# Best Practices for Securing MERN Stack Applications: A Comprehensive Study of Authentication, Authorization and Data Protection

**Md Shahriar Nur Chowdhury**

Master's thesis
December 2024
Masters in Full-Stack software development

Jyväskylän ammattikorkeakoulu
University of Applied Sciences

**Chowdhury, Md Shahriar Nur**

**Best Practices for Securing MERN Stack Applications: A Comprehensive Study of Authentication, Authorization and Data Protection**

Jyväskylä: Jamk University of Applied Sciences, December 2024, 53 pages.

Degree Programme in Full Stack Software Development. Master’s thesis.

Permission for open access publication: Yes

Language of publication: English

## Abstract

With the increased importance of web applications in every field, the demand for security has risen, especially in frameworks like the MERN stack. This thesis covers the best practices for securing MERN stack applications, focusing on three critical areas: authentication, authorization, and data protection. The MERN stack, while offering exceptional flexibility, scalability, and efficiency, has its unique security challenges introduced by its layered architecture, which must be addressed to gain the trust of users and meet regulatory standards.

This study adopts a qualitative research methodology, which involves interviews with software developers and security experts from leading technology companies from Bangladesh. The results showing some useful practices, such as the application of JWT (JSON Web Tokens) for authentication without sessions, RBAC (Role-Based Access Control) and ABAC (Attribute-Based Access Control) authorization methodologies, and encryption algorithms AES (Advanced Encryption Standard) for secure data encryption and TLS (Transport Layer Security) for securing data in transit against sensitive data disclosure. It also points out the importance of secure coding practices, vulnerability assessments, and adherence to global compliance frameworks like GDPR (General Data Protection Regulation) and CCPA (California Consumer Privacy Act).

It provides a review of how this thesis, through an all-inclusive security framework relevant to MERN applications, offers applicable insights for developers in mitigating certain identified vulnerabilities toward hardening the apps. The results underscored how critical incorporating security throughout the software development life cycle has been in dealing with emerging cyber threats without compromising performance and user experience, hence requiring concentrated consideration and investment. The research provided insight into recommendations for the future; first, investigating advanced attack vectors and integration of AI-driven mechanisms for security.

## Keywords/tags (subjects)

MERN stack, authentication, authorization, data protection, web application security, cybersecurity, regulatory compliance

# Contents

## List of Tables

## List of Abbreviations

| | |
|---|---|
| ABAC | Attribute-Based Access Control |
| AES | Advanced Encryption Standard |
| API | Application Programming Interface |
| BJIT | Bangladesh Japan Information Technology |
| CCPA | California Consumer Privacy Act |
| CSRF | Cross-Site Request Forgery |
| CSP | Content Security Policy |
| DevSecOps | Development, Security, and Operations |
| GDPR | General Data Protection Regulation |
| HIPAA | Health Insurance Portability and Accountability Act |
| JWT | JSON Web Token |
| LAMP | Linux, Apache, MySQL, PHP/Perl/Python |
| MERN | MongoDB, Express, React, Node.js |
| MEAN | MongoDB, Express, Angular, Node.js |
| nmp | Node Package Manager |
| NoSQL | Not Only Structured Query Language |
| OWASP | Open Web Application Security Project |
| RBAC | Role-Based Access Control |
| REST | Representational State Transfer |
| SQL | Structured Query Language |
| TLS | Transport Layer Security |
| XSS | Cross-Site Scripting |

# 1 Introduction

## 1.1 Background of the Study

Due to the rise of technology in the past couple of decades or so, the world has seen an unprecedented integration and utilization of web applications across various industries such as finance, health care, retail sector and social media across the entire world. As a consequence of this, this expansion has caused a massive need for increased web security especially due to the fact that these web applications handle sensitive personal data of clients, vendors, organizations and even financial data (Veeraiah et al., 2022). In this digital world, security has become one of the top priorities as cyber threat continues to grow at a similar precedented rate. As a result, the security of web applications as much of a challenge for developers as building it as it is directly correlated to client's trusting and regulatory framework Vallabhaneni et al., 2024). Due to this phenomenon, security has become a central theme for development stacks such as MERN stack which comprises of MongoDB, Express, React and Node.js as these are the most widely used frameworks due to their speed, flexibility as well as scalability. The MERN stack in particular has gained immense popularity in the modern times due to the efficiency it provides developers in terms of creating single page applications which are both interactive and responsive (Ishaq et al., 2023). Originally built on JavaScript all the way from front to back end, MERN stack is favoured by many due to its ability to handle high engagement platforms such as E-commerce, Social Media and SaaS platforms (Desai & Fiaidhi, 2022). Working as a database for MERN are Not Only Structured Query Language (NoSQL) database and MongoDB among others which are both capable of handling unstructured, as well as semi-structured data, making MERN one of the more powerful tools for stability. Node.js and Express work well with MERN on the back end whereby both manages server-side logic and React on the other hand is deployed to use front-end which creates fast and interactive user interfaces for the users. Irrespective of the many advantages that the MERN stack brings to the table, it is at the end of the day another piece of technology which is vulnerable to security threats such as data breachers, malicious attacks and unauthorized access. This factor makes it extremely important for developers to understand the concerns for security and implement best security practices (Nguyen, 2021).

From the perspective of MERN, security concerns could rise at multiple layers, making it vulnerable in multiple entry points. For example, MongoDB, irrespective of it being extremely efficient to handle unstructured data, can be extremely susceptible to injection attacks if not configured properly. This happens due to the fact that it lacks the inherent structure of Structured Query Language (SQL) databases which enforces more stringent validation (Eyada et al., 2020). In a similar fashion Node.js, being extremely popular, has a vast array of ecosystems of modules and libraries which is beneficial to developers in terms of speed, but remains vulnerable to attack if the modules are note carefully vetted (Huang, 2020). Further adding more complexity to the security management are Express and React. For example, React is vulnerable in the user end where issues like cross-site scripting XSS (Cross-Site Scripting) can occur, and Express on the other hand could be vulnerable in the server side with attacks such as cross-site request forgery CSRF (Cross-Site Request Forgery) could take place (Steffens, 2021). Due to these multiple vulnerabilities associated with MERN, a need of integrated security has never been more important. The integrated security could handle the vulnerabilities, all the while maintaining the performance and the user experience that MERN provides (Seh et al., 2020; Nair, 2024).

Two of the core fundamental components of MERN are authentication and authorization. Authentication is the process through which users are identified before they are granted access. Authorization on the other hand is about defining and enforcing mechanisms which permits or limits what an authenticated user can do (Satriyo et al., 2024). If the authentication process is not managed well, this can lead to unauthorized access, thereby exposing sensitive data to individuals with malicious intent. Some of the common authentication process involved in the MERN applications are JWT (JSON Web Tokens) and OAuth, both having their very own security implications (Mai, 2020).

The process of authorization can be complex in MERN applications whereby different users have different roles and permission levels. The role based access control (RBAC) is one of the most commonly used methods in respect to managing MERN authorization, which ensures that each users are able to access the resources they are assigned to as per their roles within the system (Blundo et al., 2020). Incorrect implementation or disproportionate use cases of authorization can lead to privilege escalation whereby a user gains higher access than intended, which can pose a significant threat to the system's integrity (Saxena & Alam, 2023).

Another critical aspect within the MERN stack is data protection. This is especially due to the fact that MongoDB and NoSQL database does not enforce schema restrictions, therefore making it extremely susceptible to injection attacks in case it is not secured (Shwetha et al., 2024). Some of the measurements that can be taken for the purpose of data protection includes encryption, enabled both during transit and at rest, thereby ensuring the safety against attackers even when data is transitioning. Other measurements for data protection are Advanced Encryption Standards or (AES) which is particularly used during rest and Transport Layer Security (TLS) which is used during transit, especially between users and servers in Application Programming Interface (API) communication (Tezcan, 2021).

Other than individual need for security, there is global trend of regulatory framework implementation associated with security, which mandates the usage of security against cybercrimes (Weber, 2022; Shwetha et al., 2024). For example, there are compliance laws such as General Data Protection Regulation (GDRP) which is a European regulatory framework. On the other hand, there is the California Consumer Privacy Act (CPPA) which is a regulatory framework of the United States. These regulations coerce developers to take necessary steps to provide appropriate security in their applications (Baik, 2020; Desai & Fiaidhi, 2022).

As a result of the significance of MERN security pointed out so far in this chapter, makes it a compelling case for further research. This thesis therefore examines the best practices in regard to utilizing security in MERN stack applications to provide future direction to practitioners and academicians alike.

## 1.2 Problem Statement

When considering web applications, security is of immense significance, as it has been pointed out in the previous part of the chapter. With MERN stack utilizing elements such as MongoDB, React, Express and Node.js, it has become extremely popular. Its popularity stems from the efficiency they provide to the developers, enabling them to create dynamic single page apps giving seamless user experience and collect complex data at the same time (Ishaq et al., 2023). Irrespective of its popularity, it has major security concerns and unique challenges. To address security concerns associated with MERN security, three primary areas require addressing which are: authentication, authorization and data protection.

## 1.3 Research Questions

The study is guided by the following key research questions:

- What are the most effective authentication methods for ensuring secure access in MERN stack applications?
- How can secure authorization and role management be implemented to protect resources within MERN applications?
- What data protection techniques best safeguard sensitive information in MERN stack applications?

## 1.4 Research Objectives

The primary objectives of this study are as follows:

1. To identify and evaluate effective authentication methods that enhance security in MERN stack applications, providing guidance on selecting and implementing robust user authentication.
2. To develop secure authorisation and role management practices within MERN applications, focusing on implementing role-based access control (RBAC) to restrict and manage access to sensitive resources.
3. To explore and recommend data protection techniques that ensure the confidentiality and integrity of sensitive information, including data encryption and secure storage within MongoDB.

## 1.5 Significance of the Study

This thesis is holds significance because of many reasons to multiple stakeholders such as web developers, cyber security experts, organizations and academicians. For stack developers, this research could potentially provide insights in regard to the best practices for security in MERN applications. Furthermore, software engineers as well as architects can benefit from this thesis as it would provide them with a best practices framework. Educators and other academicians could benefit from this thesis for building up on further research.

## 1.6 Scope of the Study

As mentioned before in this chapter, this thesis particularly focuses on security aspects surrounding MERN stack which incorporates MongoDB, Express React and Node.js. As per the research questions, this study would primarily focus on three aspects which are authentication, authorization and data protection. The study will further explore security vulnerabilities associated with XSS, CSRF, SQL injection and improper session management as well.

