Contenido de seguridad del software Fortify

Actualización 3 de 2023
viernes, 29 de septiembre de 2023

Acerca de OpenText Fortify Software Security Research

El equipo de Fortify Software Security Research transforma la investigación más avanzada en inteligencia de seguridad que impulsa la cartera de productos de Fortify, que incluye Fortify Static Code Analyzer (SCA) y Fortify WebInspect. En la actualidad, el contenido de seguridad del software Fortify admite 1.627 categorías de vulnerabilidades en 33 lenguajes y abarca más de un millón de API distintas.

Fortify Software Security Research (SSR) se complace en anunciar la disponibilidad inmediata de actualizaciones para Fortify Secure Coding Rulepacks (en inglés, versión 2023.3.0), Fortify WebInspect SecureBase (disponible mediante SmartUpdate) y Fortify Premium Content.

Fortify Secure Coding Rulepacks [Fortify Static Code Analyzer]

Con esta versión, Fortify Secure Coding Rulepacks detecta 1.403 categorías únicas de vulnerabilidades en más de 33 lenguajes y abarca más de un millón de API distintas. En resumen, esta versión incluye lo siguiente:

Soporte mejorado para Android 13 (versión compatible: 33)

La plataforma Android es una pila de software de código abierto diseñada para dispositivos móviles. Un componente principal de Android es Java API Framework, que expone las funciones de Android a los desarrolladores de aplicaciones. Esta versión amplía la detección de vulnerabilidades en aplicaciones nativas de Android escritas en Java o Kotlin que aprovechan el Java API Framework de Android. En esta versión se introducen las tres categorías siguientes de vulnerabilidades para las aplicaciones de Android:

- Intent Manipulation: Implicit Internal Intent
- Intent Manipulation: Implicit Pending Intent
- Intent Manipulation: Mutable Pending Intent

Soporte inicial para Android Jetpack (AndroidX)

Android Jetpack es un conjunto de bibliotecas, herramientas y guías que ayudan a los desarrolladores a crear aplicaciones de Android con mayor facilidad. Jetpack cubre los paquetes androidx.* y está desagregado de las API de la plataforma, lo que ayuda a facilitar la compatibilidad con versiones anteriores y permite actualizaciones más frecuentes. En esta versión, ofrecemos una cobertura inicial para este paquete de software.

La cobertura inicial para Android Jetpack es compatible con la detección de vulnerabilidades en las siguientes bibliotecas:

- androidx.appsearch (version supported: 1.1.0-alpha03)
- androidx.compose.foundation (version supported: 1.5.1)
- androidx.compose.material (version supported: 1.5.1)
- androidx.compose.material3 (version supported: 1.1.2)
- androidx.compose.ui (version supported: 1.5.1)
- androidx.core (version supported: 1.12.0)
- androidx.credentials (version supported: 1.2.0-beta04)
- androidx.datastore (version supported: 1.0.0)
- androidx.security.crypto (version supported: 1.0.0)
- androidx.sqlite (version supported: 2.3.1)

A continuación, presentamos algunas de las mejoras de cobertura de categorías:

- Access Control: Database
- Command Injection
- Denial of Service
- Denial of Service: Regular Expression
- Header Manipulation

- Insecure Transport
- Open Redirect
- Password Management: Empty Password
- Password Management: Hardcoded Password
- Path Manipulation
- Privacy Violation
- Resource Injection
- Server-Side Request Forgery
- SQL Injection
- System Information Leak
- System Information Leak: Internal

Compatibilidad con MySQL Connector/Python (versión compatible: 8.1.0)

MySQL Connector/Python es una biblioteca de software que facilita la interacción entre aplicaciones Python y bases de datos MySQL. Sirve como puente o conector entre el lenguaje de programación Python y el sistema de gestión de bases de datos MySQL, lo que permite a los desarrolladores conectar, consultar y manipular datos en bases de datos MySQL fácilmente utilizando código Python.

La cobertura de categorías mejoradas incluye lo siguiente:

- Access Control: Database
- Denial of Service
- Insecure Transport: Client Identity Verification Disabled
- Insecure Transport: Database
- Insecure Transport: Weak SSL Protocol
- Password Management
- Password Management: Empty Password
- Password Management: Hardcoded Password
- Password Management: Weak Cryptography
- Path Manipulation
- Server-Side Request Forgery
- SQL Injection

Soporte mejorado para Django (versión compatible: 3.2)

Django es un marco web escrito en Python que está diseñado para facilitar el desarrollo web rápido y seguro. La velocidad y la seguridad del desarrollo se logran gracias al alto nivel de abstracción del marco, donde se utilizan construcciones y generación de código para reducir drásticamente el código boilerplate. En esta versión, actualizamos nuestra cobertura existente de Django para admitir versiones hasta la versión 3.2.

