Fortify Audit Workbench

OWASP Top 10 2013

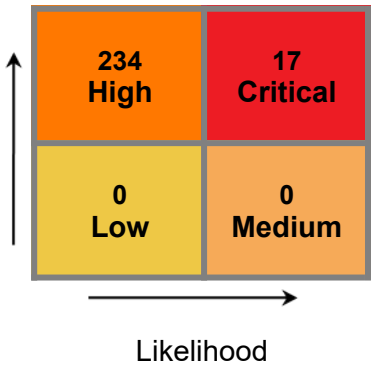
Signal Performance Metrics

Table of Contents

Executive Summary

Project Name:	Signal Performance Meti
Project Version:	
SCA:	Results Present
WebInspect:	Results Not Present
WebInspect Agent:	Results Not Present
Other:	Results Not Present

Issues by Priority



* The detailed sections following the Executive Summary contain specifics.

Project Description

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

SCA

Date of Last Analysis:	Aug 1, 2018, 12:57 PM	**Engine Version:**	18.10.0187
Host Name:	SEDTS-L0G2-17I0	**Certification:**	VALID
Number of Files:	1,674	**Lines of Code:**	72,709

Issue BreakDown

The following table summarizes the number of issues identified across the different OWASP Top 10 2013 categories and broken down by Fortify Priority Order.

	Fortify Priority				Total Issues
	Critical	High	Medium	Low	
A1 Injection	2	91	0	0	93
A2 Broken Authentication and Session Management	0	0	0	0	0
A3 Cross-Site Scripting (XSS)	0	0	0	0	0
A4 Insecure Direct Object References	10	56	0	0	66
A5 Security Misconfiguration	0	0	0	0	0
A6 Sensitive Data Exposure	1	52	0	0	53
A7 Missing Function Level Access Control	0	0	0	0	0
A8 Cross-Site Request Forgery (CSRF)	0	35	0	0	35
A9 Using Components with Known Vulnerabilities	0	0	0	0	0
A10 Unvalidated Redirects and Forwards	4	0	0	0	4

NOTE:
1. Reported issues in the above table may violate more than one OWASP Top 10 2013 category. As such, the same issue may appear in more than one row. The total number of unique vulnerabilities are reported in the Executive Summary table.

Issue Details

Below is an enumeration of all issues found in the project. The issues are organized by OWASP Top 10 2013, Fortify Priority Order, and vulnerability category. The issues are then further broken down by the package, namespace, or location in which they occur. Issues reported at the same line number with the same category originate from different taint sources.

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

JSON Injection		Critical
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Controllers/DataExportController.cs:111 Analysis: Not an Issue	**Sink:** Newtonsoft.Json.Linq.JObject.Parse() **Enclosing Method:** GetReCaptchaStatus() **Source:** System.Net.WebClient.DownloadString() **from** SPM.Controllers.DataExportController.GetReCaptchaStatus() **In** Projects/GitHub/ATSPM/SPM/Controllers/DataExportController.cs:109	SCA

SQL Injection		Critical
Package: ConvertDBForHistoricalConfigurations		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/ConvertDBForHistoricalConfigurations/Program.cs:148 Analysis: Not an Issue	**Sink:** System.Data.SqlClient.SqlCommand.set_CommandText() **Enclosing Method:** UpdateMigrationsTable() **Source:** System.Data.SqlClient.SqlCommand.ExecuteReader() **from** ConvertDBForHistoricalConfigurations.Program.UpdateMigrationsTable() **In** Projects/GitHub/ATSPM/ConvertDBForHistoricalConfigurations/Program.cs:124	SCA

ASP.NET MVC Bad Practices: Model With Optional and Required Properties		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Controllers/AggregateDataExportController.cs:90 Analysis: Bad Practice	**Sink:** Function: CreateMetric **Enclosing Method:** CreateMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DataExportController.cs:41 Analysis: Bad Practice	**Sink:** Function: RawDataExport **Enclosing Method:** RawDataExport() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DataExportController.cs:154 Analysis: Bad Practice	**Sink:** Function: GetRecordCount **Enclosing Method:** GetRecordCount() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DatabaseArchiveSettingsContro 46	**Sink:** Function: Edit **Enclosing Method:** Edit() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Model With Optional and Required Properties		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:36 Analysis: Bad Practice	**Sink:** Function: SignalSearch **Enclosing Method:** SignalSearch() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:90 Analysis: Bad Practice	**Sink:** Function: GetMap **Enclosing Method:** GetMap() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:174 Analysis: Bad Practice	**Sink:** Function: GetPhaseTerminationMetricByUrl **Enclosing Method:** GetPhaseTerminationMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:213 Analysis: Bad Practice	**Sink:** Function: GetPreemptMetricByUrl **Enclosing Method:** GetPreemptMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:224 Analysis: Bad Practice	**Sink:** Function: GetTMCMetricByUrl **Enclosing Method:** GetTMCMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:238 Analysis: Bad Practice	**Sink:** Function: GetPCDMetricByUrl **Enclosing Method:** GetPCDMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:255 Analysis: Bad Practice	**Sink:** Function: GetApproachVolumeMetricByUrl **Enclosing Method:** GetApproachVolumeMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:275 Analysis: Bad Practice	**Sink:** Function: GetApproachDelayMetricByUrl **Enclosing Method:** GetApproachDelayMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:288 Analysis: Bad Practice	**Sink:** Function: GetAoRMetricByUrl **Enclosing Method:** GetAoRMetricByUrl() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Model With Optional and Required Properties		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:299 Analysis: Bad Practice	**Sink:** Function: GetApproachSpeedMetricByUrl **Enclosing Method:** GetApproachSpeedMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:332 Analysis: Bad Practice	**Sink:** Function: GetSplitFailMetricByUrl **Enclosing Method:** GetSplitFailMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:448 Analysis: Bad Practice	**Sink:** Function: GetSplitFailMetric **Enclosing Method:** GetSplitFailMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:512 Analysis: Bad Practice	**Sink:** Function: GetApproachSpeedMetric **Enclosing Method:** GetApproachSpeedMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:576 Analysis: Bad Practice	**Sink:** Function: GetAoRMetric **Enclosing Method:** GetAoRMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:636 Analysis: Bad Practice	**Sink:** Function: GetApproachDelayMetric **Enclosing Method:** GetApproachDelayMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:699 Analysis: Bad Practice	**Sink:** Function: GetPhaseTerminationMetric **Enclosing Method:** GetPhaseTerminationMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:754 Analysis: Bad Practice	**Sink:** Function: GetPreemptMetric **Enclosing Method:** GetPreemptMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:826 Analysis: Bad Practice	**Sink:** Function: GetTMCMetric **Enclosing Method:** GetTMCMetric() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Model With Optional and Required Properties		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:894 Analysis: Bad Practice	**Sink:** Function: GetApproachVolumeMetric **Enclosing Method:** GetApproachVolumeMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:1062 Analysis: Bad Practice	**Sink:** Function: GetPCDMetric **Enclosing Method:** GetPCDMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/LinkPivotController.cs:51 Analysis: Bad Practice	**Sink:** Function: LinkPivotResult **Enclosing Method:** LinkPivotResult() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/LinkPivotController.cs:162 Analysis: Bad Practice	**Sink:** Function: LinkPivotPCDOptions **Enclosing Method:** LinkPivotPCDOptions() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/LinkPivotController.cs:185 Analysis: Bad Practice	**Sink:** Function: PCDs **Enclosing Method:** PCDs() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/RouteSignalsController.cs:59 Analysis: Bad Practice	**Sink:** Function: RouteMap **Enclosing Method:** RouteMap() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/SignalsController.cs:334 Analysis: Bad Practice	**Sink:** Function: CopyVersion **Enclosing Method:** CopyVersion() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/SignalsController.cs:552 Analysis: Bad Practice	**Sink:** Function: Edit **Enclosing Method:** Edit() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Model With Required Non-Nullable Property	High
Package: MOE.Common.Models	