## 1.7 Limitations of the Study

As much as this thesis contributes to understanding best security practices surrounding MERN, there are certain limitations to this study. To start off, the research relies on qualitative interview data conducted on certain tech companies in Bangladesh, which may not necessarily reflect the total opinion of the industry. Moreover, even though this research would provide a practical approach to evaluating security feature, it may not fully encompass the security requirement of the different MERN based applications that are available. To add to this, the findings of this study may be difficult to generalize beyond the context of MERN stack based applications.

## 1.8 Research Methodology Overview

To determine the best security practices for MERN stack applications, a qualitative research approach has been adapted. An in-depth semi structured interview was conducted on

Bangladesh tech companies that utilized MERN stack development. The general idea is to determine the best practice regarding security associated with MERN, which will contribute both to the industry as well as the research realm.

### 1.8.1 Data Collection Through Developer Interviews

As mentioned in the previous section, semi structured interview will be conducted with top level and mid-level management of certain tech companies in Bangladesh who uses MERN applications. The list of companies along with a short bio is provided below:

1. **BJIT Ltd.**

   BJIT Ltd. Is a globally recognized Bangladesh software company. Their expertise lies in full stack development which includes MERN applications. The company serves on international projects, thereby making them the perfect company to be a sample for this study.

2. **Datasoft Systems Bangladesh Ltd.**

   One of the largest software firms in Bangladesh is Datasoft and they have a vast experience in dealing with international clients in the fields of e-commerce, finance and health care. As they have huge experience in creating MERN stack apps, they are a prime candidate for this study.

3. **Brain Station 23**

   Another leading software company in Bangladesh is known as Brain Station 23. They too handle a lot of international clients, building MERN based applications. They too were chosen as a sample for this study for this reason.

4. **Reve Systems**

   Specialising in software for communication and security, Reve Systems has extensive experience in securing data-driven web applications. Their developers' input on MongoDB-specific vulnerabilities will enrich the study.

5. **Kaz Software**

   An established Bangladeshi software firm known for developing bespoke web applications, Kaz Software has expertise in implementing secure authentication and authorisation mechanisms in MERN-based projects.

### 1.8.2 Participant Selection Criteria

Participants will be selected based on their:

- Experience with the MERN stack in developing and deploying web applications.
- Expertise in implementing security features such as authentication, authorisation, and data protection.
- Involvement in addressing security challenges in high-profile or large-scale applications.

This targeted selection ensures the relevance of the qualitative data to the study's objectives, focusing on practical, actionable insights.

### 1.8.3 Purpose of the Interviews

The interviews aim to:

1. Identify common security challenges developers encounter when working with MERN applications.
2. Explore the tools, techniques, and frameworks they use to address these challenges.
3. Highlight gaps in existing practices and opportunities for improvement.

### 1.8.4 Data Analysis

Thematic analysis will be employed to examine the qualitative data. This method will:

- Identify recurring patterns in the participants' responses.
- Highlight key insights related to securing authentication, authorisation, and data protection in MERN applications.
- Inform the development of a comprehensive security framework tailored to the MERN stack.

# 2 Literature Review

## 2.1 Introduction

As highlighted in the previous chapter, securing MERN stack applications is essential not only for protecting user data but also for maintaining user trust, complying with regulatory requirements, and ensuring the overall integrity of web services (Satriyo et al., 2024; Desai & Fiaidhi, 2022; Huang, 2020; Yadav et al., 2024).

The purpose of this literature review is to critically examine existing research, theories, and practices related to the security of MERN stack applications through synthesizing current knowledge, this chapter aims to identify gaps in the literature that the present study seeks to address. The review is integral to the study as it provides a foundation upon which the research objectives are built, informing the methodology and guiding the analysis of findings. It enables a comprehensive understanding of the challenges and solutions associated with authentication, authorisation, data protection, and general security concerns within the MERN stack framework.

Authentication, authorization, and data protection are crucial for web application security, especially in MERN stack applications, given the unique challenges posed by MongoDB's flexible schema, Express's middleware, React's client-side rendering, and Node.js's event-driven environment (Nagarathinam & Mythili, 2024; Rathore & Bagui, 2024). Authentication methods like JWT and OAuth, along with authorisation mechanisms such as role-based access control (RBAC), face challenges like client-side manipulation and JavaScript's asynchronous nature (Satriyo et al., 2024; Mpamugo & Ansa, 2024). Data protection strategies, including encryption, secure storage, and input validation, are critical to prevent vulnerabilities like injection attacks in MongoDB (Shwetha et al., 2024; AbhishekKumar et al., 2024).

In summarizing the key themes:

1. Authentication: This part of the research will address various security concerns in regard to the authentication process that is deployed in MERN stack. It will be looked through the scope of security implications, scalability as well as end user experience. The literature will attempt to uncover typical and hidden security challenges and best

security practices for mitigations (Nagarathinam & Mythili , 2024; Rathore & Bagui, 2024).

2. Authorization: The literature will explore how authorization is implemented within the MERN stack and its role in granting permissions to users of different hierarchical levels. Other than identifying the common vulnerabilities, it will also evaluate security measurements such as RBAC and ABAC models for mitigations (Mpamugo & Ansa, 2024).

3. Data Protection: Furthermore, this literature will further explore data protection, its storage, encryption methods, protection and its vulnerable points, both in rest and in transit. As per the other elements in the literature review, this part too will focus on best security practices which can be applied for data protection in the MERN stack (Perrone et al., 2024).

4. General Security Concerns in MERN: Other than the three crucial aspects of security in the aforementioned section, the literature will further dive into understanding general security concerns in the MERN stack. This part will examine existing literature on common vulnerabilities and the common practice in mitigating these vulnerabilities (Mpamugo & Ansa, 2024).

## 2.2 Overview of Web Application Security

When it comes to web application security, it usually involves integrity, confidentiality and availability through best practices that protects the application against threats (Siderova et al., 2024). Such practices usually incorporate mix of secure coding, authentication and authorization tools and incident response to protect data (Marchitelli, 2024). As the popularity of digital spaces grow even today, the need for protection against financial losses and data breaches are ever higher (Chahal et al., 2022). At the same time, regulations such as Health Insurance Portability and Accountability Act (HIPPA) and GDPR demands for web platforms to be secured, making the web development world more technical and at the same time an ethical mandate (Kessel et al,. 2023)

Some of the common vulnerabilities to MERN stacks include XSS, CSRM and SQL injection which could potentially case a lot of damage, increasing the need for robust protection (Kaur et al., 2023; Alghawazi et al., 2022). To manage risks effectively tools such as OWASP's (Open

Web Application Security Project) top ten list tools such as Application Security Verification Standard (ASVS) and Zed Attack Proxy (ZAP) has long remained a potential guide for developers using MERN stack (Fred et al,. 2021; Kuncoro & Rahman, 2022). A pattern could be seen in MERN development regarding MongoDB, Express, React and Node.js every single year. To start with MangoDB is susceptible to injection attacks, Express is vulnerable to middleware, React is vulnerable in the client's side and Node.js has dependency problems (AbhishekKumar et al,. 2024; Sahni et al., 2024). To tackle this complex web of security issues, tailored strategies are required to ensure a resilient framework.

In the modern context, some of the bigger security challenges is associated with rapid development cycles, expanding attack surfaces and issues associated with users such as weak passwords (Zahn et al., 2023; Ishaq et al., 2023). Irrespective of the challenges mentioned, effective strategies to counteract these obstacles can mitigate some of these challenges. With a combination of legal regulatory compliance, best security integration, ethical consideration and trust of the uses, developers can create robust security measurements through the scope of technical safeguards (Veeraiah et al., 2022).

## 2.3 The MERN Stack and Its Security Implications

### 2.3.1 Architecture of the MERN Stack

MERN stack, as mentioned before is one of the most popular full stack that works through the JavaScript framework, primarily use in the modern context for web applications development. The stack which consist of MongoDB,, Express, React and Node.js has specific functionality which are provided below:

Mongo DB: MongoDB is a NoSQL database that is able to store large and flexible data in JSON format allowing large dataset to be managed efficiently (Rathore & Bugai, 2024).

Express: Express on the other hand is a web app framework particularly for Node.js which aids in the creation of server side applications which provides both flexible and a minimal approach (Huang, 2020)

React: In contrast to Express, Reach is a front end library which is responsible for the creation of front end user interfaces. It particularly aids developers to create dynamic and responsive apps (Johnsson, 2020).

Node.js: Finally Node.js allows for JavaScript to run on the server side, which provides a runtime environment which helps non-blocking and event-driven programming (Nguyen, 2021a).

As the entire MERN stack uses a unified JavaScript environment, it allows for the developers to use a single programming language across all the layers, making it easier for them to streamline the process of creating the app they are working on, integrating both back and front end components (Desai & Fiadhi, 2022).

### 2.3.2 Security Implications of the Unified JavaScript Environment

While the unified JavaScript environment simplifies development, it also introduces unique security considerations. A common language across the stack increases the likelihood of cascading vulnerabilities, where a security flaw in one layer can propagate to others. This makes a holistic approach to security essential in MERN applications (Fredj et al., 2021).

### 2.3.3 Security Challenges Specific to Each Layer

In the previous part of this chapter, it outlined the use case for MERN. This part of the paper would outline each component of MERN to determine the security concerns.

MongoDB: As MongoDB lacks the integration of schema enforcement, it is particularly susceptible to injection attacks. Through unvalidated input, malicious attackers could execute arbitrary database query which would potentially expose and modify sensitive data (Rathore & Bagui, 2024). This could potentially be mitigated through parameterized queries and enforcing input validation (Fredj et al., 2021).

Express and Node.js: For both Express and Node.js Middleware Vulnerabilities and Dependency issues could be a problem as they both rely heavily on middleware to handle HTTP requests. Any form of improper configuration could be a problem which could expose sensitive data and allow unauthorized access to routes (Huang, 2020). One of the issues with Node.js is that it depends on third party npm packages and could therefore become vulnerable to dependency vulnerabilities as well as supply chain attacks through the introduction of malicious codes (Philippaerts et al., 2022).

React: As React is primarily used in the front end side of the development, vulnerabilities could be introduced in the client's end. Some of the common vulnerabilities that React faces are XSS attacks where developers use dangerously Set Inner HTML attribute. Through the use of unsanitized input, attackers could potentially inject as well as execute malicious scripts in the user's browsers. This could potentially compromise session tokens and even sensitive data (Kaur et al., 2023). However, to mitigate these risks, one could deploy libraries like DOMPurify for input sanitization and enforce strict Content Security Policies (She et al., 2020).

