PenTest magazine

AUTOMATING API PENTESTING

Nessus, Yuki Chan, Static And Dynamic Analysis

The Real Key Is To Create 'Cybersecurity Culture' In The Workplace

an interview with dr. Jane Leclair

Automated Source Code Review

With Fortify SCA

The Commoditization Of Penetration Testing

AND MORE...

04/2018 ISSN: 2084-1117

EDITORIAL TEAM

Managing Editor

Bartłomiej Adach

bartek.adach@pentestmag.com

Proofreaders & Betatesters

Lee McKenzie, Samrat Das, Olivier Caleff, Ali Abdollahi, Craig Thornton, Tom Updegrove, Matthew Sabin, Da Co, Robert Fling, Benjamin Aboagye, David Molik, Abhishek Kar, Wayne Kearns

Special thanks to the Proofreaders & Betatesters who helped with this issue. Without their assistance there would not be a PenTest Magazine.

Senior Consultant/Publisher

Paweł Marciniak

CEO

Joanna Kretowicz

joanna.kretowicz@pentestmag.com

DTP

Bartłomiej Adach

bartek.adach@pentestmag.com

COVER DESIGN

Hiep Nguyen Duc

PUBLISHER

Hakin9 Media Sp. z o.o.
02-676 Warszawa
ul. Postępu 17D
Phone: 1 917 338 3631
www.pentestmag.com

PenTest magazine

Dear PenTest Readers,

We are very proud to present you the new issue of our magazine. This time, we would like to concentrate mainly on the aspect of automation in penetration testing. Therefore, we are in a sense continuing the idea of our previous issue, which is analyzing the role of the AI in penetration testing. Not only are we aiming to present specific analyzes of automated pentesting processes or how they exceed beyond the limits of manual pentesting, but we also try to answer questions that the presence of automation in the penetration testing field evokes. For instance, what are the implications for the infosec business and the job market? What is the future of penetration testing in general, and to what extent the role of penetration testers can be diminished by AI? These are some of the aspects that we would like to deal with this month.

We are extremely happy to publish a piece written by Chrissa Constantine, the second issue in a row. This time Chrissa provided a comprehensive analysis of the Automated API Testing. This is another 'must read', especially if you are interested in web services, which have become more popular in penetration tests.

Furthermore, we are truly honored to include an interview with Dr. Jane LeClair, who is the President and CEO of the Washington Center for Cybersecurity Research & Development. The interview covers a number of topics within the cyber security field - the idea and activity of the institute, security culture in the workplace, education, the role of women in IT, and the cybersecurity of the particular industrial branches. Dr. LeClaire's remarkable experience and broad perspective is a guarantee of an interesting conversation.

Speaking of the role of women in the field of cyber security, we would like to draw your attention to Nouha Ben, who is a 15-year-old Computer Science student and has her debut publication in this issue. We firmly believe that young talents should be promoted and supported.

Harpreet Singh provides an extended tutorial on the automation in penetration testing on the examples of Nessus, Yuki Chan, static and dynamic analysis. If you are looking for technical tutorials you will be satisfied with this one!

We are pleased about the fact that one of the authors in this issue is Professor John Walker, who provided us with a historical perspective on penetration testing (which contains some very interesting case studies), as well as the outlook for its future, with an emphasis on the role of the OSINT. A broad perspective is something we always appreciate, as we believe that only reflective approach can provide us with a better understanding of on-going changes and sensible prospects for the future.

Haydn Johnson wrote a thought-provoking article about the commoditization of penetration testing, and certain risks caused by automation in the context of the relation between companies and pentesters. This subject is also mentioned in our second interview of the issue. We talked with Mr Falgun Rathod, who has delivered over 100 seminars and training in various colleges and corporates across India and is listed on the Top ten Ethical Hackers of India & Top Ten Cyber Cops of India.

In the current issue, you can also find pieces about automated source code review with Fortify SCA, the implementation of automated penetration testing in a company, and cryptocurrencies and regulations.

Alright, that's enough spoilers!

Enjoy your reading,

PenTest Magazine's Editorial Team

Contents

Automating API Pentesting

Chrissa Constantine

Chrissa is an Information Security Analyst and has a Master of Science in Information Security, CISSP and CE|H certifications. She held positions as a consultant at Apple and for a Silicon Valley start-up as a penetration tester. Chrissa enjoys hacking competitions, meeting new people, and learning new things.

There is considerable value in automating portions of API pentesting. Commonly pentesters open the web application and navigate to all of the pages, capturing the requests and responses in a security testing tool like Burp or OWASP Zap. The use of API testing tools like SoapUI or Postman can help pentesters generate and submit web service requests. For SOAP calls, the WSDL can be challenging to read and derive manual tests. Tools that can be used to point to a WSDL or Swagger file (REST) are essential to use so that testers can work more efficiently. It is essential to spend time setting up the testing environment in preparation for analyzing the API.

Introduction

SOAP & REST

Web services allow applications to share functionality and allow consumers to access data without the application needing to know the format or location of the data. (Najera-Gutierrez & Ansari, 2018) There are two ways to develop web services, Simple Object Access Protocol (SOAP) and Representational State Transfer (REST).

SOAP was the traditional means to develop a web service, but many applications now use RESTful web services. SOAP web services use Extensible Markup Language (XML) to exchange data, whereas RESTful web services primarily use JavaScript Object Notation (JSON). SOAP web services are sometimes used due to Web Services (WS) extensions such as WS-Security or WS-Addressing to provide secure and reliable communications, but often RESTful web services are easy to implement and, as such, preferred.

SOAP has an envelope that defines the framework for describing the message and how to process it, encoding rules and conventions for representing remote procedure calls and responses. (Kankanamge, 2012) The WSDL describes application-specific messaging requirements. The elements of a WSDL need to provide a machine-readable description of where the service is reached, what parameters to expect, the actions at the location, the format for the messages, and the functionality offered. (Bustamante, 2007)

Several Web Services protocols for SOAP extensions facilitate communication requirements for security. (Bustamante, 2007) SOAP was designed to be extensible, but there are so many WS protocols that it is challenging to know which standards to use. SOAP is transport independent and can be transported over SMTP, JMS, or FTP (to name a few examples). However, SOAP messages are also transferred over HTTP and HTTPS. SOAP and WS allow developers to expose services over any protocol, but for purposes of penetration testing, HTTP or HTTPS are the most commonly tested protocols.

Advantages of SOAP include additional assurances for data privacy, integrity and security through WS-Security extensions. SOAP has built-in logic to compensate for failed communications. SOAP is extensible through other protocols and technologies. In addition to WS-Security, SOAP supports WS-Addressing, WS-Coordination, WS-ReliableMessaging, along with other web services standards, a full list of which is on W3C. (Stackify, 2017)

RESTful services have many of the same vulnerabilities as a standard web application. (Ansari, Imran, Kotipalli, Halton, & Weaver, 2017) REST is stateless, but SOAP is stateful, which is why it supports financial services, e-commerce payments, or telecommunications. (SoapUI, 2018) Use REST when information is read and written at a resource level, such as for online photos, social media, ordering, or reading information from servers or databases. (SoapUI, 2018)

REST uses JSON, which has key/value pairs that represent data carried inside the message. (Kankanamge, 2012) REST can be represented with a unique ID via the uniform resource identifier (URI), uses standard HTTP methods (GET, POST, PUT, DELETE), and links resources together. Named information is considered a resource identified with a unique ID via a URI. The identifier helps locate resources on the web server. The uniform resource locator (URL) is the most common URI on the web today. (Dash & Aroraa, 2018)

For purposes of this paper, REST API testing is the primary focus. However, many techniques for REST apply to SOAP testing and the tools discussed can be used for testing both types of web services.

REST advantages include no platform or programming language dependencies, standardized HTTP methods, stateless, accessible to various clients, such as web, desktop or mobile. The disadvantages for REST include lack of documentation due to no metadata, security concerns and a lack of standards, which means clients have a hard time understanding. (Dash & Aroraa, 2018)

Why Automate?

Security issues manifest in various ways, and attack vectors that impact API testing are related to web application testing. Often web services security testing is performed at the end of the development lifecycle. Security testing web services can be problematic when it is too late in the process to correct issues before the public release of the service. There is a debate about when to integrate testing, and in some cases, the security integration occurs within the release cycle as a part of continuous integration. Adding penetration testing to the development lifecycle for web services ensures that issues can be found and remediated before the public release of the API.

The assumption is that many development teams already write and run automated functional testing for their APIs. If this is the case, then adding security testing as part of the traditional development and QA process, can enhance the security of a web service.

There is considerable value in automating portions of API pentesting. Commonly pentesters open the web application and navigate to all of the pages, capturing the requests and responses in a security testing tool like Burp or OWASP Zap. The use of API testing tools like SoapUI or Postman can help pentesters generate and submit web service requests. For SOAP calls, the WSDL can be challenging to read and derive manual tests. Tools that can be used to point to a WSDL or Swagger file (REST) are essential to use so that testers can work more efficiently. It is essential to spend time setting up the testing environment in preparation for analyzing the API.

There are challenges to testing RESTful web services because often the application does not reveal the full attack surface. Many URLs and parameters used by RESTful web services are not directly exposed, and applications typically

do not fully utilize all of the functionality. The parameters in web services are not standard, which makes it hard to expose and test the API. Moreover, the number of parameters can be significant, increasing the test time. (Shezaf, 2017)

The autonomous nature of web services means that there is a higher degree of scalability and extensibility to the web application. (Kankanamge, 2012) Additionally, not all web services are built in-house, which means that third parties may host them. The testing of these services can become complicated due to all of the interactions and integrations between internal and external services.

With many organizations use of a form of Agile methodology, testing automation is often a requirement for software development. Test automation refers to the ability to run a repeatable test against the application. The advantages are related mostly to the repeatability of tests and speed of test execution. It may not be advantageous to rely on automated testing depending upon application or test, such as a check for logic flaws, which requires manual efforts from the pentester. It is up to the pentester to determine what can be automated and what needs to rely on manual efforts.

Advantages of automation include:

- **Speed and Coverage**

Automated tools take less time to use than manual efforts. Manual testing never covers all aspects of the application attack surface. Using an automated tool and then manually reviewing and checking reported vulnerabilities is faster than manually trying to discover and test all aspects of an application or API.

- **The number of tests**

Automation allows a tester to send a large number of attacks at a target and use various payloads. It is way more time consuming to send attacks one at a time.

- **Skills, and reporting**

It takes less skill to send automated attacks at an API than performing a manual test. However, it is not advisable to use or rely upon the results from a tool without having a pentester review them due to false positives from tools. Additionally, most automated tools have reporting features built-in. Reporting in automated tools also is a time saver for testers who are not able to spend the time manually documenting every issue.

Discovery

Depending upon the type of penetration test, the tester may have to spend time discovering the API endpoints. A list of common endpoints include:

- /api
- /v1
- /v2
- /v1.0
- /v10
- /api/v1
- /api/v2
- /rest
- /rest-api

- /ping
- /health
- /status
- /metrics
- /trace
- /log
- /logfile

It is advisable to find documentation about the API to obtain insight into the structure of the web service. Documentation helps the tester understand user roles, request methods and responses, and can support reporting of vulnerabilities that align to business use cases and context. Here are some common locations where a pentester can find documentation:

- /api-docs
- /swagger-ui.html (REST)
- /swagger.json (REST)
- /doc
- /application.wadl (REST)
- /application.wsdl (SOAP)

One way to discover endpoints is to use a security list and an intercepting proxy tool like OWASP Zed Attack Proxy (ZAP) or Burp to iterate over possible endpoints. Using tools is faster than manually trying to find endpoints in a browser. Although, tester interaction with the front-end UI may be the only time API calls are exposed. Missing critical web services can be easy. In all pentesting scenarios, the basics of information gathering are useful for mapping the API and supporting later testing tactics.

Tools

During API pentesting there can be numerous API requests, so a couple of tools should be set up to help with discovery and testing. They can include:

- OWASP ZAP
- Burp
- W3af (REST)
- SoapUI (SOAP and REST)
- Postman (REST and SOAP)
- Advanced Rest Client (https://install.advancedrestclient.com/#/install) – this was a browser-based REST client but is now for the desktop
- Restlet Client (Chrome, formerly known as DHC)
- RESTClient (Firefox)

Some tools have both free and paid versions, such as Burp and SoapUI. SoapUI Pro version has additional functionality that provides automation and security testing. Burp Pro has plugins and features that make it more useful than the free version for API pentesting. However, it is up to the pentester to determine the best tools to use when testing web services. The most commonly used tools often have automation built in for running tests and performing security assessments. Otherwise, the tester spends much time with configurations and manual fuzzing and testing efforts. No penetration test is entirely reliant upon the automation provided through the tools. Instead, there typically is a blend of manual and automated testing.

The focus is on Postman and SoapUI for testing web services. Zap and Burp can be used with Postman or SoapUI to capture requests and use to fuzz or scan for vulnerabilities.

Pentesting Web Services

To test in an automated fashion, ensure the tools are set up, and scenarios or methodologies are in place. It is critical to be able to reproduce and document what the pentester is doing for an API test. Ad hoc testing is not sustainable and does not support the pentesting lifecycle. Having a test that is repeatable means that later testers save time and effort. If one tester performs an API test and has findings and later testers must retest the API, it is more challenging if there is inadequate documentation from the first tester. One method to help future testing is to take the time to build a Postman file or SoapUI project file and then export and have it available for future testing.

When pentesting web services, try to obtain API documentation and sample requests/responses for the web service. Otherwise, perform discovery and information gathering to obtain the WSDL (for SOAP) or Swagger (for REST) and build requests to the web services via discovered documentation. Tools, such as SoapUI, Postman or Burp (Pro with plugins), can import the WSDL and build the request. Postman and SoapUI can also import Swagger files. Moreover, SoapUI imports Postman files.

The target API needs to be analyzed to determine what type of authentication is in use. Depending upon the service it could be Basic HTTP authentication, cookies, or API access tokens. Some API services are public and do not require any authorization. For a typical penetration test, public open APIs with no authentication would probably not be the typical test scenario.

API calls with vulnerabilities are reportable. However, pentesters frequently are not provided context regarding the use cases or requirements for the web service. If possible, obtain as much documentation and information as possible before testing, such as use cases. The pentester can leverage scenarios and use cases to provide business context about the impact of security vulnerabilities on the application. In this way, the penetration test findings show the related business impact to the business owners.

Security testing scenarios are separate from functional API testing or QA testing scenarios. The penetration test should maximize the attack surface of the web services provided. Ensure inputs to APIs are analyzed and determine which API calls to test. Having an open-ended test makes it challenging for the test to yield optimal results. If the pentester partners with developers or with QA, the team needs to ensure API testing scenarios are manageable so that the security test does not run too long. Additionally, when testing, know the functionality of the API. For example, some API calls add data to the database, and it is important to remember the impact on performance when pentesting these web services. The team should consider various impacts when working with automated testing tools.

From an automation standpoint, here are some options for how security tests can be categorized and automated:

- Identify vulnerabilities using functional security testing. Target validation of various application features such as authentication or log out. Break the application down and run testing against all of the various functions. Automate using various tools in coordination with Postman and SoapUI, such as OWASP ZAP or Burp. Further automation can occur with Selenium or WebDriver, but it depends upon the test which tool is the most appropriate.

- Testing known issues. Test for misconfigurations of headers, session cookies, or SSL-related issues like weak ciphers. These tests are easy to check for using Postman and SoapUI with OWASP ZAP or Burp. Some other testing tools can

include BDD-Security (https://www.continuumsecurity.net/bdd-security/), Mittn (https://github.com/F-Secure/mittn) or Gauntit (https://github.com/GAUNTLT/GAUNTLT).

- Use the principles of functional or performance testing APIs to support security testing. Break down testing into manageable parts such as authentication, logic, and other areas of weakness in the application. Automated scanning can only go so far in testing logic flaws, so a tester manually manipulates the application flow to determine if there are flaws.

- Build a test strategy and try to automate and improve the test cases for business-critical applications.

Test Case Options and Attack Vectors

There are many reasons why a web service has vulnerabilities ranging from design and development errors to poor system configuration. Some areas to consider include:

Fuzz testing – Automation is required and works well with manual checks against potentially vulnerable parameters. Fuzz testing sends malformed data to the API to see if it breaks. When the API expects integers, send values that may be unexpected, such as negative or large numbers. A poorly designed API is reliant upon a specific format and may open the way to security flaws via error messaging or other improperly designed services.

Injection Attacks – Injection flaws occur when the application passes information from the HTTP request to another command, service, or system call. Injection attacks can include passing SQL commands, or operating system commands through the parameters in the web service. Any language interpreter can be at risk, including JSON, XPath, XSLT, and these technologies can be compromised. It is up to the tester to know about the API and how to choose the correct type of injection attack.

(Un) Authorized endpoints and methods – Web services should implement authorization if APIs expose sensitive data. Testing authorized endpoints without authorization and with incorrect authentication, or testing levels of user privileges is important. For RESTful web services, the HTTP protocol is stateless, and if the communication fails, the client must retry sending the request. The use of authentication via tokens helps programmatically achieve security. Token-based authorization is used to help a user access data or resources over a defined period. (Dash & Aroraa, 2018)

Parameter tampering – Manipulate parameters for the API request to take advantage of validation errors. Modify and tamper hidden fields and query parameters within the API request along with testing various HTTP verbs and methods.

Cross-Site Scripting (XSS) – Cross-Site Scripting is not just applicable to web applications, but to web services as well. If the API reflects input, then the determination of how the API or client handles the input needs to be determined. XSS attacks are injection attacks, and the vulnerability needs to be assessed even for the API test. (Lensmar, 2014)

File uploads – This is another area to check to determine whether the system can process files safely. If the application requires a PDF but receives a shell script, it is essential to know if the file is executed or handled in a way that does not expose the system.

Exploiting a web service can be destructive to a business who has public APIs. More and more APIs access sensitive data that can put a business at risk. Not all security vulnerabilities are stoppable, but without security testing APIs, there is no prevention.

Getting Started with Postman

Many options are available for pentesting web services, but the focus is on Postman for REST testing and SoapUI for SOAP testing. Each of these tools can import and test both SOAP and REST web services.

Start by downloading and installing Postman desktop client at the following URL: https://www.getpostman.com/apps Install Postman on Mac, Windows or Linux platforms.

Open the application, and it displays an area called *My Workspace* in the background and a window with *Create New* options (Figure 1). If the tester does not need to *create a new* collection or add a request after opening the application, then, close the open Create New window. Otherwise, it may be required for the tester to create a Collection manually. Collections are groups of requests that can be used to run one after another for purposes of automating testing.

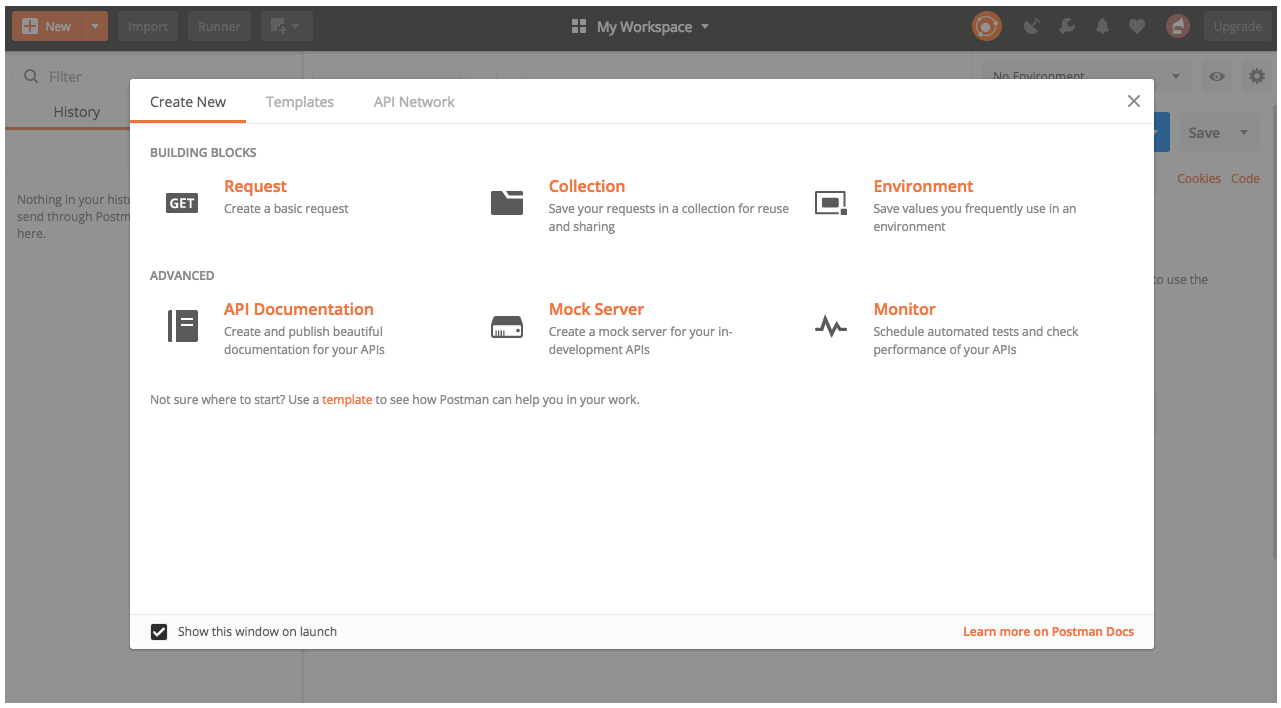


Figure 1. Starting Postman displays the My Workspace (background) and Create New Window

Several choices are available to import API calls into Postman. If testing a couple of API calls, one option is to start by creating a *Collection*. Collections are groups of requests and are used to keep all of the API calls for one test organized in a folder. This step requires the tester to build a new collection and then import the API calls into Postman. Postman imports various files or types of data, such as Swagger, a URL, or a cURL command, as shown in *Figure 2.*

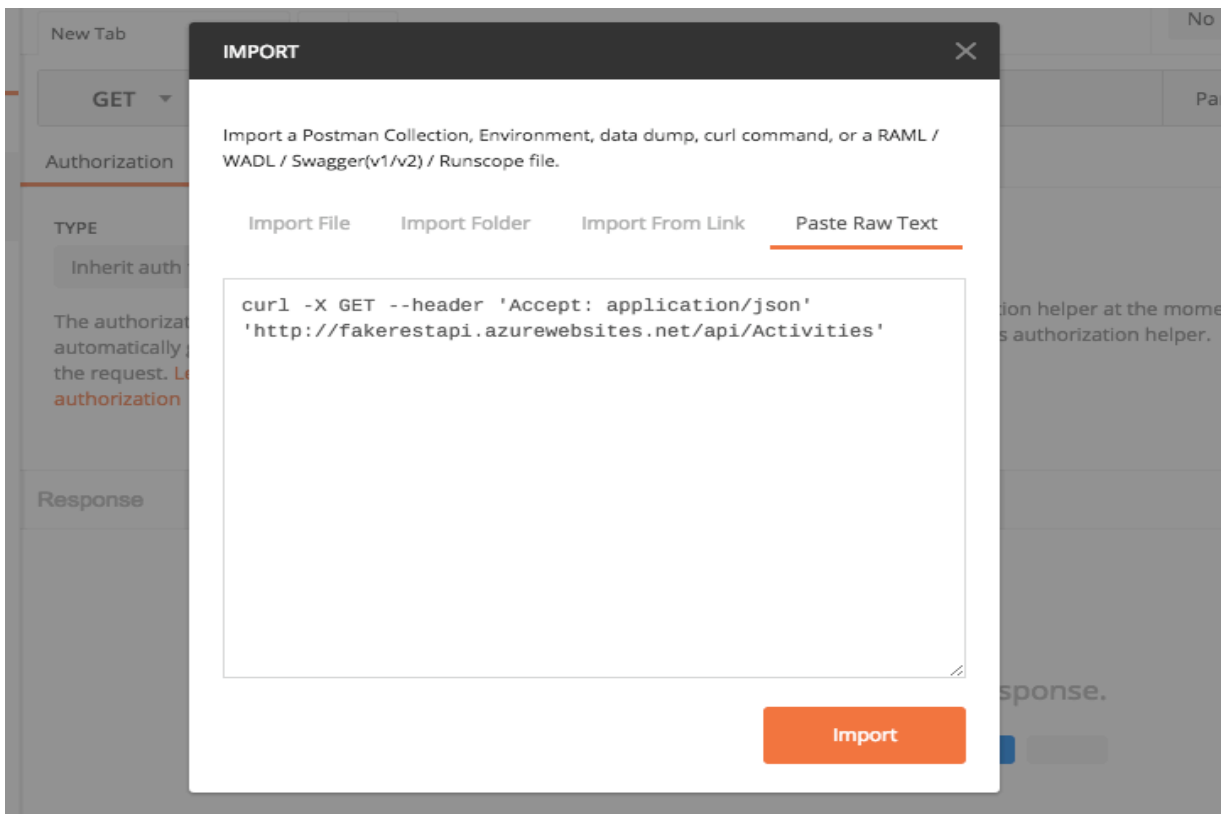


Figure 2. A sample of an import from a cURL command to Postman

For purposes of this test, FakeRestAPI was used to import API calls into Postman. FakeRestAPI has 27 API calls available for testing purposes, located at this URL: http://fakerestapi.azurewebsites.net/swagger/ui/index#!/Activities/Activities_Get

Navigate to the FakeRestAPI URL in a browser (Figure 3) to review options, such as using cURL, or testing the response directly from the site. For this test, the REST API calls from FakeRestAPI is accessible in the browser, but in some tests, the easiest way to understand the API is to review information from within a tool such as Postman, SoapUI or possibly Burp or Zap.

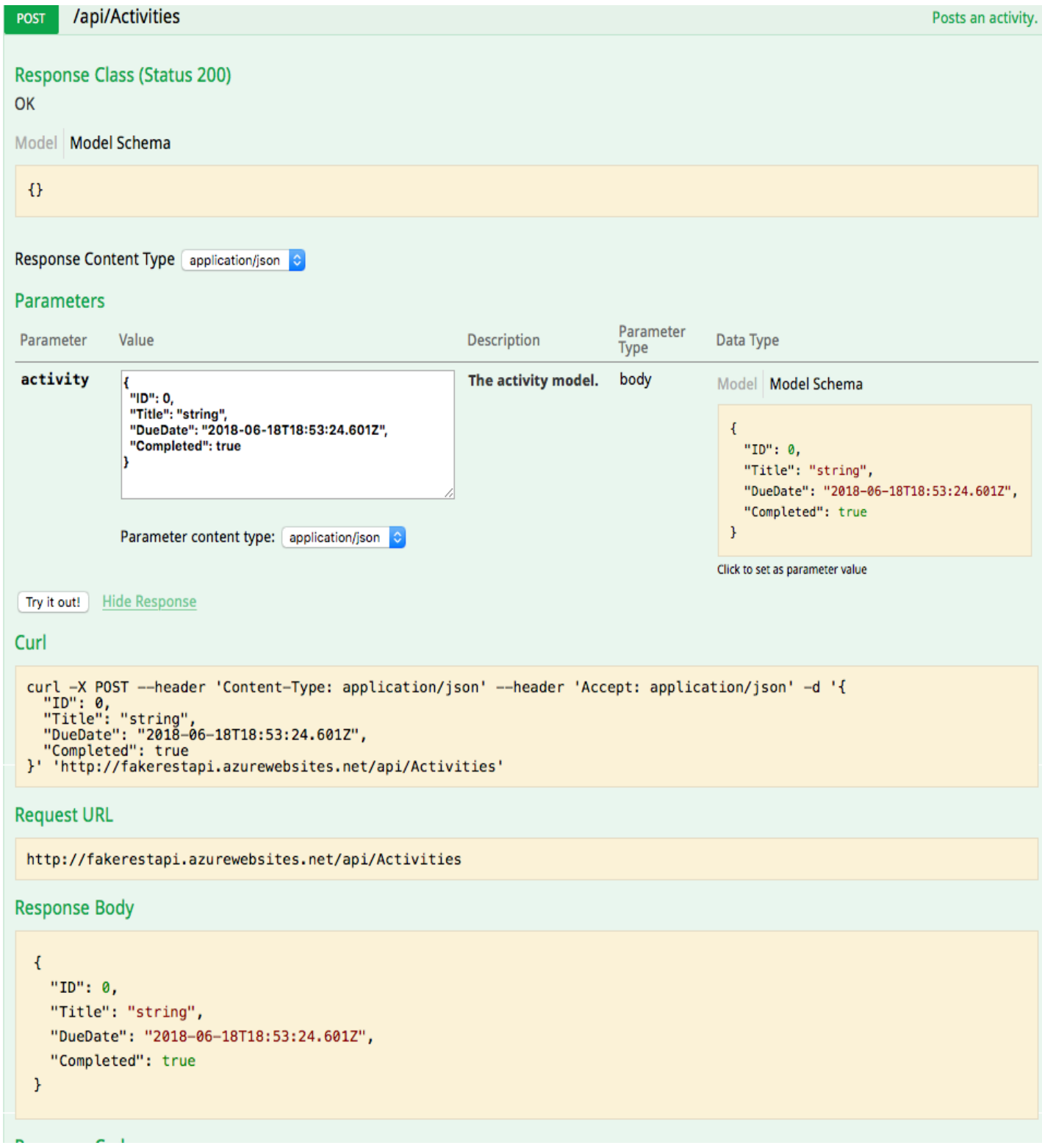


Figure 3. Activities API call accessed from the browser with Model Schema and cURL commands

In *Figure 3*, opening the browser to determine if the information is available was useful because it reveals a set of data that is already configured to return a successful API call for the service. The format is pre-built in this example, so the tester has less work to do in discovering how to use the API call.

In some penetration tests, an API key or authorization may be required to access to web services, but for the sample FakeRestAPI, no API key is required, and all 27 API calls are publicly available.

Postman can generate authorization headers, and the tester should understand what type is in use by the web service. It is beneficial to understand the type of authorization used by a web service so the tester can determine if the implementation is flawed and can be leveraged to expose additional vulnerabilities. Several options exist in Postman to configure authorization, such as bearer tokens, basic or digest authentication, OAuth, Hawk, AWS signatures or NTLM.

If the test has an authorization requirement, there are options available in Postman (and SoapUI) to configure automation to re-use session tokens for subsequent API calls. Both tools allow the tester to configure automation in using data from one request to add it into another request.

In Postman, variables are used to scrape session tokens from the request body of the API call. The tester does not have to manually update the authorization tokens for the subsequent API calls in the test. Additionally, Postman environments help testers configure a set of variables used for a specific website. If the test spans multiple sites, the tester can quickly switch between environments in Postman to use different variables.

After reviewing the FakeRestAPI, decide the best way to import data to Postman. Use the URL to add all of the API calls into a Collection. There is no one right way to perform the initial setup because it depends upon how the web service was created to determine how to add information into Postman. *Figure 2* shows another option for data import into Postman in the form of a cURL command. This command was copied from the site and generates a GET request for the Activities API.

Use Collection Runner (aka Runner as shown in *Figure 6*) to determine if the 27 FakeRestAPI calls are correctly configured. Runner sends all of the API calls in the order they are in when the calls import into the Collection. After importing API calls into Postman and using Runner to check status, several API calls failed. The POST request /api/activities (from *Figure 3*) requires JSON data in the request body, which did not import when creating the Collection from URL.

Navigate back to the URL. There is a sample in *Model Schema*, which displays on the right side of the page. Click the Model Schema to set the values as parameters in the body. Then, click *Try it out!* to display the response in the browser.

Using the browser is one way to determine how the API call works. If choosing to test in this manner, the tester can configure a proxy in the browser (i.e., FoxyProxy) to then capture the traffic into OWASP ZAP or Burp for further testing. However, in the spirit of automation, a bulk import the API calls to an API testing tool such as Postman or SoapUI is preferred.

Within the Postman import, some options require more time on the part of the tester to configure, such as copying and pasting the cURL command from the FakeRestAPI website and saving it to the collection created when first opening the application. However, there is a faster way to perform this action. Using the link at the FakeRestAPI website directly imports the entire collection of API calls into Postman.

Click *Import*, then *Import From Link* as shown in *Figure 4*. If the tester picks import from a link, Postman creates a new Collection in *My Workspace* with all of the API calls from the FakeRestAPI site.

