Performing a Security Code Review

Hans Westphal
Development Support Consultant
Microsoft Corporation
What we will cover:

- Microsoft® Security Framework (SD3+C)
- Performing a Security Code Review
- Common Security Areas
- Tools to help with a Code Review
- Best Practices
- Resources
Session Prerequisites

- Basic understanding of Security
- Familiar with Visual C#®
- Familiar with Visual Basic® .NET

Level 200
Are you doing security code reviews in your development process today?

Polls are open.

Yes

No

[ Poll A ]
Microsoft Security Framework (SD3+C)

- Secure by Design
  - Mandatory training for every developer
  - Building of threat models for every product
  - Conducting code reviews and penetration testing
  - Using automated code tools
- Secure by Default
  - Reduction of attack surface
- Security in Deployment
  - Patch management tools
- Communication
  - Books, webcasts, premier services, security bulletins

Microsoft Security Framework (SD3+C)
Microsoft Security Framework Development Process

- Design
  - Security Review: Each component team develops threat models, ensuring that design blocks applicable threats

- Development
  - M1
  - M2
  - Mn
  - Beta
  - Develop & Test: Apply security design & coding standards, tools to eliminate code flaws (PREFix & PReFast), monitor & block new attack techniques
  - Security Push: Team-wide stand down, threat model updates, code review, test & documentation scrub

- Release
  - Security Audit: Analysis against current threats, internal & 3rd party penetration testing

- Support
  - Security Response: Fix newly discovered issues, root cause analysis to proactively find and fix related vulnerabilities

Microsoft Security Framework Development Process
Performing a Security Code Review
Why do security code reviews?

- Uncovers potential security issues early in the development process
  - For example the MS03-007 security did not affect windows server 2003 because it was detected in a security code review
- Provides a fast way for new team developers to learn how to identify common security defects and learn best practices
- Some studies say every bug removed during a review saves 9 hours in testing, debugging and fixing the code

Performing a Security Code Review  Why do security code reviews?
Performing a Security Code Review

Formal Team

- Moderator
  - Manages the meeting and follows up on issues found

- Reader
  - Paraphrases the program flow (never the author)

- Recorder
  - Logs each bug found

- Author
  - Understand programs and make ambiguity clear
Performing a Security Code Review
Informal Team

- Rapid Peer Reviews
  - Developers Pair up
  - Each take turns in questioning assumptions in each others code

- Individual Review
  - Try to look at your code from an attacker perspective
  - Use Automated tools
Performing a Security Code Review
Understanding What to Look for

- Risks identified in the Threat model tell you which code to look at first and deepest
- If the code is questionable, a re-write is not out of the question
- Get your developers training on what common security bugs look like
- Checklists
Performing a Security Code Review Checklists

- Use checklists to drive in the security review

Sample questions
- Does the code run by default?
- Does the code run with elevated privileges?
- Is the network interface unauthenticated?
- Is the code written in C/C++®
- Does the code handle sensitive or private data?


Performing a Security Code Review Checklists
Performing a Security Code Review  
Multiple-Pass Approach

- First pass through the code
  - Understand the environment, data structures and initialization.
  - Identify entry points into the code
  - Linkages between functions
  - Mark complex areas for more detailed review
Performing a Security Code Review Multiple-Pass Approach (cont.)

- Second pass through the code
  - Start from each entry point identified in first pass and trace each execution path
  - Look for patterns that the developer used to implement the solution.

- Next pass through the code
  - Review code Function by Function spending most time on areas you flagged in the first pass through the code.
Common Security Areas
Un-Validated User Input

- All input from users is evil
- Does your application blindly trust input?
- Leads to buffer overflows, cross-site scripting, and SQL injection

Recommendations (requirement)
- You must validate type, length, format and range
- Use built-in validation controls in .NET
Common Security Areas Buffer Overruns

An attack in which a malicious user exploits an unchecked buffer in a program and overwrites the program code with their own data potentially executable code.

```c
void function(char *p) {
    char buff[16];
    strcpy(buff, p);
}
```

Recommendations

- Locate calls to unmanaged code by searching for System.Runtime.InteropServices
- Validate all user input for length and types

Common Security Areas Buffer Overruns
Common Security Areas  Hard Coded Strings

- Analyze the presence of any sensitive information in hard coded strings
- Common string include Database connection strings, SQL Text commands and Application configurations strings

