Fortify Audit Workbench

Developer Workbook

ManagementPlatformWeb

Table of Contents

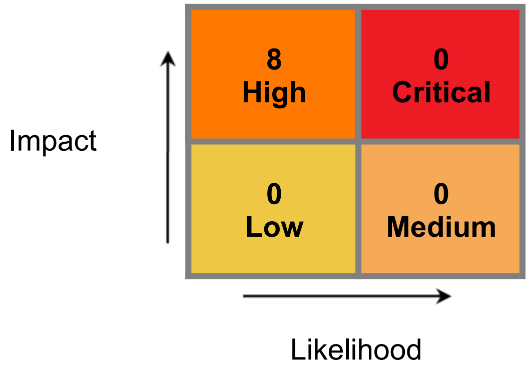
Executive Summary

This workbook is intended to provide all necessary details and information for a developer to understand and remediate the different issues discovered during the ManagementPlatformWeb project audit. The information contained in this workbook is targeted at project managers and developers.

This section provides an overview of the issues uncovered during analysis.

Project Name:	ManagementPlatformWe
Project Version:	
SCA:	Results Present
WebInspect:	Results Not Present
WebInspect Agent:	Results Not Present
Other:	Results Not Present

Issues by Priority



8 High	0 Critical
0 Low	0 Medium

Top Ten Critical Categories

This project does not contain any critical issues

Project Description

This section provides an overview of the Fortify scan engines used for this project, as well as the project meta-information.

Issue Breakdown by Fortify Categories

The following table depicts a summary of all issues grouped vertically by Fortify Category. For each category, the total number of issues is shown by Fortify Priority Order, including information about the number of audited issues.

Category	Fortify Priority (audited/total)				Total Issues
	Critical	High	Medium	Low	
Dynamic Code Evaluation: Code Injection	0	0 / 8	0	0	0 / 8

Results Outline

Dynamic Code Evaluation: Code Injection (8 issues)

Abstract

Interpreting user-controlled instructions at run-time can allow attackers to execute malicious code.

Explanation

Many modern programming languages allow dynamic interpretation of source instructions. This capability allows programmers to perform dynamic instructions based on input received from the user. Code injection vulnerabilities occur when the programmer incorrectly assumes that instructions supplied directly from the user will perform only innocent operations, such as performing simple calculations on active user objects or otherwise modifying the user's state. However, without proper validation, a user might specify operations the programmer does not intend. **Example:** In this classic code injection example, the application implements a basic calculator that allows the user to specify commands for execution.

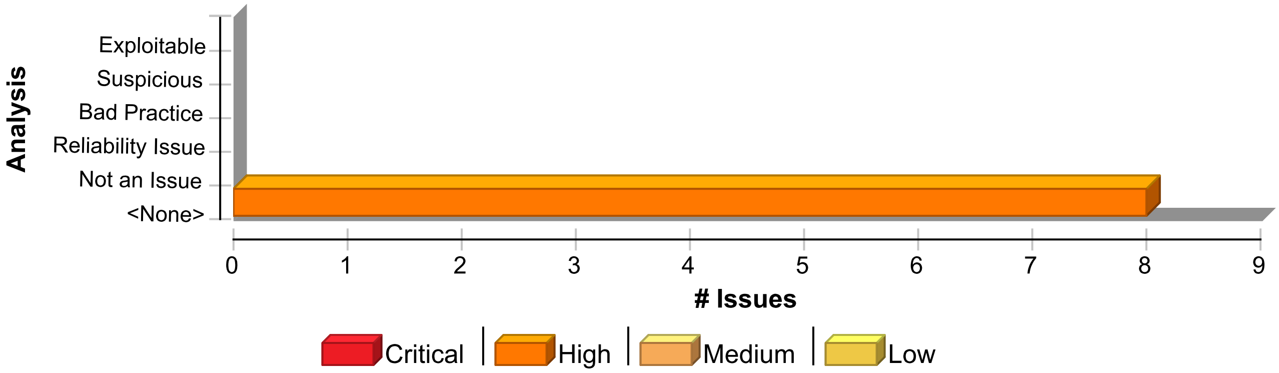
```
...
    userOp = form.operation.value;
    calcResult = eval(userOp);
...
```

