Fortify Security Report

1 Aug, 2019

saurabh.joshi

Executive Summary

Issues Overview

On 1 Aug, 2019, a source code review was performed over the 1 code base. 108 files, 2,114 LOC (Executable) were scanned and reviewed for defects that could lead to potential security vulnerabilities. A total of 436 reviewed findings were uncovered during the analysis.

Issues by Fortify Priority Order	
Low	433
High	3

Recommendations and Conclusions

The Issues Category section provides Fortify recommendations for addressing issues at a generic level. The recommendations for specific fixes can be extrapolated from those generic recommendations by the development group.

Project Summary

Code Base Summary

Code location: /Users/saurabh.joshi/Work/api_automation/src/main/java
Number of Files: 108
Lines of Code: 2114
Build Label: <No Build Label>

Scan Information

Scan time: 01:26
SCA Engine version: 19.1.0.2241
Machine Name: saurabhjoshi.local
Username running scan: saurabh.joshi

Results Certification

Results Certification Valid

Details:

Results Signature:

SCA Analysis Results has Valid signature

Rules Signature:

There were no custom rules used in this scan

Attack Surface

Attack Surface:
File System:
java.io.FileReader.FileReader

Private Information:
null.null.null
java.util.Properties.getProperty

Java Properties:
java.util.Properties.load

System Information:
null.null.null

Filter Set Summary

Current Enabled Filter Set:
Security Auditor View

Filter Set Details:

Folder Filters:
If [fortify priority order] contains critical Then set folder to Critical
If [fortify priority order] contains high Then set folder to High
If [fortify priority order] contains medium Then set folder to Medium
If [fortify priority order] contains low Then set folder to Low

Audit Guide Summary

Audit guide not enabled

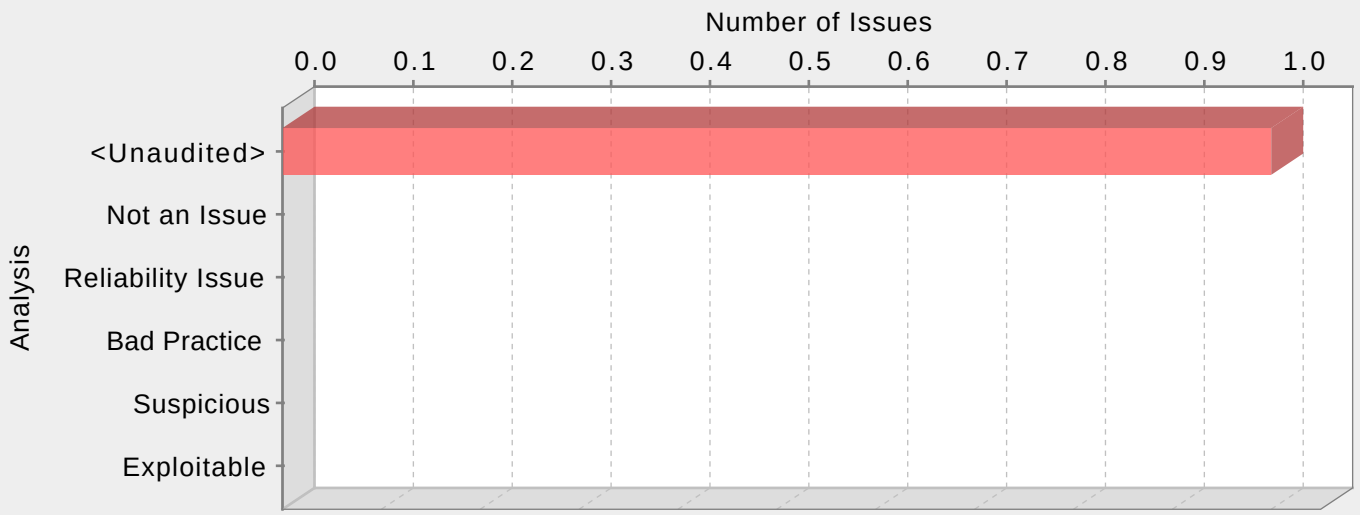
Results Outline

Overall number of results

The scan found 436 issues.

Vulnerability Examples by Category

Category: Insecure Randomness (1 Issues)



Abstract:

The random number generator implemented by nextInt() cannot withstand a cryptographic attack.

Explanation:

Insecure randomness errors occur when a function that can produce predictable values is used as a source of randomness in a security-sensitive context.

Computers are deterministic machines, and as such are unable to produce true randomness. Pseudorandom Number Generators (PRNGs) approximate randomness algorithmically, starting with a seed from which subsequent values are calculated.

There are two types of PRNGs: statistical and cryptographic. Statistical PRNGs provide useful statistical properties, but their output is highly predictable and form an easy to reproduce numeric stream that is unsuitable for use in cases where security depends on generated values being unpredictable. Cryptographic PRNGs address this problem by generating output that is more difficult to predict. For a value to be cryptographically secure, it must be impossible or highly improbable for an attacker to distinguish between the generated random value and a truly random value. In general, if a PRNG algorithm is not advertised as being cryptographically secure, then it is probably a statistical PRNG and should not be used in security-sensitive contexts, where its use can lead to serious vulnerabilities such as easy-to-guess temporary passwords, predictable cryptographic keys, session hijacking, and DNS spoofing.

Example: The following code uses a statistical PRNG to create a URL for a receipt that remains active for some period of time after a purchase.

```
String GenerateReceiptURL(String baseUrl) {
Random ranGen = new Random();
ranGen.setSeed((new Date()).getTime());
return (baseUrl + ranGen.nextInt(400000000) + ".html");
}
```

This code uses the Random.nextInt() function to generate "unique" identifiers for the receipt pages it generates. Since Random.nextInt() is a statistical PRNG, it is easy for an attacker to guess the strings it generates. Although the underlying design of the receipt system is also faulty, it would be more secure if it used a random number generator that did not produce predictable receipt identifiers, such as a cryptographic PRNG.

Recommendations:

When unpredictability is critical, as is the case with most security-sensitive uses of randomness, use a cryptographic PRNG. Regardless of the PRNG you choose, always use a value with sufficient entropy to seed the algorithm. (Do not use values such as the current time because it offers only negligible entropy.)

The Java language provides a cryptographic PRNG in java.security.SecureRandom. As is the case with other algorithm-based classes in java.security, SecureRandom provides an implementation-independent wrapper around a particular set of algorithms. When you request an instance of a SecureRandom object using SecureRandom.getInstance(), you can request a specific implementation of the algorithm. If the algorithm is available, then it is given as a SecureRandom object. If it is unavailable or if you do not specify a particular implementation, then you are given a SecureRandom implementation selected by the system.

Sun provides a single SecureRandom implementation with the Java distribution named SHA1PRNG, which Sun describes as computing:

"The SHA-1 hash over a true-random seed value concatenated with a 64-bit counter which is incremented by 1 for each operation. From the 160-bit SHA-1 output, only 64 bits are used [1]."

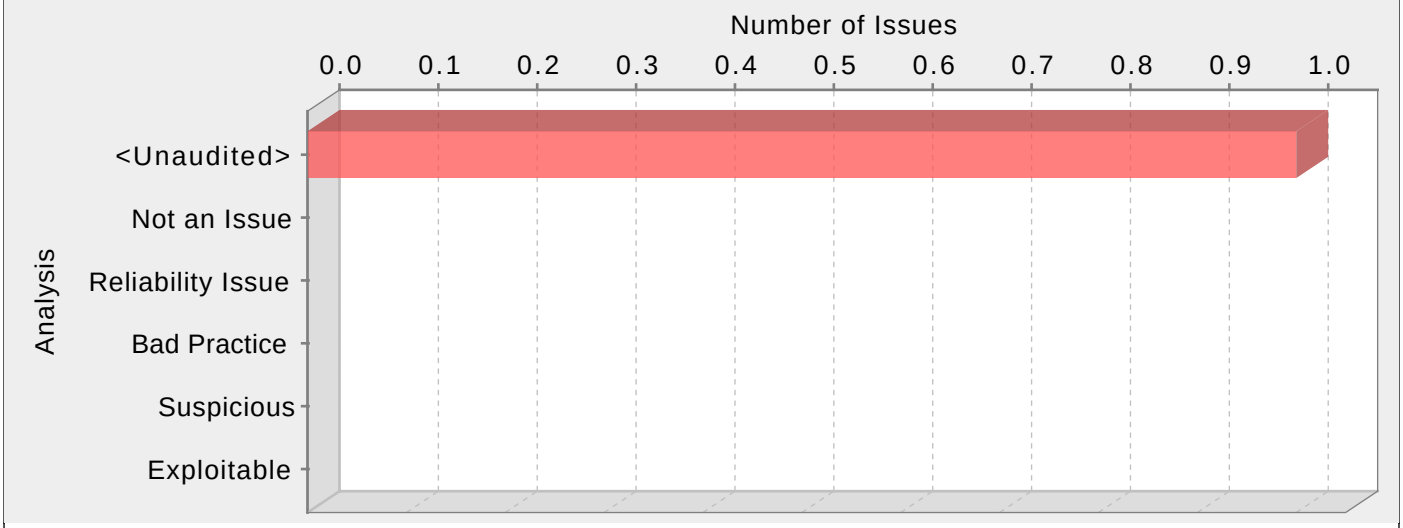
However, the specifics of the Sun implementation of the SHA1PRNG algorithm are poorly documented, and it is unclear what sources of entropy the implementation uses and therefore what amount of true randomness exists in its output. Although there is speculation on the Web about the Sun implementation, there is no evidence to contradict the claim that the algorithm is cryptographically strong and can be used safely in security-sensitive contexts.

returnTxnID.java, line 11 (Insecure Randomness)

Fortify Priority:	High	**Folder**	High
Kingdom:	Security Features		
Abstract:	The random number generator implemented by nextInt() cannot withstand a cryptographic attack.		
Sink:	returnTxnID.java:11 nextInt()		

```
9    int min = 1;
10   int max = 1000000000;
11   int randomNum1 = ThreadLocalRandom.current().nextInt(min, max + 1);
12   String randomNum2 = Integer.toString(randomNum1);
13   switch (IDType) {
```

Category: Null Dereference (1 Issues)



Abstract:

The method returnResponseBooleanParams() in returnResponseParams.java can crash the program by dereferencing a null pointer on line 77.

