Fortify 软件安全内容

2023 更新 4
2023 年 12 月 15 日

关于 OpenText Fortify Software Security Research

Fortify Software Security Research 团队将尖端研究成果转换为助力 Fortify 产品组合（包括 OpenText™ Fortify Static Code Analyzer (SCA) 和 OpenText™ Fortify WebInspect）的安全情报。现在，Fortify 软件安全内容支持超过 33 种语言的 1,657 个漏洞类别，且涵盖的单独 API 超过一百万个。

Fortify Software Security Research (SSR) 团队荣幸地宣布，我们即将推出 Fortify Secure Coding Rulepacks（2023.4.0 英文版）、Fortify WebInspect SecureBase（可通过 SmartUpdate 获取）和 Fortify Premium Content 的更新。

Fortify Secure Coding Rulepacks [Fortify Static Code Analyzer]

这次发行的 Fortify Secure Coding Rulepacks 可以检测超过 33 种语言的 1,432 个不同的漏洞类别，且涵盖的单独 API 超过一百万个。概括来说，此版本包括以下内容：

改进了对 Python 的支持（支持的版本：3.12）

Python 是一种功能强大的通用型编程语言，具有动态类型和高效的高级数据结构。它支持多种编程范式，包括结构化编程、面向对象编程和函数式编程。此版本通过扩展对 Python 标准库 API 更改的支持，增加了对 Python 最新版本的覆盖范围。更新了以下模块的现有规则覆盖范围：

- os
- pathlib
- tomllib

改进了对 Django 的支持（支持的版本：4.2）

Django 是一个使用 Python 编写的 Web 框架，旨在促进安全、快速的 Web 开发。开发的速度和安全性是通过框架中的高级抽象来实现的，其中使用代码构造和生成来大幅减少样板代码。在此版本中，我们更新了现有的 Django 覆盖范围以支持以下版本：4.0、4.1 和 4.2。

改进的覆盖范围包括以下命名空间：*asyncio*、*django.core.cache.backends.base.BaseCache*、*django.db.models.Model* 和 *django.middleware.security.SecurityMiddleware*。此外，改进的缺陷类别覆盖范围包括：

- Header Manipulation
- Insecure Cross-Origin Opener Policy
- Resource Injection
- Setting Manipulation

PyCryptodome 和 PyCrypto（支持的版本：3.19.0）

PyCryptodome 是一个独立的 Python 程序包，它提供了各种加密算法和协议的集合。它作为 PyCrypto 库的扩展版本，维护更加积极。PyCryptodome 旨在提供广泛的加密功能，使其成为需要在 Python 应用程序中实现安全通信、数据保护和加密操作的开发人员的多功能选择。缺陷类别的初始覆盖范围包括：

- Key Management: Empty Encryption Key
- Key Management: Empty PBE Password
- Key Management: Hardcoded Encryption Key
- Key Management: Hardcoded HMAC Key
- Key Management: Hardcoded PBE Password
- Key Management: Unencrypted Private Key
- Password Management: Hardcoded Password
- Password Management: Lack of Key Derivation Function
- Password Management: Password in Comment
- Weak Cryptographic Hash
- Weak Cryptographic Hash: Hardcoded PBE Salt
- Weak Cryptographic Hash: Insecure PBE Iteration Count
- Weak Cryptographic Hash: Predictable Salt
- Weak Cryptographic Signature: Insufficient Key Size
- Weak Encryption
- Weak Encryption: Insecure Initialization Vector
- Weak Encryption: Insecure Mode of Operation
- Weak Encryption: Insufficient Key Size
- Weak Encryption: Stream Cipher
- Weak Encryption: User-Controlled Key Size

检测源自机器学习 (ML) 和人工智能 (AI) 模型的风险

随着生成式 AI 和大型语言模型 (LLM) 的使用迅速改变软件行业的解决方案空间，新的风险也随之出现。最初的 Fortify 支持涵盖使用 OpenAI API、Amazon Web Services (AWS) SageMaker 或 LangChain 的 Python 项目。支持可检测因绝对信任来自 AI/ML 模型 API 的响应而导致的缺陷，另加以下独特功能：

对 Python OpenAI API 的初始支持（支持的版本：1.3.8）

OpenAI Python 库使开发人员能够方便地访问 OpenAI REST API，以与 GPT-4 和 DALL-E 等 OpenAI 模型进行交互。OpenAI API 使应用程序能够向 OpenAI 模型发送提示并接收生成的响应以及微调现有模型。OpenAI Python 模块支持发送和接收由 *httpx* 提供支持的异步和同步请求的能力。支持包括识别模型的潜在危险输出以及以下新类别：