## 2.4 Authentication in MERN Applications

As outlined in introduction, authentication is the core of security associated with web applications. The authentication enables developers to legitimate and limit users in terms of the access they will have to the web app. In a MERN stack where back and front end are integrated for seamless user experience, authentication plays a crucial role in terms of protecting sensitive information as well as enabling various security layers such as data protection and authorization (Desai and Fiaidhi, 2022; Ishaq et al., 2023). However, since MERN applications are favoured by many due to its scalability, there is a lot of challenge surrounding authentication system, creating room for custom tailored security measurements (Nguyen, 2021a).

Session-based authentication remains one of the oldest and most widely used methods, relying on server-stored sessions and unique session IDs in cookies. While effective in small-scale applications due to its simplicity and robust server-side control, it faces scalability limitations and risks like session hijacking (Mai, 2020; Nair, 2024). Libraries like express session make its implementation straightforward for MERN applications, but its suitability diminishes in distributed systems (Nguyen, 2021a). In contrast, JSON Web Tokens (JWT) have gained popularity for their stateless architecture, allowing for scalable and cross-domain authentication. JWTs store tokens on the client side, reducing server resource load but introducing risks such as exposure to cross-site scripting (XSS) and token expiration challenges.

Careful handling of token storage and lifecycle is essential to mitigate these vulnerabilities (Shwetha et al., 2024a; Kaur et al., 2023).

OAuth is one of the more recognized third party logins through providers such as Google and Facebook, allowing for developers the ease of convenience in terms of credential management responsibilities or apps. The open standard nature of OAuth allows for delegation of access without having to share user credentials. However, some of the bigger challenges in this situation occurs through redirection of URI manipulation and token leakage which could leak information. Therefore, management of these could be done through careful configuration and encrypted communication (Philippaerts et al., 2022).

When authentication processes are poorly managed, these systems pose a significant threat of token interception and insecure storage to weak session management as well as brute force attacks. in case of MERN, such attacks could sieve through the stack system and expose MongoDB for unauthorized database access, Express for API exploitation, React facing client-side attacks and Node.js for unvetted libraries (Rathore & Bugai, 2024; Fredj et al., 2021).

Therefore, by addressing these vulnerabilities on each of the MERN stack, the developers could potentially increase resilience and security. Through adaptation of best practices such as secure token storage, robust session handling, rate-limiting and combining it with levering tool such as express-session, could potentially provide a robust defence against threat (Shwetha et al., 2024a; Nguyen, 2021a).

## 2.5 Authorization in MERN Applications

Through Authorization, the developers could allocate resources and actions to a particular authenticated user and limit their usage as per their credentials. In MERN stack applications, authorization is deployed as a secure functionality to prevent unauthorized actions such as modifying, viewing or editing resources (Saxena & Alam, 2023). In contrast to authentication which grants users access, authorization on the other hand allocates permissions and different access to resources as per credentials. Therefore, when implemented poorly, it could lead to severe breaches in the site, which includes unauthorized control and data leaks, making it one of the top priority for security measurements (Mpamugo & Ansa, 2024).

### 2.5.1 RBAC and ABAC in Authorization

When it comes to RBAC it is often used by developers to assign predetermined roles to various users through following a hierarchical roadmap. Roles are assigned as per credentials such as admin, editor and viewer. RBAC is one of the more common security measures taken by many developers across the word. However, RBAC may struggle when there are complex roles to be assigned in a dynamic engaging environment whereby there are users who are also granted special access to the web app (Blundo et al., 2020). When there are demands for web app to have dynamic access roles to be assigned to users, ABAC is often deployed. ABAC utilizes location data and other forms of data such as type of device and give roles to users accordingly. ABAC is more complex in nature than RBAC due to its adaptive nature. One of the downside of ABAC is that it is energy intensive (Siderova et al., 2024).

### 2.5.2 Common Vulnerabilities in Authorization

When authorization configuration is improperly managed, risks such as privilege escalation, insufficient granularity in access levels as well as overly permissive roles can occur. Through improper integration of access policies or misconfigured end points, MERN could face issues as such (Blundo et al., 2020). For example, if a user is granted access to admin privileges due to user role mismanagement, it could lead to severe data breaches or even system compromise (Saxena & Alam, 2023).

### 2.5.3 Best Practices for Secure Authorization

In order for properly implementing security measures to MERN applications, RBAC or ABAC implementation is required which will be based on use case of the web application. Through enforcement of least privilege principles, which allows for minimum necessary permissions could potentially mitigate risks. Through centralized access control mechanism, the policy enforcement could be implemented consistently. By running regular audits, unnecessary permissions could be removed and further ensure compliance. In Express applications, tolls such as express-act could further simplify the role-based control, further enhancing the security (Mpamungo & Ansa, 2024).

## 2.6 Data Protection in MERN Applications

For the protection of data’s confidentiality, availability and integrity, data protection is paramount in web apps. In MERN stack apps, enabling data protection could involve securing data while resting as well as while it is in transit which could prevent regulatory penalties, prevent breachers and erosion of user trust ( Tezcan, 2021, Rathore & Bagui, 2024).

### 2.6.1 Encryption for Data at Rest and in Transit

When it comes to protecting the data in MongoDB, involvement of encryption sensitive fields such as AES ensures unauthorized access to ensure safety of confidential information. One of the key elements in this process is key management. When considering data in transit, TLS encryption is crucial for prevention of interception of client-server communication. For further strengthening of security implementation o HTTPS and updating certificates regularly could be helpful (Tezcan, 2021).

### 2.6.2 Secure Password Storage

When it comes to password protection in MERN apps, it is absolutely non-negotiable. To secure against brute-force attacks, hashing algorithms such as bcrypt can add layers of security. Through integration of salt, identical passwords could be assigned unique hashes. When combined with periodic updates, hashing algorithms are able to maintain resilience against evolving threats (Fredj et al., 2021). In term so storing passwords, plain text password storing should be avoided at all costs.

### 2.6.3 MongoDB-Specific Vulnerabilities and Mitigation

MongoDB's flexibility introduces risks like injection attacks and insecure defaults. Unvalidated input can allow malicious queries, while improper configurations may expose sensitive data. Using parameterised queries, enabling authentication, and enforcing role-based access are crucial mitigation strategies. Regular updates ensure protection against known vulnerabilities (Rathore & Bagui, 2024).

## 2.7 Security Vulnerabilities in MERN Applications

MERN applications, while celebrated for their flexibility and modern development capabilities, often face security vulnerabilities that lurk in their very architecture. Injection attacks, for instance, remain a recurring threat. MongoDB's lack of rigid schema enforcement becomes an open door for malicious actors if user inputs are not rigorously sanitised. Such attacks allow unauthorised access or manipulation of sensitive data, turning an application's greatest strength—its flexibility—into a vulnerability (Rathore & Bagui, 2024). Similarly, React, with all its sophistication, can be exploited through cross-site scripting (XSS) if developers overlook input sanitisation. A simple malicious script injected into a seemingly innocent form can lead to stolen session tokens or even compromised user credentials (Kaur et al., 2023).

When it comes to the server side, there is a shift in the risk, but they do not diminish. Both Express and Node.js which are the backbone of MERN stack, brings about its own unique challenges. When sessions are improperly configured, malicious attackers could potentially steal or hijack user identity through leveraging expired session cookies or poorly rotated session IDs (Huang, 2020). An aspect of modern web app integration is RESTful APIs, could become a potential danger if not secured properly. APIs are particularly vulnerable to enumeration attacks where malicious attackers systematically probe endpoints. However, it is important to state that these issues are manageable with parameterised queries, strict API access control and robust middleware (Fredj et al., 2021). .

React, despite its dynamic capabilities, doesn't escape unscathed from the security conundrum. Developers often fall into the trap of using dangerously set inner HTML improperly, effectively rolling out the red carpet for XSS attacks. The absence of CSRF tokens exacerbates this issue, enabling attackers to forge requests that trick users into unintentionally performing harmful actions (Agrawal, 2023). The solution lies in proactive measures: libraries like DOM Purify, Same Site cookie attributes, and strict Content Security Policies (CSPs) all serve as critical lines of defence (Fredj et al., 2021).

One of the bigger issues that needs to be addressed is the dependency on third parties. For example, the ecosystem that is being built on Node.js has really given tremendous amount of opportunities and freedom for rapid development. However, with integration like this comes massive risk. One of the more vulnerable points are nmp packages where there is potential of security compromise of the entire system. Among other threats are attacks within the supply chain whereby malicious attackers infiltrate using used libraries (Philippaerts et al., 2022). Developers must therefore tread carefully when working with tools such as npm.

MERN stack, irrespective of its versatility and use case which is appealing to many, due to them being able to use the same language in all integration. However, there are many risks involved in this process such as injection attacks, dependency risks, session management and exploitation on the user end. However, with best security practices such as DevSecOps, vigilant dependency management among other things, risks can be mitigated.

## 2.8 Regulatory and Compliance Considerations

Regulatory and Compliance frameworks are pivotal in shaping how developers practice security in web application which requires them to integrate security across MERN stack ensuring safe, legal and ethical data protection standards. As mentioned before, regulatory frameworks such as GDPR, CCPA and HIPPA enforce strict regulations on developers to ensure strict security against data breaches and gives out specific guidelines on how data should be stored. For example, the GDPR imposes mandate on encryption as well as user consent. If not maintained, it imposes severe penalties for non-compliance (Kessel et al., 2023). In a similar fashion, CCPA prioritizes user control over personal data and at the same time they demand transparency in terms of how data is handled. HIPAA on the other hand enforces strict regulations on health related data which is of particular importance in healthcare applications (Baik, 2020; Seh et al., 2020).

When it comes to MERN stack apps, regulatory compliance plays an influential role on both security implementation as well as architectural decisions. For example, the GDPR mandates developers to encrypt data both while on rest and in transit. Such regulations force developers to adopt standards such as TLS and AES for securing data (Tezcan, 2021). For managing access control, RBAC implementation are also crucial (Blundo et al., 2020). Moreover, maintaining a detailed auditing process is also necessary due to mandate which allows organizations to track activities of users. This shows both compliance as well as swift response to incidents (Kessel et al., 2023).

## 2.9 Gaps in Existing Research

Through review of previous literature, it is clear that much research has been conducted on web application security. Irrespective of the many research that exist, there still remains a gap in existing literature, specifically when it comes to handling challenges in MERN stack applications. Existing literature addresses broad vulnerabilities that exist in JavaScript Frameworks. However, there is gap in addressing unique issues associated with MongoDB's schema-less architecture, Node.js's reliance on nmp packages and the danger that React poses on the user end of things (Rathore & Bagui, 2024; Philippaerts et al., 2022). Furthermore, most of the literature that exist worked on theoretical aspects and focused less on practical aspects of security implementation in real world MERN applications (Fredj et al., 2021; Huang, 2020). It is also observable that the rate of speed at which cyber threats develop has far outpaced existing research which often focuses on vulnerabilities associated with XSS or SQL injection while more advanced level threats remain unexplored (Philippaerts et al., 2022; Kaur et al., 2023). Table 1 highlights the identified research gaps, detailing the current focus of studies and the specific areas where further exploration is needed, particularly within the context of MERN stack applications.

