Software Security Research 发行公告

Fortify 软件安全内容

2022 更新 4

2022 年 12 月 16 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 1,286 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（2022.4.0 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 30 种编程语言的 1,066 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：

Flask 更新（支持的版本：v2.2.x）

Flask 是一个使用 Python 编写的微型 Web 框架，它不需要使用现成的一组特定工具或库。它是一个轻量级且完善的框架，通常最适合中小型项目，但也能够处理相对复杂的项目，例如小型 API 和微服务。支持的类别包括：

- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation
- Cross-Site Scripting: Reflected
- JSON Injection
- Often Misused: File Upload
- Open Redirect
- Path Manipulation
- Privacy Violation
- Server-Side Template Injection
- System Information Leak: External
- System Information Leak: Internal

Swift 的 iOS SDK 更新（支持的版本：16）[1]

Apple 的 iOS 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。此版本包含针对 Swift 的 iOS SDK 支持的增量更新。新规则和更新后的规则扩展了 Swift iOS 和 iPadOS 应用程序的 iOS SDK 15 和 16 中 Foundation 框架的 API 覆盖范围。这些更新改进了对许多现有缺陷类别的问题检测，包括：

- Insecure SSL: Overly Broad Certificate Trust
- Insecure Transport: Weak SSL Protocol
- Privacy Violation
- Resource Injection
- System Information Leak

Salesforce Apex 和 Visualforce 更新（支持的版本：v55）[2]

[1] iOS SDK 16 的新规则需要 Fortify Static Code Analyzer 22.2 或更高版本。
[2] 需要 Fortify Static Code Analyzer 22.2 或更高版本。

Salesforce Apex 是用于创建 Salesforce 应用程序（例如业务事务、数据库管理、Web 服务和 Visualforce 页面）的编程语言。此更新改进了我们对数据库操作、SOAP Web 服务、REST Web 服务、核心 Apex 系统 API、加密 API 和 Visualforce 页面组件的支持。Apex 支持的新类别包括：

- Cookie Security: Cookie not Sent Over SSL
- Cookie Security: Missing SameSite Attribute
- Cookie Security: Overly Broad Path
- Cookie Security: Overly Permissive SameSite Attribute
- Cookie Security: Persistent Cookie
- Header Manipulation
- Header Manipulation: Cookies
- Insecure Transport
- Key Management: Hardcoded Encryption Key
- Log Forging (debug)
- Open Redirect
- Password Management: Empty Password
- Password Management: Hardcoded Password
- Path Manipulation
- Privacy Violation
- Server-Site Request Forgery
- System Information Leak: External
- System Information Leak: Internal
- Weak Cryptographic Hash
- Weak Encryption: Insecure Initialization Vector

此外，还对以下支持的类别进行了其他改进：

- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation
- Cross-Site Scripting: Reflected

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。SSR 已支持多种类型的密码，而 SCA 将密码扫描规则应用于所有文件类型，从而允许查找特定的密码，无论代码采用何种语言。支持范围已扩展为涵盖以下密码：

- Credential Management: Hardcoded API Credential，针对硬编码持有者令牌
- Password Management: Hardcoded Password，针对 SQL 服务器连接字符串中的硬编码密码
- Password Management: Password in Comment，针对 XML 注释中的密码[3]

Google Guava 初始覆盖范围（支持的版本：v31.1）

[3] 需要 Fortify Static Code Analyzer 22.2 或更高版本。

Guava 是来自 Google 的一组 Java 库，其中包括新的集合类型（例如 multimap 和 multiset）、不可变集合、图库以及用于并发、I/O、哈希缓存、原语、字符串等的实用程序。它广泛用于 Google 和其他公司的 Java 项目。支持的类别包括：

- Null Dereference
- Path Manipulation
- Privacy Violation
- Setting Manipulation
- System Information Leak
- Unreleased Resource
- Unsafe Reflection
- Weak Cryptographic Hash
- Weak Cryptographic Hash: User-Controlled Minimum Bits
- Weak Cryptographic Hash: User-Controlled Seed
- Weak Encryption