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Models/ApplicationEvent.cs:20 Analysis: Not an Issue	**Sink:** Function: set_Timestamp **Enclosing Method:** set_Timestamp() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ApplicationEvent.cs:29 Analysis: Not an Issue	**Sink:** Function: set_SeverityLevel **Enclosing Method:** set_SeverityLevel() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ApproachCycleAggregation.cs:15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ApproachPcdAggregation.cs:15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ApproachSpeedAggregation.cs:15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ApproachSplitFailAggregation.c15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ApproachYellowRedActivations15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/Comment.cs:18 Analysis: Not an Issue	**Sink:** Function: set_TimeStamp **Enclosing Method:** set_TimeStamp() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Model With Required Non-Nullable Property	High
Package: MOE.Common.Models	

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Models/Detector.cs:60 Analysis: Not an Issue	**Sink:** Function: set_DateAdded **Enclosing Method:** set_DateAdded() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/DetectorAggregation.cs:15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/DetectorEventCountAggregation.cs:15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/PhaseEventCountAggregation.cs:15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/PhasePedAggregation.cs:15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/PhaseTerminationAggregation.cs:15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/PreemptionAggregation.cs:15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/PriorityAggregation.cs:15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Model With Required Non-Nullable Property	High

Package: MOE.Common.Models

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Models/SPMWatchDogErrorEvent.cs:12 Analysis: Not an Issue	**Sink:** Function: set_TimeStamp **Enclosing Method:** set_TimeStamp() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/Signal.cs:70 Analysis: Not an Issue	**Sink:** Function: set_Start **Enclosing Method:** set_Start() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/SignalEventCountAggregation.cs:15 Analysis: Not an Issue	**Sink:** Function: set_BinStartTime **Enclosing Method:** set_BinStartTime() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/StatusOfProcessedTable.cs:15 Analysis: Not an Issue	**Sink:** Function: set_TimeEntered **Enclosing Method:** set_TimeEntered() **Source:**	SCA

Package: MOE.Common.Models.ViewModel.Chart

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Models/ViewModel/Chart/DefaultChartsViewModel.cs:24 Analysis: Bad Practice	**Sink:** Function: set_StartDateDay **Enclosing Method:** set_StartDateDay() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ViewModel/Chart/DefaultChartsViewModel.cs:38 Analysis: Bad Practice	**Sink:** Function: set_EndDateDay **Enclosing Method:** set_EndDateDay() **Source:**	SCA

ASP.NET MVC Bad Practices: Optional Submodel With Required Property	High

Package: MOE.Common.Business

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Business/ArriveOnRedChart.cs:119 Analysis: Not an Issue	**Sink:** Function: set_Options **Enclosing Method:** set_Options() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Optional Submodel With Required Property	High
Package: MOE.Common.Business	

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Business/DelayChart.cs:130 Analysis: Not an Issue	**Sink:** Function: set_Options **Enclosing Method:** set_Options() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/Detector.cs:95 Analysis: Not an Issue	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/Detector.cs:105 Analysis: Not an Issue	**Sink:** Function: set_LaneType **Enclosing Method:** set_LaneType() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/Detector.cs:107 Analysis: Not an Issue	**Sink:** Function: set_DetectorModel **Enclosing Method:** set_DetectorModel() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/DetectorCollection.cs:53 Analysis: Not an Issue	**Sink:** Function: set_DetectionHardware **Enclosing Method:** set_DetectionHardware() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/Phase.cs:37 Analysis: Not an Issue	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/RLM/RLMPlan.cs:167 Analysis: Suspicious	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/RLM/RLMPlanCollection.cs:37 Analysis: Suspicious	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/RLM/RLMSignalPhase.cs:103 Analysis: Suspicious	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Optional Submodel With Required Property	High

