# A close-up of a logo Description automatically generated

# Security Report: OWASP Top 10 Security Risks Assessment

### This report provides an analysis of how our application deals with the OWASP Top 10 security risks.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Security Risk | Likelihood | Impact | Risk | Mitigation Actions |
| A01:  Broken Access Control | Very Likely | Severe | High | Implemented robust access control mechanisms |
| A02:  Cryptographic Failures | Likely | Severe | Moderate | Encrypt all sensitive data |
| A03:  Injection | Unlikely | Severe | Low | Mitigate injection risks by validating and sanitizing inputs |
| A04:  Insecure Design | Very Likely | Severe | High | Integrate security language and controls into user stories |
| A05:  Security Misconfiguration | Likely | Moderate | Moderate | Sending security directives to clients, e.g., Security Headers. |
| A06:  Vulnerable and Outdated Components | Unlikely | Low | Low | Remove unused dependencies, unnecessary features, components, files, and documentation. |
| A07: Identification and Authentication Failures | Likely | Severe | Moderate or High | Implement multi-factor authentication, avoid default credentials, enforce strong password policies, and manage session security effectively. |
| A08:  Software and Data Integrity Failures | Unlikely | Low | Low | Ensure libraries and dependencies, such as *npm* or *Graddle*, are consuming trusted repositories |
| A09:  Security Logging and Monitoring Failures | Likely | Severe | Moderate | Establish or adopt an incident response and recovery plan, such as *National Institute of Standards and Technology* (*NIST*) |
| A10:  Server-Side Request Forgery | Likely | Severe | Moderate or High | Segment network access, enforce strict firewall rules, sanitize input data, validate URLs with allow lists, and avoid HTTP redirections. |

# Appendices

## Code Snippets

Detailed code snippets as referenced in the report.

## References

<https://owasp.org/Top10/>