La cobertura mejorada incluye los siguientes espacios de nombres: *Django.contrib.auth.models,* *Django.db.models,* y *Django.http.response*. Además, la cobertura mejorada de las categorías de vulnerabilidades incluye lo siguiente:

- Cookie Security: Overly Permissive SameSite Attribute
- Header Manipulation
- Password Management
- Password Management: Empty Password

- Password Management: Hardcoded Password
- Password Management: Null Password
- Password Management: Weak Cryptography
- Privacy Violation
- System Information Leak
- System Information Leak: External

Soporte inicial para Bicep (versión compatible: 0.21.1)[1]

Microsoft Bicep es un lenguaje específico de dominio (DSL) de código abierto para soluciones de infraestructura como código (IaC) desarrollado por Microsoft para simplificar y optimizar la implementación de recursos de Azure. Sirve como una capa de abstracción sobre las plantillas de Azure Resource Manager (ARM), y ofrece una forma más intuitiva y legible de definir y administrar la infraestructura de Azure. Con Bicep, los usuarios pueden escribir código conciso y legible por humanos para describir los recursos, las configuraciones y las dependencias de Azure.

La cobertura inicial de las categorías de vulnerabilidades incluye lo siguiente:

- Azure ARM Misconfiguration: Automation Missing Customer-Managed Encryption Key
- Azure ARM Misconfiguration: Batch Missing Customer-Managed Encryption Key
- Azure ARM Misconfiguration: Cognitive Services Missing Customer-Managed Encryption Key
- Azure ARM Misconfiguration: Databricks Missing Customer-Managed Encryption Key
- Azure ARM Misconfiguration: Event Hubs Missing Customer-Managed Encryption Key
- Azure ARM Misconfiguration: Hardcoded Secret
- Azure ARM Misconfiguration: HTTPS Not Required
- Azure ARM Misconfiguration: Improper AKS Network Access Control
- Azure ARM Misconfiguration: Improper App Service Access Control
- Azure ARM Misconfiguration: Improper Blob Storage Access Control
- Azure ARM Misconfiguration: Improper Compute VM Access Control
- Azure ARM Misconfiguration: Improper Container Registry Network Access Control
- Azure ARM Misconfiguration: Improper CORS Policy
- Azure ARM Misconfiguration: Improper Custom Role Access Control Policy
- Azure ARM Misconfiguration: Improper DocumentDB Network Access Control
- Azure ARM Misconfiguration: Improper KeyVault Access Control Policy
- Azure ARM Misconfiguration: Improper Security Group Network Access Control
- Azure ARM Misconfiguration: Improper SQL Server Network Access Control
- Azure ARM Misconfiguration: Improper Storage Network Access Control
- Azure ARM Misconfiguration: Insecure Active Directory Domain Service Transport
- Azure ARM Misconfiguration: Insecure App Service Transport
- Azure ARM Misconfiguration: Insecure CDN Transport
- Azure ARM Misconfiguration: Insecure Database for MySQL Storage
- Azure ARM Misconfiguration: Insecure Database for PostgreSQL Storage
- Azure ARM Misconfiguration: Insecure DataBricks Storage
- Azure ARM Misconfiguration: Insecure EventHub Storage
- Azure ARM Misconfiguration: Insecure EventHub Transport
- Azure ARM Misconfiguration: Insecure IoT Hub Transport
- Azure ARM Misconfiguration: Insecure MySQL Server Transport
- Azure ARM Misconfiguration: Insecure PostgreSQL Server Transport

[1] Requiere Fortify Static Code Analyzer 23.2.0 y posteriores. El contenido de seguridad inicial para Bicep se distribuye con Fortify Static Code Analyzer 23.2.x.

