Software Security Research 发行公告

Fortify 软件安全内容

2022 更新 3

2022 年 9 月 30 日

关于 CyberRes 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 Fortify Static Code Analyzer (SCA) 和 Fortify WebInspect）的安全情报。现在，Fortify 软件安全内容支持 30 种语言的 1,244 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（2022.3.0 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 30 种编程语言的 1,024 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：

ASP.NET Core 更新（支持的版本：6.0）[1]

在 Model-View-Controller (MVC) 模式中，视图是使用嵌入在 Razor 标记中的 C# 编程语言的 *.cshtml* 文件。Razor 标记是一种代码，可与 HTML 标记交互以生成发送至客户端的网页。视图负责处理应用程序的数据表示和用户交互。Fortify Static Code Analyzer 22.2.0 及更高版本中的规则现在支持查找视图中的问题。包括支持查找以下缺陷类别：

- ASP.NET MVC Bad Practices: Form Without AntiForgery Token
- Cross-Site Scripting: Inter-Component Communication (Cloud)
- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation
- Cross-Site Scripting: Reflected
- Open Redirect
- Privacy Violation
- System Information Leak

Entity Framework Core（支持的版本：6.0）

Entity Framework (EF) Core 是一种用于 .NET 应用程序的开源数据访问技术。EF Core 允许开发人员将 .NET 对象映射到数据库架构，并通过标准 API 和 LINQ 查询来调用数据库操作。包括支持查找以下缺陷类别：

- Access Control: Database
- ASP.NET Bad Practices: Leftover Debug Code
- Connection String Parameter Pollution
- Insecure Transport: Database
- Password Management: Hardcoded Password
- Setting Manipulation
- SQL injection
- System Information Leak: Overly Broad SQL Logging

[1] 需要使用 Fortify Static Code Analyzer 22.2.0 或更高版本。

GitHub Actions

GitHub Actions 是一个持续集成和持续交付 (CI/CD) 平台，可自动构建、测试和部署管道。该平台最近已暴露出缺陷，可导致各种系统面临命令注入攻击向量的威胁。此版本可以检测这一命令注入缺陷的以下常见实例类别：

- Command Injection: GitHub Actions

React（支持的版本：18.2）[2]

React 或 ReactJS 是一个开源 JavaScript 库，用于构建基于组件的用户界面。虽然此版本不支持新的缺陷类别，但覆盖范围已经过重构，可使 React 更加准确并减少误报。

React Native（支持的版本：0.70）[2]

React Native 是一个开源 UI 框架，允许使用 JavaScript 和 JSX 开发多平台用户界面。React Native 支持开发人员编写由目标平台原生渲染 API 进行渲染的移动应用程序，以打造顺畅且一致的用户体验。除了 React 支持的缺陷类别外，React Native 还增加了对以下缺陷类别的支持：

- Open Redirect
- Privacy Violation
- System Information Leak: Internal

React Native Async Storage（支持的版本：1.17）[2]

Async Storage 基于社区 *react-native-async-storage 项目*，是用于 React Native 的未加密、异步键值存储库。Async Storage 在特定于原生 iOS 和 Android 平台的存储机制之上提供抽象。支持通过 Async Storage 启用数据流，并报告特定于现有 JavaScript 及平台/库的缺陷类别。

密码扫描改进

密码扫描是指在各种源代码和配置文件中查找密码。Fortify Static Code Analyzer 的密码扫描覆盖范围涉及所有文件类型，因此无论采用哪种代码语言，均可找到特定的密码。我们增加了对以下密码的支持并将其报告为 *Password Management: Hardcoded Password* 或 *Credential Management: Hardcoded API Credentials*:

- HTTP Basic 身份验证令牌
- JWT (JSON Web Tokens)
- NPM (Node Package Manager) 访问令牌
- Postman API 密钥
- PyPI API 令牌