Package: MOE.Common.Business.ApproachVolume

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Business/ApproachVolume/Approach.cs:26 Analysis: Not an Issue	**Sink:** Function: set_ApproachModel **Enclosing Method:** set_ApproachModel() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/ApproachVolume/ApproachVolume.cs:24 Analysis: Suspicious	**Sink:** Function: set_DetectionType **Enclosing Method:** set_DetectionType() **Source:**	SCA

Package: MOE.Common.Business.CustomReport

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Business/CustomReport/Phase.cs:51 Analysis: Not an Issue	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/CustomReport/Signal.cs:39 Analysis: Suspicious	**Sink:** Function: set_SignalModel **Enclosing Method:** set_SignalModel() **Source:**	SCA

Package: MOE.Common.Business.TMC

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Business/TMC/TMCMetric.cs:36 Analysis: Not an Issue	**Sink:** Function: set_Options **Enclosing Method:** set_Options() **Source:**	SCA

Package: MOE.Common.Business.WCFServiceLibrary

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Business/WCFServiceLibrary/MetricOptions.cs:102 Analysis: Not an Issue	**Sink:** Function: set_MetricType **Enclosing Method:** set_MetricType() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Optional Submodel With Required Property		High
Package: MOE.Common.Business.WCFServiceLibrary		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.Common/Business/WCFServiceLibrary/PCDOptions.cs:79 Analysis: Not an Issue	**Sink:** Function: set_Signal **Enclosing Method:** set_Signal() **Source:**	SCA
Package: MOE.Common.Models		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.Common/Models/Approach.cs:30 Analysis: Not an Issue	**Sink:** Function: set_Signal **Enclosing Method:** set_Signal() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ApproachCycleAggregation.cs:20 Analysis: Not an Issue	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ApproachPcdAggregation.cs:20 Analysis: Not an Issue	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ApproachSpeedAggregation.cs:20 Analysis: Not an Issue	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ApproachSplitFailAggregation.c20 Analysis: Not an Issue	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/ApproachYellowRedActivations.20 Analysis: Not an Issue	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Optional Submodel With Required Property	High
Package: MOE.Common.Models	

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Models/Detector.cs:82 Analysis: Not an Issue	**Sink:** Function: set_MovementType **Enclosing Method:** set_MovementType() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/Detector.cs:89 Analysis: Not an Issue	**Sink:** Function: set_LaneType **Enclosing Method:** set_LaneType() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/Detector.cs:113 Analysis: Not an Issue	**Sink:** Function: set_Approach **Enclosing Method:** set_Approach() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/Detector.cs:121 Analysis: Not an Issue	**Sink:** Function: set_DetectionHardware **Enclosing Method:** set_DetectionHardware() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/DetectorAggregation.cs:21 Analysis: Not an Issue	**Sink:** Function: set_Detector **Enclosing Method:** set_Detector() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/DetectorComment.cs:12 Analysis: Not an Issue	**Sink:** Function: set_Detector **Enclosing Method:** set_Detector() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/MetricComment.cs:15 Analysis: Not an Issue	**Sink:** Function: set_Signal **Enclosing Method:** set_Signal() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/PreemptionAggregation.cs:24 Analysis: Not an Issue	**Sink:** Function: set_Signal **Enclosing Method:** set_Signal() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/PriorityAggregation.cs:24 Analysis: Not an Issue	**Sink:** Function: set_Signal **Enclosing Method:** set_Signal() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Optional Submodel With Required Property	High

Package: MOE.Common.Models

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Models/RoutePhaseDirection.cs:15 Analysis: Not an Issue	**Sink:** Function: set_RouteSignal **Enclosing Method:** set_RouteSignal() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/RouteSignal.cs:17 Analysis: Not an Issue	**Sink:** Function: set_Route **Enclosing Method:** set_Route() **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/Models/RouteSignal.cs:28 Analysis: Not an Issue	**Sink:** Function: set_Signal **Enclosing Method:** set_Signal() **Source:**	SCA

Package: MOE.Common.Models.ViewModel.Chart

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Models/ViewModel/Chart/DefaultChartsViewModel.cs:51 Analysis: Bad Practice	**Sink:** Function: set_SignalSearch **Enclosing Method:** set_SignalSearch() **Source:**	SCA

Package: MOE.Common.Models.ViewModel.RouteEdit

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Models/ViewModel/RouteEdit/RouteCreateViewModel.cs:14 Analysis: Suspicious	**Sink:** Function: set_Route **Enclosing Method:** set_Route() **Source:**	SCA

Package: MOE.Common.Models.ViewModel.WebConfigTool

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/MOE.Common/Models/ViewModel/WebConfigTool/WebConfigToolViewModel.cs:18 Analysis: Not an Issue	**Sink:** Function: set_SignalSearch **Enclosing Method:** set_SignalSearch() **Source:**	SCA

A1 Injection

Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

ASP.NET MVC Bad Practices: Optional Submodel With Required Property		High
Package: MOE.Common.Models.ViewModel._MainMenu		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.Common/Models/ViewModel/_MainMenu/MenuItem.cs:33 Analysis: Not an Issue	**Sink:** Function: set_MenuObject **Enclosing Method:** set_MenuObject() **Source:**	SCA

A2 Broken Authentication and Session Management

Account credentials and session tokens are often not properly protected. Attackers compromise passwords, keys, or authentication tokens to assume other users' identities.

No Issues

A3 Cross-Site Scripting (XSS)

XSS flaws occur whenever an application takes user supplied data and sends it to a web browser without first validating or encoding that content. XSS allows attackers to execute script in the victim's browser which can hijack user sessions, deface web sites, possibly introduce worms, etc.

No Issues

A4 Insecure Direct Object References

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.

