|  |  |  |
| --- | --- | --- |
| Area | Pass | Fail |
| Does the architecture identify, understand and accommodate the company’s security policy ? |  |  |
| Are restrictions imposed by infra-structure security (including available services, protocols, and ﬁrewall restrictions) identiﬁed ? |  |  |
| Does architecture recognize and accommodate restrictions imposed by hosting environments ? (including application isolation requirements) |  |  |
| Is target’s environment code-access security trust level known? |  |  |
| Does architecture identify the deployment infrastructure requirements and the deployment conﬁguration of the application ? |  |  |
| Are domain structures, remote application servers, and data-base servers identiﬁed ? |  |  |
| Does the architecture identify clustering requirements? |  |  |
| Does the architecture identify application conﬁguration maintenance points ?(such as what needs to be conﬁgured and what tools are available for an IDC admin). |  |  |
| Are secure communication features provided by the platform and the application known? |  |  |
| Does the architecture address web farm considerations ? (including session state management, machine-speciﬁc encryption keys, Secure Sockets Layer [SSL], certiﬁcate deployment issues, and roaming proﬁles) |  |  |
| Does the architecture identify the certiﬁcate authority (CA) to be used by the site to support TLS ? |  |  |
| Does the architecture address the required scalability and performance criteria ? |  |  |
| Are all entry points and trust boundaries identiﬁed by the design? |  |  |
| Is input validation applied whenever data are received from the use? |  |  |
| Does the architecture assume that user’s input is malicious? |  |  |
| Is centralized input validation used ? (where appropriate) |  |  |
| Is the input validation strategy adopted by the application modular and consistent ? |  |  |
| Does the architecture address potential canonicalization issues ? |  |  |
| Are input ﬁle names and paths avoided wherever possible? |  |  |
| Does the architecture address potential SQL injection issues ? |  |  |
| Does the architecture address potential cross-site scripting issues ? |  |  |
| Does the architecture rely on the client side validation ? |  |  |
| Does the application apply defence in depth by providing input validation across tiers? |  |  |
| Is output containing input encoded ? |  |  |
| Does the architecture identify application trust boundaries ? |  |  |
| Does the architecture identify identities that are used to access resources across the trust boundaries ? |  |  |
| Does the architecture partition a web site into public and restricted areas by using separate folders? |  |  |
| Does the architecture identify service account requirements? |  |  |
| Does the architecture identify a secure storage of credentials that are accepted from users? |  |  |
| Does the architecture identify mechanisms to protect credentials over the wire? |  |  |
| Are account management policies taken into consideration by the architecture? |  |  |
| Does the architecture ensure that minimum error information is returned in the event of authentication failure? |  |  |
| If SQL authentication is used - Are credentials adequately secured over the wire ? |  |  |
| Does the architecture adopt a policy of the least-privileges? |  |  |
| Does architecture ensure that authentication tickets (cookies) are not transmitted over the non-encrypted connections? |  |  |
| Are multiple gatekeepers used for defence in depth ? |  |  |
| Is application’s login restricted in the database to access-speciﬁc stored procedures ? |  |  |
| Does the application’s login have permissions to access tables directly? |  |  |
| Is access to system level resources restricted? |  |  |
| Does the architecture identify code access security requirements? |  |  |
| Are all identities within the application identiﬁed and ? |  |  |
| Are remote administration channels secured? |  |  |
| Are conﬁguration secrets held in a plain text ﬁles ? |  |  |
| Are administrator privileges separated based upon roles? |  |  |
| Are database connections, pass-words, keys, or other secrets stored in the encrypted form? |  |  |
| Does the architecture identify the methodology to store secrets securely? |  |  |
| Are sensitive data logged by the application in a clear text form? |  |  |
| Are sensitive data stored in persistent cookies? |  |  |
| Are sensitive data transmitted with the GET HTTP method? |  |  |
| IS TLS used to protect the authentication cookies? |  |  |
| Is the content of authentication cookies encrypted? |  |  |
| Is session lifetime limited? |  |  |
| Is session state protected from unauthorized access? |  |  |
| Are session identiﬁers passed in query strings? |  |  |
| Does architecture ensure that a platform-level cryptography is used and it has no custom implementations? |  |  |
| Is the methodology to secure encryption keys identiﬁed? |  |  |
| Does the architecture identify the key recycle policy for the application? |  |  |
| Is DPAPI used wherever possible to avoid key management issues ? |  |  |
| Are keys periodically recycled? |  |  |
| Is “View state” protected using MACs? |  |  |
| Does the architecture outline a standardized approach to exception handling across the application? |  |  |
| Does the architecture identiﬁes generic error messages that are returned to the client? |  |  |
| Does the architecture identify a level of auditing and logging necessary for the application and identiﬁes the key parameters to be logged and audited? |  |  |

Source: Microsoft.Developers Network (MSDN) Patterns and Practices <https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff647464(v=pandp.10)?redirectedfrom=MSDN>