---

[2] 需要 Fortify Static Code Analyzer 22.2.0 或更高版本。

对 Java 和 Go 的初始 gRPC 支持（支持的版本：1.49.0）

Google Remote Procedure Call (gRPC) 是一个支持多环境、多语言的现代开源高性能 RPC 框架。gRPC 可连接各种服务，并支持负载平衡、跟踪和身份验证。与传统的 JSON-over-HTTP 不同，gRPC 基于 HTTP2，通常对消息使用二进制 Protocol Buffers (protobuf) 格式。对于 gRPC 项目，用户应在 Fortify Static Code Analyzer 的转换阶段添加根据 .proto 文件定义生成的代码。

我们增加了对 Go gRPC v1.49.0 的支持，以涵盖以下缺陷类别：

- Header Manipulation
- Privacy Violation
- System Information Leak: External

我们增加了对 Java gRPC v1.49.0 的支持，以涵盖以下缺陷类别：

- Denial of Service
- gRPC Metadata Manipulation
- Insecure Transport
- Insecure Transport: gRPC Server Credentials
- Insecure Transport: gRPC Channel Credentials
- Privacy Violation
- Resource Injection
- System Information Leak: External

初始 Flask 支持（支持的版本：2.2.x）

Flask 是一个使用 Python 编写的 Web 框架。Flask 最初是 *Werkzeug* 和 *Jinja* 库的包装器，现已成为最流行的 Python Web 应用程序框架之一。此版本仅包含对 Flask Response 对象的支持，以补充我们对 Python 提供的 Google Cloud Functions 支持。
包括支持查找以下缺陷类别：

- Cookie Security: Cookie not Sent Over SSL
- Cookie Security: HTTPOnly not Set
- Cookie Security: Missing SameSite Attribute
- Cookie Security: Overly Broad Domain
- Cookie Security: Overly Broad Path
- Cookie Security: Overly Permissive SameSite Attribute
- Cookie Security: Persistent Cookie
- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation
- Cross-Site Scripting: Reflected
- Header Manipulation
- Header Manipulation: Cookies
- HTML5: Misconfigured Content Security Policy
- HTML5: Overly Permissive Content Security Policy
- HTML5: Overly Permissive CORS Policy
- HTML5: Unenforced Content Security Policy
- Open Redirect

- Privacy Violation
- System Information Leak: External

Google Cloud Functions（支持的版本：403.0.0）

Google Cloud Functions 是用于构建和连接云服务的无服务器执行环境。它可以执行代码以响应预定义事件，例如 API 调用、数据库事务、将文件上载到云存储或有关发布/订阅主题的传入消息。
Cloud Functions 提供两个产品版本：Cloud Functions（第 1 代，原始版本）和 Cloud Functions（第 2 代，基于 *Cloud Run* 和 *Eventarc* 构建的新版本，可提供增强的功能集）。此版本包括对 Python 中 Google Cloud Functions 的支持并升级了对 Java 中 Google Cloud Functions 的支持。

Python 支持的缺陷类别包括 Flask API 支持的缺陷类别，以及以下类别：

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

对于 Python Google Cloud Functions，用户应添加 JSON 或 YAML 云构建文件。或者，用户可以在扫描时设置以下属性：

- *com.fortify.sca.rules.GCPFunctionName* 应设置为函数名称。
- 如果触发器类型为 HTTP，则 *com.fortify.sca.rules.GCPHttpTrigger* 应设置为 `true`；对于其他触发器类型，则应设置为 `false`。

更新对第 2 代 Java Google Cloud Functions 的规则支持后，可识别源自 CloudEvents 请求的危险输入源。

初始 Apollo Server 支持（支持的版本：3.6.8）

Apollo Server 是一个在 JavaScript 应用程序中使用的开源 GraphQL 服务器，用于构建 GraphQL API。此版本增加了对 Apollo Server 的初始 GraphQL 服务器支持，包括检测使用 Apollo Server 开发的 GraphQL API 中的以下缺陷类别：

- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation
- Cross-Site Scripting: Reflected
- GraphQL Bad Practices: Introspection Enabled
- Privacy Violation
- System Information Leak: External

基础设施即代码 (IaC)

