Software Security Research Release Announcement

# Fortify Software Security Content

## 2021 Update 4

## December 17, 2021

### About CyberRes Fortify Software Security Research

The Fortify Software Security Research team translates cutting-edge research into security intelligence that powers the Fortify product portfolio – including Fortify Static Code Analyzer (SCA), Fortify WebInspect, and Fortify Application Defender. Today, CyberRes Fortify Software Security Content supports 1,137 vulnerability categories across 29 languages and spans more than one million individual APIs.

Fortify Software Security Research (SSR) is pleased to announce the immediate availability of updates to Fortify Secure Coding Rulepacks (English language, version 2021.4.0), Fortify WebInspect SecureBase (available via SmartUpdate), and Fortify Premium Content.

## CyberRes Fortify Secure Coding Rulepacks [SCA]

With this release, the Fortify Secure Coding Rulepacks detect 917 unique categories of vulnerabilities across 29 programming languages and span over one million individual APIs. In summary, this release includes the following:

### .NET Core and ASP.NET updates (Version Supported: .NET Core 3.1)

Improved support for various .NET Core and ASP.NET Core namespaces, including the following:

- Microsoft.AspNetCore.Http
- Microsoft.AspNetCore.Mvc
- Microsoft.Extensions.Logging
- System
- System.IO
- System.Net
- System.Threading

The support improves the coverage of the following categories:

- Cross-Site Scripting: Inter-Component Communication (Cloud)
- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation
- Cross-Site Scripting: Reflected
- Log Forging
- Log Forging (debug)
- Privacy Violation
- System Information Leak

### Azure

Azure is Microsoft's public cloud computing platform that provides a range of cloud services including compute, containers, Internet of Things, AI, and machine learning.

In this release, we provide initial support for several key Azure services: Functions, Identity, and CosmosDB. Additionally, the following specific Azure technologies are now supported:

#### Azure Functions (Versions Supported: Java 1.3.1, C# 3.x)

Functions are Microsoft Azure's serverless compute solution. Azure Functions provide a continually updated infrastructure to run your application, build web APIs, respond to database changes, and manage message queues. This update includes initial support for the following trigger types for C# and Java:

- Blob Trigger
- CosmosDB Trigger
- Event Trigger

- Http Trigger (as class library)
- Http Trigger (as C# isolated process)
- Queue Trigger
- ServiceBus Trigger

Azure Functions support includes the following categories:

- Cookie Security: Cookie not Sent Over SSL
- Cookie Security: HTTPOnly not Set
- Cookie Security: Overly Broad Path
- Cookie Security: Overly Broad Domain
- Cookie Security: Persistent Cookie
- Cross-Site Scripting: Inter-Component Communication (Cloud)
- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation
- Cross-Site Scripting: Reflected
- Denial Of Service
- Header Manipulation
- Header Manipulation: Cookies
- HTML5: Overly Permissive CORS Policy
- Privacy Violation
- Resource Injection
- Setting Manipulation
- System Information Leak: External

### Azure Identity (Versions Supported: C# 1.5.0, Java 1.4.1)

Azure Identity is Microsoft's cloud-based identity and access management service. It provides authentication and authorization to resources within an organization. This update includes initial support for the following namespaces:

C#

- Azure.Identity (version 1.5.0)

Java

- com.azure.identity (version 1.4.1)

Azure Identity support includes the following categories:

- Password Management
- Password Management: Empty/Hardcoded/Null Password
- Password Management: Weak Cryptography
- Resource Injection

### Azure CosmosDB (Version Supported: 3.x)

Azure Cosmos DB is a globally distributed, multi-model database service. With Azure Cosmos DB, you can store and access document, key-value, wide-column, and graph databases by using APIs and programming models. This update includes initial support for the following namespaces for C#:

- Microsoft.Azure.Cosmos
- Microsoft.Azure.Cosmos.Scripts
- Microsoft.Azure.Cosmos.Table

Azure Cosmos DB support includes the following categories:

- Denial of Service
- HTML5: Overly Permissive CORS Policy
- Insecure Transport
- NoSQL Injection: CosmosDB
- Resource Injection
- Setting Manipulation
- SQL Injection

**AWS**

Amazon Web Services (AWS) is a public cloud computing platform that provides a range of cloud services including computing, storage, networking, database, Internet of things and machine learning.

