Lessons Learned Building Secure ASP.NET Applications

Tom Fischer
AppSec USA 2011

OWASP

The OWASP Foundation
http://www.owasp.org

Provisos & Assumptions

- Presentation based on over 10 years' experience building web applications on the Microsoft stack for several clients in the Twin Cities
- Suspect some lessons learned will apply to any web project; not just those built with .NET
- Feel free to comment, disagree, question, *etc*.
- Finally, this is not a (very) technical presentation

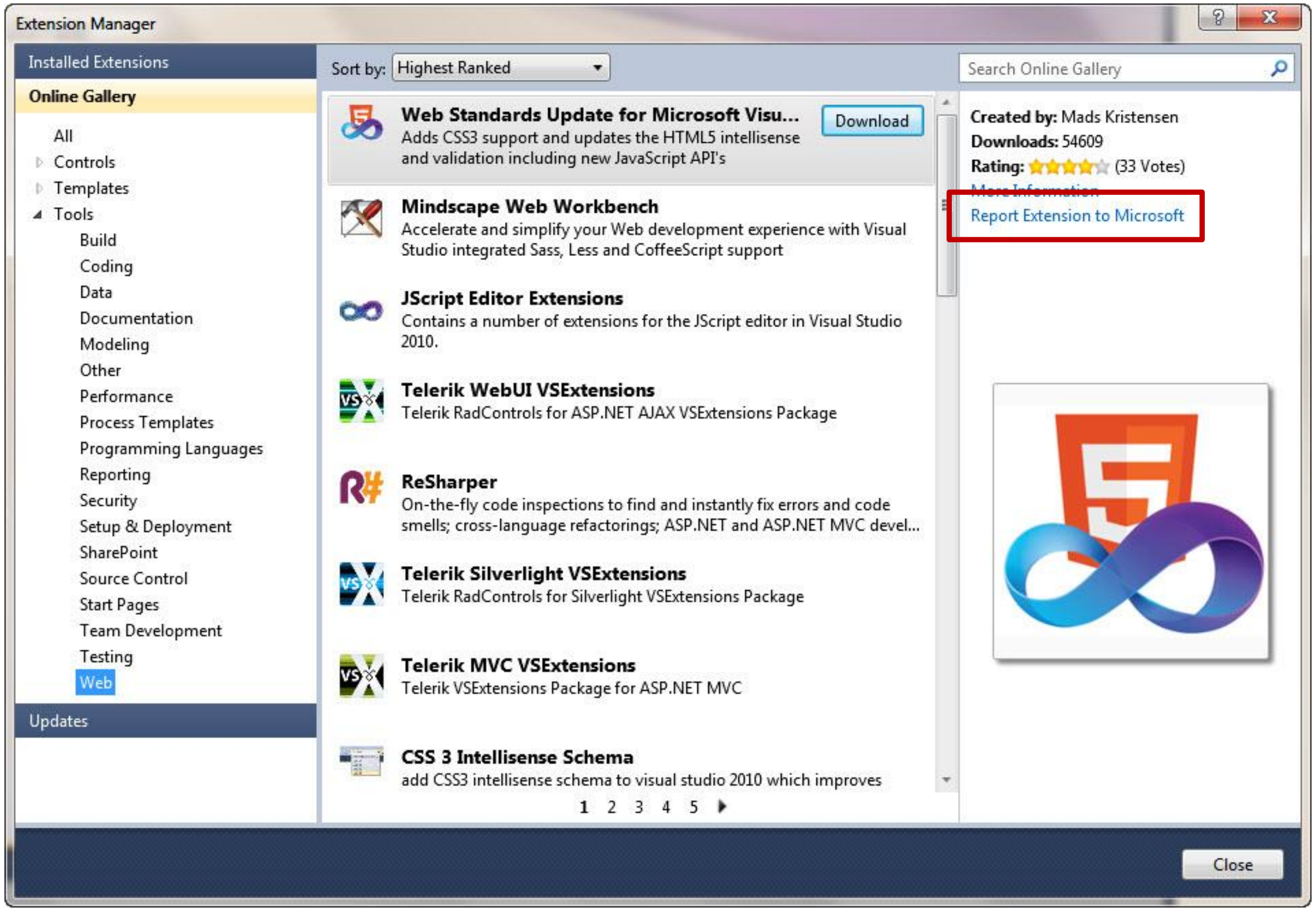
Lessons Learned Agenda

- The Environment
- Working with Tools
  - In-the-Box
  - Near-the-Box
- New Technologies, New Opportunities