Path Manipulation		Critical
Package: AlexPilotti.FTPS.Client		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:737 Analysis: Bad Practice	**Sink:** System.IO.FileStream.FileStream() **Enclosing Method:** GetFile() **Source:** System.IO.Stream.Read() **from** AlexPilotti.FTPS.Client.FTPSClient.GetDataString() **In** Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:1547	SCA
Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:737 Analysis: Bad Practice	**Sink:** System.IO.FileStream.FileStream() **Enclosing Method:** GetFile() **Source:** System.Net.Sockets.TcpClient.GetStream() **from** AlexPilotti.FTPS.Client.FTPSClient.GetDataStream() **In** Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:1530	SCA
Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:737 Analysis: Bad Practice	**Sink:** System.IO.FileStream.FileStream() **Enclosing Method:** GetFile() **Source:** System.Net.Sockets.TcpClient.GetStream() **from** AlexPilotti.FTPS.Client.FTPSClient.GetDataStream() **In** Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:1526	SCA
Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:797 Analysis: Suspicious	**Sink:** System.IO.Directory.CreateDirectory() **Enclosing Method:** GetFiles() **Source:** System.IO.Stream.Read() **from** AlexPilotti.FTPS.Client.FTPSClient.GetDataString() **In** Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:1547	SCA
Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:797 Analysis: Suspicious	**Sink:** System.IO.Directory.CreateDirectory() **Enclosing Method:** GetFiles() **Source:** System.Net.Sockets.TcpClient.GetStream() **from** AlexPilotti.FTPS.Client.FTPSClient.GetDataStream() **In** Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:1526	SCA
Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:797 Analysis: Suspicious	**Sink:** System.IO.Directory.CreateDirectory() **Enclosing Method:** GetFiles() **Source:** System.Net.Sockets.TcpClient.GetStream() **from** AlexPilotti.FTPS.Client.FTPSClient.GetDataStream() **In** Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:1530	SCA

A4 Insecure Direct Object References

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.

Path Manipulation		Critical
Package: MOE.Common.Business.WCFServiceLibrary		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.Common/Business/WCFServiceLibrary/MetricOptions.cs:193 Analysis: Not an Issue	**Sink:** System.IO.DirectoryInfo.DirectoryInfo() **Enclosing Method:** DriveAvailable() **Source:** CreateTMCChart(0) **from** MOEWcfServiceLibrary.MetricGenerator.CreateTMCChart() **In** Projects/GitHub/ATSPM/MOEWcfServiceLibrary/MetricGenerator.cs:97	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/WCFServiceLibrary/MetricOptions.cs:193 Analysis: Not an Issue	**Sink:** System.IO.DirectoryInfo.DirectoryInfo() **Enclosing Method:** DriveAvailable() **Source:** CreateMetricWithDataTable(0) **from** MOEWcfServiceLibrary.MetricGenerator.CreateMetricWithDataTable() **In** Projects/GitHub/ATSPM/MOEWcfServiceLibrary/MetricGenerator.cs:69	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/WCFServiceLibrary/MetricOptions.cs:197 Analysis: Not an Issue	**Sink:** System.IO.Directory.CreateDirectory() **Enclosing Method:** DriveAvailable() **Source:** CreateTMCChart(0) **from** MOEWcfServiceLibrary.MetricGenerator.CreateTMCChart() **In** Projects/GitHub/ATSPM/MOEWcfServiceLibrary/MetricGenerator.cs:97	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/WCFServiceLibrary/MetricOptions.cs:197 Analysis: Not an Issue	**Sink:** System.IO.Directory.CreateDirectory() **Enclosing Method:** DriveAvailable() **Source:** CreateMetricWithDataTable(0) **from** MOEWcfServiceLibrary.MetricGenerator.CreateMetricWithDataTable() **In** Projects/GitHub/ATSPM/MOEWcfServiceLibrary/MetricGenerator.cs:69	SCA

Mass Assignment: Insecure Binder Configuration		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs:292 Analysis: Bad Practice	**Sink:** Variable: model **Enclosing Method:** Login() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs:352 Analysis: Bad Practice	**Sink:** Variable: model **Enclosing Method:** VerifyCode() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs:390 Analysis: Bad Practice	**Sink:** Variable: model **Enclosing Method:** Register() **Source:**	SCA

A4 Insecure Direct Object References

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.

Mass Assignment: Insecure Binder Configuration		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs: 441 Analysis: Bad Practice	**Sink:** Variable: model **Enclosing Method:** ForgotPassword() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs: 485 Analysis: Bad Practice	**Sink:** Variable: model **Enclosing Method:** ResetPassword() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs: 545 Analysis: Bad Practice	**Sink:** Variable: model **Enclosing Method:** SendCode() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs: 598 Analysis: Bad Practice	**Sink:** Variable: model **Enclosing Method:** ExternalLoginConfirmation() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/AggregateDataExportController.90 Analysis: Bad Practice	**Sink:** Variable: aggDataExportViewModel **Enclosing Method:** CreateMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DataExportController.cs:41 Analysis: Bad Practice	**Sink:** Variable: dataExportViewModel **Enclosing Method:** RawDataExport() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DataExportController.cs:154 Analysis: Bad Practice	**Sink:** Variable: dataExportViewModel **Enclosing Method:** GetRecordCount() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DatabaseArchiveSettingsContro46 Analysis: Bad Practice	**Sink:** Variable: archiveSettingsViewModel **Enclosing Method:** Edit() **Source:**	SCA

A4 Insecure Direct Object References

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.

Mass Assignment: Insecure Binder Configuration		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:36 Analysis: Bad Practice	**Sink:** Variable: ssvm **Enclosing Method:** SignalSearch() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:90 Analysis: Bad Practice	**Sink:** Variable: dcv **Enclosing Method:** GetMap() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:174 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetPhaseTerminationMetricByUrl( ) **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:186 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetSplitMonitorMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:203 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetPedDelayMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:213 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetPreemptMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:224 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetTMCMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:238 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetPCDMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:255 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetApproachVolumeMetricByUrl() **Source:**	SCA

A4 Insecure Direct Object References

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.

Mass Assignment: Insecure Binder Configuration		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:275 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetApproachDelayMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:288 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetAoRMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:299 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetApproachSpeedMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:314 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetYRAMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:332 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetSplitFailMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:369 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetYellowAndRedMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:448 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetSplitFailMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:512 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetApproachSpeedMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:576 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetAoRMetric() **Source:**	SCA

A4 Insecure Direct Object References

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.