Hot Chocolate 初始覆盖范围（支持的版本：12.15.2）

Hot Chocolate 是一个基于 Microsoft .NET 平台构建的开源 GraphQL 服务器。Hot Chocolate 使开发人员能够为其应用程序快速创建和部署基于 GraphQL 的 API。此版本增加了对 Hot Chocolate 的初始支持，包括检测使用 Hot Chocolate 开发的 GraphQL API 中的以下缺陷类别：

- Cross-Site Scripting: Inter-Component Communication (Cloud)
- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation
- Cross-Site Scripting: Reflected
- GraphQL Bad Practices: Introspection Enabled
- Privacy Violation
- System Information Leak: External
- Trust Boundary Violation

Java 的 gRPC 扩展和 Python 的初始覆盖范围（支持的版本：1.49.1）

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。

我们扩展了对 Java gRPC 的支持，以涵盖以下附加类别：

- Access Control: gRPC Authentication Bypass
- Insecure SSL: Overly Broad Certificate Trust
- Log Forging
- Setting Manipulation
- Unreleased Resource: Streams

我们建立了对 Python gRPC 的支持，以涵盖以下类别：

- Insecure Transport
- Insecure Transport: gRPC Channel Credentials
- Insecure Transport: gRPC Server Credentials

- Privacy Violation
- System Information Leak: External

云基础设施即代码 (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。改进的支持包括用于部署到 AWS、Azure 和 GCP 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。

Amazon AWS Terraform 配置

Terraform 是一种开源的基础设施即代码工具，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 Amazon Web Services (AWS) 基础设施的配置和管理。在此版本中，我们报告了 Terraform 配置中的以下缺陷类别：

- AWS Terraform Misconfiguration: Amazon API Gateway Publicly Accessible
- AWS Terraform Misconfiguration: Amazon EBS Insecure Storage
- AWS Terraform Misconfiguration: Amazon ElastiCache Insecure Transport
- AWS Terraform Misconfiguration: Amazon MQ Publicly Accessible
- AWS Terraform Misconfiguration: Amazon Neptune Publicly Accessible
- AWS Terraform Misconfiguration: Amazon RDS Insecure Storage
- AWS Terraform Misconfiguration: Amazon RDS Proxy Insecure Transport
- AWS Terraform Misconfiguration: Amazon RDS Publicly Accessible
- AWS Terraform Misconfiguration: Amazon Redshift Publicly Accessible
- AWS Terraform Misconfiguration: Amazon SNS Insecure Storage

Microsoft Azure Terraform 配置

Terraform 是一种开源的基础设施即代码工具，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 Microsoft Azure 基础设施的配置和管理。在此版本中，我们报告了 Terraform 配置中的以下缺陷类别：

- Azure Terraform Bad Practices: AKS Cluster Missing Host-Based Encryption
- Azure Terraform Bad Practices: Azure Disk Snapshot Missing Customer-Managed Key
- Azure Terraform Bad Practices: Azure MySQL Server Missing Infrastructure Encryption
- Azure Terraform Bad Practices: Azure PostgreSQL Server Missing Infrastructure Encryption
- Azure Terraform Bad Practices: Container Registry Missing Customer-Managed Key
- Azure Terraform Bad Practices: Cosmos DB Missing Customer-Managed Key
- Azure Terraform Bad Practices: Missing Azure Storage Infrastructure Encryption
- Azure Terraform Bad Practices: Missing SQL Database Backup Encryption
- Azure Terraform Bad Practices: Scale Set Missing Host-Based Encryption
- Azure Terraform Bad Practices: Shared Image Missing Customer-Managed Key
- Azure Terraform Bad Practices: SQL Database Missing Customer-Managed Key
- Azure Terraform Bad Practices: Storage Account Missing Customer-Managed Key
- Azure Terraform Bad Practices: Storage Encryption Scope Missing Customer-Managed Key
- Azure Terraform Bad Practices: VM Missing Host-Based Encryption
- Azure Terraform Misconfiguration: AKS Cluster Missing Customer-Managed Key
- Azure Terraform Misconfiguration: Managed Disk Missing Customer-Managed Key

