Comp116: Technical Risk Analysis for CTF-2015

Mengtian Li

11/19/2015

Risk ID	Technical Risk	Techical Risk Indicators	Related IDs	Impact Rating	Impact	Mitigation	Validation Steps
1	SQL Injection	In board.php(line 55 and line 61), dblib.php(line 23) and index.php (line 38 and line 49)in score board	CWE-89	H	The function call constructs a dynamic SQL query using a variable derived from user-supplied input. An attacker could exploit this flaw to execute arbitrary SQL queries against the database.	Use parameterized prepared statement for user input, so that the raw user input may not be in the query.	Use Sqlmap to try different attacks to the database, make sure nothing is broken.
2	Sensitive Cookie Without 'Secure' Attribute	In main.php, line 9. Cookie that leads to sensitive information is sent in plain text	CWE-614	M	The Secure attribute for sensitive cookies is not set, which could cause the user agent to send those cookies in plaintext over an HTTP session.	Always set the secure attribute to cookies that lead to sensitive information	Use Cookie Manager to check if any sensitive cookie is sent in plaintext.

3	Credentials Management	In dblib.php (line 3), index.php (line 31, 114) in score board. Password is hard coded.	CWE-259	M	Hard coding password allows all of the project's developers to view the password. It also makes fixing the problem extremely difficult, once the code is in production, the password cannot be changed without patching the software.	Store password in different location such as conf file.	Check project files and make sure no plaintext password is in source file except conf file
4	XSS	In board.php (line 43, 44, 50, 58, 59, 64) and index.php (line 119) in score board. Populate the HTTP response with user input directly.	CWE-3	M	Allows an attacker to embed malicious content, such as Javascript code, which will be executed in the context of the victim's browser.	NEVER TRUST USER INPUT! Use different contextual escaping methods on all untrusted data before using it to construct any portion of an HTTP response.	Use different scripts in all user input fields. If nothing goes wrong than the website is immune to XSS.

5	Information Leakages	In board.php (line 18), dblib.php (line 8, 27) and index.php(line 34, 114) in score board, sensitive information is exposed through error message	CWE-209	L	The software generates an error message that includes sensitive information about its environment, users, or associated data. Attackers can use the error information to focus on their next attack.	Ensure that only generic error messages are returned to the end user that do not reveal any additional details.	Check all the error messages that are sent back to see if they contains valuable information
6	Social Engineering	Flag.txt	N/A	M	Irrelevant information exposure can lead attacker to social engineer towards victims	Exclude irrelevant information in directory	Without irrelevant information exposed to users, attackers have no information to use for social engineering.
7	Separation of Concerns	The website does not sufficiently require different privilege levels, rights, or permissions.	CWE-653	L	Break into weakness in low privileged users and lead to leak in high privileged users.	Set boundaries between modules. Use different modules	Since boundary exists, a leak in one module will not cause leak in other modules