Mass Assignment: Insecure Binder Configuration		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:636 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetApproachDelayMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:699 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetPhaseTerminationMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:754 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetPreemptMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:826 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetTMCMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:894 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetApproachVolumeMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:953 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetSplitMonitorMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:1017 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetPedDelayMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:1062 Analysis: Bad Practice	**Sink:** Variable: metricOptions **Enclosing Method:** GetPCDMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ GeneralSettingsController.cs:36 Analysis: Bad Practice	**Sink:** Variable: generalSettings **Enclosing Method:** Edit() **Source:**	SCA

A4 Insecure Direct Object References

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.

Mass Assignment: Insecure Binder Configuration		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Controllers/LinkPivotController.cs:51 Analysis: Bad Practice	**Sink:** Variable: lpvm **Enclosing Method:** LinkPivotResult() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/LinkPivotController.cs:162 Analysis: Bad Practice	**Sink:** Variable: lpvm **Enclosing Method:** LinkPivotPCDOptions() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/LinkPivotController.cs:185 Analysis: Bad Practice	**Sink:** Variable: pcdOptions **Enclosing Method:** PCDs() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/RouteSignalsController.cs:59 Analysis: Bad Practice	**Sink:** Variable: ssvm **Enclosing Method:** RouteMap() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/RouteSignalsController.cs:130 Analysis: Bad Practice	**Sink:** Variable: routePhaseDirection **Enclosing Method:** UpdateApproach() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/SPMUsersController.cs:97 Analysis: Bad Practice	**Sink:** Variable: sPMUser **Enclosing Method:** Edit() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/SignalsController.cs:334 Analysis: Bad Practice	**Sink:** Variable: signal **Enclosing Method:** CopyVersion() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/SignalsController.cs:552 Analysis: Bad Practice	**Sink:** Variable: signal **Enclosing Method:** Edit() **Source:**	SCA

A4 Insecure Direct Object References

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.

Mass Assignment: Insecure Binder Configuration		High
Package: TestSecurityApplication.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Controllers/ManageController.cs:129 Analysis: Bad Practice	**Sink:** Variable: model **Enclosing Method:** AddPhoneNumber() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/ManageController.cs:192 Analysis: Bad Practice	**Sink:** Variable: model **Enclosing Method:** VerifyPhoneNumber() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/ManageController.cs:241 Analysis: Bad Practice	**Sink:** Variable: model **Enclosing Method:** ChangePassword() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/ManageController.cs:272 Analysis: Bad Practice	**Sink:** Variable: model **Enclosing Method:** SetPassword() **Source:**	SCA

Path Manipulation		High
Package: AlexPilotti.FTPS.Client		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:737 Analysis: Not an Issue	**Sink:** System.IO.FileStream.FileStream() **Enclosing Method:** GetFile() **Source:** System.IO.Stream.Read() **from** AlexPilotti.FTPS.Common.FTPStream.Read() **In** Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPStream.cs:85	SCA
Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:797 Analysis: Suspicious	**Sink:** System.IO.Directory.CreateDirectory() **Enclosing Method:** GetFiles() **Source:** System.IO.Stream.Read() **from** AlexPilotti.FTPS.Common.FTPStream.Read() **In** Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPStream.cs:85	SCA
Package: MOE.Common.Business.WCFServiceLibrary		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.Common/Business/WCFServiceLibrary/MetricOptions.cs:193 Analysis: Not an Issue	**Sink:** System.IO.DirectoryInfo.DirectoryInfo() **Enclosing Method:** DriveAvailable() **Source:** CreateMetric(0) **from** MOEWcfServiceLibrary.MetricGenerator.CreateMetric() **In** Projects/GitHub/ATSPM/MOEWcfServiceLibrary/MetricGenerator.cs:14	SCA

A4 Insecure Direct Object References

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.

Path Manipulation		High
Package: MOE.Common.Business.WCFServiceLibrary		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.Common/Business/WCFServiceLibrary/MetricOptions.cs:193 Analysis: Not an Issue	**Sink:** System.IO.DirectoryInfo.DirectoryInfo() **Enclosing Method:** DriveAvailable() **Source:** GetChartAndXmlFileLocations(0) **from** MOEWcfServiceLibrary.MetricGenerator.GetChartAndXmlFileLocations() **In** Projects/GitHub/ATSPM/MOEWcfServiceLibrary/MetricGenerator.cs:41	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/WCFServiceLibrary/MetricOptions.cs:197 Analysis: Not an Issue	**Sink:** System.IO.Directory.CreateDirectory() **Enclosing Method:** DriveAvailable() **Source:** CreateMetric(0) **from** MOEWcfServiceLibrary.MetricGenerator.CreateMetric() **In** Projects/GitHub/ATSPM/MOEWcfServiceLibrary/MetricGenerator.cs:14	SCA
Projects/GitHub/ATSPM/MOE.Common/Business/WCFServiceLibrary/MetricOptions.cs:197 Analysis: Not an Issue	**Sink:** System.IO.Directory.CreateDirectory() **Enclosing Method:** DriveAvailable() **Source:** GetChartAndXmlFileLocations(0) **from** MOEWcfServiceLibrary.MetricGenerator.GetChartAndXmlFileLocations() **In** Projects/GitHub/ATSPM/MOEWcfServiceLibrary/MetricGenerator.cs:41	SCA

A5 Security Misconfiguration

Having a strong server configuration standard is critical to a secure web application. Servers have many configuration options that affect security and many are not secure out of the box.

No Issues

A6 Sensitive Data Exposure

Many web applications do not properly protect sensitive data, such as credit cards, tax IDs, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes.