- Azure Terraform Misconfiguration: Missing SQL Database Encryption
- Azure Terraform Misconfiguration: VM Storage Missing Customer-Managed Key

Google Cloud Platform (GCP) Terraform 配置

Terraform 是一种开源的基础设施即代码工具，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 Google Cloud Platform 基础设施的配置和管理。在此版本中，我们报告了 Google Cloud Platform 的 Terraform 配置中的以下缺陷类别：

- GCP Terraform Bad Practices: Apigee Missing Customer-Managed Encryption Key
- GCP Terraform Bad Practices: BigQuery Missing Customer-Managed Encryption Key
- GCP Terraform Bad Practices: Cloud Bigtable Missing Customer-Managed Encryption Key
- GCP Terraform Bad Practices: Cloud Functions Missing Customer-Managed Encryption Key
- GCP Terraform Bad Practices: Cloud Spanner Missing Customer-Managed Encryption Key
- GCP Terraform Bad Practices: Filestore Missing Customer-Managed Encryption Key
- GCP Terraform Bad Practices: Pub/Sub Missing Customer-Managed Encryption Key
- GCP Terraform Bad Practices: Secret Manager Missing Customer-Managed Encryption Key
- GCP Terraform Misconfiguration: Compute Engine Missing Confidential Computing Features
- GCP Terraform Misconfiguration: Edge Cache Service Missing HTTP-to-HTTPS Redirect
- GCP Terraform Misconfiguration: Insecure App Engine Domain Transport
- GCP Terraform Misconfiguration: Insecure App Engine Transport
- GCP Terraform Misconfiguration: Insecure Cloud Function HTTP Trigger Transport
- GCP Terraform Misconfiguration: Insecure Edge Cache Service Transport
- GCP Terraform Misconfiguration: Insecure Supply Chain
- GCP Terraform Misconfiguration: URL Map Missing HTTP-to-HTTPS Redirect

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 Fortify Static Code Analyzer 19.x 之前的版本：

正如 2022.3 发行公告 [R3 发行公告的链接] 中所述，这将是支持 Fortify Static Code Analyzer 19.x 之前版本的最后一个规则包版本。对于此版本，Fortify Static Code Analyzer 19.x 之前的版本将不加载 2022.4 规则包。这将需要对规则包进行降级或将 Static Code Analyzer 升级至版本 19.x 或更高版本。对于以后的版本，我们将继续支持 Fortify Static Code Analyzer 的最后四个主要版本。

重构缺陷类别的 Fortify Priority Order 元数据

对于用户而言，能够有效地修复问题非常重要，因此我们一直致力于研究出一种机制，以在 Fortify Priority Order 模型中更客观地确定问题分类。因此，正如 2022 R3 发布公告 [R3 发布公告的链接] 中所述，我们已经开始审查所有规则涵盖的类别，并确定了一些需要更新的领域。以下 96 个类别已更改其关联的 Fortify Priority Order 元数据，因此您可能会看到问题出现在严重性相同或较低（例如，严