- Cross-Site Scripting: AI

对 Python AWS SageMaker (Boto3) 的初始支持（支持的版本：1.33.9）

AWS SageMaker 是 Amazon AWS 庞大服务体系下的一项产品。AWS SageMaker 提供了广泛的工具来支持各种 ML 项目，从训练自定义模型到设置完整 MLOps 支持的开发管道。

Amazon 的 Python SDK (Boto3) 允许与多种 AWS 产品（包括 AWS SageMaker）进行通信。支持包括识别模型的潜在危险输出以及以下新类别：

- Cross-Site Scripting: AI

对 Python LangChain 的初始支持（支持的版本：0.0.338）[1]

LangChain 是一种通用的开源编排框架，用于使用大型语言模型 (LLM) 开发应用程序。LangChain 提供的工具和 API 可以更轻松地创建 LLM 驱动的应用程序，例如聊天机器人和虚拟代理。它们可作为基于 Python 和 JavaScript 的库使用。支持包括识别模型的潜在危险输出、检测 *Path Manipulation* 以及以下新类别：

- Cross-Site Scripting: AI

.NET 8 支持（支持的版本：8.0.0）

作为 .NET 7 的继任者，.NET 8 是一个跨平台、免费、开源的开发框架，使程序员能够使用一组标准化 API 以不同的语言（例如 C# 和 VB）编写应用程序。此版本将我们的覆盖范围扩大到最新版本的 .NET，以改进对新 API 和现有 API 的缺陷的检测。

扩展的覆盖范围涵盖以下命名空间：

- System.Collections.Frozen
- System.Net.Http.Json
- System
- System.Security
- System.Text
- System.Text.Unicode
- System.Net.Http

Java 简化加密 (Jasypt)（支持的版本：1.9.3）

Java 简化加密 (Jasypt) 是一个小型 Java 库，用于执行基于密码的加密以及创建用于存储的密码摘要。它与流行的 Java 框架（如 Spring、Wicket 和 Hibernate）集成。

缺陷类别的初始覆盖范围包括：

- Insecure Randomness
- Key Management: Empty PBE Password
- Key Management: Hardcoded PBE Password
- Key Management: Null PBE Password
- Password Management: Lack of Key Derivation Function
- Privacy Violation: Heap Inspection

[1] LangChain 仍属于新近推出的。在投入生产使用之前，必须仔细考虑安全性。

- Setting Manipulation
- Weak Cryptographic Hash
- Weak Cryptographic Hash: Empty PBE Salt
- Weak Cryptographic Hash: Hardcoded PBE Salt
- Weak Cryptographic Hash: Insecure PBE Iteration Count
- Weak Cryptographic Hash: User-Controlled PBE Salt
- Weak Encryption
- Weak Encryption: Insecure Mode of Operation

ECMAScript 2023

ECMAScript 2023（也称为 ES2023 或 ES14）是 JavaScript 语言的 ECMAScript 标准的最新版本。ES2023 的主要功能包括新的数组函数，允许通过复制和从末尾搜索来更改它们。对 ES2023 的支持将所有相关 JavaScript 缺陷类别的覆盖范围扩展到最新版本的 ECMAScript 标准。

原型污染

原型污染是 JavaScript 应用程序中的一个漏洞，它使恶意用户能够绕过或影响业务逻辑，并可能运行自己的代码。

此规则包更新可检测攻击者是否可以在以下 NPM 包中污染对象的原型：

- assign-deep
- deap
- deep-extend
- defaults-deep
- dot-prop
- hoek
- lodash
- merge
- merge-deep
- merge-objects
- merge-options
- merge-recursive
- mixin-deep
- object-path
- pathval

Kubernetes 配置

Kubernetes 是一种开源容器管理解决方案，用于自动部署、扩展和管理容器化应用程序。它提供了以容器为中心的基础架构抽象，消除了对底层基础架构的依赖，实现了可移植部署，并简化了复杂分布式系统的管理。改进的缺陷类别覆盖范围包括：