The Environment: The Open Source Situation

- A few years ago Microsoft provided almost everything needed to build web applications
- Now a growing number of sources offer free, increasingly “mission critical” components sporting varying degrees of security, such as,
  - CodePlex
  - jQuery
  - SourceForge
  - and many blogs, personal web sites & ISVs

The Open Source Side Note: Visual Studio 2010 Extensions


Extension Manager

Installed Extensions
Online Gallery
All
Controls
Templates
Tools
Build
Coding
Data
Documentation
Modeling
Other
Performance
Process Templates
Programming Languages
Reporting
Security
Setup & Deployment
SharePoint
Source Control
Start Pages
Team Development
Testing
Web
Updates

Sort by: Highest Ranked

Web Standards Update for Microsoft Visu... Download
Adds CSS3 support and updates the HTML5 intellisense and validation including new JavaScript API's

Mindscape Web Workbench
Accelerate and simplify your Web development experience with Visual Studio integrated Sass, Less and CoffeeScript support

JScript Editor Extensions
Contains a number of extensions for the JScript editor in Visual Studio 2010.

Telerik WebUI VSExtensions
Telerik RadControls for ASP.NET AJAX VSExtensions Package

ReSharper
On-the-fly code inspections to find and instantly fix errors and code smells; cross-language refactorings; ASP.NET and ASP.NET MVC devel...

Telerik Silverlight VSExtensions
Telerik RadControls for Silverlight VSExtensions Package

Telerik MVC VSExtensions
Telerik VSExtensions Package for ASP.NET MVC

CSS 3 Intellisense Schema
add CSS3 intellisense schema to visual studio 2010 which improves

1 2 3 4 5

Search Online Gallery

Created by: Mads Kristensen
Downloads: 54609
Rating: ★★★★☆ (33 Votes)
More Information
Report Extension to Microsoft

Close

OWASP

The Environment:
The Open Source Recommendation

- Communicate a policy, formal or informal, regarding “open source” components
- At a minimum such a policy should
  - Enforce a requirement that the component’s source code should be available
  - Check source code into repository
  - Specify that deployed components are built from reviewed source code

