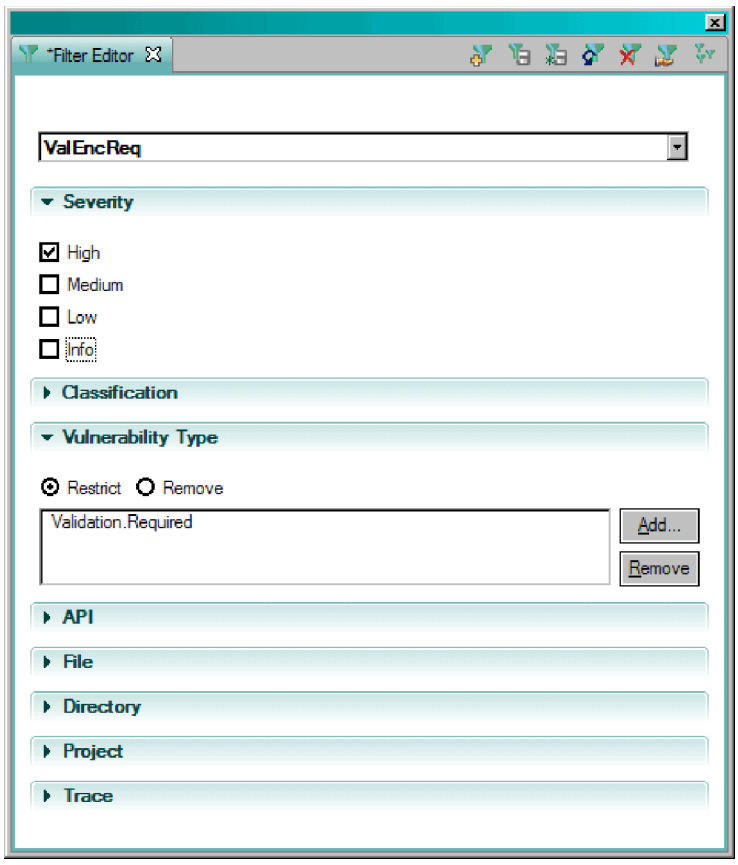
請按一下圖表來往下探查到發現項目詳細資料，然後開始分類。

提示：將滑鼠游標移到「評量摘要」視圖的直條圖上，會提供直條所代表的確切發現項目數。

「過濾器編輯器」視圖

在「過濾器編輯器」視圖中，相較於其他 AppScan Source 視圖，能夠對目前選取的過濾器進行更精細的操作。這個視圖由所有過濾準則組成。

註：在 AppScan Source for Development（Visual Studio 外掛程式）中，這個視圖是「編輯過濾器」視窗的一部分。

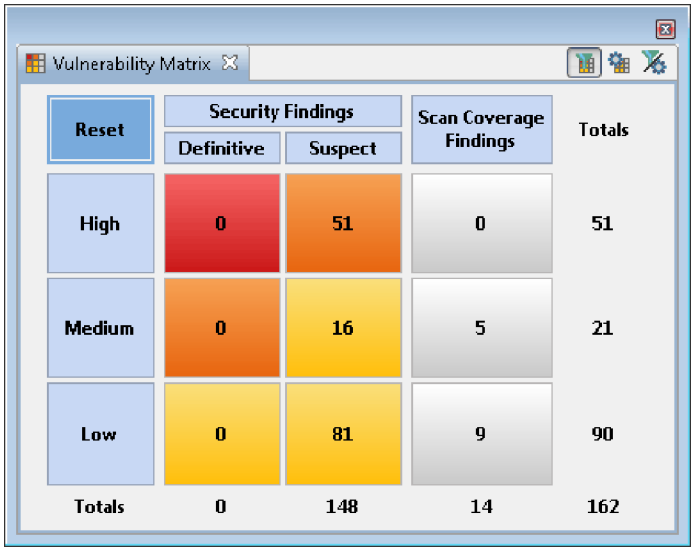


提示：在「過濾器編輯器」視圖的**追蹤**區段中，將滑鼠游標移到追蹤項目上，會提供項目的詳細資料。

「漏洞矩陣」視圖

「漏洞矩陣」視圖會顯示掃描所包含之所有應用程式的發現項目總數。修改發現項目會更新矩陣。

註：在 AppScan Source for Development（Visual Studio 外掛程式）中，這個視圖是「編輯過濾器」視窗的一部分。



安全發現項目和掃描涵蓋面發現項目是以彩色方塊呈現，指出調查或處理發現項目的優先順序：

1. 高嚴重性最後安全發現項目是紅色，將它們標示為最高優先順序。
2. 中嚴重性最後和高嚴重性可疑安全發現項目是橙色，應列為下一個處理的對象。
3. 這些矩陣項目是黃色，應列為下一個考量的對象：
   - 低嚴重性最後安全發現項目
   - 中和低嚴重性可疑安全發現項目
4. 掃描涵蓋面發現項目是以灰色方塊呈現，可列為最低優先順序。

