How To Build Secure Websites

Mahesh Chand Updated date May 27, 2021

81.2k 33 97

Building a secure website includes not only IT security but also writing secure code and data authentication and integrity. This article lists the top 10 tips for building secure Web applications.

Build Secure Website

Cybercrime rates have doubled in the past 12 months, and cybercriminals are attacking online websites left and right. Recent data breaches have exposed millions of Americans personal data, including their social securities, date of births, credit card numbers, bank accounts and addresses. Equifax, one of the largest credit rating companies in the USA, has recently experienced a data breach that has put millions of Americans in financial danger.

The following table lists some of the largest data breaches of all time:

Company	Accounts hacked	Date hacked
Yahoo	3 billion	Aug 2013
Marriott	500 million	Nov 2018
Yahoo	500 million	2014
Adult FriendFinder	412 million	Oct 2016
MySpace	360 million	May 2016
Under Armor	150 million	Feb 2018
Equifax	145 million	July 2017
eBay	145 million	May 2014
Target	110 million	Nov 2013
Quora	100 million	Nov 2018
LinkedIn	100 million	June 2012
JP Morgan Chase	83 million	July 2014

Uber	57 million	Oct 2016
Facebook	50 million	July 2017

Today, building secure websites is one of the major concerns for any CIO or CTO. Network administrators, IT managers, software architects, and web developers are responsible for building and maintaining secure websites. In this article, I will cover some basic tips that will help you build secure websites.

The tips in this article are a list of checkpoints we should follow when building new websites.

We can divide website security into two categories – developing and maintaining. Developing secure websites fall into the lap of developers and architects whereas maintaining secure websites is the responsibility of server administrators.

Developing secure websites

Building a secure web application is the first step towards building a secure website. If a web application has security holes, it is more open to attack by hackers. As a matter of fact, SQL Injection is responsible for 62% of cyber attacks and hacking. You can read a full report in my article, Top 10 Web Application Security Risks.

Before a web application can be deployed to a Web server and exposed to the outside world, it must be developed securely. The group of people who are responsible for building secure web applications is architects, database administrators, and developers. Testers are also involved in the process.

Here is a list of some simple tips for Web architects and developers to build secure web applications.

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. The cost of storing encrypted data isn’t much. As a matter of fact, these days, most of the new versions of database systems come with built-in encryption options. For example, SQL Server 2017 lets you encrypt the entire database using simple SQL commands. Check out Always Encrypt In SQL Server 2016.

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.

On top of this, you could apply double encryption on the most sensitive data such as passwords, credit card information, social security, and anything else you think is valuable. Hashing is recommended for password and other sensitive data encryption.

This one simple step is a part of application architecture and database design that does not require a ton of overhead.

2. Encrypt Websites

Securing a website using HTTPS is a must today. Check out how to set up HTTPS on your website and How To Enable HTTPS on Your Website For Free.

3. Stop SQL Injections

SQL Injection is responsible for 62% of cyber-attacks and hacking. SQL Injection is a technique hackers use to exploit SQL queries and URLs used in web applications. Here are two good articles, Protect Your Data – Prevent SQL Injection and Best Practices to Prevent SQL Injection. To learn more and how to write code to avoid SQL Injection, here is a list of more Articles on SQL Injection.

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. If you must use SQL queries, under no circumstances should SQL queries be a part of your presentation layer code (HTML, ASP.NET, JavaScript, etc.). They must be moved to your server-side code. If you're just building a UI layer, the data transfer should be performed via secure APIs.

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 and the private/public key is stored securely somewhere.

Database systems may also have a mechanism to secure database connections. For example, SQL Server and Azure SQL allow secure database connections. See Enable Encrypted Connections To Database Engine.

6. Enforce Complex Passwords

Simple passwords are one of the reasons most hackers get into a system. According to the Verizon Data Breach Investigations Report (DBIR), 63% of confirmed data breaches are due to weak or stolen passwords.

The complexity of passwords, also known as password strength, is a measure of the effectiveness against attackers. Here are some of the key points developers can enforce to create complex passwords.

- Have a minimum length of passwords of at least 8 characters
- At least one upper case, one lower case, one number, and one special character
- Don't allow names and user IDs as a part of a password
- Don't allow old passwords to be repeated
- Enforce password change (for some systems) frequently (for example, every 60 days)
- Password reset should have security questions and/or email and phone number pin verifications

Hashing is the best option to secure and save passwords. Hashing enforces no one can read a password. The only way to change the password is to reset the password with the help of security questions and other hints. Also do not send plain passwords in emails.

7. Implement Industry Standard Authentication and Authorization

Broken Authentication is the number-two cause of Web application security risks according to OWASP Web Application Top 10 Security Risks. By implementing recommended best practices, developers can avoid major security risks in their applications. Applications that implement incorrect authentication and session variables lead hackers to hijack passwords, keys, session tokens, and other credentials stored in sessions. Cookies are another method that can be used to exploit application security. Here is a good article: OWASP Authentication Cheat Sheet.

