JSON Placeholder Demo API

Prepared For: Future Interns

Prepared By: Karthikeya Dama

Program: Cyber Security Internship (2026)

Date: 16 February 2026

Contents

1.Executive Summary

This report presents the results of a read-only API Security Risk Assessment conducted on the JSON Placeholder public API. The objective of this assessment was to evaluate authentication controls, authorization enforcement, data exposure risks, and potential abuse vectors. The testing was performed using ethical and non-intrusive methods in accordance with responsible security practices.

Several security weaknesses were identified which, if present in a production SaaS environment, could result in unauthorized access, excessive data disclosure, and business risk exposure.

2.Scope and Methodology

Scope: The assessment was limited to publicly accessible endpoints of the JSON Placeholder API. Only safe GET and controlled POST requests were performed.

Methodology:
The assessment followed a structured security review process:
• Documentation review
• Endpoint enumeration
• Authentication verification
• Authorization testing
• Response analysis
• Header inspection

Tools Used:
• Postman
• Browser Developer Tools
• Manual HTTP analysis

No exploitation or denial-of-service testing was conducted.

3. API Overview

API Name: JSON Placeholder
Base URL: https://jsonplaceholder.typicode.com

Endpoints Tested:
• GET /users
• GET /users/1
• GET /users/2
• POST /users/2

Risk ID	Finding	Severity	Business Impact	Recommendation
RISK-01	Public user list accessible without authentication	Medium	User data can be enumerated by unauthorized parties	Implement authentication middleware (JWT/OAuth)
RISK-02	Insecure Direct Object Reference (BOLA)	High	Unauthorized access to other user records	Implement object-level authorization checks
RISK-03	Excessive data exposure	Medium	Sensitive fields returned unnecessarily	Return minimal required fields via response filtering
RISK-04	POST endpoint accessible without authentication	High	Potential unauthorized data manipulation	Enforce token-based authentication
RISK-05	No visible rate limiting	Medium	Risk of automated abuse or scraping	Implement request throttling (e.g., 100 requests/min/IP)

4. **Detailed Risk Analysis**

4.1 RISK-01 – Unauthenticated Endpoint Access

The endpoint /users was accessible without any authentication mechanism. No API key, token, or authorization header was required to retrieve user information. In a production SaaS environment, this would allow attackers to enumerate user data.

Business Impact:
Exposure of user data can lead to privacy violations, compliance issues, and reputational damage.

Recommendation:
Implement mandatory authentication for all user-related endpoints.

4.2 RISK-02 – Broken Object Level Authorization (BOLA)

Modifying the user ID parameter from /users/1 to /users/2 successfully returned another user's data without authentication. This indicates absence of object-level authorization validation.

Business Impact:
Attackers could access other customers' private records by manipulating object identifiers.

Recommendation:
Enforce strict server-side authorization checks to validate user ownership before returning data.

5.Risk Classification Criteria

High: Immediate risk of data compromise or unauthorized modification.

Medium: Data exposure or misuse possible with moderate impact.

Low: Minimal business impact or limited exploitability.

6. Conclusion

The API security assessment identified multiple control gaps that would pose significant security risks in a real-world SaaS deployment. The absence of authentication, authorization enforcement, and rate limiting represents critical weaknesses. Implementing secure API design principles would substantially improve the overall security posture.

