Security Testing of API using Postman and Swagger tools and its use in Internet of Things (IOT)

Dheeraj[1], Kalpana Sharma[2]

[1] Department of Computer Science and Engineering, Bhagwant University, Ajmer, Rajasthan, India
[2] Department of Computer Science and Engineering, Bhagwant University, Ajmer, Rajasthan, India

Abstract— In this paper we have outlined security testing in context of REST APIs. REST and SOAP APIs are discussed along with other APIs. From the research we found that majority of the software applications these days follow web frameworks and microservice based architectures. Most of the latest applications use APIs to interact with other applications. So as part of this paper we have covered testing of RESTful APIs using Postman tool and swagger tool. We have briefly explained postman and swagger tools which are quite popular these days across the industry. We also focused on security testing aspect of Internet of things (IOT) because APIs are used in most of the IOT devices. As per gartner study more than 20 billion devices will go on internet by 2020. Main focus of this paper remains on security of software applications, API and Internet of Things (IOT).

Index Term/Keyword: API, API Testing, Postman, Swagger, Security Testing, Internet of Things (IOT)

1. Introduction:

Security testing is a major concern these days and most of the recent software fail due to lack of security and performance testing[1]. Software security testing is the process to ensure that software conforms to the security requirements and design of the application. Software security testing can be divided into security functional testing and security vulnerability testing. Security functional testing ensures whether software security functions are implemented correctly and consistent with security requirements basing on security requirement specification. Software security requirements mainly include data confidentiality, integrity, availability, authentication, authorization, access control, audit, privacy protection, security management, etc. Security vulnerability testing is to discover security vulnerabilities as an attacker. Vulnerability refers to the flaws in system design, implementation, operation, management. Vulnerability may be used to attack, resulting in a state of insecurity, Security vulnerability testing is to identify software security vulnerabilities.

2. Application Programming Interface (API):

Application Programming Interface is a software intermediary that allows two applications to talk to each other. It could also be a function OR a method that provides a specific function. Application Programmable Interfaces (API's) are collections of procedures & functions that can be used by other applications to fulfill their functionality [2]. "A collection of methods wrapped in a library or DLL that interact to meet a functional requirement is a component. Rather than a developer having to call each method individually to achieve some usually repetitive functional outcome from the library that functionality is usually exposed via a call to a single API" [3]. Users can use API by calling the function/method by passing appropriate parameters. Major advantage of using an API is that is supports faster development because you don't need to create the same functionality that is already implemented; you would rather have to call that particular API. When you use Facebook and chatting OR using Amazon in android/apple, using google maps in android application, accessing ecommerce website in android/apple; you use APIs to easy access the application.

2.1. Classification of APIs:

There are variety of APIs as shown in figure 1. Webservice APIs like SOAP, REST; there are library based APIs like javascript; Class based APIs like Java API, Android API; Hardware APIs like Video acceleration, Hard disk drives, PCI buses. There is need of picking up the correct type of API that will last long. Most of the common types of APIs are Web APIs. These APIs known as web service provide an interface for web applications, or applications that need to connect to each other via the internet to communicate. There are thousands of public APIs that can be used for wide range of purposes ranging from checking weather, checking traffic, making payments, updating your status on social media etc. There are lots more private APIs also used by organizations which are not available for outside users and used internally to extend their features and perform certain functions. Such APIs are authorized to be used inside the company network only. There are further type of web APIs such as SOAP, REST and Remote procedure call (RPC). Most popular out of these is REST.

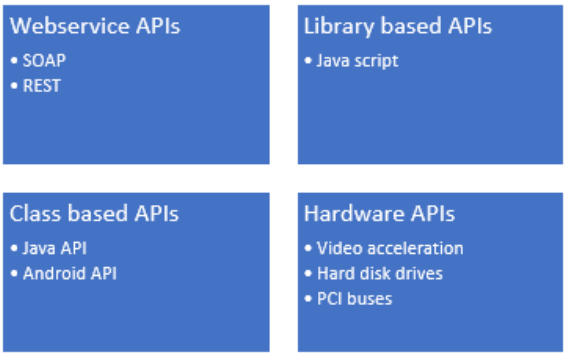

Figure 1: APIs classification

2.2. SOAP Vs REST:

SOAP and Rest [4] are explained briefly in this section. SOAP is a messaging protocol that allows programs that run on disparate operating systems (such as Windows and Linux) to communicate using Hypertext Transfer Protocol (HTTP) and its Extensible Markup Language (XML). Representational State Transfer (REST) is a software architecture style consisting of guidelines and best practices for creating scalable web services. REST is a coordinated set of constraints applied to the design of components in a distributed hypermedia system that can lead to a more maintainable architecture. REST efficiently uses HTTP verbs.