Explanation:

Null pointer exceptions usually occur when one or more of the programmer's assumptions is violated. A dereference-after-store error occurs when a program explicitly sets an object to null and dereferences it later. This error is often the result of a programmer initializing a variable to null when it is declared.

Most null pointer issues result in general software reliability problems, but if attackers can intentionally trigger a null pointer dereference, they can use the resulting exception to bypass security logic or to cause the application to reveal debugging information that will be valuable in planning subsequent attacks.

Example: In the following code, the programmer explicitly sets the variable foo to null. Later, the programmer dereferences foo before checking the object for a null value.

```
Foo foo = null;
...
foo.setBar(val);
...
}
```

Recommendations:

Implement careful checks before dereferencing objects that might be null. When possible, abstract null checks into wrappers around code that manipulates resources to ensure that they are applied in all cases and to minimize the places where mistakes can occur.

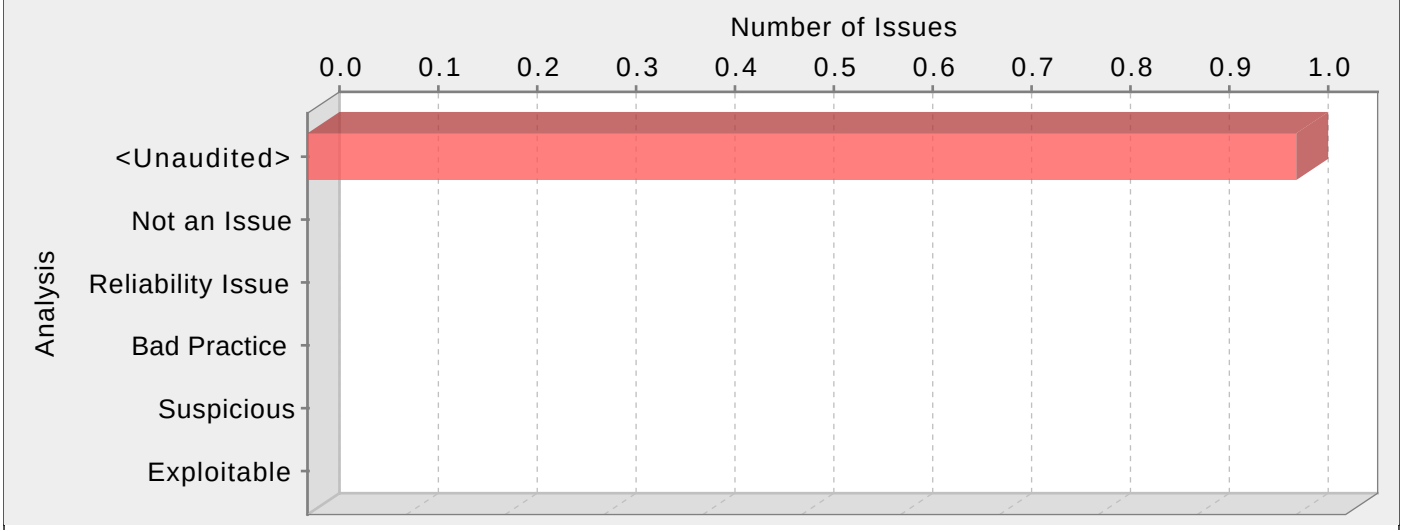
returnResponseParams.java, line 77 (Null Dereference)

Fortify Priority:	High	**Folder**	High
Kingdom:	Code Quality		
Abstract:	The method returnResponseBooleanParams() in returnResponseParams.java can crash the program by dereferencing a null pointer on line 77.		

Sink: returnResponseParams.java:77 Dereferenced : paramValue()

```
75          }
76        }
77        System.out.print("\nParam: " + paramValue +"\n");
78        return paramValue;
```

Category: Unreleased Resource: Streams (1 Issues)



Abstract:

The function ConfigProvider() in ConfigProvider.java sometimes fails to release a system resource allocated by FileReader() on line 19.

Explanation:

The program can potentially fail to release a system resource.

Resource leaks have at least two common causes:

- Error conditions and other exceptional circumstances.

- Confusion over which part of the program is responsible for releasing the resource.

Most unreleased resource issues result in general software reliability problems. However, if an attacker can intentionally trigger a resource leak, the attacker may be able to launch a denial of service attack by depleting the resource pool.

Example: The following method never closes the file handle it opens. The finalize() method for FileInputStream eventually calls close(), but there is no guarantee as to how long it will take before the finalize() method will be invoked. In a busy environment, this can result in the JVM using up all of its file handles.

```
private void processFile(String fName) throws FileNotFoundException, IOException {
FileInputStream fis = new FileInputStream(fName);
int sz;
byte[] byteArray = new byte[BLOCK_SIZE];
while ((sz = fis.read(byteArray)) != -1) {
processBytes(byteArray, sz);
}
}
```

Recommendations:

1. Never rely on finalize() to reclaim resources. In order for an object's finalize() method to be invoked, the garbage collector must determine that the object is eligible for garbage collection. Because the garbage collector is not required to run unless the JVM is low on memory, there is no guarantee that an object's finalize() method will be invoked in an expedient fashion. When the garbage collector finally does run, it may cause a large number of resources to be reclaimed in a short period of time, which can lead to "bursty" performance and lower overall system throughput. This effect becomes more pronounced as the load on the system increases.

Finally, if it is possible for a resource reclamation operation to hang (if it requires communicating over a network to a database, for example), then the thread that is executing the finalize() method will hang.

2. Release resources in a finally block. The code for the Example should be rewritten as follows:

```
public void processFile(String fName) throws FileNotFoundException, IOException {
FileInputStream fis;
try {
fis = new FileInputStream(fName);
int sz;
byte[] byteArray = new byte[BLOCK_SIZE];
while ((sz = fis.read(byteArray)) != -1) {
```

```
processBytes(byteArray, sz);
}
}
finally {
if (fis != null) {
safeClose(fis);
}
}
}

public static void safeClose(FileInputStream fis) {
if (fis != null) {
try {
fis.close();
} catch (IOException e) {
log(e);
}
}
}
```

This solution uses a helper function to log the exceptions that might occur when trying to close the stream. Presumably this helper function will be reused whenever a stream needs to be closed.

Also, the processFile method does not initialize the fis object to null. Instead, it checks to ensure that fis is not null before calling safeClose(). Without the null check, the Java compiler reports that fis might not be initialized. This choice takes advantage of Java's ability to detect uninitialized variables. If fis is initialized to null in a more complex method, cases in which fis is used without being initialized will not be detected by the compiler.

ConfigProvider.java, line 19 (Unreleased Resource: Streams)

Fortify Priority:	High	**Folder**	High
Kingdom:	Code Quality		
Abstract:	The function ConfigProvider() in ConfigProvider.java sometimes fails to release a system resource allocated by FileReader() on line 19.		
Sink:	ConfigProvider.java:19 reader = new BufferedReader(new java.io.FileReader())		

```
17          BufferedReader reader;
18          try {
19              reader = new BufferedReader(new FileReader(propertyFilePath));
20              properties = new Properties();
21              try {
```