In this release, we provide initial support for several key AWS services: IAM, DynamoDB, and RDS. This release also adds initial Lambda support for C# and updated support for Java. Additionally, the following specific AWS technologies are now supported:

**AWS Lambda updates (Versions Supported: .NET AWS SDK 3.7.x, Java AWS SDK v2 2.17.x)** [1]

Lambda is a compute service, provided by Amazon as part of Amazon Web Services (AWS), that runs code without provisioning or managing servers. The Lambda service runs code in response to events and automatically manages computing resources required by the code. This update includes initial support for C# and additional support for Java. This update includes support for the following namespaces for C# and Java:

C#

- Amazon.Lambda
- Amazon.Lambda.APIGatewayEvents
- Amazon.Lambda.Core

Java

- com.amazonaws.services.lambda.runtime
- com.amazonaws.services.lambda.runtime.events

This update includes additional support for the following event types:

- API Gateway Events (C#, Java)
- DynamoDB (Java)
- S3 Events (Java)
- Scheduled Events (Java)

AWS Lambda support includes the following categories:

- Cross-Site Scripting: Inter-Component Communication
- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation

[1] For improved analysis, include AWS SAM or CloudFormation YAML/JSON templates in the translation.

- Cross-Site Scripting: Reflected
- Header Manipulation
- Header Manipulation: Cookies
- Log Forging
- Privacy Violation
- System Information Leak: External
- System Information Leak: Internal

### AWS IAM (Versions Supported: .NET AWS SDK 3.7.x, Java AWS SDK v2 2.17.x)

AWS Identity and Access Management (IAM) is a web service that controls access to AWS resources. IAM can be used to control authenticated and authorized use of AWS resources. This update includes support for C# and Java. This update includes support for the following namespaces for C# and Java:

C#

- Amazon.IdentityManagement.Model

Java

- com.amazonaws.services.identitymanagement.model
- software.amazon.awssdk.services.iam.model

In addition to identifying sensitive information, AWS IAM support includes the following categories:

- Password Management
- Password Management: Empty/Hardcoded/Null Password
- Password Management: Weak Cryptography

### AWS DynamoDB (Versions Supported: .NET AWS SDK 3.7.x, Java AWS SDK v2 2.17.x)

AWS DynamoDB is a fully managed NoSQL database service that supports key-value and document data structures. DynamoDB can be used to store and retrieve data and serve arbitrary amounts of request traffic. This update includes initial support for C# and updated support for Java. Support includes the following namespaces:

C#

- Amazon.DynamoDBv2.Model
- Amazon.DynamoDBv2.DataModel
- Amazon.DynamoDBv2.DocumentModel

Java

- com.amazonaws.services.lambda.runtime.events.models.dynamodb
- software.amazon.awssdk.enhanced.dynamodb
- software.amazon.awssdk.enhanced.dynamodb.model

AWS DynamoDB support includes the following categories:

- Access Control: Database
- NoSQL Injection: DynamoDB
- SQL Injection: PartiQL

### AWS Relational Database Service (RDS) Data API for Aurora Serverless (Versions Supported: .NET AWS SDK 3.7.x, Java AWS SDK v2 2.17.x)

Amazon Aurora is a MySQL and PostgreSQL-compatible relational database engine that is part of the managed Amazon Relational Database Service (Amazon RDS). The AWS RDS Data API provides a web-service interface enabling applications to access and execute SQL statements against an Aurora Serverless database cluster. This update includes support for the following namespaces for C# and Java:

C#

- Amazon.RDSDataService.Model

Java

- software.amazon.awssdk.services.rdsdata.model (V2)

AWS RDS support includes the following categories:

- Access Control: Database
- Setting Manipulation
- SQL Injection

### Secret Scanning

Support for Secret Scanning. Secret Scanning is a technique to automatically search for secrets in text files. In this context "secrets" refers to passwords, API tokens, encryption keys, and similar artifacts meant to be undisclosed. The main purpose is to find accidentally hardcoded secrets in source code and configuration files. Extended support for all languages and additional file types via the new Regex analysis[2]. The categories supported include:

- Credential Management: Hardcoded API Credentials
- Key Management: Hardcoded Encryption Key
- Password Management: Hardcoded Password

### Trojan Source

Trojan Source[3] is a category of vulnerabilities published by Nick Boucher and Ross Anderson in their paper "Trojan Source: Invisible Vulnerabilities". They demonstrate 5 distinct ways Unicode special characters can be used to make code appear one way to the naked eye of a developer, but when executed work a different way. Trojan Source should be considered an insider threat scenario whereas a malicious individual can knowingly insert the Unicode characters. Due to the precision of one of the categories, we are including detection support in the core Rulepacks for the following Languages: C, C++, C#, Go, Java, JavaScript, Python, and Rust. The supported categories include:

- Encoding Confusion: BiDi Control Characters

### Static/Dynamic Issue Correlation[4]

Support for exporting data to enable correlating static and dynamic scan results in Fortify Software Security Center (SSC) for Java Spring projects. The categories supported include:

---

[2] Requires Fortify Static Code Analyzer v21.2.0 or later.
[3] Requires Fortify Static Code Analyzer v21.2.0 or later.
[4] Requires Fortify Static Code Analyzer v21.2.0 or later. To enable correlation output pass the property `com.fortify.sca.rules.enable_wi_correlation` at scan time. This can be done either with command line arguments or by modifying the SCA properties files.

- Cross-Site Scripting: Content Sniffing
- Cross-Site Scripting: Inter Component Communication
- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation
- Cross-Site Scripting: Reflected
- SQL Injection
- SQL Injection: Hibernate

### Extended IBM Mainframe COBOL Support (Version Supported: 6.3)

This update includes detection of Integer Overflow vulnerabilities in IBM Mainframe COBOL code.

### Cloud Infrastructure as Code

Support for cloud Infrastructure as Code (IaC). IaC is the process of managing and provisioning computer resources through code, rather than various manual processes. Technologies supported include AWS, AWS CloudFormation, Azure ARM, Kubernetes K8S, and Azure Kubernetes Service. Common issues related to the configuration of the services mentioned are now reported to the developer, including:

- Access Control: Azure Blob Storage
- Access Control: Azure Container Registry
- Access Control: Azure Network Group
- Access Control: Azure SQL Database
- Access Control: Azure Storage
- Access Control: Cosmos DB
- Access Control: EC2
- Access Control: Kubernetes Admission Controller
- Access Control: Kubernetes Image Authorization Bypass
- Access Control: Overly Broad IAM Principal
- Access Control: Overly Permissive S3 Policy
- AKS Bad Practices: Missing Azure Monitor Integration
- Ansible Bad Practices: Missing CloudWatch Integration
- Ansible Bad Practices: Redshift Publicly Accessible
- Ansible Misconfiguration: Log Validation Disabled
- AWS CloudFormation Bad Practices: Missing CloudWatch Integration
- AWS CloudFormation Bad Practices: Redshift Publicly Accessible
- AWS CloudFormation Bad Practices: User-Bound IAM Policy
- AWS CloudFormation Misconfiguration: API Gateway Unauthenticated Access
- AWS CloudFormation Misconfiguration: Insecure Transport
- AWS CloudFormation Misconfiguration: Insufficient CloudTrail Logging
- AWS CloudFormation Misconfiguration: Insufficient DocumentDB Logging
- AWS CloudFormation Misconfiguration: Insufficient Neptune Logging
- AWS CloudFormation Misconfiguration: Insufficient RedShift Logging
- AWS CloudFormation Misconfiguration: Insufficient S3 Logging
- AWS CloudFormation Misconfiguration: Log Validation Disabled
- AWS CloudFormation Misconfiguration: root User Access Key
- AWS CloudFormation Misconfiguration: Unrestricted Lambda Principal
- Azure Monitor Misconfiguration: Insufficient Logging
- Azure Resource Manager Bad Practices: Cross-Tenant Replication

