# Fortify 软件安全内容

## 2024 更新 2
## 2024 年 6 月 28 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 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 1,660 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（2024.2.0 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 1,435 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 Node.js 的支持（支持的版本：21.x）**[1]

Node.js 是一个跨平台的 JavaScript 运行时环境，使开发人员能够创建服务器、Web 应用程序、命令行工具等。此版本包含对 Node.js 21.x 中以下模块的 Node.js 支持的重要更新：

- async_hooks
- buffer
- child_process
- crypto
- dgram
- dns
- fs
- http
- https
- net
- os
- path
- process
- punycode
- querystring
- stream
- string_decoder
- timers
- tls
- url
- util
- v8
- vm
- worker_threads
- zlib

这些更新改进了对以下缺陷类别的问题检测：

- Command Injection
- Dynamic Code Evaluation: Code Injection
- Header Manipulation

[1] 需要 Fortify Static Code Analyzer 24.2 或更高版本。

- Insecure Transport: Weak SSL Cipher
- Insecure Transport: Weak SSL Protocol
- Key Management: Empty Encryption Key
- Key Management: Hardcoded Encryption Key
- Path Manipulation
- Privacy Violation
- Setting Manipulation
- System Information Leak
- System Information Leak: External

此外，此版本中为 Node.js 应用程序引入了以下缺陷类别：

- DNS Spoofing
- Dynamic Code Evaluation: Script Injection
- Insecure Transport: Insufficient Diffie Hellman Strength
- Key Management: Empty HMAC Key
- Key Management: Empty PBE Password
- Key Management: Hardcoded HMAC Key
- Key Management: Hardcoded PBE Password
- 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: Predictable Salt
- Weak Cryptographic Hash: User-Controlled PBE Salt
- Weak Encryption
- Weak Encryption: Insecure Initialization Vector
- Weak Encryption: Insecure Mode of Operation
- Weak Encryption: Stream Cipher

**改进了对 Java 的支持（支持的版本：21）**

Java 21 是 Java 平台的最新长期支持 (LTS) 版本。它不仅包括对现有 API 的增强，还包括大量新功能，其中一些最重要的功能有：外部函数和内存、序列化集合、密钥封装、虚拟线程、结构化并发、未命名变量和范围值。其中一些功能仍处于预览状态，但被认为足够成熟，因此涵盖在内。更新的类别如下：

- Process Control
- Unreleased Resource
- Weak Encryption
- Weak Cryptographic Hash

此外，还支持以下新类别：

- Restricted Method
- Weak Cryptographic Signature: XML Signature Secure Validation Disabled

**改进了 MyBatis 支持（支持的版本：3.5.x）**

MyBatis 是一个 SQL 映射器，用于将关系数据库中的对象与面向对象应用程序中的对象进行耦合。该框架使用 XML 描述符或代码注释将存储过程和 SQL 语句配对，以简化开发过程和数据库通信。对 MyBatis 的支持已引入到 3.5.16 版本中。改进包括对以下缺陷类别的更新支持：

- Dynamic Code Evaluation: Unsafe Deserialization
- SQL Injection
- System Information Leak
- Unreleased Resource: Database
- Unsafe Reflection

**MyBatis-Plus 初始支持（支持的版本：3.5.x）**

MyBatis-Plus 建立在现有的 MyBatis 框架之上，通过提供原始框架之外的有用且高效的开箱即用功能来简化开发。提供对 MyBatis-Plus 3.5.x 的初始支持。提供对 *SQL Injection* 的初始类别支持。

**检测源自人工智能 (AI) 和机器学习 (ML) 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。此版本改进了使用 OpenAI API（Python 和 JavaScript）、TensorFlow (Python) 或 Anthropic Claude（Python 和 JavaScript）的项目的覆盖范围。除了以下功能之外，支持还可检测因绝对信任来自 AI/ML 模型 API 的响应而导致的缺陷：

**改进了对 OpenAI 的支持（支持的版本：1.14.x [Python]、4.33.x [JavaScript]）**

Python、TypeScript 和 JavaScript 的 OpenAI 库提供了将高级 AI 功能集成到各种应用程序中的全面工具。这些库支持一系列功能，包括自然语言处理、文本生成和会话式 AI。借助直观且用户友好的 API，开发人员可以将 OpenAI 最先进的 AI 模型无缝嵌入到他们的项目中，从而增强 Python、TypeScript 和 JavaScript 环境的交互性和智能性。改进的支持扩大了 *Cross-Site Scripting: AI* 覆盖范围，并增加了两个新的缺陷类别：

