Available online www.jsaer.com

Journal of Scientific and Engineering Research, 2019, 6(4):201-205

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

Strategies to Secure Web Applications: Protecting Frontend from Common Vulnerabilities

Chakradhar Avinash Devarapalli

Software Engineer
Email: avinashd7@gmail.com

Abstract The major portion of the internet is distributed on web-based systems. The network is huge as most businesses working online are dependent on their websites. The increased usage of web applications leads to malicious activities due to their potential in terms of different advantages. The security of the system is directly linked to the reputation of the company and the protection of user's data. This is why there is a need to equip the system with strong security to protect it from cybercrimes. However, the security solution is not straightforward and needs to consider different aspects while securing a system from common vulnerabilities. There are certain challenges associated with the integration of security measures with the system. Therefore, some useful strategies are presented in this research document to make the stakeholders more confident with their web applications. Above all, the system must be updated with the latest security measures to avoid being attacked with the latest methods. It is a never-ending process and organizations must figure out the methods to manage their resources accordingly to meet the security needs of the applications.

Keywords frontend security, web security, effective strategies, industrial challenges, common vulnerabilities

1. Introduction

The history of web app security has evolved from basic techniques to modern methods where each potential attack is considered to avoid system failure. A possible example of manual testing in the past which requires a lot of resources but modern techniques of black and white box testing can help to mitigate this problem [1]. SQL injections and cross-site scripting are widely used vulnerabilities in modern times [2]. However, the evolution of security methods is an ongoing study which is why continuous updates are needed to keep the system secured.

This research document mainly focused on the identification of risks associated with the security of the web front-end and the possible challenges that can restrict or slow down the process of completely securing the system. The useful strategies are presented to help individuals and organizations in making the system more secure and robust from external attacks. It will ultimately help the organizations build customer trust. It is the extension of currently available solutions to more suitable methods.

Web Applications are the primary use case of the internet. It comes with many advantages, one of which is it doesn't need to be installed on the computer like other applications like desktop or mobile apps. However, it is also true that securing these web applications is a difficult challenge and requires more effort. The insecure web front-end can expose the user data and break the user's trust ultimately affecting the user experience with the system. There are certain risks associated with the security of web applications like data breaches, SQL injections, session hijacking, exposure of sensitive data, common vulnerabilities on the client side, legal constraints, reputation damage, and more [3]. The developers need to be equipped with effective techniques for secure websites [4].

2. Literature Review

Various vulnerabilities can be targeted in the web application to gain unauthorized access to the system. They can affect both the server side and front end of the web applications. The attackers can gain access with the use of various methods by beating the system's protection and harming the intended results of the website. These vulnerable attacks can be avoided with suitable input validations, applying restrictions on the server side, utilizing HTTP protocols, and restriction users from accessing primary files of the system [5]. The widespread vulnerabilities can also be dealt with by using appropriate tools [6]

The modern tools and methodologies if used effectively can reduce the risk of vulnerable attacks on the system. The origin based access control is also being utilized for mobile applications to utilize the features of web apps by embedding the web directly into the app. Other strategies can be origin-based access control for content, isolation of external advertisements, permission re-delegation, and sandbox for advanced testing [7].

3. Problem Statement

Web applications are associated with certain vulnerabilities that can affect the security of the system. Insufficient security of the system can lead to a bad user experience, lack of trust, sensitive data exposure, financial loss, and decreased reputation of the business. Although the need for system security is inevitable, certain systems are still not completely protected due to the challenges of continuous technology evolution, the complexity of the system, and the limitation of resources. Modern security measures are therefore required to protect the web frontend from malicious attacks and to meet industry standards.

4. Challenges to Security

4.1 Technology Evolution

With the evolution of technology, more advanced methods are being used by attackers to beat security barriers. The rapid technology change therefore challenges the industry to come up with more effective solutions to avoid the potential attacks. If a company is not following the updated methods of security then it can cost seriously in many domains.

