Future interns task 3

API Security risk analysis report

API tested: JSONPlaceholder

Introduction

This report presents a read only API security risk analysis performed on the JSONPlaceholder public test API. This had been done to identify potential API security risks, review authentication and authorization controls. The analysis was conducted using Postman and browser developer tools while following non intrusive testing methods.

Scope and Methodology

This assessment was done on a public demo API which has been intended for testing and educational purposes. The testing was focused on publicly accessible endpoints and analysed authentication requirements and security headers.

The following methodology was used in this assessment; reviews API documentation, sent GET requests using Postman, inspected API response data and headers and evaluated authentication and authorization requirements.

Tools used

Postman- for sending API requests and analysing responses

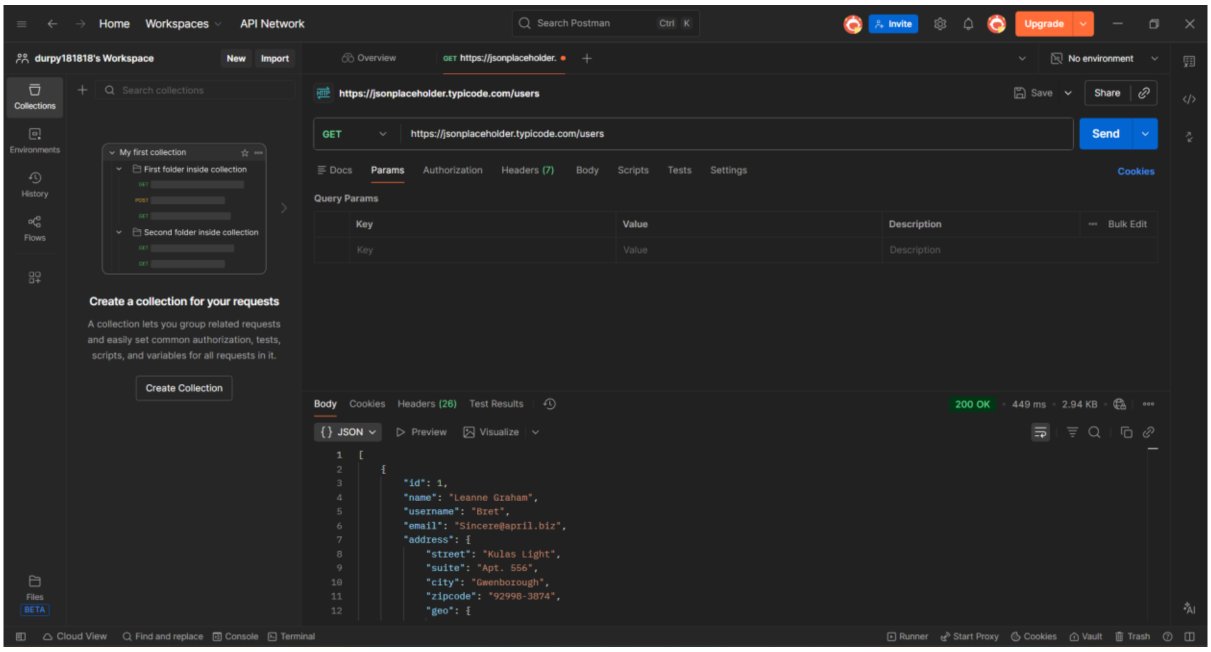
Browser Developer tools- for reviewing network requests and headers

Public API Documentation – to understand endpoint structure and usage.

API Overview

The API selected for this task was JSONPlaceholder, this is a public test API which was designed for development and learning purposes. This API gives sample endpoints that simulate real world applications. The example endpoints tested were /users, /posts, /comments.

Risk analysis and findings



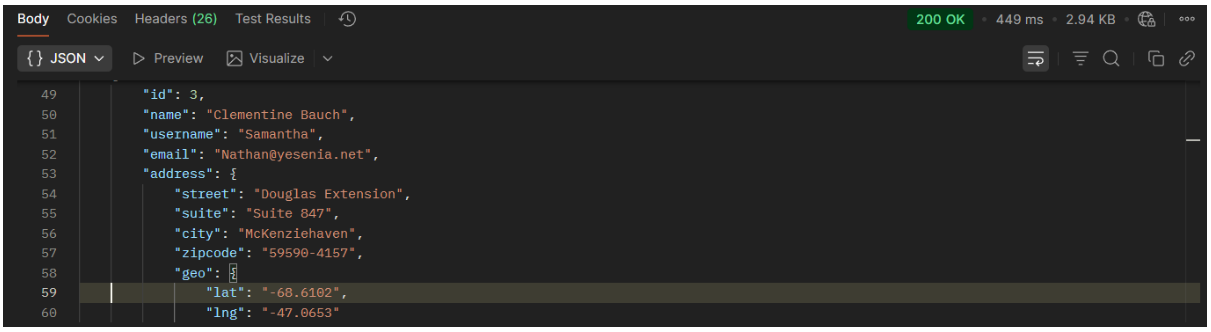
https://jsonplaceholder.typicode.com/users