Recommendations
- Encrypt all sensitive strings stored in configuration files or store encrypted string in the registry (strongly ACL'd) under HKLM, HKCU
- See HOW TO: Use the ASP.NET Utility to Encrypt Credentials and Session State Connection Strings at http://support.microsoft.com/default.aspx?scid=kb%3Ben-us%3B329290
Common Security Areas
Database Access Code

- SQL injection
  - occurs when input parameters are used to construct SQL statements.

```
Dim strSQL as string = "select * from UserTable
               where Username='" & txtUserName.text & ")"
```

- Recommendations
Demonstration 1

demo

Find Hard Coded Strings with Findstr and ildasm
A Sharing Slide

[ Share A ]
Common Security Areas
Web Page Code

Cross Site Scripting (XSS)
- Occur whenever code uses input parameters in the output HTML stream that is returned to the client.

Recommendations
- Identify any code that outputs input by searching on strings like "<%= %>", "Write" and verify the input has been validated.
- Check that output is encoded appropriately.
- validateRequest attribute in .NET Framework 1.1
Demonstration 2

XSS Scripting Exploit
A Sharing Slide

[ Share B ]
Common Security Defects
Information Disclosure

- Application errors revealing information that may lend to a future attack
- Exception handling
- Recommendations
  - Fail early, hide error details, and log error info
  - For web apps, make sure the `<custom Error>` configuration element is set to “on” or “remoteonly”
  - Log errors inside the Application_Error() event handler.
  - Configure Web Server with a common User friendly error page.
Common Security Areas
General Managed Code issues

- Secure Class Design

- Recommendations
  - Reduce attack surface by limiting type and member visibility. Verify if anything marked public needs to be public.
  - Seal non-base classes
  - Never expose fields, use Properties
  - Implement IDisposable on your classes to ensure all resources are freed when you object instance is destroyed
Common Security Areas
General Managed Code Issues

Serialization
- Basically taking some data structure and pumping it out into a stream of bytes that we can then use.

Recommendations
- Search code base on “Serializable” to locate serialization code.
- If data is sensitive, mark it with [NonSerialized] attribute.
- If sensitive data needs to be serialized be sure to perform an access check upon deserialization.
Common Security Areas
General Managed Code Issues

Multithreading
- Race conditions, static constructors, and cache security checks occur when accessed from more than one thread at a time.

Recommendations
- Search for “Thread” in code tree to verify if multithreading is present.
- Be sure to synchronous R/W access to any static members (i.e. Dispose methods)
Common Security Areas
General Managed Code Issues

Code access security

- Feature in .NET framework that allows code to be trusted to varying degrees, depending on where the code originates and on other aspects of the code's identity.

Recommendations

- All assemblies are strongly named

Common Security Areas  General Managed Code Issues
Common Security Areas  General Managed Code Issues cont

- Code access security con’t

Recommendations

- Public Member restricted Access
- Unmanaged code (search on System.Runtime.InteropServices)
- Watch the upcoming webcast “Protect your IP with Code Access Security”
Common Security Areas
Storing Secrets

Secrets
- Passwords, connectionstrings and other sensitive data required by the Assembly to run

Recommendations
- Search code base for common secrets like: "password", "pwd", "constr", "Userid"
- Can you write the code without secrets?
- Encrypt passwords and connection strings
  (see knowledge base article number 329290; http://support.microsoft.com/default.aspx?scid=kb.EN-US.329290)
Common Security Areas Cryptography

- Most development projects should be using cryptography

**Recommendations**
- Don’t roll your own encryption classes, use System.Security.Cryptography namespace
- Use AES or Triple DES for data persisted for long periods of time; DES or RC2 for data persisted for short periods of time
- Use largest key sizes possible
- Use System.Security.Cryptography.RNGCryptoServiceProvider to generate random numbers
FxCop Tool
What is it?

- A code analysis tool that checks .NET managed code assemblies for conformance to the Microsoft .NET Framework Design Guidelines
- FxCop analyzes programming elements in assemblies, called targets, by using a set of customizable and extendable rule architecture.
- Available for free at http://www.gotdotnet.com/team/fxcop/
Demonstration 3

demo

FxCop Analysis Tool

Demonstration 3
A Sharing Slide

[ Share C ]
Best Practices

- Security code reviews are a part of Security-driven development process
- “Security is a process, not a product” Bruce Schneier
- Never, ever trust incoming data
- Don’t tell the attacker anything
- Security Education for Developers is continuous
Session Summary

- Where security reviews fit in a secure development methodology
- Building a Security Code Review team
- Common security defects
- Tools to help us with Security Code Reviews
- Best Practices
Call to Action