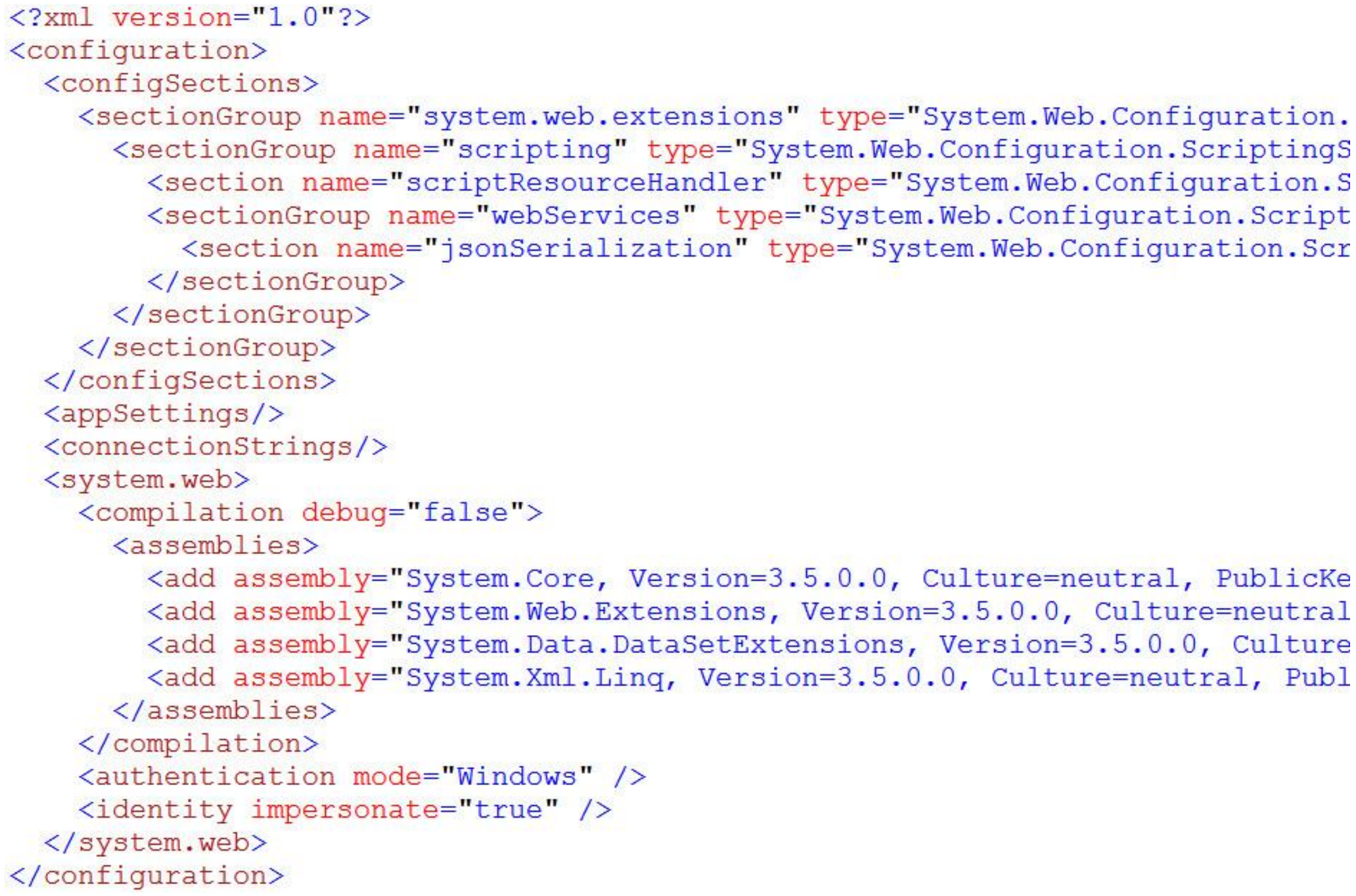
The Environment: Development Situation

- Application Concerns
  - Business requirements and “Look and Feel” dominate
  - Development best practices neither measured nor managed
- Team Composition
  - 1 senior, highly skilled lead
  - 2 to 4 moderately skilled developers
  - Not readily available… DBAs, Network Professionals, Security Compliance, Server Administrators, *etc*.

The Environment: Development Recommendation

- Add work tasks to the project plan ASAP
  - Solution previews and reviews with all parties
  - Documented code scans and reviews
- Designate a project OWASP Specialist
  - Not the lead
  - Not necessarily the “best” developer
  - Someone with an interest in security
- Train all web developers in organization
  - Create awareness of threats & solutions
  - Many inexpensive training options exist

The Environment: Configuration Sample



```
<?xml version="1.0"?>
<configuration>
  <configSections>
    <sectionGroup name="system.web.extensions" type="System.Web.Configuration.
      <sectionGroup name="scripting" type="System.Web.Configuration.ScriptingS
        <section name="scriptResourceHandler" type="System.Web.Configuration.S
        <sectionGroup name="webServices" type="System.Web.Configuration.Script
          <section name="jsonSerialization" type="System.Web.Configuration.Scr
        </sectionGroup>
      </sectionGroup>
    </sectionGroup>
  </configSections>
  <appSettings/>
  <connectionStrings/>
  <system.web>
    <compilation debug="false">
      <assemblies>
        <add assembly="System.Core, Version=3.5.0.0, Culture=neutral, PublicKe
        <add assembly="System.Web.Extensions, Version=3.5.0.0, Culture=neutral
        <add assembly="System.Data.DataSetExtensions, Version=3.5.0.0, Culture
        <add assembly="System.Xml.Linq, Version=3.5.0.0, Culture=neutral, Publ
      </assemblies>
    </compilation>
    <authentication mode="Windows" />
    <identity impersonate="true" />
  </system.web>
</configuration>
```

The Environment: Configuration Situation

- Controls almost every security aspect, such as, authentication, authorization, hashing algorithms, keys, *etc*.
- Internet Information Server (IIS) allows
  - Settings at the machine, root, site and folder
  - Supports overrides via GUI or code
- Few (if any) Developers or IT Professionals understand the behavior of all settings
- Infrastructure typically operates in either a lockdown or “it’s a developer thing” mode

The Environment: Configuration Recommendation

- Lock down machine level settings
  - Only allow Infrastructure to edit
  - Communicate variances from the default settings
- Treat application configuration files like code
  - Keep under source control
  - Adhere to existing deployment practices
- Learn about the *allowOverride* and family of *lockXXX* attributes
  - Available since .NET 2
  - Typically applied at the machine level

Configuration Side Note: Windows Communication Foundation (WCF)

- Situation
  - Hugely successful and productive feature; not always needed or included in a web application
  - Few developers understand the technology
  - Easy to create and overlook big security holes
- Recommendation
  - Establish early production and development settings
  - Enforce usage of the promulgated settings
  - Read Microsoft's *WCF Security Guide*