Privacy Violation		Critical
Package: AlexPilotti.FTPS.Client		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:1784 Analysis: Bad Practice	**Sink:** System.IO.TextWriter.WriteLine() **Enclosing Method:** HandleCmd() **Source:** Read password **from** AlexPilotti.FTPS.Client.FTPSClient.PassCmd() **In** Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:2021	SCA

Password Management: Hardcoded Password		High
Package: AlexPilotti.FTPS.Client		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/FTPSClient/FTPSClient/FTPSClient.cs:503 Analysis: Not an Issue	**Sink:** NetworkCredential() **Enclosing Method:** Connect() **Source:**	SCA
Package: GenerateControllerEventLogObjectText		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/GenerateControllerEventLogObjectText Program.cs:21 Analysis: Not an Issue	**Sink:** set_Item() **Enclosing Method:** Main() **Source:**	SCA
Package: MOE.CommonTests.Helpers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.CommonTests/Helpers/XMLtoListImporter.cs:58 Analysis: Not an Issue	**Sink:** set_Item() **Enclosing Method:** LoadControllerEventLogsFromMOEDB() **Source:**	SCA

Password Management: Password in Configuration File		High
Package: Projects.GitHub.ATSPM.ConvertDBForHistoricalConfigurations		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/ConvertDBForHistoricalConfigurations/App.config:8 Analysis: Not an Issue	**Enclosing Method:** () **Source:**	SCA

A6 Sensitive Data Exposure

Many web applications do not properly protect sensitive data, such as credit cards, tax IDs, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes.

Password Management: Password in Configuration File		High
Package: Projects.GitHub.ATSPM.ConvertDBForHistoricalConfigurations.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/ConvertDBForHistoricalConfigurations/bin/Debug/ConvertDBForHistoricalConfigu 8 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.ConvertDBForHistoricalConfigurationsTests		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/ConvertDBForHistoricalConfigurations app.config:9 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.ConvertDBForHistoricalConfigurationsTests.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/ConvertDBForHistoricalConfigurations bin/Debug/ConvertDBForHistoricalConfigu 8 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Projects/GitHub/ATSPM/ConvertDBForHistoricalConfigurations bin/Debug/ConvertDBForHistoricalConfigu 9 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.DecodePeekLogs		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/DecodePeekLogs/App.config:9 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.DecodePeekLogs.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/DecodePeekLogs/bin/Debug/DecodePeekLogs.exe.config:9 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA

A6 Sensitive Data Exposure

Many web applications do not properly protect sensitive data, such as credit cards, tax IDs, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes.

Password Management: Password in Configuration File		High
Package: Projects.GitHub.ATSPM.DecodeSiemensLogs		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/DecodeSiemensLogs/App.config:11 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.DecodeSiemensLogs.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/DecodeSiemensLogs/bin/Debug/DecodeSiemensLogs.exe.config 11 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.DecodeTrafficwareLogs		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/DecodeTrafficwareLogs/App.config:20 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Projects/GitHub/ATSPM/DecodeTrafficwareLogs/App.config:21 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.DecodeTrafficwareLogs.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/DecodeTrafficwareLogs/bin/Debug/DecodeTrafficwareLogs.exe.con 20 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Projects/GitHub/ATSPM/DecodeTrafficwareLogs/bin/Debug/DecodeTrafficwareLogs.exe.con 21 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.FTPfromAllControllersTests.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/FTPfromAllControllersTests/bin/Debug/FTPfromAllControllers.exe.confi 12 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA

A6 Sensitive Data Exposure

Many web applications do not properly protect sensitive data, such as credit cards, tax IDs, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes.

Password Management: Password in Configuration File		High
Package: Projects.GitHub.ATSPM.FTPfromasc3		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/FTPfromasc3/app.config:12 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.FTPfromasc3.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/FTPfromasc3/bin/Debug/FTPfromAllControllers.exe.confi 12 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.FileByFileASC3Decoder		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/FileByFileASC3Decoder/App.config:9 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.FileByFileASC3Decoder.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/FileByFileASC3Decoder/bin/Debug/FileByFileASC3Decoder.exe.con 9 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.GenerateControllerEventLogObjectText		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/GenerateControllerEventLogObjectText App.config:8 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.GenerateControllerEventLogObjectText.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/GenerateControllerEventLogObjectText bin/Debug/GenerateControllerEventLogObj 8 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA

A6 Sensitive Data Exposure

Many web applications do not properly protect sensitive data, such as credit cards, tax IDs, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes.

Password Management: Password in Configuration File		High
Package: Projects.GitHub.ATSPM.GetMaxTimeRecords		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/GetMaxTimeRecords/App.config:12 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Projects/GitHub/ATSPM/GetMaxTimeRecords/App.config:13 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.GetMaxTimeRecords.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/GetMaxTimeRecords/bin/Debug/GetMaxTimeRecords.exe.config 12 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Projects/GitHub/ATSPM/GetMaxTimeRecords/bin/Debug/GetMaxTimeRecords.exe.config 13 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.ImportChecker		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/ImportChecker/App.config:8 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.ImportChecker.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/ImportChecker/bin/Debug/ImportChecker.exe.config:8 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.MOE.Common.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.Common/bin/Debug/Microsoft.AspNetCore.Identity.x44 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA

A6 Sensitive Data Exposure

Many web applications do not properly protect sensitive data, such as credit cards, tax IDs, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes.

Password Management: Password in Configuration File		High
Package: Projects.GitHub.ATSPM.MOE.Common.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.Common/bin/Debug/Microsoft.AspNetCore.Identity.x1565 Analysis: Not an Issue	**Enclosing Method:** () **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/bin/Debug/Microsoft.AspNetCore.Identity.x1576 Analysis: Not an Issue	**Enclosing Method:** () **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/bin/Debug/Microsoft.AspNetCore.Identity.x1601 Analysis: Not an Issue	**Enclosing Method:** () **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/bin/Debug/Microsoft.AspNetCore.Identity.x1610 Analysis: Not an Issue	**Enclosing Method:** () **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/bin/Debug/Microsoft.AspNetCore.Identity.x1619 Analysis: Not an Issue	**Enclosing Method:** () **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/bin/Debug/Microsoft.AspNetCore.Identity.x1628 Analysis: Not an Issue	**Enclosing Method:** () **Source:**	SCA
Projects/GitHub/ATSPM/MOE.Common/bin/Debug/Microsoft.AspNetCore.Identity.x1649 Analysis: Not an Issue	**Enclosing Method:** () **Source:**	SCA

A6 Sensitive Data Exposure

Many web applications do not properly protect sensitive data, such as credit cards, tax IDs, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes.