As RESTful APIs are most widely used, we have reviewed them in detail and will review the process to ensure security in rest APIs in next section of this paper. Security is major concern when different applications interact with each other and share and update data. Public and private APIs needs to be designed, implemented and tested to ensure that they communicate, accessed, update as per requirement and maintain security of the data. The major tools used for testing REST APIs are Swagger and Postman. In next section we will look into the process of testing REST APIs using postman.

3. Test Process of APIs using Postman:

A test in Postman is fundamentally a JavaScript code, which run after a request is sent and a response has been received from the server. POSTMAN is very easy to use. It provides collection of API calls, and one has to follow that collection of API calls for testing APIs of application. With Postman you can write and run tests for each request using the JavaScript language.

1. Install the postman app. It is available as a native app for macOS, Windows, and Linux operation system. It is also available as Chrome app. Click on Apps section square if you accessing it from Chrome. Postman tool symbol is specified below in figure 2. Main screen of the Postman tool is shown in figure 3.

Figure 2: Postman tool symbol

2. Most common methods are GET (Retrieving data), POST (Updating data in existing file), PUT(For replacing existing file), and DELETE(Delete data from server).
3. Send Request:
    a. In header toolbar, hit New button
    b. Enter Request (URL- Rest endpoint) URL as specified in below image- URL- postman-echo.com/get

c. Add parameters required to get the data by hitting Params button besides Send button on left side.

d. Hit Send button and validate the body of request to check the result. It could entertain HTML, XML and JSON.

e. API server receive the request and returns a response

f. Postman receives the response

4. User can save request by clicking on Save button specified besides Send button for later use.
5. Integrate with Build management tool using Newman. It is Postman's command line companion using which you can integrate Postman with build management tool to support automated testing.
6. Authorization: Authorization process verifies whether you have permission to access the data you want from the server. It is part of security testing. There are various types of authorizations provided in the tool which we are going to cover in the next section of this paper.

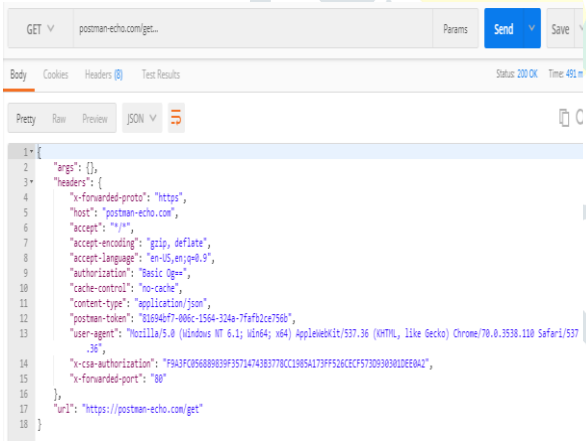

Figure 3: Postman tool.

3.1. Authentication in Postman:

There are various types of authentication as specified in figure 4. The authorization process checks whether user have permission to access the data from the server. This is where security testing is performed to ensure application features and functions are accessible as defined. Authorizations could be passed in the tool to ensure that application features are accessible when authorizations are set and if no authorization is specified then application should not be accessible outside the network. All authorizations specified below are explained in detail in postman tool website [5].

