

CHEAT SHEET

Risk	ASP.NET Webforms	ASP.NET MVC	SharePoint
SQL Injection	Parameterise queries with SqlParameter, so SQL queries have @ParamName rather than concatenated with user input. ⁱ Review stored procedures for safe usage of execute, exec or sp_executesql. ⁱⁱ		Use the SharePoint object model where possible. See ASP.NET tips when doing direct DB access.
Cross Site Scripting (XSS)	Use HttpUtility or AntiXSS's Html and Attribute Encode, and/or the Security Runtime Engine. ⁱⁱⁱ	Use Razor views, or encode output using <%= val %> in Web Form views.	Use SPHttpUtility to encode output or safely output HTML. ^{iv}
Broken Authentication & Session Management	Use built-in ASP.NET Session State and Forms Auth where possible. Store server-side state to enforce logout. Make sure cookies are 'HttpOnly' and 'secure', and all login and password entry is over SSL/TLS. When rolling your own Forms Auth provider (e.g. for Claims-based Authentication), prevent brute-force of usernames or passwords by logging failed username / password attempts. Alert sysadmins on security events.		
Insecure Direct Object References	Use location/authorization tags in web.config ^v , User.IsInRole("Admin") and custom checks in code.	Decorate controller with [Authorize(Roles="Admin")] and check inside controller actions if needed.	Use SecurableObject's DoesUserHavePermissions or CheckPermissions to take action or error. ^{iv}
Cross Site Request Forgery (CSRF)	In your page's OnInit, set Page.ViewStateUserKey ^{vi} to Session.SessionID or the user's username.	Add HtmlHelper AntiForgeryToken to every form, and add attribute to each action method. ^{vii}	Use the FormDigest field on every request, and call ValidateFormDigest to check it is set correctly. ^{viii}
Security Misconfiguration	Is all software up to date? E.g. Windows Update, libraries, database and applications. Is everything that is unused disabled? Ports closed? Passwords changed from defaults? Run the Best Practice Analyzer for each platform that is in use. Encrypt sensitive information in web.config such as connection strings. ^{ix}		
Insecure Cryptographic Storage	Passwords should not be stored in clear-text – use salted SHA-256 hashes instead. Sensitive data should be encrypted everywhere it is stored long term (e.g. backups). Credit card data requires following PCI standards. Health records or personally identifiable information may be subject to legislation. Don't invent super crypto ourselves! (DISCO)		
Failure to Restrict URL Access	Consider separating out your admin section of the site, so it's not internet-accessible. Don't put source control files, readme, version.txt, database backups etc in web-accessible locations.		
Insufficient Transport Layer Protection	Require SSL/TLS for all sensitive pages, including login, password entry, authenticated pages and credit card entry. Avoid including http URLs in https pages. Set the 'secure' flag on cookies. Ensure the server is configured to prevent SSLv2 and weak ciphers from being used.		

Risk	ASP.NET Webforms	ASP.NET MVC	SharePoint
Unvalidated Redirects & Forwards	Ensure that any Redirects or Transfers only go to white-listed domains. Check RedirectUrl handling on login page doesn't allow other URLs to be redirected to.		Use SPUtility.Redirect to send the user to another page on the same domain. ^{iv}
File downloads	Send the following HTTP headers with file downloads: Content-Disposition: Attachment X-Download-Options: noopen X-Content-Type-Options: nosniff Content-Type: [mime type] See our File Upload Considerations whitepaper for more things to think about. ^x		

For more information about these risks, visit the Open Web Application Security Project - <http://owasp.org>

Aura Information Security provides:

- Secure Developer Training
- Penetration Testing
- Code Reviews and Information Security Architecture
- Daily and PCI RedEye Vulnerability Scans

www.AuraInfoSec.com

An electronic version of this sheet is available at: www.aurainfosec.com/publications

ⁱ <http://msdn.microsoft.com/en-us/library/ms172108.aspx>
ⁱⁱ <http://msdn.microsoft.com/en-us/library/ms161953.aspx>
ⁱⁱⁱ <http://wpl.codeplex.com/>
^{iv} <http://msdn.microsoft.com/en-us/library/gg552614.aspx>
^v <http://support.microsoft.com/kb/316871>
^{vi} <http://msdn.microsoft.com/en-us/library/system.web.ui.page.viewstateuserkey.aspx>
^{vii} <http://msdn.microsoft.com/en-us/library/dd504812.aspx>
^{viii} <http://msdn.microsoft.com/en-us/library/ms472879.aspx>
^{ix} [http://msdn.microsoft.com/en-us/library/dx0f3cf2\(v=VS.85\).aspx](http://msdn.microsoft.com/en-us/library/dx0f3cf2(v=VS.85).aspx)
^x <http://www.aurainfosec.com/publications>