The Environment: Authentication Situation

- Most applications implicitly follow a *Trusted Subsystem* security design
  - User credentials checked “at the door”
  - Shared database and domain service accounts
- Credentials managed via one repository; typically Active Directory or SQL Server
- The above suggests one application’s breach may comprise other applications hosted on the same server(s) or sharing the same credentials store

The Environment: Authentication Recommendation

- Be wary of custom, application-specific credentials management solutions
- Strive for a credentials repository that is
  - Treated as an enterprise grade resource
  - Exposed by tightly controlled components
- Create domain service accounts that
  - Do serve specific functions or applications
  - Do not serve as super accounts

Authentication Side Note: Building the Enterprise Credentials Store

- Project task lists usually include security; rarely do they incorporate an enterprise resource
- Standards have lowered the costs and risks of implementing an enterprise credentials store
- Easier to implement (hide?) a credentials store within a moderately sized project
- Consider
  - Eclipse's Higgins Open Source Identity Framework
  - Microsoft's Windows Identity Foundation

In-the-Box: *validateRequest*

- Description
  - Request’s input data compared to a blacklist
- Typical Usage
  - On by default
  - Regularly turned off for various reasons, such as, broken user control or misbehaving AJAX
- Recommendation
  - Do not assume it is enabled
  - When enabled not 100% foolproof; still need to validate all input

In-the-Box: *Page.ViewStateUserKey*

- Description
  - Session unique view-state identifier checked on post backs
- Typical Usage
  - Coded on a page-by-page basis
  - Turned off for performance (via *EnableViewStateMac)*
- Recommendation
  - If used, implement in a base page
  - When used not 100% foolproof against CSRF attacks; well documented help exists

In-the-Box: *maxRequestLength*

- Description
  - Limits input stream’s buffering threshold
- Typical Usage
  - Default set to 4KB
  - Set to handle any request at the application level
- Recommendation
  - Set machine level to default; allow overrides
  - Lock at the application level (no large file uploads)
  - Force developers to set it explicitly at the page level

Side Note: *HttpRuntime* Settings

- Section contains *maxRequestLength* along with many other critical properties, such as, *maxUrlLength* and *enableHeaderChecking*
- Most *HttpRuntime* default settings work well; tweak them with care and caution

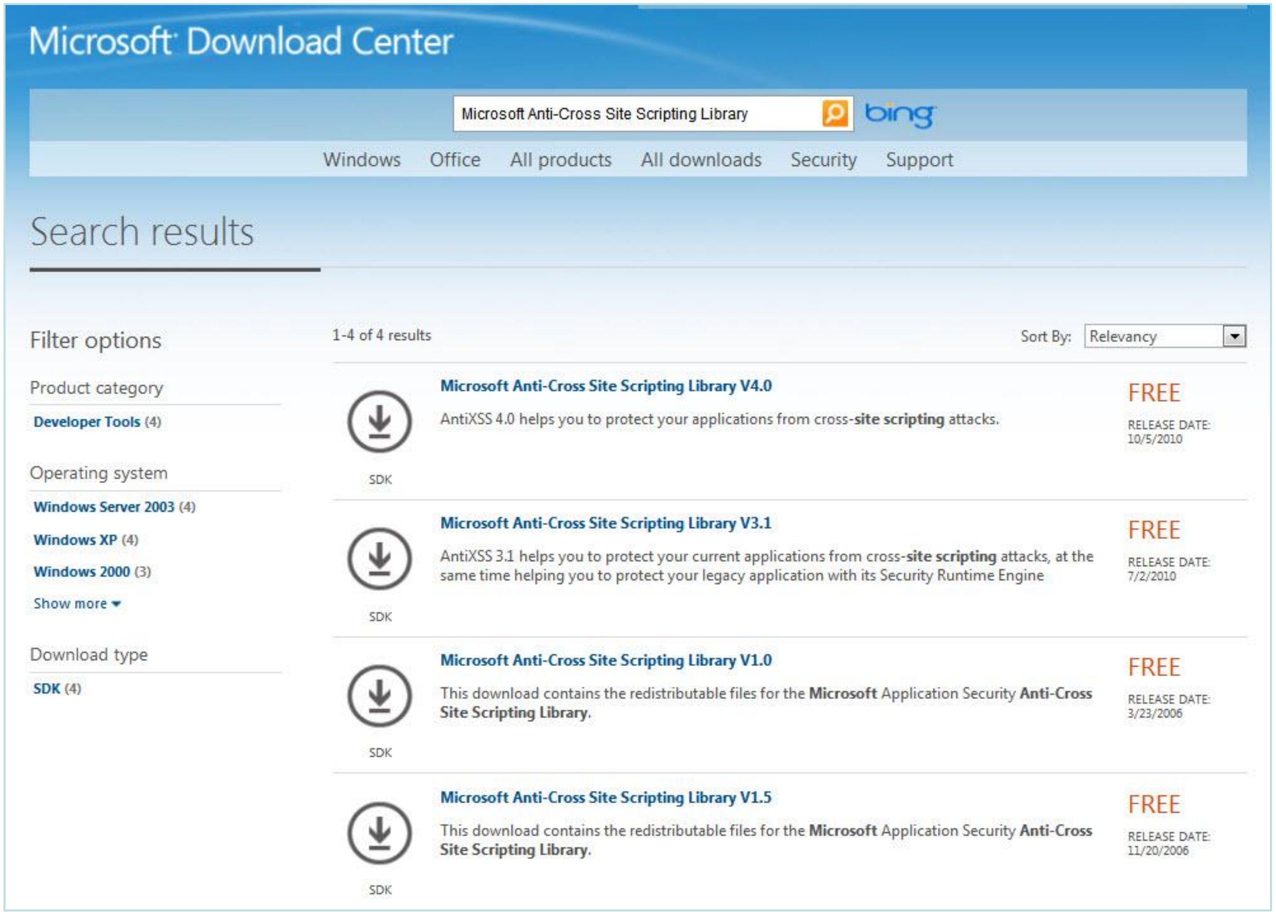
Near-the-Box: Anti-Cross Site Scripting Library (Anti-XSS)