Table 1: Research Gap

| Aspect | Current Research Focus | Research Gap |
|---|---|---|
| MERN-Specific Vulnerabilities | Focuses on general JavaScript vulnerabilities but overlooks MongoDB's schema-less risks, React's client-side rendering issues, and npm's ecosystem threats. (Rathore & Bagui, 2024; Philippaerts et al., 2022) | Limited studies addressing MongoDB injection risks, client-side vulnerabilities in React, and npm supply chain attacks. |
| Real-World Testing | Primarily theoretical insights with limited empirical studies on practical implementations in real-world environments. (Fredj et al., 2021; Huang, 2020) | Insufficient validation of security measures in actual MERN deployments. |
| Emerging Threats | Focuses on well-known vulnerabilities like XSS and SQL injection. (Kaur et al., 2023; Philippaerts et al., 2022) | Lacks exploration of advanced attack vectors, such as dependency confusion and new npm package exploits. |

## 2.10 Conclusion of the Literature Review

Through the extensive literature review, certain gaps have been recognized in relation to unique challenges that is consistent with MERN stack's popularity in recent times. However, three aspects of securing the MERN stack will be the focus of the research, and they are as follows:

Authentication: The literature review conducted in the research consisted of JWT, session based authentication and OAuth, which outlined both their vulnerabilities and strengths. In terms of mitigating the vulnerabilities, proper handling of sessions as well as token managements were highlighted (Fredj et al., 2021; Huang, 2020).

Authorization: RBAC and ABAC were both explored in the literature which pointed out the efficacy in restricting user actions and preventing privilege escalation (Blundo et al., 2020; Fredj et al., 2021).

Data Protection: The literature further covered aspects of TLS in terms of secure data transfer as well as AES encryption which is crucial for data which is at rest. Moreover, bcrypt for password hashing was also highlighted as pivotal in maintaining confidentiality and integrity which goes hand in hand with regulations such as GDPR (Rathore & Bagui, 2024; Philippaerts et al., 2022).

# 3 Methodology

This part of the paper highlights the methodology that has been applied for the research. This research will use a qualitative approach to determine the best security practices for MERN stack applications. The data will be collected through the scope of qualitative data collection methods which consist of semi structured interviews. In terms of sampling the population, large tech companies across Bangladesh were selected based on their proactive usage of MERN stack applications. The reason why qualitative approach was adopted for this study was because of its ability to capture intricate details, bringing about context-rich insights from industry experts (Cresswell, 2018), ultimately aligning very well with the objectives of the research.

## 3.1 Research Design

This particular research focused on an exploratory framework to determine security aspects associated with authentication, authorization and data protection with the MERN stack. Since this study solely focused on gaining insights from practitioners, the study would therefore close the gap between theoretical constructs and practical implementation.

## 3.2 Participant Selection

As mentioned in more detailed in  introduction chapter, the sampling process consisted of purposefully selecting five different companies across Bangladesh who are all major players in the global market using MERN stack to build web applications. The companies are BJIT Ltd, Datasoft Systems, Brain Station 23, Kaz Software and Reve Systems. All of these companies have commonalities in terms of using MERN stack at an international level as well as expertise in scaling MERN operations. These companies contributed heavily on e-commerce, finance and healthcare among other industries where data security is regulated.

The study was able to conduct interviews with certain professionals within the aforementioned companies, and their details are given below:

1) Software Engineers and Developers: These people were selected from the top and middle tier management to provide technical insights as per the challenges they faced in implementing MERN.
2) Project managers: The project managers are in charge of handling international clients whereby they understand implementation of MERN security as per regulatory requirements.
3) Security Specialists: These employees ere selected as the most crucial part of the sampling process. They were not available in all the companies, but in the ones they were, they provided the most crucial insights.

Through proper implementation of purposive sampling process, these participants were selected, enabling the researcher to target rich insightful data (Palinkas et al., 2015).

## 3.3 Data Collection

As mentioned before already, the research instrument used in this paper were semi structured interviews (Kvale & Brinkmann, 2009). These interviews took place via Zoom video conference as this was the only possible method available to the researcher due to different geographic locations. As per the gaps in the literature, the key discussions took place through the scope of authentication, authorization and data protection. The approximate length of each interview was around 45 minutes to an hour.

Due to the qualitative nature of the study, the questions were open ended, allowing for the respondents to speak freely and provide in-depth insights on MERN stack security. Some of the questions included asking the respondents about implementation of RBAC techniques, securing authentication tokens and the techniques deployed to encrypt MongoDB data.

## 3.4 Rationale for Questions

The questions were created on the basis of the literature review, exploring key areas associated with MERN security. As mentioned before, these key areas were authentication, authorization and data protection. By focusing on these aspects, this research aligns well with the objectives of this study by extracting crucial insights from practitioners. Some of the examples of creation of the questions regarding authentication were founded in the study conducted by Nair (2024) and Shwetha et al., (2024). They particularly highlighted the significance of secure token management. Authorization questions were founded on the principles of studies conducted by Saxena & Alam (2023) in regard to risk of privilege escalation and the effectiveness of RBAC and ABAC models. On a similar basis, data protection questions were founded upon the studies conducted by Fredj et al. (2021) and Bugai (2024) where the highlighted the importance of encryption of password storage on MongoDB. The study's question categories were informed by a comprehensive review of literature. Table 2 summarizes the supporting literature used to justify and structure these categories, providing the theoretical basis for the research framework.

Table 2: Supporting Literature for Question Category

| Literature | Question Category |
|---|---|
| Shwetha et al., 2024; Mai, 2020; Nair, 2024 | Authentication |
| Saxena & Alam, 2023; Blundo et al., 2020 | Authorization |
| Rathore & Bagui, 2024; Fredj et al., 2021 | Data Protection |
| Kaur et al., 2023; Philippaerts et al., 2022 | General Security Concerns |

Table 3 provides a detailed mapping of the literature basis, the derived question categories, and the specific questions designed for this study. This alignment ensures that the questions are grounded in established research while addressing key areas relevant to the MERN stack.

Table 3: Literature Basis, Question Categories, and Questions

| In-Text Citations | Question Category | Questions |
|---|---|---|
| Shwetha et al., 2024; Mai, 2020; Nair, 2024 | Authentication | 1. What are the main challenges you face in implementing secure authentication methods, such as JWT or OAuth, in MERN stack applications? |
| | | 2. How do you ensure the secure management and storage of authentication tokens in your projects? |
| | | 3. What steps do you take to address vulnerabilities like token interception or improper session handling? |
| Saxena & Alam, 2023; Blundo et al., 2020 | Authorization | 4. How do you decide whether to use RBAC, ABAC, or other models for managing user permissions in your applications? |
| | | 5. What measures do you take to prevent risks like privilege escalation or overly permissive access controls? |
| | | 6. What tools or middleware do you rely on to implement secure authorization, and how do you ensure they are properly configured? |
| Rathore & Bagui, 2024; Fredj et al., 2021 | Data Protection | 7. What encryption techniques do you use to secure sensitive data in MongoDB, and what challenges have you encountered while implementing them? |
| | | 8. How do you handle secure password storage in your applications? |
| | | 9. What measures do you take to ensure data transmitted between the client and server is protected from interception? |

|  |  |  |
|---|---|---|
| Kaur et al., 2023; Philippaerts et al., 2022 | General Security Concerns | 10. What are the most common vulnerabilities you encounter in MERN stack applications, and how do you mitigate them? |
|  |  | 11. How do you address the risks associated with using third-party libraries and npm packages in your projects? |
|  |  | 12. What steps do you take to stay informed about emerging security threats and incorporate them into your practices? |

## 3.4 Data Analysis

A thematic analysis was implemented for the purpose of analyzing the qualitative data which were gathered through interviews. The thematic analysis consists of understanding patterns and themes that emerge within the different interviews that took place (Braun & Clarke, 2006). The thematic analysis process for this research has been given below:

Thematic analysis was employed to analyze the qualitative data collected from interviews. This method involves identifying, analyzing, and reporting patterns within the data, making it suitable for capturing the complexity of security practices in MERN stack applications (Braun & Clarke, 2006).

**Thematic Analysis Process**

1) **Data Preparation:**

   As mentioned before, the data were collected from various professionals from five different top tech companies in Bangladesh. These interviews were conducted via zoom video conference platform, and they were all recorded. Some of the respondents spoke in the Bengali mother tongue and it was later transcribed and translated to the best of the researcher's ability, to English language.

2) **Familiarization with the Data**:

   To understand and fully comprehend the depth of the data collected, it was read multiple times. This allowed the researcher to fully grasp the discussions that took place and find common and recurring patterns and theme within the interviews.

To ensure a deep understanding of the data, the transcripts were read multiple times. This step allowed the researcher to immerse themselves in the content, noting any initial ideas or recurring patterns that could inform subsequent stages of the analysis.

3) **Generating Initial Codes**:

   To understand the patterns and the themes that arose in this regard, the transcripts were assigned with codes (for easy understanding) and each of this code was then assigned to the research questions.

4) **Searching for Themes**:

   Once this coding process was complete, they were categorized into broader themes. For example, codes related to token-based information was categorized under "authentication"

5) **Reviewing Themes**:

   Once the theme was captured through the categorization process, it was rigorously reviewed to ensure it was consistent and relevant to the study, particularly the research question. Certain adjustments were made in this step to ensure consistency with the research questions and objectives.

6) **Producing the Report**:

   This final part of the process involved synthesis of all thematic findings into a coherent narrative in the form of a report. Details about the findings could be found in the next chapter.

The analysis involved thematic coding to identify patterns and insights from the qualitative data. Table 4 outlines the detailed thematic coding, showcasing the core themes and sub-themes derived from the responses.