- General Education of Computer Security
- Read “Writing Secure Code” by Michael Howard and David LeBlanc
- Subscribe to security DL’s
  - http://www.securecoding.org/
  - http://www.ntbugtraq.com/

Call to Action
Additional Resources


- Writing Secure Code 2nd Edition by Michael Howard and David LeBlanc

- Open Web Application Security Project http://www.owasp.org/index


- Improving Web App Security - Threats and Countermeasures (June 2003). Available online at

- Data Access Application Block. Available online at

- Exception Management Architecture Guide” at

- FxCop - a code analysis tool that checks .NET assemblies for conformance to the Microsoft .NET Framework Design Guidelines. Available online at
  http://www.gotdotnet.com/team/fxcop/
Questions and Answers

Submit questions using the text box
### Get Up to Speed on .NET

Get Trained on Microsoft Developer Technologies


All times are Pacific Standard Time

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monday, Feb 16</td>
<td>1:00PM</td>
<td>MSDN Webcast: Implementing Authentication and Authorization with AD/AM and AzMan</td>
</tr>
<tr>
<td>Tuesday, Feb 17</td>
<td>9:00AM-10:30AM</td>
<td>MSDN Webcast: Computer Crime and Security</td>
</tr>
<tr>
<td>Tuesday, Feb 17</td>
<td>11:00AM-12:00PM</td>
<td>MSDN Architecture Webcast: Advanced Architecture Issues</td>
</tr>
<tr>
<td>Tuesday, Feb 17</td>
<td>1:00PM-2:30PM</td>
<td>MSDN Webcast: Working with Classes and Class Libraries</td>
</tr>
<tr>
<td>Tuesday, Feb 17</td>
<td>1:00PM-2:30PM</td>
<td>MSDN Webcast: Creating a Single Sign-On Enterprise Security Portal</td>
</tr>
<tr>
<td>Wednesday, Feb 18</td>
<td>9:00AM-10:30AM</td>
<td>MSDN Webcast: Dave’s Secure Remoting Chat Application</td>
</tr>
<tr>
<td>Wednesday, Feb 18</td>
<td>1:00PM-2:30PM</td>
<td>MSDN Webcast: Protecting Your System From SQL Injection Attacks</td>
</tr>
</tbody>
</table>

Get Up to Speed on .NET
MSDN Security Webcast Week
February 16 - 20, 2004

13 Live Webcasts - Top Industry Security Experts
MSDN Webcast: .NET Code Access Security - Level 200
MSDN Webcast: How to Perform a Security Review - Level 200
MSDN Webcast: Implementing Authentication and Authorization with AD/AM and AzMan
MSDN Webcast: Computer Crime and Security - Level 200
MSDN Webcast: Creating a Single Sign-On Enterprise Security Portal - Level 200
MSDN Webcast: Dave's Secure Remoting Chat Application - Level 300
MSDN Webcast: Writing Secure Code - Best Practices - Level 300
MSDN Webcast: Protecting Your System From SQL Injection Attacks - Level 200
MSDN Webcast: Application Hacking Techniques and How to Stop Them - Level 200
MSDN Webcast: Protect Your IP with Code Access Security - Level 200
MSDN Webcast: Essentials of Application Security - Level 300
MSDN Webcast: Writing Secure Code - Threat Defense - Level 300
MSDN Webcast: Dave's Top 10 Ways to Secure Your Web Application - Level 300

Register now at: http://msdn.microsoft.com/security
Learn more on Security....

- **MSDN Events** are free events for developers
- **March and April MSDN Events** focused on Security
  - Session #1: Writing Security Code – Threat Defense
    - Discuss common security threats faced by application developers
    - Effective strategies and best practices to defend against threats.
  - Session #2: Implementing Application Security using the .NET Framework
    - Implement additional security features to secure applications built on the .NET Framework.
    - Security features integrated into the .NET Framework
    - How to use the cryptographic provider support in the .NET Framework to encrypt and sign data.
    - Security best practices for Web applications and Web services built using ASP.NET.
  - Register today at: [www.msdnevents.com](http://www.msdnevents.com)

Learn more on Security....
DevDays is coming to a city near you

Register now for just $99 USD AND get a FREE Developer Welcome Kit.

http://www.microsoft.com/devdays2004

Each DevDays attendee will receive:

- DevDays DVD featuring presentations and sample code for all sessions
- “Whidbey” pre-release software
- ASP.NET Resource CD – NEW!
- Smart Client Resource DVD
- SQL Server 2000 Evaluation Edition