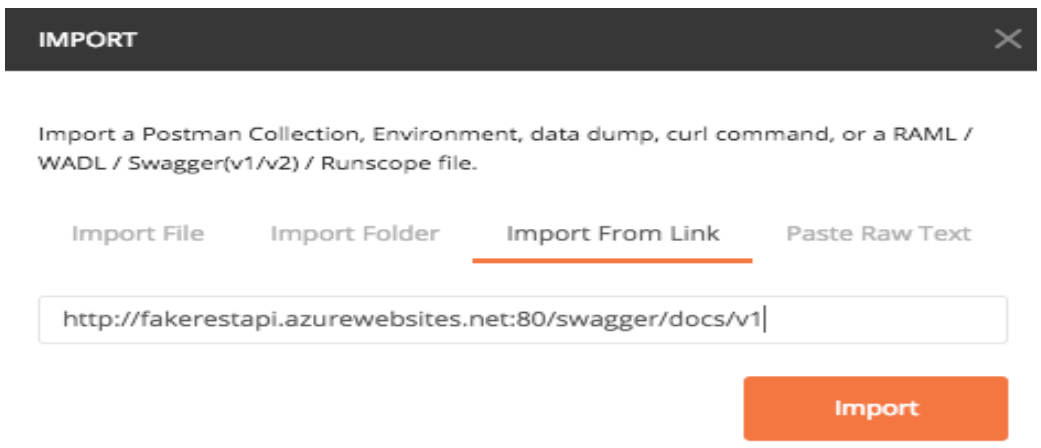


Figure 4. Automate importing a Collection

Once the import completes from FakeRestWeb API, 27 requests from the site load into Postman as shown in *Figure 5.*

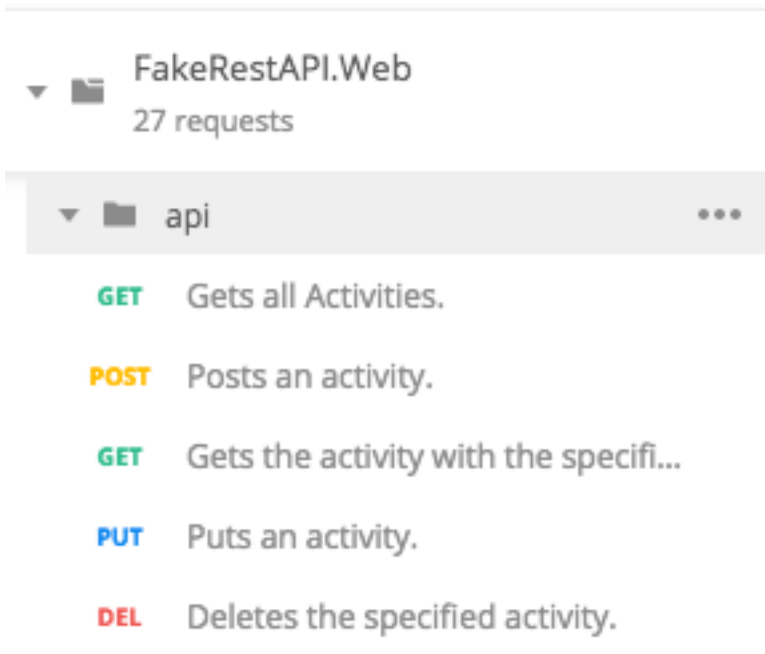


Figure 5. A sample of the 27 FakeRestAPI API calls loaded into Postman

With all of the API calls loaded, one option is to use the Postman *Collection Runner* to determine the status of each API call. Using Runner via proxy to either Zap or Burp is useful to continue testing the API web services for other vulnerabilities. Collection Runner can be used to quickly determine if all of the imported API calls returned a 200 OK (*Figure 7*).

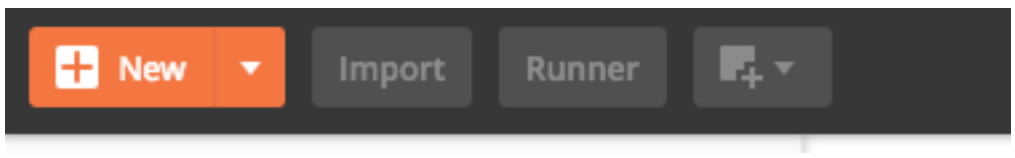


Figure 6. Location of Runner in the toolbar

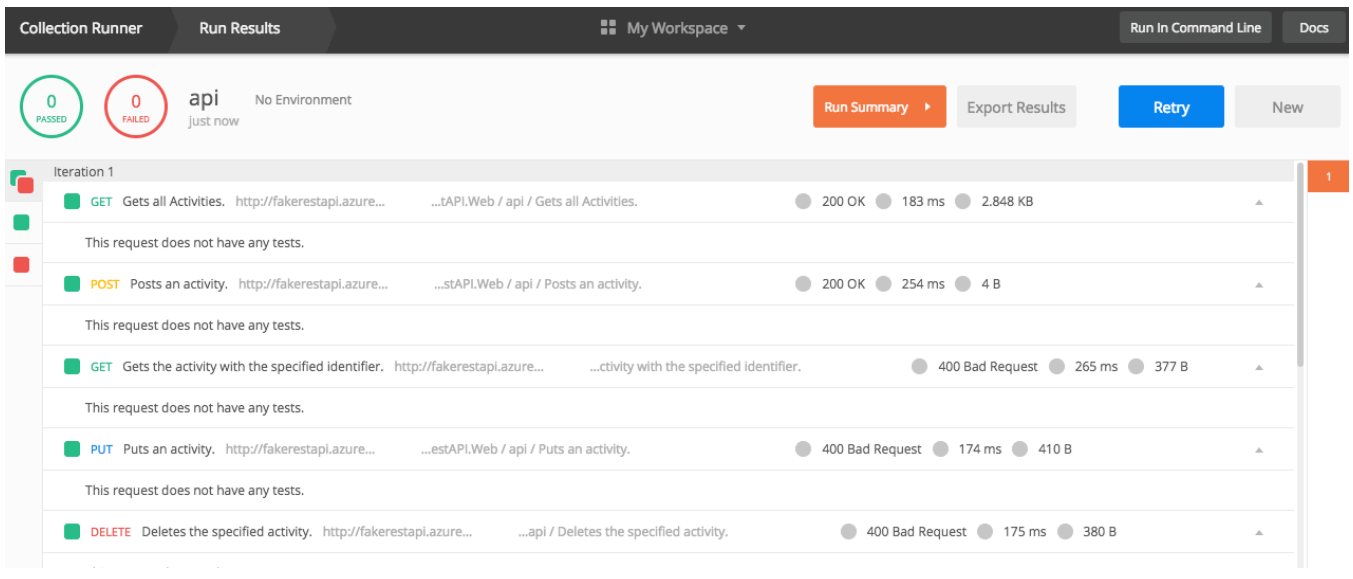


Figure 7. Sample API calls from Runner

In Figure 7, some of the API calls failed with a 400 Bad Request message, which requires further examination to determine corrections to the requests before continuing testing. Get a baseline of API functionality with working request/responses so that you can then move into the pentesting steps.

Review of the failed API calls in Runner looks like data must be added to get the Collection of API calls to send 200 OK responses. At this point, start configuration of the Postman environment and updates to the imported API calls.

Postman Variables, Environments, and Scripts

A significant feature of Postman is the ability to use variables and set Environments. By configuring Postman for different environments, and by using variables, data can be extracted and reused from responses within a collection to chain requests together. Postman has a "runtime based on Node.js" and allows a tester to configure tests and build API calls with dynamic parameters. (Postman, 2018) These tests can be sent either before or after the request and are used to create workflow and automation.

There are two areas to understand in the Postman UI, one is the *Pre-Request*, and the other is the *Tests* tab. Scripts in the *Pre-Request Script* tab execute before the request goes to the server. Scripts in the *Tests* tab execute after receiving a response from the server. Both of these tabs use scripts written in JavaScript. Also, both have code snippets available for testing purposes.

In cases where a timestamp should be in the request header or when a refresh or Bearer token is required use a Pre-request script. The FakeRestAPI does not require authentication, but if the test had endpoints that required tokens, it is easy to use Postman to obtain the token and populate it in the header of another API call for re-use.

The steps would be as follows:

- Create variables for the token
- Use the Pre-Request Script to check the token and get a fresh one if it expired
- Go to the Authorization tab and set the token to the {{currentAccessToken}}, which is the value obtained when using the *Pre-Request Script*.

The code would be added to the *Pre-Request Script* tab and modified for the correct URL and test information. Sample code that could be used to extract a Bearer token is here (needs to be modified to work): https://gist.github.com/bcnzer/073f0fc0b959928b0ca2b173230c0669

Figure 8. Pre-Request Script tab

To access variables requires setting up an environment. In the request builder, anywhere the tester uses text Variables can be used, such as headers, authorization, the request body, and the URL or URL parameters. (Joyce, 2017) String substitution is used to replace variable names enclosed in double curly braces like {{variableName}}. In *Figure 9*, the {{id}} is red and when hovering over it, the application displays a tooltip that shows it is an undefined variable.

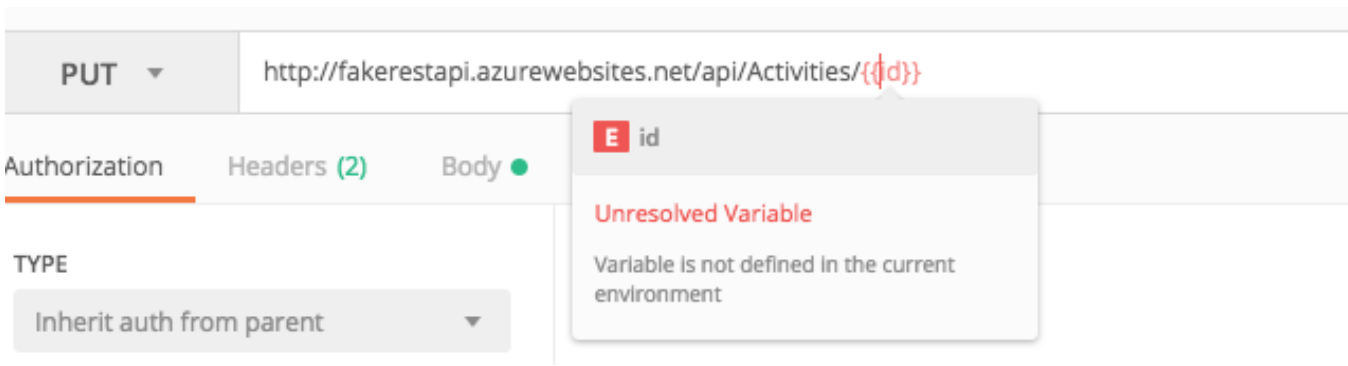


Figure 9. Undefined variables

For this test, to add automation to scrape the ID from the response body, add the following script in the *Test tab*:

```
var jsonData = JSON.parse(responseBody);

postman.setEnvironmentVariable("activity",jsonData[0].ID);
```

The idea behind this configuration is to add the test in the first request that returns the activity ID. Then, add an Environmental variable.

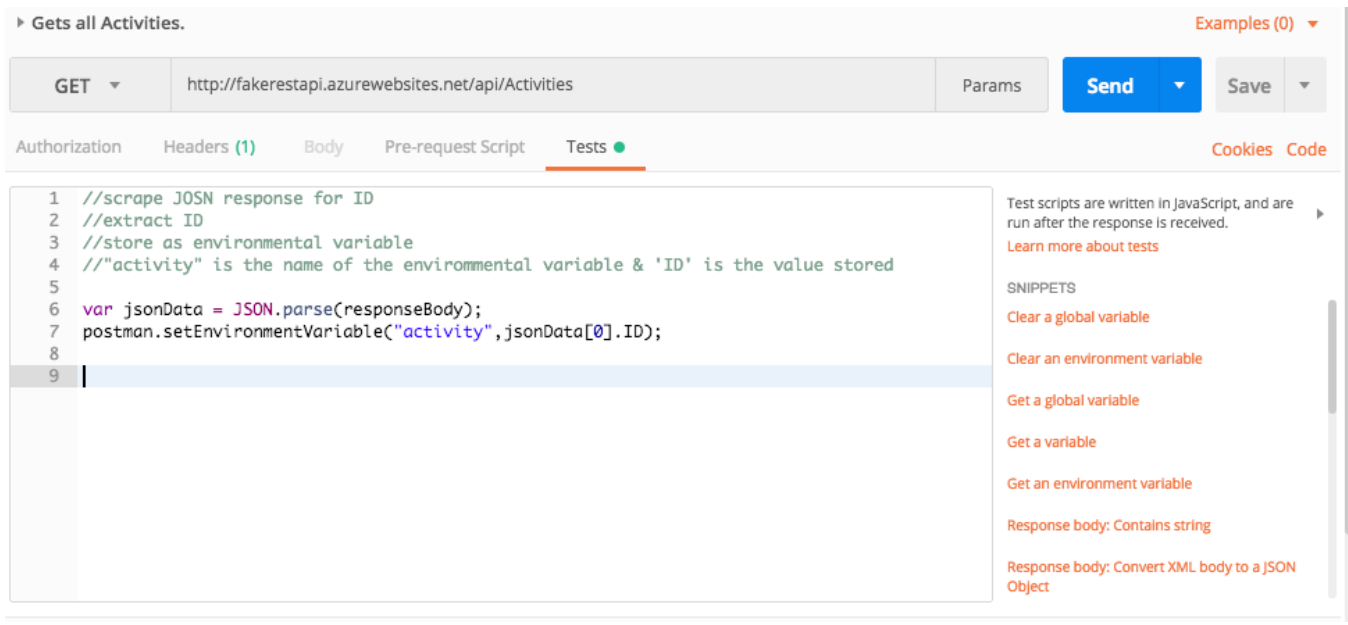


Figure 10. Adding a Test script

Go to the first API call to get all activities and add a test script. This API is returning an array, so add the location in the JSON array.

Next, go to the Environment and add a new variable. Once the request runs, Postman auto-populates it with the scraped value from the API request.

Environments are key-value pairs that use the key as a representation of the variable. (Postman, 2018)

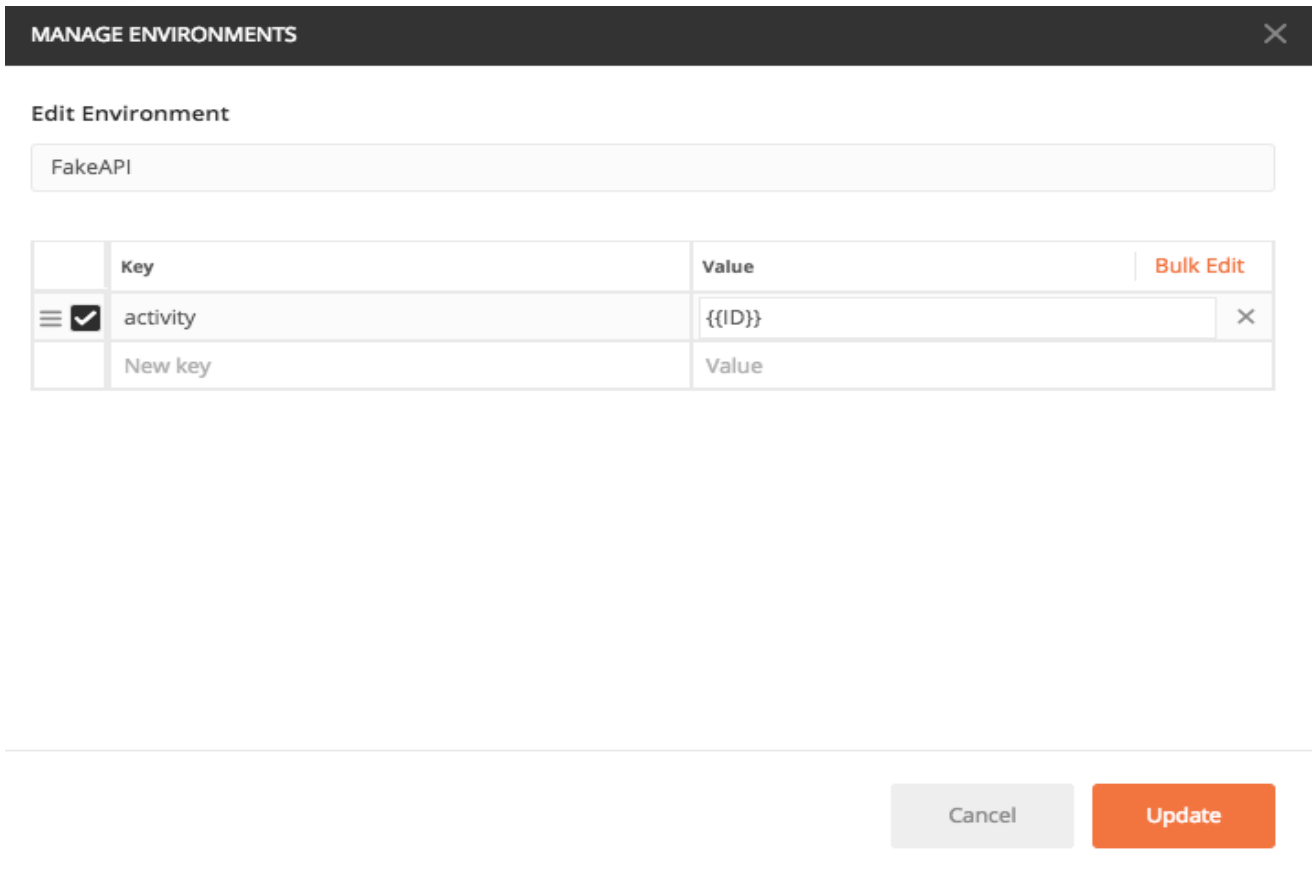


Figure 11. Adding an Environment and Variable ID

Figure 11, sets the key to the value in the Test tab ("activity") and set the Value to the JSON data scraped from the request. In this case, it is the variable *{{ID}}*. *In Figure 12*, when sending the API call, the Value gets replaced, and an ID number displays in place of the variable.

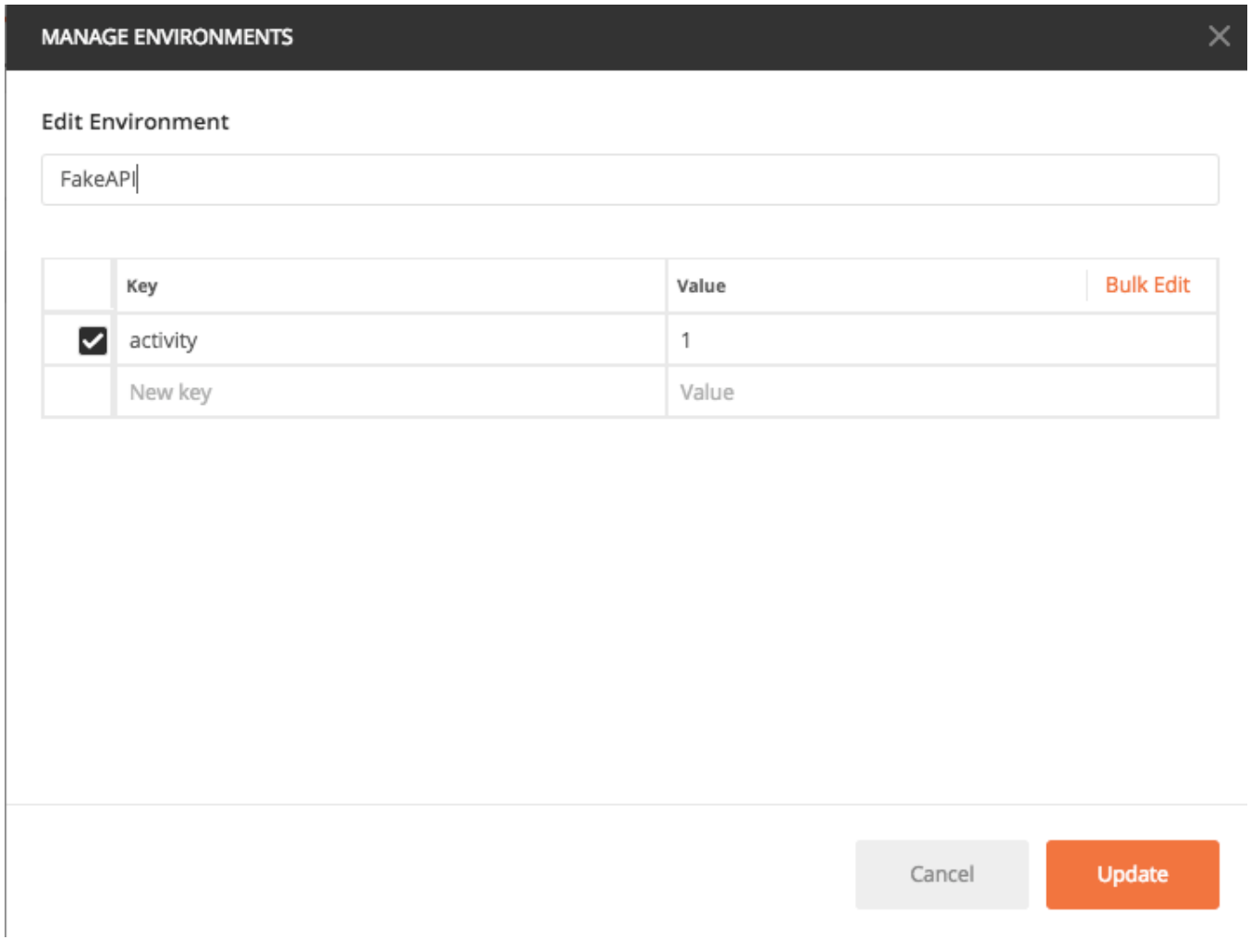


Figure 12. Activity Key populates with a value because of the test script

Configure the Environment, variables, and any Pre-request scripts or test scripts before initiating a test run through all of the API calls using Runner. Errors are immediately apparent in the Runner window, which makes it easy to pick out what needs to be corrected to get a complete test.

SOAP Requests in Postman

To create a SOAP request, enter the SOAP endpoint as the URL or enter the path to the WSDL as the URL. In my example, I created a new POST request and pasted the WSDL URL. Next, click *Body* and select *raw* and set it to *XML (text/xml)*. Define the SOAP Envelope, Header, and Body tags as required. Here is an example WSDL that I used: http://www.holidaywebservice.com//HolidayService_v2/HolidayService2.asmx?wsdl

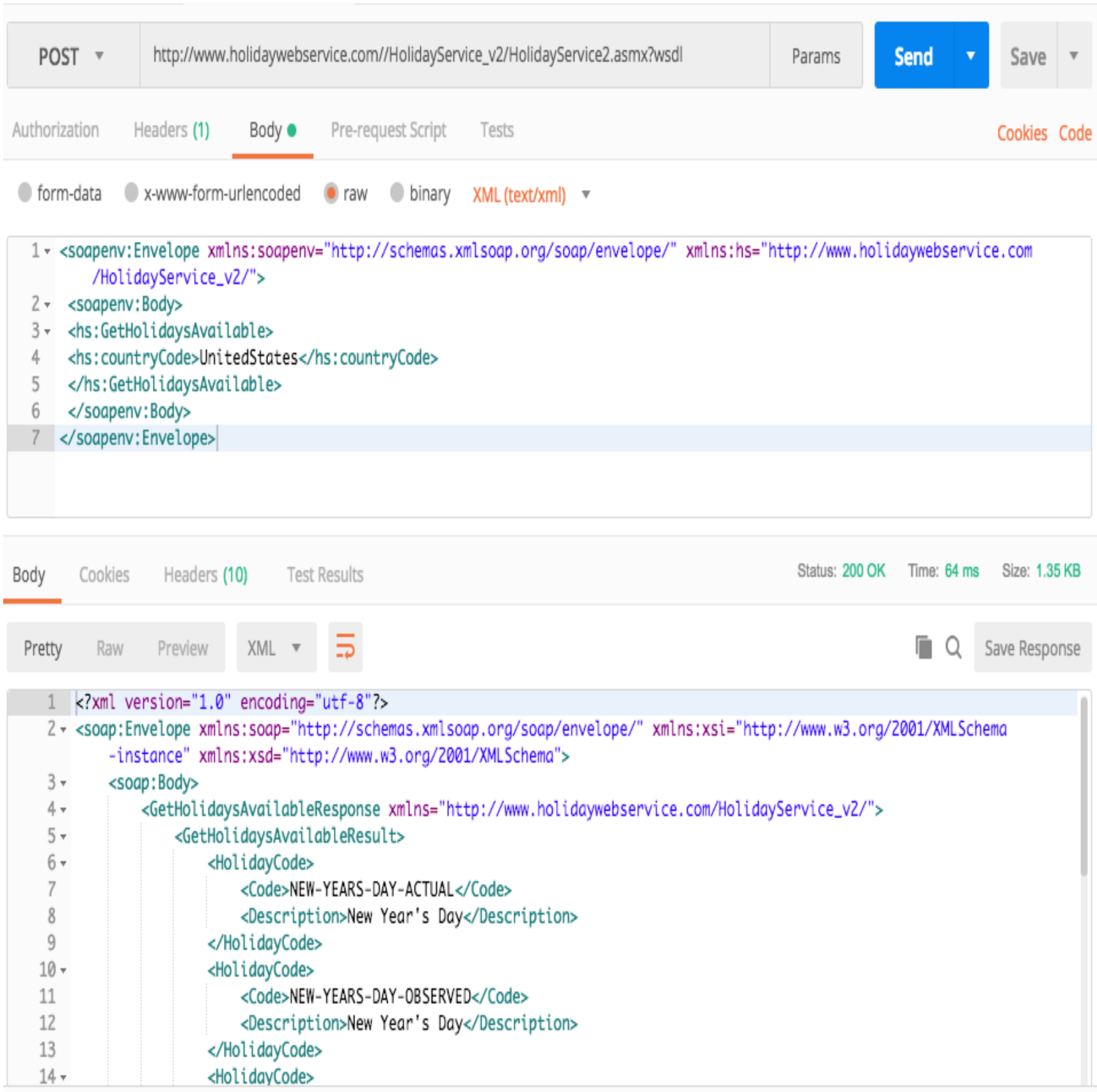


Figure 13. SOAP API in Postman

SoapUI

SoapUI is free, open source, and uses Java. SoapUI also has a commercial edition with custom utilities, scanning functionality, and enhanced testing capabilities.

SoapUI has the following aspects:

- It supports various standards such as HTTP and HTTPS. It tests both SOAP and REST web services. SoapUI supports most web service specifications, such as WS-Security and WS-Addressing.
- Using SoapUI mock services, testers can simulate web services before implementation. Mock Services are used in a development environment but are available to try out.
- SoapUI uses either Groovy or JavaScript for pre- and post-processing test configurations, similar to Postman, which allows the tester to perform dynamic testing and operations against the service.
- There are also options for integrations to automated test frameworks such as Junit or Apache Maven or Ant. While these are outside of the scope of our discussion, they are available to testers who also want additional features.
- In SoapUI, testing web services are under Projects, which is similar to the Postman organization. However, in Postman, testing is organized under Collections.
- Similar to Postman, SoapUI can create automated testing. In Postman, Runner automates the execution of all of the API calls in order. In SoapUI, TestSuites are used to structure and execute functional tests. For pentesting, use TestSuites to run and execute all of the API calls in a well-organized manner to ensure coverage and to ensure that the upstream tools that are used quickly capture the data for further testing. Instead of manually executing and validating responses one at a time, these features help testers work in an automated and more comprehensive manner.

Download the free version of SoapUI and install it from here: https://www.soapui.org/downloads/soapui.html

Start with a demo project that is available for SoapUI. Go to *File* > *Import Project* and Navigate to the SoapUI-Tutorials directory. There are a couple of sample API projects already there, pick the Sample-SOAP-Project and once the project has loaded, expand the directory. *In Figure 14*, the *Simple Login and Logout* was expanded from the *TestSuite* to display the *Security Tests*.

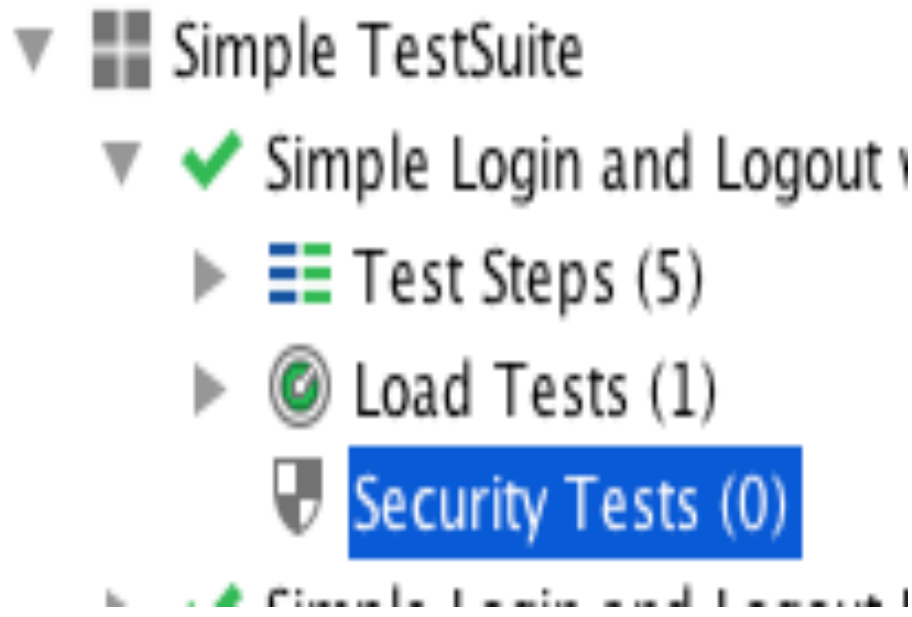


Figure 14. Security Tests in SoapUI Free Version

In the existing project, the *Security Tests* show zero (0) tests available because they are available only with the Pro version. The Pro version has an automated scanner with security testing for common attack vectors like SQL injection. The free version of SoapUI does not include automated security testing.

For the sample requests, endpoints refer to mock services. To get the tests to work in the sample project, run the mock service first by double-clicking *Service SoapBinding MockService* and, in the editor window that displays, click the green arrow.

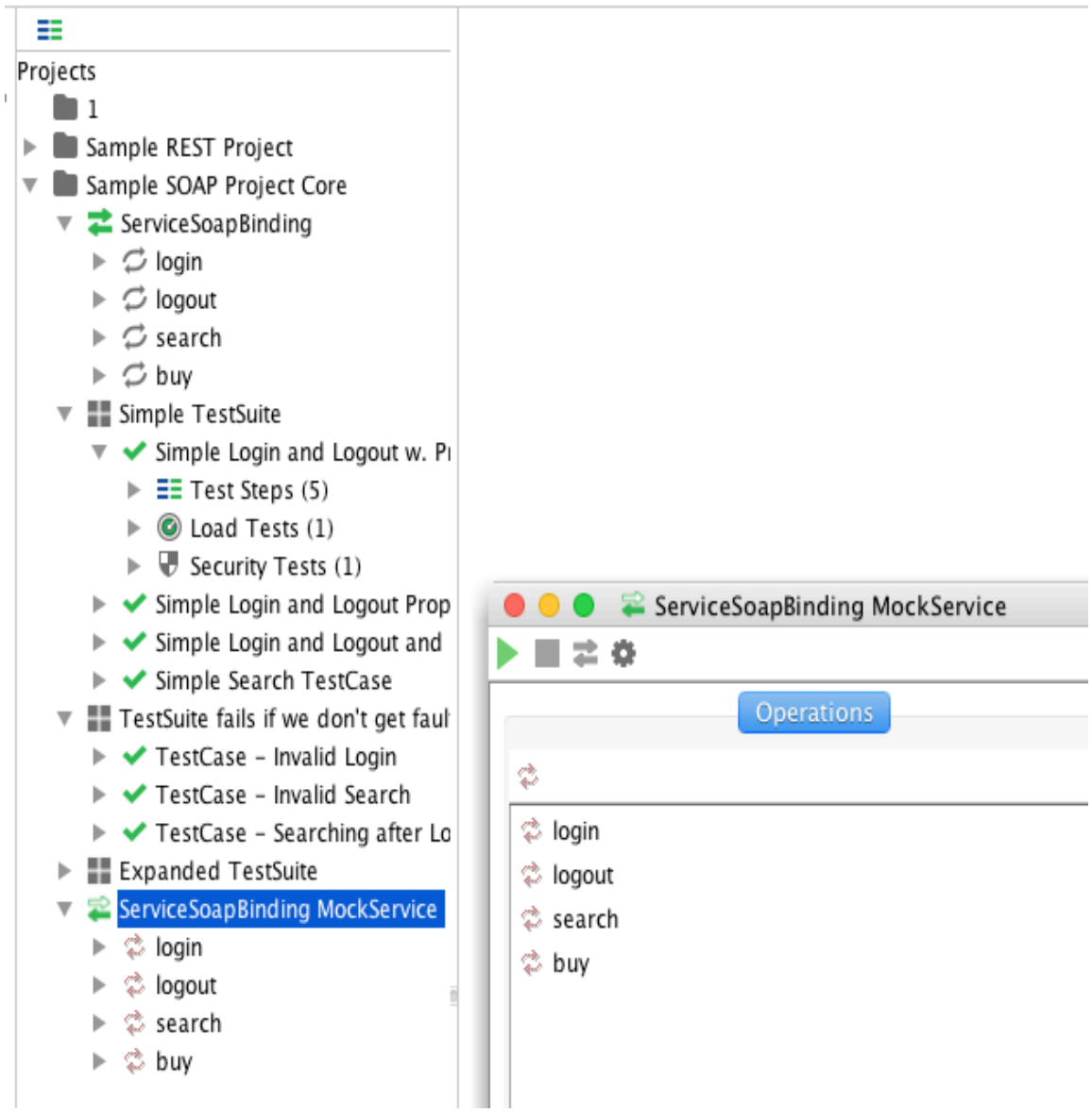


Figure 15. Run MockServices

Locate all of the information for the WSDL of the SOAP service for a project by right-clicking on the *ServiceSoapBinding* to display the context menu, select *Show Interface Viewer* to display a new window with several tabs. Select *WSDL Content*, then click on the *WSDL Content* tab to display the WSDL or description of the service. In the hierarchy under Messages is a node for *loginRequest*, which if expanded, displays the content for the SOAP message (Figure 16). The WSDL contains information about the workings of the web service, which can help testers understand an API and get better coverage when pentesting.