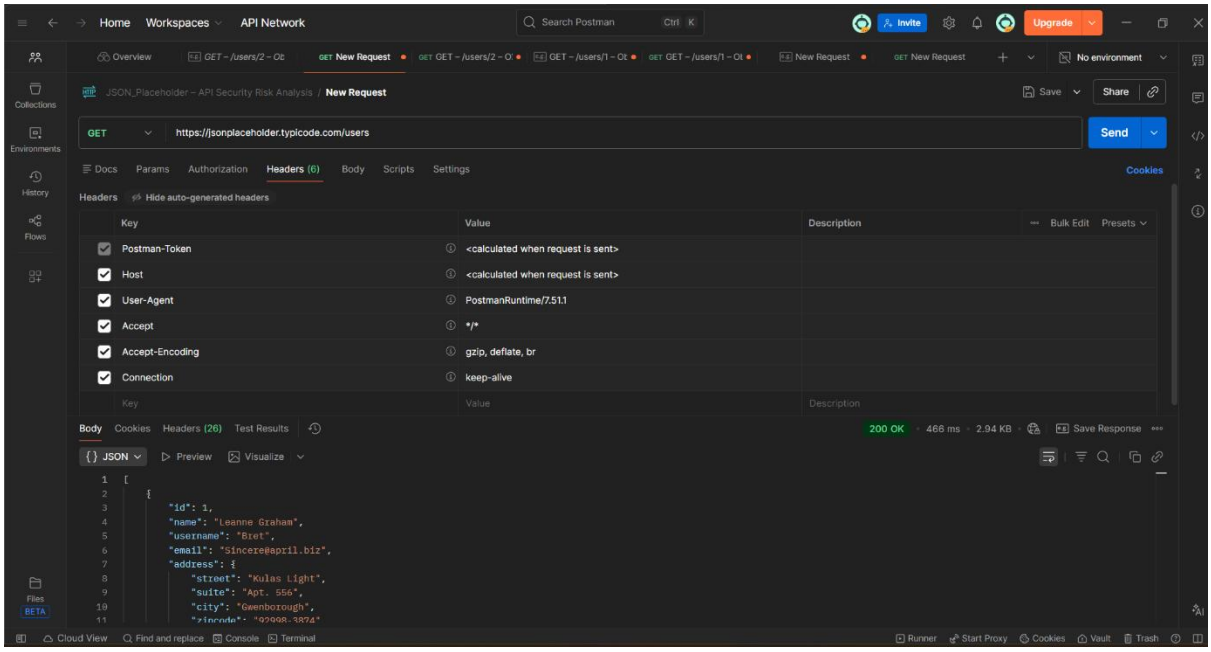

Figure 1: Public user list accessible without authentication

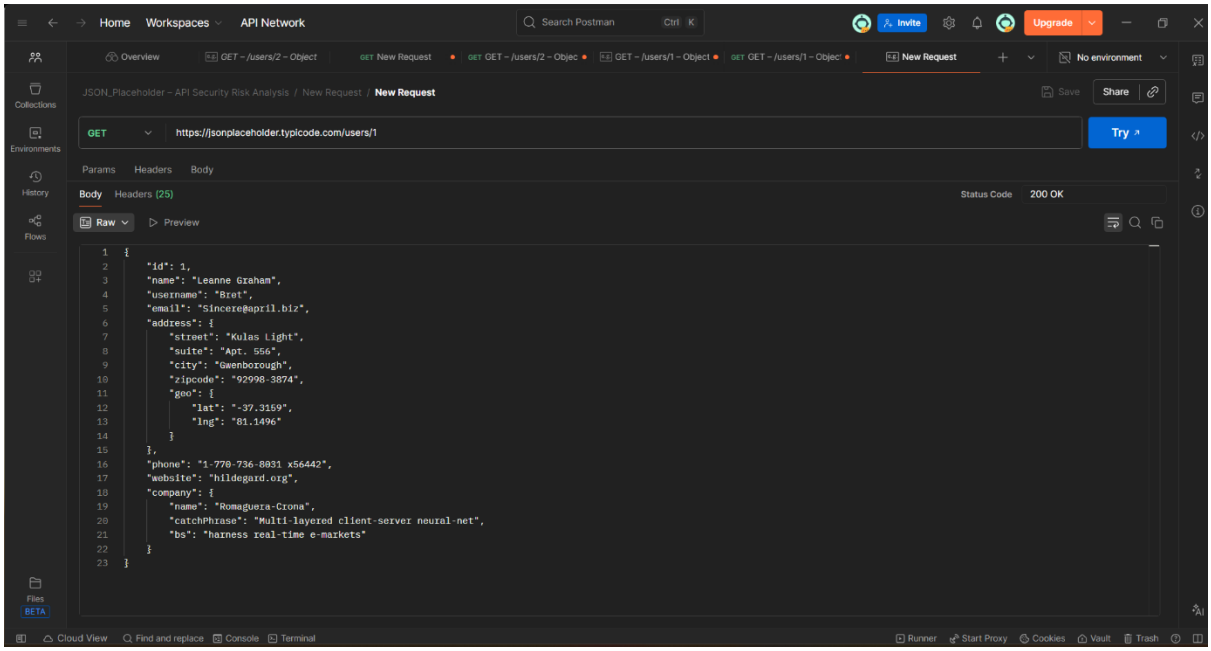

Figure 2: Object ID manipulation showing BOLA vulnerability

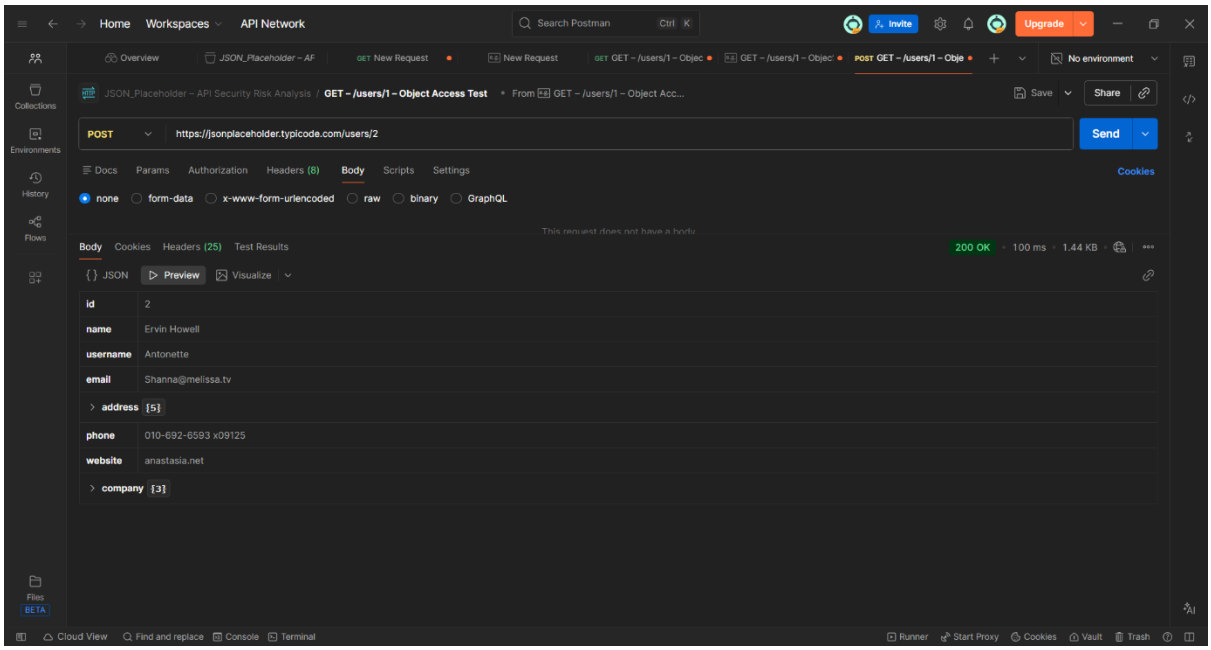

Figure 3: POST request executed without authentication