8. 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 may not fix the application security but can lead to troubleshooting the problem that can be patched. Developers must make the habit of implementing exception and error handling part of their coding practices.

10. Implement Logs and Analytics

Logs and analytics do not fall in the security bucket but can lead to finding and fixing the hole. Tracking and logging in activities such as user login, location, browser and so on can help track suspected users of a website. Developers should make a habit of implementing analytics such as Google Analytics for public websites that keep track of almost every activity of the website's visitors.

Keep websites secure and out of reach of hackers

Once a website is developed and deployed, it is up to network administrators and IT managers to secure the website and keep it secure from the attackers. Here are some of the key items to consider.

1. Keep Web Server Secure

A Web Server is one of the most important and critical components of web infrastructure. The Web server is responsible for hosting a Web site and its related code, services, and all required files.

Here is a list of tasks that Web server administrators should perform to keep Web and Database servers secure.

- Separate development, staging, and production environments
- Keep the Operating System on its own hard drive partition
- Enable tight security on the Web Server including permissions and access
- Keep separate user logins and their permissions based on their roles
- Remove unnecessary services and don't install them during installations
- Disable remote access. If you must provide remote access, it should be on a secure network
- Keep the web application, scripts, and all code on a separate partition of the hard drive
- Install a firewall and necessary products
- Websites should be secure using the latest version of SSL and other protocols
- Close all default open ports
- Make sure to change and separate Admin logins and passwords from Web application administrators
- Configure and enable Web server and other logs
- Provision web server for the latest technologies such as containers
- Make sure to allocate and separate proper resources for web applications and services
- Avoid using shared servers among multiple clients
- Do not enable write permissions on the server's file system

2. 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

3. Security Patches and Updates

Keep your servers up to date with the current patches including OS patches, database upgrades, and other software upgrades.

4. Monitor Traffic

Implement a proper mechanism to monitor server traffic and implement a fraud protection mechanism for suspected traffic.

5. Monitor Application Logs and Exceptions

Web applications must implement the recording of recommended logs and exceptions. Server administrators should work with application managers to monitor application logs and exceptions frequently.

6. Audit Server Logs

Monitor server logs frequently and analyzes them with the team. Server logs provide details on the traffic, exceptions, and warnings.

7. Educate Users

Server administrators must educate Web administrators, developers, and even management about the importance of security and discourage them to download and make frequent changes. All changes on the servers must be logged, reviewed, and approved.

Further readings

Are you building web applications? Here is a recommended article: Top 10 Tips To Build High-Performance Websites

References

https://www.acunetix.com/websitesecurity/webserver-security/
http://www.applicure.com/blog/database-security-best-practice
http://www.c-sharpcorner.com/article/top-10-web-application-security-risks-in-2017/

Cyber security
Secure website

Mahesh Chand Admin

Founder C# Corner. Founder & CEO Mindcracker Inc. Investor, Advisor, Board member of several startups and non profit foundations. Try to implement emerging technologies when trying to solve the next problem.

https://www.c-sharpcorner.com

201.2m 1 MVP 14

97 33 View Previous Comments

Type your comment here and press Enter Key (Minimum 10 characters)

Thank you for sharing

Disne Sivalingam Mar 22, 2020

• 1820 • 237 • 4.5k 0 0 Reply

Useful information. Security is a major corner of the applications.

Subbarao A Jan 23, 2020

• 1558 • 500 • 34 0 0 Reply

Good informative article sir.

Sourav Kumar Das Nov 19, 2019

• 729 • 2.4k • 121.8k 1 0 Reply

Very important subject in a simple understandable approach. Thanks.

Ayan Ghosh Jun 05, 2019

• 1773 • 284 • 0 2 0 Reply

Thank you sir.

Prakash Chasiya Nov 27, 2018

• 867 • 1.8k • 107.5k 2 0 Reply

It's very Important information, thanks a lot

Controller SYR Apr 20, 2018

• 2011 • 46 • 0 2 0 Reply

Awesome, A single article explains this sensitive topic very well. Thank a lot for the effort.

C# Corner Apr 15, 2018

• Tech Writer • 93 • 0 3 0 Reply

Useful tips... Thanks

Satish Kumar Vadlavalli

868 1.8k 19.2m

Apr 03, 2018

👍 2 ↩ 0 Reply

Wow best article for to build Secure Websites, Thank you

Rafnas T P

168 13k 3.2m

Mar 24, 2018

👍 4 ↩ 0 Reply

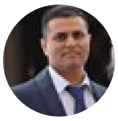
Wow my article link is here :)

Sourabh Somani

13 49.2k 8.5m

Mar 20, 2018

👍 7 ↩ 1 Reply



Yes I knew in advance that you'll write that article :)

Mahesh Chand

Admin 341.2k 201.2m

Mar 20, 2018

👍 8