- Azure ARM Misconfiguration: Insecure Recovery Services Backup Storage
- Azure ARM Misconfiguration: Insecure Recovery Services Vaults Storage
- Azure ARM Misconfiguration: Insecure Redis Enterprise Transport
- Azure ARM Misconfiguration: Insecure Redis Transport
- Azure ARM Misconfiguration: Insecure Service Bus Storage
- Azure ARM Misconfiguration: Insecure Service Bus Transport
- Azure ARM Misconfiguration: Insecure Storage Account Storage
- Azure ARM Misconfiguration: Insecure Storage Account Transport
- Azure ARM Misconfiguration: Insufficient AKS Monitoring
- Azure ARM Misconfiguration: Insufficient Application Insights Logging
- Azure ARM Misconfiguration: Insufficient Application Insights Monitoring
- Azure ARM Misconfiguration: Insufficient Microsoft Defender Monitoring
- Azure ARM Misconfiguration: Insufficient SQL Server Logging
- Azure ARM Misconfiguration: Insufficient SQL Server Monitoring
- Azure ARM Misconfiguration: IoT Hub Missing Customer-Managed Encryption Key
- Azure ARM Misconfiguration: NetApp Missing Customer-Managed Encryption Key
- Azure ARM Misconfiguration: Public Access Allowed
- Azure ARM Misconfiguration: Service Bus Missing Customer-Managed Encryption Key
- Azure ARM Misconfiguration: Storage Account Missing Customer-Managed Encryption Key
- Azure ARM Misconfiguration: Weak App Service Authentication
- Azure ARM Misconfiguration: Weak SignalR Authentication
- Password Management: Empty Password
- Password Management: Hardcoded Password
- Privacy Violation
- Privacy Violation: Missing Secure Decorator

Soporte inicial para Solidity (versión compatible: 0.8.x)[2]

Solidity es un lenguaje de programación orientado a objetos que se utiliza para desarrollar contratos inteligentes en varios entornos blockchain descentralizados, en particular en la blockchain Ethereum. Los contratos inteligentes escritos en Solidity se ejecutan principalmente en una máquina virtual Ethereum (EVM), pero también pueden ejecutarse en otras máquinas virtuales compatibles.

La cobertura inicial de las categorías de vulnerabilidades incluye lo siguiente:

- Authorization Bypass: tx.origin
- Code Correctness: Failing Assertion
- Code Correctness: Reentrancy
- Code Correctness: Typographical Error
- Dead Code
- Denial of Service: External Call
- Dynamic Code Evaluation: Delegatecall
- Integer Overflow
- Obsolete
- Often Misused: Block Values
- Poor Style: Confusing Naming

[2] Requiere Fortify Static Code Analyzer 23.2.0 y posteriores. El contenido de seguridad inicial para Solidty se distribuye con Fortify Static Code Analyzer 23.2.x.

- Poor Style: Variable Never Used
- Solidity Bad Practices: Default Function Visibility
- Solidity Bad Practices: Ether Balance Check
- Solidity Bad Practices: Hardcoded Gas Amount
- Solidity Bad Practices: Lack of Explicit Variable Visibility
- Solidity Bad Practices: Missing Constructor
- Solidity Misconfiguration: Compiler With Known Vulnerabilities
- Solidity Misconfiguration: Floating Pragma
- Unchecked Return Value
- Uninitialized Variable

Infraestructura de la nube como código (IaC)

Infraestructura como código es el proceso de administrar y aprovisionar recursos informáticos a través de código en lugar de varios procesos manuales. La cobertura ampliada de tecnologías compatibles incluye configuraciones de Terraform para implementación en Microsoft Azure, así como configuraciones para AWS Ansible. Los problemas comunes relacionados con la configuración de los servicios mencionados ahora se notifican al desarrollador.

Configuraciones de Microsoft Azure Terraform

Terraform es una herramienta IaC de código abierto para construir, cambiar y versionar la infraestructura de la nube. Utiliza su propio lenguaje declarativo conocido como HashiCorp Configuration Language (HCL). La infraestructura de la nube está codificada en archivos de configuración para describir el estado deseado. Los proveedores de Terraform admiten la configuración y administración de la infraestructura de Microsoft Azure. La cobertura mejorada de las categorías de vulnerabilidades incluye lo siguiente para las configuraciones de Terraform:

- Azure Terraform Misconfiguration: App Service Auto Upgrade Disabled
- Azure Terraform Misconfiguration: Improper AKS Access Control
- Azure Terraform Misconfiguration: Improper AKS Network Access Control
- Azure Terraform Misconfiguration: Improper App Service Access Control
- Azure Terraform Misconfiguration: Improper Cognitive Search Network Access Control
- Azure Terraform Misconfiguration: Improper Container Registry Access Control
- Azure Terraform Misconfiguration: Improper Functions Access Control
- Azure Terraform Misconfiguration: Improper MariaDB Network Access Control
- Azure Terraform Misconfiguration: Improper MySQL Network Access Control
- Azure Terraform Misconfiguration: Improper SQL Database Network Access Control
- Azure Terraform Misconfiguration: Improper Storage Account Access Control
- Azure Terraform Misconfiguration: Improper Virtual Network Access Control
- Azure Terraform Misconfiguration: Insecure Disk Storage
- Azure Terraform Misconfiguration: Insecure PostgreSQL Storage
- Azure Terraform Misconfiguration: Insufficient AKS Monitoring
- Azure Terraform Misconfiguration: Insufficient Application Gateway Monitoring
- Azure Terraform Misconfiguration: Insufficient Defender for Cloud Monitoring
- Azure Terraform Misconfiguration: Insufficient Front Door Monitoring
- Azure Terraform Misconfiguration: Insufficient MariaDB Backup
- Azure Terraform Misconfiguration: Insufficient Monitor Logging
- Azure Terraform Misconfiguration: Insufficient Network Watcher Logging
- Azure Terraform Misconfiguration: Insufficient PostgreSQL Monitoring
- Azure Terraform Misconfiguration: Insufficient SQL Database Monitoring
- Azure Terraform Misconfiguration: Redis Cache Auto Upgrade Disabled
- Azure Terraform Misconfiguration: Reduced Virtual Network Availability

- Azure Terraform Misconfiguration: Weak App Service Authentication
- Azure Terraform Misconfiguration: Weak Functions Authentication
- Azure Terraform Misconfiguration: Weak Linux Virtual Machines Authentication
- Azure Terraform Misconfiguration: Weak Service Fabric Authentication

Configuraciones de Ansible de Amazon Web Services (AWS)
Ansible es una herramienta de automatización de código abierto que proporciona administración de configuración, implementación de aplicaciones, aprovisionamiento en la nube y orquestación de nodos en varios entornos. Ansible incluye módulos que son compatibles con la configuración y administración de Amazon Web Services (AWS). La cobertura mejorada de las categorías de vulnerabilidades incluye lo siguiente para las configuraciones de AWS Ansible:

- AWS Ansible Misconfiguration: EFS Missing Customer-Managed Encryption Key
- AWS Ansible Misconfiguration: Improper API Gateway Network Access Control
- AWS Ansible Misconfiguration: Improper ECR Access Control
- AWS Ansible Misconfiguration: Improper ECS Network Access Control
- AWS Ansible Misconfiguration: Improper S3 Access Control
- AWS Ansible Misconfiguration: Improper Stack Access Control
- AWS Ansible Misconfiguration: Insecure API Gateway Transport
- AWS Ansible Misconfiguration: Insecure CloudFront Transport
- AWS Ansible Misconfiguration: Insecure CloudTrail Storage
- AWS Ansible Misconfiguration: Insecure CodeBuild Storage
- AWS Ansible Misconfiguration: Insecure RDS Transport
- AWS Ansible Misconfiguration: Insufficient API Gateway Logging
- AWS Ansible Misconfiguration: Insufficient CloudFront Logging
- AWS Ansible Misconfiguration: Insufficient Lambda Logging
- AWS Ansible Misconfiguration: Insufficient RDS Backup
- AWS Ansible Misconfiguration: Insufficient S3 Backup
- AWS Ansible Misconfiguration: Insufficient S3 Logging
- AWS Ansible Misconfiguration: Insufficient S3 Monitoring
- AWS Ansible Misconfiguration: Insufficient Stack Monitoring
- AWS Ansible Misconfiguration: Privileged Batch Container
- AWS Ansible Misconfiguration: RDS Auto-Upgrade Disabled
- AWS Ansible Misconfiguration: RDS Missing Customer-Managed Encryption Key
- AWS Ansible Misconfiguration: Reduced CloudFront Availability
- AWS Ansible Misconfiguration: Reduced EC2 Availability
- AWS Ansible Misconfiguration: Reduced ELB Availability
- AWS Ansible Misconfiguration: Weak IAM Password Policy

Common Weakness Enumeration (CWE™) Top 25 de 2023
La Common Weakness Enumeration (CWE™) Top 25 (enumeración de las 25 principales debilidades comunes) se introdujo en 2019 y remplaza la SANS Top 25. La CWE Top 25 de 2023 se publicó en junio de 2023 y se determinó mediante una fórmula heurística que normaliza la frecuencia y la gravedad de las vulnerabilidades comunicadas a la base de datos nacional de vulnerabilidades (NVD) en los últimos dos años. Para ofrecer soporte a nuestros clientes que deseen dar prioridad en sus auditorías a las vulnerabilidades críticas más registradas en la NVD, se agregó una correlación de Fortify Taxonomy a la CWE Top 25 de 2023.

