API Security Risk Analysis Report

Security risk report

Prepared by:

KAKUMANU JAYADHARANI

Future Interns Project

Cybersecurity Task 3

OBJECTIVE

The objective of this task is to perform a read-only API Security Risk Analysis using public or demo APIs. The analysis focuses on identifying common security risks such as unauthenticated access, excessive data exposure, and missing security controls without exploiting or attacking the system. This task helps understand how API security audits are performed in real-world security consulting environments.

API SELECTED

API Name:JSONPlaceholder

Base URL:

https://jsonplaceholder.typicode.com

Why this API?

Public and safe for testing
Designed for learning purposes
No real sensitive data
Suitable for beginner security analysis

TOOLS & TESTING ENVIRONMENT

The following tools were used during analysis:

Postman

- Sending API requests
- Viewing responses
- Header inspection

Kali Linux Terminal

- Header analysis using curl command

Command Used:

- curl -I https://jsonplaceholder.typicode.com/posts

TESTING PROCESS

- Reviewed API documentation in browser.
- Selected safe public API endpoints.
- Sent GET requests using Postman.
- Observed response data structure.
- Checked authentication requirements.
- Inspected headers and status codes.
- Identified potential security risks.
- Classified risk severity and documented observations.

ENDPOINTS TESTED

./posts Endpoint

Method: GET
Returns post objects with userId, title, body.
Accessible without authentication.

./users Endpoint

Method: GET
Returns user details such as name, email, phone, address, and company info.

. /comments Endpoint

Method: GET
Returns comment records linked to posts.

SECURITY FINDINGS

⚠️ Finding 1 — Unauthenticated Endpoint

All tested endpoints are publicly accessible without authentication.
Risk Level: Medium
Impact: Unauthorized users can access API data.

⚠️ Finding 2 — Excessive Data Exposure

The users endpoint returns full user objects including contact and address details.
Risk Level: Medium
Impact: In real APIs this could lead to information leakage.

⚠️ Finding 3 — Missing Rate Limiting Indicators

No clear rate-limiting information observed in responses.
Risk Level: Medium
Impact: API may be vulnerable to abuse through excessive requests.

⚠️ Finding 4 — Public Demo Data

Data appears to be test/demo information.
Risk Level: Low
Impact: No real sensitive data exposure.

RISK SEVERITY TABLE

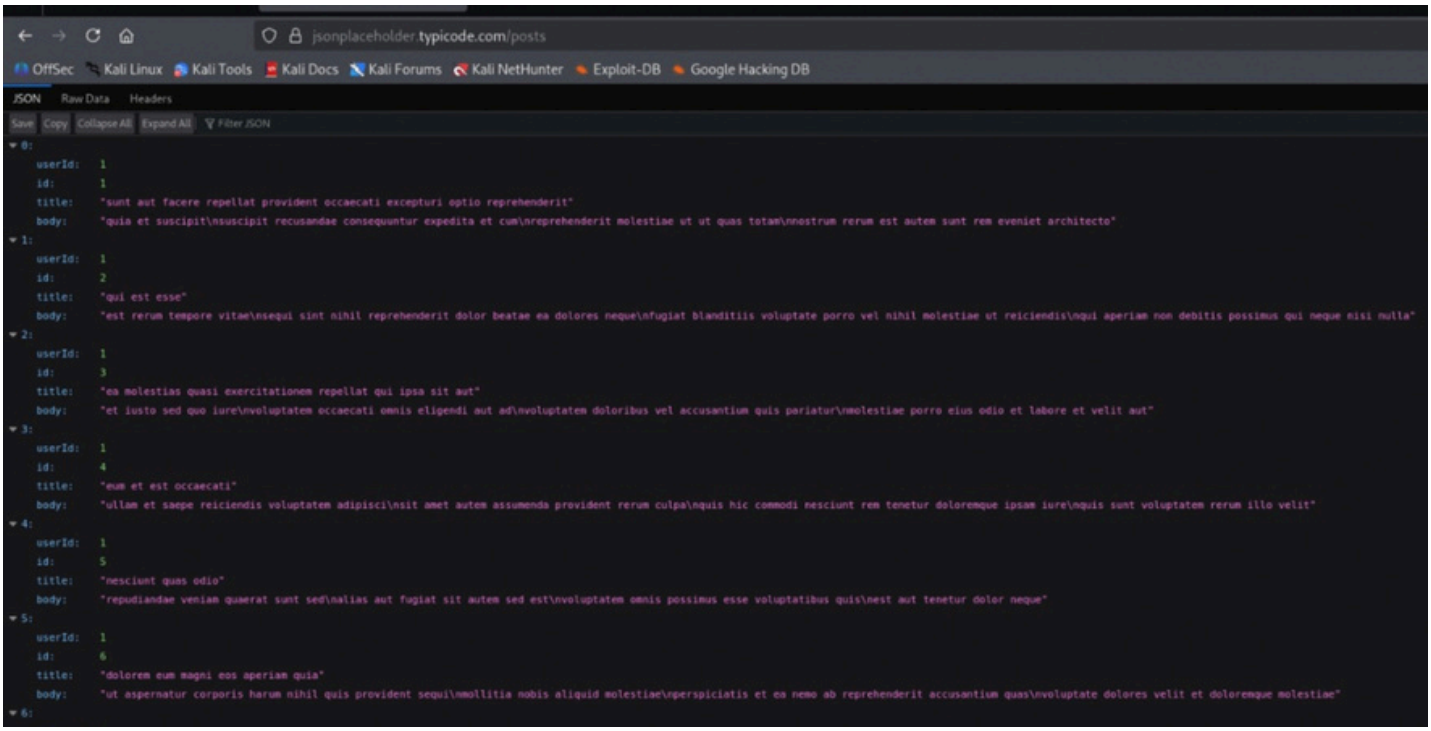
RISK	SEVERITY	IMPACT
Unauthenticated endpoint access	Medium	Unauthorized access
Excessive data exposure	Medium	Information leakage
Missing rate limiting	Medium	API abuse possible
Demo/Public data	Low	Minimal impact