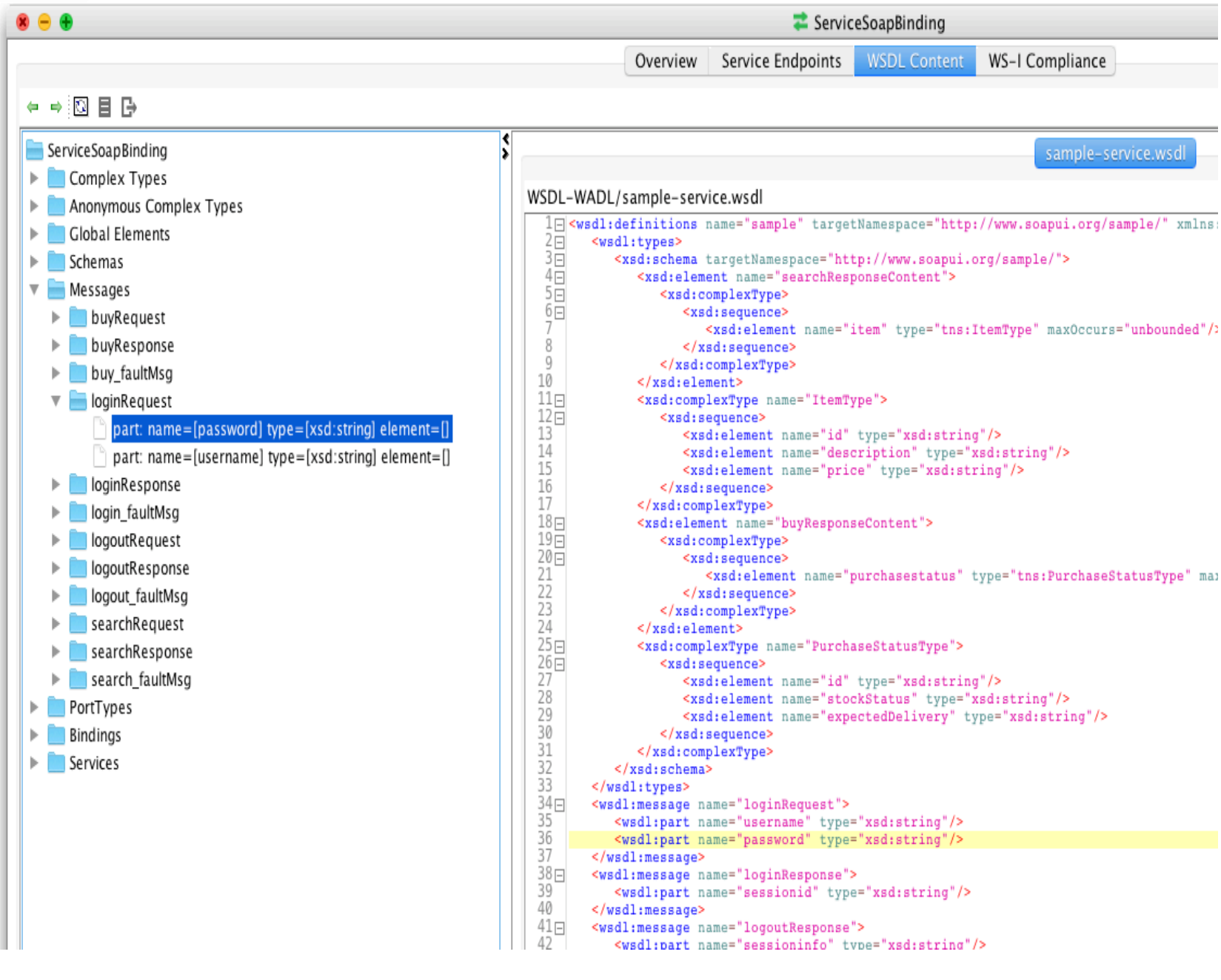


Figure 16. Click ServiceSoapBinding, Click WSDL Content, expand a Login Request in the node

The request can be submitted by expanding the login node and then clicking the green arrow to run the login. The sample has authentication credentials in the request.

Figure 17. Login

It is essential to understand how to build tests in SoapUI. The SoapUI sample files come already configured for all options of available tests, such as load and security testing. In pentesting, the focus is not to perform load testing but focuses on security testing.

Automation is considered part of Functional Testing in SoapUI. If the tester wants to extract data from an XML message, such as a session ID or token, and then write it to a message to perform additional testing, the *Property Transfer TestSteps* are used to perform these actions.

The sample file has a property transfer setup. In this case, double-clicking the *Property Transfer TestStep* opens the window with the configured transfers.

For our purposes, the login service returns a session ID that must be used for authentication and then returned in the body of the SOAP envelope for logout. The property transfer can be used to extract the session ID and write it to a property that can be referred to by all *TestSteps*.

There are three necessary steps:

1. Make a TestSuite. Create the login.

2. Create a Transfer in the new TestSuite Test Step for login.

3. Create a new property.

There are various ways to scrape the data from the SOAP envelope to obtain the session ID. The easiest way is to write directly to the target *TestStep*. However, the alternate way is to create a property to use for any request.

In the Sample SOAP Project, expand the *ServiceSoapBinding* node and right click on it to create a new *TestSuite*. For this test, de-select everything except the login.

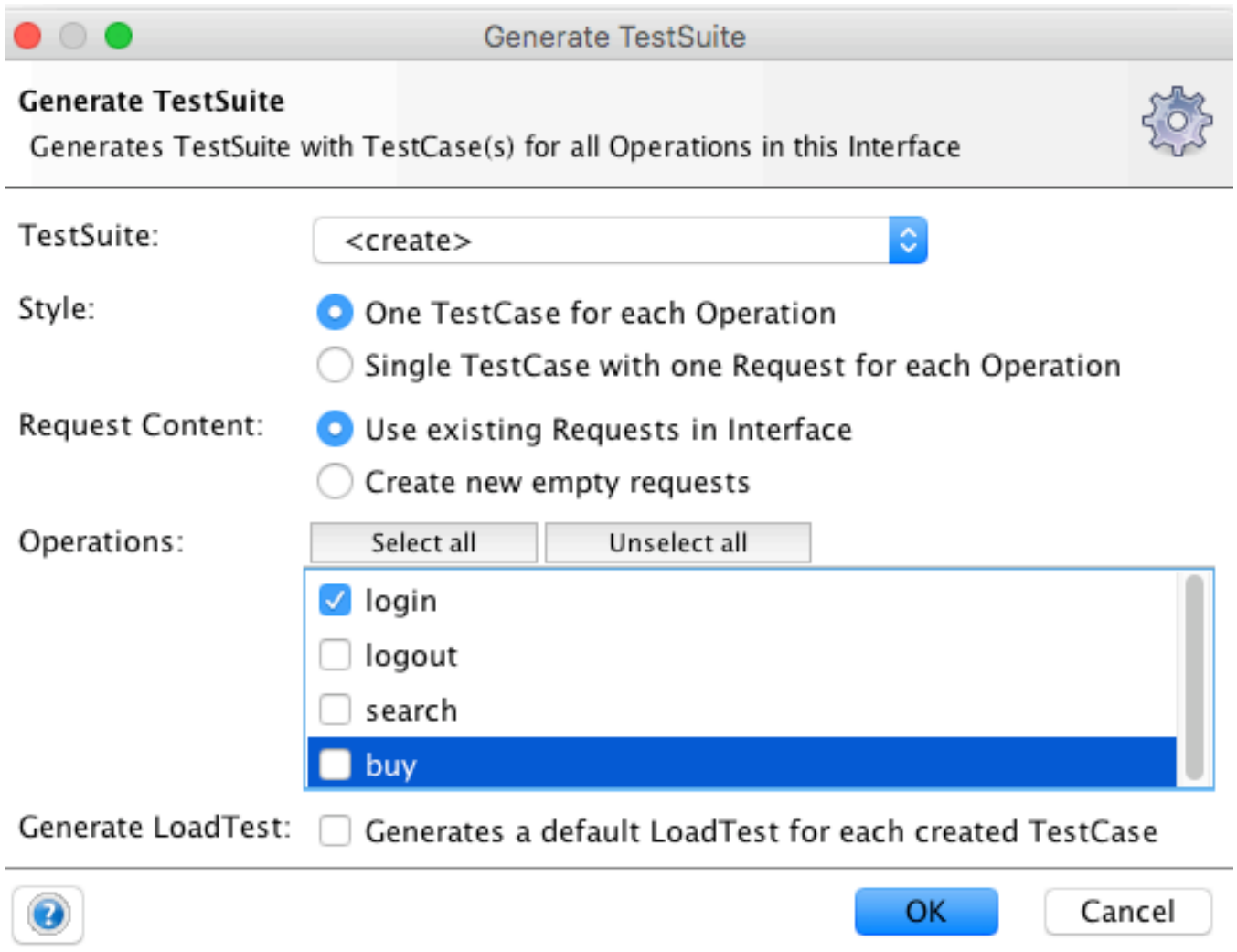


Figure 18. Create new TestSuite with login only

After adding the login, execute the request to obtain the session ID.

Right-click the *Test Steps* within the *TestSuite*, and the context menu displays the Property Transfer. Add a Property Transfer and then add a Transfer called TransferSessionID by clicking the green plus (+).

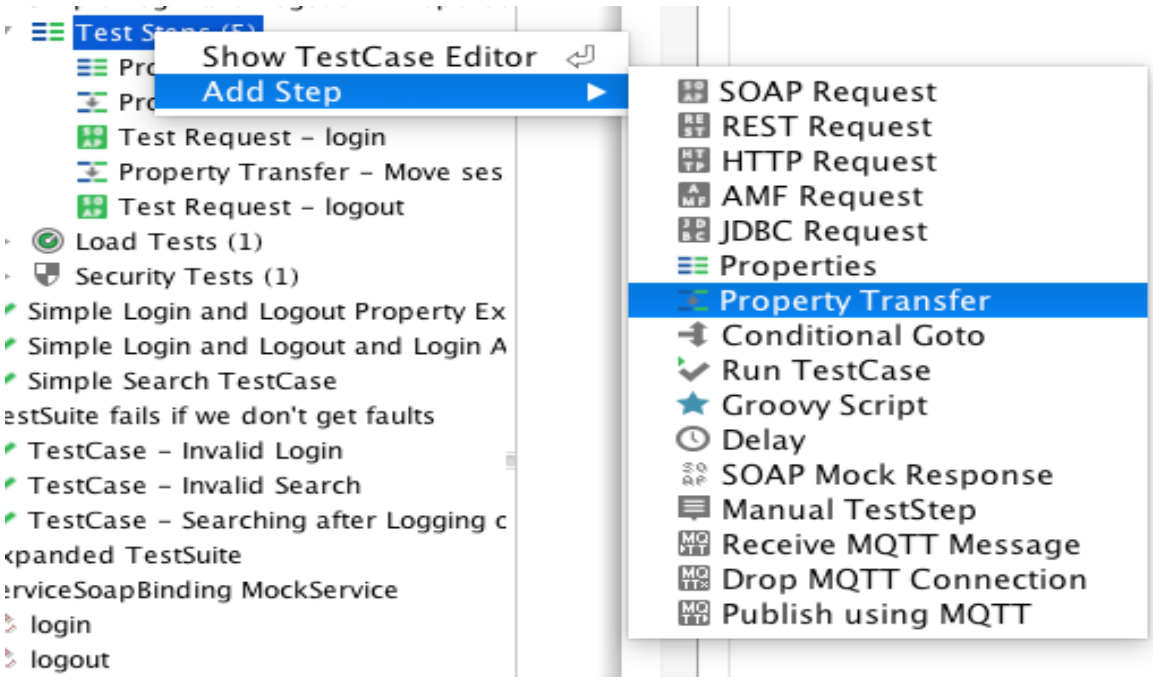


Figure 19. Create a Property Transfer

Specify an appropriate name for your *Property Transfer*. Moreover, then click the plus (+) button on the *Property Transfer* to *Add a Transfer*. Specify a name for the transfer, in this case, TransferSessionID.

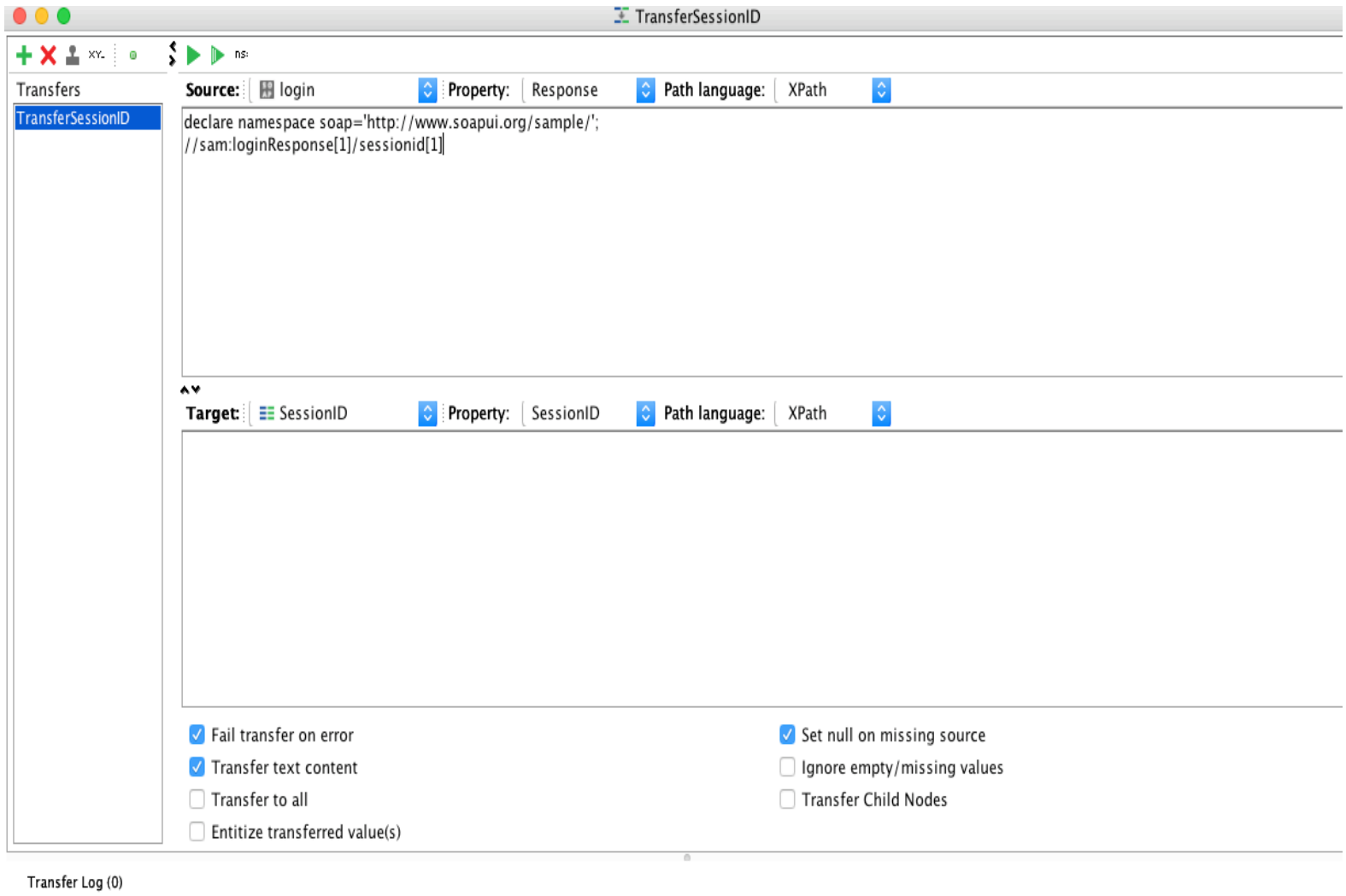


Figure 20. Adding Property TransferSessionID

The SoapUI Pro version has a wizard for XPath selection. Otherwise, in the free version, the tester manually enters the property. Since our version is free, look at the response body of the SOAP login.

```
<sam:loginResponse>
   <sessionid>3409306223235211</sessionid>
</sam:loginResponse>
```

Figure 21. Sample Login Session ID

Notice the format of the session ID is sam:loginResponse. The format from the response is used to create the XPath expression for the Source in the TransferSessionID.

```
declare namespace soap='http://www.soapui.org/sample/';

//sam:loginResponse[1]/sessionid[1]
```

In the *TransferSessionID* window, the *Source* is the login request, and *Property* is the response, and the target is the new *SessionID* property created.

When executed, the SessionID is transferred to the TestCase property and can be used in any request through property-transfer or expansion, for example in a logout request, such as:

```
<logout>
<sessionID>${#TestCase#SessionID}</sessionID>
</logout>
```

The code above replaces the expansion with the saved SessionID when sending the request. (SoapUI, 2018)

In the existing test case for *Simple Login and Logout w. Properties*, the *Properties*, and *Property Transfer* is already created to scrape and move username and password to other requests. Note that in Postman, these steps are created in a *Collection* using the *Test Script* after a request is made and transferred to an *Environment* using variables. Both tools support automation for REST and SOAP, and each has various methods to get to the same end.

Conclusion

Security testing issues manifest in various ways, and many attack vectors impact API testing. Web services have become more popular to penetration test because they offer another attack vector for the application. In many cases, web services integrate directly into systems, business processes or data and they can be a great place to form an attack and bypass application controls. Most developers and administrators also fail to realize the security issues for web services. Often, pentesters discover during a test that web services are configured to bypass many of the protections in place for an application.

RESTful and SOAP web services have various pros and cons to use and implementation. There are options to pentesting both types of web services. However, a primary goal is to ensure limited effort in the manual configuration for testing. Automating testing supports greater test coverage, decreases time spent on configuration and handling requests to the server and allows the tester to focus on aspects of testing that require more manual effort.

In Postman, various features allow the tester to automate API pentests such as the use of variables and environments. Variables allow testers to reuse values, configure the setup for various users or environments, and to scrape data from responses and chain requests in a collection to create a workflow.

In SoapUI, various features are available such as Test Cases and Transfer Properties, which allow the tester to reuse values, configure the projects for various environments, and to scrape data from responses and chain requests.

Both tools have the required functionality to perform automated and manual API testing, but SoapUI has additional security testing and automation available in the paid version.

Most of the web services testing tools are focused towards quality assurance or developers, and not penetration testers. Use SoapUI and Postman with OWASP ZAP or Burp to facilitate additional automated testing and fuzzing. Metasploit also has modules for testing web services, but most of our testing uses Postman or SoapUI as the first step in the testing toolkit.

An issue with testing is that many web application penetration tests do not include web services, and often pentesters struggle to test them because there is a lack of understanding about security implications for web services.

Overall, it is essential to understand when automation serves the purposes required for the penetration test.

Works Cited

Ansari, J. A., Imran, M. A., Kotipalli, S. R., Halton, W., & Weaver, B. (2017). *Penetration Testing: A Survival Guide.* Packt Publishing.

Bustamante, M. L. (2007, May 16). *Making Sense of all these Crazy Web Service Standards* . Retrieved from InfoQ: https://www.infoq.com/articles/ws-standards-wcf-bustamante

Dash, T., & Aroraa, G. (2018). *Building RESTful Web Services with .NET Core.* Packt Publishing.

Joyce. (2017, December 29). *10 tips for working with Postman variables*. Retrieved from Postman Blog: http://blog.getpostman.com/2017/12/29/10-tips-for-working-with-postman-variables/

Kankanamge, C. (2012). *Web Services Testing with soapUI.* Packt Publishing.

Lensmar, O. (2014, 11 19). *API Security Testing - How to Hack an API and Get Away with It (Part 2 of 3)*. Retrieved from SMARTBEAR: https://blog.smartbear.com/apis/api-security-testing-how-to-hack-an-api-and-get-away-with-it-part-2-of-3/

Miessler, D. (2018, 4). *SecLists*. Retrieved from GitHub: https://github.com/danielmiessler/SecLists/blob/master/Discovery/Web-Content/common-api-endpoints-mazen160.txt

Najera-Gutierrez, G., & Ansari, J. A. (2018). *Web Penetration Testing with Kali Linux - Third Edition.* Packt Publishing.

Postman. (2018). *Intro to environments and globals* . Retrieved from *What are environments and globals?*: https://www.getpostman.com/docs/v6/postman/environments_and_globals/intro_to_environments_and_globals

Postman. (2018). *Intro to Scripts*. Retrieved from Postman: https://www.getpostman.com/docs/v6/postman/scripts/intro_to_scripts

Shezaf, O. (2017, 09 11). *REST Assessment Cheat Sheet*. Retrieved from OWASP Cheat Sheets: https://www.owasp.org/index.php/REST_Assessment_Cheat_Sheet

SoapUI. (2018). *SOAP vs. REST Infographic*. Retrieved from SoapUI: https://www.soapui.org/resources/infographic/api-testing/soap-vs-rest-infographic.html

SoapUI. (2018). *Transferring Property Values*. Retrieved from SoapUI: https://www.soapui.org/docs/functional-testing/properties/transferring-properties.html

Stackify. (2017, March 14). *SOAP vs. REST:* The Differences and Benefits Between the Two Widely-Used Web Service Communication Protocols. Retrieved from Stackify: https://stackify.com/soap-vs-rest/

The Real Key Is To Create A 'Cybersecurity Culture' In The Workplace

An interview with Dr. Jane LeClair

Dr. Jane A. LeClair

Dr. Jane LeClair is President and CEO of the Washington Center for Cybersecurity Research & Development whose mission is to serve as a training and research center dedicated to increasing knowledge of the cybersecurity discipline. She currently serves on the Cybersecurity Advisory Board and as Cybersecurity Program Advisor for Thomas Edison State University. Prior to assuming her current position, Dr. LeClair was the Chief Operating Officer at the National Cybersecurity Institute (NCI) in Washington, D.C. and previously served as Dean of the School of Business and Technology at Excelsior College. She had a 20 year career in the commercial nuclear power industry and is an ongoing consultant with the IAEA. Dr. LeClair has written and edited numerous books, journals and articles related to cybersecurity and nuclear technology. She is a staunch advocate for women in technology. She is often sought by the media for her perspectives on cybersecurity issues that are in the headlines. Her latest book, "Cybersecurity and Infrastructure Protection" was recently released.

While each sector is 'critical', they are bound by the single thread of energy without which none can effectively function. One of the problems associated with the energy sector is that it is primarily owned and managed by independent private organizations. As such, cybersecurity is not uniform in these areas and subject to attack by those with malicious intent including rogue nations.

[PenTest Magazine]: In one of your books, you have written that in the landscape of cybersecurity the mechanism is "*Like a game of chess, for each move that is made, a countermove occurs.*" Would you describe the emergence of the Washington Center for Cybersecurity Research & Development as a general move or rather as the specific countermove for particular types of threats? Could you please tell us about the idea which was the cornerstone of the institute, its achievements and current activities?

[Dr. Jane LeClair]: Those with a vested interest in cybersecurity recognize that 'people' are the main problem in cybersecurity. Despite all the hardware and software, over 90% of all cyber breaches are in some way related to human interaction. When I left the National Cybersecurity Institute (NCI) and established the Washington Center for Cybersecurity Research & Development, it was with the thought of providing government and private enterprise with the kind of training that would reduce the level of human performance errors causing the breaches. I had an extensive career in the nuclear industry and wanted to transfer that knowledge on zero performance errors, a must in the nuclear field, to cybersecurity. We have been very successful at doing just that, providing information and training to various agencies, educational institutions and businesses.

[PT]: How would you evaluate the common awareness of threats and understanding the vulnerabilities by the government institutions and the business sector? How has it been changing throughout the years, looking at the matters from your perspective?

[JLC]: It varies a great deal from agency to agency, business to business. Some leaders have been enlightened to the issues of cybersecurity while others are either in denial or blissfully ignorant. Generally speaking, I would say that there is more awareness than five years ago, but a great deal still needs to be done. I was at a meeting recently and a middle level manager from a major organization indicated that he was having a difficult time getting upper level leaders to put money into cybersecurity...I was shocked that a large organization would be so blasé about their security....but it happens...despite all the media coverage of breaches, some leaders just don't get it.

[PT]: What changes can we observe in developing the cyber workforce, in order to address the gaps in the cybersecurity field? Would you say that certain subsectors, also the ones within the national security field (for instance, the commercial nuclear power industry), are prepared adequately for potential cyber threats?

[JLC]: In developing the cyberworkforce, there are several observable changes. First, numerous learning institutions are expanding or developing cyber programs to help fill the pipeline for much needed professionals in the cyber arena. Thomas Edison State University for one is rapidly expanding their cyber program offering online certificates in cyber, a bachelors degree, and will shortly be offering a Masters in Cybersecurity. We are also seeing more young people join the field as well as a slowly increasing number of women and minorities. These people are the future of cybersecurity with their fresh outlook and differing perspectives. As for who is adequately prepared, that is a hit and miss situation. Some organizations are well prepared while others aren't. Government agencies as well as small to mid-sized organizations, for example, are in a difficult situation in that they may be aware of the need, but have difficulty securing funding for the training they need. Of particular concern for those agencies is the cybersecurity of their partners, suppliers and contractors that work for them. Hackers seek to gain entry to suppliers and contractors that do not have adequate defenses and then leverage that to gain access to the more well-defended organization or agency. But overall, vital organizations, for example nuclear power, are well defended, but must always be aware of changing conditions and technology that could undo their defenses.

[PT]: Do you think that global cybersecurity education is nowadays sufficient? Or maybe we need a more sophisticated educational system within this field? Is the supply of cybersecurity sufficient for this constantly growing job market?

[JLC]: For eons we were a paper-oriented society, but now we are totally committed to electronics and rapidly evolving technology. In many ways, we have fallen behind in the education of individuals to deal with this advancing technology and fill the seats of cyber professionals. Currently, there are well over 300,000 vacancies in the cybersecurity field and the numbers are increasing daily as the shortfall grows. By some estimates, there will be over 3 million vacancies in a few short years in the cyber field. To meet that expected shortfall we need to greatly expand the programs that train and educate our future cyber professionals...and we need to encourage and recruit women and minorities to join the STEM field.

[PT]: There is no doubt about the fact that cybersecurity is a shared responsibility of government institutions, business and the rest of the citizens. Unfortunately, regular cases of data breach prove that the cyber threat awareness is still insufficient. What would be your ideas to spread the cybersecurity knowledge within the general public?

[JLC]: Human error by employees is by far the greatest cause of data breaches. People know they shouldn't open a questionable email, or provide PII over the phone, but they do so anyway. Education and training is important in stopping this behavior, but the real key is to create a 'cybersecurity culture' in the workplace similar to the 'safety culture' that exists in the nuclear industry to prevent human performance errors. It would probably be a good idea for a government sponsored ad campaign aimed at cybersecurity that runs along the lines of fire prevention, anti-smoking or litter control ads to constantly remind the populace to take cybersecurity seriously.

[PT]: What are your reflections on the role of the women in the field of cybersecurity? Is the gender disparity in cybersecurity professions going to be diminished? What has to be done in order to achieve this goal?

[JLC]: Women make up over 50% of the workforce in the US but less than 10% of the 'seats' in cyber are taken by women. That is a huge waste of potential. I am hoping that those numbers will change, but increasing the percentage of women (and minorities) in STEM has moved at a glacial pace. There needs to be a three pronged plan to fix that imbalance. First, at home where parents need to allow their daughters to explore their interests and encourage them in sciences and in non-traditional roles. Second is in education where teachers, counselors and administration must to create a level playing field so that women can explore and compete in the classroom equally. Finally in the workplace where gender bias and harassment must be eliminated and equality addressed – equal pay, chance of promotion, etc., are needed. A large number of women drop out of IT positions leaving many vacancies that are slow to fill...managers need to address that issue and look for ways to keep women working for their organizations.

[PT]: Could you please tell us something about the initiatives which aim to support women who want to pursue their careers in the field of IT security?

[JLC]: There are a number of initiatives that seek to support and encourage women in IT. For example: The Executive Women's Forum, the League of Women in CyberSecurity, The Raytheon Women's Cybersecurity Scholarship, The Women in Cybersecurity Conference (WiCyS), The Annual Women in Cyber Security Reception hosted by CyberWire, and the Women's Society of Cyberjutsu (WSC) to name a few. These organizations, and others, work to encourage young women to join IT programs, stay the course and finish their education and training, and support them as they work their way into lucrative positions in the field. They do so with mentoring, scholarships, acting as role models, hosting conferences for networking, and encouraging young women at job fairs. I personally have seen the benefits of such actions and strongly advocate for women in the IT field. While I was the head of the National Cybersecurity Institute (NCI) in Washington, I began the Initiative for Women in Cybersecurity, and started a scholarship for Women in Technology. All these initiatives work together in a synergistic manner to advance women in IT. While more needs to be done to advance women, there is a growing groundswell that I hope will eventually level the playing field for women.

[PT]: What would you define as the next more vulnerable sector in the future?

[JLC]: Without question our critical infrastructure is at risk. With Presidential Policy Directive 21 (PPD-21), the government has identified sixteen critical sectors in our infrastructure as vital to the ongoing operation of our nation. Those identified sectors are in Chemical, Commercial Facilities, Communications, Critical Manufacturing, Dams, Defense Industrial Base, Emergency Services, Energy, Financial Services, Food and Agriculture, Government Facilities, Healthcare and Public Health, Information Technology, Nuclear Reactor Materials and Waste, Transportation, Water and Wastewater. While each sector is 'critical', they are bound by the single thread of energy without which none can effectively function. One of the problems associated with the energy sector is that it is primarily owned and managed by independent private organizations. As such, cybersecurity is not uniform in these areas and subject to attack by those with malicious intent including rogue nations. Over the past few years, there have been an increasing number of attempts to breach the security of energy-related sites by outsiders including attempts to gain control of SCADA systems within energy producing plants. To effectively address this threat, independent owners must work closely with government agencies to develop uniform standards and procedures that forestall or minimize outsider threats to this most vital sector.

[PT]: Do you predict any complex, systemic improvements on the global cybersecurity arena? If so, what would it look like?

[JLC]: In the short term – no. In the near future, it will continue to be a cat and mouse game between cyber defenders and those with malicious intent as incremental changes are made. Not too far over the horizon, however, I see Artificial Intelligence (AI) playing a major role in predicting and addressing cybersecurity issues. Unfortunately, the 'bad guys' will also have access to AI....and the chess game may very well continue.

Automation in Penetration Testing: Nessus, Yuki Chan, Static and Dynamic Analysis

Harpreet Singh

Harpreet Singh is an Information Security enthusiast with 3+ years of experience in different domains of Information security. He has been acknowledged and rewarded by Standard Chartered Bank for an outstanding performance and a leading payment solution firm for finding vulnerabilities in their online and local services. He is an author at *cybrary.it* [H5p], *GreyCampus*, *resources.infosec* and has his own blog as well - harpreetsinghpassi@wordpress.com. Harpreet holds CEH v9 and many other online certifications. He loves to meet new people and always up for extempore and pep talks. Apart from information security his areas of interest are playing the harmonica, surfing the YouTube and reviewing the food joints [FoodBond_oo7@zomato].

During the course of a penetration test, you may encounter tasks that a tool may not accomplish. One such task that I faced during one of the tests is to identify if a set of default directories were present in the target. I know that there are tools like dirbuster that can be of use, but I had none with no accessibility to the internet. These kind of situations forced me to develop a code that will do the job. The code is simple. There are two objects that we are going to play with. One is the IP that is the target and the other is the list of directories that will be checked. The code will generate the complete URL by appending the directory names to the target IP.

