CS 6417 – Software Security

Final Project Report

Submitted by

Name: Ashique Arman
ID: 3739860

Contents

Abstract

This project report covers the development of a full-stack website using the MERN (MongoDB, Express.js, React.js, Node.js) stack and following agile methodologies for efficient and iterative development. While developing the website, I have taken several strong security measures to safeguard against potential cyber-attacks, particularly focusing on securing the login and contact fields from malicious exploitation. Through the implementation of agile practices, the project was approached with adaptability and responsiveness, ensuring timely delivery and effective mitigation of emerging challenges. An attack tree and surface were meticulously designed by me, offering a comprehensive perspective from the standpoint of potential adversaries. Lastly, I have mentioned what I have learned throughout the whole development process of this project.

Introduction

I have developed this full-stack website that has the following functionalities:

1. Registration
2. Login
3. Dashboard for the user
4. Contact Page
5. Buying functionality

This project was built using the MERN stack (MongoDB, Express, ReactJS, and NodeJS). To safeguard the website against potential cyber threats I have implemented several security measures which are described in this report.

I have chosen the agile methodology for the development of this project. The reasons are discussed below:

1. Agile is an iterative process. As a result, I had the choice of developing some of the more important aspects of the development earlier than the less important ones. For example, I developed some parts of the front-end section of the project. I created the home page and registration page first. After developing these, I went on to create the login page and then the contact page. After fully finishing the front-end part, I completed the back-end part. This helped me to develop the website bit by bit while maintaining my time and right state of mind; not rushing things.

2. As we already know agile promotes iterative development, allowing for regular security reviews throughout the development lifecycle. By incorporating security assessments into each sprint or iteration, I was able to identify and address vulnerabilities early, and as a

result, reduced the risk of security breaches and mitigating potential damage.

3. Agile offers flexibility. I benefited from it as on several occasions, I forgot to implement certain features of a webpage or a component. So, I went back and wrote more codes to implement those. There were no issues and this flexibility aspect of agile was a great feature in my eyes.

4. Agile is an adaptable process. During the project development time, new ideas came to my mind, and added those ideas to the project. This helped me to have a sense of adaptability and not be fixated on a certain set of features.

5. Agile methodology helped me to reduce potential risks as the development process was broken down into smaller pieces. This helped me to manage and monitor the overall project throughout the project development timeline. I identified multiple errors and managed to fix them, thanks to the agile process.

Attack Surface

Registration Page

1. **Duplicate Registrations:** Changing email addresses that are already there by adding special characters like: "+1@" or leaving empty spaces.

Login Page

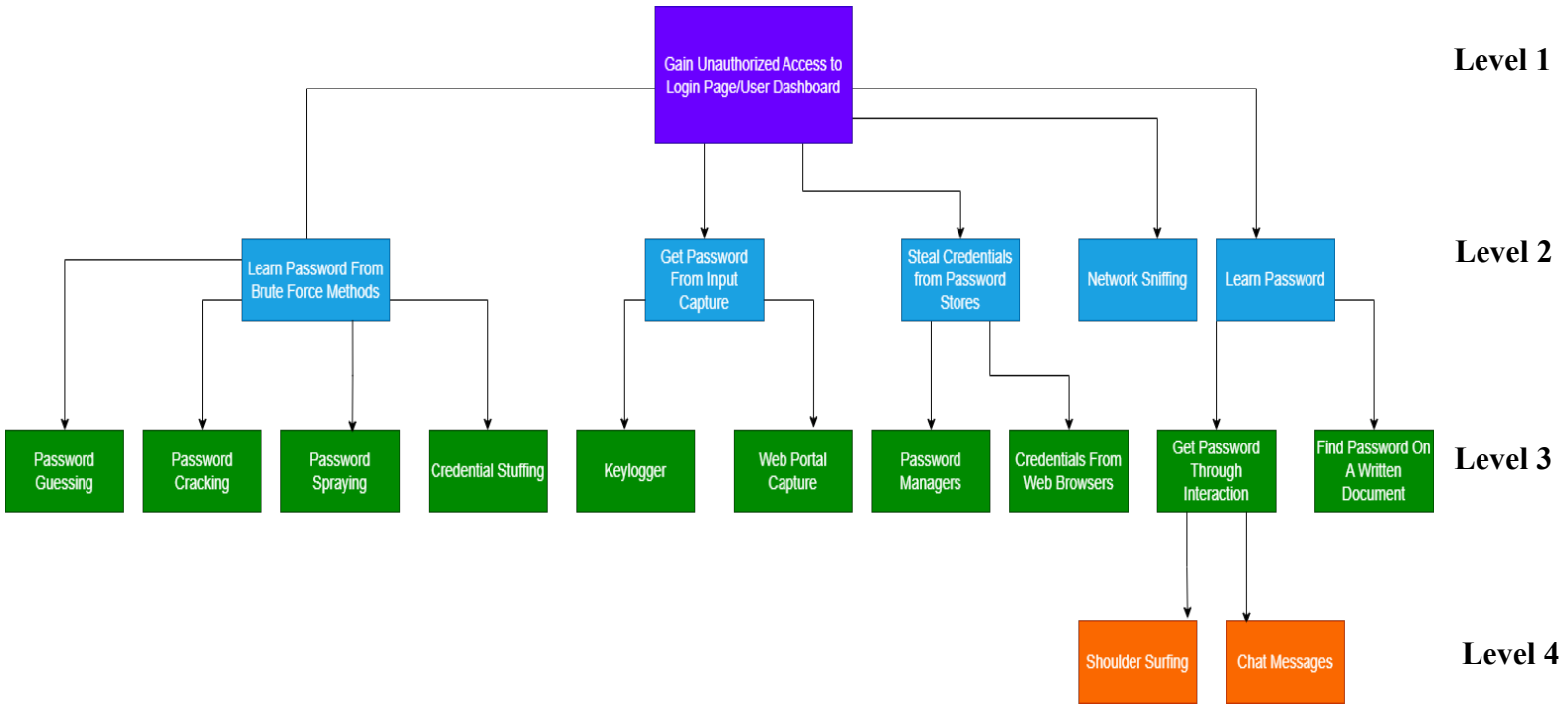
1. **Brute Force Attack:** Attackers might use different tools and techniques to guess passwords.
2. **Credential Stuffing:** Attackers could try using usernames and passwords stolen from leaked databases to get into the website.
3. **Session Hijacking:** If the website doesn't properly secure session tokens well, attackers could take them and pretend to be real users.