OWASP API Security Top 10 2023

La API Security Top 10 2023 del Open Worldwide Application Security Project (OWASP) proporciona una lista de los principales riesgos de seguridad que afectan a las API en 2023. Su objetivo es crear conciencia sobre las vulnerabilidades de seguridad de las API y educar a quienes participan en el desarrollo y mantenimiento de las API, como desarrolladores, diseñadores, arquitectos, administradores y/u organizaciones en general que necesitan proteger las API web.

El OWASP API Security Top 10 se centra en las vulnerabilidades que afectan a las API web y no está diseñado para usarse únicamente por sí solo, sino en combinación con otros estándares y prácticas recomendadas para capturar exhaustivamente todos los riesgos relevantes. Por ejemplo, debe usarse en combinación con OWASP Top 10 para identificar problemas relacionados con la validación de entradas, como las inyecciones. Para proporcionar soporte a los clientes que desean mitigar el riesgo de las aplicaciones web, se agregó una correlación de Fortify Taxonomy con OWASP Security Top 10 2023, que se publicó recientemente.

Puntos de referencia Center for Internet Security (CIS)

Los puntos de referencia del Centro de Seguridad de Internet (CIS) son una recopilación de recomendaciones de configuración segura desarrolladas por la comunidad que se asignan a los controles de seguridad críticos del CIS. Estas recomendaciones tienen como objetivo permitir proteger la infraestructura de la nube y demostrar el cumplimiento de los estándares de la industria. Los puntos de referencia CIS se actualizan continuamente para adaptarse al estado cambiante de la ciberseguridad para las más de 25 familias de productos de proveedores cubiertas. Las familias de productos compatibles incluyen las siguientes:

- Amazon Elastic Kubernetes Service (EKS) Benchmark v1.3.0
- Amazon Web Services Foundations Benchmark v2.0.0
- Azure Kubernetes Service (AKS) Benchmark v1.3.0
- Google Cloud Computing Platform Benchmark v2.0.0
- Google Kubernetes Engine (GKE) Benchmark v1.4.0
- Kubernetes Benchmark v1.7.1
- Microsoft Azure Foundations Benchmark v2.0.0

Smart Contract Weakness Classification (SWC)[3]

La clasificación de vulnerabilidades de contratos inteligentes (SWC) es un marco sistemático que categoriza y explica las vulnerabilidades en los contratos inteligentes. Proporciona una forma estandarizada de comprender y abordar las vulnerabilidades de estos códigos autoejecutables que se ejecutan en blockchains como Ethereum. En particular, el contenido del registro SWC no se ha actualizado de manera integral desde 2020, lo que resulta en datos incompletos, errores y omisiones importantes. Para ayudar a nuestros clientes que desean mitigar los riesgos en los contratos inteligentes, se ha añadido la correlación de Fortify Taxonomy a la versión actual de SWC.

[3] Requiere un escaneo de Fortify Static Code Analyzer 23.2.0 y posterior.

Otras erratas

En esta versión, seguimos invirtiendo recursos para garantizar la reducción del número de falsos positivos, lograr una mejor consistencia y para mejorar la capacidad de auditoría de los problemas por parte de los clientes. Los clientes también verán cambios en los problemas comunicados en relación con lo siguiente:

Obsolescencia de las versiones de Fortify Static Code Analyzer anteriores a 20.x

Como se observó con la versión 2022.4, continuamos admitiendo las últimas cuatro versiones principales de Fortify Static Code Analyzer. Por lo tanto, esta será la última versión de los paquetes de reglas compatibles con las versiones de Fortify Static Code Analyzer anteriores a la 20.x. Para la próxima versión, las versiones de Fortify Static Code Analyzer anteriores a la 20.x no cargarán los paquetes de reglas más recientes. Se deberá cambiar a una versión inferior de Rulepacks o actualizar la versión de Fortify Static Code Analyzer. En las versiones futuras, continuaremos admitiendo las últimas cuatro versiones principales de Fortify Static Code Analyzer.