Detailed Project Summary

Files Scanned

Code base location: /Users/saurabh.joshi/Work/api_automation/src/main/java

Files Scanned:

configProvider/ConfigProvider.java java 13 Lines 1.2 KB 15 Jul, 2019 3:44:35 PM
configProvider/setBaseURI.java java 24 Lines 25 Jul, 2019 12:44:53 PM
flipkart/enquiryFK.java java 11 Lines 15 Jul, 2019 3:44:35 PM
flipkart/enquiryForRefundFK.java java 11 Lines 15 Jul, 2019 3:44:35 PM
flipkart/initiateFK.java java 9 Lines 15 Jul, 2019 3:44:35 PM
flipkart/payFK.java java 13 Lines 1.2 KB 15 Jul, 2019 3:44:35 PM
flipkart/refundFK.java java 12 Lines 1.1 KB 15 Jul, 2019 3:44:35 PM
flipkart/resendOTPFK.java java 11 Lines 15 Jul, 2019 3:44:35 PM
flipkart/runEnquiryFK.java java 7 Lines 15 Jul, 2019 3:44:35 PM
flipkart/runEnquiryForRefundFK.java java 6 Lines 15 Jul, 2019 3:44:35 PM
flipkart/runInitiateFK.java java 8 Lines 15 Jul, 2019 3:44:35 PM
flipkart/runPayFK.java java 5 Lines 15 Jul, 2019 3:44:35 PM
flipkart/runRefundFK.java java 5 Lines 15 Jul, 2019 3:44:35 PM
flipkart/runResendOTPFK.java java 5 Lines 15 Jul, 2019 3:44:35 PM
flipkart/testFlipkartAdhoc.java java 9 Lines 15 Jul, 2019 3:44:35 PM
flipkart/testFlipkartError.java java 54 Lines 5.7 KB 15 Jul, 2019 3:44:35 PM
flipkart/testFlipkartMain.java java 29 Lines 2.3 KB 15 Jul, 2019 3:44:35 PM
lazyPay/createDisputeLP.java java 11 Lines 1.1 KB 15 Jul, 2019 3:44:35 PM
lazyPay/eligibilityLP.java java 10 Lines 15 Jul, 2019 3:44:35 PM
lazyPay/initiateLP.java java 13 Lines 1.2 KB 15 Jul, 2019 3:44:35 PM
lazyPay/merchantRefundLP.java java 10 Lines 1.1 KB 15 Jul, 2019 3:44:35 PM
lazyPay/payLP.java java 14 Lines 1.1 KB 15 Jul, 2019 3:44:35 PM
lazyPay/payWithTokenLP.java java 16 Lines 1.4 KB 15 Jul, 2019 3:44:35 PM
lazyPay/resolveDisputeLP.java java 9 Lines 15 Jul, 2019 3:44:35 PM
lazyPay/runEligibilityLP.java java 7 Lines 15 Jul, 2019 3:44:35 PM
lazyPay/runInitiateLP.java java 6 Lines 15 Jul, 2019 3:44:35 PM
lazyPay/runPayLP.java java 8 Lines 15 Jul, 2019 3:44:35 PM
lazyPay/runSendOTPForTokenLP.java java 4 Lines 15 Jul, 2019 3:44:35 PM
lazyPay/runValidateOTPForTokenLP.java java 5 Lines 15 Jul, 2019 3:44:35 PM
lazyPay/sendOTPForTokenLP.java java 9 Lines 15 Jul, 2019 3:44:35 PM
lazyPay/testLazyPayAdhoc.java java 13 Lines 1.1 KB 15 Jul, 2019 3:44:35 PM
lazyPay/testLazyPayMain.java java 46 Lines 3.4 KB 15 Jul, 2019 3:44:35 PM
lazyPay/validateOTPForTokenLP.java java 13 Lines 1.1 KB 15 Jul, 2019 3:44:35 PM
loanDisbursal/calculateEMI.java java 7 Lines 15 Jul, 2019 3:44:35 PM
loanDisbursal/showEMI.java java 4 Lines 15 Jul, 2019 3:44:35 PM
loanDisbursal/testLoanDisbursalMain.java java 5 Lines 15 Jul, 2019 3:44:35 PM
payloads/FetchOTPRequest.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/FlipkartInitiateRequest.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/FlipkartPayRequest.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/FlipkartRefundRequest.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/FlipkartResendOTPRequest.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/LPDisputeCreate.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/LPDisputeResolve.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/LPEligibilityRequest.java java 2 Lines 1.1 KB 15 Jul, 2019 3:44:35 PM
payloads/LPInitiateRequest.java java 2 Lines 1.5 KB 15 Jul, 2019 3:44:35 PM
payloads/LPMerchantRefund.java java 2 Lines 15 Jul, 2019 3:44:35 PM

payloads/LPPayRequest.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/OTPTokenRequest.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/S2SEligibilityRequest.java java 2 Lines 1.1 KB 15 Jul, 2019 3:44:35 PM
payloads/S2SInitiateRequest.java java 2 Lines 1.2 KB 15 Jul, 2019 3:44:35 PM
payloads/S2SPayRequest.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/UpiConfirmTxnRequest.java java 20 Lines 1.3 KB 29 Jul, 2019 1:47:12 PM
payloads/UpiGenerateSmsRequest.java java 8 Lines 25 Jul, 2019 3:33:08 PM
payloads/UpiInitiateTxnRequest.java java 38 Lines 2.5 KB 29 Jul, 2019 1:33:55 PM
payloads/UpiNotifySmsStatusRequest.java java 10 Lines 25 Jul, 2019 3:36:10 PM
payloads/UpiPostIntentRequest.java java 63 Lines 3.9 KB 1 Aug, 2019 4:18:27 PM
payloads/UpiValidateMerchantRequest.java java 10 Lines 25 Jul, 2019 5:13:00 PM
payloads/WebEligibilityRequest.java java 2 Lines 1.2 KB 15 Jul, 2019 3:44:35 PM
payloads/WebInitiateRequest.java java 2 Lines 1.5 KB 15 Jul, 2019 3:44:35 PM
payloads/calculateEMIRequest.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/sendOtpForTokenRequest.java java 2 Lines 15 Jul, 2019 3:44:35 PM
payloads/validateOtpForTokenRequest.java java 2 Lines 15 Jul, 2019 3:44:35 PM
resources/JsonUtil.java java 11 Lines 1.5 KB 25 Jul, 2019 12:44:53 PM
resources/loadConfig.java java 62 Lines 6.1 KB 1 Aug, 2019 12:36:03 PM
resources/returnAccessTokenLP.java java 9 Lines 15 Jul, 2019 3:44:35 PM
resources/returnArrayResponseParams.java java 8 Lines 15 Jul, 2019 3:44:35 PM
resources/returnOTP.java java 18 Lines 1.6 KB 15 Jul, 2019 3:44:35 PM
resources/returnResource.java java 103 Lines 3.1 KB 29 Jul, 2019 6:43:45 PM
resources/returnResponseParams.java java 43 Lines 3.1 KB 31 Jul, 2019 7:32:00 PM
resources/returnSignature.java java 9 Lines 15 Jul, 2019 3:44:35 PM
resources/returnSignatureData.java java 74 Lines 3.4 KB 15 Jul, 2019 3:44:35 PM
resources/returnTxnID.java java 20 Lines 15 Jul, 2019 3:44:35 PM
resources/validateError.java java 6 Lines 25 Jul, 2019 12:44:53 PM
s2s/eligibilityS2S.java java 9 Lines 15 Jul, 2019 3:44:35 PM
s2s/initiateS2S.java java 10 Lines 15 Jul, 2019 3:44:35 PM
s2s/payS2S.java java 11 Lines 1 KB 15 Jul, 2019 3:44:35 PM
s2s/runEligibilityS2S.java java 6 Lines 15 Jul, 2019 3:44:35 PM
s2s/runInitiateS2S.java java 5 Lines 15 Jul, 2019 3:44:35 PM
s2s/runPayS2S.java java 5 Lines 15 Jul, 2019 3:44:35 PM
s2s/testS2SMain.java java 24 Lines 2 KB 15 Jul, 2019 3:44:35 PM
upi/GenerateSmsApi.java java 6 Lines 25 Jul, 2019 3:38:40 PM
upi/GetIntentApi.java java 9 Lines 25 Jul, 2019 3:22:39 PM
upi/NotifySmsStatusApi.java java 6 Lines 25 Jul, 2019 3:38:40 PM
upi/PostIntentApi.java java 6 Lines 31 Jul, 2019 6:27:16 PM
upi/RunGenerateSmsApi.java java 10 Lines 1.2 KB 25 Jul, 2019 3:38:40 PM
upi/RunGetIntentApi.java java 8 Lines 25 Jul, 2019 3:28:02 PM
upi/RunNotifySmsStatusApi.java java 13 Lines 1.5 KB 25 Jul, 2019 3:38:40 PM
upi/RunPostIntentApi.java java 22 Lines 5.6 KB 1 Aug, 2019 12:36:03 PM
upi/RunStatusApi.java java 9 Lines 25 Jul, 2019 12:44:53 PM
upi/RunUpiConfirmTxnApi.java java 6 Lines 29 Jul, 2019 1:52:25 PM
upi/RunUpiInitiateTxnApi.java java 6 Lines 29 Jul, 2019 1:27:02 PM
upi/RunUpiTxnIntentCollectRequestApi.java java 7 Lines 29 Jul, 2019 6:47:46 PM
upi/RunValidateMerchantApi.java java 9 Lines 1.2 KB 25 Jul, 2019 12:44:53 PM
upi/StatusApi.java java 9 Lines 25 Jul, 2019 12:44:53 PM
upi/TestUpiRegistrationError.java java 106 Lines 12.5 KB 25 Jul, 2019 3:58:54 PM
upi/TestUpiRegistrationMain.java java 105 Lines 8.8 KB 1 Aug, 2019 2:10:15 PM
upi/TestUpiTransactionError.java java 351 Lines 43.6 KB 1 Aug, 2019 6:28:16 PM

upi/TestUpiTransactionMain.java java 263 Lines 23 KB 1 Aug, 2019 4:20:13 PM
upi/Tests.java java 25 Lines 4.3 KB 31 Jul, 2019 7:52:45 PM
upi/UpiConfirmTxnApi.java java 6 Lines 29 Jul, 2019 6:44:01 PM
upi/UpiInitiateTxnApi.java java 6 Lines 29 Jul, 2019 6:44:01 PM
upi/UpiTxnIntentCollectRequestApi.java java 11 Lines 29 Jul, 2019 6:46:55 PM
upi/ValidateMerchantApi.java java 6 Lines 25 Jul, 2019 12:44:53 PM
webflow/EligibilityWeb.java java 10 Lines 1.1 KB 15 Jul, 2019 3:44:35 PM
webflow/InitiateWeb.java java 11 Lines 1.2 KB 15 Jul, 2019 3:44:35 PM
webflow/RunEligibilityWeb.java java 7 Lines 15 Jul, 2019 3:44:35 PM
webflow/RunInitiateWeb.java java 7 Lines 15 Jul, 2019 3:44:35 PM
webflow/TestWebFlowMain.java java 10 Lines 1 KB 15 Jul, 2019 3:44:35 PM