Contact Page

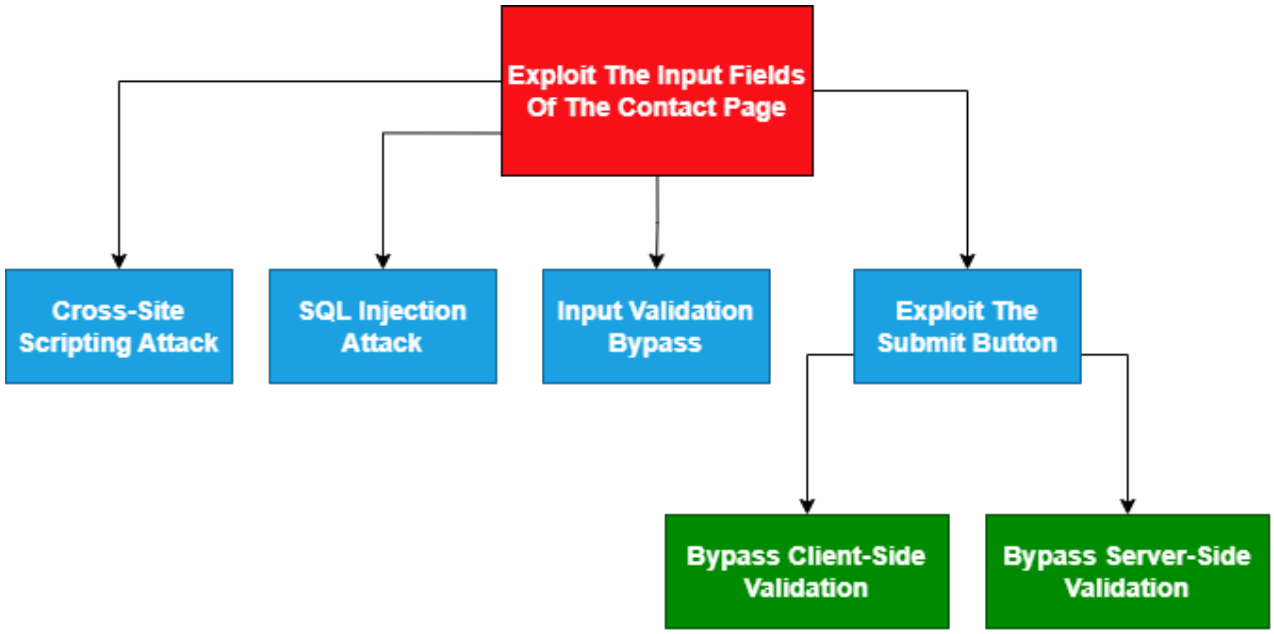
1. **SQL Injection:** If the info typed in (like Name, Email, Message) isn't validated or sanitized properly, hackers could sneak harmful stuff into the website's database.
2. **Cross-Site Scripting (XSS):** This lets attackers add malicious code to the page, which can mess up other people's browsers.

Attack Tree

*Attack Tree for the **Login Page***



*Attack Tree for the **Contact Page***



I've been utilizing the MITRE ATT&CK framework, developed by the MITRE Corporation, which is widely recognized for offering structured insights into adversary tactics and techniques based on real-world observations. ATT&CK, standing for Adversarial Tactics, Techniques, and Common Knowledge, has been instrumental in my approach.