4.2 Web Apps Complexity

The architecture of web applications is now more complex with the advancements of new features and the use of technologies like node.js, and react. The management of this complex structure is therefore difficult with the increased components. Also, more systems are now incorporated with external services through the use of APIs and therefore careful handling is needed in terms of both architecture and other implementations.

4.3 Resource Limitations

Not all organizations are capable enough to meet the large budget requirements for security nor equipped with modern tools to stop malicious activities. Also, the security of the system requires a separate team with appropriate skills to restrict the attacks. All in all, there is a need for different types of resources like human resources, financial resources, physical resources, and intellectual resources for effective security of the front end.

4.4 Legal Consequences

The global and local regulations need to be followed by the companies depending upon the availability of their business. The different challenges in this regard can be, data protection, privacy, notification updates, data transfers in different regions, accessibility concerns, and intellectual property rights.

5. Strategies for Secure System

Following are the strategies that can be employed to secure web applications for safe user access [8].

5.1 Data Encryption

The data security of web applications is mainly ensured using encryption techniques. The sensitive information is protected according to the privacy policy accepted by the users [9]. For instance, UTF8 encoding is one of the most common encoding/decoding methods. The example code is,

```
function utf8Encoder(text) {
 var enc = new TextEncoder();
 return enc.encode(text);
}
function utf8Decoder(enc) {
 var decoder = new TextDecoder();
 return decoder.decode(enc);
}
var text = "Security of Web Applications";
var en = utf8Encoder(text);
var de = utf8Decoder(en);
```

The two methods that can be adapted for data encryption are,

Rest Data Encryption: The data that is stored in the database or servers of the system including sensitive information of the user is encrypted before it gets stored. The methods like Advanced Encryption Standards (AES) are used to protect data from external attacks. Here is how AES can be used for Encryption,

```
//Encryption with AES
var encrypted = CryptoJS.AES.encrypt("text", key).toString();
//Decryption with AES
var de = CryptoJS.AES.decrypt(encrypted, key).toString(CryptoJS.enc.Utf8);
```

Transfer Data Encryption: The intercepted malicious attacks while transferring data between systems and servers are blocked with strong encryption techniques. Encryption protocols like the Hypertext Transfer Protocol (HTTP) are used to encrypt data during transmission between systems.

5.2 User Authentication

The user authentication needs to be closely monitored along with data encryption to ensure the security of users. Thus, two-step verification methods can be used to avoid unwanted attacks from external entities.

5.3 Input Validations

The input from users mainly in registration and login forms needs to be validated. SQL injections are common attacks from hackers that can lead to the ultimate failure of the security of the system [10]. Here is how SQL injection can be applied if no input validation is used,

```
SELECT * FROM users WHERE username = '' OR '1'='1' AND password = 'dummy'
```

5.4 Key Encryption

The keys can be encrypted and generated securely before storing them. The measures should be taken to protect keys from unauthorized external access.

5.5 Secured Third Party Integration

The third-party integration needs to be secured. The external APIs should be integrated appropriately to avoid common attacks. Third-party services like Open API can be used while maintaining encryption. The gateway adapter helps to securely transfer data between the application and external service as can be seen in Figure 1,

