Version	Author	Date	Reviewed By
2.0	Rohit Kumar Sharma	06-July-2016	Mr. Anil Kumar Jha

Introduction

Application Security Division conducts web application security testing on web sites, which are about to be hosted or already hosted on NIC web servers. The Purpose of this document is to assist the Developers/site owners of web applications on sites to ensure that the necessary steps have been taken in the development of applications which are secure and which cannot be exploited, before submitting the site for a security audit.

By answering the checklist below and examination of the answers you will be able to point out the areas in your application necessitating change.

Note:

1. Refer to the Secure Programming Guidelines (ver-2.0) to determine the OWASP vulnerability category your web site/application may be prone to.
2. The OWASP Top 10 labels the top 10 vulnerability as A1 to A10.
3. Application developed at vendor side and source code is not available, following recommendation will not applied.

The table below list is the checklist as per these vulnerabilities put ✔ against the entry below the column Y, N.

Y=Yes N=NO

	Check List for Developers/Site Owners	**Y**	**N**	**Comments**
A1	**SQL Injection**			
1	Does the application interface to a Database?			
2	Type of DB – MS Access / My SQL / DB2 / Oracle			
3	Does the code include SQL statement			
4	Is the SQL statement being formulated during run time using concatenation operators (Ex: +, & etc.)?			
5	Is parameterized query used in the SQL statement?			
6	Are the table level privileges applied to database user used by the application to connect to DB?			
8	Are the permissions on table is set and reviewed?			
9	Validations: a. For string input: (any of the following method may be used)			
	i. Regular expression: Does the code check the input string against a valid			

	regular expression (Ex: for alphanumeric "^[a-zA-Z0-9]*$")			
	ii. Whitelisting: Does the code check the input string against a valid allowed character set. (Ex: a-z, A-Z, 0-9)			
	b. Is check for field input of numeric type done?			
	c. Is the validation is done on client and server side.			
A2	**Cross Site Scripting**			
1	Does your application use data stored in database for rendering in HTML page for output?			
2	Have you applied encoding for data fetched from database and used during rendering of page?			
3	Have you checked the code to identify all places where input from an HTTP request is directly used in output?			
4	Is input validation done on values received from form fields, query string, hidden fields?			
5	Validations: a. For string input: (any of the following method may be used)			
	i. Regular expression: Does the code check the input string against a valid regular expression (Ex: for alphanumeric "^[a-zA-Z0-9]*$")			
	ii. Whitelisting: Does the code check the input string against a valid allowed character set. (Ex: a-z, A-Z, 0-9)			
	b. Is check for field input of numeric type done?			
	c. Is the validation is done on client and server side.			
6	Have you checked the underlying database cleared of any script that may be stored already?			
A3	**Broken authentication and session management**			
1	Does the application use session to keep track of the streams of request from each user?			
2	Does the web hosting server environment provide a session capability?			
3	Does the application development environments support developers to create their own session token?			

4	Does the application provide the user with logout option?			
5	Is the session created, initialize, managed and destroyed properly by the application?			
	a. Is the session still active after it is left unattended for than specified minimum amount of time (session timeout)			
	b. Is it possible to invoke a deeper (restricted) page after a user has logged out from a previously logged in session			
6	Are the user credentials travelling in clear text while trying to login? Also check during change password.			
7	Is a simple hash of the password generated on the client prior to send to the server in authentication module?			
8	Is the entire login transaction encrypted using something like SSL/VPN?			
9	Are the session IDs included in the URL?			
10	Is the session ID represented by a long, complicated, random number that can't be easily guessed?			
11	Is any session ID chosen by a user and submitted to the server and accepted by the application?			
12	Does the session ID change in following scenario:			
	a. Is the session ID same for a user before and after he logs (authenticate) in to the application?			
	b. Is the session ID exposed when switching from (non-ssl) HTTP to HTTPS (ssl) section of the site/application?			
	c. During any major transition			
13	Password strength-			
	a. Does the application enforce strength of password			
	i. Include a minimum size			
	ii. Complexity, use of minimum combination of alphabetic, numeric, and/or non-alphanumeric character			
	a. Are user required to change their password periodically (including after first login)?			
	b. Are users allowed for using previous password?			