In designing the attack surface and attack trees, I've heavily leaned on the MITRE ATT&CK framework. This involved consulting the framework extensively. I have discussed this below.

1. I've delved into the framework's categorization of adversary behavior into tactics and techniques. This allowed me to grasp the objectives attackers aim for (tactics) and the specific methods they employ (techniques).

2. When designing attack trees, I've focused on aligning primary goals or tactics, such as gaining unauthorized access, with the corresponding categories in MITRE ATT&CK.

3. I have carefully enumerated various techniques attackers might employ to achieve their goals, including exploiting vulnerabilities, bypassing authentication mechanisms, or stealing credentials.

4. Organizing the attack tree hierarchically, with the primary goal at the top and branching into more specific techniques and sub-techniques, allowed me to create a detailed representation of potential attack vectors.

5. I've consistently cross-referenced the attack tree with the MITRE ATT&CK framework to ensure comprehensive coverage of adversary tactics and techniques.

6. The insights provided by the MITRE ATT&CK framework have been invaluable in identifying potential attack vectors and vulnerabilities in the authentication mechanisms of the login page.

MITRE ATT&CK's standardized classification has been essential for organizing and prioritizing security controls and countermeasures to mitigate identified risks effectively. The structured approach facilitated by MITRE ATT&CK has significantly enhanced my analysis of the attack surface, providing a deeper understanding of potential threats to the login page and the overall security posture of the website.

Technical Controls

As the developer of the website, I've taken several crucial steps to fortify its defenses against potential cyber-attacks and malicious activities. Here's a discussion of the technical controls I've implemented:

1. **Password Security with bcrypt:** I've prioritized the protection of user passwords by employing the bcrypt hashing algorithm. This choice ensures that even if password hashes are compromised, they remain exceedingly difficult to crack, thwarting brute-force attacks effectively.

2. **Robust Authentication and Authorization:** I have built a robust authentication and authorization system ensuring that only authenticated users can access sensitive areas of the website. This setup effectively blocks unauthorized access attempts, safeguarding user accounts and sensitive data.

3. **Input Validation and Sanitization:** To prevent injection attacks like SQL injection or cross-site scripting (XSS), I have made sure to validate and sanitize all user inputs. By scrubbing input data of any potentially malicious content, I prevent attackers from injecting harmful code into the application.

4. **Effective Session Management:** By implementing best practices in session management, such as secure session cookies and robust session token generation and validation, I've effectively mitigated risks associated with session hijacking.

Testing

I have chosen NMap for testing the vulnerabilities of my website. With NMap, I conducted scans to identify any open ports, services running on those ports, and potential vulnerabilities in my network infrastructure.

During the testing, I discovered some open ports that were not necessary for the website's functionality. By closing these unnecessary ports, I reduced the potential attack surface and enhanced the overall security of my website. Additionally, NMap helped me identify outdated services and configurations that needed to be updated to patch known vulnerabilities.

Discussion

Through the development of my website using the MERN stack and the adoption of Agile methodology, I've gained invaluable insights into the world of web development and cybersecurity. One of the most important takeaways from this project is the significance of security measures in safeguarding digital assets.

By incorporating the MITRE ATT&CK framework into my security strategy, I've learned to anticipate and defend against a wide range of potential cyber threats. This framework provided a structured approach to identifying and addressing vulnerabilities, ensuring that my website remains resilient against various attack vectors.

Furthermore, employing NMap for vulnerability scanning has enabled me to proactively identify potential weaknesses in my website's infrastructure. This hands-on experience in vulnerability assessment has deepened my understanding of common security pitfalls and reinforced the importance of regular security audits in maintaining robust defenses.

In addition to security measures, this project has reinforced the importance of user-centric design and functionality. Features such as registration, login, and purchasing capabilities were implemented with the user experience in mind, aiming to provide seamless and intuitive interactions.

Moreover, adhering to Agile principles throughout the development process has taught me the value of adaptability and collaboration. Embracing iterative development cycles allowed for continuous improvement and responsiveness to evolving requirements, ultimately resulting in a more robust and user-friendly website.

Appendix

Here is the GitHub link to the project:

https://github.com/AshiqueArman/marketplace

References

1. *Credential access* (no date) *Credential Access, Tactic TA0006 - Enterprise | MITRE ATT&CK®*. Available at: https://attack.mitre.org/tactics/TA0006/ (Accessed: 04 April 2024).

2. *Registration & Takeover vulnerabilities: HackTricks* (no date) *HackTricks*. Available at: https://book.hacktricks.xyz/pentesting-web/registration-vulnerabilities (Accessed: 04 April 2024).

3. *CS 6411 Slides (Chapter: Risk Management. Page: 23-25)*