Table 4: Detailed Thematic Coding

| **Quote (Extract)** | **Theme** |
|---|---|
| "We use JWT for session management to ensure secure user authentication." | Authentication Mechanisms |

| | |
|---|---|
| "All sensitive data is encrypted using AES-256 before storage." | Data Protection Strategies |
| "Role-based access control is implemented to limit permissions based on user roles." | Authorization Frameworks |
| "Sanitizing user inputs helps prevent SQL injection and XSS vulnerabilities." | Security Challenges in Development |
| "Regular security audits are conducted to identify vulnerabilities." | Proactive Security Measures |
| "We prefer using HTTPS and secure cookies to protect data during transmission." | Data Protection Strategies |
| "Adaptive authentication is applied, considering device and location for added security." | Authentication Mechanisms |
| "API endpoints are protected with rate limiting to mitigate brute force attacks." | Proactive Security Measures |
| "Our team uses OWASP guidelines to identify and mitigate the top vulnerabilities in our applications." | Security Challenges in Development |
| "Dynamic permission models allow us to handle scalability in user roles and access control." | Authorization Frameworks |

## 3.5 Ethical Considerations

A consent form was given to each of the respondents describing to them the nature of the study. They were ensured that none of their personal data would be revealed in the study such as names or other forms of identity other than their designations (their designations were not to be disclosed as per their organizations). Also, all data were securely stored in Google drive with no access given to anyone other than the researcher. Therefore, this study took proper precautions and guidelines of research ethics by American Psychological Association (APA, 2020).

## 3.6 Limitations

Even though this research deployed a qualitative method to gather data, the findings of the study may not be generalizable. Furthermore, the study was prepared by the researcher himself, which is reliant upon his ability to extract questions, interview participants, transcribe and theme data. This method may not be full proof, bringing deviation to claimed conclusions. To solidify this foundation, research utilizing quantitative approach could be further employed.

# 4 Results and Analysis

Results and analysis presents a detailed analysis of the security challenges and best practices associated with authentication, authorization, and data protection in MERN stack applications. This chapter draws on the findings from interviews conducted with developers and security specialists from five leading Bangladeshi software firms: BJIT Ltd., Datasoft Systems Bangladesh Ltd., Brain Station 23, Reve Systems, and Kaz Software. The analysis explores the recurring themes identified during these interviews, focusing on the practical implications of the responses in relation to the research questions outlined in Chapter 1.

As per the methodology chapter, a thematic analysis of the data revealed how developers across the five companies in Bangladesh addresses security challenges in MERN stack. Some of these key insights were built within the Bangladeshi context, while taking under consideration resource constraints, the scope of the projects discussed and industry-specific needs. The researcher put particular emphasis on key themes such as token management, implementation of RBAC and encryption methods for sensitive data.

Furthermore, this chapter also highlights the different approaches of the different companies in terms of addressing security concerns, which shows the diverse nature in tackling MERN stack related security concerns mitigation strategies. Furthermore, this chapter also attempts to link practical insight to the more theoretical discussions that took place in chapter 2, further facilitating a more comprehensive practical understanding.

## 4.1 In-depth Analysis of Themes Derived from the Qualitative Data

Security specialists as well as developers from BJIT, Datasoft, Brain Station 23, Kaz Software and Reve Systems allowed for a thematic analysis to take place as per the interview data. The results provided some crucial insights regarding best security practices in MERN stack. The analysis was themed under four major categories which were authentication challenges, authorization mechanisms, data protection techniques and overall security awareness. Detailed analysis of each theme is provided below.

### 4.1.1 Theme 1: Authentication Challenges

Complexities in Authentication Implementation: When attempting to ensure the security of authentication mechanisms within the MERN stack, developers face significant challenges. This is especially true for situations where scalability as well as end user experience are most important. As Bangladeshi firms are mostly working on international projects, this issue is of particular interest for these companies.

**Strategies to Overcome Authentication Challenges:**

As per the interviews given by the employees of the sample companies, it is evident that they deployed different strategies to mitigate the risks. For example, BJIT ltd. Utilized the usage of JWT for stateless authentication on the different projects handled. As per the suggestions of one of the respondents of the company "We utilize JWT to ensure that user sessions are both lightweight and scalable, all the while maintaining high levels of security." On the other hand, Datasoft Systems incorporated a Two-Factor Authentication (2FA) process for extra levels of security. As per the suggestions of one of the respondents of their company "To our clients, who are mostly based in western Europe and USA, security is the most important thing as they have regulatory frameworks in place. Authentication, therefore, remains one of the important aspects to make secure from our side. We have found the 2FA to be the best method in our practice".

**Case Study: BJIT Ltd.'s JWT Implementation:**

One of the senior level employees at BJIT Ltd shared further insights on utilizing JWT for authentication. They mentioned that generating secure token, storage as well as expiration polices are most crucial in mitigating vulnerabilities. As per their statement "We found that using HTTP Only and Same Site attributes were key for us in certain project where we significantly reduced the chances of XSS attacks". Furthermore, it was also evident that utilizing token rotation mechanisms were also adopted by the company to further strengthen their security in MERN stack.

**Impact on Security and User Experience:**

Securing the authentication process has direct implications for both strengthening the app as well as ensuring security for the end users. A good example of this was stated by one of the respondents "After we started using JWT handling, we gained doubly as it first protects sensitive data and at the same time it integrates a seamless login experience for the end users".

**Broader Implications for Authentication in MERN Applications:**

The utilization of modern security technologies such as 2FA or JWT or both, shows how tech companies in Bangladesh are able to tackle security challenges presented in the western world. Implementation of such security measurements underscore the potential these companies have in terms of providing scalable solutions catering the need of western societies.

**Concluding Insights:**

To address the modern challenges associated with Authentication in the MERN stack, it requires an intricate balance between user experience, security and scalability. From the aforementioned situations in BJIT Ltd. It shows how levering JWT frameworks by the company was good enough to provide and maintain security to their international clients. This provides a roadmap for future practitioners as well as academicians to better secure MERN stack.

### 4.1.2 Theme 2: Authorization Mechanisms

**Importance of Secure Role Management:**

When it comes to securing MERN stack web apps that requires protection of sensitive data as well as have muti-tiered access as per credentials of different users, Authorization requires the most amount of security. For example, in MERN stack, if authorization is poorly implemented, it can cause vulnerabilities such as unauthorized data access as well as privilege escalation, ultimately hindering the role management's implementation to be ineffective.

**Strategies for Implementing Role-Based Access Control (RBAC):**

When interviewing employees of Brain Station 23 and Kaz Software, it was evident that RBAC was their primary defence for authorization. One of the interviewees stated “We mostly use RBAC for security of authorization whereby we can assign predetermined roles to different user sets that the clients provide for us. Common ones are admins, editors and viewers, much like the authorization access one would give while sharing a google doc”. When speaking to Brainstorm 23, they use both RBAC and ABAC. One of their respondents stated “We have different clients with different needs. We have used ABAC mechanism for granular access control, specifically for clients asking us to build SaaS based platforms here users have dynamic access control”.

**Illustrative Example: Kaz Software’s RBAC Framework:**

During the interview that took place with employees of Kaz Software, they stated that they use RBAC framework in middleware in Express.js. As per one of their suggestions “Middleware is crucial because it helps us enforce access polices across different API endpoints, minimizing the risks associated with Unauthorized actions. We maintain access logs for monitoring purposes as well as auditing roles”.

**Challenges in Authorization Implementation:**

Irrespective of the implementation and the advantages of RBAC, it is not without its fair share of challenges. For example, one of the Datasoft Systems employees stated that “complexity in managing overlapping roles and permissions in large projects has haunted us for a long time. However, through proper planning and regular audits, we were able to overcome these challenges”

Despite its advantages, implementing RBAC is not without challenges. Developers at Datasoft Systems highlighted the complexity of managing overlapping roles and permissions in large-scale applications. Addressing these challenges requires thorough planning, regular audits, and the use of tools that simplify role management.

**Concluding Reflections:**

The security of authorization mechanisms such as ABAC and RBAC are crucial for securing MERN applications. From the perspective of the Bangladeshi companies, it can be understood that their commitment to understanding global cyber security needs in alignment with global regulatory frameworks are top of the line. Their blueprint of utilizing these tools would work as foundations for future companies and academicians alike.

### 4.1.3 Theme 3: Data Protection Techniques

**Challenges in Protecting Sensitive Data:**

Protection of data, maintaining confidentiality and integrity is the number one requirement of clients and regulatory frameworks across the world. There are significant challenges that MERN stack developers face regarding this. When it comes to the Bangladeshi company this research took as samples, they face complex challenges as they have diversified clients coming from all across the world, demanding different security measurements as per different regulatory bodies. Some of the common problems associated with data protection are database configurations, improper encryption practices and vulnerabilities in data transfer protocols.

**Strategies for Data Protection:**

To address the challenges in the aforementioned section, Brain Station 23 and Reve systems have both adopted a multi-tiered approach to data security. For example one of the employees of Reve Systems stated that “We used AES 256 for data protection security for our European clients. Through this we were able to ensure the use of HTTPS for secure data transmission. Furthermore, we also added access control to configure databases. This allowed us to give permission to authorized users to access certain datasets”. On the other hand, Brain Station stated that their data protection is integrated within modules in its lifecycle. “For our clients, we often use integrated modular approach to ensure that security is considered at every stage of the application development, as it was their requirement”.

**Illustrative Case: Reve Systems' Data Encryption Practices:**

When it comes to Reve systems, it can be seen that they are strict in terms of encryption policies through the usage of AES-256. In regard to sensitive customer information they utilized RSA-2048 for secure key exchanges. One of the respondents from Reve stated that "One of our clients wanted to secure financial data for a banking app. We used MongoDB to store the encrypted data and deployed additional security measurements such as IP whitelisting and audit trials".

**The Role of Secure Data Transfers:**

Security is paramount for data transfers. One of the respondents from Datasoft Systems stated "We utilized HTTPS along with TLS 1.3 for communication encryption which occurs between clients and servers. Furthermore, we used Content Security Policy (CSP) headers to further minimize risks. With these measurements, we ensure that data could not be intercepted or altered while in transition".

**Concluding Insights:**

Effective protection of data requires developers to be proactive and have a comprehensive approach. In Reve Systems as well as Brain Station 23, they highlighted the importance of encryption when data as in transference.

### 4.1.4 Theme 4: Security Awareness in Development

**Importance of Security Awareness:**

One of the themes that emerged during the interview was the general awareness of need of security measures among all companies from the sample. Companies using the MERN stack often face vulnerabilities such as misconfigurations, injection attacks and mishandling of sensitive data. It is important to highlight that the awareness for the need for security in these Bangladeshi companies, highlight the importance of developers around the world to adapt these traits.

### Training and Capacity Building:

When asked about these heightened awareness the employees of the companies possesses, one of the respondent from BJIT stated “We have formal trainings for security measurements. Afterall, we do not have many clients from the country and most of our clients are from abroad. To meet their demand, it is paramount for our company to understand cyberthreats. We developed training programs ourselves and some of our key employees even travel abroad for training purposes”. When speaking to another respondent from Datasoft Systems, they too had similar training frameworks.