Password Management: Password in Configuration File		High
Package: Projects.GitHub.ATSPM.MOE.Common.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.Common/bin/Debug/Microsoft.AspNetCore.Identity.x:2328 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Projects/GitHub/ATSPM/MOE.Common/bin/Debug/Microsoft.AspNetCore.Identity.x:2561 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.MOE.CommonTests		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.CommonTests/App.config:10 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.MOE.CommonTests.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOE.CommonTests/bin/Debug/MOE.CommonTests.dll.config:10 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Projects/GitHub/ATSPM/MOE.CommonTests/bin/Debug/SPM.dll.config:12 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Projects/GitHub/ATSPM/MOE.CommonTests/bin/Debug/WavetronicsSpeedListener.exe.:12 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA

A6 Sensitive Data Exposure

Many web applications do not properly protect sensitive data, such as credit cards, tax IDs, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes.

Password Management: Password in Configuration File		High
Package: Projects.GitHub.ATSPM.MOEWcfServiceLibrary		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOEWcf ServiceLibrary/App.config:15 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.MOEWcfServiceLibrary.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/MOEWcf ServiceLibrary/bin/Debug/MOEWcfServiceLibrary.dll.confi 15 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.NEWDecodeandImportASC3Logs		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/NEWDec odeandImportASC3Logs/App.config:11 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.NEWDecodeandImportASC3Logs.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/NEWDec odeandImportASC3Logs/bin/Debug/NEWDecodeandImportASC3Log 11 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA
Package: Projects.GitHub.ATSPM.SPM		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/We b.config:12 Analysis: Not an Issue	`Enclosing Method: ()` `Source:`	SCA

A6 Sensitive Data Exposure

Many web applications do not properly protect sensitive data, such as credit cards, tax IDs, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct credit card fraud, identity theft, or other crimes.

Password Management: Password in Configuration File		High
Package: Projects.GitHub.ATSPM.SPM.bin		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/bin/SPM.dll.config:12 Analysis: Not an Issue	**Enclosing Method:** () **Source:**	SCA
Package: Projects.GitHub.ATSPM.WavetronicsSpeedListener		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/WavetronicsSpeedListener/App.config:12 Analysis: Not an Issue	**Enclosing Method:** () **Source:**	SCA
Package: Projects.GitHub.ATSPM.WavetronicsSpeedListener.bin.Debug		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/WavetronicsSpeedListener/bin/Debug/WavetronicsSpeedListener.exe.(12 Analysis: Not an Issue	**Enclosing Method:** () **Source:**	SCA

Privacy Violation: Heap Inspection		High
Package: FTPfromAllControllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/FTPfromasc3/Program.cs:64 Analysis: Suspicious	**Sink:** Assignment to password **Enclosing Method:** Main() **Source:** Read this.Password **from** MOE.Common.Models.Repositories.SignalFTPInfo.get_Password() **In** Projects/GitHub/ATSPM/MOE.Common/Models/Repositories/SignalFTPInfo.cs:9	SCA

A7 Missing Function Level Access Control

Most web applications verify function level access rights before making that functionality visible in the UI. However, applications need to perform the same access control checks on the server when each function is accessed.

No Issues

A8 Cross-Site Request Forgery (CSRF)

CSRF attacks force an authenticated victim's browser to send an unauthenticated request to a vulnerable web application, which then performs unauthorized action on behalf of the attacker. CSRF can be as powerful as the web application that it targets.

ASP.NET MVC Bad Practices: Controller Action Not Restricted to POST		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DataExportController.cs:154 Analysis: Not an Issue	**Sink:** Function: GetRecordCount **Enclosing Method:** GetRecordCount() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:36 Analysis: Not an Issue	**Sink:** Function: SignalSearch **Enclosing Method:** SignalSearch() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:90 Analysis: Not an Issue	**Sink:** Function: GetMap **Enclosing Method:** GetMap() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:174 Analysis: Not an Issue	**Sink:** Function: GetPhaseTerminationMetricByUrl **Enclosing Method:** GetPhaseTerminationMetricByUrl( ) **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:186 Analysis: Not an Issue	**Sink:** Function: GetSplitMonitorMetricByUrl **Enclosing Method:** GetSplitMonitorMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:213 Analysis: Not an Issue	**Sink:** Function: GetPreemptMetricByUrl **Enclosing Method:** GetPreemptMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:224 Analysis: Not an Issue	**Sink:** Function: GetTMCMetricByUrl **Enclosing Method:** GetTMCMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:238 Analysis: Not an Issue	**Sink:** Function: GetPCDMetricByUrl **Enclosing Method:** GetPCDMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:255 Analysis: Not an Issue	**Sink:** Function: GetApproachVolumeMetricByUrl **Enclosing Method:** GetApproachVolumeMetricByUrl() **Source:**	SCA

A8 Cross-Site Request Forgery (CSRF)

CSRF attacks force an authenticated victim's browser to send an unauthenticated request to a vulnerable web application, which then performs unauthorized action on behalf of the attacker. CSRF can be as powerful as the web application that it targets.