當您按一下**漏洞矩陣**中的某個資料格、列標頭或直欄標頭時，它會更新現有的過濾器，只包含這個資料格、列或直欄中的結果。按一下**重設**，會回到具有所有發現項目的視圖。

在「漏洞矩陣」視圖中，工具列按鈕會控制彩色方塊內的數字。您可以檢視：

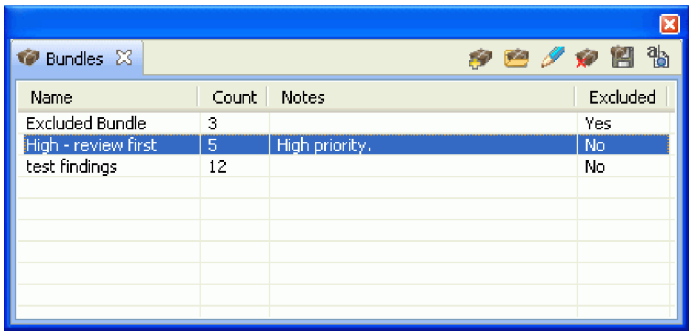
- 只限已過濾的發現項目的計數和總數
- 發現項目的計數和總數

  註：品質發現項目和以**參考資訊**嚴重性層次來分類的發現項目，不會併到「漏洞矩陣」視圖中。
- 已過濾及所有的發現項目的計數和總數

註：在「漏洞矩陣」視圖之外套用的過濾器，不會影響「漏洞矩陣」視圖。您必須選取「漏洞矩陣」視圖的**顯示過濾的發現項目計數**工具列按鈕，過濾器才會反映在「漏洞矩陣」視圖中。

「組合」視圖

在「組合」視圖中，您可以建立新的組合、新增發現項目到組合中、檢視組合和附註、重新命名或刪除組合。這個視圖會列出組合名稱、附加於組合的任何附註、組合中的發現項目數，以及是否已排除這個組合。開啟組合來查看內容之後，您可以將發現項目移到其他組合中、修改發現項目、編輯程式碼，或將組合提交給問題追蹤系統。



如需相關資訊，請參閱第 144 頁的『利用組合分類』。

「組合」視圖

「組合」視圖會顯示組合中的發現項目。組合是在 AppScan Source for Analysis 中建立的發現項目組。

如果要檢視組合中的發現項目，請在「組合」視圖中，按兩下組合名稱。組合名稱會成為「組合」視圖的標題。您也可以在「組合」視圖中，匯入組合及檢視其內容。您無法修改或刪除組合中的發現項目。

