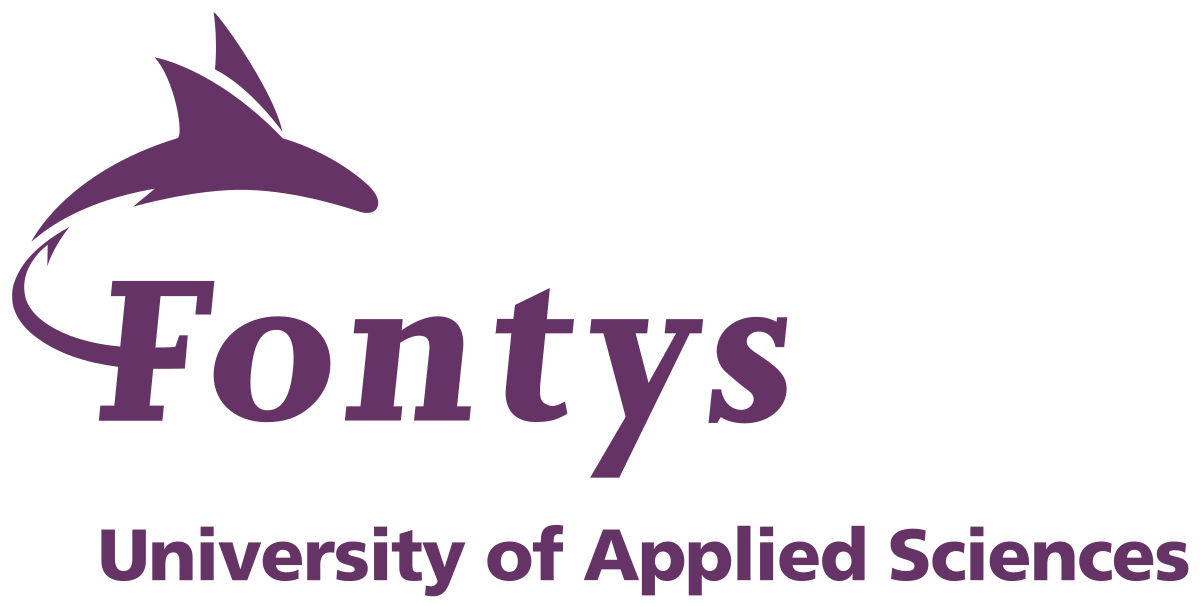
OWASP Report

Individual Track



Done by: Stanislav Petkov (4222024)

Class: S3-CB04

Contents

[1. Risk Analysis 3](#_Toc93076331)

[2. Reasoning 4](#_Toc93076332)

[3. Conclusion 4](#_Toc93076333)

[References 5](#_Toc93076334)

# Risk Analysis

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Likelihood | Impact | Risk | Actions possible | Planned |
| A01: Broken Access Control | High | High | Critical | 1. Implement role-based authorization/authentication 2. Deny access to all but public resources by default | YES |
| A02: Cryptographic Failures | Medium | High | High | 1. 1. Encrypt all sensitive data 2. 2. Store passwords using BCrypt | YES |
| A03: Injection | Low | High | Medium | 1. Make use of ORM 2. Make use of parameterized SQL queries 3. Implement user-input validation | YES |
| A04: Insecure Design | Medium | Medium | Medium | 1. Have unit tests for all possible scenarios of each functionality 2. Take security into consideration during your initial planning | YES |
| A05: Security Misconfiguration | Medium | High | High | 1. Do not install any unnecessary features or frameworks 2. Do not return any overly informative messages to the users | YES |
| A06: Vulnerable and Outdated Components | Low | Medium | Low | 1. Remove unused dependencies, unnecessary features, and files | YES |
| A07: Identification and Authentication Failures | Low | High | Medium | 1. Force the user to choose a strong password via standard password checks (e.g., length, complexity) 2. Do not deploy your application with any default credentials | YES |
| A08: Software and Data Integrity Failures | Low | High | Medium | 1. Ensure that your CI/CD pipelines are secure, and any malicious code doesn’t go in | YES |
| A09: Security Logging and Monitoring Failures | Low | Medium | Low | 1. Ensure that you log all login and failed attempts 2. Ensure that the logs contain all the relevant data | NO |
| A10: Server-Side Request Forgery | Low | Medium | Low | 1. Validate all client-supplied input data 2. Maintain an allow-list of URLs that you would make requests to. | NO |

# Reasoning

Table

Description automatically generated with medium confidenceAs you can see in the table below, each security risk is determined based on the likelihood for an attacker to discover a vulnerability in my application related to the specified risk and the technical impact it can cause on the system (e.g., data loss or exposure of sensible information).

# Conclusion

In conclusion I would say that I have taken some measurements to minimize the risks my system is going to face once it’s in production, but overall, I don’t think they are enough to cover all the possible vulnerabilities involved with each risk, moreover I won’t have the time to implement security features to deal with the last two risks (e.g., sufficient logging). That’s why there will be probably the need for an improvement of the security of my system.

# References

The OWASP Foundation Inc. (2021, 09 26). *OWASP Top 10 - 2021*. Retrieved from https://owasp.org/Top10/

The OWASP Foundation, Inc. (2017, 12 23). *OWASP Risk Rating Methodology*. Retrieved from https://owasp.org/www-community/OWASP\_Risk\_Rating\_Methodology