- Azure Resource Manager Bad Practices: Remote Debugging Enabled
- Azure Resource Manager Bad Practices: SSH Password Authentication
- Azure Resource Manager Misconfiguration: Insecure Transport
- Azure Resource Manager Misconfiguration: Overly Permissive CORS Policy
- Azure Resource Manager Misconfiguration: Security Alert Disabled
- Azure SQL Database Misconfiguration: Insufficient Logging
- Insecure SSL: Inadequate Certificate Verification
- Insecure Storage: Missing DocumentDB Encryption
- Insecure Storage: Missing EBS Encryption
- Insecure Storage: Missing Elasticache Encryption
- Insecure Storage: Missing Neptune Encryption
- Insecure Storage: Missing RDS Encryption
- Insecure Storage: Missing Redshift Encryption
- Insecure Storage: Missing S3 Encryption
- Insecure Storage: Missing SNS Topic Encryption
- Insecure Transport: Azure Storage
- Insecure Transport: Database
- Insecure Transport: Missing Elasticache Encryption
- Key Management: Excessive Expiration
- Kubernetes Bad Practices: API Server Publicly Accessible
- Kubernetes Bad Practices: Default Namespace
- Kubernetes Bad Practices: Host Write Access
- Kubernetes Bad Practices: Missing API Server Authorization
- Kubernetes Bad Practices: Missing Kubelet Authorization
- Kubernetes Bad Practices: Missing Node Authorization
- Kubernetes Bad Practices: Missing RBAC Authorization
- Kubernetes Bad Practices: NamespaceLifecycle Enforcement Disabled
- Kubernetes Bad Practices: readOnlyPort Enabled
- Kubernetes Bad Practices: Static Authentication Token
- Kubernetes Bad Practices: Unconfigured API Server Logging
- Kubernetes Misconfiguration: API Server Anonymous Access
- Kubernetes Misconfiguration: API Server Logging Disabled
- Kubernetes Misconfiguration: HTTP Basic Authentication
- Kubernetes Misconfiguration: Insecure Transport
- Kubernetes Misconfiguration: Kubelet Anonymous Access
- Kubernetes Misconfiguration: Missing Garbage Collection Threshold
- Kubernetes Misconfiguration: Missing Kubelet Client Certificate
- Kubernetes Misconfiguration: Overly Permissive Capabilities
- Kubernetes Misconfiguration: Privileged Container
- Kubernetes Misconfiguration: Server Identity Verification Disabled
- Kubernetes Misconfiguration: Unbound Controller Manager
- Kubernetes Misconfiguration: Unbound Scheduler
- Poor Logging Practice: Excessive Cloud Log Retention
- Poor Logging Practice: Insufficient Cloud Log Retention
- Poor Logging Practice: Insufficient Cloud Log Rotation
- Poor Logging Practice: Insufficient Cloud Log Size

- Privacy Violation: Exposed Default Value
- Privilege Management: Overly Broad Access Policy
- Privilege Management: Overly Permissive Role
- System Information Leak: Kubernetes Profiler

## OWASP Top 10 2021

The Open Web Application Security Project (OWASP) Top 10 2021 provides a powerful awareness document for web application security, focused on informing the community about the consequences of the most common and most critical web application security risks. The OWASP Top 10 represents a broad agreement about what the most critical web application security flaws are with consensus drawn from data collection and survey results. To support our customers who want to mitigate Web Application risk, correlation of the Micro Focus Fortify Taxonomy to the newly released OWASP Top 10 2021 has been added.

## Miscellaneous Errata

In this release, we have continued to invest resources to ensure we can reduce the number of false positive issues and improve the ability for customers to audit issues. Customers can also expect to see changes in reported issues related to the following:

### *Deprecation of Fortify Static Code Analyzer versions prior to 18.x:*

As mentioned in our 2021.3 release announcement, that was the last release of the Rulepacks that support Fortify Static Code Analyzer versions prior to 18.x. For this release, Fortify Static Code Analyzer versions prior to 18.x will not load the Rulepacks. This will require either downgrading the Rulepacks or upgrading the version of SCA. For future releases, we will continue to support the last four major releases of Fortify Static Code Analyzer.

### *PHP Improvements*

Improved support for identifying password and encryption keys in the Key Management: Empty /Hardcoded/Null Encryption Key categories.

### *Python Improvements*

Improved support for the *subprocess* module resulting in improved detection of issues, such as Command Injection.

### *False Positive improvements:*

Work has continued with the effort to remove false positives in this release. On top of other improvements, customers can expect to see additional removal of false positives in the following areas:

- Issues coming from Akka actors in Scala projects when the application is not using Play.
- Cross-site Scripting issues in JavaScript when only partial control over URLs is obtainable.
- Password Management issues in JSON files when referring to localization of strings
- Dataflow issues in Java and .NET projects coming from HTTP methods.

## CyberRes Fortify SecureBase [Fortify WebInspect]

Fortify SecureBase combines checks for thousands of vulnerabilities with policies that guide users in the following updates available immediately via SmartUpdate:

### API Discovery

This release includes a check for API Discovery. The API Discovery check is flagged when WebInspect detects an API definition in the swagger specification(spec) at the user-specified location provided through check input. These spec files might not be directly referenced in any page and therefore are not detected in the crawl. In addition to checking for swagger specifications, in the user specified locations, definitions found during the scan that are not explicitly specified with the check input will also be flagged and tested. While these findings do not necessarily indicate a security vulnerability, they increase resources that are potentially vulnerable to attacks.