「組合」視圖類似於發現項目表格，它包含下列詳細資訊：

表 *39.* 「組合」視圖直欄

直欄	**說明**
追蹤	這個直欄中的圖示指出遺失或已知的接收槽有追蹤資料存在。
檔案	出現安全發現項目或掃描涵蓋面發現項目的程式檔名稱。發現項目中的檔案路徑是相對於掃描的專案工作目錄。
分類	發現項目的類型：**明確**或**可疑**安全發現項目 - 或**掃描涵蓋面**發現項目。 **註：**在某些情況下，**無**這項分類可用來表示既非安全發現項目也非掃描涵蓋面發現項目的分類。
嚴重性	• 高：造成資料在機密性、完整性或可用性方面的風險，以及/或造成處理資源在完整性或可用性方面的風險。高度嚴重性狀況應該優先立即補救。 • 中：造成資料安全和資源完整性的風險，但狀況不是很容易遭到攻擊。中度嚴重性狀況應該儘可能檢查及補救。 • 低：造成最低的資料安全或資源完整性風險。 • 參考資訊：發現項目本身不容易產生危害。它說明程式碼中使用的技術、架構性質或安全機制。
漏洞類型	漏洞種類，例如 `Validation.Required` 或 `Injection.SQL`。

表 39. 「組合」視圖直欄 (繼續)

直欄	說明
環境定義	漏洞週遭的程式碼片段。
呼叫方法	發出有漏洞呼叫的函數（或方法）。
CWE	ID and topic of the community-developed dictionar一般軟體弱點的社群開發字典 ID 和主題（一般弱點列舉 (CWE) 主題）。
行	程式檔中有漏洞的 API 所在的行號。
附註	任何新增到這個發現項目的附註。
問題報告 ID	問題追蹤系統中的問題報告 ID。

安裝和使用者資料檔位置

當您安裝 AppScan Source 時，使用者資料和配置檔會儲存在安裝目錄之外。

- 『預設安裝位置』
- 『預設 AppScan Source 資料目錄』
- 第 283 頁的『AppScan Source 暫存檔位置』

預設安裝位置

當安裝 AppScan Source 時，軟體會放在下列預設位置之一：

- 32 位元版本的 Microsoft Windows：

  `<SYSTEMDRIVE>:\Program Files\IBM\AppScanSource`
- 64 位元版本的 Microsoft Windows：

  `<SYSTEMDRIVE>:\Program Files (x86)\IBM\AppScanSource`
- Linux：如果您是 root 使用者，「安裝精靈」會將您的軟體安裝在 `/opt/ibm/appscansource` 中。如果您不是 root 使用者，您可以安裝 AppScan Source for Development Eclipse 外掛程式；依預設，其會安裝至 `<home_directory>/AppScan_Source`。
- macOS：`/Applications/AppScanSource.app`

重要：

- 安裝目錄名稱只能包含英文字元。不允許使用名稱包含非英文字元的資料夾。
- 如果您要在 Windows 上安裝，您必須有管理者專用權，才能安裝 AppScan Source 元件。
- 如果您要在 Linux 上安裝，您必須有 root 專用權，才能安裝 AppScan Source 伺服器元件。

預設 AppScan Source 資料目錄

AppScan Source 資料由配置、範例和日誌檔之類的項目組成。當安裝 AppScan Source 時，依預設，資料檔會放在下列位置：

- Microsoft Windows：`<SYSTEMDRIVE>:\ProgramData\IBM\AppScanSource`

  註：`ProgramData\` 是一個隱藏的資料夾，如果要查看它，您必須修改**瀏覽器**中的檢視喜好設定，來顯示隱藏的檔案和資料夾。

- Linux：/var/opt/ibm/appscansource
- macOS：/Users/Shared/AppScanSource

如果要學習如何變更 AppScan Source 資料目錄的位置，請參閱『變更 AppScan Source 資料目錄』。

AppScan Source 暫存檔位置

有些 AppScan Source 作業會導致建立暫存檔，依預設，這些暫存檔會儲存在下列位置：

- Microsoft Windows：<SYSTEMDRIVE>:\ProgramData\IBM\AppScanSource\temp

  註：ProgramData\ 是一個隱藏的資料夾，如果要查看它，您必須修改**瀏覽器**中的檢視喜好設定，來顯示隱藏的檔案和資料夾。
- Linux：/var/opt/ibm/appscansource/temp
- macOS：/Users/Shared/AppScanSource/temp