- Cross-Site Scripting: DOM AI
- Prompt Injection

**TensorFlow（支持的版本：2.16.x）**

TensorFlow 是 Google 领先的开源机器学习框架，它提供了一套强大的工具来创建和部署机器学习模型。借助内置库和预训练模型，它简化了深度学习应用程序的构建。TensorFlow 可扩展用于各种项目，涵盖从研究原型到大规模生产系统。初始覆盖范围包括对以下类别的支持：

- Path Manipulation
- Privacy Violation
- System Information Leak: Internal

此外，支持还添加了新的缺陷类别：

- Dynamic Code Evaluation: Unsafe TensorFlow Deserialization

**Anthropic Claude SDK（支持的版本：0.21.3 [Python]、0.20.5 [JavaScript]）**

Python 和 JavaScript 的 Anthropic Claude 库提供了全面的工具，可将复杂的 AI 语言模型 Claude 集成到应用程序中。初始覆盖范围包括对 *Cross-Site Scripting: AI* 的支持，并增加了两个新的缺陷类别：

- Cross-Site Scripting: DOM AI
- Prompt Injection

**改进了 Django 支持（支持的版本：5.0.x）**

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，最高可支持版本 5.0.x。

这些更新改进了对以下缺陷类别的问题检测：

- Access Control: Database
- Cookie Security: CSRF Cookie not Sent Over SSL
- Cookie Security: HTTPOnly not Set on CSRF Cookie
- Cookie Security: HTTPOnly not Set on Session Cookie
- Cookie Security: Session Cookie not Sent Over SSL
- Cross-Frame Scripting
- Cross-Site Request Forgery
- HTML5: Cross-Site Scripting Protection
- HTML5: MIME Sniffing
- HTML5: Misconfigured Content Security Policy
- HTML5: Overly Permissive Content Security Policy
- Insecure Transport
- Insecure Transport: HSTS Does Not Include Subdomains
- Insecure Transport: HSTS not Set
- Insecure Transport: Insufficient HSTS Expiration Time
- Privacy Violation: BREACH
- SQL Injection

**Paramiko 初始支持（支持的版本：3.4.x）**

Paramiko 是一个用于通过 SSH 连接机器的 Python 库。Paramiko 提供了一套功能来从开发人员那里抽象加密方法。它提供了类似于套接字编程的高级功能，并授予开发人员访问 SSH 连接的微观管理配置的低级方法的权限。初始支持涵盖以下缺陷类别：

- Command Injection
- Insecure Transport: Cipher Suite Downgrade
- Insecure Transport: Weak SSL Cipher
- Password Management: Hardcoded Password
- SSH Misconfiguration: Missing Authentication

**改进了对 PHP 的支持（支持的版本：8.3）**

PHP 是一种广泛使用的通用脚本语言，最常用于 Web 开发。此版本更新了对 PHP 的支持，最高可支持版本 8.3。具体来说，此版本包含了对扩展的改进支持：

- DOM（支持的版本：8.3）

  PHP 的 DOM 扩展允许通过使用文档对象模型对 PHP 中的 XML 和 HTML 文档进行操作。对该库的扩展支持包括对 DOM 操作的改进数据流支持，以及识别*设置操作*缺陷的额外覆盖范围。

- JSON（支持的版本：8.3）

  PHP 的 JSON 扩展允许使用根据 PHP 许可证编写和许可的 JSON 解析器。该扩展的初始支持包括对扩展功能的数据流支持。

- OpenSSL（支持的版本：8.3）

  PHP 的 OpenSSL 扩展实现了 OpenSSL 库中各种加密操作的功能。对该库的扩展支持包括对加密密钥对的改进数据流支持。

- Simdjson（支持的版本：8.3）

  PHP 的 Simdjson 扩展实现了 simdjson 项目的 PHP 特定绑定，以提供快速的 JSON 解码。初始支持包括以下针对 PHP 的新类别：

  - JSON Path Manipulation

**改进了对 iOS 的支持（支持的版本：17）**[2]

Apple 的 iOS 和 IPadOS SDK 提供了一组框架，使开发人员能够构建适用于 Apple iPhone 和 iPad 设备的移动应用程序。此版本包含对 Swift 和 Objective-C 的 iOS SDK 支持的增量更新。新规则和更新后的规则扩展了 iOS 17 中以下框架的 API 覆盖范围：