The program behaves correctly when the `operation` parameter is a benign value, such as "8 + 7 * 2", in which case the `calcResult` variable is assigned a value of 22. However, if an attacker specifies languages operations that are both valid and malicious, those operations would be executed with the full privilege of the parent process. Such attacks are even more dangerous when the underlying language provides access to system resources or allows execution of system commands. In the case of JavaScript, the attacker can utilize this vulnerability to perform a cross-site scripting attack.

Recommendation

Avoid dynamic code interpretation whenever possible. If your program's functionality requires code to be interpreted dynamically, the likelihood of attack can be minimized by constraining the code your program will execute dynamically as much as possible, limiting it to an application- and context-specific subset of the base programming language. If dynamic code execution is required, unvalidated user input should never be directly executed and interpreted by the application. Instead, a level of indirection should be introduced: create a list of legitimate operations and data objects that users are allowed to specify, and only allow users to select from the list. With this approach, input provided by users is never executed directly.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Dynamic Code Evaluation: Code Injection	8	0	0	8
Total	**8**	**0**	**0**	**8**

Dynamic Code Evaluation: Code Injection — High

Package: 00Lexi.01SVN.CountInsurance.trunk.01.Develop.ManagementPlatformWeb.CountNet.Insurance.ManagementPlatformWeb.Scripts

00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.js, line 2809 (Dynamic Code Evaluation: Code Injection) — **High**

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read window.location
From: trackPageView
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.js:2786

```
2783  name = window.document && window.document.title || "";
2784  }
2785  if (typeof url !== "string") {
2786  url = window.location && window.location.href || "";
2787  }
2788  var pageViewSent = false;
2789  var customDuration = 0;
```

Sink Details

Sink: setInterval()
Enclosing Method: trackPageView()
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.js:2809
Taint Flags: VALIDATED_OPEN_REDIRECT, WEB, XSS

```
2806
ApplicationInsights._InternalLogging.throwInternalNonUserActionable(ApplicationInsights.LoggingS
new
ApplicationInsights._InternalLogMessage(ApplicationInsights._InternalMessageId.NONUSRACT_Navigat
"trackPageView: navigation timing API used for calculation of page duration is not supported
in this browser. This page view will be collected without duration and timing info."));
2807  return;
2808  }
2809  var handle = setInterval(function () {
2810  try {
2811  if (Telemetry.PageViewPerformance.isPerformanceTimingDataReady()) {
2812  clearInterval(handle);
```

Dynamic Code Evaluation: Code Injection — High

Package: 00Lexi.01SVN.CountInsurance.trunk.01.Develop.ManagementPlatformWeb.CountNet.Insurance.ManagementPlatformWeb.Scripts

00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.js, line 2809 (Dynamic Code Evaluation: Code Injection) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read window.location
From: trackPageView
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.js:2786

```
2783   name = window.document && window.document.title || "";
2784   }
2785   if (typeof url !== "string") {
2786   url = window.location && window.location.href || "";
2787   }
2788   var pageViewSent = false;
2789   var customDuration = 0;
```

Sink Details

Sink: setInterval()
Enclosing Method: trackPageView()
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.js:2809
Taint Flags: VALIDATED_OPEN_REDIRECT, WEB, XSS

```
2806
ApplicationInsights._InternalLogging.throwInternalNonUserActionable(ApplicationInsights.LoggingS
new
ApplicationInsights._InternalLogMessage(ApplicationInsights._InternalMessageId.NONUSRACT_Navigat
"trackPageView: navigation timing API used for calculation of page duration is not supported
in this browser. This page view will be collected without duration and timing info."));
2807   return;
2808   }
2809   var handle = setInterval(function () {
2810   try {
2811   if (Telemetry.PageViewPerformance.isPerformanceTimingDataReady()) {
2812   clearInterval(handle);
```