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

200 OK · 449 ms · 2.94 KB

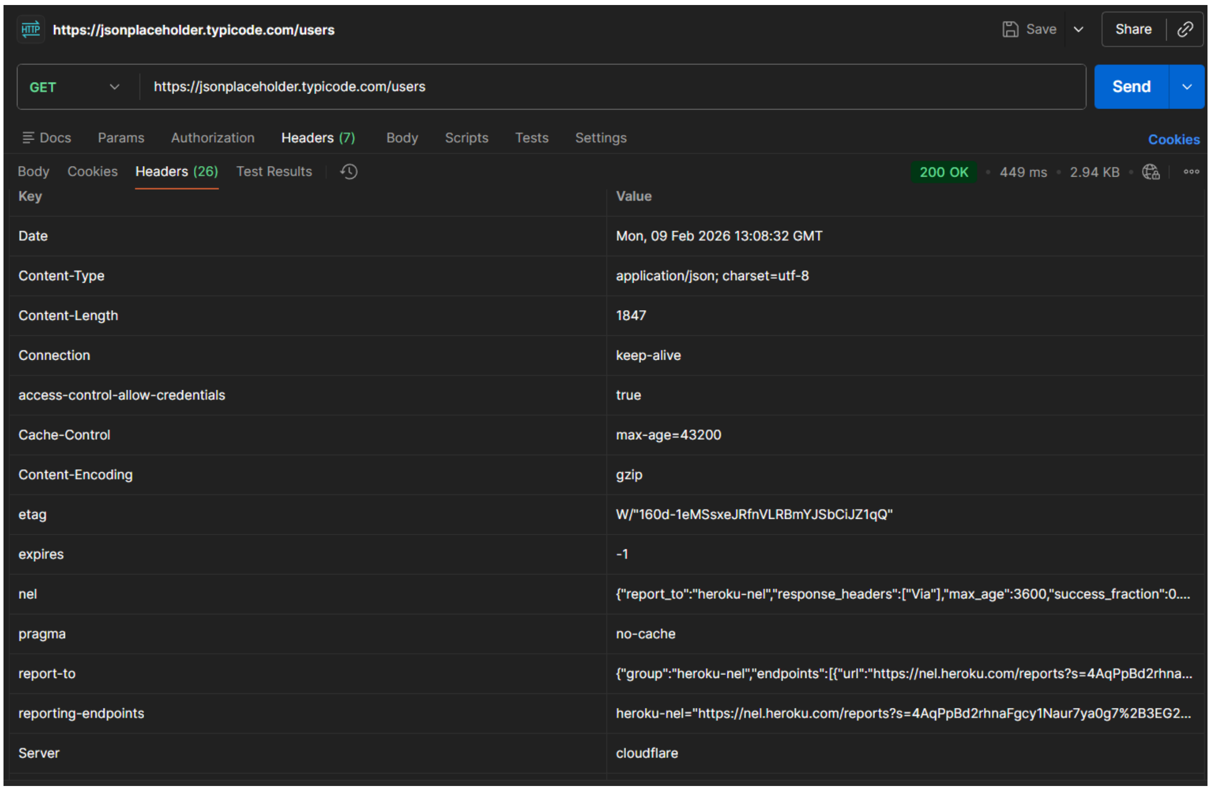
```
[
    {
        "id": 1,
        "name": "Leanne Graham",
        "username": "Bret",
        "email": "Sincere@april.biz",
        "address": {
            "street": "Kulas Light",
            "suite": "Apt. 556",
            "city": "Gwenborough",
            "zipcode": "92998-3874",
            "geo": {
```

The / users endpoint allowed data access without any authentication. If implemented in a real application endpoints would allow unauthorized users to access sensitive data. A way to solve this would be to implement authentication mechanisms such as API keys to allow only authorized users to access sensitive data.



```
    "id": 3,
    "name": "Clementine Bauch",
    "username": "Samantha",
    "email": "Nathan@yesenia.net",
    "address": {
        "street": "Douglas Extension",
        "suite": "Suite 847",
        "city": "McKenziehaven",
        "zipcode": "59590-4157",
        "geo": {
            "lat": "-68.6102",
            "lng": "-47.0653"
```

This screenshot is the API response that contains detailed user information. Such detailed responses could expose sensitive information if accessed by unauthorized users. The way to avoid such an incident would be to implement data minimization principles and return only essential fields.



Key	Value
Date	Mon, 09 Feb 2026 13:08:32 GMT
Content-Type	application/json; charset=utf-8
Content-Length	1847
Connection	keep-alive
access-control-allow-credentials	true
Cache-Control	max-age=43200
Content-Encoding	gzip
etag	W/"160d-1eMSsxeJRfnVLRBmYJSbCiJZ1qQ"
expires	-1
nel	{"report_to":"heroku-nel","response_headers":["Via"],"max_age":3600,"success_fraction":0....
pragma	no-cache
report-to	{"group":"heroku-nel","endpoints":[{"url":"https://nel.heroku.com/reports?s=4AqPpBd2rhna...
reporting-endpoints	heroku-nel="https://nel.heroku.com/reports?s=4AqPpBd2rhnaFgcy1Naur7ya0g7%2B3EG2...
Server	cloudflare

This image shows that the headers contain certain keys that each have their own unique value and adds functionality to the API.

Overall Risk Summary

This API analysis identified several common API security risks such as unauthenticated endpoints, excessive data exposure and missing rate limiting controls. While the tested the API is designed for research purposes it helps to highlight potential vulnerabilities that could impact real world applications.

Conclusion

This task had shown how common API security risks can be identified through read only analysis and response infection. The findings highlight the importance of implementing authentication controls, limiting data exposure and applying rate limiting protections, by addressing these risks organizations can significantly improve API security and reduce the chance of unauthorized access.