暫存檔位置一律在 AppScan Source 資料目錄的 temp 目錄中。您可以依照『變更 AppScan Source 資料目錄』中所說明，變更資料目錄來改變暫存檔位置。這會使 temp 位於您選擇的資料目錄中。

變更 AppScan Source 資料目錄

您可以變更 AppScan Source 資料目錄的位置，以便管理硬碟空間。在安裝 AppScan Source 之後，您可以遵循這個主題的步驟來變更位置。

開始之前

完成這個作業之前，請確定所有 AppScan Source 用戶端應用程式都已結束或關閉。AppScan Source 用戶端應用程式包括：

- AppScan Source for Analysis
- AppScan Source for Development（Eclipse 或 Visual Studio 外掛程式）（只在 Windows 和 Linux 上受到支援）
- AppScan Source 指令行介面 (CLI)
- AppScan Source for Automation

此外，如果您已安裝 AppScan Source for Automation，請確定自動化伺服器已經關閉：

- 在 Windows 上，停止 **IBM Security AppScan Source Automation** 服務。
- 在 Linux 上，發出這個指令：/etc/init.d/ounceautod stop
- 在 macOS 上，發出這個指令：launchctl stop com.ibm.appscan.autod

程序

1. 定義 APPSCAN_SOURCE_SHARED_DATA=<data_dir> 環境變數，其中 <data_dir> 是您要儲存 AppScan Source資料的位置。

   註：

   - <data_dir> 位置必須是一個完整的絕對路徑，而且與 AppScan Source 的安裝存在於相同的機器上。

- <data_dir> 目錄名稱只能包含英文字元。不允許使用名稱包含非英文字元的資料夾。

2. 尋找安裝 AppScan Source 時建立的預設資料目錄（請參閱第 282 頁的『預設 AppScan Source 資料目錄』，以瞭解預設資料目錄位置）。
3. 將預設資料目錄的內容複製或移動到該環境變數所指定的 <data_dir> 位置。
4. **只適用於安裝在 Linux 上的 AppScan Source for Automation**：
   a. 編輯 /etc/init.d/ounceautod 檔案。
   b. 尋找這一行，

```
su - ounce -c
'export LD_LIBRARY_PATH="/opt/IBM/AppScan_Source/bin":$LD_LIBRARY_PATH &&
cd "/opt/IBM/AppScan_Source/bin" &&
"/opt/IBM/AppScan_Source/bin/ounceautod" -s' >>
"/var/opt/ibm/appscansource/logs/ounceautod_output.log" 2>&1 &
```

   並將它取代如下：

```
su - ounce -c
'export APPSCAN_SOURCE_SHARED_DATA=<new data directory path here> &&
export LD_LIBRARY_PATH="/opt/IBM/AppScan_Source/bin":$LD_LIBRARY_PATH &&
cd "/opt/IBM/AppScan_Source/bin" &&
"/opt/IBM/AppScan_Source/bin/ounceautod" -s' >>
"<new data directory path here>/logs/ounceautod_output.log" 2>&1 &
```

   註：上述指令在同一行。
   c. 儲存 /etc/init.d/ounceautod 檔案。

下一步

如果您已安裝 AppScan Source for Automation，請啟動 自動化伺服器：

- 在 Windows 上，啟動 **IBM Security AppScan Source Automation** 服務。
- 在 Linux 上，發出這個指令：/etc/init.d/ounceautod start
- 在 macOS 上，發出這個指令：launchctl start com.ibm.appscan.autod

第 15 章 CWE 支援

「一般弱點列舉 (Common Weakness Enumeration, CWE)」是一份業界標準清單，可提供常見的軟體弱點的一般名稱。本主題列出 AppScan Source 的現行版本中支援的 CWE ID。