Automation in Penetration Testing

Penetration testing is one of the vital roles when it comes to security. It is the responsibility of the pentester to come up with exploitable vulnerabilities in the network, web application, etc., after a successful pentesting. The terms ‘penetration testing’ and ‘vulnerability analysis’ are used interchangeably but the two are different. Vulnerability analysis will result in a list of vulnerabilities for a specific target. Penetration testing focuses on identifying and exploiting vulnerabilities to gain access to a target.. A simple example is a missing Windows patch on a system. This is a vulnerability. Exploiting this to compromise the system partially or fully is called penetration testing. The first thing I would like to highlight is that penetration testing is an art and cannot be mastered in a day or two. Depending on the target, it takes time to perform the testing. The hard part is that this cannot be fully automated but can be automated to some extent to reduce the analysis time. In the below article, we will be discussing various tools and techniques that can be used, or rather assist, to automate the process of penetration testing of networks, web applications and Android applications. These tools and techniques can be used to identify the vulnerabilities and the tester can utilise the time to identify the vulnerabilities that can be successfully exploited.

We will be discussing Nessus and YukiChan framework that can help us to identify the vulnerabilities in no time.

Nessus is a paid tool that can be used to schedule, customize and launch scans on a single target or multiple targets at once. The scans are highly customizable and the tool comes up with various prebuilt profiles as well so that you just have to define the target and launch the scan. Nessus is a crippleware software; the free version has some capabilities and some of the features are only available in the paid version. For demonstration purposes, we will be using the free version of Nessus. Let's get to the details.

NESSUS

Nessus can be installed from: https://www.tenable.com/products/nessus/nessus-professional

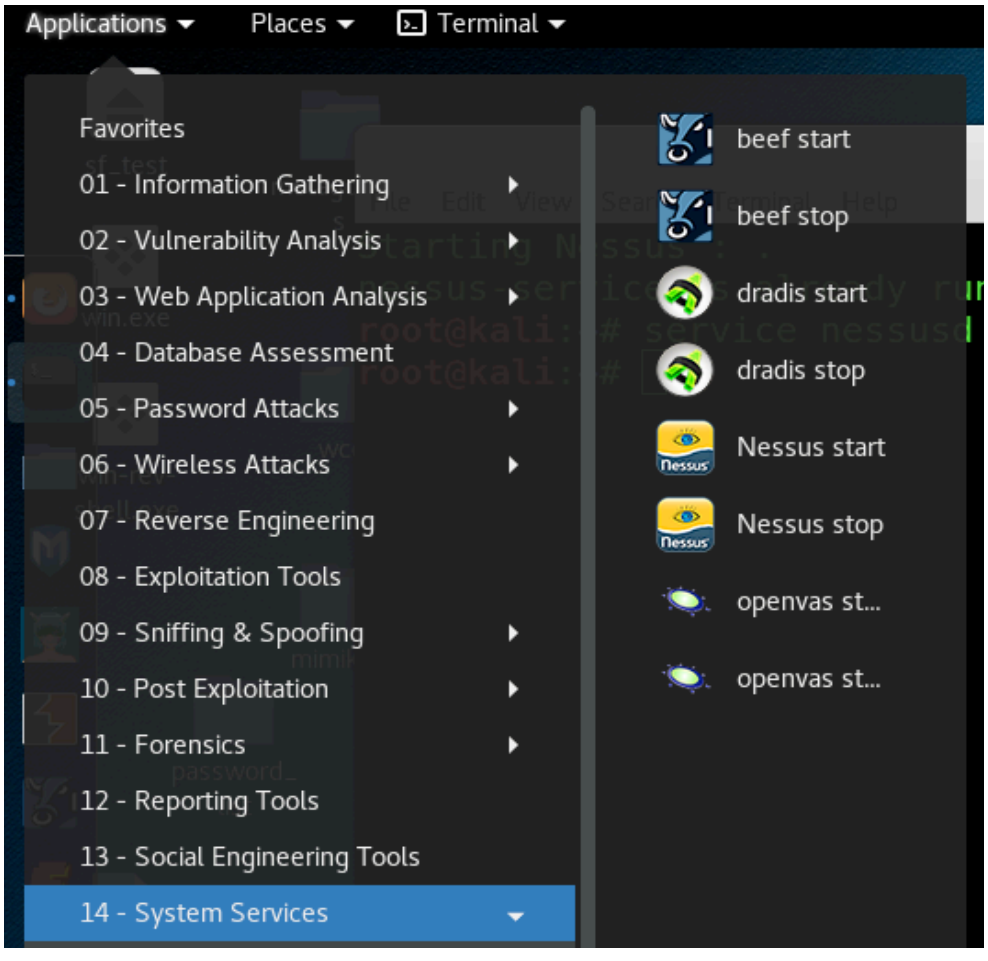
We will not discuss the installation part as that is easily available in Nessus documentation.

Once the software is installed, we will be required to start the demo for Nessus.

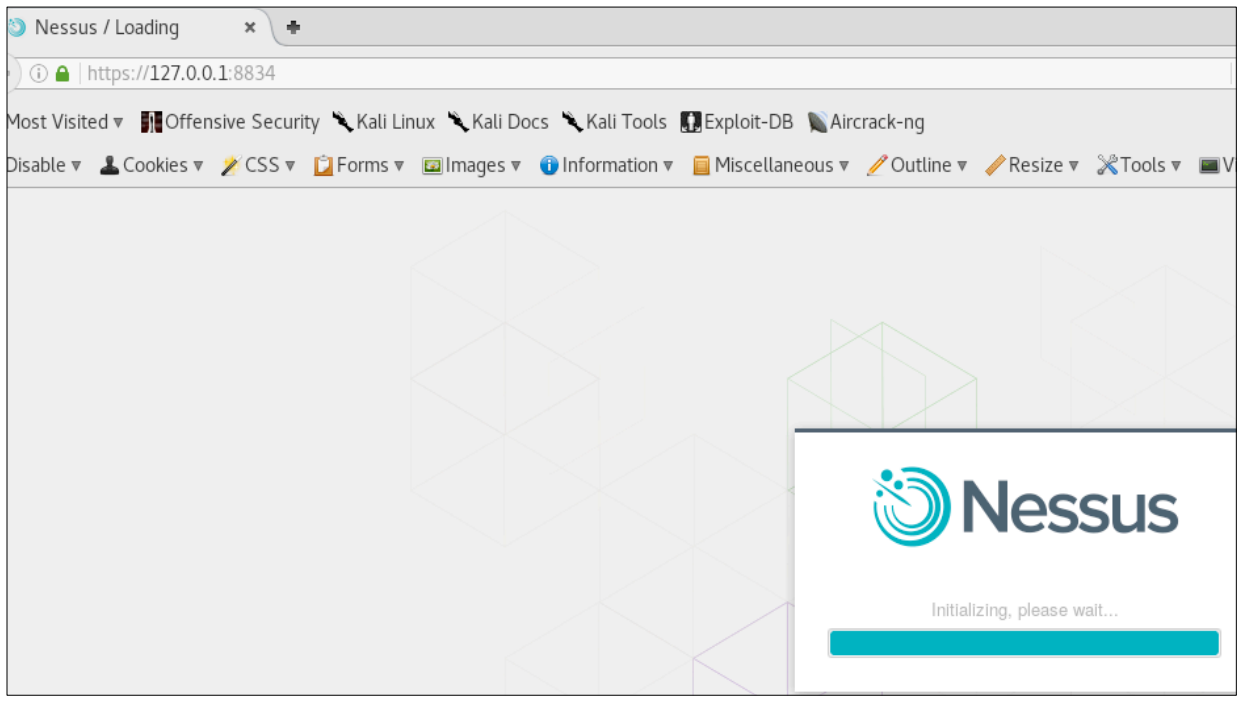
[DEMO]

The below installation has been done on Kali Linux and services can be started from the system services section:

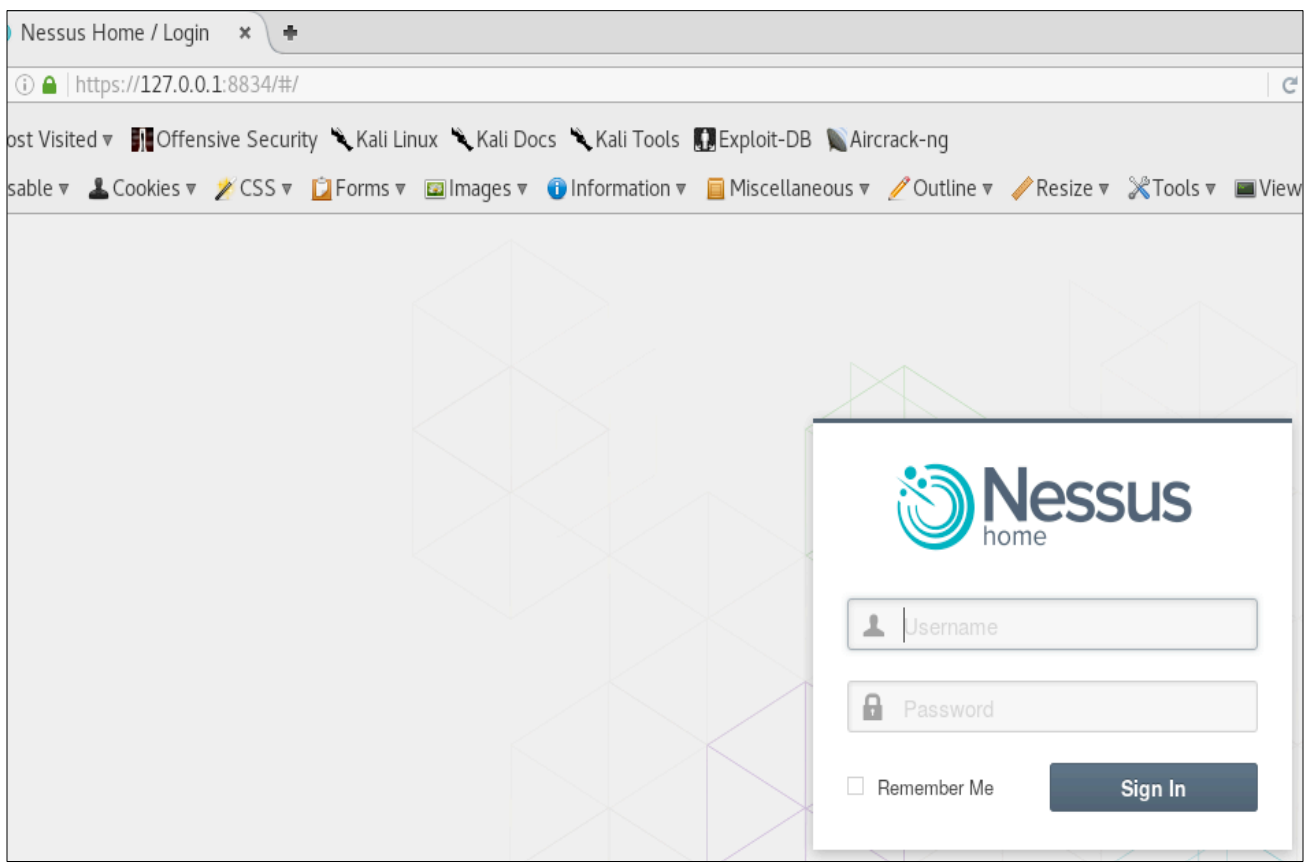
Step 1: Start Nessus



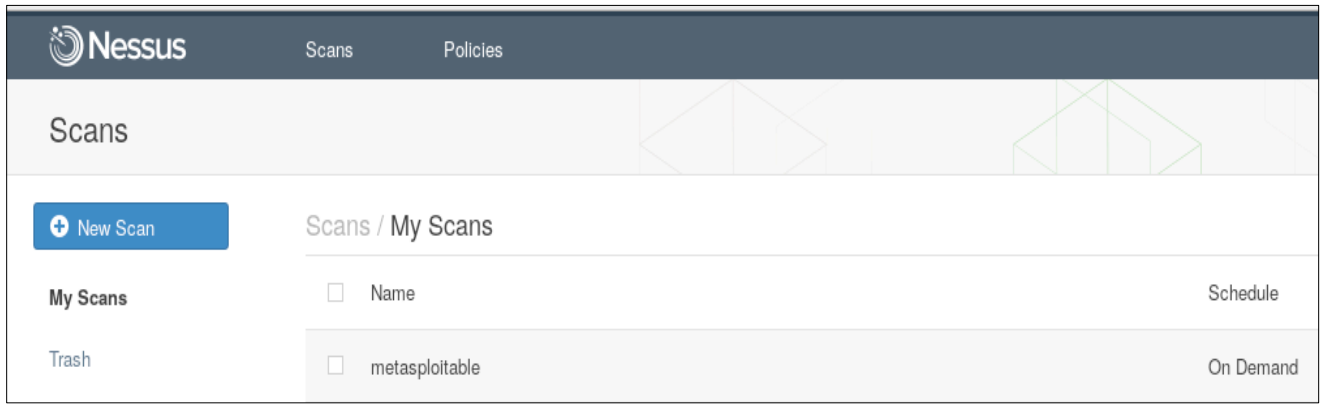
Step 2: Once Nessus has started, you can access it on localhost (127.0.0.1) on port 8834 through web browser. At first it might take some time to initialize. Be patient.



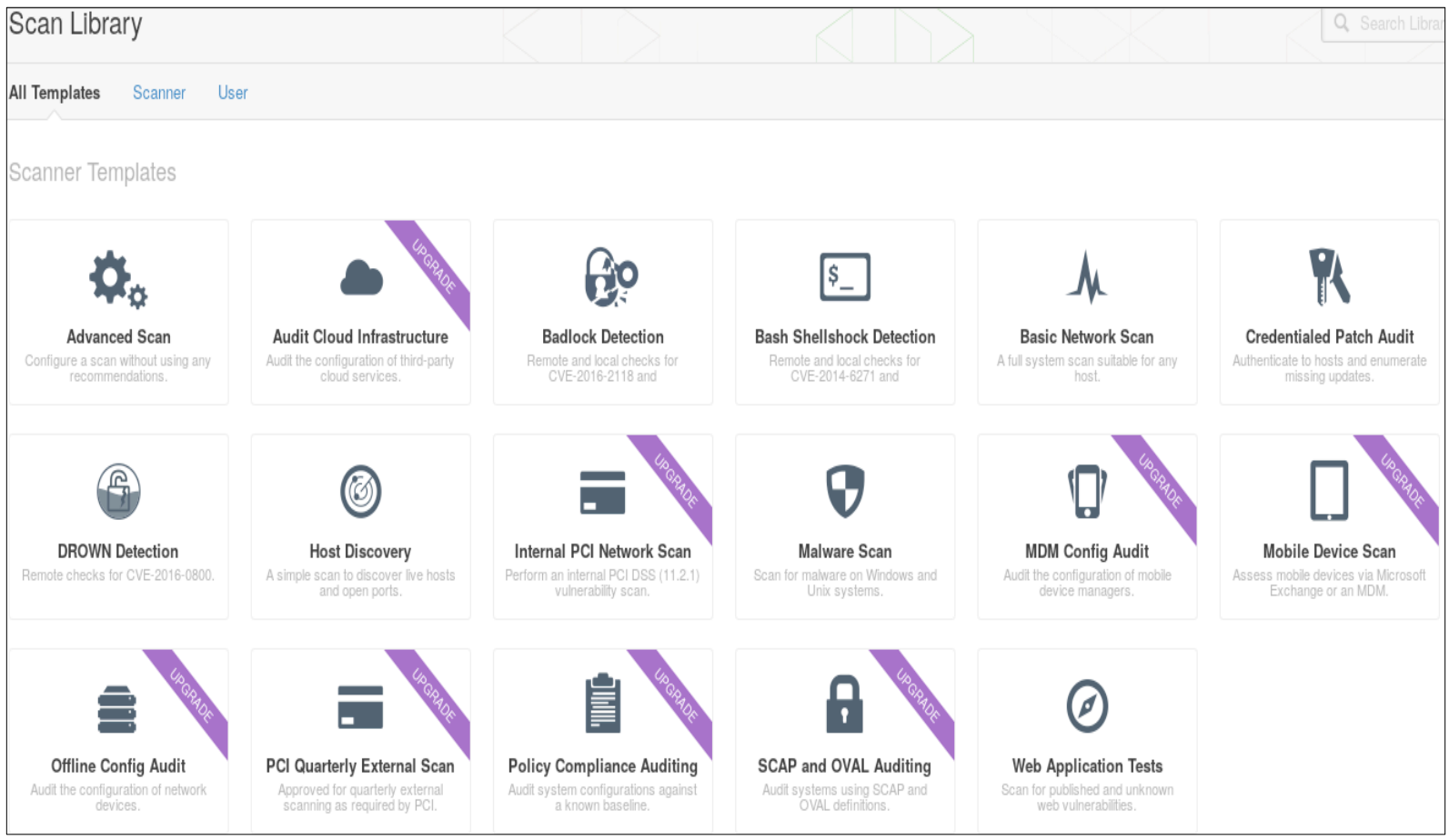
Step 3: Once the initialization is complete, enter the username and password to get in.



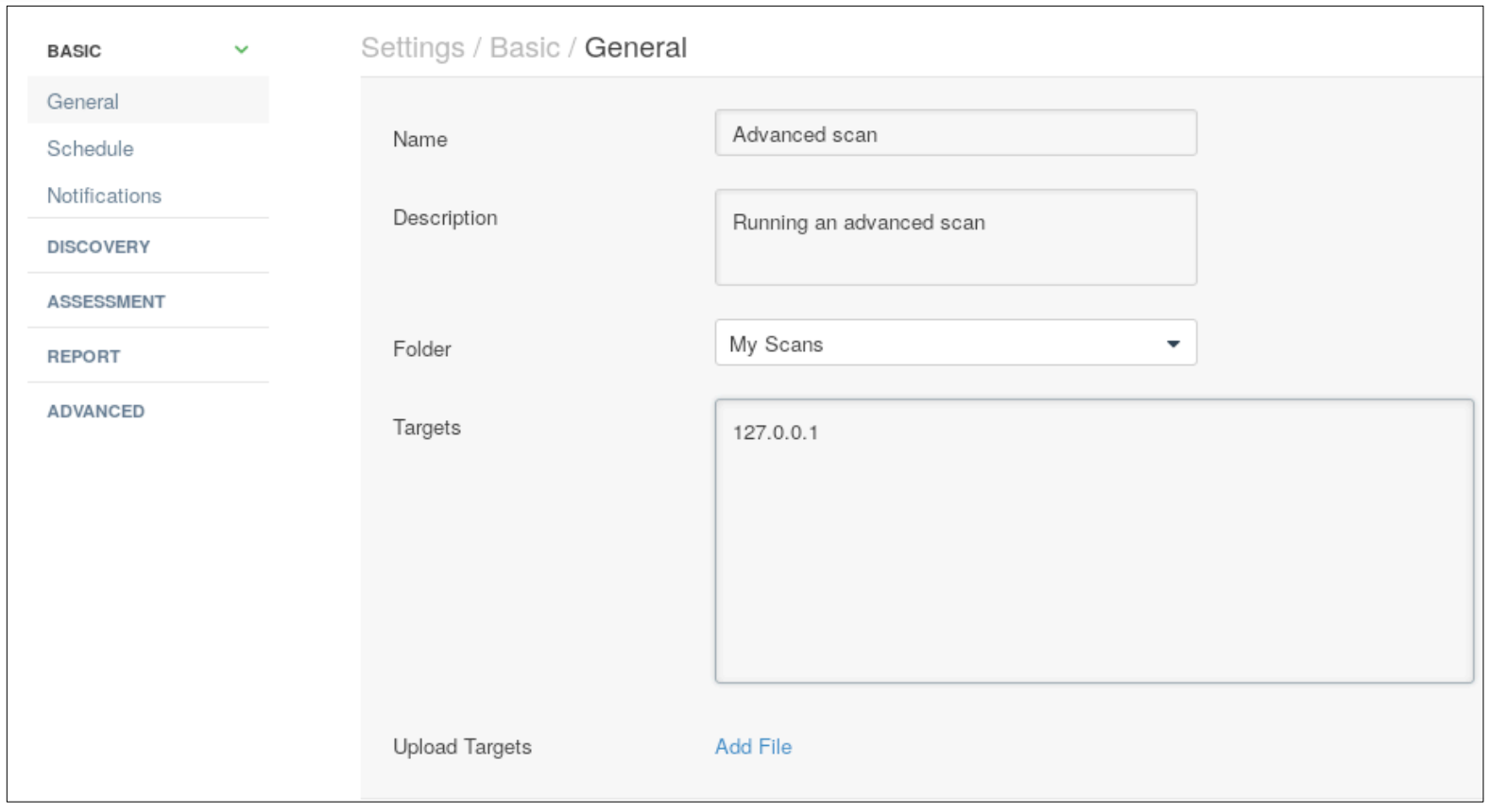
Step 4: Below is the scan screen, where you can add the scans. Just add the scan name and configuration can be done later.



Step 5: Now you can select the scan type; if you prefer, you can choose to run a basic network scan, which is quick, or an advanced scan with various customizable parameters. Some of the options need an upgrade but it's worth the buck.



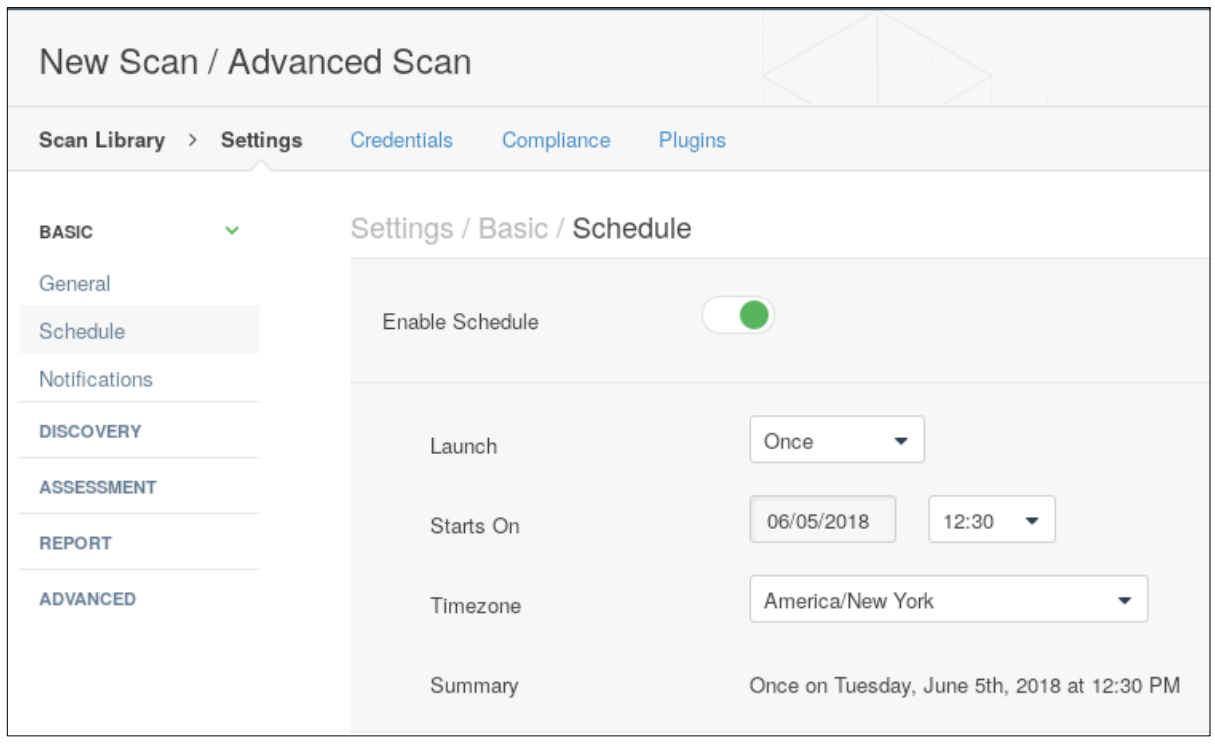
We have selected advanced scan just to walk you through the various options. Add the description for the scan and add the target list. The target can be a single IP or a list of IPs. Multiple targets can be added by uploading a file.



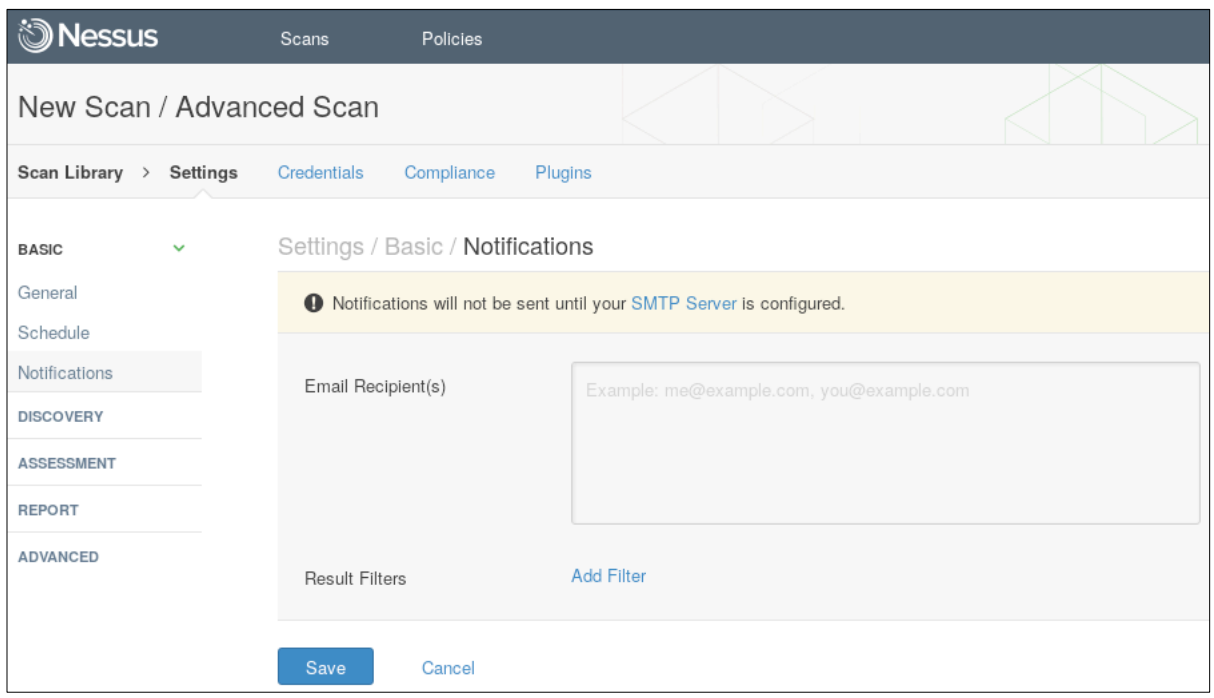
On the left side, you can see five major tabs: General, discovery, assessment, report and advanced options.

Below are a few options under those tabs.

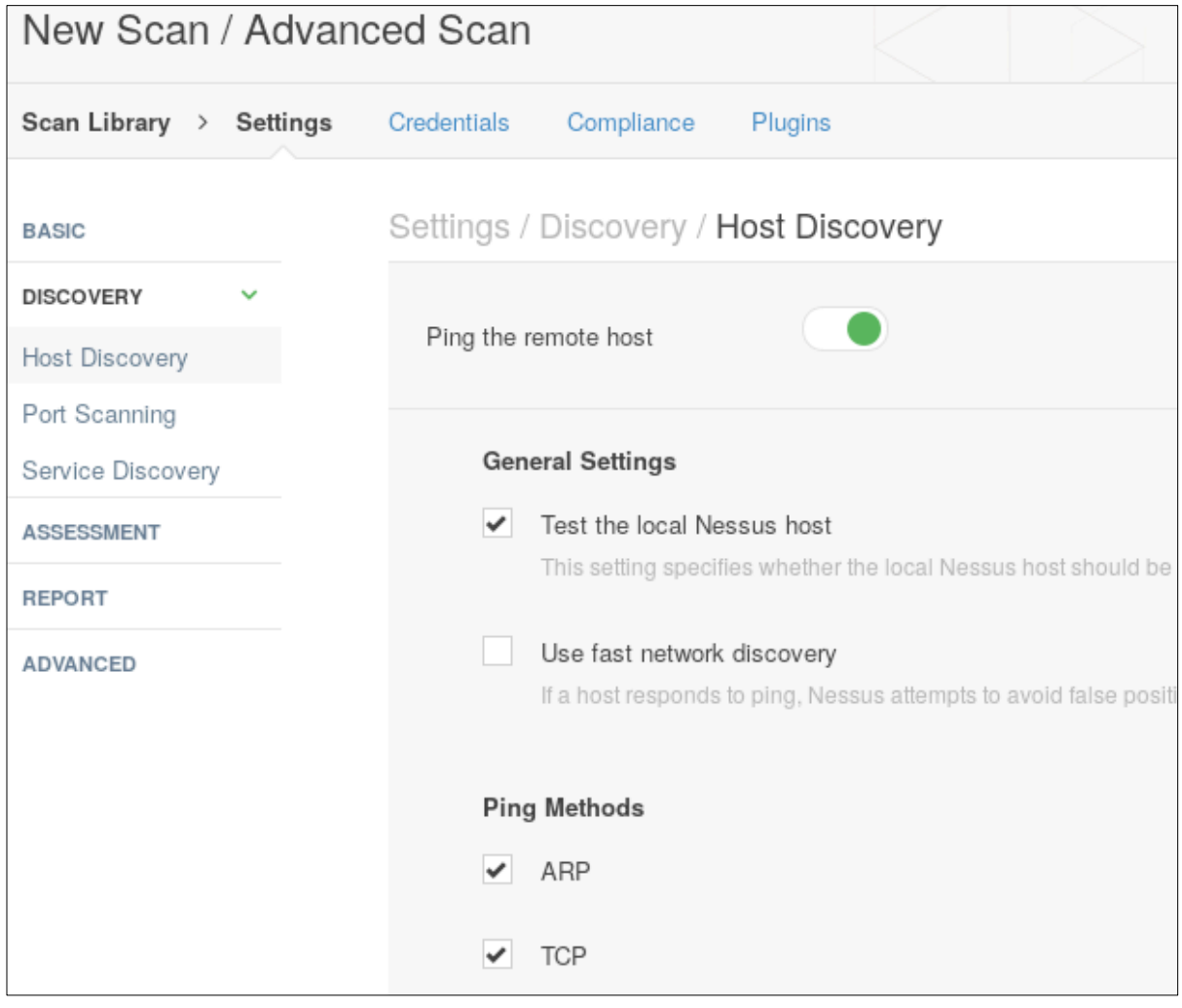
Scheduling the scan. Can be performed during non-peak hours.



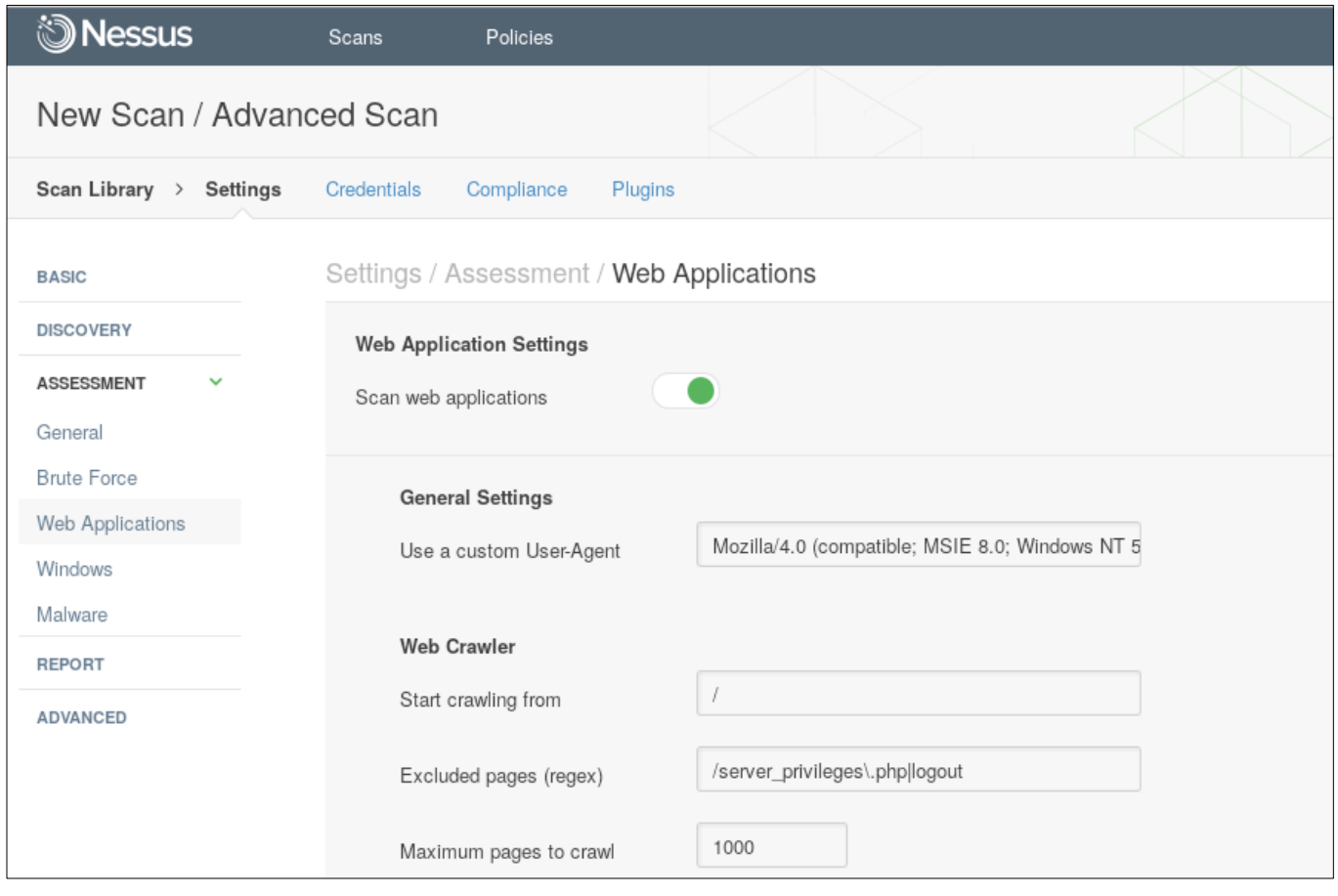
Configuring the SMTP for mail notifications.