Authorizations In Postman tool	Inherit auth from parent
	No Auth
	Bearer Token
	Basic Auth
	Digest Auth
	OAuth 1.0
	Oauth 2.0
	Hawk Authentication
	AWS Signature
	NTLM Authentication

Figure 4- Types of Authorizations in Postman tool

4. Testing of APIs using Swagger:

Swagger is a set of open-source tools built around the OpenAPI Specification that can help you design, build, document and consume REST APIs. It shows the capability of services without much access to Code, logic or network. Swagger UI makes API design easy to use. A developer can insert his API logic to make swagger UI more user friendly. It takes predefined set of inputs from user to hit a service and generate responses in form of specific codes or keywords.

Swagger UI link : http://<host> : <port>/swagger-ui.html requires input as – Test env being used & basic authentication for user. User can hit any rest end from below list and verify the response. There are various methods available like GET, PUT, POST, DELETE etc as shown in figure 5. Rest endpoints which communicates and integrated with other applications are tested to ensure they are secure and results of GET, PUT, POST and DELETE methods return results based on authentication and authorization. Response of the Swagger tool could be analysed to ensure if it is a SUCCESS OR FAILURE. If it returns 200, then it is considered as successful and 401 response would be considered as authorization error. User has to pass required parameters into it and validate the response.

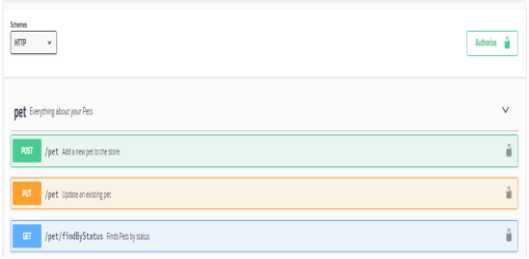

Figure 5- Swagger UI

Swagger has certain benefits compared with other frameworks such as:

a. It's useful for developers and testers. Its interactive and user friendly UI and documentation is helpful for Product Managers, Business Analysts, Dev and QA teams.
b. It could be shared with internal and external teams as it is machine readable and user friendly too. This same swagger API documentation can be used for automating APIs.
c. It's adjustable because it could be used for testing and debugging API issues.

5. Internet of Things

As explained by P.P. ray [6] Internet of Things is a platform where every day devices become smarter, every day processing becomes intelligent, and every day communication becomes informative. While the Internet of Things is still seeking its own shape, its effects have already stared in making incredible strides as a universal solution media for the connected scenario. As per Gartner report, more than 20 billion devices would be connected to internet by 2020.

5.1. Security Concern in IOT:

As per Mark patton et al [7] The Internet of Things (IoT) continues to grow as uniquely identifiable objects are added to the internet. The addition of these devices, and their remote connectivity, has brought a new level of efficiency into our lives. However, the security of these devices has come into question. While many may be secure, the sheer number creates an environment where even a small percentage of insecure devices may create significant vulnerabilities. Anna Kornfeld Simpson et al [8] proposed a central security manager that is built on top of the smarthome's hub or gateway router and positioned to intercept all traffic to and from devices. As mentioned by Gurjan Lally [9] weak security in IoT devices could have dangerous consequences, such as to a car crash, or an intruder entering in our home. As mentioned by Paul in security ledger [10] in October 2016, the distributed denial of service attack on Dyn, a company controlling and managing several DNS services, brought down most of America's Internet, and was caused by an IoT botnet (Mirai). This is mainly due to an increasing number of vulnerabilities in IoT devices being discovered on a daily basis, and that are the consequence of poor IoT security practices. To properly address the security and testing of IoT devices, the first step is the description of a threat model. However, few IoT manufactures base their testing on sound threat modelling techniques and comprehensive IoT security guidelines.

5.2. APIs in IOT Devices:

The application program (or programming) interface, or API, is arguably what really ties together the connected "things" of the "internet of things." IoT APIs are the points of interaction between an IoT device and the internet and/or other elements within the network.

Shaila Sharmeen et al [11] analyzed the efforts regarding malware threats aimed at the devices deployed in industrial mobile-IoT networks and related detection techniques. They considered static, dynamic, and hybrid detection analysis. In this performance analysis, they compared static, dynamic, and hybrid analyses on the basis of data set, feature extraction techniques, feature selection techniques, detection methods, and the accuracy achieved by these methods. Therefore, they identify suspicious API calls, system calls, and the permissions that are extracted and selected as features to detect mobile malware. This will assist application developers in the safe use of APIs when developing applications for industrial IoT networks.