- Kubernetes Misconfiguration: Improper API Server Network Access Control
- Kubernetes Misconfiguration: Improper CronJob Access Control
- Kubernetes Misconfiguration: Improper DaemonSet Access Control
- Kubernetes Misconfiguration: Improper Deployment Access Control
- Kubernetes Misconfiguration: Improper Job Access Control
- Kubernetes Misconfiguration: Improper Pod Access Control
- Kubernetes Misconfiguration: Improper RBAC Access Control
- Kubernetes Misconfiguration: Improper ReplicaSet Access Control
- Kubernetes Misconfiguration: Improper Replication Controller Access Control
- Kubernetes Misconfiguration: Improper StatefulSet Access Control
- Kubernetes Misconfiguration: Insecure Secret Transport
- Kubernetes Misconfiguration: Insufficient Kubelet Logging
- Kubernetes Misconfiguration: Scheduler System Information Leak
- Kubernetes Misconfiguration: Uncontrolled Kubelet Resource Consumption
- Kubernetes Terraform Misconfiguration: Improper Daemon Set Access Control
- Kubernetes Terraform Misconfiguration: Improper Deployment Access Control
- Kubernetes Terraform Misconfiguration: Improper Job Access Control
- Kubernetes Terraform Misconfiguration: Improper Pod Access Control
- Kubernetes Terraform Misconfiguration: Improper Pod Network Access Control
- Kubernetes Terraform Misconfiguration: Improper RBAC Access Control
- Kubernetes Terraform Misconfiguration: Improper Replication Controller Access Control
- Kubernetes Terraform Misconfiguration: Improper Stateful Set Access Control
- Kubernetes Terraform Misconfiguration: Insecure Secret Transport

DISA STIG 5.3

为了向联邦客户提供合规性方面的支持，我们增加了 Fortify Taxonomy 与美国国防信息系统局 (DISA) 应用安全与开发 STIG 版本 5.3 之间的关联。

OWASP Mobile Top 10 Risks 2023

Open Worldwide Application Security Project (OWASP) Mobile Top 10 Risks 2023 旨在提高人们对移动安全风险的认识，并教导参与移动应用程序开发和维护的人员。为了支持希望降低 Web 应用程序风险的客户，我们添加了 Fortify Taxonomy 与 OWASP Mobile Top 10 2023 初始版本的关联。

杂项勘误表

在此版本中，我们已投入大量资源来确保能够减少误报问题的数量、重构以实现一致性并提高客户审核问题的能力。此外，客户可能还会发现与以下各项相关的已报告问题发生了变化：

弃用 20.x 之前的 Fortify Static Code Analyzer 版本

正如 2023.3 发行公告中所述，这将是支持 Static Code Analyzer 20.x 之前版本的最后一个规则包版本。对于此版本，20.x 之前的 Static Code Analyzer 版本将不会加载规则包。这将需要降级规则

包或升级 Static Code Analyzer 的版本。对于将来的版本，我们将继续支持 Static Code Analyzer 最新的四个主要版本。

减少误报和其他显著的检测改进

此版本仍在继续努力改进，消除误报。客户可以期待进一步消除误报，以及与以下方面相关的其他显著改进：

- *ASP.NET Misconfiguration: Persistent Authentication* – 消除了使用表单身份验证服务的 ASP.NET 应用程序中的该误报
- *Credential Management: Hardcoded API Credentials* – 消除了与 HTTP 持有者令牌相关的密码扫描中的该误报
- *Credential Management: Hardcoded API Credentials* – 检测到 Avature API 密钥的新问题
- *Cross-Site Request Forgery* – 在使用 `Express.js` JavaScript 框架的 NodeJS 应用程序中检测到新问题
- *Cross-Site Scripting* – 在使用 `html/template` 包的 Go 应用程序中检测到新问题
- *Cross-Site Scripting: Reflected* – 消除了使用 `ListControl` 类的 ASP.NET 应用程序中的该误报
- *Denial of Service: Format String* – 与 OWASP 前 10 个类别的映射不正确
- *Insecure Transport* – 消除了与处理私有用户数据的控制器方法相关的 ASP.NET 应用程序中的该误报
- *Insecure Transport: Mail Transmission* – 消除了使用 `smtplib.SMTP` 类的 Python 应用程序中的该误报
- *Key Management: Hardcoded Encryption Key* – 消除了使用 `RSAKeyGenParameterSpec` 类的 Java 应用程序中的该误报
- *Link Injection: Missing Validation* – 消除了使用 `WKNavigationDelegate `协议的 Swift 和 Objective-C 应用程序中的该误报[2]
- *Mass Assignment: Insecure Binder Configuration* – 消除了使用 Jakarta EE API 的 Java 应用程序中的该误报
- *Password Management: Password in Configuration File* – 消除了配置文件中的该误报
- *Path Manipulation* – 在 PHP 应用程序的文件上传中检测到新问题
- *SQL Injection* – 在使用 marsdb 数据库的 NodeJS 应用程序中检测到新问题
- *SQL Injection: MyBatis Mapper* – 在 MyBatis Mapper XML 文件中检测到新问题
- *String Termination Error* – 消除了使用 `printf()` 及其变体的 C/C++ 应用程序中的该误报
- *System Information Leak: Incomplete Servlet Error Handling* – 消除了 Java 应用程序中的该误报
- *Weak Encryption: Insecure Initialization Vector* – 消除了使用 `Pycryptodome` 库的 Python 应用程序中的该误报
- *Unreleased Resource: Streams* – 在使用 `java.nio.file` API 的 Java 应用程序中识别出误报
- Visualforce 应用程序中与用户配置文件信息相关的各种数据流误报

