Web Application Audit Checklist

Basic Configuration

- Is default content or sample content installed on the web server?
  - If so, why is this necessary? Is any of this material executable code?
- Check that directory indexing has been disabled
- Compare the configuration of the server with manufacturer or community security recommendations. Have appropriate controls and options been configured?
- Have the server headers been sanitized?
- What security baseline has been implemented on the underlying operating system? Has the security of this configuration been audited?
- Does the network architecture support the security and information flow requirements of the web architecture?
- Use at least one automated tool to evaluate the web site
  - Are there any issues identified by the tools?
  - What sorts of risks do these issues represent?
  - What controls already exist to mitigate the risks?
  - Are the responsible individuals capable of remediating the risks identified?

Authentication

- Is authentication required? If so:
  - How is authentication accomplished?
  - If Basic authentication is used, is it appropriate for the level of sensitivity for the data?
  - If Basic is used, is SSL required?
  - If forms are used, is the POST method used?
  - If forms are used, is SSL required?
  - If certificates are used, how are certificates controlled?
  - If certificates are used, how is the CRL managed?
  - How are account lockouts handled?
    - Are speed bump lockouts in use?

Session Management

- What session management and tracking technique is in use?
- Basic Authentication
  - Is SSL required at all times after the username and password are requested?

    - Is there a sign off procedure to force the credentials to be dropped from the browser?
    - Is there some form of tracking in place to identify brute force password guessing attempts?
    - How are account lockouts handled?
- URL Rewriting/Hidden form fields
    - Are the session IDs sufficiently random?
    - Are the session IDs sufficiently large?
    - Is the generation of the session ID based on any aspect of the user or password information?
    - Are session IDs sent over appropriately secure paths? If not, how is this mitigated?
    - How are account lockouts handled?
    - Is there a clear sign off procedure to expire a session ID?
    - Are session IDs perishable?
- Are the session IDs in the session token of sufficient length for the application?
- Are the session IDs secured appropriately based on cloning detection capabilities?
- What type of session hijacking/cloning detection capabilities exist?
- What actions does the application take when a session violation is detected?
- Are the session IDs sufficiently random?
- Do session IDs expire after some period of time?
- Is a valid session required in all appropriate circumstances? How is this enforced or mediated?

Input

- How is input to the application sanitized?
- Is input sanitized in all cases, even if some cases have less restrictive rules?
- Is sensitive information always sent using a POST rather than a GET?
- How robust is the application when dealing with unexpected or illegal input?

Output

- How are error conditions handled?
- Is it possible to cause the application to generate an unhandled error?
- Is encryption used in all cases where sensitive information is returned?
- Are there any anti-caching techniques in use when sensitive information is returned?
- Are all special characters properly stripped or escaped when returned in a web page?