The main contributions of their research effort was divided into the following main areas:

a. Defining the mobile malware detection process for IoT networks.
b. Determining the security limitations for mobile platforms in industrial IoT networks.

c. A comparative analysis of static, dynamic, and hybrid detection processes and their limitations and scopes.

d. Identifying the suspicious permission, API call, and system call lists to enable IoT application developers in the safe use of APIs.

There could be different APIs as specified in figure 6 which would require secure definition, designing, communication and testing.

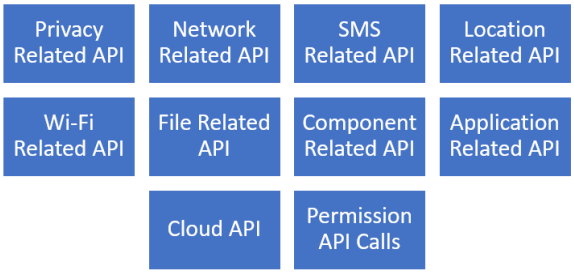

Figure 6 – Various APIs used in IOT

6. CONCLUSION:

This paper reviewed security testing aspect of Application Programming Interface (API) using Postman and Swagger tools in brief which are quiet popular across the industry these days. Most of the applications communicate with outside applications to share and get data feed using API be it google, facebook, mobile applications OR web applications. RESTful APIs are in high demand these, so we focused mainly on security aspect of RESTful APIs. We found that APIs are also used heavily in Internet of Things (IOT) devices which as per gartner study will reach more than 20 billion devices by 2020. Security is again a major concern in IOT devices. So, as part of this paper reviewed the use of APIs in IOT devices and illustrated various APIs used in IOT. A lot of research is required in this area because security is a major concern these days and there is need of designing and formulating a framework and tool to act as real time test agent focusing on functionality, security and performance testing of the applications.

REFERENCES

[1] Dheeraj Chhillar, "Proposed T-Model to cover 4S quality metrics based on empirical study of root cause of software failures", International Journal of Electrical and Computer Engineering (IJECE), Vol. 9, No. 2, April 2019

[2] Ajitha and Amrit Shah, "SofTReL Software Testing Guide Book Part1".

[3] S. L. Bangare, A. R. Khare, P. S. Bangare, "Quality measurement of modularized object oriented software using metrics", ACMInternational Conference ICWET-2011 at Mumbai, ACM 978-1-4503-0449-8/11/02, ISBN: 978-1-4503-0449-8.

[4] Vibha, "Web Services Protocol: SOAP vs REST". International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 4 Issue 5, May 2015

[5] Postman Learning Centre, Authorization-https://learning.getpostman.com/docs/postman/sending_api_requests/authorization/

[6] P.P. Ray Journal of King Saud University – Computer and Information Sciences (2018) 30, 291 – 319

[7] Mark Patton ; Eric Gross ; Ryan Chinn ; Samantha Forbis ; Leon Walker ; Hsinchun Chen , "Uninvited Connections: A Study of Vulnerable Devices on the Internet of Things (IoT)" - 2014 IEEE Joint Intelligence and Security Informatics Conference.

[8] Anna Kornfeld Simpson ; Franziska Roesner ; Tadayoshi Kohno, "Securing vulnerable home IoT devices with an in-hub security manager"- 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)

[9] Gurjan Lally, Daniele Sgandurra, "Towards a Framework for Testing the Security of IoT Devices Consistently", ETAA 2018: Emerging Technologies for Authorization and Authentication pp 88-102, Springer, Part of the Lecture Notes in Computer Science book series (LNCS, volume 11263)

[10] Paul: Mirai Internet of Things Botnet Linked to Internet Outage Flashpoint. https://securityledger.com/2016/10/mirai-internet-of-things-botnet-linked-to-internet-outage-flashpoint/

[11] Shaila Sharmeen, Shamsul Huda et al, "Malware Threats and Detection for Industrial Mobile-IoT Networks", Special section on security and trusted computing for industrial internet of things, IEEEAccess, DOI 10.1109/ACCESS.2018.2815660.