- CryptoKit

[2] iOS 17 API 需要 Xcode 15 或更高版本，而它们又需要 Fortify Static Code Analyzer 23.2 或更高版本。但是，使用 Source Code Analyzer 23.2 构建使用 iOS 17 API 的应用程序时可能会出现编译器警告。建议使用 Fortify Static Code Analyzer 24.2 或更高版本以确保有效的编译和扫描。

- Foundation
- Network
- os
- System
- SwiftUI
- UIKit

这些更新改进了对以下缺陷类别的问题检测：

- Insecure Transport
- Path Manipulation
- Privacy Violation
- Privacy Violation: Health Information
- System Information Leak: External
- System Information Leak: Internal
- Unreleased Resource: Synchronization
- Unsafe Reflection
- Weak Cryptographic Hash
- Weak Cryptographic Hash: User-Controlled Salt
- Weak Encryption: User-Controlled Key Size

**改进了 MISRA C 2012 支持**

MISRA 是一个标准组织，负责制定和维护在安全关键环境中使用的应用程序开发的各种标准，这些环境要求软件具有高完整性或高可靠性。此版本支持两个新类别，它们与 MISRA C 2012 标准中的两个强制性指导规则紧密对应：

- Undefined Behavior: File Pointer Dereference
- Undefined Behavior: File Pointer Use After Close

**密码正则表达式属性更新**

Fortify Static Code Analyzer 23.1 版中引入的密码正则表达式属性是可自定义的属性，其中包含正则表达式，用于规定 Fortify 规则如何匹配各种语言的密码标识符。在此版本中，我们扩展了 com.fortify.sca.rules.password_regex.global 属性的默认值，以识别涉及单词“secret”的密码标识符。此外，我们添加了新规则，利用密码正则表达式属性来分析动态生成的 JSON 字符串。因此，客户可以期待跨语言的以下类别的检测改进：

- Password Management: Empty Password
- Password Management: Hardcoded Password
- Password Management: Null Password
- Privacy Violation

**改进了对 Golang 的支持（支持的版本：最高可支持 1.21）**

Go（也称为 Golang）是在 Google 中创建的一种静态类型的编译型编程语言。它以简单、高效以及对并发的强大支持而知名，适合用于构建可扩展的 Web 服务、数据管道和分布式系统。此版本包括对*未释放资源*缺陷的检测，并针对使用 GORM v2 的项目引入了新的 *SQL Injection* 检测。

**WordPress API 改进（支持的版本：最高可支持 6.5）（API 数量：2）**

WordPress 应用程序编程接口 (API) 可以分为多个 API 部分/主题，每个部分/主题涵盖一组给定功能所涉及的功能及其使用。它们共同构成了所谓的 WordPress API，即整个 WordPress 项目创建的插件/主题/加载项界面。此版本增加了对识别以下 API 问题的初始支持：

- REST API
- Shortcode API

**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：

***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 MVC Bad Practices: Optional Submodel With Required Property* - ASP.NET 应用程序中的误报减少
- *Insecure Transport: Mail Transmission* - Java 应用程序中的误报减少
- *Password Management: Hardcoded Password* – JSON/YAML 文件中的误报减少
- *Unreleased Resource: Streams* - Java 应用程序中的误报减少
- *Password Management: Hardcoded Password* - 在 Python 应用程序中检测到与字典类型相关的新问题
- *Password Management: Hardcoded Password* - 在 ASP.NET 应用程序中检测到与插值字符串相关的新问题
- 删除了许多来自内置 JDK 系统属性的误报

***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，对以下三个类别进行了重命名：

| 2024 R1 类别名称 | 2024 R2 类别名称 |
| --- | --- |
| Access Control: gRPC Authentication Bypass | Access Control: gRPC Fail Open |
| AWS CloudFormation Misconfiguration: Secrets Manager Missing Customer-Managed Encryption Key | AWS CloudFormation Misconfiguration: SecretsManager Missing Customer-Managed Encryption Key |
| AWS Terraform Misconfiguration: Secrets Manager Missing Customer-Managed Encryption Key | AWS Terraform Misconfiguration: SecretsManager Missing Customer-Managed Encryption Key |

