November 25, 2025

API Security Testing

API Security Testing Case Study

Project Overview

Key vulnerabilities in API security testing

This project uncovers critical **authentication weaknesses** and vulnerabilities, such as token exposure and hashing flaws, that threaten data integrity and user security in APIs.

Importance of APIs

Understanding the risks they pose

With **94% of organizations experiencing** API incidents, it's vital to address data exposure risks and the misalignment between authentication and authorisation practices.

Tools Utilised

Essential tools for effective testing

In this project, we leveraged tools like Postman and MD5 Generator to perform comprehensive analysis, ensuring thorough evaluation of API vulnerabilities and security weaknesses.

Weak Authentication

Understanding the risks of credential exposure

Exposing credentials in URLs presents significant security risks, as they can be logged or intercepted. Always use the POST body to securely transmit sensitive information.

Token Exposure

Understanding the risks of sensitive data

Token and data exposure can lead to **serious impersonation risks**. Emails, identifiers, and tokens that are leaked may allow attackers to easily access user accounts.

Weak Hashing

The dangers of using MD5 hashing

MD5 hashing is **easily cracked** and vulnerable to attacks. It is crucial to replace it with stronger algorithms like bcrypt or Argon2 to ensure data security.

IDOR Vulnerability

Understanding Insecure Direct Object Reference

Insecure Direct Object Reference (IDOR) allows attackers to **modify identifiers** and gain unauthorized access to another user's account, leading to **full compromise** of sensitive data.

Broken Link Enumeration

Understanding the risks of deprecated endpoints

Deprecated endpoints expose sensitive information, revealing environments and creating a pathway for attackers to conduct reconnaissance. Awareness of these risks is essential for effective API security.

Improper Error Handling

Understanding the Risks of Revealing Information

Error messages like "Method Not Allowed" can unintentionally expose sensitive backend logic. It's crucial to use **generic messages** to enhance security and reduce information leakage.

OWASP Mapping

Understanding API security vulnerabilities and standards

This section outlines key vulnerabilities as per OWASP standards, highlighting the importance of awareness in securing APIs against **potential threats** and ensuring robust security measures.

Lessons Learned

Understanding vulnerabilities in API security

Throughout this project, I discovered that **small weaknesses can compound**, emphasizing the importance of proper hashing, authorization enforcement, and the risks associated with error message leakage.

Full Project

Explore the complete documentation here

This project includes detailed **reports, findings,** and **recommendations** to improve API security. Check out my GitHub for comprehensive insights and resources.

Get in Touch

Github

https://github.com/Sreenath-thekkedan/RedOps

LinkedIn
https://www.linkedin.com/in/sreenath-thekkedan/