IaC 是通过代码而非各种手动过程来管理并配置计算机资源的过程。支持的技术包括用于部署到 GCP、OpenAPI 规范和 MuleSoft 的 Terraform 配置。与这些服务的配置相关的常见问题现已报告给开发人员。

Google Cloud Platform (GCP) Terraform 配置

Terraform 是一种开源 IaC 工具，用于构建、更改云基础设施以及对其进行版本控制。它使用自己的声明性语言，称为 HashiCorp Configuration Language (HCL)。云基础设施被编码入配置文件以描述所需状态。Terraform 提供程序支持配置并管理 GCP 基础设施。此版本涵盖了 GCP Terraform 配置的以下缺陷类别：

- GCP Terraform Misconfiguration: Cloud SQL Database Publicly Accessible
- GCP Terraform Misconfiguration: Cloud Storage Bucket Uniform Access Disabled
- GCP Terraform Misconfiguration: Compute Engine IP Forwarding Enabled
- GCP Terraform Misconfiguration: Compute Engine Serial Console Enabled
- GCP Terraform Misconfiguration: Compute Engine Shielded VM Option Disabled
- GCP Terraform Misconfiguration: GKE Cluster Node Auto-Repair Disabled
- GCP Terraform Misconfiguration: GKE Cluster Publicly Accessible
- GCP Terraform Misconfiguration: Overly Permissive Role
- GCP Terraform Misconfiguration: Permissive Firewall

OpenAPI 规范

OpenAPI 规范为 HTTP API 定义了与编程语言无关的标准描述。符合 OpenAPI 规范的 OpenAPI 文档可以用 JSON 或 YAML 格式进行编写。该标准无需访问实现、文档或通过网络检查，即可定义服务的各项功能。此版本涵盖了 OpenAPI 配置的以下缺陷类别：

- OpenAPI Misconfiguration: Credential Leakage
- OpenAPI Misconfiguration: Empty Global Security Requirement
- OpenAPI Misconfiguration: Empty Operation Security Requirement
- OpenAPI Misconfiguration: Insecure Transport
- OpenAPI Misconfiguration: Missing Error Handling
- OpenAPI Misconfiguration: Missing Global Security Requirement
- OpenAPI Misconfiguration: Missing Operation Security Requirement
- OpenAPI Misconfiguration: Missing Security Schemes
- OpenAPI Misconfiguration: Optional Global Security Requirement
- OpenAPI Misconfiguration: Optional Operation Security Requirement
- OpenAPI Misconfiguration: Weak Authentication

Mule

Mule Runtime（通常简称为 Mule）是 MuleSoft 提供的企业服务总线和集成框架。Mule 支持现有系统的集成，例如 Web Services、HTTP、Java 数据库连接 (JDBC) 等。Mule 可充当企业网络内或跨互联网的应用程序之间的传输系统，允许不同的应用程序之间相互通信。此版本涵盖了 Mule 配置的以下缺陷类别：

- Mule Misconfiguration: Hardcoded Password
- Mule Misconfiguration: Insecure Database Transport
- Mule Misconfiguration: Insecure Transport
- Mule Misconfiguration: Server Identity Verification Disabled

2022 CWE Top 25

Common Weakness Enumeration (CWE™) Top 25 Most Dangerous Software Weaknesses (CWE Top 25) 于 2019 年引入，取代了 SANS Top 25。2022 CWE Top 25 于 6 月发布，是使用启发式公式确定的，该公式可规范化过去两年中向国家漏洞数据库 (NVD) 报告的漏洞的频率和严重程度。对于要针对 NVD 中最常报告的关键漏洞确定审核优先级的客户，为了向其提供支持，我们增加了 CyberRes Fortify Taxonomy 与 2022 CWE Top 25 之间的关联。

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：

弃用低于 Fortify Static Code Analyzer 19.x 的版本

正如您在 2021.4 版本中所观察到的，我们将继续支持 Fortify Static Code Analyzer 的最后四个主要版本。因此，这将是支持低于 Fortify Static Code Analyzer 19.x 版本的最后一个规则包版本。下次发布新版本时，低于 Fortify Static Code Analyzer 19.x 的版本将不加载最新的规则包。在此情况下，将需要对规则包进行降级或升级 Fortify Static Code Analyzer 版本。以后发布新版本时，我们将继续支持 Fortify Static Code Analyzer 的最后四个主要版本。