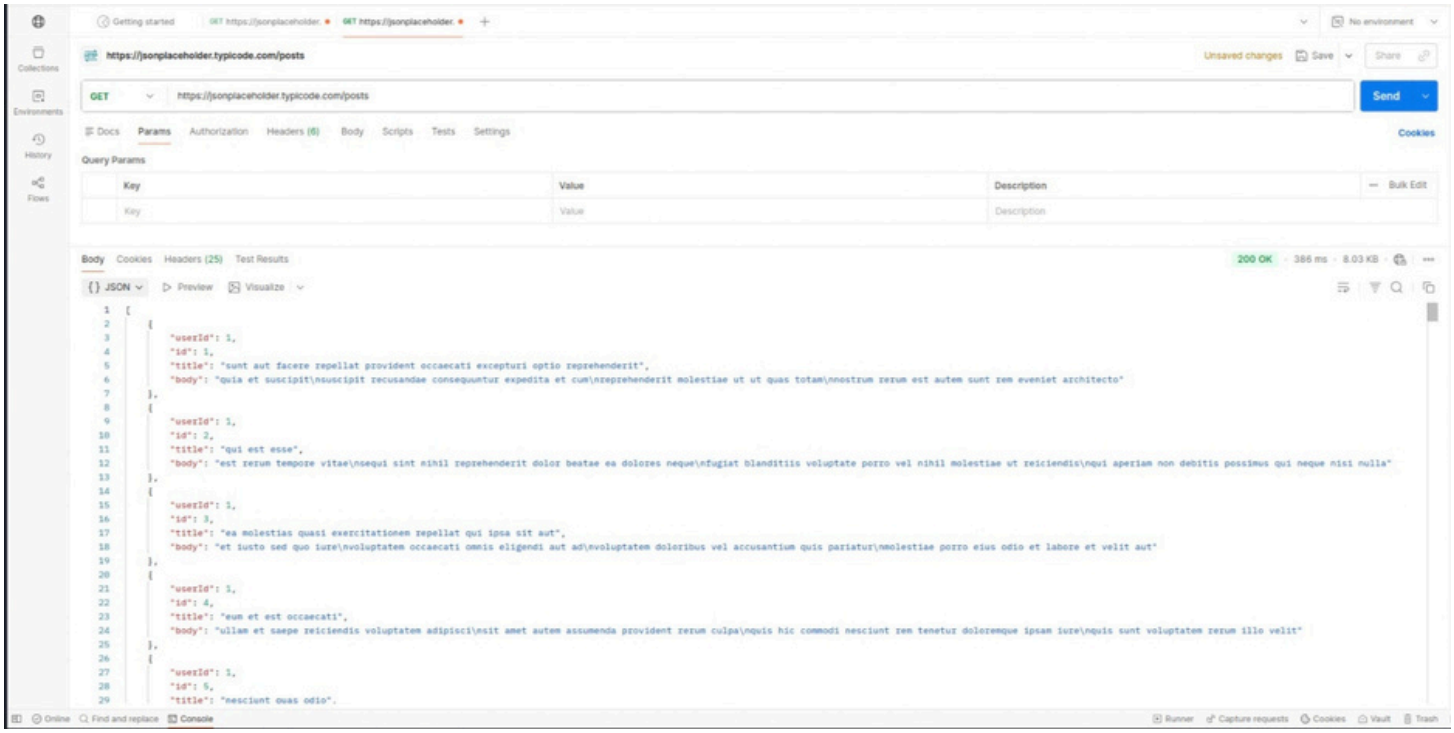
REMEDIATION RECOMMENDATIONS

To improve API security, the following practices are recommended:

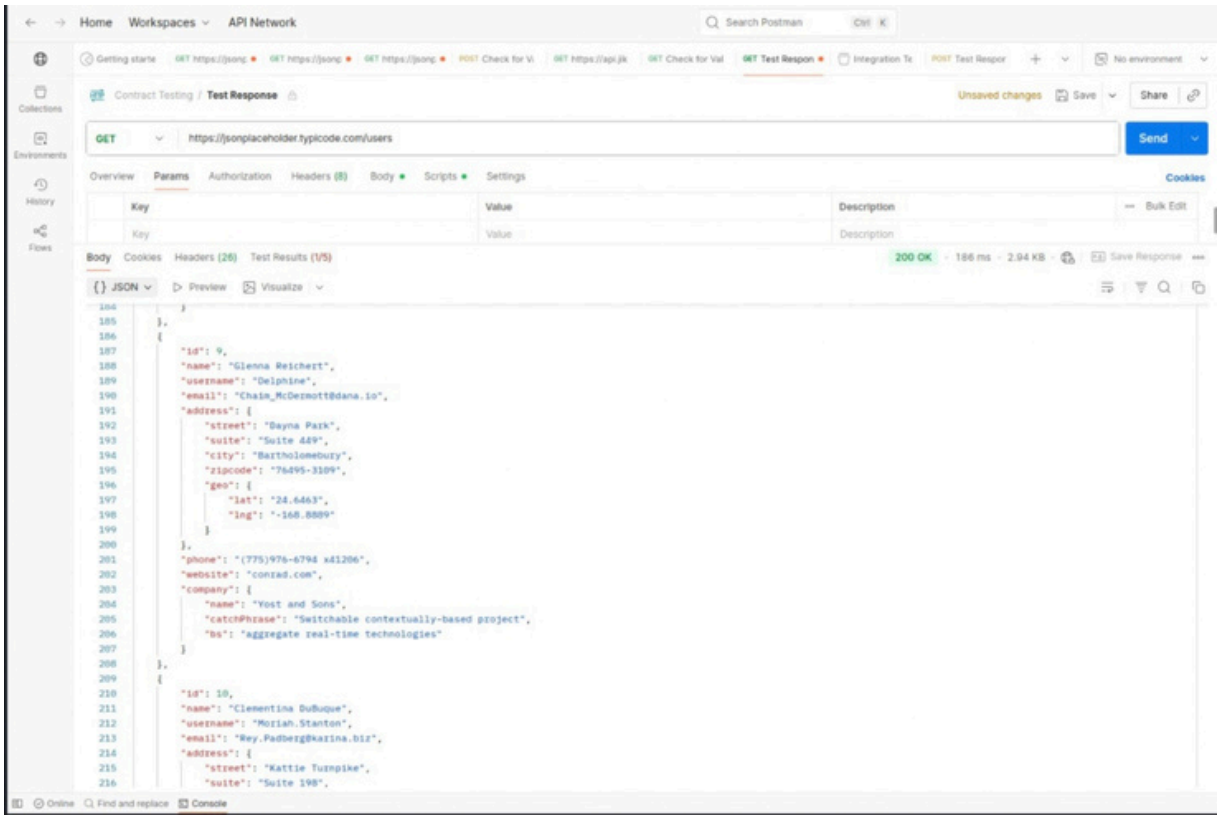
- Implement authentication tokens (API keys / OAuth).
- Return only required fields in responses.
- Enable rate limiting to prevent abuse.
- Apply input validation for request data.
- Use security headers consistently.