ASP.NET MVC Bad Practices: Controller Action Not Restricted to POST	High
Package: SPM.Controllers	

Location	Analysis Info	Analyzer
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:275 Analysis: Not an Issue	**Sink:** Function: GetApproachDelayMetricByUrl **Enclosing Method:** GetApproachDelayMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:288 Analysis: Not an Issue	**Sink:** Function: GetAoRMetricByUrl **Enclosing Method:** GetAoRMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:299 Analysis: Not an Issue	**Sink:** Function: GetApproachSpeedMetricByUrl **Enclosing Method:** GetApproachSpeedMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:314 Analysis: Not an Issue	**Sink:** Function: GetYRAMetricByUrl **Enclosing Method:** GetYRAMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:332 Analysis: Not an Issue	**Sink:** Function: GetSplitFailMetricByUrl **Enclosing Method:** GetSplitFailMetricByUrl() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:369 Analysis: Not an Issue	**Sink:** Function: GetYellowAndRedMetric **Enclosing Method:** GetYellowAndRedMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:448 Analysis: Not an Issue	**Sink:** Function: GetSplitFailMetric **Enclosing Method:** GetSplitFailMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:512 Analysis: Not an Issue	**Sink:** Function: GetApproachSpeedMetric **Enclosing Method:** GetApproachSpeedMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/DefaultChartsController.cs:576 Analysis: Not an Issue	**Sink:** Function: GetAoRMetric **Enclosing Method:** GetAoRMetric() **Source:**	SCA

A8 Cross-Site Request Forgery (CSRF)

CSRF attacks force an authenticated victim's browser to send an unauthenticated request to a vulnerable web application, which then performs unauthorized action on behalf of the attacker. CSRF can be as powerful as the web application that it targets.

ASP.NET MVC Bad Practices: Controller Action Not Restricted to POST		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:636 Analysis: Not an Issue	**Sink:** Function: GetApproachDelayMetric **Enclosing Method:** GetApproachDelayMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:699 Analysis: Not an Issue	**Sink:** Function: GetPhaseTerminationMetric **Enclosing Method:** GetPhaseTerminationMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:754 Analysis: Not an Issue	**Sink:** Function: GetPreemptMetric **Enclosing Method:** GetPreemptMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:826 Analysis: Not an Issue	**Sink:** Function: GetTMCMetric **Enclosing Method:** GetTMCMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:894 Analysis: Not an Issue	**Sink:** Function: GetApproachVolumeMetric **Enclosing Method:** GetApproachVolumeMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:953 Analysis: Not an Issue	**Sink:** Function: GetSplitMonitorMetric **Enclosing Method:** GetSplitMonitorMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/ DefaultChartsController.cs:1062 Analysis: Not an Issue	**Sink:** Function: GetPCDMetric **Enclosing Method:** GetPCDMetric() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/FAQsController.cs:100 Analysis: Not an Issue	**Sink:** Function: Delete **Enclosing Method:** Delete() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Co ntrollers/LinkPivotController.cs:51 Analysis: Not an Issue	**Sink:** Function: LinkPivotResult **Enclosing Method:** LinkPivotResult() **Source:**	SCA

A8 Cross-Site Request Forgery (CSRF)

CSRF attacks force an authenticated victim's browser to send an unauthenticated request to a vulnerable web application, which then performs unauthorized action on behalf of the attacker. CSRF can be as powerful as the web application that it targets.

ASP.NET MVC Bad Practices: Controller Action Not Restricted to POST		High
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Controllers/LinkPivotController.cs:185 Analysis: Not an Issue	**Sink:** Function: PCDs **Enclosing Method:** PCDs() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/MenusController.cs:111 Analysis: Not an Issue	**Sink:** Function: Delete **Enclosing Method:** Delete() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/MetricCommentsController.cs:84 Analysis: Not an Issue	**Sink:** Function: Delete **Enclosing Method:** Delete() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/RouteSignalsController.cs:59 Analysis: Not an Issue	**Sink:** Function: RouteMap **Enclosing Method:** RouteMap() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/RouteSignalsController.cs:212 Analysis: Not an Issue	**Sink:** Function: Delete **Enclosing Method:** Delete() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/RoutesController.cs:94 Analysis: Not an Issue	**Sink:** Function: Delete **Enclosing Method:** Delete() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/SPMUsersController.cs:187 Analysis: Not an Issue	**Sink:** Function: Delete **Enclosing Method:** Delete() **Source:**	SCA
Projects/GitHub/ATSPM/SPM/Controllers/SignalsController.cs:625 Analysis: Not an Issue	**Sink:** Function: Delete **Enclosing Method:** Delete() **Source:**	SCA

A9 Using Components with Known Vulnerabilities

Components, such as libraries, frameworks, and other software modules, almost always run with full privileges. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover.

No Issues

A10 Unvalidated Redirects and Forwards

Redirects allow web applications to direct users to different pages within the same application or to external sites. Attackers can utilize open redirects to trick users into visiting a URL to a trusted site and redirecting them to a malicious site. Open redirects are often abused as part of phishing scams to harvest sensitive end-user data.

Open Redirect		Critical
Package: SPM.Controllers		
Location	**Analysis Info**	**Analyzer**
Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs: 697 Analysis: Not an Issue	**Sink:** System.Web.Mvc.Controller.Redirect() **Enclosing Method:** RedirectToLocal() **Source:** ExternalLoginCallback(0) **from** SPM.Controllers.AccountController.ExternalLoginCallback() **In** Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs:563	SCA
Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs: 697 Analysis: Not an Issue	**Sink:** System.Web.Mvc.Controller.Redirect() **Enclosing Method:** RedirectToLocal() **Source:** ExternalLoginConfirmation(1) **from** SPM.Controllers.AccountController.ExternalLoginConfirmation() **In** Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs:598	SCA
Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs: 697 Analysis: Not an Issue	**Sink:** System.Web.Mvc.Controller.Redirect() **Enclosing Method:** RedirectToLocal() **Source:** VerifyCode(0) **from** SPM.Controllers.AccountController.VerifyCode() **In** Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs:352	SCA
Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs: 697 Analysis: Not an Issue	**Sink:** System.Web.Mvc.Controller.Redirect() **Enclosing Method:** RedirectToLocal() **Source:** Login(1) **from** SPM.Controllers.AccountController.Login() **In** Projects/GitHub/ATSPM/SPM/Controllers/AccountController.cs:292	SCA

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.