在掃描期間，AppScan Source 會尋找這些 CWE 清單 ID 以及它們的母項或子項 ID：

表 *40. CWE* 支援

15、16、20、73、74、77、79、88、89、90、91、95、98
105、109、112、113、116、117、120、129、130、131、134、185、190
201、209、242、250、257、264、266、267、285、287、288、295
310、311、312、319、327、331、335、345、352、359、367、382、388、390、398
400、404、407、425、434、447、470、472、477、489、497
506、507、511、517、520、521、522、523、524、525、532、538、543、544、546、547、565、569、586
601、613、615、624、643、645

名詞解釋

這個名詞解釋包括 AppScan Source 的術語和定義。

這個名詞解釋使用下列交互參照：

- 請參閱將您從術語轉介到偏好的同義字，或從字首語或縮寫轉介到定義好的完整形式。
- 另請參閱將您轉介到相關或對照的術語。

如果要檢視其他 IBM 產品的名詞解釋，請移至 www.ibm.com/software/globalization/terminology。

三劃

工作台 (workbench)
這是指 Eclipse 和 Eclipse 型工具（如 IBM Rational Application Developer）中的使用者介面和整合開發環境 (IDE)。

四劃

分類 (triage)
這是指評估發現項目，以及判斷如何加以解決的程序。

六劃

回呼 (callback)
這是執行緒通知另一個應用程式執行緒發生某事件的方式。

污染 (taint)
這是指不安全，卻能夠在程式碼中流動的資料。

七劃

攻擊 (attack)
這是指未獲授權的人，意圖危害軟體程式或網路系統作業的任何嘗試。

八劃

呼叫圖 (call graph)
這是利用線條來代表在程式各子常式間之資料流的圖形。

十一劃

問題報告 (defect)
這是一種識別工作成果中異常或缺失的變更要求類型。

堆疊 (stack)
這是記憶體內的一個區域，以後進先出 (LIFO) 原則為基礎，通常用來儲存暫存器資訊、參數值及子常式傳回位址等資訊。

接收槽 (sink)
這是資料能夠寫出的任何外部格式。資料庫、檔案、主控台輸出和 Socket 都是接收槽的範例。

掃描 (scan)
這是指 AppScan 探索及測試應用程式並提供結果的程序。

掃描規則 (scan rule)
這是在掃描期間所搜尋的型樣或正規表示式。

排除項目 (exclusion)
這是使用者可以標示及忽略的發現項目。

異常狀況 (exception)
這是表示有可疑及有潛在漏洞的狀況，需要參考其他資訊或進行調查。

組合 (assembly)
這是指 .NET Framework 應用程式中，形成部署單元、版本控制、重複使用、啟動範圍設定和安全許可權之類型和資源的集合。

組合 (bundle)
這是使用者所建立的發現項目集。在人員和應用程式之間，可以匯出和共用這些組合。

十二劃

發現項目 (finding)
這是指在程式碼中，發現安全風險的實例。 AppScan 將發現項目劃分成下列兩種類：漏洞和異常狀況。

視景 (perspective)
這是一個顯示工作台資源各個層面的視圖群組。

評量 (assessment)
這是掃描程式碼所產生的發現項目集合，可供使用者處理、儲存，以及與他人共用。

十三劃

補救 (remediation)
這是指如何修正問題的建議。

跨網站 Scripting (cross-site scripting)
這是一種強迫網站回應用戶端所提供的資料，並在使用者 Web 瀏覽器中執行的攻擊技術。

過濾器 (filter)
這是一組規則，用來定義有某些特點的發現項目。

十四劃

漏洞分析快取 (vulnerability analysis cache)

這是指在掃描原始碼期間找到的漏洞快取，可供後續的掃描使用，以縮短掃描時間。

十五劃

編碼 (encode)

這是指在電腦安全中，利用一種編碼系統，將純文字轉換成難理解的形式。

十六劃

遺失的接收槽 (lost sink)

這是指無法再追蹤的 API 方法。

十七劃

應用程式 (application)

這是指提供功能來直接支援一或多個特定商業程序的一或多個電腦程式或軟體元件。

二十一劃

屬性 (attribute)