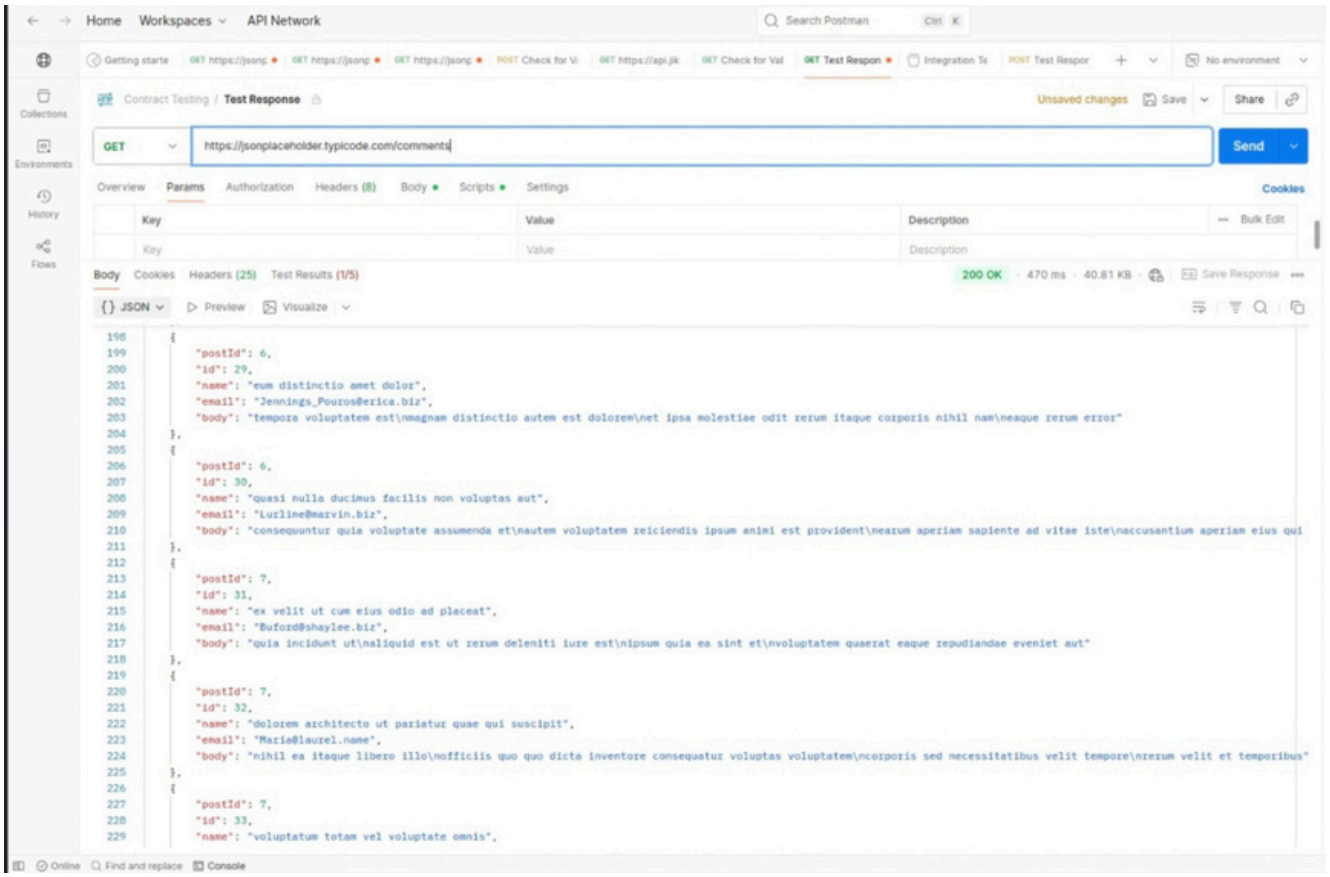
SCREENSHOTS




Getting started
GET https://jsonplaceholder.
GET https://jsonplaceholder.
No environment
Collections
Environments
History
Flows
https://jsonplaceholder.typicode.com/posts
Unsaved changes
Save
Share
GET
https://jsonplaceholder.typicode.com/posts
Send
Docs
Params
Authorization
Headers (6)
Body
Scripts
Tests
Settings
Cookies
Query Params
Key
Value
Description
Bulk Edit
Body
Cookies
Headers (25)
Test Results
200 OK
386 ms
8.03 KB
{} JSON
Preview
Visualize
[
{
"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"
},
{
"userId": 1,
"id": 5,
"title": "nesciunt quas odio",
"Online
Find and replace
Console
Runner
Capture requests
Cookies
Vault
Trash

```
┌──(root㉿kali)-[/home/kali]
└─# curl -I https://jsonplaceholder.typicode.com/posts
HTTP/2 200
date: Tue, 17 Feb 2026 13:02:00 GMT
content-type: application/json; charset=utf-8
access-control-allow-credentials: true
cache-control: max-age=43200
etag: W/"6b80-Ybsq/K6GwwqrYkAsFxqDXGC7DoM"
expires: -1
nel: {"report_to":"heroku-nel","response_headers":["Via"],"max_age":3600,"success_fraction":0.01,"failure_fraction":0.1}
pragma: no-cache
report-to: {"group":"heroku-nel","endpoints":[{"url":"https://nel.heroku.com/reports?s=9PWz1wYqSXFl%2BOkpvCXHzdvru%2FCLEF%2Fz1HL197eq8Og%3D\u0026sid=e11707d5-02a7-43ef-b45e-2cf4d2036f7d\u0026ts=1771161155"}],"max_age":3600}
reporting-endpoints: heroku-nel="https://nel.heroku.com/reports?s=9PWz1wYqSXFl%2BOkpvCXHzdvru%2FCLEF%2Fz1HL197eq8Og%3D&sid=e11707d5-02a7-43ef-b45e-2cf4d2036f7d&ts=1771161155"
server: cloudflare
vary: Origin, Accept-Encoding
via: 2.0 heroku-router
x-content-type-options: nosniff
x-powered-by: Express
x-ratelimit-limit: 1000
x-ratelimit-remaining: 999
x-ratelimit-reset: 1771161197
age: 28132
cf-cache-status: HIT
cf-ray: 9cf57a065a3dfd1f-SIN
alt-svc: h3=":443"; ma=86400
```




Home
Workspaces
API Network
Search Postman