重、高、中、低）的存储桶中。由于 Fortify Priority Order 值及其各个组件的更改，现有过滤器可能会导致其他问题被隐藏。

- Android Bad Practices: Encryption Secret Held in Static Field
- Android Bad Practices: Use of Released Camera Resource
- Android Bad Practices: Use of Released Media Resource
- Android Bad Practices: Use of Released SQLite Resource
- ASP.NET Bad Practices: Non-Serializable Object Stored in Session
- Buffer Overflow
- Buffer Overflow: Signed Comparison
- Code Correctness: Arithmetic Operation on Boolean
- Code Correctness: Function Not Invoked
- Code Correctness: Incorrect Serializable Method Signature
- Code Correctness: Memory Free on Stack Variable
- Code Correctness: Negative Content-Length
- Code Correctness: Premature Thread Termination
- Code Correctness: readObject() Invokes Overridable Function
- Code Correctness: Readonly Collection Reference
- ColdFusion Bad Practices: Unauthorized Include
- Cookie Security: Cookie not Sent Over SSL
- Cross-Site Scripting: Handlebars Helper
- Cross-Site Scripting: Inter-Component Communication
- Cross-Site Scripting: Inter-Component Communication (Cloud)
- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Reflected
- Cross-Site Scripting: Untrusted HTML Downloads
- Dangerous Method
- Denial of Service
- Denial of Service: Parse Double
- Denial of Service: Regular Expression
- Denial of Service: Stack Exhaustion
- Denial of Service: StringBuilder
- Double Free
- Dynamic Code Evaluation: Code Injection
- Dynamic Code Evaluation: Script Injection
- File Disclosure: Django
- File Disclosure: J2EE
- File Disclosure: Spring
- File Disclosure: Spring Webflow
- File Disclosure: Struts
- Format String: Argument Number Mismatch
- Header Manipulation: Cookies
- Header Manipulation: SMTP
- J2EE Bad Practices: Non-Serializable Object Stored in Session
- Log Forging

- Null Dereference[4]
- Often Misused: Authentication
- Often Misused: Boolean.getBoolean()
- Path Manipulation: Base Path Overwriting
- Path Manipulation: Zip Entry Overwrite
- Portability Flaw: File Separator
- Portability Flaw: Locale Dependent Comparison
- Privacy Violation
- Privacy Violation: Android Internal Storage
- Privacy Violation: BREACH
- Privacy Violation: Heap Inspection
- Privacy Violation: HTTP GET
- Privacy Violation: Image
- Privacy Violation: Keyboard Caching
- Privacy Violation: Screen Caching
- Privacy Violation: Sensitive Data Accessible From iTunes
- Privacy Violation: Shoulder Surfing
- Privacy Violation: Unobfuscated Logging
- Privilege Management: Android Disable
- Privilege Management: Missing API Permission
- Privilege Management: Missing Content Provider Permission
- Privilege Management: Missing Intent Permission
- Process Control
- Query String Injection: Android Provider
- Race Condition: PHP Design Flaw
- Race Condition: Singleton Member Field
- Race Condition: Static Database Connection
- SOQL Injection
- SOSL Injection
- System Information Leak: Overly Broad SQL Logging
- System Information Leak: PHP Errors
- System Information Leak: PHP Version
- Unreleased Resource: Cursor Snarfing
- Unsafe JNI
- Unsafe JSNI
- Unsafe Mobile Code: Access Violation
- Unsafe Mobile Code: Database Access
- Unsafe Native Invoke
- Use After Free
- Weak Encryption: Insecure Initialization Vector

[4] 空指针解引用更改仅适用于 Java 和 .NET。

- Weak Encryption: Insecure Mode of Operation
- Weak Encryption: Insufficient Key Size
- Weak Encryption: Missing Required Step
- Weak Encryption: User-Controlled Key Size
- Weak XML Schema: Lax Processing
- Weak XML Schema: Type Any
- Weak XML Schema: Unbounded Occurrences
- Weak XML Schema: Undefined Namespace