Reference Elements

Classpath:

/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/charsets.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/ext/cldrdata.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/ext/dnsns.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/ext/jaccess.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/ext/jfxrt.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/ext/localedata.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/ext/nashorn.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/ext/sunec.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/ext/sunjce_provider.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/ext/sunpkcs11.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/ext/zipfs.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/jce.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/jfr.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/jsse.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/resources.jar
/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/jre/lib/rt.jar
/System/Library/Java/Extensions/MRJToolkit.jar
/Users/saurabh.joshi/.m2/repository/com/beust/jcommander/1.72/jcommander-1.72.jar
/Users/saurabh.joshi/.m2/repository/com/fasterxml/jackson/core/jackson-annotations/2.9.0/jackson-annotations-2.9.0.jar
/Users/saurabh.joshi/.m2/repository/com/fasterxml/jackson/core/jackson-core/2.9.9/jackson-core-2.9.9.jar
/Users/saurabh.joshi/.m2/repository/com/fasterxml/jackson/core/jackson-databind/2.9.9/jackson-databind-2.9.9.jar
/Users/saurabh.joshi/.m2/repository/com/sun/xml/bind/jaxb-osgi/2.2.10/jaxb-osgi-2.2.10.jar
/Users/saurabh.joshi/.m2/repository/commons-codec/commons-codec/1.9/commons-codec-1.9.jar
/Users/saurabh.joshi/.m2/repository/commons-logging/commons-logging/1.2/commons-logging-1.2.jar
/Users/saurabh.joshi/.m2/repository/io/rest-assured/json-path/4.0.0/json-path-4.0.0.jar
/Users/saurabh.joshi/.m2/repository/io/rest-assured/rest-assured-common/4.0.0/rest-assured-common-4.0.0.jar
/Users/saurabh.joshi/.m2/repository/io/rest-assured/rest-assured/4.0.0/rest-assured-4.0.0.jar
/Users/saurabh.joshi/.m2/repository/io/rest-assured/xml-path/4.0.0/xml-path-4.0.0.jar
/Users/saurabh.joshi/.m2/repository/javax/activation/activation/1.1.1/activation-1.1.1.jar
/Users/saurabh.joshi/.m2/repository/javax/xml/bind/jaxb-api/2.2.12/jaxb-api-2.2.12.jar
/Users/saurabh.joshi/.m2/repository/junit/junit/4.12/junit-4.12.jar
/Users/saurabh.joshi/.m2/repository/org/apache-extras/beanshell/bsh/2.0b6/bsh-2.0b6.jar
/Users/saurabh.joshi/.m2/repository/org/apache/commons/commons-lang3/3.4/commons-lang3-3.4.jar
/Users/saurabh.joshi/.m2/repository/org/apache/httpcomponents/httpclient/4.5.3/httpclient-4.5.3.jar
/Users/saurabh.joshi/.m2/repository/org/apache/httpcomponents/httpcore/4.4.6/httpcore-4.4.6.jar

/Users/saurabh.joshi/.m2/repository/org/apache/httpcomponents/httpmime/4.5.3/httpmime-4.5.3.jar
/Users/saurabh.joshi/.m2/repository/org/apache/sling/org.apache.sling.javax.activation/0.1.0/org.apache.sling.javax.activation-0.1.0.jar
/Users/saurabh.joshi/.m2/repository/org/ccil/cowan/tagsoup/tagsoup/1.2.1/tagsoup-1.2.1.jar
/Users/saurabh.joshi/.m2/repository/org/codehaus/groovy/groovy-json/2.5.6/groovy-json-2.5.6.jar
/Users/saurabh.joshi/.m2/repository/org/codehaus/groovy/groovy-xml/2.5.6/groovy-xml-2.5.6.jar
/Users/saurabh.joshi/.m2/repository/org/codehaus/groovy/groovy/2.5.6/groovy-2.5.6.jar
/Users/saurabh.joshi/.m2/repository/org/hamcrest/hamcrest-core/2.1/hamcrest-core-2.1.jar
/Users/saurabh.joshi/.m2/repository/org/hamcrest/hamcrest-library/2.1/hamcrest-library-2.1.jar
/Users/saurabh.joshi/.m2/repository/org/hamcrest/hamcrest/2.1/hamcrest-2.1.jar
/Users/saurabh.joshi/.m2/repository/org/json/json/20180813/json-20180813.jar
/Users/saurabh.joshi/.m2/repository/org/testng/testng/6.14.3/testng-6.14.3.jar

Libdirs:

No libdirs specified during translation

Rulepacks

Valid Rulepacks:

Name: Fortify Secure Coding Rules, Extended, Java
Version: 2019.2.0.0009
ID: AAAC0B10-79E7-4FE5-9921-F4903A79D317
SKU: RUL13007

Name: Fortify Secure Coding Rules, Extended, JSP
Version: 2019.2.0.0009
ID: 00403342-15D0-48C9-8E67-4B1CFBDEFCD2
SKU: RUL13026

Name: Fortify Secure Coding Rules, Core, Android
Version: 2019.2.0.0009
ID: FF9890E6-D119-4EE8-A591-83DCF4CA6952
SKU: RUL13093

Name: Fortify Secure Coding Rules, Extended, Content
Version: 2019.2.0.0009
ID: 9C48678C-09B6-474D-B86D-97EE94D38F17
SKU: RUL13067

Name: Fortify Secure Coding Rules, Extended, Configuration
Version: 2019.2.0.0009
ID: CD6959FC-0C37-45BE-9637-BAA43C3A4D56
SKU: RUL13005

Name: Fortify Secure Coding Rules, Core, Annotations
Version: 2019.2.0.0009
ID: 14EE50EB-FA1C-4AE8-8B59-39F952E21E3B
SKU: RUL13078

Name: Fortify Secure Coding Rules, Core, Java
Version: 2019.2.0.0009
ID: 06A6CC97-8C3F-4E73-9093-3E74C64A2AAF
SKU: RUL13003

External Metadata:
Version: 2019.2.0.0009

Name: CWE
ID: 3ADB9EE4-5761-4289-8BD3-CBFCC593EBBC
The Common Weakness Enumeration (CWE), co-sponsored and maintained by MITRE, is international in scope and free for public use. CWE provides a unified, measurable set of software weaknesses that is enabling more effective discussion, description, selection, and use of software security tools and services that can find these weaknesses in source code and operational systems as well as better understanding and management of software weaknesses related to architecture and design.

Name: DISA CCI 2
ID: 7F037130-41E5-40F0-B653-7819A4B3E241
The purpose of a Defense Information Systems Agency (DISA) Control Correlation Identifier (CCI) is to provide a standard identifier for policy based requirements which connect high-level policy expressions and low-level technical implementations. Associated with each CCI is a description for each of the singular, actionable, statements compromising an information assurance (IA) control or IA best practice. Using CCI allows high-level policy framework security requirements to be decomposed and explicitly associated with low-level implementations, thus enabling the assessment of related compliance assessment results spanning heterogeneous technologies. The current IA controls and best practices associated with each CCI, that are specified in NIST SP 800-53 Revision 4, can be viewed using the DISA STIG Viewer.
The following table summarizes the number of issues identified across the different CCIs broken down by Fortify Priority Order. The status of a CCI is considered "In Place" when there are no issues reported for a given CCI.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, CCI-003187 is not considered "In Place". Similarly, if the project is missing a Micro Focus Fortify WebInspect scan, or the scan contains any critical findings, CCI-000366 and CCI-000256 are not considered "In Place".

Name: FISMA
ID: B40F9EE0-3824-4879-B9FE-7A789C89307C
The Federal Information Processing Standard (FIPS) 200 document is part of the official series of publications, issued by the National Institute of Standards and Technology (NIST), relating to standards and guidelines adopted and promulgated under the provisions of the Federal Information Security Management Act (FISMA). Specifically, FIPS Publication 200 specifies the "Minimum Security Requirements for Federal Information and Information Systems."