Contract Testing / Test Response
Unsaved changes
Save
Share
GET
https://jsonplaceholder.typicode.com/comments
Send
Overview
Params
Authorization
Headers (8)
Body
Scripts
Settings
Cookies
Key
Value
Description
Bulk Edit
Body
Cookies
Headers (25)
Test Results (1/5)
200 OK
470 ms
40.81 KB
Save Response
JSON
Preview
Visualize
{
"postId": 6,
"id": 29,
"name": "eum distinctio amet dolor",
"email": "Jennings_Pouros@erica.biz",
"body": "tempora voluptatem est\nmagnam distinctio autem est dolorem\net ipsa molestiae odit rerum itaque corporis nihil nam\neaque rerum error"
},
{
"postId": 6,
"id": 30,
"name": "quasi nulla ducimus facilis non voluptas aut",
"email": "Lurline@marvin.biz",
"body": "consequuntur quia voluptate assumenda et\nautem voluptatem reiciendis ipsum animi est provident\nearum aperiam sapiente ad vitae iste\naccusantium aperiam eius qui
},
{
"postId": 7,
"id": 31,
"name": "ex velit ut cum eius odio ad placeat",
"email": "Buford@shaylee.biz",
"body": "quia incidunt ut\naliquid est ut rerum deleniti iure est\nipsum quia ea sint et\nvoluptatem quaerat eaque repudiandae eveniet aut"
},
{
"postId": 7,
"id": 32,
"name": "dolorem architecto ut pariatur quae qui suscipit",
"email": "Maria@laurel.name",
"body": "nihil ea itaque libero illo\nofficiis quo quo dicta inventore consequatur voluptas voluptatem\ncorporis sed necessitatibus velit tempore\nrerum velit et temporibus"
},
{
"postId": 7,
"id": 33,
"name": "voluptatum totam vel voluptate omnis",
"Online
Find and replace
Console

CONCLUSION

This assessment demonstrated a basic API security review using safe and ethical testing methods. The API endpoints were openly accessible and highlighted common risks such as unauthenticated access and excessive data exposure. Although the API contains demo data, the analysis reflects real-world API security considerations and provides practical experience in documenting security risks professionally.