API Security Risk Analysis Report on

https://jsonplaceholder.typicode.com

Date – 22/01/26

Name- Gatik Pal

Table Of Content

Executive Summary

This report presents the results of an API security risk analysis conducted on publicly accessible API endpoints. The objective of this assessment was to evaluate the security posture of the APIs from an attacker's perspective while maintaining a non-intrusive, analysis-only approach aligned with industry best practices.

The assessment identified multiple security weaknesses related to missing authentication, broken authorization controls, and excessive data exposure. Several endpoints were accessible without any form of authentication, allowing unrestricted access to structured data. These weaknesses significantly increase the risk of unauthorized data access, automated data scraping, and abuse of backend resources.

Overall, the API exhibits a **HIGH risk security posture**. If exploited, the identified issues could lead to privacy violations, regulatory non-compliance, infrastructure misuse, and loss of customer trust. Addressing these issues at an early stage will substantially reduce the attack surface and improve the overall resilience of the system.

Project Objective

The primary objective of this project was to analyze the security configuration of exposed API endpoints and identify risks arising from misconfigurations and missing controls.

The assessment focused on the following goals:

- Identifying API security misconfigurations and weak access controls
- Evaluating authentication and authorization enforcement at the endpoint level
- Mapping identified risks to the **OWASP API Security Top 10** framework
- Assessing the potential impact of these risks on business operations and end users

This analysis was conducted in an educational and defensive security context, simulating how a malicious actor could abuse the API if the issues remain unresolved.

Scope of Assessment

In Scope

The following publicly accessible API endpoints were included in the assessment:

- `GET /users`
- `GET /posts`

The assessment evaluated endpoint accessibility, response behavior, data exposure, and authorization controls.

Out of Scope

The following areas were explicitly excluded from this assessment:

- Authentication implementation testing (login mechanisms)
- Source code review
- Backend infrastructure or database security testing
- Denial-of-Service or stress testing

Methodology

The assessment followed a structured, non-intrusive methodology aligned with real-world API security reviews.

Documentation Review

Available API documentation and endpoint behavior were reviewed to understand expected functionality, access requirements, and data structures.

Manual API Testing

Endpoints were manually accessed using standard HTTP requests to observe authentication requirements, authorization enforcement, and response consistency.

Response Analysis

API responses were analyzed to identify exposed data fields, object identifiers, and metadata that could aid attackers.

OWASP API Top 10 Mapping

Each identified issue was mapped to relevant categories from the OWASP API Security Top 10 to align findings with industry standards.

Severity Classification

Findings were classified as **High** or **Medium** severity based on exploitability, impact, and likelihood of abuse.

API Documentation & Initial Observations

This section documents factual observations made during the assessment without assigning risk or severity.

Documentation Review Findings

- API endpoints were publicly accessible
- No authentication mechanism or API key was required
- User-related and content-related data was returned in responses
- No explicit rate limiting policies were documented
- Access control requirements were not defined

Technical Observations

- Basic rate limiting appeared to be present but inconsistently enforced
- Limited security-related HTTP headers were observed
- Backend and framework-related information was exposed via response headers
- CORS configuration appeared permissive and potentially overexposed

Purpose: This section highlights what information and access an attacker can observe immediately without any exploitation effort.

Identified Security Findings – `/users`

Unauthenticated Access to User Data

Description: The `/users` endpoint can be accessed without any authentication. An attacker can retrieve user-related data by directly invoking the endpoint, without proving identity or authorization.

Severity: High

Impact: - Unauthorized access to user data - Privacy violations - High risk of automated data scraping

Excessive Data Exposure

Description: The API response exposes more user attributes than necessary for the intended functionality. This increases the amount of information available to unauthorized parties.

Severity: Medium

Impact: - Increased data leakage surface - Higher privacy and compliance risk - Facilitates profiling and enumeration

Broken Object Level Authorization (BOLA)

Description: The API does not validate whether the requester is authorized to access a specific user object. By modifying object identifiers, an attacker can access data belonging to other users.

Severity: High

Impact: - Cross-user data access - Account and identity exposure - Serious trust and compliance implications

Predictable Object Identifiers

Description: User object identifiers follow a predictable pattern. This enables attackers to enumerate user records through automated requests.