### *DISA 控制相关标识符 (CCI) 版本 2*

国防信息系统局 (DISA) CCI 是一份文件，它通过提供一组标准标识符和单一可操作的声明来弥合高级和低级网络安全指导规则之间的差距。DISA 应用程序安全和开发 STIG 与 DISA CCI 紧密映射，其中单个 STIG 控制可应用于一个或多个 CCI。此版本使映射的 CCI 与之前版本中针对 Fortify 分类法的 STIG 映射的最新更新保持一致。

### *NIST 特殊出版物 800-53 修订版 4 和 5*

美国国家标准与技术研究所 (NIST) 特别出版物 800-53 是一份文件，它提供了信息系统的安全和隐私控制目录，整个网络安全领域可以利用它来提供关于如何保护系统的指导。NIST 特别出版物 800-53 与 DISA CCI 紧密映射，其中单个 CCI 可应用于一个或多个 NIST 800-53 控制。此版本使映射的 NIST 800-53 控制与针对 Fortify 分类法的 DISA CCI 映射的最新更新保持一致。

### *OWASP Mobile Top 10 2023*

正如之前所宣布的，在此版本的 Fortify 软件安全内容中，OWASP Mobile Top 10 2023 映射现已被弃用，仅保留更新后的 OWASP Mobile Top 10 2024。

## 使软件安全内容版本与 OpenText 版本控制保持一致

下一版本将包含安全内容版本控制的更改。这将是遵循“2024 Update 2”命名约定的最后一个 OpenText Fortify 安全内容更新版本。为了符合 OpenText 版本控制标准，每年每季度计划发布一个版本，并根据年份和季度进行编号 - 因此，OpenText™ Fortify™ 软件安全内容的下一个版本将是 24.4，表示将在 2024 年第 4 季度的第一个月发布。

# Fortify SecureBase [Fortify WebInspect]

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

### 漏洞支持

#### *Denial of Service: GraphQL*

API 的 GraphQL 查询语言提供了查询现有数据的运行时。GraphQL 架构是由数据对象、其字段和类型以及它们与其他数据对象的关系组成的模型。不同数据对象之间的引用会形成循环。攻击者可以通过制作恶意嵌套且昂贵的循环查询来触发过度的 CPU 和内存使用，从而导致拒绝服务 (DoS)。此版本包括一项检查功能，用于检测 GraphQL 架构中的循环引用。

#### *Access Control: Authorization Bypass (CVE-2024-27198)*

CVE-2024-27198 已被确定为 JetBrains 软件中的一个严重漏洞，会带来重大的安全威胁。此漏洞凸显了与不充分的身份验证机制相关的风险，它可能允许未经身份验证的攻击者获得对受影响系统的管理控制权。最新版本包括一项检查功能，用于检测目标服务器上是否存在此漏洞。

#### *Directory Traversal (CVE-2024-27199)*

2023.11.4 之前的 JetBrains TeamCity On-Premises 服务器版本容易受到路径遍历缺陷的影响，该漏洞编号为 CVE-2024-27199。攻击者可以利用此缺陷绕过身份验证控制，严重威胁系统完整性和机密性。最新版本包括一项检查功能，用于检测目标服务器上是否存在此漏洞。

#### *Dynamic Code Evaluation: Unsafe Deserialization (CVE-2023-26360)*

Adobe ColdFusion 版本 2018 Update 15 及更早版本和 2021 Update 5 及更早版本受到 Dynamic Code Evaluation 漏洞的影响，该漏洞编号为 CVE-2023-26360。该漏洞可能导致在当前用户的上下文中执行任意代码。该漏洞问题的利用不需要用户交互。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。

#### *Insecure Deployment: Unpatched Application (CVE-2024-32962)*

CVE-2024-32962 是与 xml-crypto（Node.js 的 XML 数字签名和加密库）相关的一个严重漏洞。该漏洞是在 4.0.0 版中引入的，已在 6.0.0 版中得到解决。该漏洞出现的原因是，在受影响的版本中，默认配置不检查签名者的授权。攻击者可以通过修改 XML 文档、将现有签名替换为使用恶意私钥生成的签名并将相应的证书附加到 <KeyInfo/> 元素来利用该漏洞。此版本包括一项检查功能，用于检测在使用受影响的 xml-crypto 版本的目标服务器上是否存在此漏洞。

### 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 Deployment: OpenSSL

此版本包括对 OpenSSL ChangeCipherSpec Man-in-the-Middle (MitM) 检查的改进，以减少误报并提高结果的准确性。

## 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。

### 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