Digital Veterans Platform

Quality/Secure Code Review Validation ReadMe

# Overview

This document provides a description of the contents of the Digital Veterans Platform (DVP) health-apis-vista-fhir-query Quality Code Review Validation and Secure Code Review Validation package. This information is provided to assist VA Software Assurance team members in performing their validation work. Questions about the package may be directed to evan.clendenning@va.gov for immediate clarification.

Analysis Information:

* **Fortify Version:** 21.2.3
* **Rulepacks Version:** 2022.1.0.0007

# Package Contents

* **ReadMe.docx** – This document.
* **health-apis-vista-fhir-query.zip** – This zip file contains the source code, suitable for comparison with the FPR file to ensure all applicable code has been properly scanned.
* **health-apis-vista-fhir-query-fortify-logs.zip** - This zip file contains any relevant logs that were output during the fortify build/scan.
* **health-apis-vista-fhir-query-fortify-202204131934.fpr**– The fortify scan file.
* **scanner** – A shell script to be used when running scans applications on the DVP.
* **maven** – A shell script to be used with scanner for scanning maven based projects.
* **vfq-dependencies.zip** – A zip file containing a vfq-dependencies.txt which lists the dependencies as well as the dependency jars themselves.
* **VA SwA Composition Analysis health-apis-vista-fhir-query v2.0.158 2022-04-12 Appendix** – audit of composition analysis results.

# Build Instructions

The following instructions describe how to work with the project code and related development tools:

1. Install Java JDK 14 installed.
   1. Download the JDK 14 executable[here](https://adoptopenjdk.net/releases?variant=openjdk14&jvmVariant=hotspot).
   2. Ensure that the JAVA\_HOME system variable is set to <jdk14-install-directory> and that the PATH system variable contains %JAVA\_HOME%/bin
2. Install Apache Maven:
   1. Download Apache Maven from [this link](https://maven.apache.org/download.cgi). (If you're not sure which archive to download, you most likely need the binary zip archive.)
   2. Extract the zip and take not of where you extracted it.
   3. Add the <maven-install-directory>/bin folder to your PATH system variable.
3. To build each project:
   1. Unzip the folder then navigate to the projects root directory
   2. Run the command: mvn clean install -DskipTests

If any issues or questions occur, please contact us.

# Excluded Files and Folders

The tables below list files and folders which are present in the delivered code (see project-code.zip), but have been excluded from Fortify scanning. For ease of review, each project or category of folder has been split into its own table. The “Folder/File Excluded” column lists the full path to each excluded folder or file, and the “Reason” column describes the rationale for the exclusion. Whenever a folder is excluded, all contents of the folder and any subfolders have been excluded for the same reason.

Table 1 – health-apis-vista-fhir-query

|  |  |
| --- | --- |
| **Folder/File Excluded** | **Reason** |
| .\vista-fhir-query\src\test | This folder contains test code exclusively. There is no code in this folder or its subfolders which is part of the production deployment. |
| .\vista-fhir-query-tests | This module contains test code exclusively. There is no code which is part of the production deployment. |
| .\vista-fhir-query-mock-services | This module contains development/test support code exclusively. There is no code which is part of the production deployment. |

# Scan Errors and Warnings

Stuff about including scan output and write off for Fortify not being able to find packages etc.

health-apis-vista-fhir-query

|  |  |
| --- | --- |
| Warning | [1214]  Multiple definitions found for class |
| Resolution | False Positive. The warnings provided classpaths point to the same exact class. |
| Warning | [13554]  ‘lambda0’ matched a primitive type ‘void’ instead of a lambda |
| Resolution | This warning is a false positive. Per Java's documentation, method references are a valid way of defining functions. The method reference is semantically the same as a lambda expression. |