Mejoras en falsos positivos

Se ha seguido trabajando con el fin de eliminar los falsos positivos en esta versión. Además de otras mejoras, los clientes pueden esperar una mayor eliminación de falsos positivos en las siguientes áreas:

- *ASP.NET MVC Bad Practices: Optional Submodel With Required Property:* se han eliminado los falsos positivos en relación con los campos virtuales en las aplicaciones ASP.NET
- *Code Correctness: Double-Checked Locking*: eliminación de falsos positivos en aplicaciones Java
- *Cross-Site Request Forgery:* eliminación de falsos positivos para formularios HTML usando `AntiForgery.GetHtml()` o `Html.AntiForgeryToken()` en aplicaciones .NET
- *Cross-Site Scripting: Persistent:* eliminación de falsos positivos relacionados con la etiqueta `cycle` en aplicaciones Django
- *Double Free:* eliminación de falsos positivos en aplicaciones C/C++ que usan `throw_error()` de la biblioteca boost
- *HTML5: Missing Content Security Policy*: eliminación de falsos positivos en aplicaciones Java
- *JSON Injection:* eliminación de falsos positivos en aplicaciones PHP
- *Mass Assignment: Insecure Binder Configuration:* eliminación de falsos positivos relacionados con los tipos de enumeración en aplicaciones .NET
- *Often Misused: File System:* eliminación de falsos positivos relacionados con `GetFullPathNameW()` y llamadas a funciones similares en C++ aplicaciones
- *Path Manipulation:* eliminación de falsos positivos en aplicaciones Java que utilizan el SDK de Amazon AWS
- *Type Mismatch: Signed to Unsigned:* eliminación de falsos positivos relacionados con valores booleanos en aplicaciones C/C++
- *Unreleased Resource:* eliminación de falsos positivos al usar `CreateFileW()` en aplicaciones C++

Cambios de categoría

Cuando se producen cambios en el nombre de la categoría de vulnerabilidad, los resultados del análisis al fusionar escaneos anteriores con nuevos escaneos darán como resultado categorías añadidas o eliminadas.

Para mejorar la coherencia, se han cambiado los nombres de las siguientes 14 categorías:

Categoría eliminada	Categoría añadida
AWS CloudFormation Misconfiguration: Insecure Elasticache Storage	AWS CloudFormation Misconfiguration: Insecure ElastiCache Storage
AWS CloudFormation Misconfiguration: Insecure Elasticache Transport	AWS CloudFormation Misconfiguration: Insecure ElastiCache Transport
AWS Terraform Misconfiguration: Elasticache Missing Customer-Managed Encryption Key	AWS Terraform Misconfiguration: ElastiCache Missing Customer-Managed Encryption Key
Azure Terraform Bad Practices: AKS Cluster Missing Host-Based Encryption	Azure Terraform Misconfiguration: AKS Cluster Missing Host-Based Encryption
Azure Terraform Bad Practices: Azure MySQL Server Missing Infrastructure Encryption	Azure Terraform Misconfiguration: MySQL Missing Infrastructure Encryption
Azure Terraform Bad Practices: Azure PostgreSQL Server Missing Infrastructure Encryption	Azure Terraform Misconfiguration: PostgreSQL Missing Infrastructure Encryption
Azure Terraform Bad Practices: Missing Azure Storage Infrastructure Encryption	Azure Terraform Misconfiguration: Storage Account Missing Infrastructure Encryption
Azure Terraform Bad Practices: Missing SQL Database Backup Encryption	Azure Terraform Misconfiguration: SQL Server Backup Missing Encryption
Azure Terraform Bad Practices: Scale Set Missing Host-Based Encryption	Azure Terraform Misconfiguration: Scale Set Missing Host-Based Encryption
Azure Terraform Bad Practices: VM Missing Host-Based Encryption	Azure Terraform Misconfiguration: VM Missing Host-Based Encryption
GCP Terraform Bad Practices: Overly Permissive Service Account	GCP Terraform Misconfiguration: Improper Compute Engine Access Control
GCP Terraform Misconfiguration: Weak Key Management	GCP Terraform Misconfiguration: Compute Engine Missing Customer-Managed Encryption Key
Kubernetes Bad Practices: Improper Admission Controller Access Control	Kubernetes Misconfiguration: Improper Admission Controller Access Control
Kubernetes Misconfiguration: Missing Service Account Admission Controller	Kubernetes Misconfiguration: Missing ServiceAccount Admission Controller

Cambios de Fortify Priority Order

Para mejorar la coherencia entre las categorías de vulnerabilidad relacionadas con la falta de claves de encriptación administradas por el cliente, el Fortify Priority Order de las siguientes 20 categorías se cambió a "bajo":