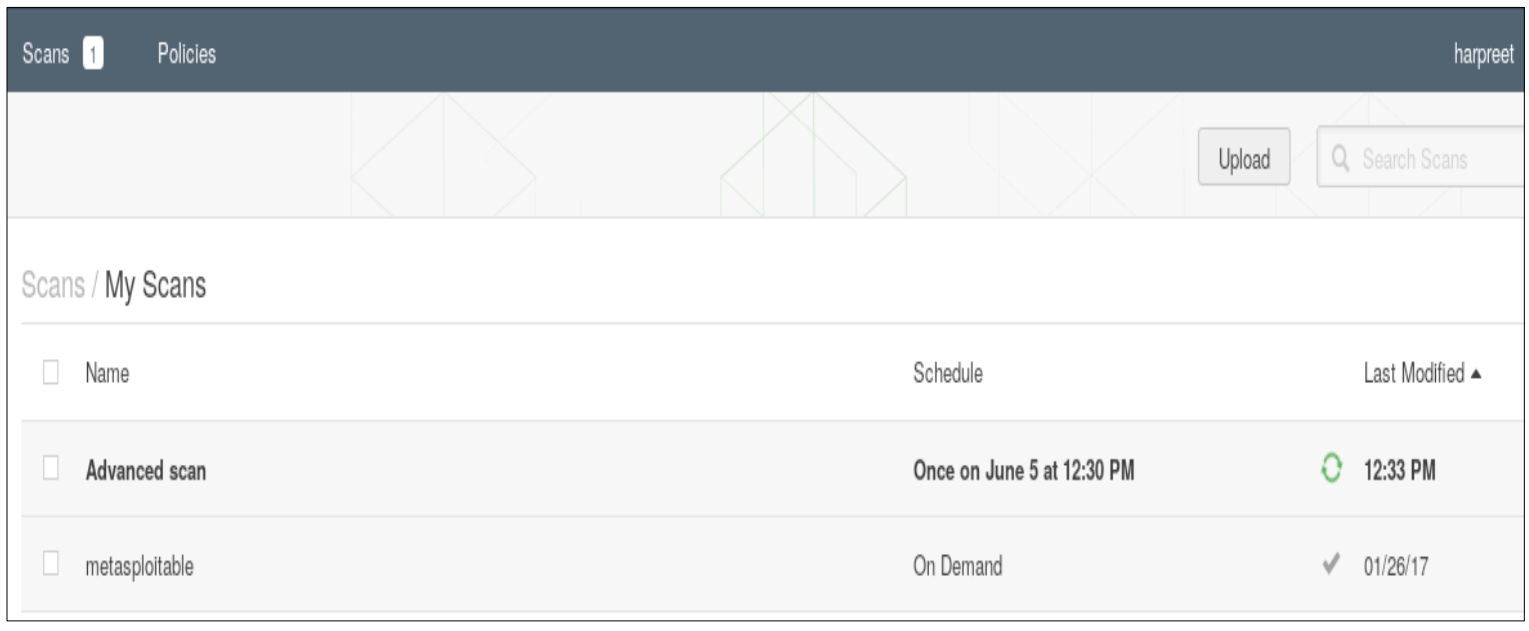
Setting the host discovery options, using various protocols that can be selected from.



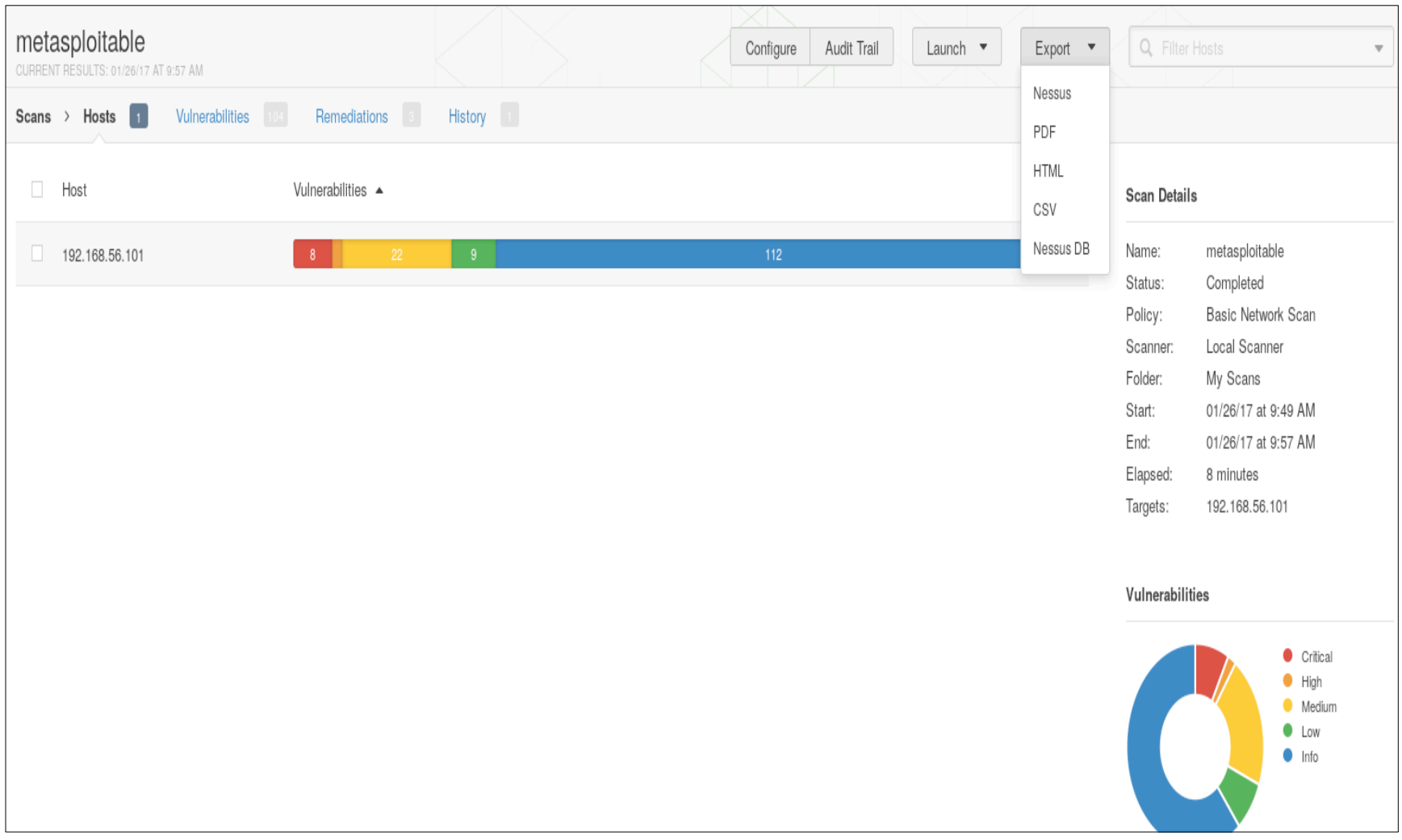
If it is a web application, you can define the scanning for that as well. It will crawl the directories and pages, which can be helpful later.

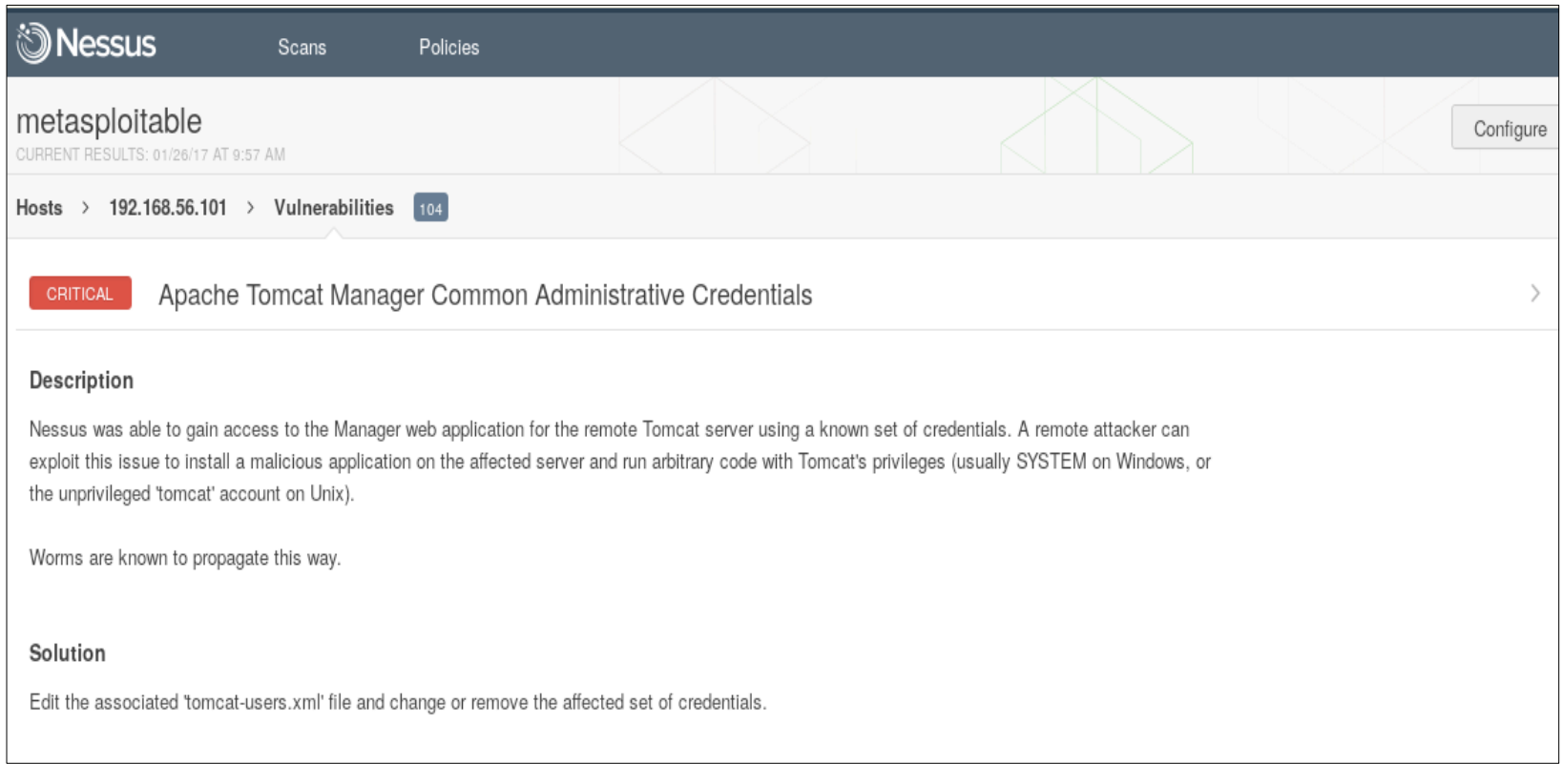


Scanning in progress: Home page screen



Report generation: We have performed the scan on a vulnerable box (metasploitable). The screenshot of the generated report is below. The vulnerabilities have been categorized into high, medium, low and information.





The report can be further exported into various formats for review. Metasploit framework has the capability to import the reports generated by Nessus that can then be used to exploit the target. Now the pentester can analyze the report and spend time penetrating the vulnerabilities.

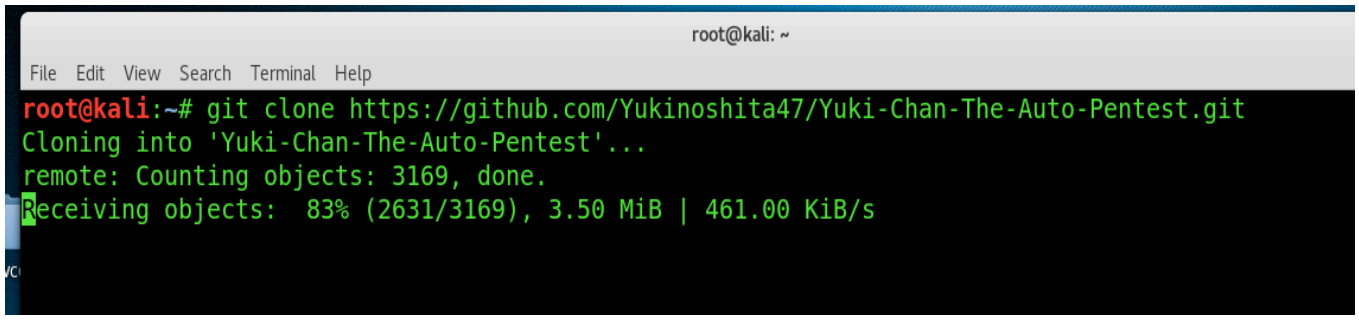
YUKI CHAN framework

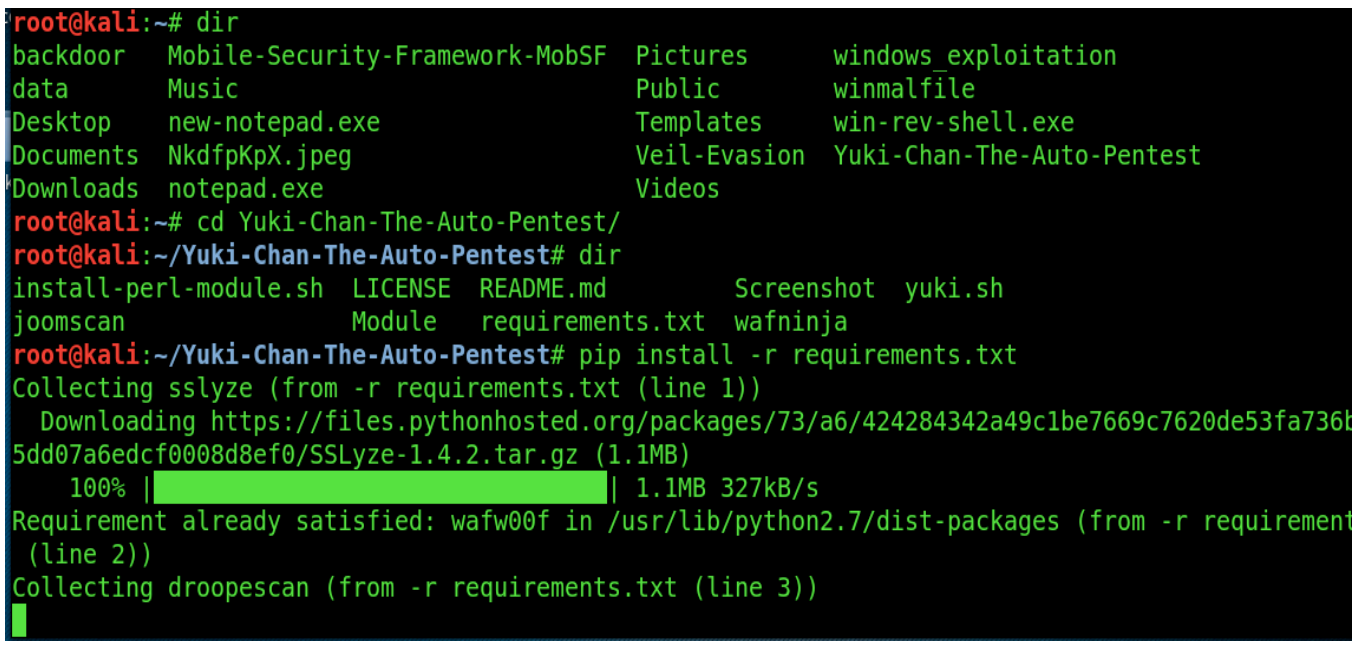
It can be used to perform the analysis of websites using various free tools. The automation process is simple: it has various tools integrated that can try to identify and extract most of the information about the website that the pentester might miss during the information gathering phase. It is really a time saver that the tool is gathering all the information, performing vulnerability analysis, perform system enumeration, SSL auditing, etc., and getting you the results, which would have taken a lot of time if performed manually. The tools that are integrated with Yuki Chan are mentioned below:

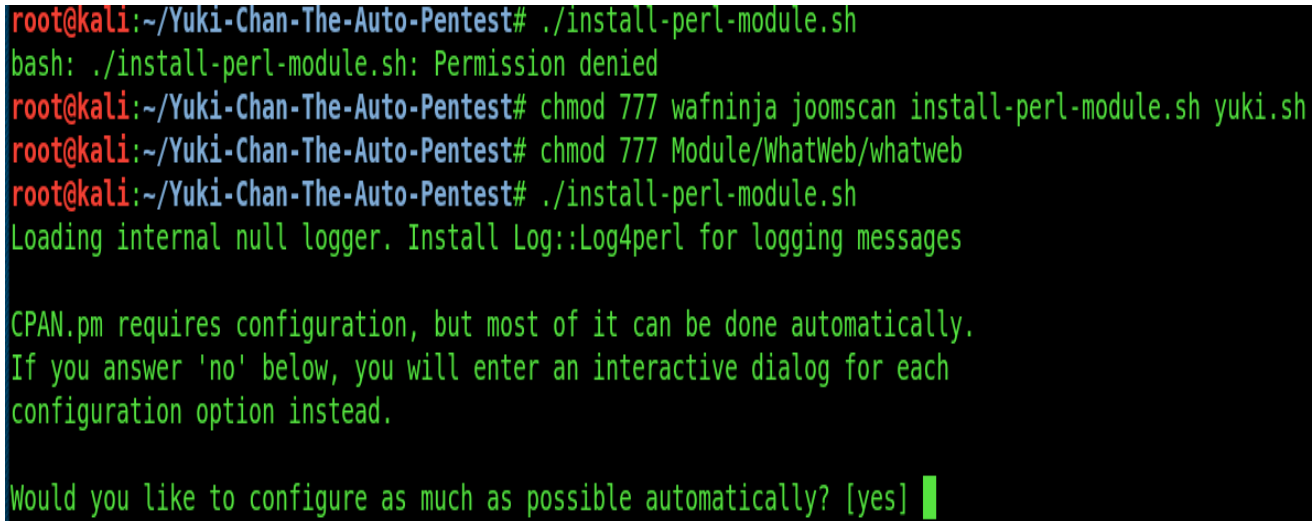
- Whois domain analyzer
- Nslookup
- Nmap
- TheHarvester
- Metagoofil
- DNSRecon
- Sublist3r
- Wafw00f
- WAFNinja
- XSS Scanner
- WhatWeb
- Spaghetti
- WPscan
- WPscanner
- WPSeku
- Droopescan
- SSLScan
- SSLyze
- A2SV
- Dirsearch

[DEMO]

Step 1: The framework can be installed from *github* using git clone.

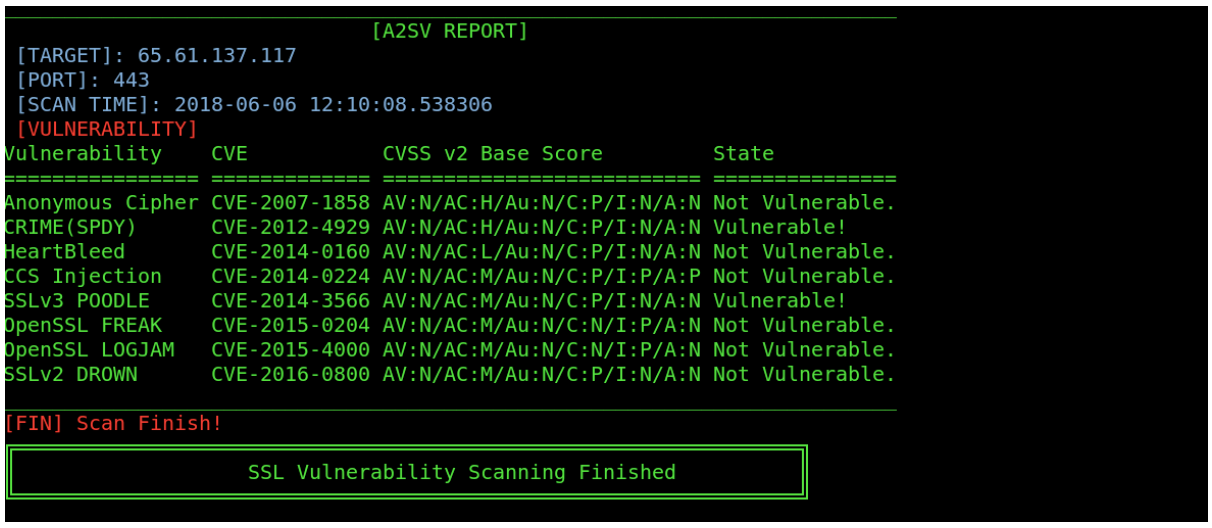






Step 2: Provide the *yuki.sh* file with execute permissions and run it. The tool will ask for the target website. Do not prefix the target with https or http. Framework will add that automatically.

Below is one part of the result. It describes the various vulnerabilities that are present in SSL.



Automated penetration testing of Android applications

MobSF is an open source and intelligent tool that can be used to perform both static and dynamic analysis of Android and iOS mobile applications. It also helps with Web API Security testing with its API Fuzzer that can do Information Gathering, analyze Security Headers, identify Mobile API specific vulnerabilities, like XXE, SSRF, Path Traversal, IDOR, and other logical issues related to Session and API Rate Limiting. One of the important tasks for testing a mobile

application is to set the environment correctly. This differs from the web application as web applications can run directly in the browser whereas mobile applications require a different platform altogether to run. Let's start with the installation.

Setting it up

Requirements:

Software to run virtual machines, we will use virtual box for demonstration

Linux VM // Install in Vbox

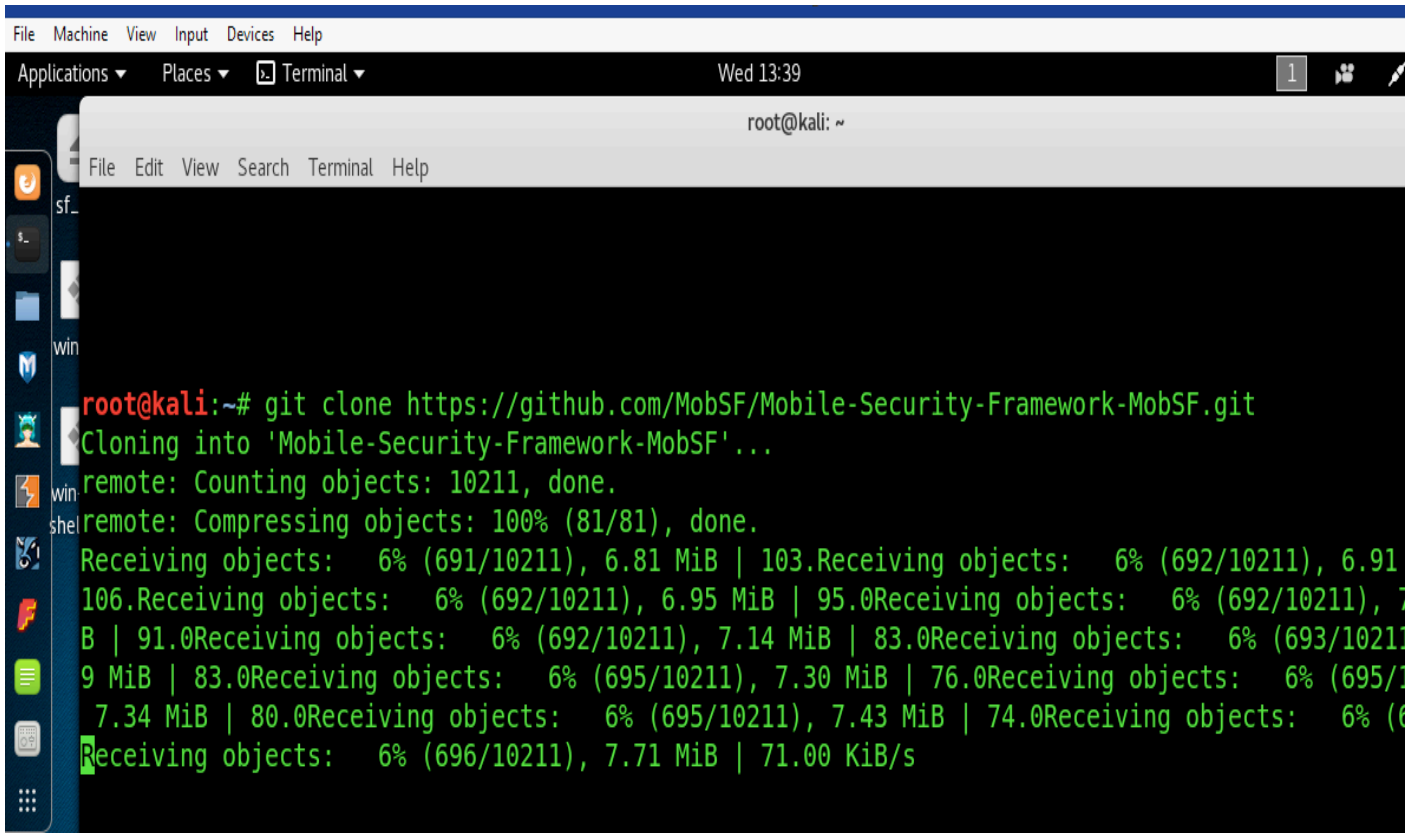
MobSF VM ova file (for dynamic analysis) // Import in Vbox

[Setting up Static analyzer]

Step 1: Open the Linux VM

Step 2: Run the below in terminal to install the server

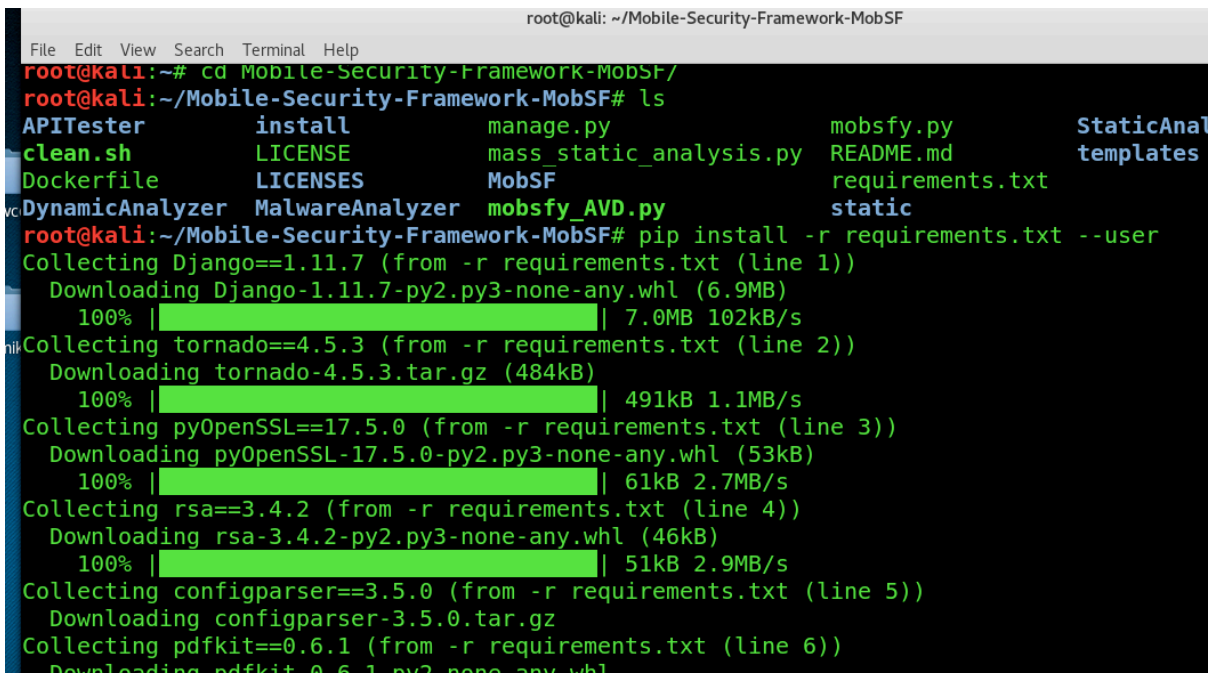
```
git clone https://github.com/MobSF/Mobile-Security-Framework-MobSF.git

cd Mobile-Security-Framework-MobSF
```



Step 3: Ensure that Python 3.6+ is installed (*Link to download*)