React Bad Practices: Dangerously Set InnerHTML

在 React 应用程序中使用“dangerouslySetInnerHTML”现在被标记为不良做法。

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：

- *Access Control: Database* – 消除了当输入来自另一个数据库调用时 Apex 中的问题
- *ASP.NET MVC Bad Practices: Model With Required Non-Nullable Property* – 减少了使用将强制执行数据某些特征的属性时的误报
- *Dockerfile Misconfiguration: Dependency Confusion* – 减少了通过从头开始扩展创建最小图像时的误报
- *GraphQL Bad Practices: Introspection Enabled* – 减少了注册基于类的 Flask 视图时 Flask 应用程序中的误报
- *Dynamic Code Evaluation: JNDI Reference Injection* – 减少了当 Spring Boot 项目将 'log4j2.version' Maven 属性设置为不受 Log4Shell 影响的版本时的误报
- *Memory Leak* – 减少了使用 std::unique_ptr 时的误报
- *Mass Assignment: Insecure Binder Configuration* – 减少了将 JSONConverter 注释与 DataContract、DataMember 或 IgnoreDataMember 一起使用时的误报
- *Privacy Violation* – 减少了有关 .NET 中枚举值的误报
- *SQL Injection* – 减少了在 MyBatis 查询注释中使用包含 '$.' 的准备语句时的误报
- *Unreleased Resource* – 减少了捆绑集合中的资源时 C/C++ 扫描中的误报

移除了 PHP Misconfiguration: magic_quotes 类别

移除了以下三个缺陷类别，因为它们在支持的 PHP 版本中不再相关：

- PHP Misconfiguration: magic_quotes_gpc Enabled
- PHP Misconfiguration: magic_quotes_runtime Enabled
- PHP Misconfiguration: magic_quotes_sybase Enabled

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。

- *Code Correctness: Class Does Not Implement equals* 现在报告为 *Code Correctness: Class Does Not Implement Equivalence Method*
- *Code Correctness: Class Does Not Implement Equals* 现在报告为 *Code Correctness: Class Does Not Implement Equivalence Method*
- *Code Correctness: toString on Array* 现在报告为 *Code Correctness: ToString on Array*
- *Code Correctness: null Argument to equals()* 现在报告为 *Code Correctness: null Argument To Equivalence Method*
- *Code Correctness: null Argument to Equals()* 现在报告为 *Code Correctness: null Argument To Equivalence Method*

此外，重构了最新 IaC 支持的以下 24 个类别，以提高一致性。

- *Access Control: Azure Container Registry* 现在报告为 *Azure ARM Misconfiguration: Improper Container Registry Network Access Control*
- *Access Control: Azure SQL Database* 现在报告为 *Azure ARM Misconfiguration: Improper SQL Server Network Access Control*
- *Access Control: Cosmos DB* 现在报告为 *Azure ARM Misconfiguration: Improper DocumentDB Network Access Control*
- *Access Control: Kubernetes Admission Controller* 现在报告为 *Kubernetes Bad Practices: Improper Admission Controller Access Control*
- *Access Control: Kubernetes Image Authorization Bypass* 现在报告为 *Kubernetes Misconfiguration: Image Authorization Bypass*
- *Ansible Bad Practices: CloudWatch Log Group Retention Unspecified* 现在报告为 *AWS Ansible Misconfiguration: Insufficient CloudWatch Logging*
- *Ansible Bad Practices: Redshift Publicly Accessible* 现在报告为 *AWS Ansible Misconfiguration: Improper Redshift Network Access Control*
- *Ansible Bad Practices: Unrestricted AWS Lambda Principal* 现在报告为 *AWS Ansible Misconfiguration: Improper Lambda Access Control Policy*
- *Ansible Bad Practices: User-Bound AWS IAM Policy* 现在报告为 *AWS Ansible Bad Practices: Improper IAM Access Control Policy*
- *Ansible Misconfiguration: Azure Monitor Missing Administrative Events* 现在报告为 *Azure Ansible Misconfiguration: Insufficient Azure Monitor Logging*
- Azure Resource Manager Bad Practices: *Cross-Tenant Replication* 现在报告为 *Azure ARM Misconfiguration:* Improper Storage Account Network Access Control
- *Azure Resource Manager Bad Practices: Remote Debugging Enabled* 现在报告为 *Azure ARM Misconfiguration: Improper App Service Access Control*
- *Azure Resource Manager Bad Practices: SSH Password Authentication* 现在报告为 *Azure ARM Misconfiguration: Improper Compute VM Access Control*
- *Azure Resource Manager Misconfiguration: Insecure Transport* 现在报告为 *Azure ARM Misconfiguration: Insecure App Service Transport*
- *Azure Resource Manager Misconfiguration: Overly Permissive CORS Policy* 现在报告为 *Azure ARM Misconfiguration: Improper CORS Policy*
- *Azure Resource Manager Misconfiguration: Security Alert Disabled* 现在报告为 *Azure ARM Misconfiguration: Insufficient Microsoft Defender Monitoring*
- *Azure SQL Database Misconfiguration: Insufficient Logging* 现在报告为 *Azure ARM Misconfiguration: Insufficient SQL Server Monitoring*