Name: GDPR
ID: 771C470C-9274-4580-8556-C12F5E4BEC51
The EU General Data Protection Regulation (GDPR) replaces the Data Protection Directive 95/46/EC and was designed to harmonize data privacy laws across Europe, to protect and empower all EU citizens data privacy and to reshape the way organizations across the region approach data privacy. Going into effect on May 25, 2018, GDPR provides a framework for organizations on how to handle personal data. According to GDPR regulation personal data "means any information relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person." GDPR articles that pertain to application security and require businesses to protect personal data during design and development of its product and services are:

- Article 25, Data protection by design and by default - which requires "The controller shall implement appropriate technical and organisational measures for ensuring that, by default, only personal data which are necessary for each specific purpose of the processing are processed."
- Article 32, Security of processing - which requires businesses to protect its systems and applications "from accidental or unlawful destruction, loss, alteration, unauthorized disclosure of, or access to personal data". This report may be used by organizations as a framework to help identify and protect personal data as it relates to application security.

Name: MISRA C 2012
ID: 555A3A66-A0E1-47AF-910C-3F19A6FB2506
Now in its third edition, the Motor Industry Software Reliability Association (MISRA) C Guidelines describe a subset of the C programming language in which there is reduced risk of introducing mistakes in critical systems. While the MISRA C Guidelines focus upon safety-related software development, a subset of the rules also reflect security properties. Fortify interprets the MISRA C Guidelines under the context of security and provides correlation of security vulnerability categories to the rules defined by MISRA. Fortify provides these security focused detection mechanism with the standard rulepacks, however, further support of the MISRA C Guidelines related to safety can be added through the use of custom rules. The results in this report can assist in the creation of a compliance matrix for MISRA.

Name: MISRA C++ 2008
ID: 5D4B75A1-FC91-4B4B-BD4D-C81BBE9604FA
The Motor Industry Software Reliability Association (MISRA) C++ Guidelines describe a subset of the C++ programming language in which there is reduced risk of introducing mistakes in critical systems. While the MISRA C++ Guidelines focus upon safety-related software development, a subset of the rules also reflect security properties. Fortify interprets the MISRA C++ Guidelines under the context of security and provides correlation of security vulnerability categories to the rules defined by MISRA. Fortify provides these security focused detection mechanism with the standard rulepacks, however, further support of the MISRA C++ Guidelines related to safety can be added through the use of custom rules. The results in this report can assist in the creation of a compliance matrix for MISRA.

Name: NIST SP 800-53 Rev.4
ID: 1114583B-EA24-45BE-B7F8-B61201BACDD0
NIST Special Publication 800-53 Revision 4 provides a list of security and privacy controls designed to protect federal organizations and information systems from security threats. The following table summarizes the number of issues identified across the different controls and broken down by Fortify Priority Order.

Name: OWASP Mobile 2014
ID: EEE3F9E7-28D6-4456-8761-3DA56C36F4EE
The OWASP Mobile Top 10 Risks 2014 provides a powerful awareness document for mobile application security. The OWASP Mobile Top 10 represents a broad consensus about what the most critical mobile application security flaws are. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: OWASP Top 10 2004
ID: 771C470C-9274-4580-8556-C023E4D3ADB4
The OWASP Top Ten 2004 provides a powerful awareness document for web application security. The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: OWASP Top 10 2007
ID: 1EB1EC0E-74E6-49A0-BCE5-E6603802987A
The OWASP Top Ten 2007 provides a powerful awareness document for web application security. The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: OWASP Top 10 2010
ID: FDCECA5E-C2A8-4BE8-BB26-76A8ECD0ED59
The OWASP Top Ten 2010 provides a powerful awareness document for web application security. The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: OWASP Top 10 2013
ID: 1A2B4C7E-93B0-4502-878A-9BE40D2A25C4
The OWASP Top Ten 2013 provides a powerful awareness document for web application security. The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: OWASP Top 10 2017
ID: 3C6ECB67-BBD9-4259-A8DB-B49328927248
The OWASP Top Ten 2017 provides a powerful awareness document for web application security focused on informing the community about the consequences of the most common and most important web application security weaknesses. The OWASP Top Ten represents a broad agreement about what the most critical web application security flaws are with consensus being drawn from data collection and survey results. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: PCI 1.1
ID: CBDB9D4D-FC20-4C04-AD58-575901CAB531
The Payment Card Industry (PCI) Data Security Standard (DSS) 1.1 compliance standard describes 12 requirements which are organized into 6 logically related groups, which are "control objectives". PCI DSS requirements are applicable if Primary Account Number (PAN) is stored, processed, or transmitted by the system.

Name: PCI 1.2
ID: 57940BDB-99F0-48BF-BF2E-CFC42BA035E5
Payment Card Industry Data Security Standard Version 1.2 description

Name: PCI 2.0
ID: 8970556D-7F9F-4EA7-8033-9DF39D68FF3E
The PCI DSS 2.0 compliance standard, particularly sections 6.3, 6.5, and 6.6, references the OWASP Top 10 vulnerability categories as the core categories that must be tested for and remediated. The following table summarizes the number of issues identified across the different PCI DSS requirements and broken down by Fortify Priority Order.

Name: PCI 3.0
ID: E2FB0D38-0192-4F03-8E01-FE2A12680CA3
The following is a summary of the application security portions of Payment Card Industry (PCI) Data Security Standard (DSS) v3.0. Fortify tests for 32 application security related requirements across sections 1, 2, 3, 4, 6, 7, 8, and 10 of PCI DSS and reports whether each requirement is In Place or Not In Place to indicate whether requirements are satisfied or not. This report is intended to measure the level of adherence the specific application(s) possess when compared to PCI DSS 3.0 compliance and is not intended to serve as a comprehensive Report on Compliance (ROC). The information contained in this report is targeted at project managers, security auditors, and compliance auditors.

Name: PCI 3.1
ID: AC0D18CF-C1DA-47CF-9F1A-E8EC0A4A717E
The following is a summary of the application security portions of Payment Card Industry (PCI) Data Security Standard (DSS) v3.1. Fortify tests for 31 application security related requirements across sections 1, 2, 3, 4, 6, 7, 8, and 10 of PCI DSS and reports whether each requirement is In Place or Not In Place to indicate whether requirements are satisfied or not. This report is intended to measure the level of adherence the specific application(s) possess when compared to PCI DSS 3.1 compliance and is not

intended to serve as a comprehensive Report on Compliance (ROC). The information contained in this report is targeted at project managers, security auditors, and compliance auditors.

Name: PCI 3.2
ID: 4E8431F9-1BA1-41A8-BDBD-087D5826751A
The following is a summary of the application security portions of Payment Card Industry (PCI) Data Security Standard (DSS) v3.2. Fortify tests for 31 application security related requirements across sections 1, 2, 3, 4, 6, 7, 8, and 10 of PCI DSS and reports whether each requirement is In Place or Not In Place to indicate whether requirements are satisfied or not. This report is intended to measure the level of adherence the specific application(s) possess when compared to PCI DSS 3.2 compliance and is not intended to serve as a comprehensive Report on Compliance (ROC). The information contained in this report is targeted at project managers, security auditors, and compliance auditors.

Name: PCI 3.2.1
ID: EADE255F-6561-4EFE-AD31-2914F6BFA329
The following is a summary of the application security portions of Payment Card Industry (PCI) Data Security Standard (DSS) v3.2.1. Fortify tests for 31 application security related requirements across sections 1, 2, 3, 4, 6, 7, 8, and 10 of PCI DSS and reports whether each requirement is In Place or Not In Place to indicate whether requirements are satisfied or not. This report is intended to measure the level of adherence the specific application(s) possess when compared to PCI DSS 3.2.1 compliance and is not intended to serve as a comprehensive Report on Compliance (ROC). The information contained in this report is targeted at project managers, security auditors, and compliance auditors.

Name: PCI SSF 1.0
ID: 0F551543-AF0E-4334-BEDF-1DDCD5F4BF74
The following is a summary of the application security portions of the Secure Software Requirements and Assessment Procedures defined in the Payment Card Industry (PCI) Software Security Framework (SSF) v1.0. Fortify tests for 23 application security related control objectives across Control Objective sections 2, 3, 4, 5, 6, 7, 8, and A.2 of PCI SSF and reports whether each control objective is In Place or Not In Place to indicate whether requirements are satisfied or not. This report is intended to measure the level of adherence the specific application(s) possess when compared to PCI SSF 1.0 compliance and is not intended to serve as a comprehensive Report on Compliance (ROC). The information contained in this report is targeted at project managers, security auditors, and compliance auditors.

Name: SANS Top 25 2009
ID: 939EF193-507A-44E2-ABB7-C00B2168B6D8
The 2009 CWE/SANS Top 25 Programming Errors list the most significant programming errors that can lead to serious software vulnerabilities. They occur frequently, are often easy to find, and easy to exploit. They are dangerous because they will frequently allow attackers to completely take over the software, steal data, or prevent the software from working at all. The list is the result of collaboration between the SANS Institute, MITRE, and many top software security experts.

Name: SANS Top 25 2010
ID: 72688795-4F7B-484C-88A6-D4757A6121CA
SANS Top 25 2010 Most Dangerous Software Errors provides an enumeration of the most widespread and critical errors, categorized by Common Weakness Enumeration (CWE) identifiers, that lead to serious vulnerabilities in software (http://cwe.mitre.org/). These software errors are often easy to find and exploit. The inherent danger in these errors is that they can allow an attacker to completely take over the software, steal data, or prevent the software from working at all.

