



Software Security Research Release Announcement

HPE Security Fortify Software Security Content

2016 Update 2

June 30, 2016

HPE Security Fortify Software Security Research (SSR) is pleased to announce the immediate availability of updates to HPE Security Fortify Application Defender, HPE Security Fortify WebInspect SecureBase (available via SmartUpdate), HPE Security Fortify Secure Coding Rulepacks (English language, version 2016.2.0), and HPE Security Fortify Premium Content.

About HPE Security SSR

The Software Security Research team translates cutting-edge research into security intelligence that powers the HPE Security products portfolio. Today, HPE Security Fortify Software Security Content supports 907 vulnerability categories across 23 programming languages and spans more than 839,000 individual APIs.

Learn more at

hpe.com/software/ssr

HPE Security Fortify Secure Coding Rulepacks [SCA]

With this release, the Fortify Secure Coding Rulepacks detect 704 unique categories of vulnerabilities across 23 programming languages and span over 839,000 individual APIs. In summary, the release includes the following:

Swift Enhancements¹

This release provides further support for the Swift programming language on iOS, enabling dataflow detection across numerous weakness categories. Enhanced coverage for detecting 25 additional categories includes the following:

- Access Control: Database
- Cross-Site Scripting: Reflected
- Denial of Service: Regular Expression
- Dynamic Code Evaluation: Code Injection
- Insecure SSL: Overly Broad Certificate Trust
- Insecure SSL: Server Identity Verification Disabled
- Insecure Transport
- JSON Injection
- Log Forging
- Often Misused: Encoding
- Often Misused: File System
- Open Redirect
- Path Manipulation
- Privacy Violation
- Resource Injection
- Unsafe Reflection
- Weak Cryptographic Hash: User-Controlled PBE Salt
- Weak Encryption: User-Controlled Key Size
- XML Injection

¹ Requires HPE Security Fortify SCA 16.10.

Express JS Support²

New support for Express JS framework modeling and the identification of the use of helmet.js protections. This release includes new dataflow sources and sinks covering 35 categories, of which nine categories are new to JavaScript, as well as the following two entirely new categories:

- Cross-Site Scripting: Untrusted HTML Downloads
- Helmet Misconfiguration: Insecure XSS Filter

Struts 2 Enhancements

Support for detecting double evaluation of OGNL expressions (CVE-2016-0785) and Dynamic Method Invocations (DMI) (CVE-2016-3081, CVE-2016-3087):

- OGNL Expression Injection: Double Evaluation
- OGNL Expression Injection: Dynamic Method Invocation

Server-Side Template Injection

Detection of a new category for Server-Side Template Injection is now supported in a Java template engine Velocity as well as Python Jinja2 and Django. Detected weaknesses may allow attackers to execute arbitrary code on the application server.

Certificate Pinning

SCA will now detect when SSL/TLS connections are being established with the default pre-loaded system keystore and thus are lacking any kind of certificate or public key pinning (Insecure SSL: Overly Broad Certificate Trust). In addition, SCA will also report Insecure SSL: Inadequate Certificate Verification when methods that are not appropriate for checking the certificate chain are used for that purpose in both iOS (OBJC/Swift) and Android (Java).

.NET improvements

Deserializing user provided, or untrusted data, using types such as BinaryFormatter, SoapFormatter, and NetDataContractSerializer can cause CLR to load and create an object of an arbitrary attacker-specified type (potentially leading to dynamic code execution during deserialization process). The following two new weakness categories have been introduced to detect dynamic code execution during deserialization, in addition to adding detection capabilities for seven existing weakness categories, for the above mentioned types:

- Dynamic Code Evaluation : Unsafe Deserialization
- Dynamic Code Evaluation : Serializable Delegate

PCI DSS 3.2

In order to support our e-commerce and financial services customers in the area of compliance, this release supports correlation between the HPE Security Fortify Taxonomy categories and the requirements specified in the latest version of the Payment Card Industry Data Security Standard, version 3.2.

² Express JS support requires SCA version 4.42 or higher and JavaScript to be enabled as a language using higher order analysis. For best results, use 16.10 or higher.

HPE SecureBase [WebInspect]

HPE SecureBase combines checks for thousands of vulnerabilities with policies that guide users in identifying critical weaknesses in web and mobile software. In summary, this release includes the following updates available immediately via SmartUpdate:

Vulnerability support

OGNL Expression Injection: Double Evaluation

Forced double OGNL evaluation allows attackers to execute arbitrary OGNL expressions when controlling the output of the first evaluation. The recently disclosed Apache Struts 2 issue, outlined in CVE-2016-0785, manifests when an application forces double OGNL evaluation of unsanitized user input in tag attributes. This release includes a check to test web applications against this issue.

OGNL Expression Injection: Dynamic Method Invocation

Using the "method:" or "!" prefix in the action URL can lead to the invocation of any public method in the action, or other specially crafted OGNL expression payload, as disclosed in the recent Apache Struts 2 related CVE-2016-3081 and CVE-2016-3087. This release includes multiple checks to evaluate web applications against these issues.

SQL Injection Enhancement – Hibernate Query Language

Hibernate Query Language (HQL) is an SQL-like structured language to query Hibernate Java objects. This release includes enhancement to WebInspect SQL Injection check to also detect HQL injection vulnerabilities in Web Applications using Hibernate.

Compliance Report

PCI DSS 3.2 Compliance Template

This release includes support for the latest version of the Payment Card Industry Data Security Standard (DSS) Compliance Template, version 3.2. This template is a significant improvement over earlier versions of WebInspect PCI reports, as it provides correlation to many additional checks as well as an alignment to the PCI DSS report generated for HPE Security Fortify SCA scans and results integrated within HPE Security Fortify Software Security Center (SSC).

HPE Security Fortify Application Defender

HPE Security Application Defender is a runtime application self-protection (RASP) solution that helps organizations manage and mitigate risk from homegrown or third-party applications. It provides centralized visibility into application use and abuse while protecting from software vulnerability exploits and other violations in real time. For this release, the Software Security Research team provides the following feature improvements:

LDAP Injection

Non-signature based detection for LDAP Injection. The rule analyzes LDAP search filters for dangling search terms and cross-boundary inputs from HTTP.

OGNL Expression Injection: Dynamic Method Invocation

A virtual patch for the recent Apache Struts 2 related vulnerabilities CVE-2016-3081 and CVE-2016-3087.

Enhancements

Improved accuracy for Cross-site Scripting and SQL Injection signatures.

HPE Security Fortify Premium Content

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

PCI DSS 3.2 Report

To accompany the new mapping, this release also contains a new report bundle for HPE Security Fortify SSC with support for PCI DSS 3.2, which is available for download from the Fortify Customer Portal under Premium Content.

© Copyright 2016 Hewlett Packard Enterprise Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HPE products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HPE shall not be liable for technical or editorial errors or omissions contained herein.