- Description
  - Encoding functions for CSS, HTML, HTML attributes, JavaScript, XML, *etc*. based on a globalized whitelist
- Typical Usage
  - Library not extremely well-known or utilized
  - Newer versions not always applied
- Recommendation
  - Incorporate into a project from the start
  - Update to latest version when scheduling 100% integration testing run

Near-the-Box: Anti-Cross Site Scripting Library (Anti-XSS)

- Some of what's new in 4.0
  - Adjustable safe-listing for HTML/XML encoding
  - Invalid Unicode character detection
  - *HtmlFormUrlEncode*
  - LDAP encoding changes
- Each new version reflects a solution to late breaking attacks; albeit delayed

Side Note: Anti-XSS Updates History


Microsoft Download Center
Microsoft Anti-Cross Site Scripting Library
bing
Windows
Office
All products
All downloads
Security
Support

Search results

Filter options

Product category
Developer Tools (4)

Operating system
Windows Server 2003 (4)
Windows XP (4)
Windows 2000 (3)
Show more

Download type
SDK (4)

1-4 of 4 results

Sort By: Relevancy

Microsoft Anti-Cross Site Scripting Library V4.0
AntiXSS 4.0 helps you to protect your applications from cross-**site scripting** attacks.
SDK — FREE — RELEASE DATE: 10/5/2010

Microsoft Anti-Cross Site Scripting Library V3.1
AntiXSS 3.1 helps you to protect your current applications from cross-**site scripting** attacks, at the same time helping you to protect your legacy application with its Security Runtime Engine
SDK — FREE — RELEASE DATE: 7/2/2010

Microsoft Anti-Cross Site Scripting Library V1.0
This download contains the redistributable files for the **Microsoft** Application Security **Anti-Cross Site Scripting Library.**
SDK — FREE — RELEASE DATE: 3/23/2006

Microsoft Anti-Cross Site Scripting Library V1.5
This download contains the redistributable files for the **Microsoft** Application Security **Anti-Cross Site Scripting Library.**
SDK — FREE — RELEASE DATE: 11/20/2006

Near-the-Box:
FXCop ASP.NET Security Rules

- Description
  - Specialized static code analysis executable by Visual Studio or the stand alone FXCop tool
  - Checks ASP.NET and ASP.NET MVC best practices
- Typical Usage
  - Not commonly applied (like most other FXCop rules)
- Recommendation
  - Get them incorporated into the build cycle; consider applying them at all code check-ins
  - Requires selling Architect, Build Manager and Project Manager

Side Note:
Why Not FXCop?