14	Password use:			
	i. Is the user allowed to make indefinite login attempt?			
	ii. Are repeated failed login attempts logged?			
	iii. Is account lockout policy applied if an incorrect password is entered a specified number of times.			
15	Password change control			
	a. Does the password change module requires to provide both their old and new password when changing their password?			
	b. Does the application mandate a re-authentication before giving access to any critical functionality?			
16	Reset or Forgot password			
	a. Is password reset link sent to user mail id?			
	b. Does link expire in 24 hours?			
17	Password storage			
	a. Are the password stored in either hashed or encrypted form?			
	b. Are passwords hardcoded in any source code including the client and server code?			
	c. Encryption should be used when plain password is needed, such as when using the password to login to another system. If encryption passwords are used, are the decryption keys are strongly protected?			
18	Browser Caching			
	a. Are authentication (used id and password) and session id data submitted as part of GET method?			
	b. Is it possible for a browser to remember the credentials through autocomplete on?			
	c. Is it possible for someone to use back button in a user's browser to go back up to the login page and re-submit the previously typed credentials?			
19	Has logout module implemented session clear, remove attribute and abandon procedure?			
A4	**Insecure Direct Object References**			
1	Does application has multiple roles?			

2	Is authorization done for requested object (file, directory, database key)?			
A5	**Security Misconfiguration**			
1	Is directory listing disabled for all directory in the application?			
2	Is exception handling done?			
3	Does application has customized error page?			
4	Is user redirected to appropriate error page on any exception or error in application?			
5	Does application contain any unused page?			
A6	**Sensitive Data Exposure**			
1	Is system generated information displayed on browser?			
	If Yes, do these information reveal server environment like ODBC, Vb script etc.			
2	Does application have login module?			
3	Is salted SHA256 hashing technique implemented for password transmission from login page?			
4	Is simple SHA256 hashing technique used for new user creation and password change/reset page?			
5	Is SSL/TSL used to secure data in transit?			
A7	**Missing Function Level Access Control**			
1	Does application have multiple role?			
2	Is deny access by default done?			
3	Does application use principle of least privileges?			
4	Is role based access control matrix implemented?			
5	Is authorization done in business logic?			
6	Does your application rely on security by obscurity by just hiding buttons and links to functionality within web pages?			
A8	**Cross-Site Request Forgery**			
1	Is cookies values shared across different tabs of the same browser?			
2	Has developer inserted custom random tokens into every form and URL			
3	Has developer checked the existence of these random tokens?			
4	In case of random value is not what is expected, is request rejected as invalid?			

4	Has Cross Site Scripting vulnerability been fixed as mentioned in A2			
5	Is protection used for			
	a. OWASP CSRF Guard			
	b. PHP Guard			
	c. .NET Guard			
A9	**Using Components with known vulnerabilities**			
1	Has developer identified external components and security issues associated with?			
2	Has developer performed an extensive research on any possible vulnerabilities linked to those components?			
3	Is suggested resolution for vulnerabilities successfully implemented?			
4	IS unused features and functionalities disabled?			
A10	**Open Redirects**			
1	Does application contain redirects and forwards?			
2	If used, does it involve user parameters in calculating the destination?			
3	If parameters can't be avoided, is validation done on supplied data and authorized for user?			
4	Is destination parameter mapped to certain websites only?			
Malicious File Upload				
1	Is filename and its extension validated on uploaded document?			
2	Is there a list of extension of file allowed for upload?			
3	Is there a subroutine to generate a file name for storing it in database?			
5	Is there a separate directory for file upload?			
5	Is execute permission is disabled for upload directory?			
6	Is POST method used for file upload?			
7	Is content type check done?			
8	Is maximum and minimum file size is defined?			

For additional information:

References:

1. http://www.owasp.org
2. OWSP Top 10 Document
3. https://security.nic.in/docs/SecureProgramming.pdf