這是應用程式的一種性質，有助於將掃描結果組織成有意義的分組，例如：按照部門或專案領導人分組。

S

Socket

這是 TCP/IP 所用的通訊控點。

V

V-密度 (V-Density)

這是可供採用一致的方式來評估應用程式漏洞的數值表示式。將漏洞與異常狀況的數目及嚴重性關聯到目前分析的應用程式或專案的大小，可以計算出「V-密度」。

X

XSS 請參閱跨網站 Scripting (cross-site scripting)。

注意事項

本資訊係針對 IBM 在美國所提供之產品與服務所開發；而在其他國家中，IBM 不見得有提供本文件所提及之各項產品、服務或功能。如需當地目前提供之產品與服務的相關資訊，請洽詢當地 IBM 業務代表。本文件在提及 IBM 產品、程式或服務時，不表示或暗示您只能使用 IBM 產品、程式或服務。只要未侵犯 IBM 的智慧財產權，任何功能相當的產品、程式或服務都可以取代 IBM 的產品、程式或服務。不過，任何非 IBM 之產品、程式或服務，使用者必須自行負責作業之評估和驗證責任。

本文件中可能包含著 IBM 所擁有之專利或專利申請案。本文件使用者並不享有前述專利之任何授權。您可以書面提出授權查詢，來函請寄到：

IBM Director of Licensing
IBM Corporation North Castle Drive
Armonk, NY 10504-1785 U.S.A.

如果是有關雙位元組字集 (DBCS) 資訊的授權查詢，請洽詢所在國的 IBM 智慧財產部門，或書面提出授權查詢，來函請寄到：

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

下列段落若與該國之法律條款抵觸，即視為不適用：

INTERNATIONAL BUSINESS MACHINES CORPORATION 僅以現狀提供本書，而不提供任何明示或默示之保證（包括但不限於未侵害他人之智慧財產權、可售性或符合特定效用的保證）。

若有些地區在某些交易上並不允許排除上述保證，則該排除無效。

本資訊中可能會包含技術上或排版印刷上的訛誤。因此，IBM 會定期修訂；並將修訂後的內容納入新版中。IBM 可能隨時改進及/或變更本出版品所描述的產品及/或程式，但不另行通知。

這項資訊中任何對非 IBM 網站的敘述僅供參考，IBM 對該網站並不提供保證。這些網站所提供的資料不是本 IBM 產品的資料內容，如果要使用這些網站的資料，您必須自行承擔風險。

IBM 得以各種 IBM 認為適當的方式使用或散布 貴客戶提供的任何資訊，而無需對 貴客戶負責。

如果本程式之獲授權人為了 (i) 在個別建立的程式和其他程式（包含本程式）之間交換資訊，以及 (ii) 相互使用所交換的資訊，因而需要相關的資訊，請洽詢：

IBM Corporation 2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

上述資料之取得有其特殊要件，在某些情況下必須付費方得使用。

IBM 基於雙方之「IBM 客戶合約」、「國際程式授權合約」或任何同等合約之條款，提供本文件中所述之授權程式與其所有適用的授權資料。

這裡包含的效能資料是在控制環境下所決定。因此，在其他作業環境中獲得的結果可能有明顯的差異。在開發層次的系統上可能已進行一些測量，但不保證這些測量在一般可用系統上會有相同的結果。再者，部分測量可能是經由推斷來預估。實際結果可能不同。本文件的使用者應驗證適用於其特定環境的資料。

本文件所提及的非 IBM 產品資訊，取自產品的供應商、其公佈聲明或其他公開管道。IBM 並未測試過這些產品，也無法確認這些非 IBM 產品的執行效能、相容性或任何對產品的其他主張是否完全無誤。有關非 IBM 產品的性能問題，應直接洽詢產品供應商。

