Asfiya $ha!kh

Follow 434 Followers About

Web Services & API Pentesting-Part 1

Asfiya $ha!kh May 7, 2019 · 5 min read

Hey Pentester, I am back with my series of blogs.

This blog series will ride you through what is a web service and API and how the attacks can be performed and re-mediated on them.

So, what is a web service?

And here comes the bookish thing…

Web service is a technology to communicate one programming language with another. For example, java programming language can interact with PHP and .Net by using web services. In other words, web service provides a way to achieve interoperability and platform independency.

In simple words, I would say we have a remote database which needs to be used by a web based application(made in .NET, Java, PHP etc) and also an android or IOS application.So for 2 different platform to use the same database, we will need to code in .NET/Java/PHP and mobile language to fetch data from remote database.However this is not recommended, as we follow code reuse methodology. And there comes the use of web service. Web service is intermediate to the front end and the backend. We will code the data fetching functions in web service and will just call these functions through our web application and mobile

application, instead of writing data fetching code for each of the different platforms. Web service will only interact with the database.

For example, How do each of the trading applications know the share value of a stock? When they don't know the value stored by NSE or BSE database.Then how the share value is the same everywhere?

The thing here is all the applications uses web service function calls given by NSE or BSE to fetch the data from their database.

There are three major Web Service Components.

1. *SOAP*
2. *WSDL*
3. *UDDI*

Refer to the link below for more information on components

https://www.javatpoint.com/web-service-components

Now, What is an API (Application programming interface)?

Here is a small primer on what an API is, APIs allows applications to communicate with one another.

People generalize it to be something web based that returns data. However, the API is not a database. It is just an access point to an application that can access database.

So why do we need an API?

Suppose you have an application that needs to display the most popular hacking news on your app related to #RCE.So you request the same from hackernews.com. Then you integrate the same data in your database of the application.But next day the most popular hacking news with #RCE changes.So you will have to again ask the data from hacker news and again integrate in your environment.

However this is a huge work, So it would be better for hackernews to provide a way to query their application's search results through an API so that you can use the same in your application.

Web Service vs API

API and Web service both serves as a means of communication. The major difference is that a Web service allows interaction between two machines over a network to obtain platform independency. An API whereas is an interface between two different applications so that they both can communicate with each other.

You must be having a vague idea about types of Web Services i.e. SOAP and REST

Refer to the link below for the difference between SOAP and REST

https://www.upwork.com/hiring/development/soap-vs-rest-comparing-two-apis/

There are some vulnerable machines for vulnerable web services that we can setup for practice.

- *DVWS*
- *PenTester Lab: Axis2 Web Service and Tomcat Manager*
- *OWASP Mutillidae*
- *OWASP WebGoat*

Let's have a look at Manual Pentesting tools — SOAPUI Free & Postman

Later we can check out Automated tools and Extensions:

Automated tools:

SoapUI Pro, OWASP ZAP, IBM AppScan, HP Webinspect, WSBang, WSMap, WSDigger

Extensions:

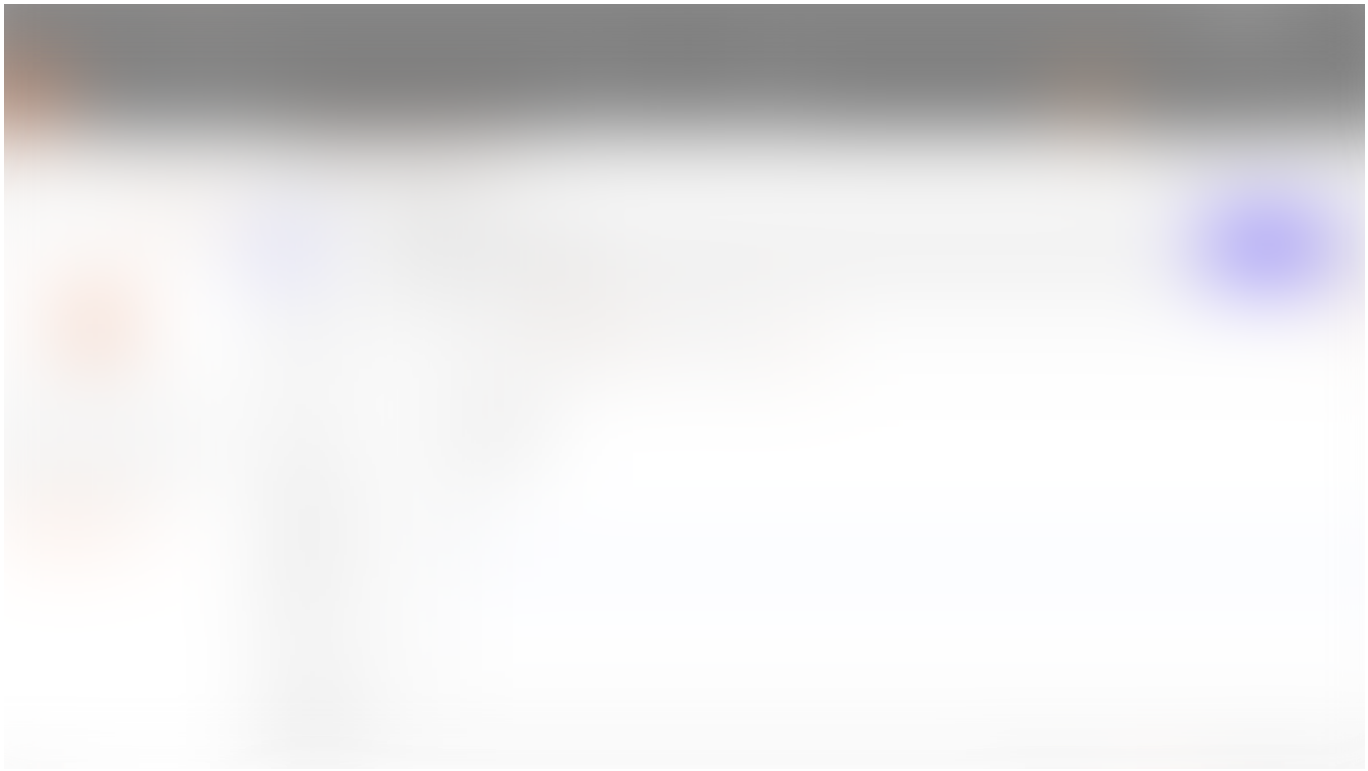
SAML Editor, SAML Encoder / Decoder, WSDL Wizard, Wsdler, SOA Client

Postman -

Basic use of Postman is to send SOAP or REST based requests and get the response

Let's configure postman for the same.

1.Set your HTTP request to POST.



2.In the request URL field, input link

3.Set Authorization if any

4.Set headers if any

5.Insert body in the respective format for example raw as shown

6.Click Send to receive response.

Also we would need to capture the request through burp suite proxy to make extensive use of burp suite functionality.

Now, let's setup a proxy

7.Click on Setting

8.Setup following values

9.Allow burp suite to listen on the configured host and port as shown

10. Start intercepting in burp suite then click on the send button in postman to get the request captured in burp proxy.

11.Once the request is captured, we can tweak around for possible responses and can check for different vulnerabilities an API may have.

SoapUI free -

To create a new SOAP project

1. Select File > New SOAP Project.

2. Specify a name for your new project and WSDL file that SoapUI will use for the initial configuration, then select the necessary options. Click OK.

To create a new REST project

1. Select File > New REST project.

2. Specify a URI to use for the project generation in the subsequent dialog and click OK.

OR

1. You can also click Import WADL to switch to the New WADL project dialog.

To setup proxy in SoapUI, click on settings then go to proxy settings.

> *Click Here to Continue Reading Part-2 of this Blog Series!*

Cybersecurity Penetration Testing Owasp API Pentesting