00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.min.js, line 1 (Dynamic Code Evaluation: Code Injection) — High

Issue Details

Kingdom: Input Validation and Representation

Dynamic Code Evaluation: Code Injection — High

Package: 00Lexi.01SVN.CountInsurance.trunk.01.Develop.ManagementPlatformWeb.CountNet.Insurance.ManagementPlatformWeb.Scripts

00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.min.js, line 1 (Dynamic Code Evaluation: Code Injection) — High

Scan Engine: SCA (Data Flow)

Source Details

Source: Read window.location
From: trackPageView
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.min.js:1

```
1 [Too long 86777 chars line truncated to 3500 ones]var
__extends,AI,Microsoft;(function(n){var t;(function(n){var r,t,i,u;
(function(n){n[n.CRITICAL=0]="CRITICAL";n[n.WARNING=1]="WARNING"})
(n.LoggingSeverity||(n.LoggingSeverity={}));r=n.LoggingSeverity,function(n)
{n[n.NONUSRACT_BrowserDoesNotSupportLocalStorage=0]="NONUSRACT_BrowserDoesNotSu
2
3 undefined
4 undefined
5 undefined
6 undefined
7 undefined
```

Sink Details

Sink: setInterval()
Enclosing Method: trackPageView()
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.min.js:1
Taint Flags: VALIDATED_OPEN_REDIRECT, WEB, XSS

```
1 [Too long 86777 chars line truncated to 3500 ones]var __extends,AI,Microsoft;(function(n)
{var t;(function(n){var r,t,i,u;(function(n)
{n[n.CRITICAL=0]="CRITICAL";n[n.WARNING=1]="WARNING"})(n.LoggingSeverity||
(n.LoggingSeverity={}));r=n.LoggingSeverity,function(n)
{n[n.NONUSRACT_BrowserDoesNotSupportLocalStorage=0]="NONUSRACT_BrowserDoesNotSupportLocalStorage
2
3 undefined
4 undefined
5 undefined
6 undefined
7 undefined
```

00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.min.js, line 1 (Dynamic Code Evaluation: Code Injection) — High

Issue Details

Dynamic Code Evaluation: Code Injection — High

Package: 00Lexi.01SVN.CountInsurance.trunk.01.Develop.ManagementPlatformWeb.CountNet.Insurance.ManagementPlatformWeb.Scripts

00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.min.js, line 1 (Dynamic Code Evaluation: Code Injection) — High

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read window.location
From: trackPageView
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.min.js:1

```
1 [Too long 86777 chars line truncated to 3500 ones]var __extends,AI,Microsoft;(function(n){var t;(function(n){var r,t,i,u;(function(n){n[n.CRITICAL=0]="CRITICAL";n[n.WARNING=1]="WARNING"})(n.LoggingSeverity||(n.LoggingSeverity={}));r=n.LoggingSeverity,function(n){n[n.NONUSRACT_BrowserDoesNotSupportLocalStorage=0]="NONUSRACT_BrowserDoesNotSu
2
3 undefined
4 undefined
5 undefined
6 undefined
7 undefined
```

Sink Details

Sink: setInterval()
Enclosing Method: trackPageView()
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/CountNet.Insurance.ManagementPlatformWeb/Scripts/ai.0.22.9-build00167.min.js:1
Taint Flags: VALIDATED_OPEN_REDIRECT, WEB, XSS

```
1 [Too long 86777 chars line truncated to 3500 ones]var __extends,AI,Microsoft;(function(n){var t;(function(n){var r,t,i,u;(function(n){n[n.CRITICAL=0]="CRITICAL";n[n.WARNING=1]="WARNING"})(n.LoggingSeverity||(n.LoggingSeverity={}));r=n.LoggingSeverity,function(n){n[n.NONUSRACT_BrowserDoesNotSupportLocalStorage=0]="NONUSRACT_BrowserDoesNotSupportLocalStorage
2
3 undefined
4 undefined
5 undefined
6 undefined
7 undefined
```

