Things to Build Secure Website

1. Encrypt Data

Data security is the most important aspect of Web security. Most of the data stored in databases are plain and open. While most of the data can be stored plain, sensitive data must be encrypted in the database. Some of the common data that must be encrypted include user IDs, emails, passwords, social security numbers, date of birth records, credit card details, password hint answers, personal health records, private chats and messages, financial records, and banking information.Some encryption techniques such as MD5, SHA1 can be used.

1. Encrypt Website

Securing a website using HTTPS.Several ways to do this.Most common way is to host website on IIS via https protocol.

1. Stop SQL Injections

SQL Injection is responsible for cyber-attacks and hacking. SQL Injection is a technique hackers use to exploit SQL queries and URLs used in web applications.

Procedures and functions that use dynamic SQL queries by concatenating the text inputs to the dynamic SQL are prone to SQL Injection attack as someone can provide extra commands/malicious text through the input parameter and when executed can result in the unexpected results.

Best Practices to prevent SQL Injection is:

Execute Dynamic SQL queries, using SP\_EXECUTESQL procedure with the parameters.

4. Remove Embedded SQL

Using embedded SQL queries in your code may lead to an easier path for hackers. If possible, use stored procedures or encrypted queries to make it more difficult for hackers.

5. Secure Credentials

Developers often store database server credentials in configuration files. No matter what, all database servers and other server connections and settings must be encrypted. Try to avoid hardcoding server credentials. If you must hardcode credentials in your code, make sure they are encrypted

6. Enforce Complex Passwords

Simple passwords are one of the reasons most hackers get into a system.The complexity of passwords, also known as password strength, is a measure of the effectiveness against attackers.Make sure password will meet secure password policy.

Hashing is the best option to secure and save passwords. Hashing enforces no one can read a password

7. Secure APIs

APIs are a common data exchange mechanism between applications. Developers must ensure that all APIs are secure and use SSL and other best practices. The connection credentials and other sensitive data must be properly encrypted.

9. Implement Exceptions and Error Handling

Proper exception and error handling can safe site from hacking.This means not to show the exact error message thrown by code to user because that may contain some useful information and show him a valid error message

10.Trusted Url:

If from a website we are redirected to certain url then make sure that url is trusted.

This can be implemented using sce technique (Strict Contextual Escaping)

11. Preventing Cross-Site Scripting (XSS) Attack-

XSS attack exploits vulnerabilities in Web page validation by injecting client-side script code.

Scenario #1

Using a cookie a hacker can hack a victim’s username and password and save their credentials in his/her database.

Scenario #2

Using a prompt box, asking for a password verification and once the end user enters the password then the entered password will be saved in hacker's database.

Prevention mechanism

Step 1: Check that ASP.NET request validation is enabled.

Step 2: Verify ASP.NET code that generates HTML output.

Step 3: Find out whether HTML output includes input parameters.

12. Secure Database Server

Here is a list of tasks database administrators must do to secure database servers.

Make sure database server is separate from a Web server

Secure and encrypt login credentials

Implement separate user logins for separate web applications

Don’t give database users write and delete permissions unless necessary

Use object permissions on database tables and objects

Use a secure mechanism to provide data access

Store and monitor database logs

13. Secure Sessions

If sessions are used in applications then make sure they are secured and should be closed in given time frame and if user is in ideal state during that time of frame user should be logged out from application.