HISPUB 007.1.56

The United States House of Representatives Information Security Publication – Web Site Developers Security Checklist

Version:	**111 (111th Congress)**
Approved:	**August 2009**
Approval Authority:	**Director, Information Systems Security**

Notes:

- You should not attempt to implement any of the settings in this checklist without first testing in a non-operational environment.

- This document contains recommended security settings. Some applications may be adversely affected by the settings. If a setting cannot be implemented as suggested, the Information Systems Security Office will work with you to find an alternate solution.

- This is a living document and will be reviewed regularly; the change log at the end of this document will list modifications.

- All sections of the following checklist must be completed unless noted as "recommended" or "optional".

Item #	Item description	Problematic?	
1.	The OS must be hardened in accordance with the proper hardening guidelines.	Problem	Non-Problem
2.	The web engine must be hardened in accordance with the proper hardening guidelines.	Problem	Non-Problem
3.	All available patches must be installed. This includes OS, web engine, and application patches.	Problem	Non-Problem
4.	If the site is externally facing, then site should only host data that you want the whole world to see. The server that the external site is hosted on should likewise contain only data that you want the whole world to see.	Problem	Non-Problem
5.	If the site does host data that you don't want the world to see, then it should not be externally accessible and should not be hosted on a server that is externally accessible.	Problem	Non-Problem
6.	Remove all test, dev, backup, and unnecessary files.	Problem	Non-Problem
7.	Disable any unnecessary Web services if not required. For example on IIS, disable Remote Data Services (RDS), WebDAV, on Apache disable mod_dav, mod_dav_fs, mod_dav_lock etc.	Problem	Non-Problem
8.	For Content Management Systems (CMS) all administrative components must be completely removed from the public facing website. The administrative CMS functionality must be hosted on a separate site that is only accessible from the House networks. (See diagram 1)	Problem	Non-Problem
9.	Authentication methods for content management systems should match all House password and account management policies and guidelines.	Problem	Non-Problem
10.	CMS must utilize encryption to ensure that user credentials cannot be compromised in transit.	Problem	Non-Problem
11.	CMS must utilize encryption to ensure that user credentials cannot be compromised when at rest.	Problem	Non-Problem
12.	Every site should have its own virtual hostname. *membername.house.gov* directory sites like *www.house.gov/membername* should be avoided.	Problem	Non-Problem
13.	For dynamic sites that require the use of a database, should only use databases that allow very granular level of permissions in the database. Access, Foxpro, and DB should not be used.	Problem	Non-Problem

14.	For dynamic sites that require the use of a database, each site must use its own database. Sites should not share databases.	Problem	Non-Problem
15.	For dynamic sites that require the use of a database, each database must have a corresponding account that is being used for the public internet user to query the database. This account must have an extremely robust password.	Problem	Non-Problem
16.	For dynamic sites that require the use of a database, the account being used to query the database for the public internet user must have minimal permissions within the database. In most cases, query permissions are all that are required. In some cases, where forms are being submitted into a database, insert permissions may also be required.	Problem	Non-Problem
17.	Every user input value, including cookies, must be validated by the application. Applications using .asp, .aspx, .php, .jsp, .pl, .cfm, etc, must all perform input validation functions to ensure that the variables being passed to the application are the variables expected.	Problem	Non-Problem
18.	Input validation must use "good lists" where possible. This means that the input validation filters are setup to only pass expected data and all unexpected data does not pass. The other method uses "bad lists" which will specifically filter based on listed "bad" characters. This is not the preferred method of input validation, because new methods will not be detected.	Problem	Non-Problem
19.	When validation of user input fails, the site must return a 404 error. The default 404.htm is preferred.	Problem	Non-Problem
20.	In all web servers, a web user account is defined. Windows for example uses the *IUSR* account. Whatever the account name may be, that account should only have READ rights throughout the directories that are part of the website. Occasionally, the account may require execute or scripts rights as well. This account should have explicit deny rights throughout the remainder of the file systems on the server or as close as possible while maintaining functionality.	Problem	Non-Problem
21.	All web forms must be protected against multiple	Problem	Non-Problem