- *Azure ARM Misconfiguration: Automation Missing Customer-Managed Encryption Key*
- *Azure ARM Misconfiguration: Batch Missing Customer-Managed Encryption Key*
- *Azure ARM Misconfiguration: Cognitive Services Missing Customer-Managed Encryption Key*
- *Azure ARM Misconfiguration: Databricks Missing Customer-Managed Encryption Key*
- *Azure ARM Misconfiguration: Event Hubs Missing Customer-Managed Encryption Key*
- *Azure ARM Misconfiguration: IoT Hub Missing Customer-Managed Encryption Key*
- *Azure ARM Misconfiguration: NetApp Missing Customer-Managed Encryption Key*
- *Azure ARM Misconfiguration: Service Bus Missing Customer-Managed Encryption Key*

- *Azure ARM Misconfiguration: Storage Account Missing Customer-Managed Encryption Key*
- *Azure Terraform Misconfiguration: AKS Cluster Missing Customer-Managed Key*
- *Azure Terraform Misconfiguration: Azure Disk Snapshot Missing Customer-Managed Key*
- *Azure Terraform Misconfiguration: Container Registry Missing Customer-Managed Key*
- *Azure Terraform Misconfiguration: Cosmos DB Missing Customer-Managed Key*
- *Azure Terraform Misconfiguration: Managed Disk Missing Customer-Managed Key*
- *Azure Terraform Misconfiguration: Shared Image Missing Customer-Managed Key*
- *Azure Terraform Misconfiguration: SQL Database Missing Customer-Managed Key*
- *Azure Terraform Misconfiguration: Storage Account Missing Customer-Managed Key*
- *Azure Terraform Misconfiguration: Storage Encryption Scope Missing Customer-Managed Key*
- *Azure Terraform Misconfiguration: VM Storage Missing Customer-Managed Key*
- *GCP Terraform Misconfiguration: Compute Engine Missing Customer-Managed Encryption Key*

Fortify SecureBase [Fortify WebInspect]

Fortify SecureBase combina las comprobaciones para miles de vulnerabilidades con las directivas que guían a los usuarios en las siguientes actualizaciones disponibles inmediatamente con SmartUpdate.

Compatibilidad de vulnerabilidades

Insecure Deployment: Unpatched Application

CVE-2023-25135 identificó una vulnerabilidad de ejecución remota de código (RCE) de autorización previa en las versiones de vBulletin 5.6.0 a 5.6.8. vBulletin, un software popular para crear foros y comunidades dinámicas en línea desinfecta incorrectamente las entradas proporcionadas por el usuario para una deserialización no autenticada. Este problema permite a los atacantes ejecutar código arbitrario en el servidor, abusar de la lógica de la aplicación o montar ataques de denegación de servicio (DoS). Esta versión incluye una comprobación que permite detectar esta vulnerabilidad en los servidores de destino.

Contaminación del prototipo: lado del servidor

La contaminación del prototipo del lado del servidor ocurre cuando un atacante puede manipular el prototipo de un objeto. Esto es posible en lenguajes basados en prototipos como JavaScript, que permite modificar propiedades y métodos en el tiempo de ejecución. La gravedad del exploit depende de dónde se utiliza el objeto contaminado en la aplicación. Los ataques incluyen denegación de servicio, cambio de configuración de la aplicación y, en algunos casos, ejecución remota de código. Esta versión incluye una comprobación que permite detectar la contaminación del prototipo en las aplicaciones web.

Informes de cumplimiento

Common Weakness Enumeration (CWE™) Top 25 de 2023

La Common Weakness Enumeration (CWE™) Top 25 (enumeración de las 25 principales debilidades comunes) se introdujo en 2019 y remplaza la SANS Top 25. La CWE Top 25 de 2023 se publicó en junio y se determinó mediante una fórmula heurística que normaliza la frecuencia y la gravedad de las vulnerabilidades comunicadas a la base de datos nacional de vulnerabilidades (NVD) en los últimos dos años. La actualización de SecureBase incluye las verificaciones que realizan asignaciones de manera directa a la categoría identificada en CWE Top 25, o bien a un ID de CWE relacionado con uno de los incluidos en el Top 25 a través de una relación “ChildOf”.