重命名基础设施即代码 (IaC) 缺陷类别

在检测与 IaC 相关的错误配置和不良实践方面，随着技术的不断成熟，我们的下一个安全内容版本将包括对缺陷类别子集进行的类别名称更改（2022 更新 4）。当缺陷类别名称发生更改时，将先前的扫描与新的扫描合并后的扫描结果将导致增加/移除某些类别。

重构缺陷类别的 Fortify Priority Order 元数据

随着应用程序安全域的不断成熟，对于缺陷类别对机密性、完整性和可用性的影响，我们的整体认知和理解也在不断加深。我们的下一个安全内容版本将包括对缺陷类别子集的缺陷元数据字段“准确性”和“影响”进行的更改（2022 更新 4）。当缺陷元数据字段发生更改时，未来的扫描结果中可能会存在出现在不同过滤器集文件夹中的问题（例如，严重、高、中、低）。初始更新将导致某些问题从严重程度较高的 Fortify Priority Order (FPO) 文件夹转移到严重程度较低的 FPO 文件夹。对于这一更改可能对现有的过滤器集和模板产生的影响，客户应做好准备。

误报改进

此版本仍在继续努力改进，消除误报。除了其他改进之外，客户可能还会发现以下方面的误报得到了进一步消除：

- *Cross-Site Request Forgery* – 消除了使用高于 .NET Framework 4.5.2 版本的 .NET 应用程序中出现的误报
- *JavaScript Hijacking* – 问题（见下文）

- *Key Management* – 减少了 JavaScript 扫描的误报
- *Key Management* – 减少了主要影响 SAPUI5 项目的误报
- *Key Management* – 基于对比的问题产生了大量误报，现已被消除
- *Password Management: Hardcoded/Empty/Null Password* – 防止了 C# 条件语句的误报
- *Password Management* – 减少了 NPM、Yarn 和 Bower 文件的误报
- *Privacy Violation: Autocomplete* – 减少了设置新密码时产生的误报
- *Setting Manipulation* – 减少了清除环境变量时产生的误报
- *Weak Cryptographic Signature* – 防止了 java.security 程序包中的误报
- *XML Entity Expansion Injection* – 减少了使用 JAXP 转换器的 Java 程序中的误报

删除 JavaScript Hijacking

以下类别在最新的 ECMAScript 中不再适用，已被删除：

- JavaScript Hijacking
- JavaScript Hijacking: Constructor Poisoning
- JavaScript Hijacking: Vulnerable Framework

因此，属于上述类别的所有问题都将从扫描结果中删除。

类别更改

除了消除误报之外，我们还发现了一些类别本应统一或标记错误的地方。当缺陷类别名称发生更改时，将先前的扫描与新的扫描合并后的扫描结果将导致增加/移除某些类别。

- *Insecure SSL: Android Hostname Verification Disabled* 现在报告为 *Insecure SSL: Server Identity Verification Disabled*
- 在 Dockerfile 中，*Password Management: Hardcoded Password* 问题现在报告为 *Password Management: Password in Configuration Files*
- 在 .NET 中，设置数据库连接字符串时 *Setting Manipulation* 的某些实例现在报告为 *Connection String Parameter Pollution*

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

Insecure Deployment: Unpatched Application

dotCMS 是一个内容管理系统，能够在一个集中位置创建和重用内容、图像和资产。ContentResource API 易受标识为 CVE-2022-26352 的远程代码执行 (RCE) 漏洞的影响。用于存储内容的文件名是根据多部分请求中提供的用户输入而构造的，且不会被 dotCMS 清理。