Name: SANS Top 25 2011
ID: 92EB4481-1FD9-4165-8E16-F2DE6CB0BD63
SANS Top 25 2011 Most Dangerous Software Errors provides an enumeration of the most widespread and critical errors, categorized by Common Weakness Enumeration (CWE) identifiers, that lead to serious vulnerabilities in software (http://cwe.mitre.org/). These software errors are often easy to find and exploit. The inherent danger in these errors is that they can allow an attacker to completely take over the software, steal data, or prevent the software from working at all.

Name: STIG 3.1
ID: F2FA57EA-5AAA-4DDE-90A5-480BE65CE7E7
Security Technical Implementation Guide Version 3.1 description

Name: STIG 3.10
ID: 788A87FE-C9F9-4533-9095-0379A9B35B12
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APP5080: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APP5100: CAT II is not considered "In Place".

Name: STIG 3.4
ID: 58E2C21D-C70F-4314-8994-B859E24CF855
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG identifies several severities with respect to vulnerabilities:

<LI>CAT I: allow an attacker immediate access into a machine, allow super user access, or bypass a firewall.</LI>
<LI>CAT II: provide information that have a high potential of giving access to an intruder.</LI>
<LI>CAT III: provide information that potentially could lead to compromise.</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

Name: STIG 3.5
ID: DD18E81F-3507-41FA-9DFA-2A9A15B5479F
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG identifies several severities with respect to vulnerabilities:

<LI>CAT I: allow an attacker immediate access into a machine, allow super user access, or bypass a firewall.</LI>
<LI>CAT II: provide information that have a high potential of giving access to an intruder.</LI>
<LI>CAT III: provide information that potentially could lead to compromise.</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

Name: STIG 3.6
ID: 000CA760-0FED-4374-8AA2-6FA3968A07B1
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a

STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG identifies several severities with respect to vulnerabilities:

<LI>CAT I: allow an attacker immediate access into a machine, allow super user access, or bypass a firewall.</LI>
<LI>CAT II: provide information that have a high potential of giving access to an intruder.</LI>
<LI>CAT III: provide information that potentially could lead to compromise.</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APP5080: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APP5100: CAT II is not considered "In Place".

Name: STIG 3.7
ID: E69C07C0-81D8-4B04-9233-F3E74167C3D2
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG identifies several severities with respect to vulnerabilities:

<LI>CAT I: allow an attacker immediate access into a machine, allow super user access, or bypass a firewall.</LI>
<LI>CAT II: provide information that have a high potential of giving access to an intruder.</LI>
<LI>CAT III: provide information that potentially could lead to compromise.</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APP5080: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APP5100: CAT II is not considered "In Place".

Name: STIG 3.9
ID: 1A9D736B-2D4A-49D1-88CA-DF464B40D732
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APP5080: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APP5100: CAT II is not considered "In Place".

Name: STIG 4.1
ID: 95227C50-A9E4-4C9D-A8AF-FD98ABAE1F3C
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a

STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.2
ID: 672C15F8-8822-4E05-8C9E-1A4BAAA7A373
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.3
ID: A0B313F0-29BD-430B-9E34-6D10F1178506
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.4
ID: ECEC5CA2-7ACA-4B70-BF44-3248B9C6F4F8
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.5
ID: E6010E0A-7F71-4388-B8B7-EE9A02143474
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.6
ID: EFB9B012-44D6-456D-B197-03D2FD7C7AD6
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify

WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.7
ID: B04A1E01-F1C1-48D3-A827-0F70872182D7
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.8
ID: E6805D9F-D5B5-4192-962C-46828FF68507
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.9
ID: 7B9F7B3B-07FC-4B61-99A1-70E3BB23A6A0
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: WASC 2.00
ID: 74f8081d-dd49-49da-880f-6830cebe9777
The Web Application Security Consortium (WASC) was created as a cooperative effort to standardize, clarify, and organize the threats to the security of a web site. Version 2.00 of their Threat Classification outlines the attacks and weaknesses that can commonly lead to a website being compromised.

Name: WASC 24 + 2
ID: 9DC61E7F-1A48-4711-BBFD-E9DFF537871F
The Web Application Security Consortium (WASC) was created as a cooperative effort to standardize, clarify, and organize the threats to the security of a web site.

Properties

WinForms.CollectionMutationMonitor.Label=WinFormsDataSource
awt.toolkit=sun.lwawt.macosx.LWCToolkit
com.fortify.AuthenticationKey=/Users/saurabh.joshi/.fortify/config/tools
com.fortify.Core=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/Core
com.fortify.InstallRoot=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0
com.fortify.InstallationUserName=saurabh.joshi
com.fortify.SCAExecutablePath=
com.fortify.TotalPhysicalMemory=8589934592
com.fortify.VS.RequireASPPrecompilation=true
com.fortify.WorkingDirectory=/Users/saurabh.joshi/.fortify
com.fortify.locale=en
com.fortify.sca.AddImpliedMethods=true
com.fortify.sca.AntCompilerClass=com.fortify.dev.ant.SCACompiler
com.fortify.sca.AppendLogFile=true
com.fortify.sca.BuildID=1
com.fortify.sca.BundleControlflowIssues=true
com.fortify.sca.BytecodePreview=true
com.fortify.sca.CollectPerformanceData=true
com.fortify.sca.CustomRulesDir=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/Core/config/customrules
com.fortify.sca.DaemonCompilers=com.fortify.sca.util.compilers.GppCompiler,com.fortify.sca.util.compilers.GccCompiler,com.fortify.sca.util.compilers.AppleGppCompiler,com.fortify.sca.util.compilers.AppleGccCompiler,com.fortify.sca.util.compilers.MicrosoftCompiler,com.fortify.sca.util.compilers.MicrosoftLinker,com.fortify.sca.util.compilers.LdCompiler,com.fortify.sca.util.compilers.ArUtil,com.fortify.sca.util.compilers.SunCCompiler,com.fortify.sca.util.compilers.SunCppCompiler,com.fortify.sca.util.compilers.IntelCompiler,com.fortify.sca.util.compilers.ExternalCppAdapter,com.fortify.sca.util.compilers.ClangCompiler
com.fortify.sca.DeadCodeFilter=true
com.fortify.sca.DeadCodeIgnoreTrivialPredicates=true
com.fortify.sca.DefaultAnalyzers=semantic:dataflow:controlflow:nullptr:configuration:content:structural:buffer
com.fortify.sca.DefaultFileTypes=java,rb,erb,jsp,jspx,jspf,tag,tagx,tld,sql,cfm,php,phtml,ctp,pks,pkh,pkb,xml,config,Config,settings,properties,dll,exe,winmd,cs,vb,asax,ascx,ashx,asmx,aspx,master,Master,xaml,baml,cshtml,vbhtml,inc,asp,vbscript,js,ini,bas,cls,vbs,frm,ctl,html,htm,xsd,wsdd,xmi,py,cfml,cfc,abap,xhtml,cpx,xcfg,jsff,as,mxml,cbl,cscfg,csdef,wadcfg,wadcfgx,appxmanifest,wsdl,plist,bsp,ABAP,BSP,swift,page,trigger,scala,ts,conf,json,yaml,yml

com.fortify.sca.DefaultJarsDirs=default_jars
com.fortify.sca.DefaultRulesDir=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/Core/config/rules
com.fortify.sca.DisableDeadCodeElimination=false
com.fortify.sca.DisableFunctionPointers=false
com.fortify.sca.DisableGlobals=false
com.fortify.sca.DisableInferredConstants=false
com.fortify.sca.EnableInterproceduralConstantResolution=true
com.fortify.sca.EnableNestedWrappers=true
com.fortify.sca.EnableStructuralMatchCache=true
com.fortify.sca.EnableWrapperDetection=true
com.fortify.sca.FVDLDisableDescriptions=false
com.fortify.sca.FVDLDisableProgramData=false
com.fortify.sca.FVDLDisableSnippets=false
com.fortify.sca.FVDLStylesheet=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/Core/resources/sca/fvdl2html.xsl
com.fortify.sca.IndirectCallGraphBuilders=WinFormsAdHocFunctionBuilder,VirtualCGBuilder,J2EEIndirectCGBuilder,JNICGBuilder,StoredProcedureResolver,JavaWSCGBuilder,StrutsCGBuilder,DotNetWSCGBuilder,SqlServerSPResolver,ASPCGBuilder,ScriptedCGBuilder,NewJspCustomTagCGBuilder,DotNetCABCGBuilder,StateInjectionCGBuilder,SqlServerSPResolver2,PHPLambdaResolver,JavaWebCGBuilder
com.fortify.sca.JVMArgs=-XX:SoftRefLRUPolicyMSPerMB=3000 -Xmx6442450944 -Xss16M
com.fortify.sca.JavaSourcepathSearch=true
com.fortify.sca.JdkVersion=1.8
com.fortify.sca.LogFileDir=/Users/saurabh.joshi/.fortify/sca19.1/log
com.fortify.sca.LogFileExt=.log
com.fortify.sca.LogFileName=sca.log
com.fortify.sca.LogFileNameNoExt=sca
com.fortify.sca.LogFilePath=/Users/saurabh.joshi/.fortify/sca19.1/log/sca.log
com.fortify.sca.LogLevel=INFO
com.fortify.sca.LowSeverityCutoff=1.0
com.fortify.sca.MachineOutputMode=
com.fortify.sca.MultithreadedAnalysis=true
com.fortify.sca.NoNestedOutTagOutput=org.apache.taglibs.standard.tag.rt.core.RemoveTag,org.apache.taglibs.standard.tag.rt.core.SetTag
com.fortify.sca.OldVbNetExcludeFileTypes=vb,asax,ascx,ashx,asmx,aspx,xaml,cshtml,vbhtml
com.fortify.sca.PID=67621
com.fortify.sca.Phase0HigherOrder.Languages=python,ruby,swift,javascript,typescript
com.fortify.sca.Phase0HigherOrder.Level=1
com.fortify.sca.PrintPerformanceDataAfterScan=false
com.fortify.sca.ProjectRoot=/Users/saurabh.joshi/.fortify
com.fortify.sca.ProjectRoot=/Users/saurabh.joshi/.fortify
com.fortify.sca.Renderer=fpr
com.fortify.sca.RequireMapKeys=classrule
com.fortify.sca.ResultsFile=/Users/saurabh.joshi/Work/api_automation/1_scan.fpr
com.fortify.sca.SolverTimeout=15
com.fortify.sca.SqlLanguage=PLSQL
com.fortify.sca.SuppressLowSeverity=true
com.fortify.sca.ThreadCount.NameTableLoading=1
com.fortify.sca.TypeInferenceFunctionTimeout=60
com.fortify.sca.TypeInferenceLanguages=javascript,typescript,python,ruby
com.fortify.sca.TypeInferencePhase0Timeout=300
com.fortify.sca.UnicodeInputFile=true
com.fortify.sca.UniversalBlacklist=.*yyparse.*