Dynamic Code Evaluation: Code Injection — High

Package: 00Lexi.01SVN.CountInsurance.trunk.01.Develop.ManagementPlatformWeb.packages.Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167.content.scripts

00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.js, line 2809 (Dynamic Code Evaluation: Code Injection) — High

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read window.location
From: trackPageView
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.js:2786

```
2783  name = window.document && window.document.title || "";
2784  }
2785  if (typeof url !== "string") {
2786  url = window.location && window.location.href || "";
2787  }
2788  var pageViewSent = false;
2789  var customDuration = 0;
```

Sink Details

Sink: setInterval()
Enclosing Method: trackPageView()
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.js:2809
Taint Flags: VALIDATED_OPEN_REDIRECT, WEB, XSS

```
2806
ApplicationInsights._InternalLogging.throwInternalNonUserActionable(ApplicationInsights.LoggingS
new
ApplicationInsights._InternalLogMessage(ApplicationInsights._InternalMessageId.NONUSRACT_Navigat
"trackPageView: navigation timing API used for calculation of page duration is not supported
in this browser. This page view will be collected without duration and timing info."));
2807  return;
2808  }
2809  var handle = setInterval(function () {
2810  try {
2811  if (Telemetry.PageViewPerformance.isPerformanceTimingDataReady()) {
2812  clearInterval(handle);
```

Dynamic Code Evaluation: Code Injection — High

Package: 00Lexi.01SVN.CountInsurance.trunk.01.Develop.ManagementPlatformWeb.packages.Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167.content.scripts

00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.js, line 2809 (Dynamic Code Evaluation: Code Injection)	**High**

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read window.location
From: trackPageView
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.js:2786

```
2783  name = window.document && window.document.title || "";
2784  }
2785  if (typeof url !== "string") {
2786  url = window.location && window.location.href || "";
2787  }
2788  var pageViewSent = false;
2789  var customDuration = 0;
```

Sink Details

Sink: setInterval()
Enclosing Method: trackPageView()
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.js:2809
Taint Flags: VALIDATED_OPEN_REDIRECT, WEB, XSS

```
2806
ApplicationInsights._InternalLogging.throwInternalNonUserActionable(ApplicationInsights.LoggingS
new
ApplicationInsights._InternalLogMessage(ApplicationInsights._InternalMessageId.NONUSRACT_Navigat
"trackPageView: navigation timing API used for calculation of page duration is not supported
in this browser. This page view will be collected without duration and timing info."));
2807  return;
2808  }
2809  var handle = setInterval(function () {
2810  try {
2811  if (Telemetry.PageViewPerformance.isPerformanceTimingDataReady()) {
2812  clearInterval(handle);
```

Dynamic Code Evaluation: Code Injection	High
Package: 00Lexi.01SVN.CountInsurance.trunk.01.Develop.ManagementPlatformWeb.packages.Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167.content.scripts	
00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.min.js, line 1 (Dynamic Code Evaluation: Code Injection)	**High**

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read window.location
From: trackPageView
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.min.js:1

```
1 [Too long 86777 chars line truncated to 3500 ones]var
__extends,AI,Microsoft;(function(n){var t;(function(n){var r,t,i,u;
(function(n){n[n.CRITICAL=0]="CRITICAL";n[n.WARNING=1]="WARNING"})
(n.LoggingSeverity||(n.LoggingSeverity={}));r=n.LoggingSeverity,function(n)
{n[n.NONUSRACT_BrowserDoesNotSupportLocalStorage=0]="NONUSRACT_BrowserDoesNotSu
2
3 undefined
4 undefined
5 undefined
6 undefined
7 undefined
```

Sink Details

Sink: setInterval()
Enclosing Method: trackPageView()
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.min.js:1
Taint Flags: VALIDATED_OPEN_REDIRECT, WEB, XSS

```
1 [Too long 86777 chars line truncated to 3500 ones]var __extends,AI,Microsoft;(function(n)
{var t;(function(n){var r,t,i,u;(function(n)
{n[n.CRITICAL=0]="CRITICAL";n[n.WARNING=1]="WARNING"})(n.LoggingSeverity||
(n.LoggingSeverity={}));r=n.LoggingSeverity,function(n)
{n[n.NONUSRACT_BrowserDoesNotSupportLocalStorage=0]="NONUSRACT_BrowserDoesNotSupportLocalStorage
2
3 undefined
4 undefined
5 undefined
6 undefined
7 undefined
```