Severity: Medium

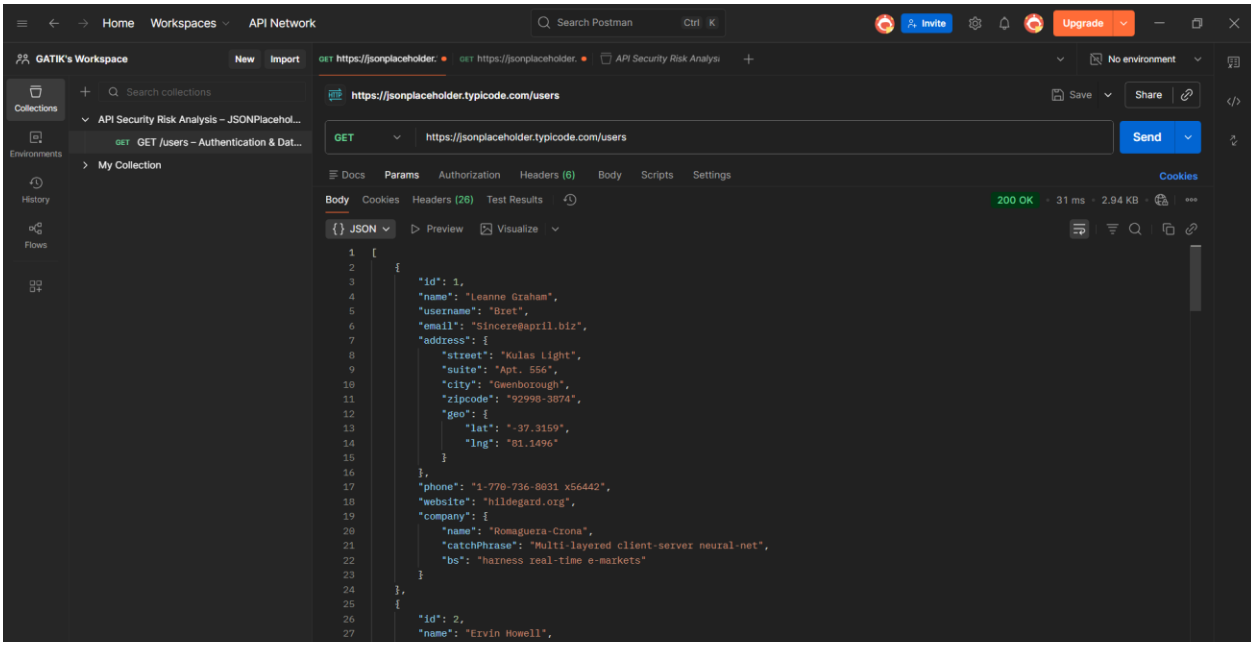
Impact: - Enables large-scale data enumeration - Simplifies automated abuse

Privacy & Compliance Risk

Description: The combination of unauthenticated access and exposed user data introduces potential violations of data protection regulations.

Severity: Medium

Impact: - Regulatory non-compliance - Legal and reputational risks



https://jsonplaceholder.typicode.com/users

GET https://jsonplaceholder.typicode.com/users

200 OK · 31 ms · 2.94 KB

```
[
    {
        "id": 1,
        "name": "Leanne Graham",
        "username": "Bret",
        "email": "Sincere@april.biz",
        "address": {
            "street": "Kulas Light",
            "suite": "Apt. 556",
            "city": "Gwenborough",
            "zipcode": "92998-3874",
            "geo": {
                "lat": "-37.3159",
                "lng": "81.1496"
            }
        },
        "phone": "1-770-736-8031 x56442",
        "website": "hildegard.org",
        "company": {
            "name": "Romaguera-Crona",
            "catchPhrase": "Multi-layered client-server neural-net",
            "bs": "harness real-time e-markets"
        }
    },
    {
        "id": 2,
        "name": "Ervin Howell",
```

Identified Security Findings – /posts

Unauthenticated Access to Content Data

Description: The /posts endpoint is accessible without authentication, allowing unrestricted access to content data.

Severity: Medium

Impact:

- Uncontrolled data access
- Enables scraping and content abuse

Broken Object Level Authorization (BOLA)

Description: The API does not enforce ownership or access validation for individual post objects.

Severity: High

Impact:

- Unauthorized access to content

- Potential manipulation in write-enabled scenarios

Bulk Data Exposure

Description: The endpoint allows retrieval of large volumes of data in a single or repeated requests.

Severity: Medium

Impact:

- Mass data scraping
- Backend performance impact

Predictable Object Identifiers

Description: Sequential identifiers enable attackers to enumerate content records easily.

Severity: Medium

Impact:

- Facilitates automated harvesting

Missing Rate Limiting

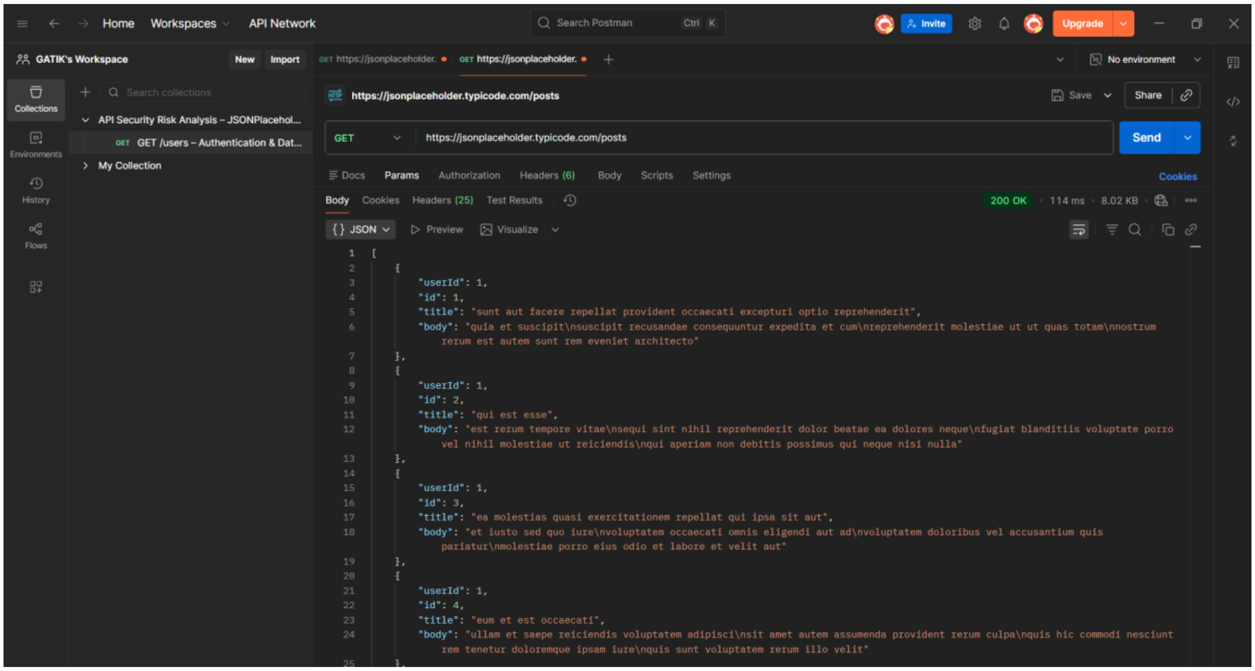
Description: No strict rate limiting controls were observed for the endpoint.

Severity: Medium

Impact:

- API abuse and resource exhaustion

Key Insight: Even APIs without direct PII can be abused if authorization and rate limiting controls are missing.



https://jsonplaceholder.typicode.com/posts

GET https://jsonplaceholder.typicode.com/posts

200 OK · 114 ms · 8.02 KB

```
[
    {
        "userId": 1,
        "id": 1,
        "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",
        "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum
            rerum est autem sunt rem eveniet architecto"
    },
    {
        "userId": 1,
        "id": 2,
        "title": "qui est esse",
        "body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro
            vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui neque nisi nulla"
    },
    {
        "userId": 1,
        "id": 3,
        "title": "ea molestias quasi exercitationem repellat qui ipsa sit aut",
        "body": "et iusto sed quo iure\nvoluptatem occaecati omnis eligendi aut ad\nvoluptatem doloribus vel accusantium quis
            pariatur\nmolestiae porro eius odio et labore et velit aut"
    },
    {
        "userId": 1,
        "id": 4,
        "title": "eum et est occaecati",
        "body": "ullam et saepe reiciendis voluptatem adipisci\nsit amet autem assumenda provident rerum culpa\nquis hic commodi nesciunt
            rem tenetur doloremque ipsam iure\nquis sunt voluptatem rerum illo velit"
    },
```

OWASP API Top 10 Mapping

OWASP API Risk	Description
API1	Broken Object Level Authorization (BOLA)
API2	Broken Authentication
API3	Excessive Data Exposure
API4	Lack of Rate Limiting

Severity Justification

Unauthenticated Access – High

Easy to exploit, no technical barrier, and leads to direct data exposure.

Broken Object Level Authorization – High

Allows cross-user data access with severe privacy and compliance impact.

Excessive Data Exposure – Medium

Increases risk but typically requires chaining with other issues.

Predictable Object Identifiers – Medium

Facilitates automation but does not directly compromise systems alone.

Missing Rate Limiting – Medium

Enables abuse at scale but impact depends on traffic volume.

Business Impact Analysis

Impact Area	Business Risk
Data Privacy	Exposure of user data and regulatory penalties
Brand Trust	Loss of customer confidence
Compliance	GDPR and data protection violations
Infrastructure	Increased operational costs due to abuse
Reputation	Long-term brand damage

Remediation & Security Recommendations

Authentication

- Enforce authentication on all sensitive endpoints
- Use token-based authentication mechanisms

Authorization

- Implement object-level authorization checks
- Validate user ownership for each request

Data Exposure Control

- Apply data minimization principles
- Return only required fields in API responses

Rate Limiting

- Enforce strict per-user and per-IP rate limits

Secure Object Identifiers

- Use non-predictable, UUID-based identifiers

Remediation Summary Table

Issue	Risk Level	Recommended Fix
Unauthenticated Access	High	Enforce authentication
BOLA	High	Implement object-level authorization
Excessive Data Exposure	Medium	Limit response fields
Predictable IDs	Medium	Use non-sequential IDs
Missing Rate Limiting	Medium	Apply rate limits

Overall Risk Rating

Overall Risk: HIGH

The combined presence of missing authentication, broken authorization, and excessive data exposure significantly elevates the overall risk level.

Conclusion

The API is functionally operational but lacks essential security controls. The identified issues are easily exploitable and require minimal attacker effort. Implementing the recommended remediations will drastically reduce the attack surface and improve the API's security posture.

Early corrective actions will not only enhance security but also protect business reputation, user trust, and regulatory compliance.