com.fortify.sca.alias.mode.csharp=fs
com.fortify.sca.alias.mode.javascript=fi
com.fortify.sca.alias.mode.scala=fi
com.fortify.sca.alias.mode.swift=fi
com.fortify.sca.alias.mode.typescript=fi
com.fortify.sca.alias.mode.vb=fs
com.fortify.sca.analyzer.controlflow.EnableLivenessOptimization=false
com.fortify.sca.analyzer.controlflow.EnableMachineFiltering=false
com.fortify.sca.analyzer.controlflow.EnableRefRuleOptimization=false
com.fortify.sca.analyzer.controlflow.EnableTimeOut=true
com.fortify.sca.clang.AcceptedParams=-arcmt-migrate-report-output+1,-compatibility_version+1,-current_version+1,-cxx-isystem+1,-D+1,-dependency-dot+1,-dependency-file+1,-F+1,-filelist+1,-flts-model+1,-fmodule-implementation-of+1,-framework+1,-gmodules,-idirafter+1,-iframework+1,-imacros+1,-imultilib+1,-include+1,-include-pch+1,-install_name+1,-iprefix+1,-iquote+1,-isystem+1,-ivfsoverlay+1,-iwithprefix+1,-iwithprefixbefore+1,-iwithsysroot+1,-I+1,-MF+1,-MMD,-MQ+1,-MT+1,-mllvm+1,-o+1,--serialize-diagnostics+1,-U+1,-undefined+1,-working-directory+1,-X*+1,-x+1
com.fortify.sca.clang.SuppressedParams=--analyze,-e+1,-fapplication-extension,-fembed-bitcode,-fembed-bitcode-marker,-l*,-l+1,-L*,-L+1,-module-file-info+1,-mwatchos-version-min,--param+1,-w,-weak_framework+1,-Xanalyzer+1,-Xassembler+1,-Xlinker+1
com.fortify.sca.compilers.ant=com.fortify.sca.util.compilers.AntAdapter
com.fortify.sca.compilers.ar=com.fortify.sca.util.compilers.ArUtil
com.fortify.sca.compilers.armcc=com.fortify.sca.util.compilers.ArmCcCompiler
com.fortify.sca.compilers.armcpp=com.fortify.sca.util.compilers.ArmCppCompiler
com.fortify.sca.compilers.c++=com.fortify.sca.util.compilers.AppleGppCompiler
com.fortify.sca.compilers.cc=com.fortify.sca.util.compilers.AppleGccCompiler
com.fortify.sca.compilers.clang=com.fortify.sca.util.compilers.ClangCompiler
com.fortify.sca.compilers.clang++=com.fortify.sca.util.compilers.ClangCompiler
com.fortify.sca.compilers.clearmake=com.fortify.sca.util.compilers.TouchlessCompiler
com.fortify.sca.compilers.fortify=com.fortify.sca.util.compilers.FortifyCompiler
com.fortify.sca.compilers.g++=com.fortify.sca.util.compilers.AppleGppCompiler
com.fortify.sca.compilers.g++-*=com.fortify.sca.util.compilers.AppleGppCompiler
com.fortify.sca.compilers.g++2*=com.fortify.sca.util.compilers.AppleGppCompiler
com.fortify.sca.compilers.g++3*=com.fortify.sca.util.compilers.AppleGppCompiler
com.fortify.sca.compilers.g++4*=com.fortify.sca.util.compilers.AppleGppCompiler
com.fortify.sca.compilers.gcc=com.fortify.sca.util.compilers.AppleGccCompiler
com.fortify.sca.compilers.gcc-*=com.fortify.sca.util.compilers.AppleGccCompiler
com.fortify.sca.compilers.gcc2*=com.fortify.sca.util.compilers.AppleGccCompiler
com.fortify.sca.compilers.gcc3*=com.fortify.sca.util.compilers.AppleGccCompiler
com.fortify.sca.compilers.gcc4*=com.fortify.sca.util.compilers.AppleGccCompiler
com.fortify.sca.compilers.gmake=com.fortify.sca.util.compilers.TouchlessCompiler
com.fortify.sca.compilers.gradle=com.fortify.sca.util.compilers.GradleAdapter
com.fortify.sca.compilers.gradlew=com.fortify.sca.util.compilers.GradleAdapter
com.fortify.sca.compilers.icc=com.fortify.sca.util.compilers.IntelCompiler
com.fortify.sca.compilers.icpc=com.fortify.sca.util.compilers.IntelCompiler
com.fortify.sca.compilers.jam=com.fortify.sca.util.compilers.TouchlessCompiler
com.fortify.sca.compilers.javac=com.fortify.sca.util.compilers.JavacCompiler
com.fortify.sca.compilers.ld=com.fortify.sca.util.compilers.LdCompiler
com.fortify.sca.compilers.make=com.fortify.sca.util.compilers.TouchlessCompiler
com.fortify.sca.compilers.mvn=com.fortify.sca.util.compilers.MavenAdapter
com.fortify.sca.compilers.scalac=com.fortify.sca.util.compilers.ScalacCompiler
com.fortify.sca.compilers.swift=com.fortify.sca.util.compilers.SwiftCompiler
com.fortify.sca.compilers.swiftc=com.fortify.sca.util.compilers.SwiftCompiler

com.fortify.sca.compilers.tcc=com.fortify.sca.util.compilers.ArmCcCompiler
com.fortify.sca.compilers.tcpp=com.fortify.sca.util.compilers.ArmCppCompiler
com.fortify.sca.compilers.touchless=com.fortify.sca.util.compilers.FortifyCompiler
com.fortify.sca.compilers.xcodebuild=com.fortify.sca.util.compilers.XcodebuildScraper
com.fortify.sca.cpfe.441.command=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/Core/private-bin/sca/cpfe441.rfct
com.fortify.sca.cpfe.command=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/Core/private-bin/sca/cpfe48
com.fortify.sca.cpfe.file.option=--gen_c_file_name
com.fortify.sca.cpfe.options=--remove_unneeded_entities --suppress_vtbl -tused
com.fortify.sca.cpfe.options=--remove_unneeded_entities --suppress_vtbl -tused
com.fortify.sca.env.exesearchpath=/usr/bin:/bin:/usr/sbin:/sbin
com.fortify.sca.fileextensions.ABAP=ABAP
com.fortify.sca.fileextensions.BSP=ABAP
com.fortify.sca.fileextensions.Config=XML
com.fortify.sca.fileextensions.abap=ABAP
com.fortify.sca.fileextensions.appxmanifest=XML
com.fortify.sca.fileextensions.as=ACTIONSCRIPT
com.fortify.sca.fileextensions.asp=ASP
com.fortify.sca.fileextensions.bas=VB6
com.fortify.sca.fileextensions.bsp=ABAP
com.fortify.sca.fileextensions.cfc=CFML
com.fortify.sca.fileextensions.cfm=CFML
com.fortify.sca.fileextensions.cfml=CFML
com.fortify.sca.fileextensions.cls=VB6
com.fortify.sca.fileextensions.conf=HOCON
com.fortify.sca.fileextensions.config=XML
com.fortify.sca.fileextensions.cpx=XML
com.fortify.sca.fileextensions.cscfg=XML
com.fortify.sca.fileextensions.csdef=XML
com.fortify.sca.fileextensions.ctl=VB6
com.fortify.sca.fileextensions.ctp=PHP
com.fortify.sca.fileextensions.erb=RUBY_ERB
com.fortify.sca.fileextensions.faces=JSPX
com.fortify.sca.fileextensions.frm=VB6
com.fortify.sca.fileextensions.htm=HTML
com.fortify.sca.fileextensions.html=HTML
com.fortify.sca.fileextensions.ini=JAVA_PROPERTIES
com.fortify.sca.fileextensions.java=JAVA
com.fortify.sca.fileextensions.js=TYPESCRIPT
com.fortify.sca.fileextensions.jsff=JSPX
com.fortify.sca.fileextensions.json=JSON
com.fortify.sca.fileextensions.jsp=JSP
com.fortify.sca.fileextensions.jspf=JSP
com.fortify.sca.fileextensions.jspx=JSPX
com.fortify.sca.fileextensions.jsx=TYPESCRIPT
com.fortify.sca.fileextensions.mxml=MXML
com.fortify.sca.fileextensions.page=VISUAL_FORCE
com.fortify.sca.fileextensions.php=PHP
com.fortify.sca.fileextensions.phtml=PHP
com.fortify.sca.fileextensions.pkb=PLSQL
com.fortify.sca.fileextensions.pkh=PLSQL
com.fortify.sca.fileextensions.pks=PLSQL