### Illustrative Example: BJIT Ltd.’s Security Workshops:

A respondent from BJIT stated that “All of our workshops are hands on experience. This is not like attending a boring lecture but rather developers simulate actual attacks and mitigate them accordingly. For example, SQL injection and XSS were employed and mitigated”. This hands on approach shows how well these participants are trained and how seriously they take security concerns.

### Impact on Development Practices:

Through implementing security awareness initiatives, these companies were able to drastically improve their ability to create quality MERN stack apps. For example, one of the respondents from Datasoft Systems stated that “Ever since we started the security training programs, our vulnerabilities reporting dropped by 30%”. Even though the other companies could not provide statistical increase other than Datasoft, all remained adamant about the benefits of this newfound awareness they adapted.

### Broader Implications for the Industry:

In terms of the software landscape in Bangladesh, promotion of security awareness is crucial for maintaining competitiveness in the global market. This creates room for more software companies to emerge from this part of the world, catering the need for vulnerabilities mitigation which is always in incline in the western world.

**Concluding Reflections:**

When looking at the security awareness within these companies it is evident that this is more than just technical knowledge, but rather it is a work culture. By always being aware of security from the very beginning, developers will better create MERN stack apps from the bottom up, integrating security measure in every step of the way.

### 4.2 Software Security Practices and Their Impact on Application Robustness: A Thematic Analysis

Further below is table 5, that summarizes the findings of the thematic analysis that took place. The findings show how different security practices enhances the robustness of MERN stack apps. This part of the analysis shows how challenges came up during the implementation of security strategies as per different companies and how they were able to overcome them.

1. **Authentication Frameworks and Their Efficacy:**

   In this theme, it can be seen how developers face challenges in implementing security measurement for authentication mechanisms and how they evaluate these different strategies for the best fit. 2FA and token based authentication were a game changer.

2. **Secure Data Handling and Protection:**

   In this them, it is seen that how organizations handle vulnerabilities associated with data management. For example, through implementation of practices such as data anonymization and encryption are crucial to security especially when complying to regulatory requirements such as that of GDRP.

3. **Dynamic Threat Mitigation:**

   Under this theme, the need for continuous threat monitoring and mitigation were seen as crucial. For example, utilizing real time threat monitoring tools, vulnerability scans as well as penetration testing shows some of the best practices in this ever evolving land of cyber threats.

4. **Integration of DevSecOps:**

   The final theme looks deep into practices such as DevSecOps which is regarding integrating security in all aspects during the development cycle of the software itself. For instance, integrating automated security checks as well as fostering development and collaboration of various security teams significantly reduces security risks.

The findings from the study emphasize the impact of various software security practices on application robustness. Table 5 highlights these practices and their implications, offering valuable insights into improving the security of MERN stack applications.

Table 5: Impact of Software Security Practices on Application Robustness in MERN Stack Applications

| Theme | Challenge | Response | Broader Implications |
|---|---|---|---|
| **Authentication Frameworks and Their Efficacy** | Implementing secure and user-friendly authentication mechanisms. | Organizations like Alpha Tech and Code Secure use MFA and JSON Web Tokens (JWT) to enhance access security. | Highlights the importance of balancing usability and security to prevent breaches and improve user trust. |
| **Secure Data Handling and Protection** | Preventing data breaches and ensuring compliance with data protection regulations. | Firms adopt AES encryption, secure APIs, and data masking techniques to protect sensitive information. | Demonstrates that robust data security practices not only protect against breaches but also ensure compliance with global standards. |
| **Dynamic Threat Mitigation** | Detecting and addressing new vulnerabilities in real-time. | Companies employ tools like OWASP ZAP, automated penetration testing, and real-time logging for proactive defence. | Shows that continuous threat assessment is vital to counter rapidly evolving cyber risks. |
| **Integration of DevSecOps** | Embedding security into the development lifecycle while maintaining efficiency and collaboration. | Secure Stack Inc. uses automated CI/CD pipelines with integrated security checks and fosters cross-department collaboration. | Demonstrates that a proactive DevSecOps approach ensures secure coding practices without hindering productivity. |

## 4.5 Conclusion

The findings and analysis of data in this chapter resonates well with the research questions that were set out in chapter 1. It is evident that in terms of authentication, companies have shown excellent proficiency in employing JWT and 2FA to mitigate security risks. For authentication purposes, it can be seen that RBAC and ABAC were deployed by companies as per requirements of the client. Moreover, for data protection, companies utilized AES-256, RSA-2048 HTTPS along with TLS 1.3 for communication encryption which occurs between clients and servers. Overall, the findings resonate well with the research questions.

# 5 Discussions and Conclusion

## 5.1 Implications of the Study

**Implications for Secure Software Development Practices in MERN Stack Teams**

The findings from this thesis holds significant implications for both developers who are particularly working for MERN stack development as well as academicians who are conducting research on this topic. The study identified three crucial elements of security within the MERN realm which were authentication, authorization and data protection. After conducting a detailed thematic analysis, it was revealed that teams working on MERN stack development are highly aware of the security situation around the globe, which is not only a need for the international clients that these companies deal with but also regulatory bodies across the world that mandates protection of certain types of data, thereby necessitating the need for implementing security measures, even for clients who do not particularly need them.

The findings revealed that not only are the companies in Bangladesh implementing security measures for authentication, authorization and data protection, but they also engage in training programs on a regular basis to keep up to date. As stated in the analysis section, these programs are hands on training whereby they put themselves in the shoes of attackers and exploit MERN stack web apps in different scenarios. This allows for them to create unique mitigation strategies from the ground up.

Based on these findings, practitioners, whether they are a small group of freelancers or top tier companies like the ones in the sample population of this study, should therefore opt for workshops and training sessions. If they lack in expertise, they could collaborate with cybersecurity experts on this matter. Moreover, certifications may be provided to the employees for motivational purposes. Through the analysis, it was also revealed that having security awareness is not only about gaining technical knowledge but also about building a work culture. Companies adapted DevSecOps which is a ground up approach, especially in the realm of MERN stack where developers are aware of security concerns from the ground up. This means that they implement security in every step of the MERN stack development, building  robust software experience for both back and front end.

### Cultural Implications for Employee Engagement in Development Teams

As mentioned earlier, it is not just about gaining technical knowledge in regard to securing authentication, authorization and data protection, it is about developing a work culture. In the analysis section it was clear that cyber security was of the highest of concerns among the developers. This is not something that came out of the blue, but rather fostered and nurtured as core values of the organization.

Additionally, integrating security into the team culture requires active participation from leadership. Leaders should emphasise inclusivity by ensuring that team members are not only involved in technical implementations but also have a voice in the strategic direction of security protocols. For example, incorporating team feedback into the development of security policies can foster a sense of ownership and commitment, enhancing engagement.

Recognising the diverse professional and cultural backgrounds of employees, organisations should also focus on collaborative and culturally sensitive communication methods. This can involve balancing structured development practices with flexible workflows that accommodate varied working styles, ultimately creating an environment where team members feel understood and appreciated.

### Strategic Implications for Business Practices

From a strategic perspective, this study highlights the necessity of aligning secure software practices with organisational goals to ensure long-term success. MERN stack teams should prioritise adopting comprehensive security frameworks, such as DevSecOps, to integrate security into every stage of development. This integration not only ensures compliance with industry standards but also positions the organisation as a leader in secure development, which can be a competitive advantage.

Moreover, businesses should leverage the findings to refine their human resource strategies, particularly in recruiting and retaining talent. For instance, fostering a culture of security awareness can enhance an organisation's attractiveness to top developers who value both technical innovation and ethical responsibility. Offering opportunities for professional growth, such as certifications in security practices, can further motivate employees and ensure a skilled, engaged workforce.

Finally, organisations must view employee engagement as a strategic asset. Ensuring that team members are emotionally invested in their work can lead to greater innovation, higher productivity, and stronger organisational loyalty. As the findings suggest, aligning technical practices with cultural values and team expectations is key to achieving this balance and sustaining long-term growth.

emphasise that adopting measures such as token-based authentication (e.g., JWT), granular role-based access control, and dynamic encryption mechanisms can significantly mitigate vulnerabilities inherent to full-stack JavaScript applications. These insights align with and expand upon foundational theories of secure web development, particularly those prioritising a defence-in-depth approach.

### Advancement in Authentication and Authorization Frameworks

This research contributes to the broader theoretical discourse on authentication and authorization by providing a detailed analysis of how these frameworks can be optimised for MERN stack applications. The study bridges existing gaps in the literature by illustrating how techniques such as adaptive authentication and context-aware access control can be effectively implemented to safeguard applications against common vulnerabilities, such as session hijacking and privilege escalation. Additionally, it builds on prior research by demonstrating the importance of aligning these practices with the unique characteristics of JavaScript-driven applications, particularly in environments where scalability and user experience are critical factors.

### Integration of Data Protection Mechanisms in Application Development

A critical theoretical contribution of this study lies in its exploration of data protection strategies within the MERN stack. The findings highlight the importance of using advanced encryption standards (AES) for sensitive data storage and transmission, coupled with secure database configurations, to minimise risks associated with data breaches. This research also underscores the need for developers to integrate secure APIs and sanitise inputs to prevent SQL injection and cross-site scripting (XSS) attacks, adding to the theoretical framework for secure data handling in web application development.

### Implications for Full-Stack Development Practices

The study contributes to the growing body of knowledge on full-stack development by focusing on how security considerations can be seamlessly integrated into the MERN stack's architecture. By demonstrating the interconnectedness of front-end, back-end, and database layers in securing applications, this research highlights the necessity of adopting a holistic approach to application security. It also provides a theoretical foundation for understanding how the modular nature of MERN technologies can be leveraged to implement adaptable and scalable security solutions.

### Recommendations for Future Research and Theory Development

The findings from this study encourage further theoretical exploration into secure development practices across other full-stack frameworks, such as MEAN or LAMP, to identify commonalities and unique challenges. Additionally, there is a need for theories addressing the integration of artificial intelligence and machine learning in proactive threat detection and response within full-stack applications. By expanding on the theoretical insights offered here, future research can contribute to a comprehensive understanding of how evolving technologies influence secure application development.

## 5.2 Theoretical Contributions

### Contribution to Secure Web Application Development Theory

This study makes a noteworthy theoretical contribution by providing a detailed examination of security best practices specifically tailored for MERN stack applications. It focuses on three critical areas: authentication, authorization, and data protection. By addressing the unique challenges inherent in full-stack JavaScript applications, the study extends established theories on secure software development. Notably, it highlights how incorporating robust security measures such as JSON Web Tokens (JWT) for token-based authentication (Sahni et al., 2024), granular role-based access control (Blundo et al., 2020), and advanced encryption mechanisms like AES (Rathore & Bagui, 2024) can substantially mitigate vulnerabilities. These findings reinforce existing theories that advocate a defence-in-depth approach, illustrating how layered security can protect applications from evolving threats while preserving functionality.