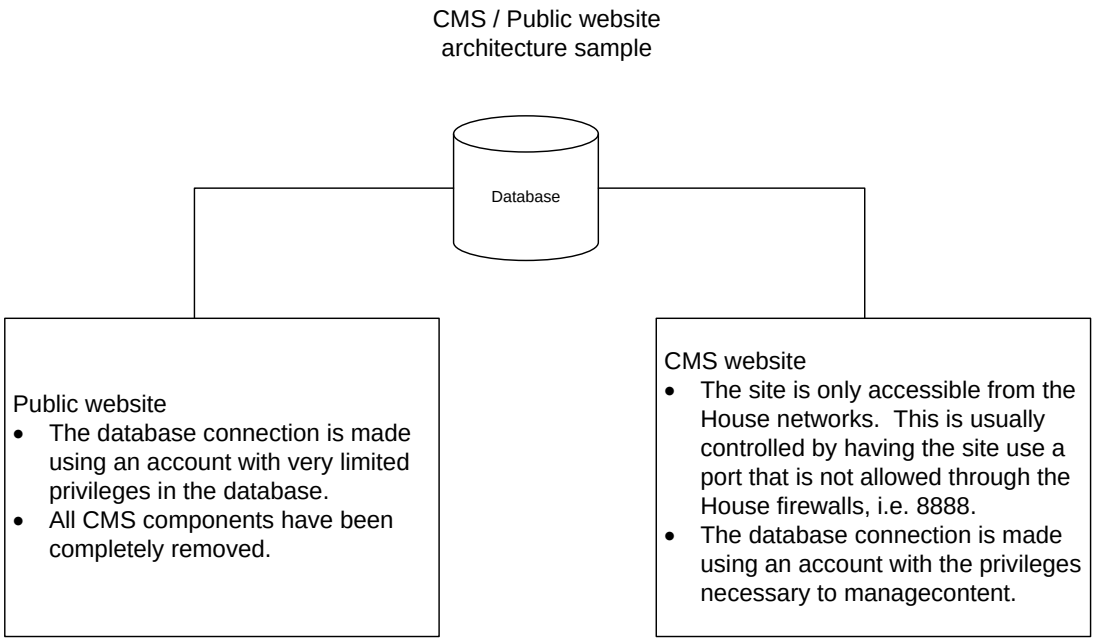
	submissions. Multiple submissions of web forms can often create denial of service conditions on the web server and affect servers in the enterprise.		
22.	All web forms must include direction for public users to not pass any sensitive data via the web form.	Problem	Non-Problem
23.	Web forms to email a link to a friend need to be secured so that they can't be used to relay or spoof email.	Problem	Non-Problem
24.	Whenever a file or directory that doesn't exist is requested, the site must return a 404 error. The default 404.htm is preferred.		
25.	Filters should be implemented at the web server layer to intercept malicious URLs and return a 404 error for any URL that doesn't pass the filter. For file extension requests, the filter should block the following extensions and return a 404 error: *.exe *.com *.dll *.conf *.log *.htr *.cer *.cdx *.bat *.cmd *.mdb *.php *.asp *.aspx *.zip *.rar *.cfg *.dbf *.udl *.old *.bak The following characters should also be filtered and a 404 error should be returned when encountered: .. ./ \ : % & # < > $ @ ! , ~ ‘ ; passwd _vti backup root bak bkup test	Problem	Non-Problem

	temp etc odbc w3svc _derived netcat .c password admin nobody		
26.	All new public web sites for Members, Committees, or Leadership must be hosted on a server managed by HIR, or by an authorized vendor with a server located in the House data center.	Problem	Non-Problem
27.	All public Web sites must be hosted on servers that physically reside on internet address space owned and controlled by the House.	Problem	Non-Problem
28.	The "house.gov" domain name must only be used in reference to hosts on internet address space owned and controlled by the House.	Problem	Non-Problem
29.	Only "house.gov" domain names can be used to reference hosts on internet address space owned and controlled by the House.	Problem	Non-Problem
30.	All forms based submissions must be encrypted with SSL to protect potentially sensitive information. The ISSO can provide a certificate if needed.	Problem	Non-Problem
31.	Website shall not permit content submitted by public users to be immediately published on the site. Any content submitted by public users must be examined, authorized, and published by an authorized website administrator.	Problem	Non-Problem
32.	CMS must provide an audit trail that clearly indicates who made site changes. This trail shall include the following information: user name, date, time, data that was added, deleted, or changed, and source IP address of user. Only authorized website administrators may modify House websites.	Problem	Non-Problem

Table 1: Glossary

Abbreviation	Name
CMS	Content Management System
ISSO	Information Systems Security Office
SSL	Secure Sockets Layer
HIR	House Information Resources
OS	Operating System

Diagram 1: Architecture



Change Log:

- Changes 8/2009
  - Items 6&8, items 17-19
  - Added item 7
  - Added items 31&32
  - Added diagram 1
  - Added glossary