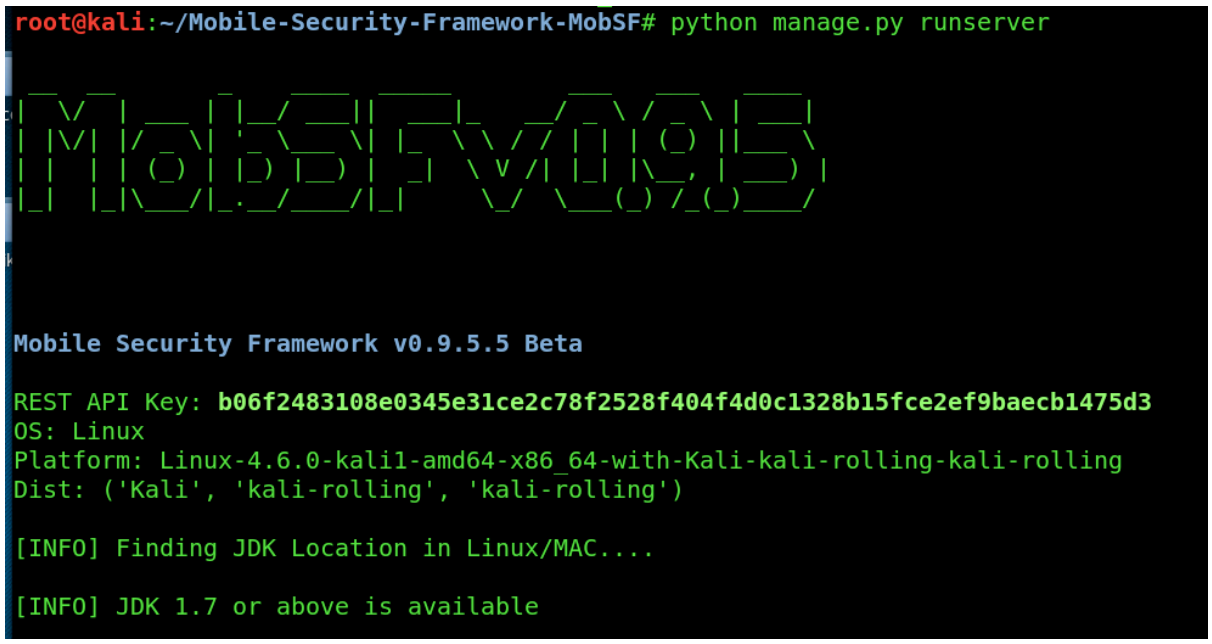
Step 4: Install Python dependencies - these are present in the requirements.txt file. Run the below command in the terminal to install

pip install -r requirements.txt

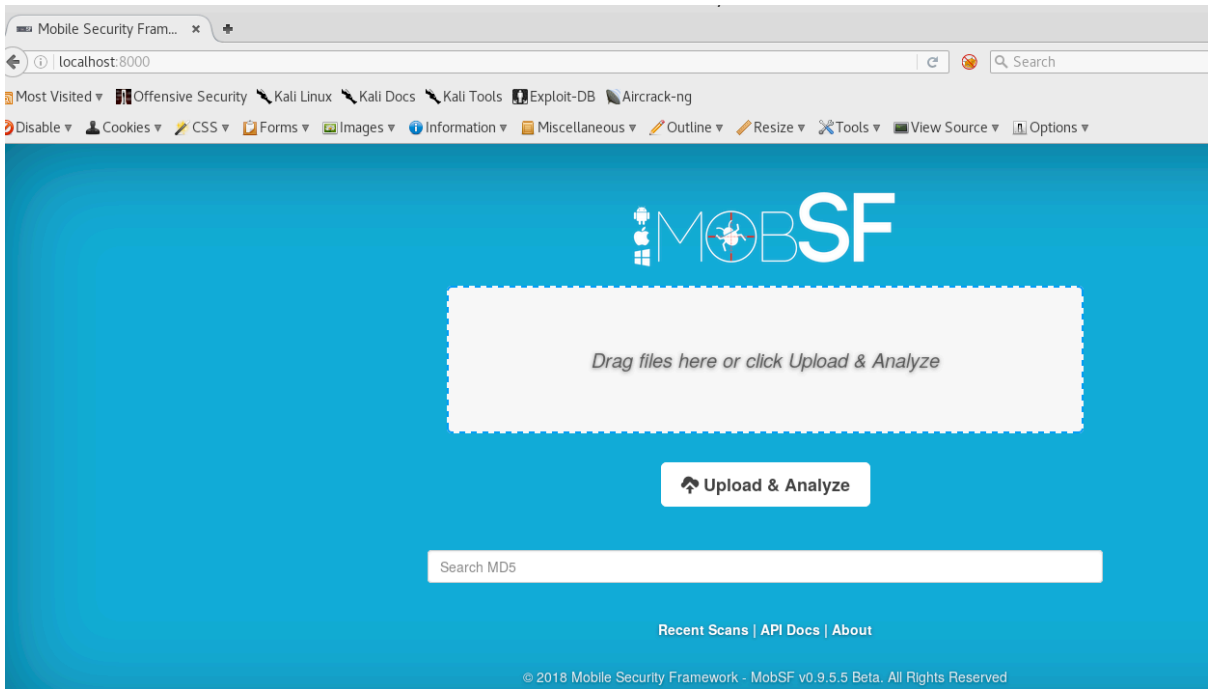


Step 5:Running the MobSF server

Python manage.py runserver

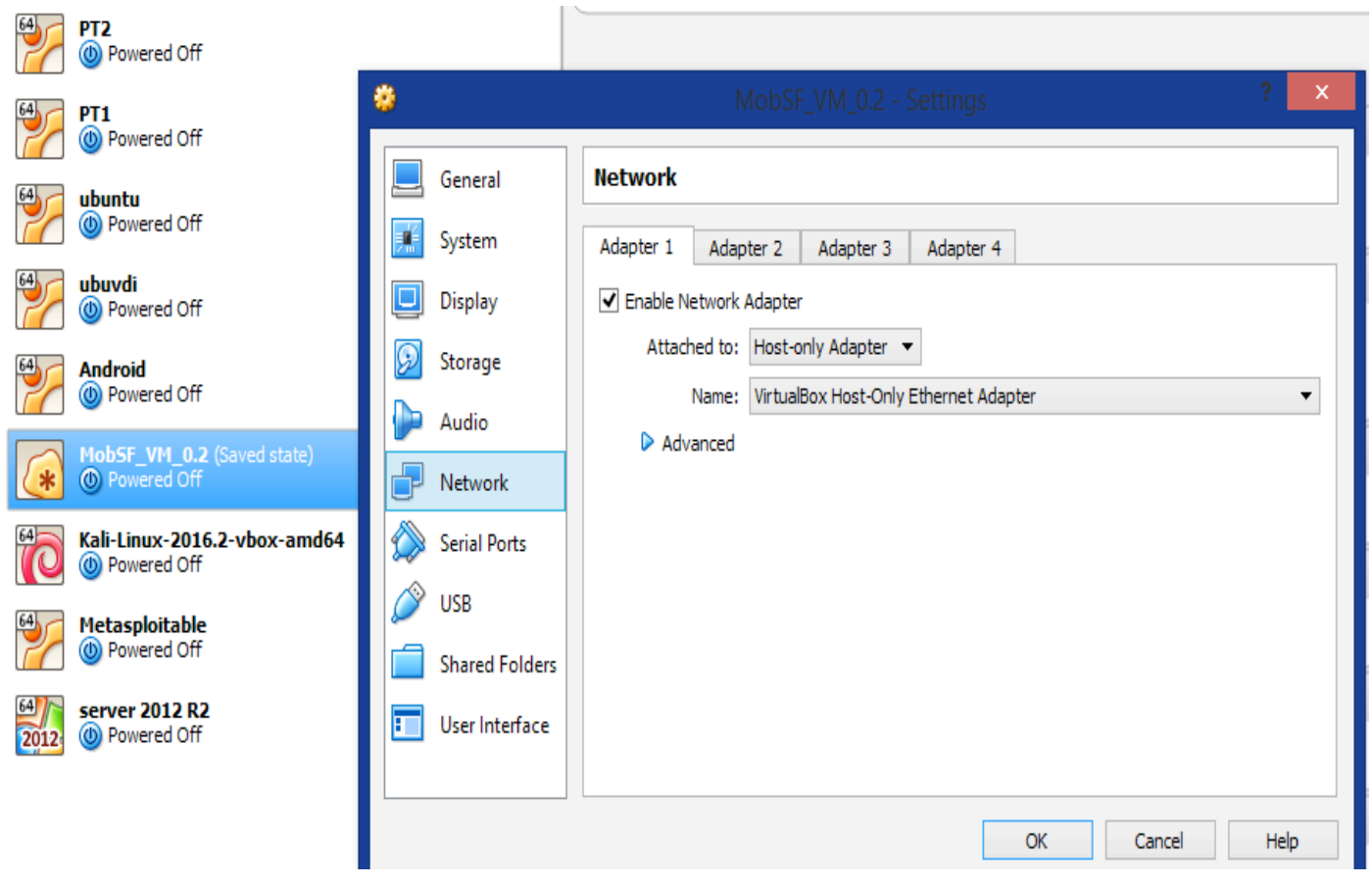


Step 6: Open the browser and you have the web interface at port 8000



[Setting up dynamic analyzer]

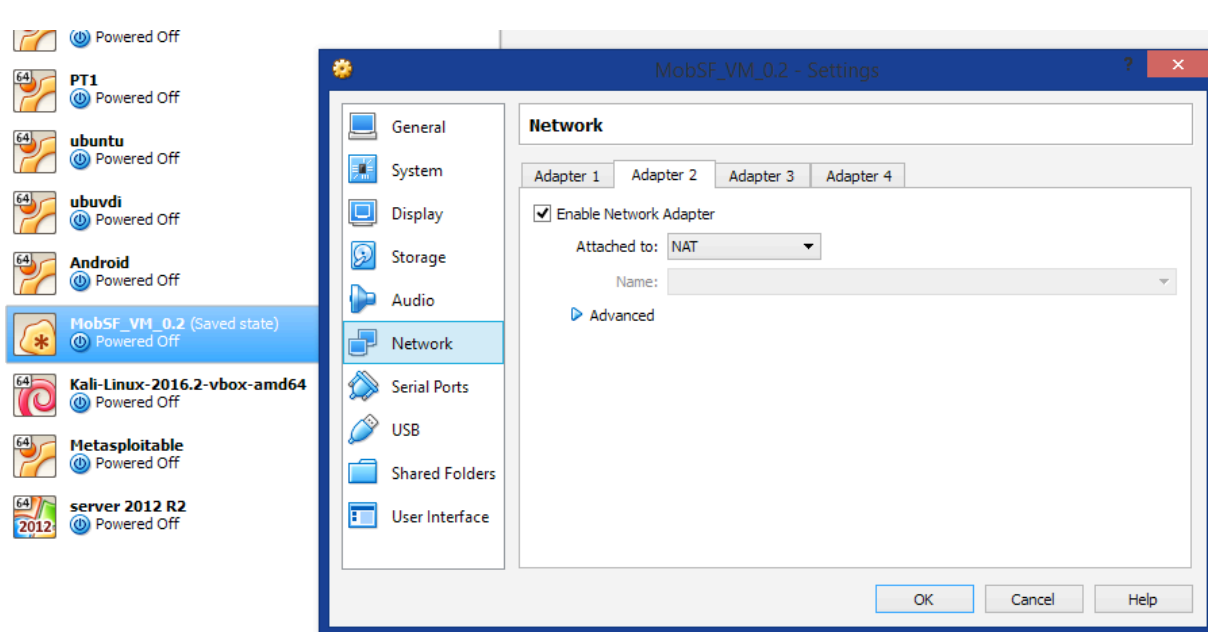
Step 1: Import the Android VM- MobSF (.ova) in Vbox



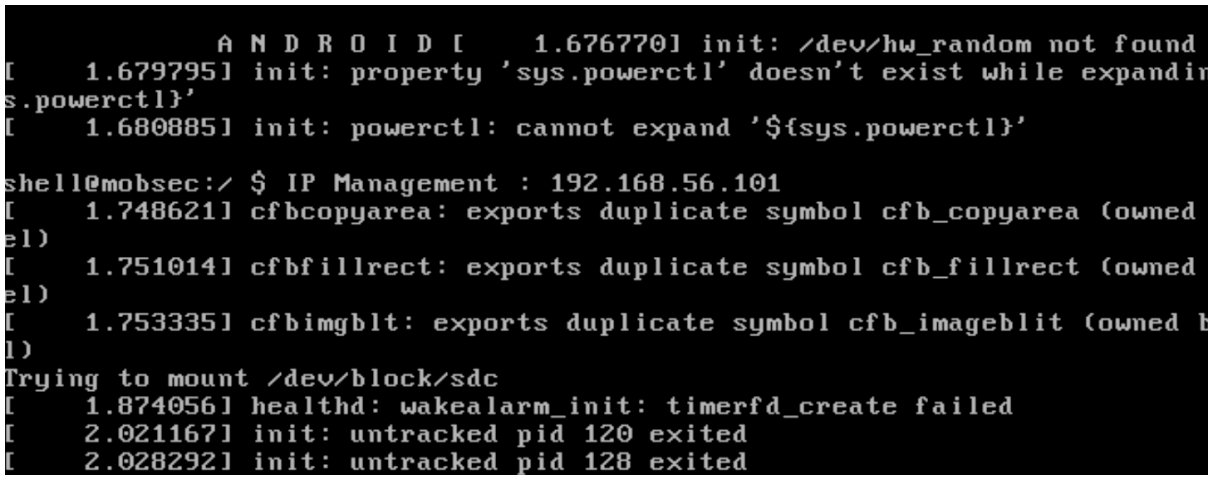
After starting, it should look like the below:

NOTE: password is 1234

Step 2: Change the below network settings, we will explain this setting in the architecture part later.



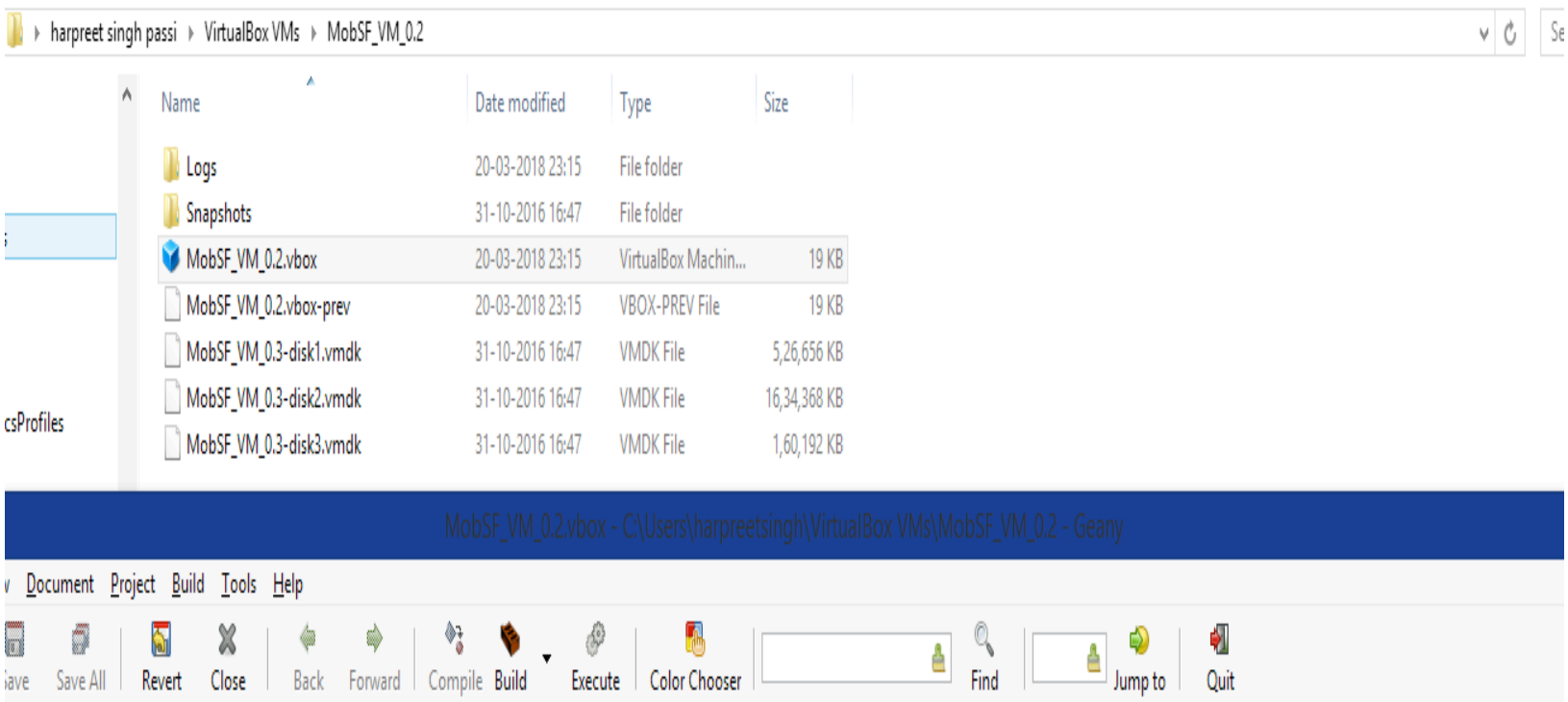
Step 3: Note the IP when the VM starts, I believe it is set to 192.168.56.101.

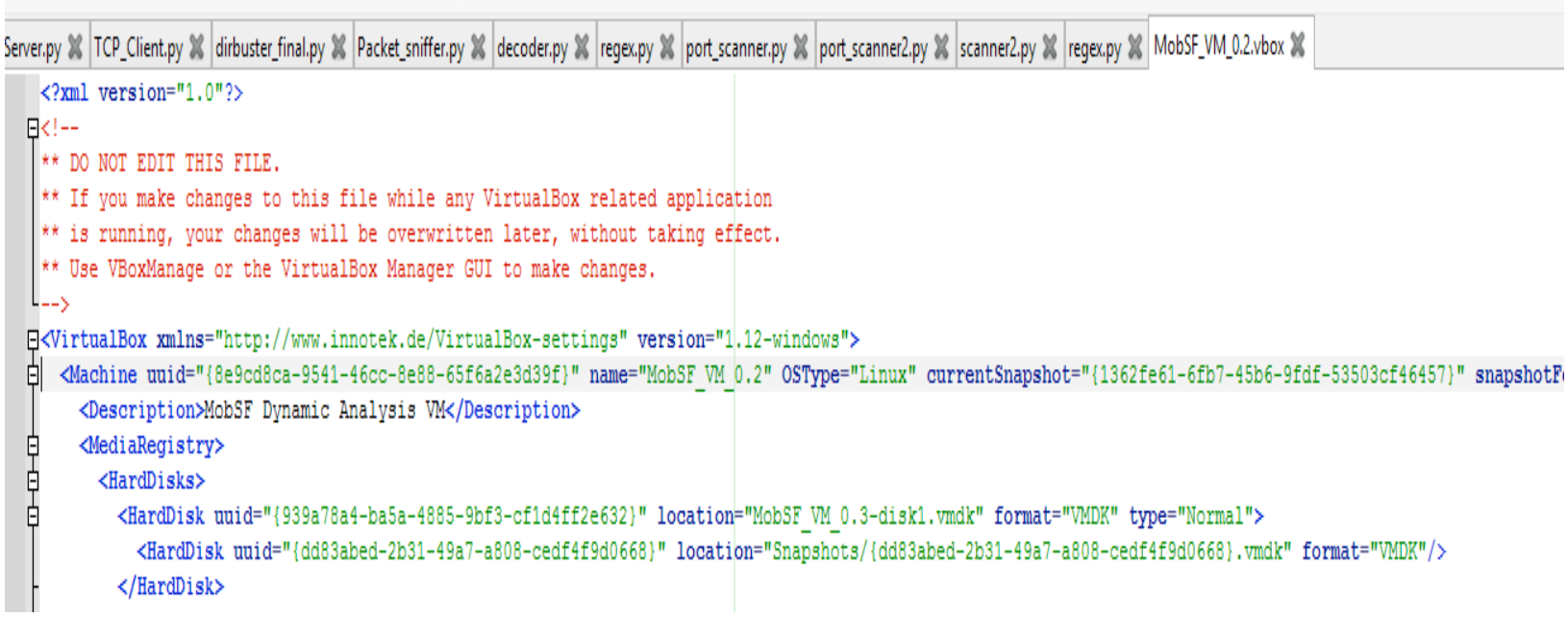


Step 4: Check the IP for VirtualBox host only network.

The IP in step 3 and step 4 should belong to the same network. If not, go to virtual box -> file-> preferences and set the IP to be in the same network.

Step 5: Save the VM state, right click on it, choose open in browser. You will find a file with .vbox extension. Open it in any editor, I have used Geany here. Note the UUID and current snapshot ID.





Step 6: Go back to the Linux box and go to MobSF/settings .py file. Find the dynamic analyzer settings part in the file and change the below:

VM_IP: change it to the one you noted in step 3

Proxy_IP: Change to the one in step 4

UUID: UUID of step 5

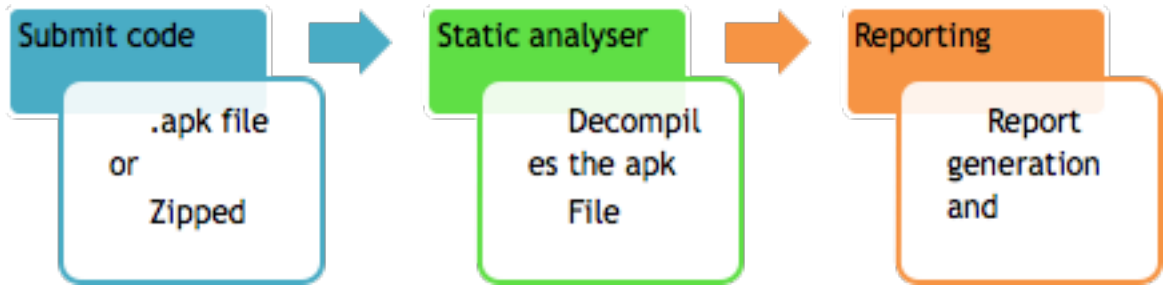
SUUID: Current snapshot ID step 5

```
root@kali: ~/Mobile-Security-Framework-MobSF/MobSF
File  Edit  View  Search  Terminal  Help
 GNU nano 2.7.4                                File: settings.py

  # DYNAMIC ANALYZER SETTINGS
  #-------------------------

  #========ANDROID DYNAMIC ANALYSIS SETTINGS================================

  ANDROID_DYNAMIC_ANALYZER = "MobSF_VM"

  # You can choose any of the below
  # 1. MobSF_VM
  # 2. MobSF_AVD
  # 3. MobSF_REAL_DEVICE

  '''
  MobSF_VM - x86 Android 4.4.2 running on VirtualBox (Fast, not all Apps work)
  MobSF_AVD - ARM Android 4.1.2 running on Android Emulator (Slow, Most Apps work)
  MobSF_REAL_DEVICE - Rooted Android 4.03 - 4.4 Device (Very Fast, All Apps work)
  '''

  #=========================================================================
  #================================================

  #====ANDROID MOBSF VIRTUALBOX VM SETTINGS =====
  # VM UUID
  UUID = '8e9cd8ca-9541-46cc-8e88-65f6a2e3d39f'
  # Snapshot UUID
  SUUID = '1362fe61-6fb7-45b6-9fdf-53503cf46457'
  # IP of the MobSF VM
  VM_IP = '192.168.56.101'
  VM_ADB_PORT = 5555
  VM_TIMEOUT = 100
  #================================================

  #-------------------------
  # MobSF MITM PROXY SETTINGS
  #-------------------------

  #================HOST/PROXY SETTINGS ===============
  PROXY_IP = '192.168.56.1'  # Host/Server/Proxy IP
  PORT = 1337  # Proxy Port
  ROOT_CA = '0025aabb.0'
  SCREEN_IP = PROXY_IP  # ScreenCast IP
  SCREEN_PORT = 9339  # ScreenCast Port(Do not Change)
  #=====================================================
```

We are all set.

Static analysis

Static analysis deals with analyzing the dead code without running it. Below is the architecture of the static analyser of MobSF.

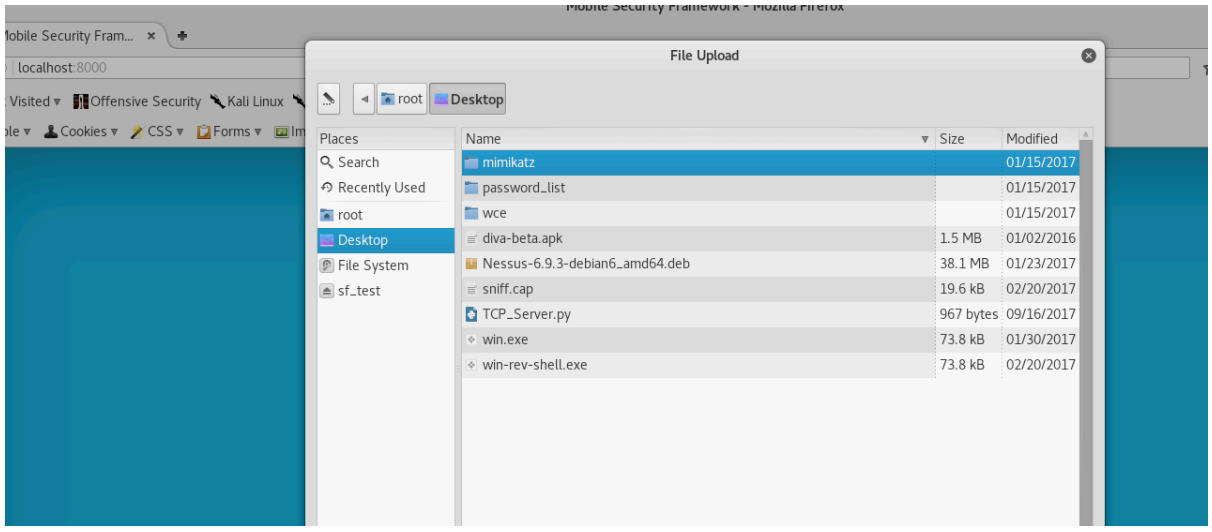


The analyser has the ability to accept and analyse both .apk files and Java source code written in either Eclipse or Android studio. Once the files are uploaded, the analyser starts the job and the services can be viewed in the server terminal window.

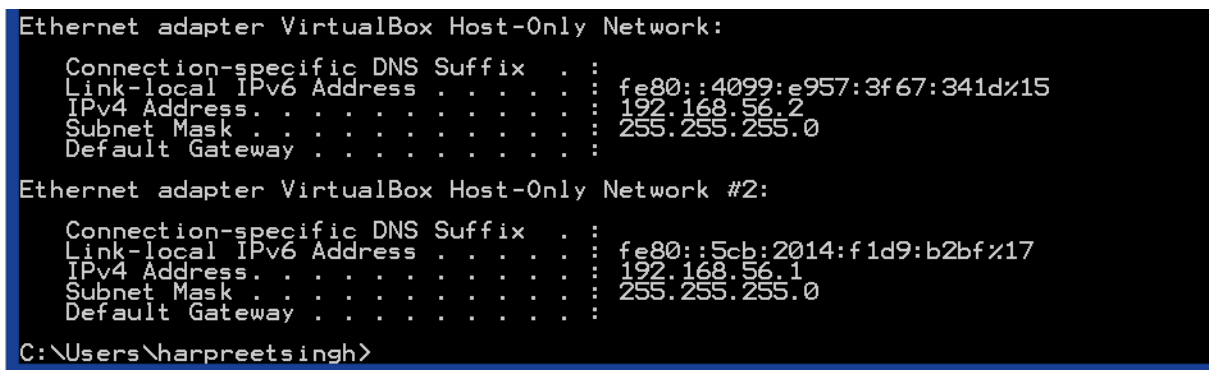
[DEMO]

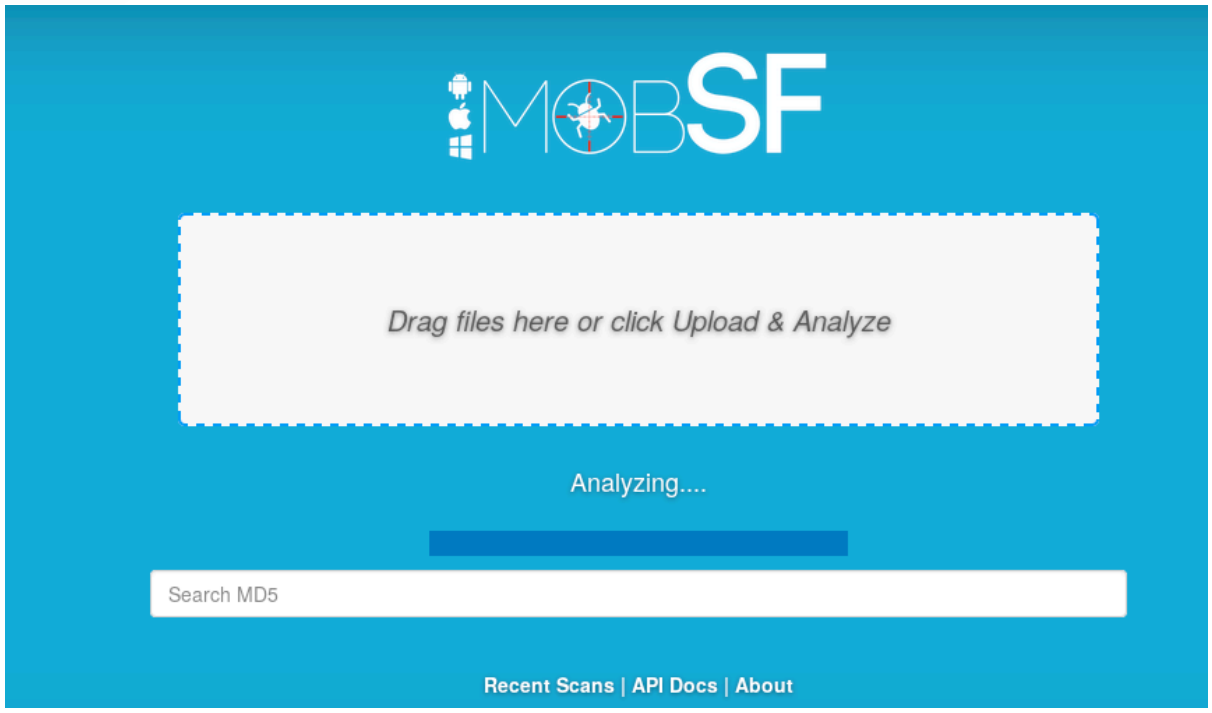
For the demo, I have taken the already vulnerable DIVA application.

Step 1: Run the server (refer installation) and open the browser to upload the application

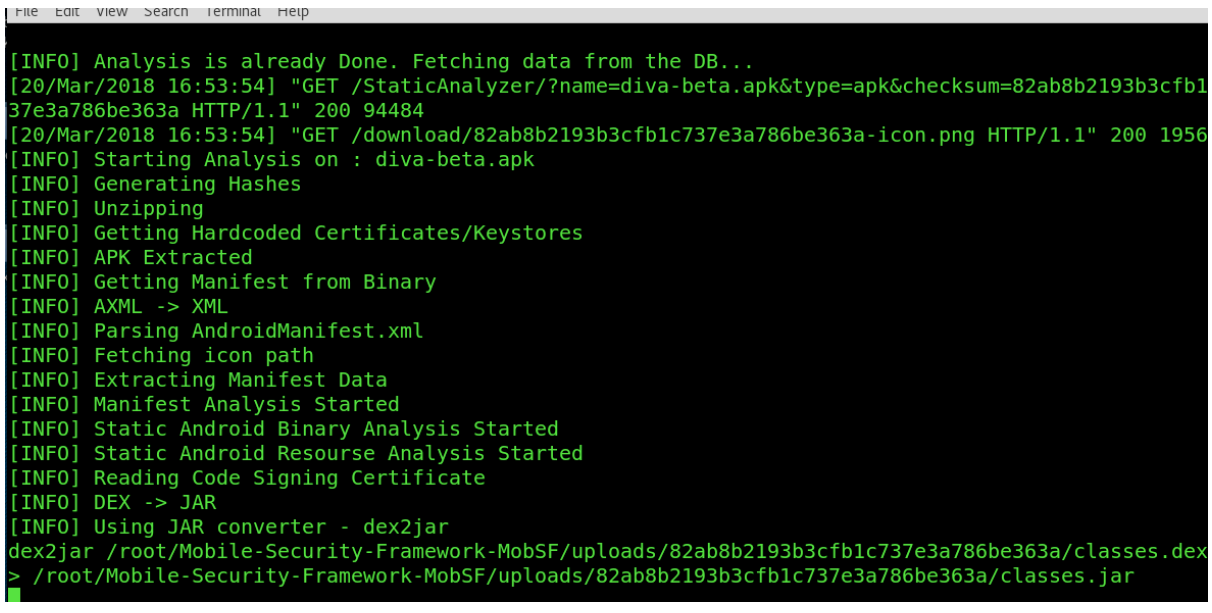


Step 2: Once the analysis starts (below screen), check the terminal where the runserver command was executed.





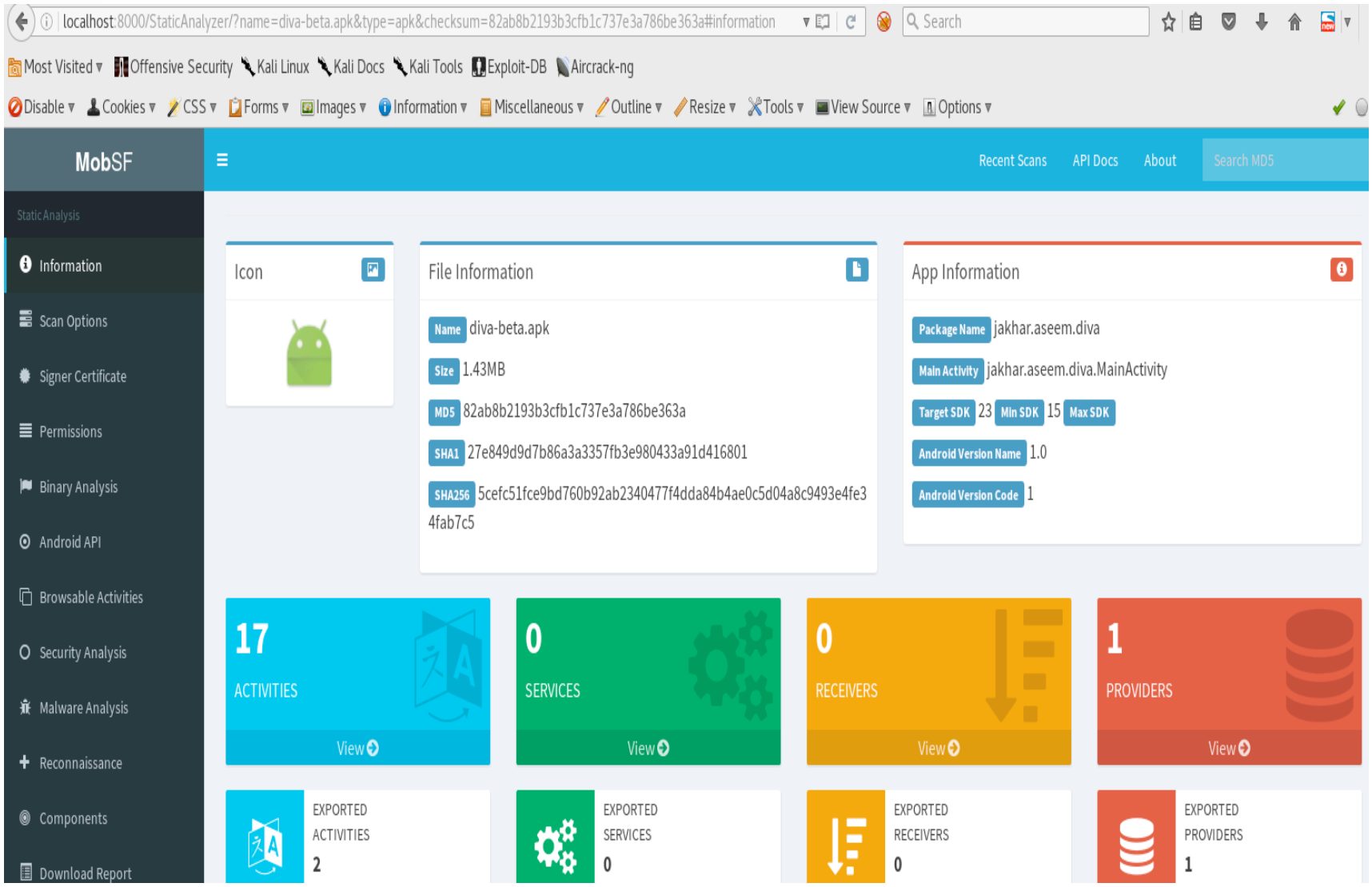
Here you can see the operations being performed on the application being analyzed. If there is any error, that can be tracked from here and rectified. This is just for reference for what the analyzer is doing with the application.