所有關於 IBM 未來方針或目的之聲明，隨時可能更改或撤銷，不必另行通知，且僅代表目標與主旨。

所有顯示的 IBM 價格為 IBM 的建議零售價格，而且是現行的價格，隨時可能有變動，而不另行通知。經銷商價格可能有所不同。

本資訊僅限於規劃用途。在所述之產品上市之前，此處的資訊可能隨時更動。

這項資訊含有日常商業運作所用之資料和報告範例。為求儘可能地完整說明，範例包括了個人、公司、品牌和產品的名稱。所有這些名稱全為虛構，任何與實際商場企業使用的名稱及地址類似之處，純屬巧合。

著作權：

本資訊含有原始語言之範例應用程式，用以說明各作業平台中之程式設計技術。 貴客戶可以為了研發、使用、銷售或散佈符合範例應用式所適用的作業平台之應用程式介面的應用程式，以任何形式複製、修改及散佈這些範例程式，不必向 IBM 付費。這些範例尚未經過所有情況的完整測試。因此，IBM 不保證或暗示這些程式的可靠性、有用性或功能。貴客戶可以基於研發、使用、銷售或散布符合 IBM 應用程式介面的應用程式等目的，以任何形式複製、修改及散布這些範例程式，而不必向 IBM 付費。

這些範例程式或任何衍生成果的每份複本或任何部分，都必須依照下列方式併入著作權聲明：

©（貴公司名稱）（年份）。本程式之若干部分係衍生自 IBM 公司的範例程式。© Copyright IBM Corp. _enter the year or years_. All rights reserved.

若 貴客戶正在閱讀本項資訊的電子檔，可能不會有照片和彩色說明。

商標

IBM、IBM 標誌和 ibm.com 是 International Business Machines Corp. 在世界各地多個適用範圍中所註冊的商標或註冊商標。其他產品和服務名稱可能是 IBM 或其他公

司的商標。如需目前的 IBM 商標清單，請參閱網路上的「著作權與商標資訊」，網址是：www.ibm.com/legal/copytrade.shtml。

Adobe、Acrobat、PostScript 及所有 Adobe 型商標是 Adobe Systems Incorporated 在美國及 (或) 其他國家或地區的註冊商標或商標。

IT Infrastructure Library 是 Central Computer and Telecommunications Agency（現已納入 Office of Government Commerce）的註冊商標。

Intel、Intel 標誌、Intel Inside、Intel Inside 標誌、Intel Centrino、Intel Centrino 標誌、Celeron、Intel Xeon、Intel SpeedStep、Itanium 及 Pentium 是 Intel Corporation 或其子公司在美國及（或）其他國家或地區的商標或註冊商標。

Linux 是 Linus Torvalds 在美國及/或其他國家/地區的商標。

Microsoft、Windows、Windows NT 及 Windows 標誌是 Microsoft Corporation 在美國及/或其他國家或地區的商標。

ITIL 是 Office of Government Commerce 在美國 Patent and Trademark Office 註冊的註冊商標及註冊社群商標。

UNIX 是 The Open Group 在美國及其他國家或地區的註冊商標。

Java 和所有以 Java 為基礎的商標或標誌是 Oracle 及/或其子公司的商標或註冊商標。

Cell Broadband Engine 是 Sony Computer Entertainment, Inc. 在美國及/或其他國家或地區的商標，並獲其授權使用。

Linear Tape-Open、LTO、LTO 標誌、Ultrium 及 Ultrium 標誌是 HP、IBM Corp 及 Quantum 在美國及其他國家或地區的商標。

索引

索引順序以中文字，英文字，及特殊符號之次序排列。

〔一劃〕

〔三劃〕

〔四劃〕

〔五劃〕

〔六劃〕

〔七劃〕

〔八劃〕

〔九劃〕

〔十劃〕

〔十一劃〕

〔十二劃〕

〔十三劃〕

〔十四劃〕

〔十五劃〕

〔十六劃〕

〔十七劃〕

〔十八劃〕

〔二十劃〕

〔二十一劃〕

〔二十三劃〕

A

C

Printed in Taiwan