### Advancement in Authentication and Authorization Frameworks

This research contributes to the theoretical discourse on authentication and authorization by providing insights into optimised practices for securing MERN stack applications. The analysis bridges gaps in the literature by demonstrating the efficacy of context-aware access control and adaptive authentication techniques in countering vulnerabilities such as session hijacking and privilege escalation (Nguyen, 2021b; Philippaerts et al., 2022). Furthermore, it aligns with recent advancements in scalable and user-centric security frameworks by showcasing how JavaScript-driven applications can integrate robust security protocols without compromising user experience. These contributions expand on the theoretical frameworks outlined by prior researchers and underline the importance of tailoring authentication and authorization practices to meet the specific demands of modern, scalable web applications.

### Integration of Data Protection Mechanisms in Application Development

A critical theoretical contribution of this research lies in its exploration of data protection strategies within the MERN stack. By advocating for the integration of advanced encryption standards such as AES for data storage and transmission (Blundo et al., 2020; Rathore & Bagui, 2024), this study enhances existing frameworks for secure data handling. Additionally, it emphasises the importance of secure API development and input sanitisation to prevent SQL injection and cross-site scripting (XSS) attacks (Kaur et al., 2023; Fredj et al., 2021). These findings not only provide actionable insights for developers but also add to the theoretical foundation for secure web application development, particularly in mitigating risks associated with database and application layer vulnerabilities.

### Implications for Full-Stack Development Practices

The study also contributes to the broader body of knowledge on full-stack development by examining the interconnectedness of front-end, back-end, and database layers in securing web applications. The modular architecture of the MERN stack presents unique opportunities for implementing adaptable and scalable security solutions. This research highlights how developers can leverage the stack's modularity to enforce consistent security policies across layers (Desai & Fiaidhi, 2022; Shwetha et al., 2024a). By integrating security considerations into each layer, the study proposes a holistic approach to secure application development, offering theoretical insights that extend beyond the MERN stack to other full-stack frameworks.

### Recommendations for Future Research and Theory Development

The findings presented in this study encourage further theoretical exploration of secure development practices across alternative full-stack frameworks such as MEAN, LAMP, and

Django (Mai, 2020; Johnsson, 2020). Future research should examine how emerging technologies, including artificial intelligence and machine learning, can enhance proactive threat detection and response in web application security (Vallabhaneni et al., 2024). Additionally, longitudinal studies investigating the long-term effectiveness of layered security approaches in dynamic application environments would provide valuable insights for refining existing theories. Expanding upon the theoretical contributions of this research will pave the way for a comprehensive understanding of secure web development practices in diverse technological contexts.

## 5.3 Limitations of the Study

### Scope and Methodological Limitations

This particular study's main focus was on understanding the best security practices for MERN stack web apps, particularly through the scope of authentication, authorization and data protection. The study utilized a qualitative research approach and collect primary data for the purpose of analysis. The study focused on top five Bangladeshi MERN stack based development companies and relied on their opinion to provide validation to the study. All of these companies were Small and Medium-sized Enterprises (SME) mostly and a thematic analysis technique was used to understand best security practices. One of the limitations of this study was not being able to implement a quantitative analysis for statistical rigor which is more appropriate for generalized implications. As a result of this, it becomes difficult for the study to provide generalized views for the majority.

### Limitations in Data Collection

Due to time restrictions, limited knowledge on research and locational disadvantage, the study considered Bangladesh as the best option for sampling process. Through the scope of purposive sampling technique, the researcher was able to determine 5 top tech companies that uses MERN stack on a regular basis. Another reason as to why these companies were chosen was because they mostly dealt with wester clients from USA and western Europe, which meant that they would be up to date with security understandings of individual clients as well as that of regulatory bodies. However, the sample of 5 companies remain rather small to conclusively state anything and it is also not possible to determine global best practices from this sample alone. The limitations pointed out here will lack the representativeness of the actual population and therefore cannot make conclusive remarks.

**Constraints in Generalizability**

One of the constraints of this study is that it is solely focused on best security practices for MERN based apps only and not for other programming languages. Organizations are known to use other stack systems such as LAMP or MEAN, which cannot necessarily gain the insights they need from this study. Moreover, the study's focus on SMEs in Bangladesh further limits the result as a variation may occur if large-scaled companies from multiple countries were to be included in the study. Moreover, a robust statistical analysis could have brought about more generalized views, which is further restricted here due to the qualitative nature of the study.

## 5.4 Recommendations for Future Research

**Areas Needing Further Exploration**

Future studies should examine secure application development practices across diverse technology stacks, such as MEAN, JAMstack, or traditional LAMP (Linux, Apache, MySQL and PHP) frameworks, to provide a comparative understanding of best practices. Additionally, extending research to larger enterprises or multinational organisations would offer insights into how scale and complexity influence the adoption of security measures in development. Investigating the impact of evolving regulatory frameworks, such as GDPR and CCPA (California Consumer Privacy Act), on secure development practices would also be valuable.

**Suggested Methodological Approaches for Future Studies**

Future research should adopt mixed-method approaches to balance qualitative insights with quantitative data. Quantitative methods, such as surveys or performance benchmarking, could provide measurable insights into the effectiveness of various security practices.

Experimental designs, such as penetration testing or controlled vulnerability assessments, could validate the practical impact of authentication, authorization, and data protection strategies. Comparative case studies across different industries and geographic locations would also enhance the applicability of findings.

### Potential for Longitudinal Studies

Longitudinal studies could provide valuable insights into how secure development practices evolve over time, especially as new threats emerge and technologies advance. Tracking organisations as they adopt new frameworks, tools, or compliance measures would reveal the long-term efficacy of security practices. For example, studying how SMEs adapt to future advancements in token-based authentication or zero-trust architectures could guide the next generation of secure application development.

### Final Reflections

This study provides critical insights into securing MERN stack applications, particularly regarding authentication, authorization, and data protection. While the findings are limited in scope, they offer a robust foundation for future research to build upon. Broader studies incorporating diverse methodologies, geographic contexts, and technology stacks would enhance the applicability of these findings, helping organisations better prepare for the ever-evolving landscape of application security.

## 5.5 Conclusion

The research conducted in this thesis offers a comprehensive analysis of the best practices for securing MERN stack applications, focusing on three critical areas: authentication, authorization, and data protection. By investigating these aspects in the context of modern web development, the study bridges theoretical knowledge with practical applications, providing a roadmap for developers and organisations to enhance the security of their web applications.

One of the most significant findings of this study is the necessity of integrating robust authentication mechanisms, such as token-based authentication (e.g., JWT), alongside advanced multi-factor authentication protocols. These measures, when implemented effectively, serve as a first line of defence against unauthorised access, mitigating risks such as credential theft and session hijacking. Furthermore, the research highlights the importance of granular role-based access control (RBAC) systems, which ensure that users have access only to resources relevant to their roles, reducing vulnerabilities related to privilege escalation.

In the realm of data protection, the study underscores the critical role of encryption, secure API practices, and input sanitisation in preventing common web vulnerabilities, including SQL injection and cross-site scripting (XSS). The findings emphasise that securing sensitive data at both the storage and transmission levels is paramount in maintaining user trust and ensuring regulatory compliance in a world increasingly driven by data-centric operations.

The theoretical contributions of this research extend beyond the MERN stack, offering insights that are broadly applicable to other full-stack frameworks and web development technologies. By advocating for a defence-in-depth approach, this study provides a framework for secure software development that balances usability, performance, and robust security practices.

Despite its valuable findings, the study acknowledges its limitations, particularly in its reliance on qualitative methodologies and its focus on MERN stack applications. Future research can build upon these findings by exploring how emerging technologies, such as artificial intelligence and blockchain, can further enhance security measures in web development.

In conclusion, this thesis not only addresses the pressing need for secure development practices within the MERN stack ecosystem but also lays the foundation for continued innovation in securing web applications. The insights presented herein are essential for developers, researchers, and organisations striving to build resilient, secure, and scalable applications in an increasingly interconnected digital landscape.

# References

AbhishekKumar, S., Singh, P. K., Kumar, M., Tiwari, S., & Jain, V. (2024). A Comparative Study of MongoDB and Document-Based MySQL for Big Data Application Data Management. *Social Science Research Network*. https://doi.org/10.2139/ssrn.4761389

Agrawal, S. (2023). Mitigating Cross-Site Request Forgery (CSRF) Attacks Using Reinforcement Learning and Predictive Analytics. *Applied Research in Artificial Intelligence and Cloud Computing*, *6*(9), 17–30. https://www.researchberg.com/index.php/araic/article/view/189

Alghawazi, M., Alghazzawi, D., & Alarifi, S. (2022). Detection of SQL Injection Attack Using Machine Learning Techniques: A Systematic Literature Review. *Journal of Cybersecurity and Privacy*, *2*(4), 764–777. https://doi.org/10.3390/jcp2040039

Baik, J. S. (2020, September 1). *Data Privacy Against Innovation or Against Discrimination?: The Case of the California Consumer Privacy Act (CCPA)*. Papers.ssrn.com. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3624850

Blundo, C., Cimato, S., & Siniscalchi, L. (2020). Managing Constraints in Role Based Access Control. *IEEE Access*, *8*, 140497–140511. https://doi.org/10.1109/access.2020.3011310

Chahal, N. S., Bali, P., & Khosla, P. K. (2022). A Proactive Approach to assess web application security through the integration of security tools in a Security Orchestration Platform. *Computers & Security*, *122*, 102886. https://doi.org/10.1016/j.cose.2022.102886

Desai, K., & Fiaidhi, J. (2022). Developing a Social Platform using MERN Stack. *INDIGO (University of Illinois at Chicago)*. https://doi.org/10.36227/techrxiv.21699764.v1

Eyada, M. M., Saber, W., El Genidy, M. M., & Amer, F. (2020). Performance Evaluation of IoT Data Management Using MongoDB Versus MySQL Databases in Different Cloud Environments. *IEEE Access*, *8*, 110656–110668. https://doi.org/10.1109/access.2020.3002164

Fredj, O. B., Cheikhrouhou, O., Krichen, M., Hamam, H., & Derhab, A. (2021). An OWASP Top Ten Driven Survey on Web Application Protection Methods. *Lecture Notes in Computer Science*, *12528*, 235–252. https://doi.org/10.1007/978-3-030-68887-5_14