- *Insecure Storage: Missing EC2 AMI Encryption* 现在报告为 *AWS CloudFormation Misconfiguration: Insecure EC2 AMI Storage*
- *Insecure Storage: Missing EFS Encryption* 现在报告为 *AWS CloudFormation Misconfiguration: Insecure EFS Storage*
- *Insecure Storage: Missing Kinesis Stream Encryption* 现在报告为 *AWS CloudFormation Misconfiguration: Insecure Kinesis Data Stream Storage*
- *Insecure Transport: Azure App Service* 现在报告为 *Azure Ansible Misconfiguration: Insecure App Service Transport*
- *Kubernetes Bad Practices: API Server Publicly Accessible* 现在报告为 *Azure ARM Misconfiguration: Improper AKS Network Access Control*
- *Privacy Violation: Exposed Default Value* 现在报告为 *Azure ARM Misconfiguration: Hardcoded Secret*
- *Privilege Management: Overly Permissive Role* 现在报告为 *Azure ARM Misconfiguration: Improper Custom Role Access Control Policy*

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

Server-Side Request Forgery[5]

当攻击者可以影响应用程序服务器建立的网络连接时，将发生 Server-Side Request Forgery (SSRF)。网络连接将源自应用程序服务器的内部 IP，攻击者可以利用此连接绕过网络控制，并扫描或攻击未以其他方式公开的内部资源。此版本包括一项检查功能，用于检测接受用户输入的 Web 应用程序中是否存在 SSRF 漏洞。

Expression Language Injection[6]

常用的 Apache Commons Text 库版本 1.5 至 1.9 中存在一个严重的远程代码执行漏洞，已标识为 CVE-2022-42889。默认配置可能会允许不安全的脚本评估和任意代码执行。此版本包括用于检测目标 Web 服务器上是否存在 CVE-2022-42889 漏洞的检查功能。因为此项检查会发送大量请求，所以其不包含在标准策略中。使用“所有检查”策略或自定义现有策略以包含此项检查，或者创建自定义策略来运行此项检查。

Insecure Transport: Weak SSL Cipher

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

Transport Layer Security (TLS) 和 Secure Sockets Layer (SSL) 协议提供了一种机制来帮助保护客户端和 Web 服务器之间传输的数据的真实性、机密性和完整性。例如，使用弱密码或长度不足的加密密钥可能会使攻击者破坏保护机制并窃取或修改敏感信息。此版本包含一项标识为 ID 11285 的新检查，用以标记严重性为“严重”的“传输层保护不足 - 不安全的密码”漏洞。不安全的密码套件包含多个已知漏洞，容易受到琐碎的攻击。

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: Weak SSL Cipher

此版本包含对 Insufficient Transport Layer Protection – Weak Cipher check (11716) 的改进客户应该会发现，此项检查的严重性从“严重”降低为“高”，因为针对这些弱密码的攻击非常复杂且需要大量资源。此项检查的严重性已降低，我们引入了一项新检查来识别“Insufficient Transport Layer Protection – Insecure Cipher”漏洞，并标记严重性为“严重”的问题。未来，推荐的密码将不包含不具有 Perfect Forward Secrecy (PFS) 的密码套件。

XML External Entity Injection[7]

我们已将标识为 ID 11337 的检查修改为使用支持 Out-of-band Application Security Testing (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。

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。

[7] 需要 WebInspect 21.2.0.117 修补程序或更高版本中的 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 团队高级经理
CyberRes Fortify
hoole@microfocus.com
+1 (650) 258-5916

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