这使得攻击者能够在系统中上载任意文件，从而导致 RCE。此版本包括以下检查功能：在运行受影响的 dotCMS 版本的目标服务器上检测是否存在此漏洞。

Insecure Deployment: Unpatched Application

Apache APISIX 是一个开源 API 网关，提供负载均衡、动态上游等流量管理功能。此 API 网关易受标识为 CVE-2022-24112 的 RCE 漏洞的影响。攻击者可以通过 batch-request 插件绕过 Apache APISIX 的 IP 限制。如果 APISIX 使用默认 Admin 密钥，同时启用了 Admin API，且未分配自定义管理端口，则攻击者可以通过 batch-request 插件调用 Admin API，从而导致 RCE。此版本包括以下检查功能：在运行受影响的 Apache APISIX 版本的目标服务器上检测是否存在此漏洞。

Dynamic Code Evaluation: JNDI Reference Injection[3]

Java 命名和目录接口 (JNDI) 是一种 Java API，使客户端能够按名称发现和查找数据与对象。您可以通过不同的命名或目录服务来存储和检索相应对象，例如远程方法调用 (RMI)、通用对象请求代理体系结构 (CORBA)、轻量级目录访问协议 (LDAP) 或域名服务 (DNS)。如果攻击者能够控制 JNDI 查找操作的参数，则可以将查找指向由其控制的命名或目录服务，并返回使用远程工厂进行对象实例化的 JNDI 引用。此攻击可以在执行查找操作的目标服务器上执行任意远程代码。此版本包括以下检查功能：在目标 Web 服务器上检测是否存在此漏洞。

Dynamic Code Evaluation: Unsafe Deserialization[3]

Oracle Fusion Middleware 12.2.1.3.0 和 12.2.1.4.0 版本的 ADF Faces 组件中出现一个不安全的预授权 Java 反序列化漏洞，已标识为 CVE-2022-21445。此漏洞会影响所有依赖 ADF Faces 组件的应用程序，包括 Business Intelligence、Enterprise Manager、Identity Management、SOA Suite、WebCenter Portal、Application Testing Suite 和 Transportation Management。此问题使攻击者能够在服务器上执行任意代码、滥用应用程序逻辑或发起拒绝服务 (DoS) 攻击。此版本包括以下检查功能：在目标 Web 服务器上检测是否存在此漏洞。

合规性报告

2022 CWE Top 25

Common Weakness Enumeration (CWE™) Top 25 Most Dangerous Software Weaknesses (CWE Top 25) 于 2019 年引入，取代了 SANS Top 25。2022 CWE Top 25 于 6 月发布，是使用启发式公式确定的，该公式可规范化过去两年中向国家漏洞数据库 (NVD) 报告的漏洞的频率和严重程度。

---

[3] 需要 WebInspect 21.2.0.117 修补程序或更高版本中的 OAST 功能。

此次 SecureBase 更新所包括的检查，要么直接映射到标识为 CWE Top 25 的类别，要么映射到通过“ChildOf”关系与 Top 25 中的 CWE-ID 关联的 CWE-ID。

策略更新 2022

CWE Top 25

在受 WebInspect SecureBase 支持的策略列表中，添加了一项自定义策略以纳入与 2022 CWE Top 25 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：

Dynamic Code Evaluation: Unsafe Deserialization[4]

我们已将标识为 ID 11504 的检查修改为使用支持 OAST 功能的有效负载。改进此检查后，可减少误报并提高显示其结果的速度以及结果的准确性。

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。

2022 CWE Top 25

除新关联之外，此版本还包含附带 2022 CWE Top 25 支持的 Fortify Software Security Center 新报告包，您可从 Premium Content 下的 Fortify 客户支持门户下载该报告包。

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。如果客户要从旧站点上查找最新支持的更新，可从 Fortify 支持门户获取此更新内容。

[4] 需要使用 WebInspect 21.2.0.117 修补程序或更高版本中提供的 OAST 功能。

联系 Fortify 技术支持

CyberRes Fortify
http://softwaresupport.softwaregrp.com/
+1 (844) 260-7219

联系 SSR

Alexander M. Hoole
Software Security Research 高级经理
CyberRes Fortify
hoole@microfocus.com
+1 (650) 258-5916

Peter Blay
Software Security Research 经理
CyberRes Fortify
peter.blay@microfocus.com