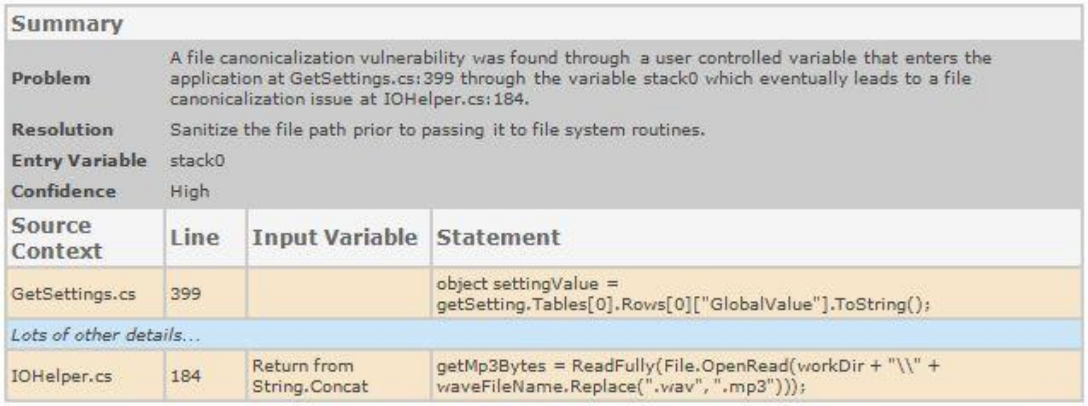
- Failed rules could prevent code repository check-ins or break the build
- Generates many messages
  - Especially if applied after code complete
  - Likely ignored or requiring unplanned repair work
- While customizable and flexible, managing FXCop consumes time and skill

Near-the-Box:
CAT.NET

- Description
  - Static code analysis identifying security vulnerabilities
- Typical Usage
  - Limited due to beta status at Microsoft
  - 3rd party runs tool, interprets and presents results
- Recommendation
  - Given the price (free), it's worth exploring
  - When (if) released, evaluate it
  - Alternatives (pricey) exist of varying quality

Side Note: CAT.NET Output from a Small Application

- Data flow graph of 231,295 nodes
- Execution time of 51.7 minutes
- 1,083 issues reported (many duplicates)
- Sample of an issue…



Summary			
Problem	A file canonicalization vulnerability was found through a user controlled variable that enters the application at GetSettings.cs:399 through the variable stack0 which eventually leads to a file canonicalization issue at IOHelper.cs:184.		
Resolution	Sanitize the file path prior to passing it to file system routines.		
Entry Variable	stack0		
Confidence	High		
Source Context	Line	Input Variable	Statement
GetSettings.cs	399		object settingValue = getSetting.Tables[0].Rows[0]["GlobalValue"].ToString();
Lots of other details...			
IOHelper.cs	184	Return from String.Concat	getMp3Bytes = ReadFully(File.OpenRead(workDir + "\\" + waveFileName.Replace(".wav", ".mp3")));

OWASP

New Technologies, New Opportunities: Azure

- Description
  - Microsoft’s cloud computing solution
  - Likely to grow with cloud-based computing movement
- Potential Risks
  - Cost-driven vulnerabilities, such as,
    - Riskier JavaScript when “doing more” on the browser
    - Accidentally comingling data between storage models
  - 24x7 high-volume traffic may mask probing
  - Unauthorized access to administrative UI
    - Unknown party
    - Previously authorized party

New Technologies, New Opportunities: Entity Framework

- Description
  - Strongly-typed LINQ-based data access
  - Well received feature, usage likely to grow
- Potential Risks
  - Buries connection in new type of configuration setting
  - Validates with database constraints and code (declarative & imperative)
  - Executes SQL without stored procedures
    - Loss of DBA oversight
    - Over granting of permissions to enable feature
  - Eases direct UI-Database communications

New Technologies, New Opportunities: MVC

- Description
  - Microsoft's implementation of a Model-View-Controller
  - Well received feature, usage likely to grow
- Potential Risks
  - Facilitates secure coding practices, does not obviate the need to do so disappear
  - Eases incorporating DOS and SQL injection vulnerabilities when improperly combined with Entity Framework
  - Enjoys a very innovative environment which may let questionable code slither in

Side Note: What about Ajax and jQuery?

- .NET enjoys the same capabilities as most applications to easily create vulnerabilities with these rich technologies
- Don’t forget to watch over those uniquely .NET, well documented properties, such as,
  - *IsDebuggingEnabled*
  - *ScriptManager.ScriptMode*
- Many ASP.NET methods can help
  - *Ajax.ActionLink*
  - *ValidateAntiForgeryToken*