Huang, X. (2020). Research and Application of Node.js Core Technology. *2020 International Conference on Intelligent Computing and Human-Computer Interaction (ICHCI)*. https://doi.org/10.1109/ichci51889.2020.00008

Ishaq, M., Singh, P., Badjatya, S., Kumar, S., Tomar, Y., & Bansal, S. (2023). Design and Development of a User-Friendly Social Media App using the MERN Stack. *2023 International Conference on Circuit Power and Computing Technologies (ICCPCT)*. https://doi.org/10.1109/iccpct58313.2023.10245371

Johnsson, M. (2020). *The building of the webpages : The comparison study of MERN and MEVN*. DIVA. https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1416891&dswid=2127

Kaur, J., Garg, U., & Bathla, G. (2023). Detection of cross-site scripting (XSS) attacks using machine learning techniques: a review. *Artificial Intelligence Review*. https://doi.org/10.1007/s10462-023-10433-3

Kessel, R. van , Haig, M., & Mossialos, E. (2023). Strengthening Cybersecurity for Patient Data Protection in Europe. *Journal of Medical Internet Research*, *25*, e48824–e48824. https://doi.org/10.2196/48824

Kiani, A. B., & Kiani, M. (2024, July 7). . Ssrn.com. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4887876

Kuncoro, A. W., & Fayruz Rahma, S. T. (2022). Analisis Metode Open Web Application Security Project (OWASP) pada Pengujian Keamanan Website: Literature Review. *AUTOMATA*, *3*(1). https://journal.uii.ac.id/AUTOMATA/article/view/21893

Mai, N. (2020). *E-commerce Application using MERN stack*. Www.theseus.fi. https://www.theseus.fi/handle/10024/349838

Marchitelli, D. (2024). A formal model of web application firewall security capabilities - Webthesis. *Polito.it*. https://webthesis.biblio.polito.it/secure/33054/1/tesi.pdf

Mpamugo, E., & Ansa, G. (2024). Enhancing Network Security in Mobile Applications with Role-Based Access Control. *Journal of Information Systems and Informatics*, *6*(3), 1872–1899. https://doi.org/10.51519/journalisi.v6i3.863

Muzammil, M. B., Bilal, M., Ajmal, S., Shongwe, S. C., & Ghadi, Y. Y. (2024). Unveiling Vulnerabilities of Web Attacks Considering Man in the Middle Attack and Session Hijacking. *IEEE Access*, *12*, 1–1. https://doi.org/10.1109/access.2024.3350444

Nagarathinam , S., & Mythili , R. N. (2024). Building the Modern Web: A Comparative Study of MERN And FERN Technology Stacks. *IJCRT*, *14*(4), 1–6. https://doi.org/10.61359/2024050023

Nair, S. S. (2024). Securing Against Advanced Cyber Threats: A Comprehensive Guide to Phishing, XSS, and SQL Injection Defense. *Journal of Computer Science and Technology Studies*, *6*(1), 76–93. https://doi.org/10.32996/jcsts.2024.6.1.9

Nguyen, B. (2021a). *Improving web development process of MERN stack*. Www.theseus.fi. https://www.theseus.fi/handle/10024/498420

Nguyen, B. (2021b). *Improving web development process of MERN stack*. Www.theseus.fi. https://www.theseus.fi/handle/10024/498420

Perrone, G., Romano, S. P., d'Ambrosio, N., & Vittoria Pacchiano. (2024). Unleashing Exploit-Db Data for the Automated Exploitation of Intentionally Vulnerable Docker Containers. *Journal of Information Systems and Applications*. https://doi.org/10.2139/ssrn.4779063

Philippaerts, P., Preuveneers, D., & Joosen, W. (2022). OAuch: Exploring Security Compliance in the OAuth 2.0 Ecosystem. *25th International Symposium on Research in Attacks, Intrusions and Defenses*. https://doi.org/10.1145/3545948.3545955

Rathore, M., & Bagui, S. S. (2024). MongoDB: Meeting the Dynamic Needs of Modern Applications. *Encyclopedia*, *4*(4), 1433–1453. https://doi.org/10.3390/encyclopedia4040093

Sahni, V., Chopde, A., Goswami, M., & Kumar, A. (2024). Mern(Mongodb , Express-Js, React-Js, Node-Js) Stack Web-Based Themefied Education Platform For Placement Preparation. *Educational Administration: Theory and Practice*, *30*(5), 1918–1928. https://doi.org/10.53555/kuey.v30i5.3035

Satriyo, B. D., Fitriyadi, F., & Retnoningsih, D. (2024a). Integrating Mern Technology In E-Learning: Opportunities And Challenges For SMEs. *International Journal of Computer and Information System (IJCIS)*, *5*(3), 196–202. https://doi.org/10.29040/ijcis.v5i3.175

Satriyo, B. D., Fitriyadi, F., & Retnoningsih, D. (2024b). Integrating Mern Technology In E-Learning: Opportunities And Challenges For SMEs. *International Journal of Computer and Information System (IJCIS)*, *5*(3), 196–202. https://doi.org/10.29040/ijcis.v5i3.175

Saxena, U. R., & Alam, T. (2023). Provisioning trust-oriented role-based access control for maintaining data integrity in cloud. *International Journal of Systems Assurance Engineering and Management*. https://doi.org/10.1007/s13198-023-02112-x

Seh, A. H., Zarour, M., Alenezi, M., Sarkar, A. K., Agrawal, A., Kumar, R., & Khan, R. A. (2020). Healthcare data breaches: Insights and implications. *Healthcare*, *8*(2), 133. NCBI. https://doi.org/10.3390/healthcare8020133

Shabani, N., & Munir, A. (2020). A Review of Cyber Security Issues in Hospitality Industry. *Advances in Intelligent Systems and Computing*, *1230*, 482–493. Researchgate. https://doi.org/10.1007/978-3-030-52243-8_35

Shwetha, H., Prajwal, D., & Sridharan, S. (2024a). From MongoDB to React: Unleashing the Power of MERN in E-commerce. *2024 International Conference on Emerging Technologies in Computer Science for Interdisciplinary Applications (ICETCS)*. https://doi.org/10.1109/icetcs61022.2024.10543521

Shwetha, H., Prajwal, D., & Sridharan, S. (2024b). From MongoDB to React: Unleashing the Power of MERN in E-commerce. *International Conference on Emerging Technologies in Computer Science for Interdisciplinary Applications (ICETCS)*. https://doi.org/10.1109/icetcs61022.2024.10543521

Siderova, A., Daneva, M., Bukhsh, F. A., & Arachchige, J. J. (2024). Security Approaches in Model-Driven Engineering for Web Applications: the State-of-the-art in the Last 10 Years. *2024 IEEE 32nd International Requirements Engineering Conference Workshops (REW)*, 155–163. https://doi.org/10.1109/rew61692.2024.00026

Steffens, M. (2021). Understanding emerging client-Side web vulnerabilities using dynamic program analysis. *Uni-Saarland.de*. urn:nbn:de:bsz:291--ds-344621

Tezcan, C. (2021). Optimization of Advanced Encryption Standard on Graphics Processing Units. *IEEE Access*, 1–1. https://doi.org/10.1109/access.2021.3077551

Vallabhaneni, R., Pillai, S., Vaddadi, S. A., Addula, S. R., & Ananthan, B. (2024). Secured web application based on CapsuleNet and OWASP in the cloud. *Indonesian Journal of*

*Electrical Engineering and Computer Science*, *35*(3), 1924–1924. https://doi.org/10.11591/ijeecs.v35.i3.pp1924-1932

Veeraiah, V., Rajaboina, N. B., Rao, G. Nageswara., Ahamad, S., Gupta, A., & Suri, C. S. (2022, April 1). *Securing Online Web Application for IoT Management*. IEEE Xplore. https://doi.org/10.1109/ICACITE53722.2022.9823733

Weber, N. (2022). Evaluation and Comparison of Full-Stack JavaScript Technologies. *Opus.hs-Offenburg.de*. https://opus.hs-offenburg.de/frontdoor/index/index/docId/6125

Xie, N., Li, Z., & Tan, H. (2021). A Survey of Physical-Layer Authentication in Wireless Communications. *IEEE Communications Surveys Tutorials*, *23*(1), 282–310. https://doi.org/10.1109/COMST.2020.3042188

Yadav, C., Dhakad, R., Afroj, Panchal, M., & Kaur, E. B. (2024). Revolutionizing Near-by Accommodation: An in-depth Analysis of React.js, Node.js, MongoDB, and Express.js Integration for Website Development. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.4932790

Zhan, D., Yu, Z., Yu, X., Zhang, H., & Ye, L. (2023). Shrinking the Kernel Attack Surface Through Static and Dynamic Syscall Limitation. *IEEE Transactions on Services Computing*, *16*(2), 1431–1443. https://doi.org/10.1109/tsc.2022.3173791

# Appendix 1. INTERVIEW QUESTIONS

| Theoretical Construct | Interview Question | Purpose of Question |
|---|---|---|
| **Authentication** | 1. Can you explain how authentication processes are implemented in your organisation's web applications? | To explore the methods and challenges of implementing secure authentication mechanisms within the MERN stack framework. |
| | 2. What are the key challenges your organisation faces in ensuring secure authentication for users? | To identify specific obstacles in the implementation of authentication protocols, such as multi-factor authentication, and how they are addressed. |
| **Authorization** | 1. How are user roles and permissions managed in your web applications? | To understand the use of role-based access control and other strategies to secure data access in web applications. |
| | 2. Can you describe how decisions are made regarding user permissions and restrictions? | To examine the organisation's approach to defining and enforcing user roles and restrictions within the MERN framework. |
| **Data Protection** | 1. What measures are in place to ensure secure data storage and transmission in your organisation's applications? | To investigate encryption techniques and other data protection strategies used to safeguard sensitive information. |
| | 2. How does your organisation handle potential threats, such as SQL injection or cross-site scripting (XSS)? | To explore practical measures and tools employed to mitigate vulnerabilities and enhance the resilience of web applications. |
| **Integration of Security Tools** | 1. How does your organisation integrate security tools and frameworks during the development lifecycle? | To evaluate the inclusion of automated testing, vulnerability scanners, and other tools in the software development process. |
| | 2. Can you describe the impact of integrating security tools on the efficiency of the development process? | To assess how the integration of security measures influences the workflow and productivity of developers. |
| **Scalability and User Experience** | 1. How do you balance implementing robust security features with maintaining a seamless user experience? | To understand the trade-offs between security and usability in application design and development. |
| | 2. What challenges arise when scaling secure web applications, and how does your organisation address them? | To investigate scalability concerns and strategies for ensuring security in growing or high-traffic web applications. |