### Vulnerability Support

### OGNL Expression Injection: Double Evaluation

A critical OGNL Expression Injection vulnerability identified by CVE-2021-26084 affects Atlassian Confluence Server and Data Center. This vulnerability allows an unauthenticated attacker to execute arbitrary code on vulnerable applications. The affected Atlassian server versions are before version 6.13.23, from version 6.14.0 to before 7.4.11, from version 7.5.0 to before 7.11.6, and from version 7.12.0 to before 7.12.5. This release includes a check to detect this vulnerability in affected Atlassian servers.

### Directory Traversal

Apache HTTP Server has been found to be vulnerable to Directory Traversal attacks identified by CVE-2021-41773 and CVE-2021-42013. The vulnerabilities enable an attacker to manipulate URLs that map URLs to files outside the directories configured by alias-like directives. Attackers might recover the contents of files on the server, leading to sensitive data disclosure, potential recovery of proprietary business logic, and for some configurations, remote code execution. These issues only affect Apache HTTP Server versions 2.4.49 and 2.4.50. This release contains a check to detect these vulnerabilities in Apache HTTP Server.

### Path Manipulation: Special Characters

A Path Manipulation vulnerability identified by CVE-2021-28164 affects Eclipse Jetty. The default compliance mode in affected versions allows requests with URIs that contain segments with special characters to access protected resources in the WEB-INF directory. This can reveal sensitive information regarding the implementation of a web application and bypass some security constraints. This release contains a check to detect vulnerable Jetty instances.

### Dynamic Code Evaluation: Unsafe XStream Deserialization

XStream is a commonly used tool for converting data between Java objects and XML. The processed stream at unmarshalling time contains type information to recreate the formerly written objects. An attacker can manipulate the processed input stream and replace or inject objects, which results in the execution of arbitrary code loaded from a remote server. This release includes a check to detect the latest unsafe XStream deserialization vulnerability CVE-2021-39149 vulnerability on target web servers.

### Path Manipulation: Special Characters

Control characters such as 0x09 should not be allowed in a URL path and must be percent-encoded by clients. The inconsistent parsing of these control characters between proxy and backend server might introduce various threats. This release includes a check to detect if some common control characters could be allowed to be inserted in the URL path and negatively impact the backend Web Server.

### Compliance Reports

### OWASP Top 10 2021

The Open Web Application Security Project (OWASP) Top 10 2021 provides a powerful awareness document for web application security, focused on informing the community about the consequences of the most common and most critical web application security risks. The OWASP Top 10 represents a broad agreement about what the most critical web application security flaws are with consensus drawn from data collection and survey results. This SecureBase update includes a new compliance report template that provides correlation between OWASP Top 10 2021 categories and WebInspect checks.

### Policy Updates

### OWASP Top 10 2021

A policy customized to include checks relevant to OWASP Top 10 2021 has been added to the WebInspect SecureBase list of supported policies. This policy contains a subset of the available WebInspect checks that allow customers to run compliance specific WebInspect Scans.

### Miscellaneous Errata

In this release, we have continued to invest resources to ensure we can reduce the number of false positive issues and improve the ability for customers to audit issues. Customers can also expect to see changes in reported issues related to the following:

#### SSL check improvement

The SSL Cipher List check was improved to reflect that the following configuration does not support perfect forward secrecy: TLS_DH_RSA_WITH_AES_128_GCM_SHA256.

## CyberRes Fortify Premium Content

The research team builds, extends, and maintains a variety of resources outside our core security intelligence products.

### OWASP Top 10 2021

To accompany the new correlations, this release also contains a new report bundle with support for OWASP Top 10 2021, which is available for download from the Fortify Customer Support Portal under Premium Content.

## CyberRes Fortify Taxonomy: Software Security Errors

The Fortify Taxonomy site, which contains descriptions for newly added category support, is available at https://vulncat.fortify.com. Customers looking for the legacy site, with the last supported update, can obtain it from the CyberRes Fortify Support Portal.

## Contact Fortify Technical Support

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

## Contact SSR

**Alexander M. Hoole**
Senior Manager, Software Security Research
CyberRes Fortify
hoole@microfocus.com
+1 (650) 258-5916

**Peter Blay**
Manager, Software Security Research
CyberRes Fortify
peter.blay@microfocus.com