Dynamic Code Evaluation: Code Injection	High
Package: 00Lexi.01SVN.CountInsurance.trunk.01.Develop.ManagementPlatformWeb.packages.Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167.content.scripts	
00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.min.js, line 1 (Dynamic Code Evaluation: Code Injection)	**High**

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read window.location
From: trackPageView
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.min.js:1

```
1 [Too long 86777 chars line truncated to 3500 ones]var
__extends,AI,Microsoft;(function(n){var t;(function(n){var r,t,i,u;
(function(n){n[n.CRITICAL=0]="CRITICAL";n[n.WARNING=1]="WARNING"})
(n.LoggingSeverity||(n.LoggingSeverity={}));r=n.LoggingSeverity,function(n)
{n[n.NONUSRACT_BrowserDoesNotSupportLocalStorage=0]="NONUSRACT_BrowserDoesNotSu
2
3 undefined
4 undefined
5 undefined
6 undefined
7 undefined
```

Sink Details

Sink: setInterval()
Enclosing Method: trackPageView()
File: 00Lexi/01SVN/CountInsurance/trunk/01.Develop/ManagementPlatformWeb/packages/Microsoft.ApplicationInsights.JavaScript.0.22.9-build00167/content/scripts/ai.0.22.9-build00167.min.js:1
Taint Flags: VALIDATED_OPEN_REDIRECT, WEB, XSS

```
1 [Too long 86777 chars line truncated to 3500 ones]var __extends,AI,Microsoft;(function(n)
{var t;(function(n){var r,t,i,u;(function(n)
{n[n.CRITICAL=0]="CRITICAL";n[n.WARNING=1]="WARNING"})(n.LoggingSeverity||
(n.LoggingSeverity={}));r=n.LoggingSeverity,function(n)
{n[n.NONUSRACT_BrowserDoesNotSupportLocalStorage=0]="NONUSRACT_BrowserDoesNotSupportLocalStorage
2
3 undefined
4 undefined
5 undefined
6 undefined
7 undefined
```

Description of Key Terminology

Likelihood and Impact

Likelihood
Likelihood is the probability that a vulnerability will be accurately identified and successfully exploited.

Impact
Impact is the potential damage an attacker could do to assets by successfully exploiting a vulnerability. This damage can be in the form of, but not limited to, financial loss, compliance violation, loss of brand reputation, and negative publicity.

Fortify Priority Order

Critical
Critical-priority issues have high impact and high likelihood. Critical-priority issues are easy to detect and exploit and result in large asset damage. These issues represent the highest security risk to the application. As such, they should be remediated immediately.

SQL Injection is an example of a critical issue.

High
High-priority issues have high impact and low likelihood. High-priority issues are often difficult to detect and exploit, but can result in large asset damage. These issues represent a high security risk to the application. High-priority issues should be remediated in the next scheduled patch release.

Password Management: Hardcoded Password is an example of a high issue.

Medium
Medium-priority issues have low impact and high likelihood. Medium-priority issues are easy to detect and exploit, but typically result in small asset damage. These issues represent a moderate security risk to the application. Medium-priority issues should be remediated in the next scheduled product update.

Path Manipulation is an example of a medium issue.

Low
Low-priority issues have low impact and low likelihood. Low-priority issues can be difficult to detect and exploit and typically result in small asset damage. These issues represent a minor security risk to the application. Low-priority issues should be remediated as time allows.

Dead Code is an example of a low issue.

About Fortify Solutions

Fortify is the leader in end-to-end application security solutions with the flexibility of testing on-premise and on-demand to cover the entire software development lifecycle. Learn more at software.microfocus.com/en-us/solutions/application-security.