Auth & Security API Testing Project

Project Overview

This project focuses on **testing various authentication mechanisms** (API Key, Basic Authentication, Bearer Token, OAuth 2.0) and **security validation** on sample APIs. The goal was to explore how APIs handle authentication and ensure they are secure against common vulnerabilities.

Technologies Used

- **Postman**: Used for creating and organizing API collections, defining requests, and testing authentication flows.
- **Newman**: The command-line companion for Postman used to automate the collection run and generate reports.
- **APIs Tested**:
    - OpenWeatherMap API for API key authentication
    - Postman Echo for Basic Authentication and Bearer Token testing
    - DummyJSON API for testing JWT tokens and OAuth 2.0 authentication

Testing Approach

1. **API Key Authentication**: Testing API requests using an API key for accessing secured endpoints.
2. **Basic Authentication**: Implemented basic authentication tests to validate login and access control.
3. **Bearer Token Authentication**: Validated JWT token usage for protected endpoints.
4. **OAuth 2.0**: Ensured proper OAuth 2.0 authentication flow with secure token-based access.
5. **Negative Security Tests**: Ran tests to ensure the system handles invalid or missing tokens correctly, returning appropriate errors (e.g., 401 Unauthorized).

Project Steps

1. **Postman Collection Creation**:
    - Designed API requests for each authentication type mentioned above, with various endpoints and test cases.
    - Created an **environment** with variables such as `API_KEY` and `jwt_token` for dynamic request handling.
2. **Running the Collection with Newman**:
    - Executed the entire collection using **Newman**, which is Postman's command-line tool.
    - Used the **HTML** and **HTMLExtra** reporters to generate a detailed, colorful report showing the test results.
3. **Generated Test Reports**:
    - **Newman Dashboard**: The run was successful, with all tests passing.
    - **Assertions**: 12 assertions were made, all of which passed, indicating the correct behavior of the APIs under various authentication schemes.
    - **Run Summary**: The total run time was 7.6 seconds, with a total data received of 3.43KB, reflecting the efficiency and speed of the tests.

Key Features

- **Authentication Mechanisms**: Demonstrates multiple types of authentication in real-world API scenarios.
- **Security Testing**: Ensures secure handling of sensitive information like tokens and passwords.
- **Automated Reporting**: The use of **Newman** allows for automatic generation of HTML reports, which are great for tracking testing results and visualizing data.

Project Results

The project successfully completed all the defined tests with zero failed assertions. The HTML report generated provides a comprehensive overview of the test run, making it easy to understand the overall health of the authentication mechanisms tested.