com.fortify.sca.fileextensions.plist=XML
com.fortify.sca.fileextensions.properties=JAVA_PROPERTIES
com.fortify.sca.fileextensions.py=PYTHON
com.fortify.sca.fileextensions.rb=RUBY
com.fortify.sca.fileextensions.scala=SCALA
com.fortify.sca.fileextensions.settings=XML
com.fortify.sca.fileextensions.sql=SQL
com.fortify.sca.fileextensions.swift=SWIFT
com.fortify.sca.fileextensions.tag=JSP
com.fortify.sca.fileextensions.tagx=JSP
com.fortify.sca.fileextensions.tld=TLD
com.fortify.sca.fileextensions.trigger=APEX_TRIGGER
com.fortify.sca.fileextensions.ts=TYPESCRIPT
com.fortify.sca.fileextensions.tsx=TYPESCRIPT
com.fortify.sca.fileextensions.vbs=VBSCRIPT
com.fortify.sca.fileextensions.vbscript=VBSCRIPT
com.fortify.sca.fileextensions.wadcfg=XML
com.fortify.sca.fileextensions.wadcfgx=XML
com.fortify.sca.fileextensions.wsdd=XML
com.fortify.sca.fileextensions.wsdl=XML
com.fortify.sca.fileextensions.xcfg=XML
com.fortify.sca.fileextensions.xhtml=JSPX
com.fortify.sca.fileextensions.xmi=XML
com.fortify.sca.fileextensions.xml=XML
com.fortify.sca.fileextensions.xsd=XML
com.fortify.sca.fileextensions.yaml=YAML
com.fortify.sca.fileextensions.yml=YAML
com.fortify.sca.jsp.UseNativeParser=true
com.fortify.sca.parser.python.ignore.module.1=test.badsyntax_future3
com.fortify.sca.parser.python.ignore.module.2=test.badsyntax_future4
com.fortify.sca.parser.python.ignore.module.3=test.badsyntax_future5
com.fortify.sca.parser.python.ignore.module.4=test.badsyntax_future6
com.fortify.sca.parser.python.ignore.module.5=test.badsyntax_future7
com.fortify.sca.parser.python.ignore.module.6=test.badsyntax_future8
com.fortify.sca.parser.python.ignore.module.7=test.badsyntax_future9
com.fortify.sca.parser.python.ignore.module.8=test.badsyntax_nocaret
com.fortify.sca.skip.libraries.AngularJS=angular.js,angular.min.js,angular-animate.js,angular-aria.js,angular_1_router.js,angular-cookies.js,angular-message-format.js,angular-messages.js,angular-mocks.js,angular-parse-ext.js,angular-resource.js,angular-route.js,angular-sanitize.js,angular-touch.js
com.fortify.sca.skip.libraries.ES6=es6-shim.min.js,system-polyfills.js,shims_for_IE.js
com.fortify.sca.skip.libraries.jQuery=jquery.js,jquery.min.js,jquery-migrate.js,jquery-migrate.min.js,jquery-ui.js,jquery-ui.min.js,jquery.mobile.js,jquery.mobile.min.js,jquery.color.js,jquery.color.min.js,jquery.color.svg-names.js,jquery.color.svg-names.min.js,jquery.color.plus-names.js,jquery.color.plus-names.min.js,jquery.tools.min.js
com.fortify.sca.skip.libraries.javascript=bootstrap.js,bootstrap.min.js,typescript.js,typescriptServices.js
com.fortify.sca.skip.libraries.typescript=typescript.d.ts,typescriptServices.d.ts
com.fortify.sca.xcode.Overrides=GCC_PRECOMPILE_PREFIX_HEADER=NO,RUN_CLANG_STATIC_ANALYZER=NO
com.fortify.search.defaultSyntaxVer=2
file.encoding=UTF-8
file.encoding.pkg=sun.io
file.separator=/
ftp.nonProxyHosts=local|*.local|169.254/16|*.169.254/16

gopherProxySet=false
http.nonProxyHosts=local|*.local|169.254/16|*.169.254/16
java.awt.graphicsenv=sun.awt.CGraphicsEnvironment
java.awt.headless=true
java.awt.printerjob=sun.lwawt.macosx.CPrinterJob
java.class.path=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/Core/lib/exe/sca-exe.jar
java.class.version=52.0
java.endorsed.dirs=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre/lib/endorsed
java.ext.dirs=/Users/saurabh.joshi/Library/Java/Extensions:/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre/lib/ext:/Library/Java/Extensions:/Network/Library/Java/Extensions:/System/Library/Java/Extensions:/usr/lib/java
java.home=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre
java.io.tmpdir=/var/folders/sv/h2800t652qn7kymkh_zlflys7f0900/T/
java.library.path=/Users/saurabh.joshi/Library/Java/Extensions:/Library/Java/Extensions:/Network/Library/Java/Extensions:/System/Library/Java/Extensions:/usr/lib/java:.
java.runtime.name=OpenJDK Runtime Environment
java.runtime.version=1.8.0_181-b02
java.specification.name=Java Platform API Specification
java.specification.vendor=Oracle Corporation
java.specification.version=1.8
java.vendor=Azul Systems, Inc.
java.vendor.url=http://www.azulsystems.com/
java.vendor.url.bug=http://www.azulsystems.com/support/
java.version=1.8.0_181
java.vm.info=mixed mode
java.vm.name=OpenJDK 64-Bit Server VM
java.vm.specification.name=Java Virtual Machine Specification
java.vm.specification.vendor=Oracle Corporation
java.vm.specification.version=1.8
java.vm.vendor=Azul Systems, Inc.
java.vm.version=25.181-b02
line.separator=

log4j.configurationFile=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/Core/config/log4j2.xml
log4j.isThreadContextMapInheritable=true
max.file.path.length=255
os.arch=x86_64
os.name=Mac OS X
os.version=10.13.6
path.separator=:
socksNonProxyHosts=local|*.local|169.254/16|*.169.254/16
stderr.isatty=false
stdout.isatty=false
sun.arch.data.model=64
sun.boot.class.path=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre/lib/resources.jar:/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre/lib/rt.jar:/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre/lib/sunrsasign.jar:/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre/lib/jsse.jar:/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre/lib/jce.jar:/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre/lib/charsets.jar:/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre/lib/jfr.jar:/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre/classes
sun.boot.library.path=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/jre/lib
sun.cpu.endian=little
sun.cpu.isalist=

sun.io.unicode.encoding=UnicodeBig
sun.java.command=sourceanalyzer -Djava.awt.headless=true -XX:SoftRefLRUPolicyMSPerMB=3000 -Dcom.fortify.sca.env.exesearchpath=/usr/bin:/bin:/usr/sbin:/sbin -Dcom.fortify.sca.ProjectRoot=/Users/saurabh.joshi/.fortify -Dstdout.isatty=false -Dstderr.isatty=false -Dcom.fortify.sca.PID=67621 -Xmx6442450944 -Dcom.fortify.TotalPhysicalMemory=8589934592 -Xss16M -Dcom.fortify.sca.JVMArgs=-XX:SoftRefLRUPolicyMSPerMB=3000 -Xmx6442450944 -Xss16M -Djava.class.path=/Applications/Fortify/Fortify_SCA_and_Apps_19.1.0/Core/lib/exe/sca-exe.jar -scan @/Users/saurabh.joshi/.fortify/Eclipse.Plugin-19.1.0/1/1Scan.txt
sun.jnu.encoding=UTF-8
sun.management.compiler=HotSpot 64-Bit Tiered Compilers
sun.os.patch.level=unknown
user.country=IN
user.dir=/Applications/Eclipse.app/Contents/MacOS
user.home=/Users/saurabh.joshi
user.language=en
user.name=saurabh.joshi
user.timezone=Asia/Kolkata

Commandline Arguments

-scan
-b
1
-format
fpr
-machine-output
-f
/Users/saurabh.joshi/Work/api_automation/1_scan.fpr

Warnings

[101] File /1/src/test/java not found

Issue Count by Category

Issues by Category	
Poor Error Handling: Overly Broad Throws	214
Poor Style: Non-final Public Static Field	161
Poor Logging Practice: Use of a System Output Stream	45
System Information Leak	9
J2EE Bad Practices: Threads	2
Dead Code: Expression is Always true	1
Denial of Service: Parse Double	1
Insecure Randomness	1
Null Dereference	1
Unreleased Resource: Streams	1

Issue Breakdown by Analysis

Issues by Analysis