OWASP API Security Top 10 2023

La API Security Top 10 2023 del Open Worldwide Application Security Project (OWASP) proporciona una lista de los principales riesgos de seguridad que afectan a las API en 2023. Su objetivo es crear conciencia sobre las vulnerabilidades de seguridad de las API y educar a quienes participan en el desarrollo y mantenimiento de las API, como desarrolladores, diseñadores, arquitectos, administradores y organizaciones en general que necesitan proteger las API web. El OWASP API Security Top 10 se centra en las vulnerabilidades que afectan a las API web y no está diseñado para ser utilizado por sí solo. En cambio, está destinado a usarse en combinación con otros estándares y prácticas recomendadas para registrar exhaustivamente todos los riesgos relevantes. Por ejemplo, utilice OWASP API Security Top 10 2023 en combinación con OWASP Top 10 para identificar problemas relacionados con la validación de entradas, como las inyecciones. Esta actualización de SecureBase incluye una nueva plantilla de informe de cumplimiento que proporciona una correlación de categorías de OWASP API Security Top 10 2023 con comprobaciones de WebInspect.

Actualizaciones de directivas

2023 CWE Top 25

Se incorporó una directiva personalizada a la lista de directivas admitidas en WebInspect SecureBase para incluir las comprobaciones pertinentes para 2023 CWE Top 25.

OWASP API Security Top 10 2023

Se incorporó una directiva personalizada a la lista de directivas admitidas en WebInspect SecureBase para incluir las comprobaciones pertinentes para OWASP API Security Top 10 2023. Esta política contiene un subconjunto de las comprobaciones de WebInspect que están disponibles, con el fin de ayudar a los clientes a ejecutar exploraciones de WebInspect específicamente orientadas al cumplimiento.

Otras erratas

En esta versión hemos invertido recursos para reducir aún más el número de falsos positivos y para mejorar la capacidad de auditar problemas por parte de los clientes. Los clientes también verán cambios en los resultados comunicados en relación con las siguientes áreas:

LDAP Injection

Esta versión incluye mejoras en la verificación de LDAP Injection para reducir los falsos positivos y mejorar la precisión de los resultados.

Discrepancia en el nombre de host del certificado SSL

El contenido del informe de verificación de discrepancia en el nombre de host del certificado SSL ahora incluye información más detallada que debería ayudar a los clientes a aplicar una solución adecuada para este problema de seguridad.

Cobertura agresiva mediante entradas de comprobaciones

Para algunas comprobaciones de WebInspect, es posible habilitar la cobertura agresiva que guía a WebInspect para enviar una lista más larga de ataques dirigidos a una gama más amplia de puntos finales. Esta versión incluye mejoras en estas comprobaciones, que permiten a los clientes configurar la cobertura agresiva cambiando las entradas de comprobación en lugar de agregar comprobaciones independientes a la política de análisis. Las comprobaciones que tienen capacidades de cobertura agresiva incluyen los siguientes: *Log4Shell*, *JNDI Reference Injection*, *Server-Side Request Forgery*, *OS Command Injection*, y *Server-Side Prototype Pollution*. Las comprobaciones con cobertura agresiva habilitada proporcionan un escaneo más preciso; sin embargo, es importante tener en cuenta que la cantidad de solicitudes y el tiempo de escaneo pueden aumentar drásticamente. Por lo tanto, Fortify recomienda encarecidamente que ejecute comprobaciones con la cobertura agresiva habilitada en una política independiente sin otras comprobaciones.

Web Server Misconfiguration: Archivo desprotegido

Esta versión incluye una corrección de errores menores para mejorar la detección de archivos de configuración relacionados con Java.

Fortify Premium Content

El equipo de investigación crea, amplía y mantiene diversos recursos independientes de nuestros principales productos de inteligencia de seguridad.

2023 CWE Top 25

Para acompañar las nuevas correlaciones, esta versión también contiene un nuevo paquete de informes para Fortify Software Security Center compatible con 2023 CWE Top 25, que se puede descargar de Fortify Customer Support Portal, en la sección Premium Content.

OWASP API Security Top 10 2023

Para acompañar las nuevas correlaciones, esta versión también contiene un nuevo paquete de informes para Fortify Software Security Center compatible con OWASP API Security Top 10, que se puede descargar de Fortify Customer Support Portal, en la sección Premium Content.

Fortify Taxonomy: errores en la seguridad del software

El sitio de Fortify Taxonomy, que contiene descripciones de la compatibilidad con las nuevas categorías añadidas, está disponible en https://vulncat.fortify.com.

Comuníquese con el soporte técnico de Fortify

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

Comuníquese con SSR

Alexander M. Hoole
Director sénior del equipo de Software Security Research
OpenText Fortify
hoole@opentext.com
+1 (650) 427-9973

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