|  |  |  |  |
| --- | --- | --- | --- |
| Category | Description | Pass | Fail |
| General | Are there any backdoors/unexposed business logic classes? |  |  |
| Business Logic and Design | Are there any unused configurations related to the business logic? |  |  |
| Business Logic and Design | If request parameters are used to identify business logic methods, is there a proper mapping of user privileges and methods/actions allowed to them? |  |  |
| Business Logic and Design | Check if unexposed instance variables are present in form objects that get bound to user inputs. If present, check if they have default values |  |  |
| Business Logic and Design | Check if unexposed instance variables present in form objects that get bound to user inputs. If present, check if they get initialized before form binding. |  |  |
| Access Control | Is the placement of authentication and authorization check correct? |  |  |
| Access Control | Is there execution stopped/terminated after for invalid request? I.e. when authentication/authorization check fails? |  |  |
| Access Control | Are the checks correct implemented? Is there any backdoor parameter? |  |  |
| Access Control | Is the check applied on all the required files and folder within web root directory? |  |  |
| Access Control | Are security checks placed before processing inputs? |  |  |
| Business Logic and Design | Check if unexposed instance variables are present in form objects that get bound to user inputs. If present, check if they have default values. |  |  |
| Business Logic and Design | Check if unexposed instance variables present in form objects that get bound to user inputs. If present, check if they get initialized before form binding. |  |  |
| Business Logic and Design | Are the checks correct implemented? Is there any backdoor parameter? |  |  |
| Business Logic and Design | Is the check applied on all the required files and folder within web root directory? |  |  |
| Business Logic and Design | Is there any default configuration like Access- ALL? |  |  |
| Business Logic and Design | Does the configuration get applied to all files and users? |  |  |
| Authorization | In case of a container-managed authentication - Is the authentication based on web methods only? |  |  |
| Access Control | In case of container-managed authentication - Does the authentication get applied on all resources? |  |  |
| Session Management | Does the design handle sessions securely? |  |  |
| Access Control | Is Password Complexity Check enforced on the password? |  |  |
| Cryptography | Is password stored in an encrypted format? |  |  |
| Access Control | Is password disclosed to user/written to a file/logs/console? |  |  |
| Cryptography | Are database credentials stored in an encrypted format |  |  |
| Business Logic and Design | Does the design support weak data stores like fat files |  |  |
| Business Logic and Design | Does the centralized validation get applied to all requests and all the inputs? |  |  |
| Business Logic and Design | Does the centralized validation check block all the special characters? |  |  |
| Business Logic and Design | Are there any special kinds of requests skipped from validation? |  |  |
| Business Logic and Design | Does the design maintain any exclusion list for parameters or features from being validated? |  |  |
| Input Validation | Are all the untrusted inputs validated? |  |  |
| Cryptography | Is the data sent on encrypted channel? |  |  |
| Session Management | Does the design involve session sharing between components/modules? Is session validated correctly on both ends? |  |  |
| Business Logic and Design | Does the design use any elevated OS/system privileges for external connections/commands? |  |  |
| Business Logic and Design | Is there any known flaw(s) in API’s/Technology used? |  |  |
| Business Logic and Design | Does the design framework provide any inbuilt security control? |  |  |
| Business Logic and Design | Are privileges reduce whenever possible? |  |  |
| Business Logic and Design | Is the program designed to fail gracefully? |  |  |
| Logging and Auditing | Are logs logging personal information, passwords or other sensitive information? |  |  |
| Logging and Auditing | Do audit logs log connection attempts (both successful and failures)? |  |  |
| Logging and Auditing | Is there a process(s) in place to read audit logs for unintended/malicious behaviours? |  |  |
| Cryptography | Are sensitive information sent over the network encrypted form. |  |  |
| Access Control | Does application design call for server authentication (anti-spoofing measure)? |  |  |
| Access Control | Does application support password expiration? |  |  |
| Cryptography | Does application use custom schemes for hashing and or cryptographic? |  |  |
| Cryptography | Are cryptographic functions used by the application the most recent version of these protocols, patched and process in place to keep them updated? |  |  |
| General | Are external libraries, tools, plugins used by the application functions the most recent version of these protocols, patched and process in place to keep them updated? |  |  |
| General | Classes that contain security secrets (like passwords) are only accessible through protected API’s |  |  |
| Cryptography | Does are there any special kind of request skipped from validation? |  |  |
| Cryptography | Keys are not held in code. |  |  |
| General | Plain text secrets are not stored in memory for extended periods of time. |  |  |
| General | Array bounds are checked. |  |  |
| User Management | User and role based privileges are documented |  |  |
| General | All sensitive information used by application has been identified |  |  |
| User Management | Authentication cookies are not persisted |  |  |
| User Management | Authentication cookies are encrypted |  |  |
| User Management | Authentication credentials are not passed by HTTP GET |  |  |
| User Management | Authorization checks are granular (page and directory level) |  |  |
| User Management | Authorization based on clearly defined roles |  |  |
| User Management | Authorization works properly and cannot be circumvented by parameter manipulation |  |  |
| User Management | Authorization cannot be bypassed by cookie manipulation |  |  |
| Session Management | Session cookies expire in a reasonable short time |  |  |
| Session Management | Session cookies are encrypted |  |  |
| Session Management | Session data is validated |  |  |
| Session Management | Session id is complex |  |  |
| Session Management | Session storage is secure |  |  |
| Session Management | Session inactivity timeouts are enforced |  |  |
| Data Management | Data is validated on server side |  |  |
| Data Management | HTTP headers are validated for each request |  |  |
| Business Logic and Design | Are all of the entry points and trust boundaries identified by the design and are in risk analysis report? |  |  |
| Data Management | Is all XML input data validated against an agreed schema? |  |  |
| Data Management | Is output that contains untrusted data supplied input have the correct type of encoding (URL encoding, HTML encoding)? |  |  |
| Data Management | Has the correct encoding been applied to all data being output by the application |  |  |
| Web Services | Web service has documentation protocol is disable if the application does not need dynamic generation of WSDL. |  |  |
| Web Services | Web service endpoints address in Web Services Description Language (WSDL) is checked for validity |  |  |
| Web Services | Web service protocols that are unnecessary are disabled (HTTP GET and HTTP POST |  |  |
| Access Control | All internal and external connections (user and entity) go through an appropriate and adequate form of authentication |  |  |
| Access Control | All pages enforce the requirement for authentication |  |  |
| Business Logic and Design | All methods/functions call that return a value have a proper error handling |  |  |
| Business Logic and Design | Exceptions and error conditions are properly handled |  |  |
| Business Logic and Design | Application fails in a secure manner |  |  |
| Business Logic and Design | Components that should not be directly accessible to the users cannot be reached |  |  |
| Business Logic and Design | Examine the application for “magic and main()” executable functions as well as debug harnesses/backdoors |  |  |
| Business Logic and Design | Search for commented out code, commented out test code, which may contain sensitive information. |  |  |
| Business Logic and Design | All logical decisions have a default clause |  |  |
| Business Logic and Design | No development environment kit is contained on the build directories |  |  |
| Business Logic and Environment | Search for any calls to the underlying operating system or file open calls and examine the error possibilities |  |  |

Reference: OWASP Application Security Veriﬁcation Standard Project,” http://www.owasp.org