Nick Teleky

11/12/13

Comp116: Computer Security

**Technical Risk Analysis of Capture The Flag**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Technical Risk | Indicators | Impact Rating | Impact | Mitigation | Validation Steps |
| 1 | Code injection (PHP) | The eval() function is used in code; malicious code appears to be running on the site. | High | Allows an attacker to execute arbitrary code. | Refactor code so that eval() is unnecessary; if that is not possible, then validate ALL input so that code could not be properly sent to the eval() function. | Try to input malicious code that returns either alerts or other output. If it is possible, try to mitigate. |
| 2 | SQL Injection | Large number of logon attempts; SQL queries with strings like:” ‘ OR ‘1’=’1” common in logs; improper logons; the database is improperly altered or dropped. | High | Allows an attacker to gain access to the application; allows for an attacker to manipulate database queries and do things like drop the entire table. | Validate all user input and/or use parameterized prepared statements when communicating with the SQL server as opposed to dynamically creating SQL queries. | Attempt to gain improper access to the application using SQL injections or get the database to return information like usernames and passwords. Verify that this cannot be done. |
| 3 | Use of hard-coded password | The account or application has been compromised with the admin or root password. | Med. | Allows an attacker to compromise an account by finding the hard-coded password. | Don’t store passwords in the application code; use best practices when storing and utilizing passwords. | Verify that the password cannot be found by viewing source code or source files in the application. |
| 4 | Cross-site Scripting (XSS) | There are javascript alerts or console activities running that shouldn’t be; parts of the html of the application have changed in a way not intended. | Med. | Allows an attacker to embed malicious content, steal cookies, or find information about a user. | All user input should be validated and sanitized using a method called contextual escaping. | Attempt to inject javascript instead of user input to attack the website. |
| 5 | Information exposure through error messages | An error message containing information about environment or users. | Low | Allows an attacker to gain key information about how an application is set up, possibly making it easier to launch other, more serious attacks. | Edit code to ensure that only generic error messages are produced. | View all error messages on the application and confirm that they are not exposing extra information. |