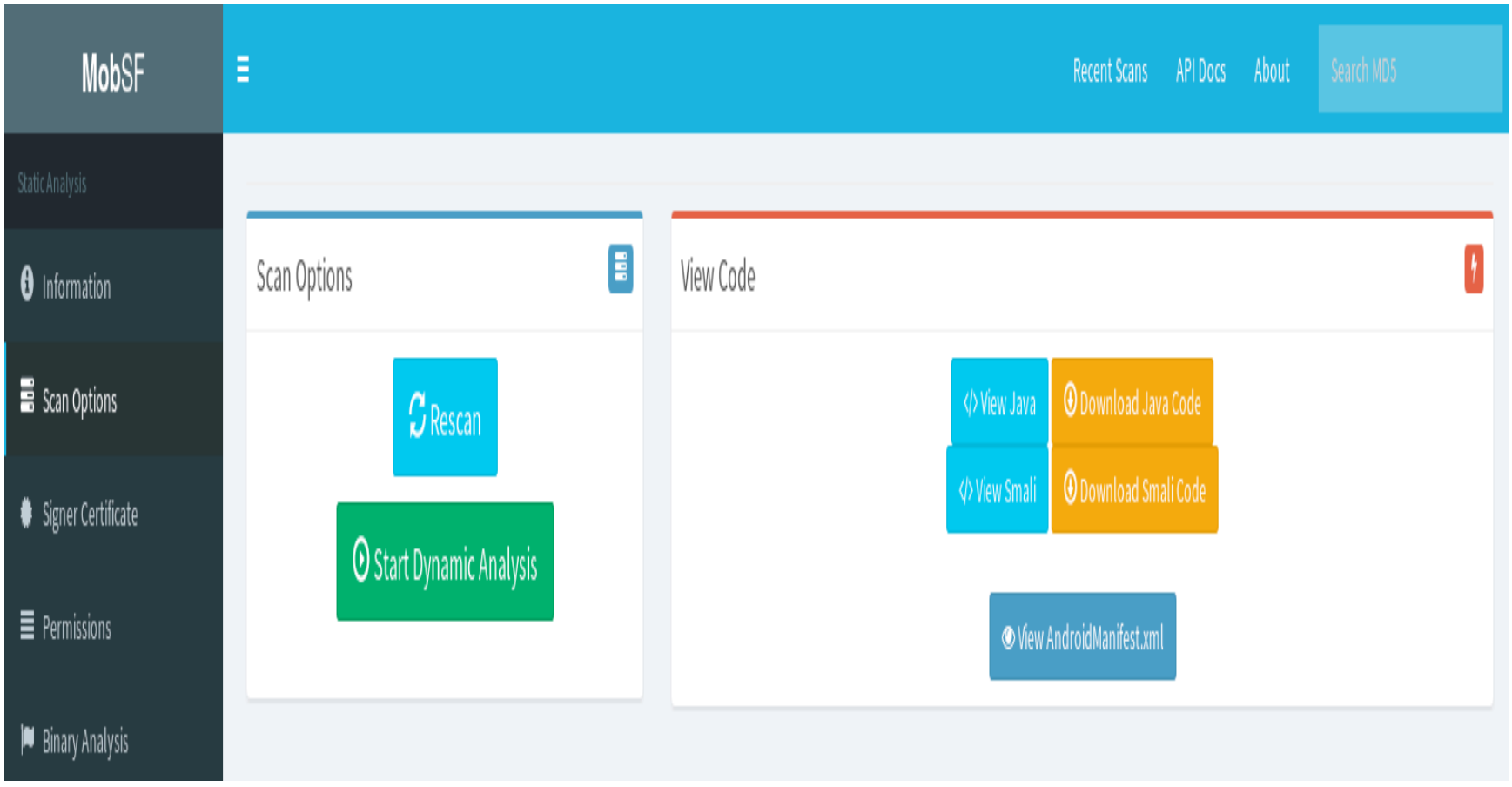
Analyzer will automatically generate the report once done with the analysis. Below is the report for DIVA. Let's have a closer look at the report.

The report is divided into multiple tabs. First is the Information tab, which covers the general information, like app icon, app name, size, package name, etc.

You may notice that the MD5 and SHA1 are also shown, which can be useful to detect known malicious applications.



Then we have the scan options, you may choose to either rescan the application, start the dynamic analysis or check the Java code and the manifest file. We will get to the dynamic analysis later in this article.

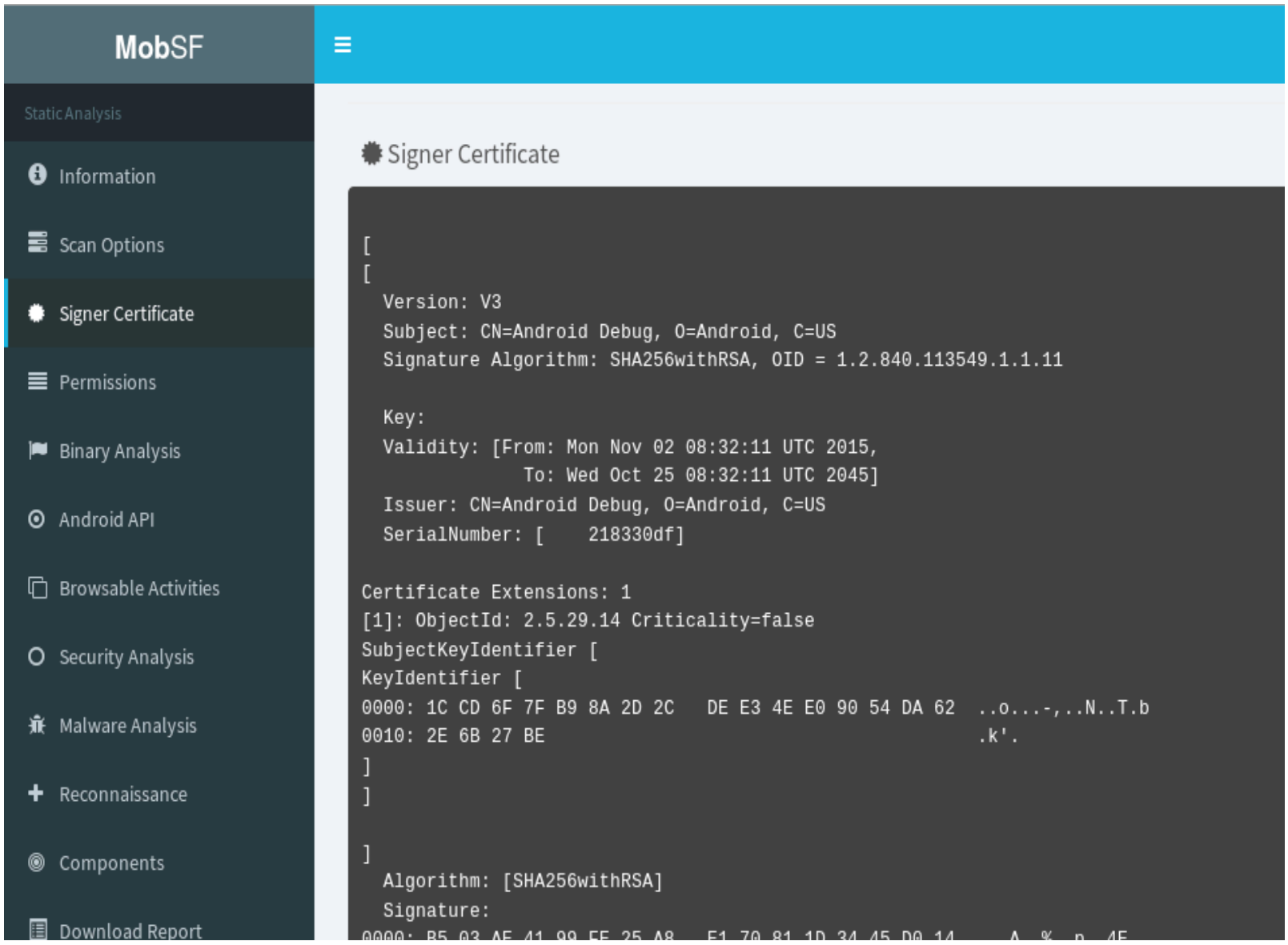


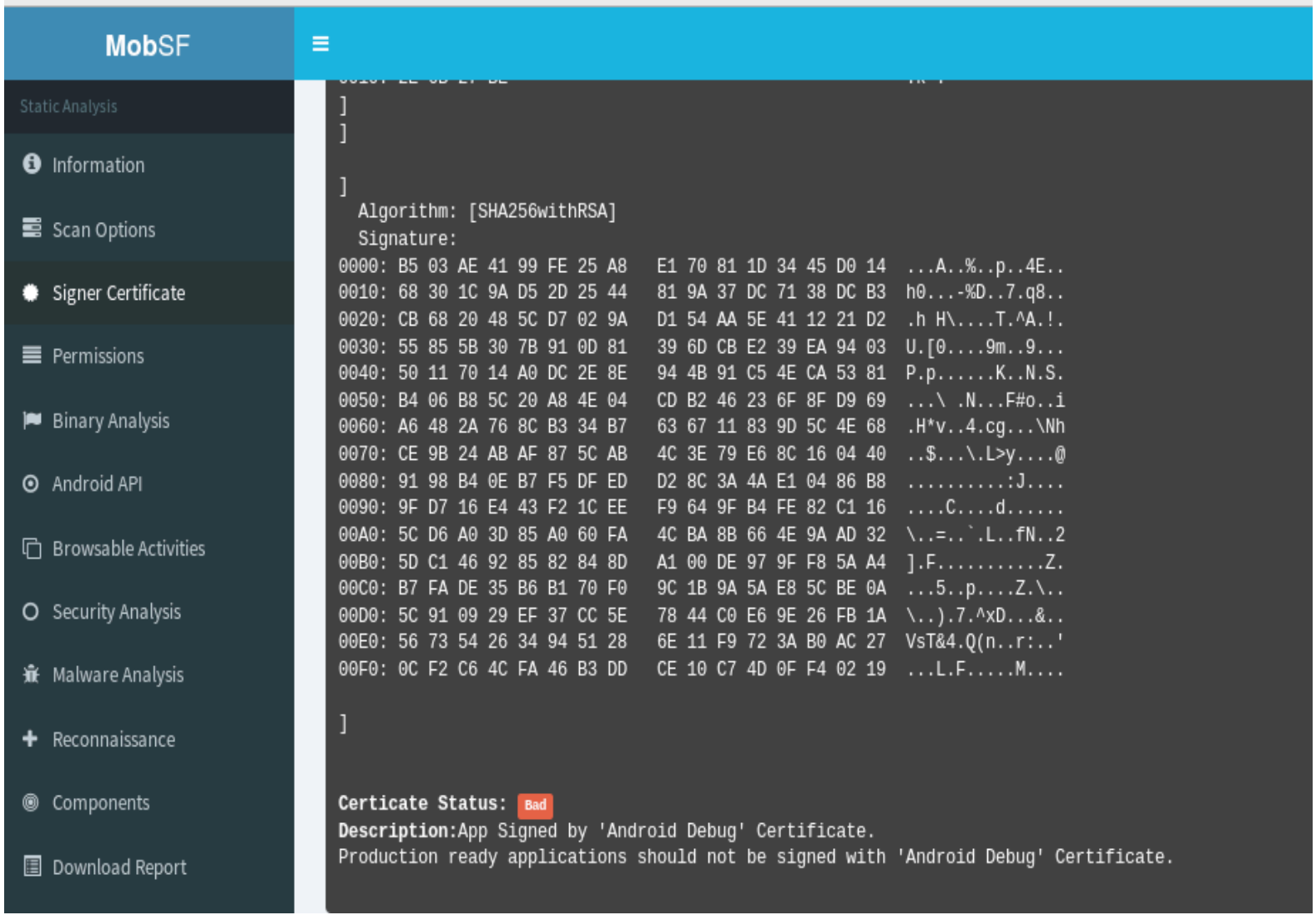
Below is a sample manifest file generated by the analyzer. The alignment in this case is a little off but that can be adjusted. There is a lot of information that can be inferred from the manifest file, like permissions required.

AndroidManifest.xml

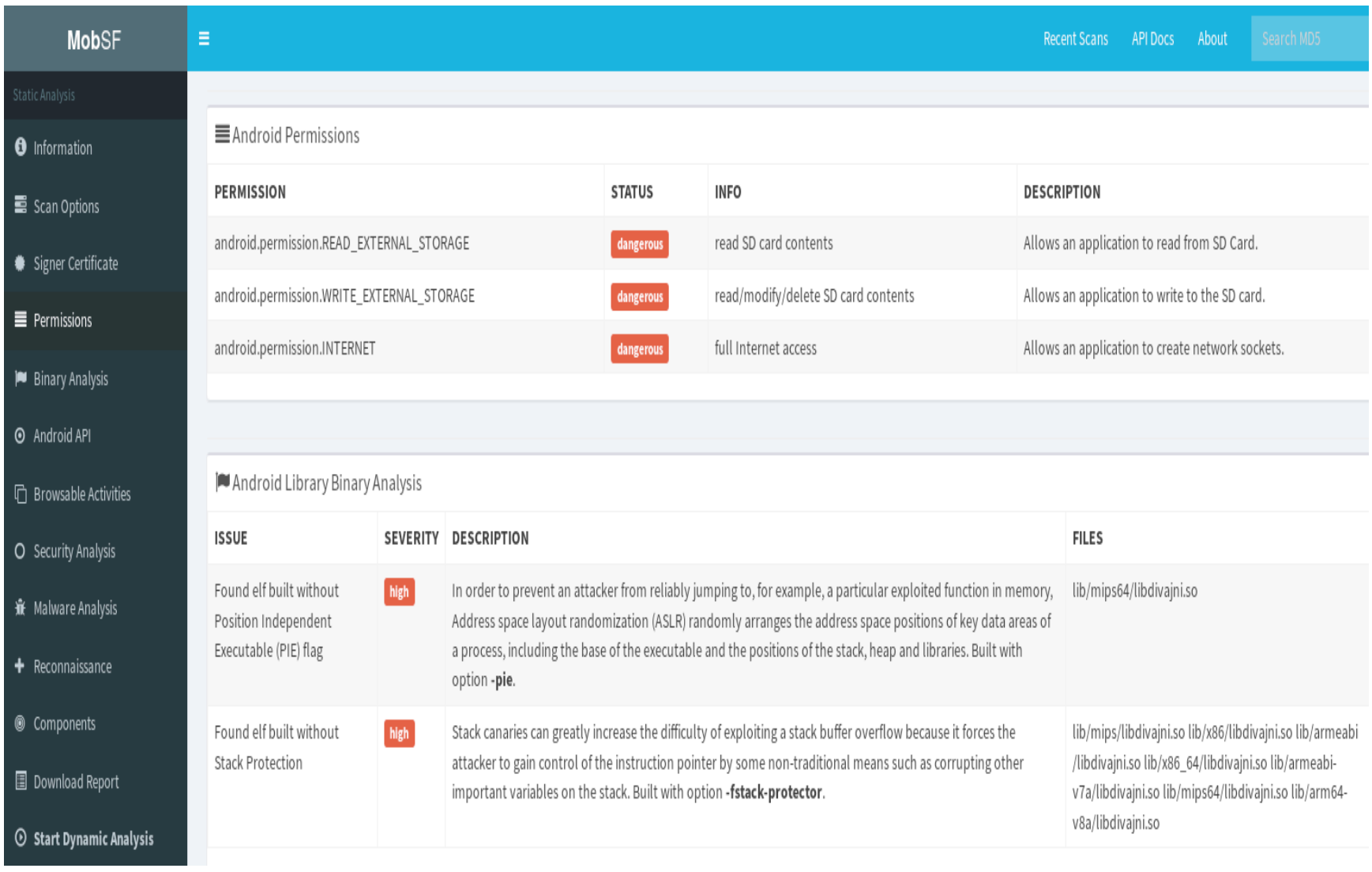
```
<?xml version="1.0" encoding="utf-8"?><manifest xmlns:android="http://schemas.android.com/apk/res/android" android:vers
ionCode="1" android:versionName="1.0" package="jakhar.aseem.diva" platformBuildVersionCode="23" platformBuildVersion
Name="6.0-2166767" > <uses-sdk android:minSdkVersion="15" android:targetSdkVersion="23"
> </uses-sdk> <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"
> </uses-permission> <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"
> </uses-permission> <uses-permission android:name="android.permission.INTERNET" >
</uses-permission> <application android:theme="@7F090083" android:label="@7F060023"
android:icon="@7F030000" android:debuggable="true" android:allowBackup="true" andr
oid:supportsRtl="true" > <activity android:theme="@7F090030" andr
oid:label="@7F060023" android:name="jakhar.aseem.diva.MainActivity" > <int
ent-filter > <action android:name="androi
d.intent.action.MAIN" > </action> <cat
egory android:name="android.intent.category.LAUNCHER" >
</category> </intent-filter> </activity> <activity
android:label="@7F060027" android:name="jakhar.aseem.diva.LogActivity" >
</activity> <activity android:label="@7F06002C" android:name
="jakhar.aseem.diva.HardcodeActivity" > </activity> <activity andr
oid:label="@7F06002D" android:name="jakhar.aseem.diva.InsecureDataStorage1Activity" >
</activity> <activity android:label="@7F06002E" android:name="jakhar
.aseem.diva.InsecureDataStorage2Activity" > </activity> <activity
android:label="@7F06002F" android:name="jakhar.aseem.diva.InsecureDataStorage3Activity"
> </activity> <activity android:label="@7F060030" andr
oid:name="jakhar.aseem.diva.InsecureDataStorage4Activity" > </activity> <activity
android:label="@7F060031" android:name="jakhar.aseem.diva.SQLInjectionActivity"
> </activity> <activity android:label="@7F060032"
android:name="jakhar.aseem.diva.InputValidation2URISchemeActivity" > </activity>
<activity android:label="@7F060033" android:name="jakhar.aseem.diva.AccessContro
```

Next is the certificate information. Let us assume a case when a particular banking application is available only at a particular website or a store. The legitimacy that the application has come from the original source will be decided using this certificate. If a hacker has changed the application, then it has to be signed and that will be different from the original certificate.

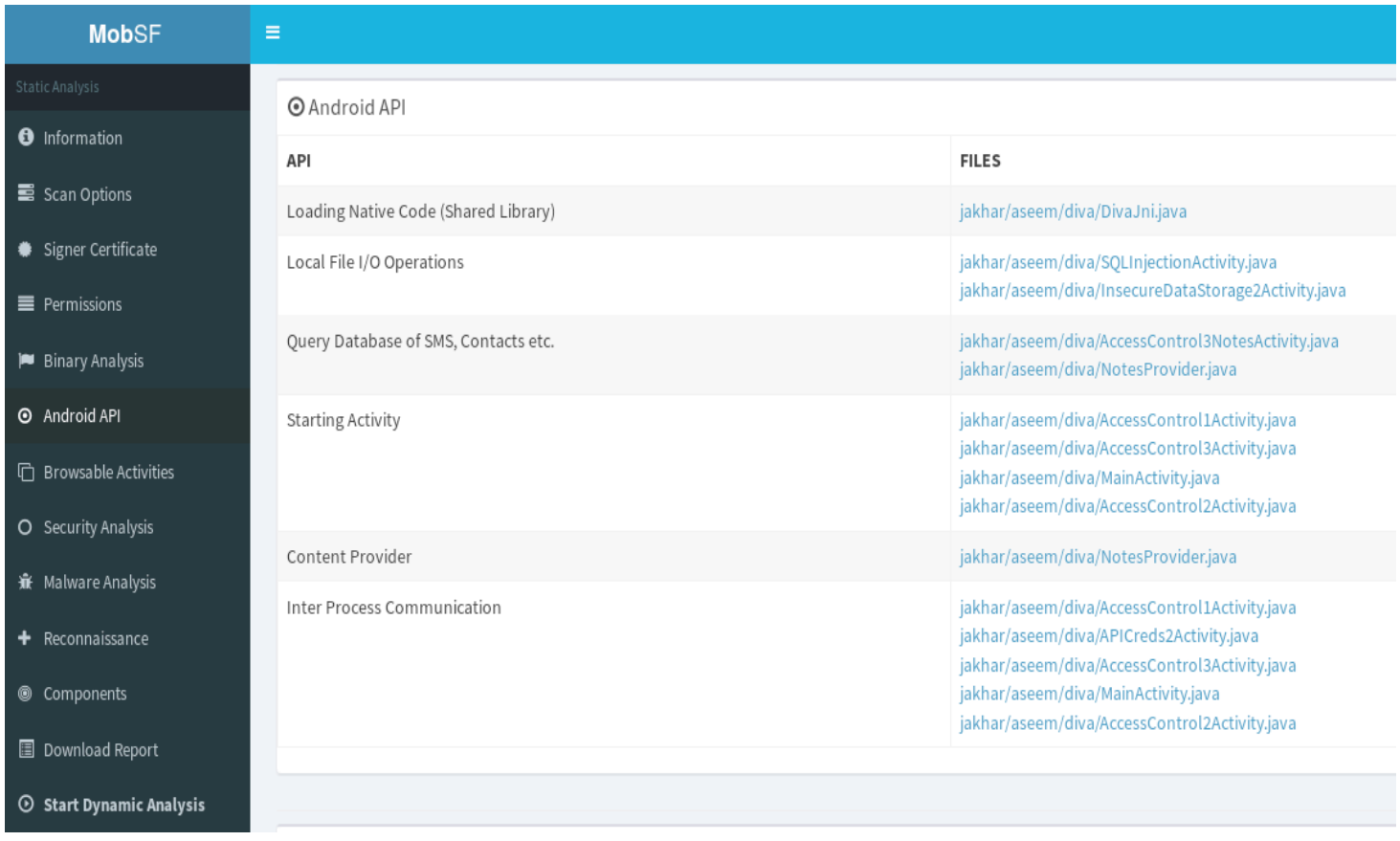




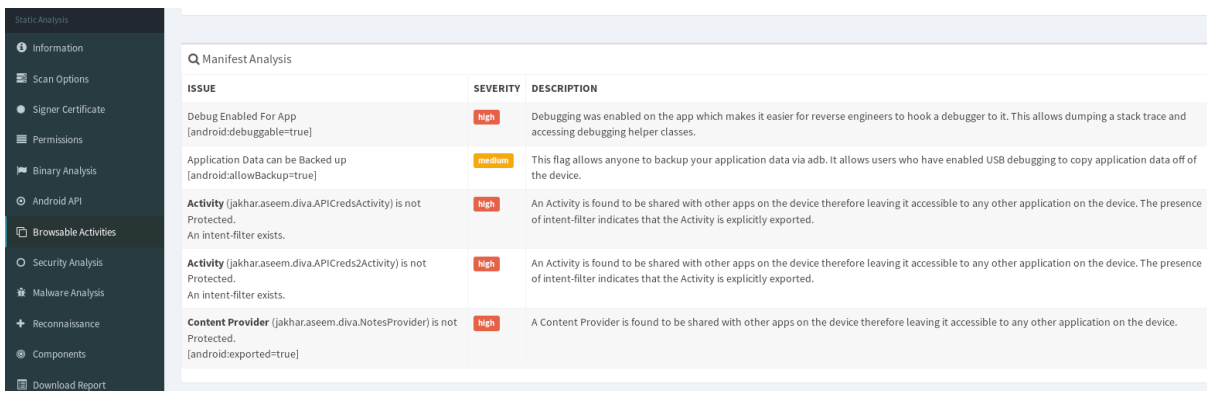
Then we have Android permissions and binary analysis. Permissions are present in the manifest files, the reporting format makes it easy for you to analyze the dangerous permissions that may not be required.

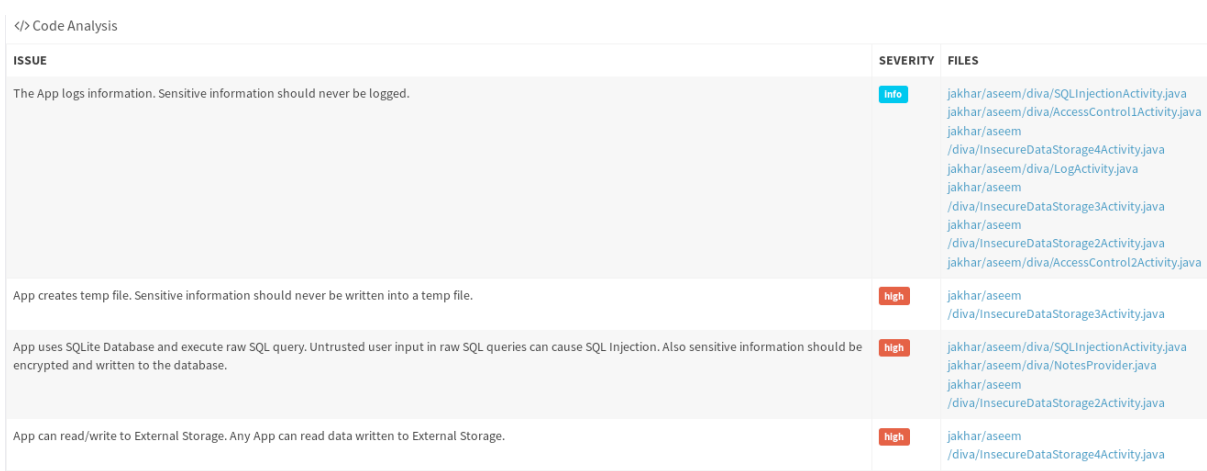


Then we have the API information, this will help the users who are doing a black box testing learn about the flow of the application. The details of the Java files are mentioned with respect to the categories, like interprocess communication, SMS APIs, crypto services, broadcasting APIs, etc.



The manifest file contains a lot of details, like permissions, which are exclusively covered previously in the report, this section covers the overall analysis of the manifest file. The analysis is further classified on the basis of severity and description. The tool also performs code analysis, which covers issues in the code and how the flow works, how logs are getting stored, is the application storing any sensitive data, is the database encrypted, etc. Malware analysis is also performed and reported for suspicious activities. Then we again have the option to start dynamic analysis, which will be discussed now.



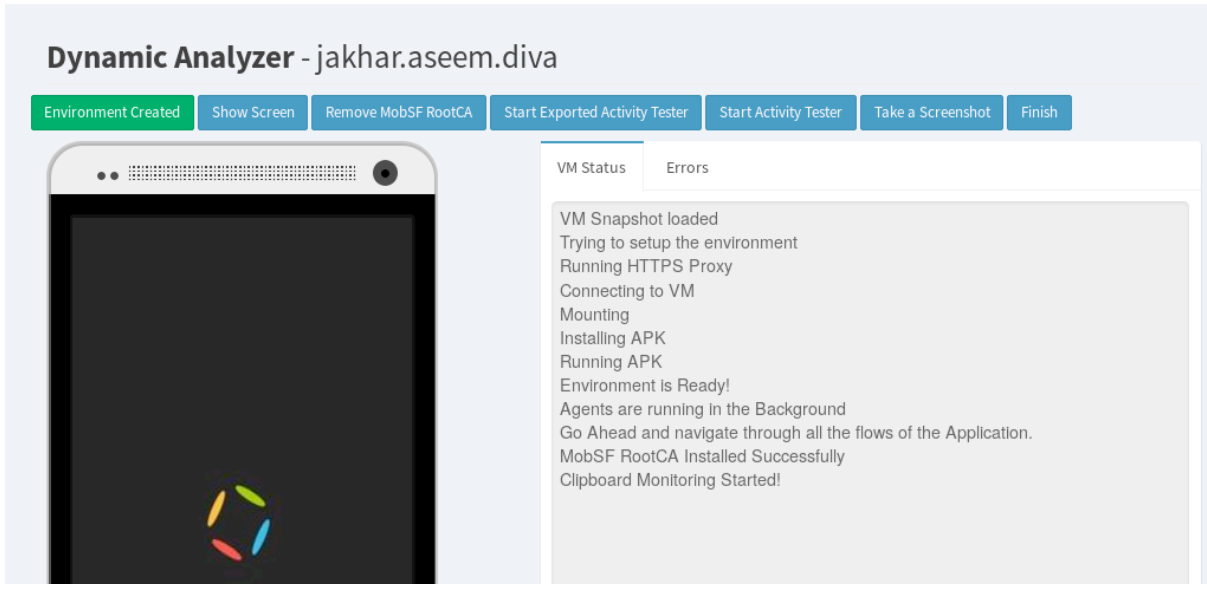


Dynamic analysis

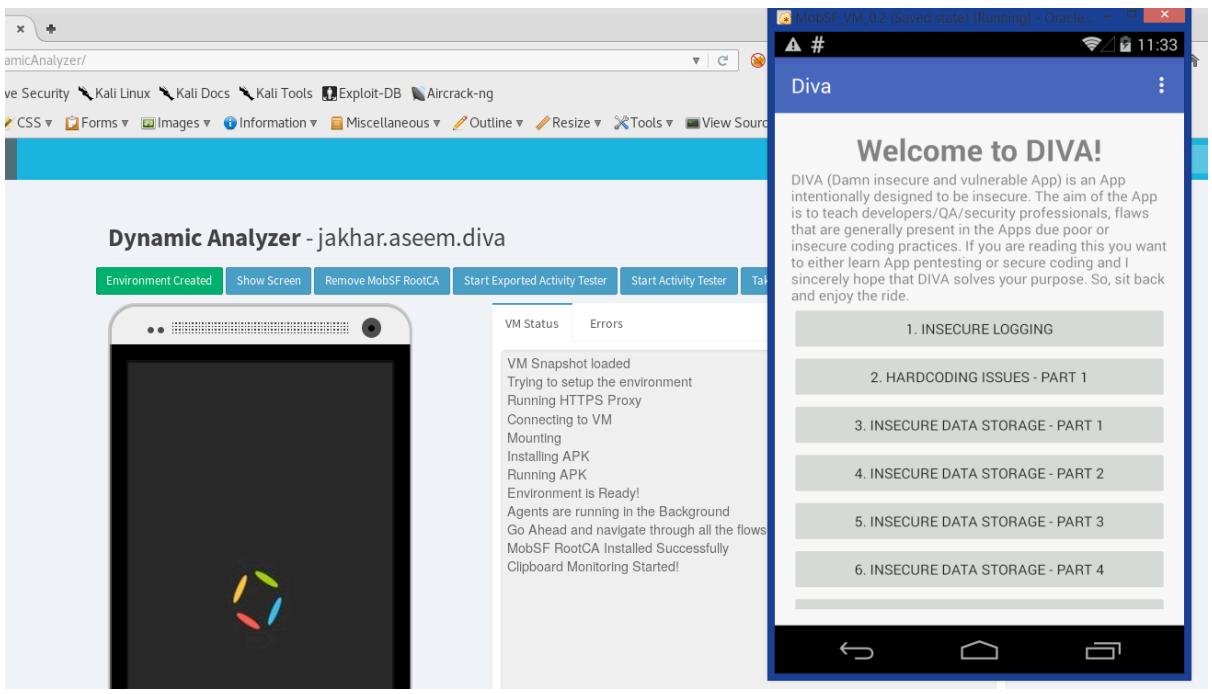
Dynamic analysis is performed by running the application in either a real device or a simulator. The user has to iterate through different flows of the application with agents monitoring the application flows and tracking them. The collected information is then analyzed for sensitive data access, hardcoded details, traffic analysis, insecure requests, etc., which are all reported. Even though this requires manual intervention, it is much more efficient than intercepting the traffic at each step and performing the analysis. Here for demo we are using the MobSF VM but you have an option to choose VM, arm or real device (no agents required). Testing done with a real device is fast whereas the VM may not run the application sometimes.