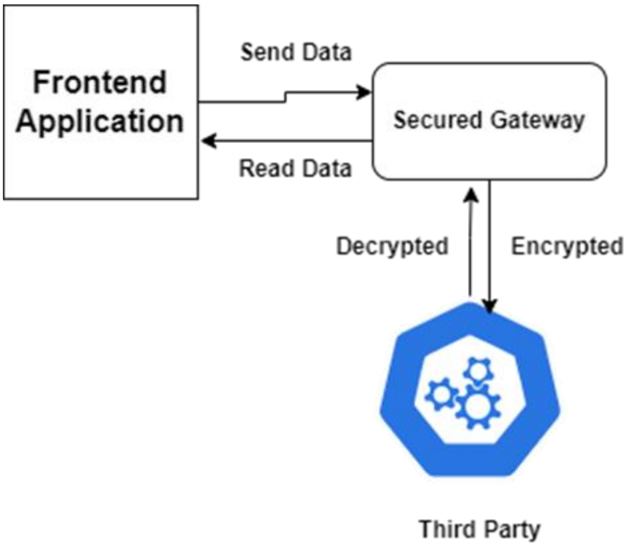

Figure 1: Secured Integration with External System

5.6 Regular Testing

To ensure continuous security, the system needs to go through testing at regular intervals. So, the manipulation efforts can be restricted on the first attempt of attacks. The encryption techniques need to be regularly tested and should be well according to market requirements to avoid attacks from the latest methods.

5.7 Continuous Encryption Updates

The attackers find loopholes in modern times and it is important to update the encryption methods regularly to restrict them. Ongoing research is needed on new strategies to stay one step ahead of the attackers as part of the security updates.

5.8 Privacy Policy

The “Privacy Policy” section is important within the system to inform the users what is going to happen with their sensitive data. All the sensitive information like payment details need to be kept private to the user only. Only limited data about the user should be made available according to the privacy policy accepted by the user.

6. Research Impact

After the effective implementation of suggested strategies, the risk of malicious attacks or common vulnerabilities can be reduced and the sensitive user’s data will be more secure in the system. This will not only help to gain customer trust but will also positively impact the business. Other benefits may include a competitive advantage in the market, legal advantages, and increased reputation. However, this needs to be considered that the security of the system is an ongoing process and more precautionary measures may need to be adopted with the evolution of the digital world.

7. Conclusion

To sum up, all the points discussed, it is significant to secure the web frontend to protect users' data and the system from potential attacks. For that, there is a need to understand the challenges against the protective measures of front-end applications. However, solutions exist to beat these challenges, and the security of the system can be achieved by employing appropriate strategies. It is significant for organizations to consider frontend security by applying effective strategies to avoid security breaches and build trust between the company and the end users. Regular research and updates are needed for the effective security of the system.

References

[1]. S. Nidhra and J. Dondeti, "Black Box and White Box Testing Techniques," *International Journal of Embedded Systems and Applications (IJESA)*, vol. 2, no. 2, Jun. 2012.

[2]. Y.-W. Huang and D. T. Lee, "Web Application Security—Past, Present, and Future," in *Computer Security in the 21st Century*, National Science Council, 2005, pp. 183-227.

[3]. J. Meier, "Web application security engineering," *IEEE Security & Privacy*, vol. 4, no. 4, pp. 16-24, Aug. 2006.

[4]. M. Cross, Developer's Guide to Web Application Security, United States and Canada: Syngress Publishing, Inc. , 2007.

[5]. D. Yadav, D. Gupta, D. Singh, D. Kumar and U. Sharma, "Vulnerabilities and Security of Web Applications," in *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, Greater Noida, India, Dec. 2018.

[6]. A. Alzahrani, A. Alqazzaz, Y. Zhu, H. Fu and N. Almashfi, "Web Application Security Tools Analysis," in *2017 ieee 3rd international conference on big data security on cloud (bigdatasecurity), ieee international conference on high performance and smart computing (hpsc), and ieee international conference on intelligent data and security (ids)*, Beijing, China, Jul. 2017.

[7]. M. Georgiev, S. Jana and V. Shmatikov, "Rethinking Security of Web-Based System Applications," in *WWW '15: Proceedings of the 24th International Conference on World Wide Web*, May. 2015.

[8]. S. Kumar, R. Mahajan, N. Kumar and S. K. Khatri, "A study on web application security and detecting security vulnerabilities," in *2017 6th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, Noida, India, Apr. 2018.

[9]. W. He, D. Akhawe, S. Jain, E. Shi and D. Song, "ShadowCrypt: Encrypted Web Applications for Everyone," in *CCS '14: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, Nov. 2014.

[10]. J. Clarke-Salt, SQL Injection Attacks and Defense, Waltham, USA: Elsevier, 2012.