[2] 需要 Fortify Source Code Analyzer 23.1 或更高版本

类别名称更改

当缺陷类别名称发生更改时，将先前扫描的分析结果与新扫描相合并可能导致增加/移除某些类别。

为了提高一致性，我们已对以下两个类别进行了重命名：

删除的类别	添加的类别
Azure ARM Misconfiguration: Insecure DataBricks Storage	Azure ARM Misconfiguration: Insecure Databricks Storage
Azure ARM Misconfiguration: Insecure Redis Enterprise Transport	Azure ARM Misconfiguration: Insecure Redis Transport

Fortify SecureBase [Fortify WebInspect]

Fortify SecureBase 不仅能对成千上万的漏洞进行检查，还可通过策略引导用户通过 SmartUpdate 立即获得以下更新。

漏洞支持

Access Control: 管理界面

此版本包括一项检查，用于在网关执行器端点启用、公开且不安全时检测 Spring Cloud Gateway 的不安全配置。在这种情况下，攻击者可以创建新路由并代表应用程序访问内部或敏感资产。这可能会导致云元数据密钥被盗、内部应用程序暴露或拒绝服务 (DoS) 攻击。

Expression Language Injection: Spring

Spring Cloud Gateway 版本 3.1.0、3.0.0 至 3.0.6 以及 3.0.0 之前的版本包含 CVE-2022-22947 标识的安全漏洞。当网关执行器端点处于启用、公开且不安全状态时，此漏洞允许代码注入攻击。此版本包括以下检查功能：在使用受影响的 Spring Cloud Gateway 版本的目标服务器上检测是否存在此漏洞。

Insecure Deployment: Unpatched Application

TeamCity On-Premises 服务器版本 2023.05.3 及更早版本容易出现身份验证绕过，这使未经身份验证的攻击者能够在服务器上获得远程代码执行 (RCE)。此漏洞已被标识为 CVE-2023-42793。此版本包括用于检测目标服务器上是否存在此漏洞的检查功能。

信息发现：未记录的 API

API 端点的未记录或有限的文档可能会为攻击者提供未充分测试安全漏洞的攻击面。攻击者可能会执行勘测以发现已弃用、未修补和未维护的端点，从而访问敏感信息或不安全功能。此版本包括一项检查，旨在发现可访问但未在 API 规范文档中定义的受版本控制的 API 端点。

合规性报告

DISA STIG 5.3

为了支持我们的联邦客户合规性需求，此版本包含 WebInspect 检查与最新版本的美国国防信息系统局应用安全与开发 (DISA) STIG 版本 5.3 的关联。

策略更新

DISA STIG 5.3

在 WebInspect SecureBase 的支持策略列表中，添加了一项自定义策略以纳入与 DISA STIG 5.3 相关的检查。

杂项勘误表

在此版本中，我们已投入大量资源来进一步减少误报数量，并提高客户审核问题的能力。此外，客户可能还会发现与以下各项相关的报告结果发生了变化。

Insecure Transport: SSLv3/TLS 重新协商流

TLS 1.3 不支持重新协商。此版本包括对重新协商流注入检查的改进，以减少误报并提高结果的准确性。

HTML5: Cross-Site Scripting Protection

X-XSS-Protection 标头在所有常用浏览器中均已弃用。在此版本中，我们弃用了缺失或错误配置的 X-XSS-Protection 标头检查。

Fortify Premium Content

此研究团队负责构建、扩展并维护我们的核心安全情报产品之外的各种资源。

DISA STIG 5.3 和 OWASP Mobile Top 10 2023

为了配合新关联，此版本还包含附带 DISA STIG 5.3 和 OWASP Mobile Top 10 2023 支持的 OpenText™ Fortify Software Security Center 新报告包，您可以从 Fortify 客户支持门户的 Premium Content 下载该报告包。

Fortify Taxonomy：软件安全错误

要访问 Fortify Taxonomy 站点以查看对新增类别支持的说明，请访问：https://vulncat.fortify.com。

联系 Fortify 客户支持

OpenText Fortify
https://softwaresupport.softwaregrp.com/
+1 (800) 509-1800

联系 SSR

Alexander M. Hoole
Software Security Research 高级经理
OpenText Fortify
hoole@opentext.com
+1 (650) 427-9973

Peter Blay
Software Security Research 经理
OpenText Fortify
pblay@opentext.com
+1 (669) 309-1634