[DEMO]

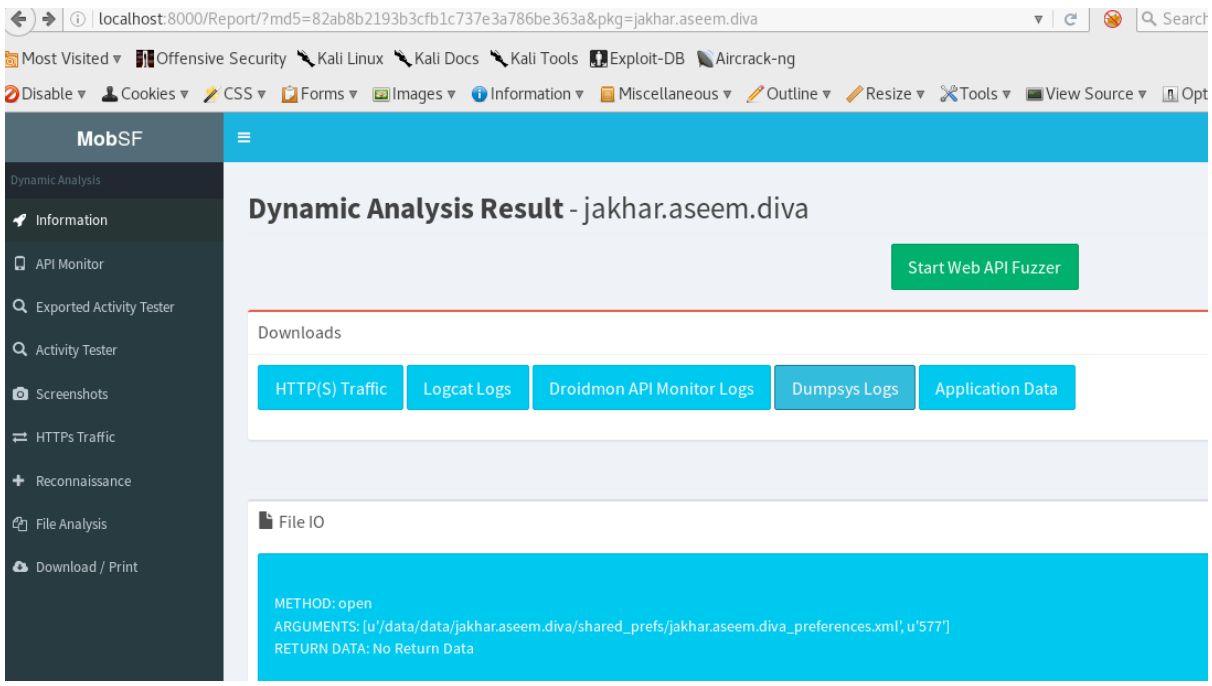
Select the option to start the dynamic analysis in the static analysis report. Status will be updated on the screen and the application will be flashed on the Android screen. Once the application starts, you have multiple options to remove the certificate, test the activities/screens/flow of the applications, capture screenshot and finish the testing.



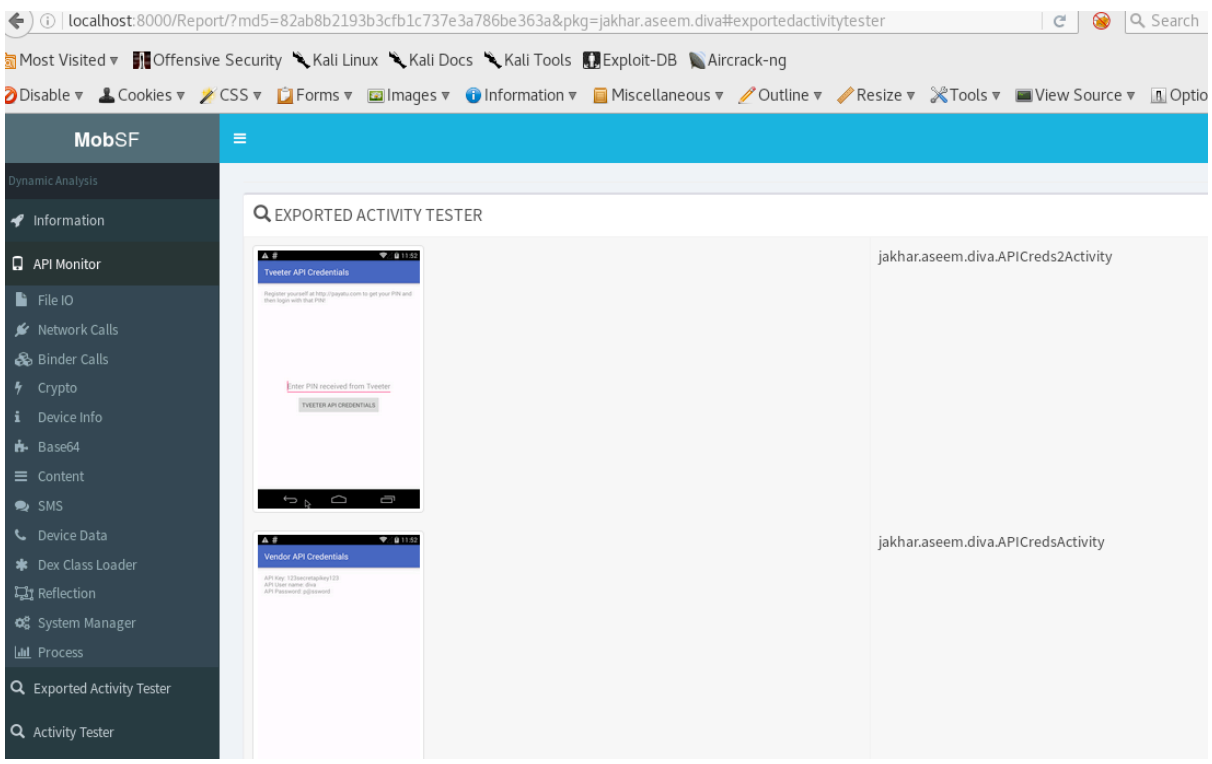
Once the application starts, iterate through the various flows of the applications and use the application, try all the possible options so that the agents collect the information.



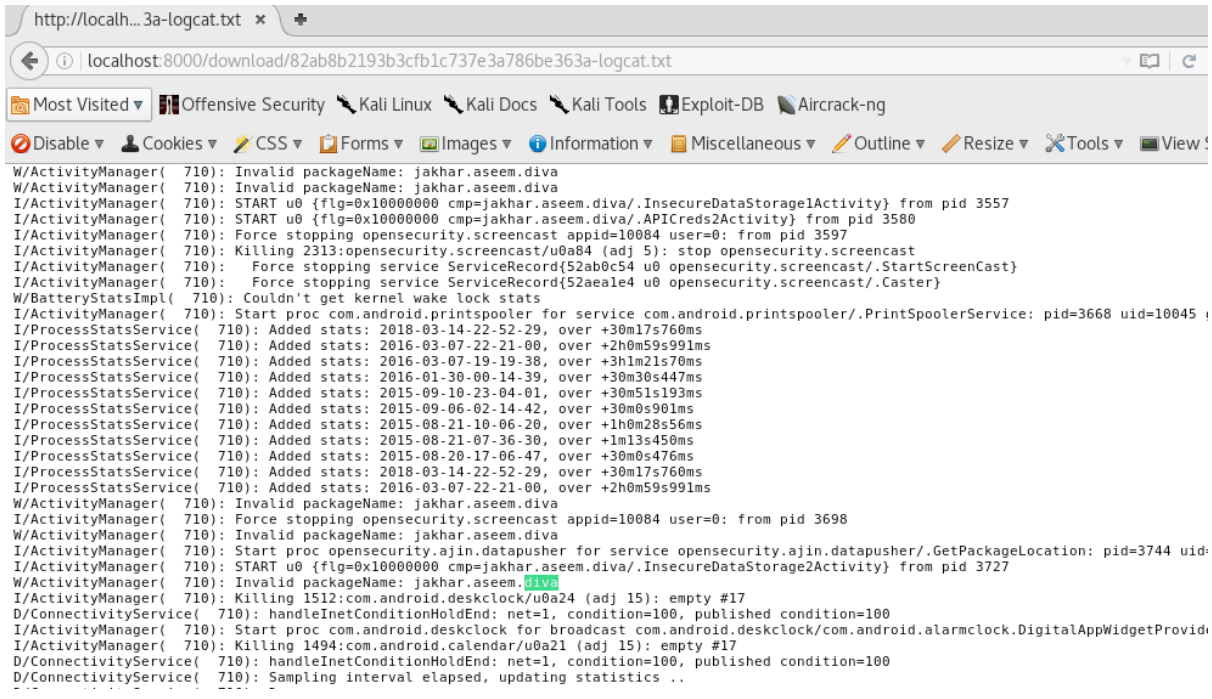
Once you click finish, the report for dynamic analysis will be generated and displayed. The report has tabs for HTTP(s) traffic, log analysis, API analysis and analysis of application analysis.



API monitor has further subdivisions in the report for file operations, crypto options, dynamically invoked URLs and emails, process calls, etc.



Logs can be viewed for further searching and analysis. Logs can help you find stored tokens, hardcoded strings, if any, whether sensitive data is getting logged, etc.



One of the great things about the framework is the capability to perform both static and dynamic analysis along with reporting capability. The report can be downloaded for further reference and analysis. MobSF reduces a pentester's time considerably to test the application using various other tools and analyzing smali files and flows. This can help beginner pentesters to start the analysis and then take it to the next level, saving time and effort.

Automate your own stuff

There are a lot of things the tool will not be able to do. This can range from a simple, yet time consuming, task or a complex one time task. These can be automated by the pentester, which will save him time and effort. This will also enhance the capability since some security tools detect and block the tool based scanning and analysis, etc.

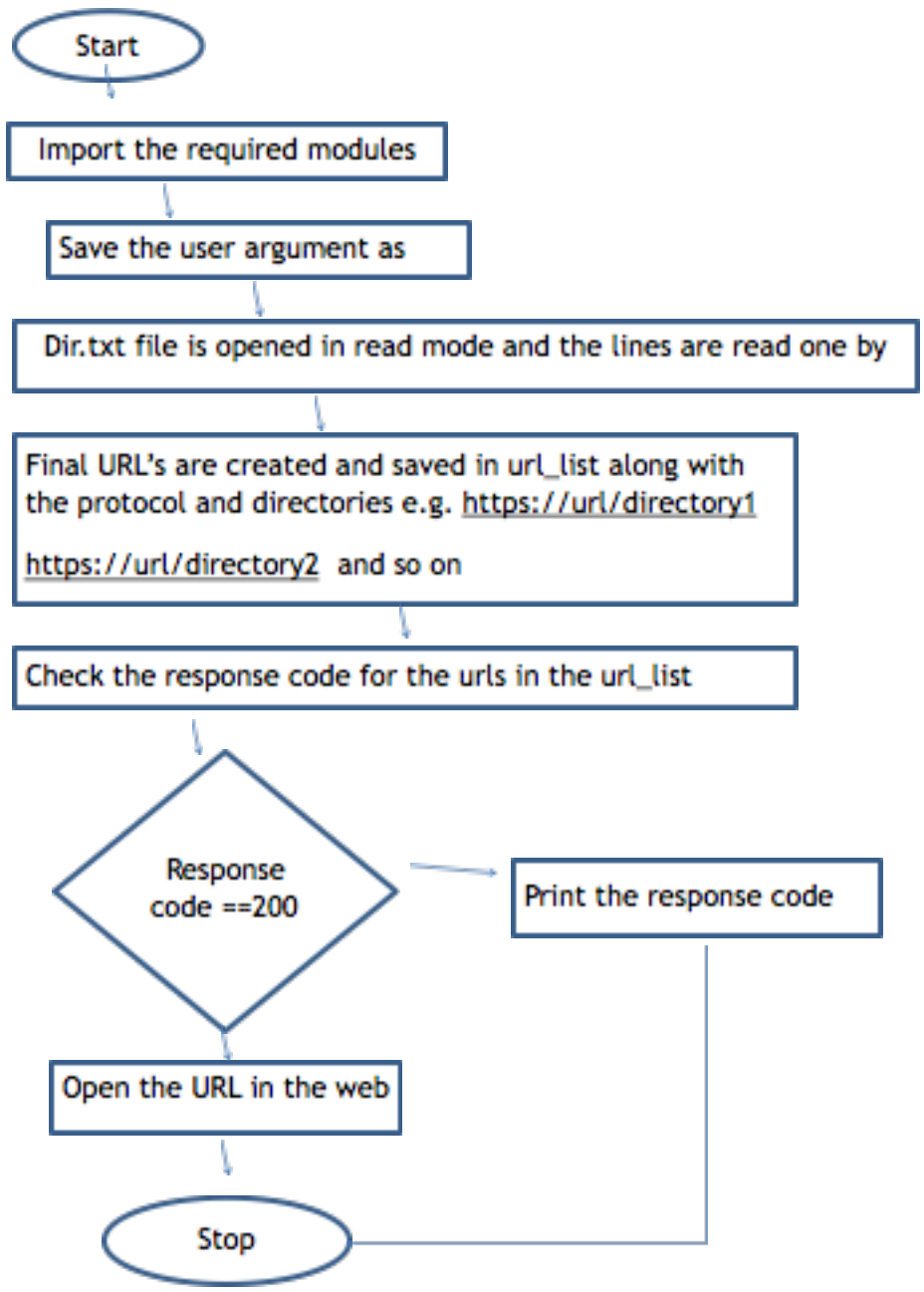
We will be using Python to automate a few tasks. Pentesters can code their stuff based on their requirements.

Bursting the directories

During the course of a penetration test, you may encounter tasks that a tool may not accomplish. One such task that I faced during one of the tests is to identify if a set of default directories were present in the target. I know that there are tools like dirbuster that can be of use, but I had none with no accessibility to the internet. These kind of situations forced me to develop a code that will do the job. The code is simple. There are two objects that we are going to play with. One is the IP that is the target and the other is the list of directories that will be checked. The code will generate the complete URL by appending the directory names to the target IP. Once that is done, it will be checked whether that path is accessible on the internet. If yes, then the response code is 200 ok, which will prove that the directory is present. The directory will be picked up one by one from a list of directories.

The below discussed code will take two inputs – URL/IP address and the directory list that you would like to test. It will test for the existence of the directories and will open the URI in the browser if it exists.

Code Flow



Code and comments

```
"""

***** USAGE ****

Change the paths in the code below where the script is saved.

This one is saved in 'C:\Users\harpreetsingh\Desktop\py_scripts'

Example 1: python dirbuster_final.py 4.2.2.2

Example 2: python dirbuster_final.py google.com

"""
```

```
# Import required packages

import os,webbrowser,sys,urllib2


# Print the input of the user on the screen

print "The URL/IP entered is "

print str(sys.argv[1])

print ("\n")

url=str(sys.argv[1])

files=['dir.txt']

url_list=[]

for f in files:

     hellow=open(os.path.join('C:\Users\harpreetsingh\Desktop\py_scripts', f))

     directories = hellow.readlines()

# iterate through the directory list and create a list with directories appended to the IP/URL and save it

     for i in directories:

          i=i.strip()

          url_list.append('http://'+url+i)


# Iterate through the items from the newly created list and check the response code
```

```
# In case the response code is 200, open the link in the browser

for url in url_list:

    print url

    try:

        connection = urllib2.urlopen(url)

        print connection.getcode()

        connection.close()

        if connection.getcode() == 200:

                webbrowser.open(url)

    except urllib2.HTTPError, e:

        print e.getcode()
```

 # Output

The *dir.txt* contains the below directories:

/jmx-console

/images

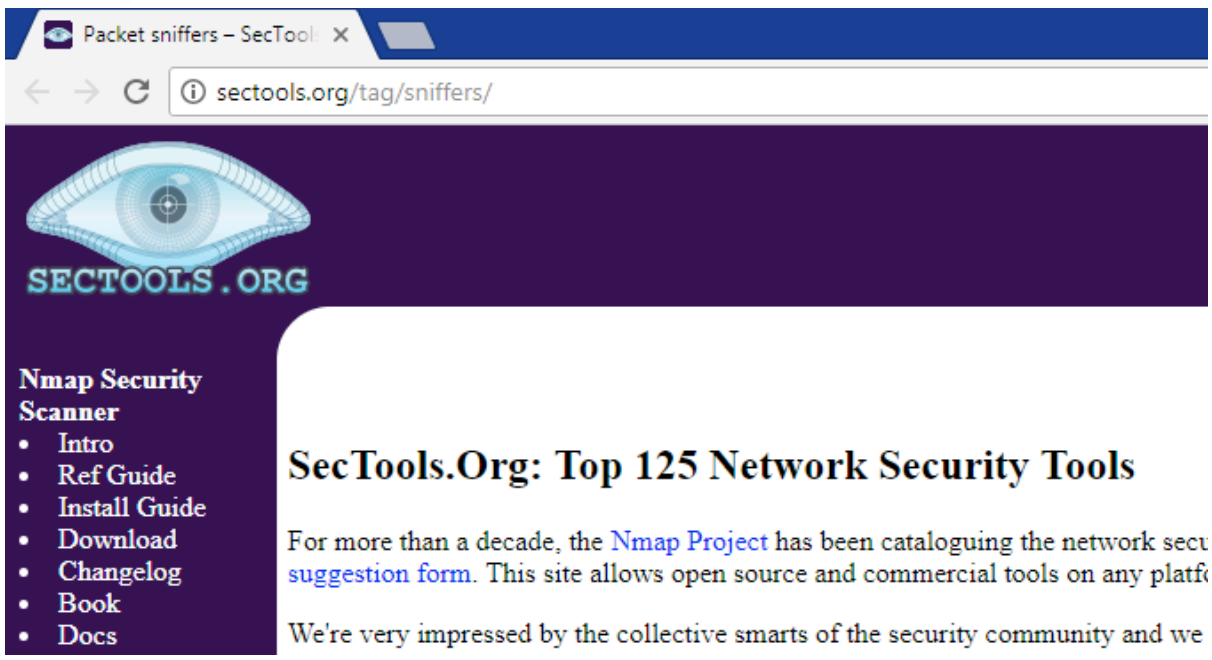
/audio

/php-my-admin

/tag/sniffers

 # Response codes on the output screen

```
C:\Users\harpreetsingh\Desktop\py_scripts>python dirbuster_final.py sectools.org

The URL/IP entered is
sectools.org


http://sectools.org/jmx-console
404
http://sectools.org/images
403
http://sectools.org/audio
404
http://sectools.org/php-my-admin
404
http://sectools.org/tag/sniffers/
200

C:\Users\harpreetsingh\Desktop\py_scripts>
```



The program can be used to check for the existence of a particular directory or a set of directories for a particular URL. I was once assigned a task to check if the jmx-console was opened for 160 public IP addresses. Manually checking this would have been boring and taken a lot of time. The list of directories can be downloaded from the internet or the same list as used by the dirbuster (tool in Kali) can be used.

Port scanning and Banner grabbing

The code discussed below will scan a particular host and port range to check for the status of the port. If the port is found open, a request will be sent so that the host responds to the request. The response is the banner which can then be analysed for the services running on that port. We will be performing a TCP connect scan for doing this.

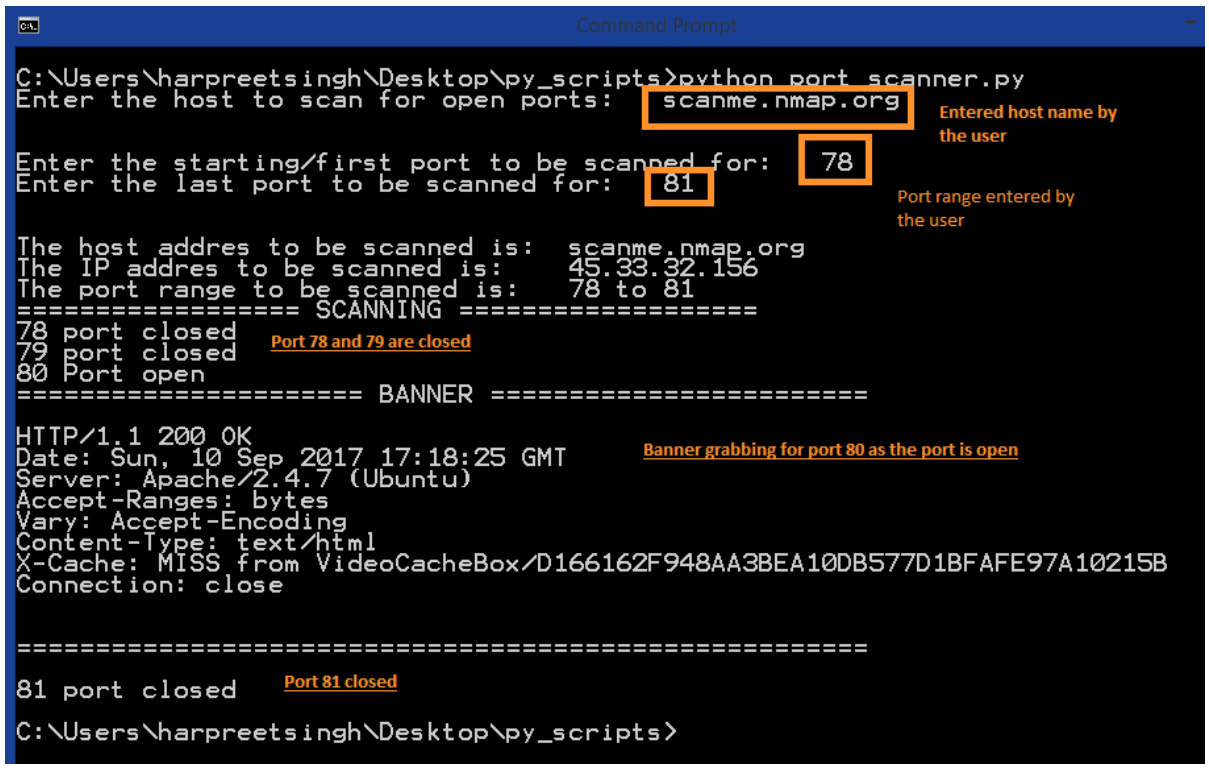
NOTE: *TCP connect scan can be easily detected and blocked since we are trying to create a connection with the server to be scanned.*

Code and comments

```
'''
Port Scanner and Banner grabber
'''
# Importing the required packages
import socket, sys


# Creating a TCP socket connection
conect = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
```

```
# Make the user enter the host address to be scanned

host=raw_input("Enter the host to scan for open ports:   ")

print "\n"


# The user can enter garbage value or fatfinger the host name while typing, check if this gets resolved for a valid IP address

try:

     IP=socket.gethostbyname(host)

except:

     print "%s --> Oops! Entered host cannot be resolved to an IP address" %host

     exit()


# Get the starting and ending of the ports to be scanned

first_port=raw_input("Enter the starting/first port to be scanned for:   ")

last_port=raw_input("Enter the last port to be scanned for:   ")


# The ports entered by the user need to be converted to integer for feeding this to the range function or else this will give an error.

int_first_port=int(first_port)

int_last_port=int(last_port)


# Print the information just in case the user wants to save the result to a text file

print "\n"

print "The host address to be scanned is:  %s" %host

print "The IP address to be scanned is:    %s" %IP

print "The port range to be scanned is:   %d to %d" %(int_first_port,int_last_port)

print "================== SCANNING ==================="
```

```
# Loop through the port range and check if we are able to create a successful connection

for port in range(int_first_port,int_last_port+1):

     try:

          conect.connect((IP,port))

          print "%s Port open" %port

          print "====================== BANNER ======================== \n"


# If we are able to create a connection, send a request.

          conect.send(b'GET /\n')


# Print the BANNER which we will receive back

          print(conect.recv(10000))

          print "======================================================== \n"

          conect.close()


# If we are not able to connect, the port is closed.

     except:

          print "%s port closed" %port

exit()
```



Usage and enhancements

The code can be used when we need to check quickly if a particular port is opened or not during a security testing. A website can be tested if port 80 is being opened, this can be a security risk as it is less secure. The code will not be of much help if the port range to be scanned is large. Thus we have the scope of enhancement.

The code can be enhanced to perform the following:

Take multiple hostnames from a text file and perform the scan. The user can start the scan and perform some other work or take a break for coffee.

Introduce timestamps: It will be useful if timestamp is recorded of when the scan started and ended. This can also be used to calculate the duration of the scan and this can be used as a parameter to speed up our tool as well.

Introduce threading: You will realise that the scan is fast when we have a small range of ports but this becomes slow when the range is large. Make the code threaded so that the speed will increase since multiple threads will be running in parallel. *(HINT: Threading module in Python will be of help)*

The Evolution of Penetration Testing

Professor John Walker CFIP FRSA

22 years in Royal Air Force Security/Investigations and Counter Intelligence operations [Overt/Covert] service, working alongside GCHQ, CESG, UK and US Agencies, ITSO and Systems Security Manager for CIA Accredited Systems, Visiting Professor School of Science/Technology - Nottingham Trent University [NTU], Advisory Board, Research Centre in Cyber Security (KirCCS) at University of Kent, Mentor to Tallinn University (Estonia) Masters Students Cyber Research, Practicing and Registered Expert Witness, Certified Forensics Investigator Practitioner [CFIP], Editorial Member at MedCrave Research for Forensics & Criminology, ENISA CEI Listed Expert, Editorial Member of the Cyber Security Research Institute [CRSI], Digital Forensics/Cyber Security Listed Trainer at Meirc [Dubai] of Certified courses, and Fellow of Royal Society for the Arts [FRSA], writer for Apress Publishing New York, and a Belkasoft (Digital Forensics) Partner.

In the current age of Cyber Security, I believe that the overall Penetration Testing activity must now move up a level to embrace the new era of subliminal isolated objects that, for many an organisation, can represent vulnerabilities in the form of unknown-unknowns that when aggregated and assessed, can lead to the discovery of other forms of usable intelligence, which may possibly lead to discovering other paths for usable exploitation – here I refer to the art of practicing effective OSINT (Open Source Intelligence) methodologies.

Introduction

In the beginning of the on-line, interconnected world, we encountered the early growth of security issues when it came to the operational perimeter. But then, in those early days of computing, the logical entity of the perimeter did not exist in the needy promiscuous form it does today. As the years progressed, we encountered an evolved era that has embraced the on-line world of interconnectivity at all levels; we encountered the genesis of network-to-network connectivity, and, of course, the evolution of that thing called the Internet. Thus, it was only natural to embrace the concept of security to follow the curve of risk, seeing the eventual creation of enhanced perimeter security being safeguarded with support from the then, *new* security devices such as Firewalls, IDS (Intruder Detection System) and IPS (Intruder Prevention System). As the interconnected era progressed, we then saw the march of computer viruses, and other forms of logical adversity, arrive at the personal and corporate logical front door to enhance the opportunity of Cyber Insecurity – from the Cascade to Joshi viruses, and from Stuxnet to the dangers posed by Ransomware, all bringing the challenges of infection and other forms of malicious payload. Driven by the early days of logical threats, to date, it was time to look to a mechanism to both discover security weaknesses, vulnerabilities, misconfigurations and other forms of insecurity, whilst at the same time confirming the security profile of the entity was of a robust profile - enter Penetration Testing.

The Evolution of Penetration Testing

It was around the 1990s when the concept of Penetration Testing started to really grab the attention of the CISO and IT Security Management's imagination as a means by which to fire, what was then considered by some to be, a magic silver bullet to assure the security profile of their interconnected world. It was also around this time when the then Network Associates rolled out their updated version of Net Recon, alongside the conversion from SATAN (Security Administrator Tool for Analyzing Networks) to SAINT (Security Administrator's Integrated Network Tool) which were utilised to run Security Assessments (*Not* necessarily Penetration Testing). However, to some extent, the expectations as to what such services and tools would deliver, at that time, did not always fully deliver against expectations. For example, in my time as the Head of a Security Team at Experian, I engaged the *specialised* services of a company I met at the London Infosecurity event. The company presented their *unique specialist* approach to delivering their bespoke Pen Test solution and were subsequently commissioned to carry out Penetration Testing of an Internet facing resource. The problem came when they delivered their *unique* and *specialised* report – a report that had been produced direct from the Network Associates Net Recon application, as produced by the tool with zero human involvement or interaction, other than pressing the key to generate the report!

The second example of *not* meeting expectations concerns a major brand bank based in Scotland. In this example, the bank's security management team committed significant financial support to commission a well-known service provider to conduct their year-on-year annual Penetration Testing. Again, and unfortunately for the client, the provider focus suffered from tunnel vision, and whilst the report gave the bank a clean bill of health, they had suffered *extant* security exposures at the core of their network for many years, where a wide-open SAMBA share had existed! Thus, the conclusion here is, whilst Penetration, or a convoluted Security Assessment, may have provided a clean, secure opinion, it nevertheless had failed to provide an expanded view, commensurate with the service provider's ability to produce a single vision of the truth, which sadly was based on their previous years of *known-known* scope, based upon multiples of previous tests which completely lacked any refreshment of their standing operational testing scope – in other words *same-old*, *same-old* year on year with the only change being made at the annual increase of their invoice.

There have also been occasions in some areas of the Utilities Sector where the results of a Penetration Test may have proved to be *awkward* to the end management expectation. For example, this is from some three or four years ago which relates to security testing of Smart Meters; in this case, having written the scope of work for the testing activity, the Penetration Test, and Security Assessments were run. However, the end report produced some *uncomfortable* reading for the owning corporate entity indicating some discovered, and unexpected, security issues. This knocked the company off the path required to meet government requirements for timely delivery into the public arena. In this case, there were *two* options under consideration. *One* of which was to address the located issues – but that could be a very long process, create a delay and cost money. Option number *two* was to rescope the testing activity and run it again with a much less rigorous testing scheme. You may have guessed by now which methodology was the choice of the management and executive – the scope was refreshed to factor out the *discomfort* of the previous report and run. As if by magic, a satisfactory report was produced, which moved their Smart Meter closer to delivery into the active supply chain – notwithstanding it was *not* fully secure.

On another occasion of a Penetration Test being performed at a UK Government site, the team involved managed to gain access to just about every critical system on the site in their first day on site – how? Like most good testers, they looked for some easy, low hanging fruit at the outset of the operation and discovered an on-line Password Vault which had been *secured* with a very simple password, leading to it being cracked and giving up all its 'secured' contents. In this case, whilst the test had discovered a simple vulnerability, it was sadly down to poor management, insecure configuration (this was but one of many), which was a known-known to the owner department, rendering the pen test activity in many ways worthless. It would have been far more beneficial if the lacklustre Security Manager could have invested some time and effort to ensure that his assets and infrastructure were secured to meet *best*, or even *acceptable,* level security practice, to present the testing team with a challenge, and not with a wide open sacrificial lamb to be led to the logical slaughterhouse.

So, given here, we have four good (bad) examples of what was considered the security product of a Penetration Testing activity. It may leave us with the considered opinion that the *trustworthiness* of any output from such an activity must be based on:

- The *Integrity* of the service supplier
- The *skill* of the service supplier
- The *ability* of the internal security team to set the scope, and assess and question the output
- *Preparedness* for the engagement
- The *integrity* of the commissioning body – i.e. is the activity to assure security, *or* one that only pays lip-service to a mandated expectation?

Moving Forward

In the current age of Cyber Security, I believe that the overall Penetration Testing activity must now move up a level to embrace the new era of subliminal isolated objects that, for many an organisation, can represent vulnerabilities in the form of *unknown-unknowns* that when aggregated and assessed, can lead to the discovery of other forms of *usable intelligence,* which may possibly lead to discovering other paths for usable exploitation – here I refer to the art of practicing effective OSINT (Open Source Intelligence) methodologies.

OSINT is one of those techniques that a few Penetration Testers have tended to discount and avoid, based on it being considered a Dark Art that does not conform to their technical world of focusing on all that is IP connected. However, there are many examples of how such a conjoined Pen Test to OSINT activity can and has produced very usable results. In one case, where a previous test had been conducted by a well-respected service provider who had not considered or utilised OSINT in their expensive Penetration Testing activity, saw the contracting client being successfully hacked within weeks of paying for the service – an attack believed to be based on an OSINT reconnaissance that had located a serious form of valuable, exploitable data leakage!

To put OSINT into definition context, and in simple terms, it is a methodology that seeks to discover:

Information/intelligence from a series of multi-format connected, and disconnected, sources, which may be discovered, acquired, and be subjective to further analysis to produce meaningful, usable intelligence

Examples of such OSINT discoveries are as follows:

- The Central London Bank who were not aware from their previous security testing engagements that a segment of their core infrastructure had been compromised to two .cn (Chinese) active connections
- The UK Government Agency who had released discoverable high value information to the Internet which exposed a *low-profile* government department outside of the protected square mile, who were previously an unknown sensitive target connected to the UK Security Services – e.g. MI6
- The Third-Party Developer site who were working on behalf of the US Security Services (the CIA and others) who had accidently left a DNS Zone Transfer open, allowing the discovery of internal servers, which at that time were appropriately named with the agencies identity (*there is another lesson to be learned here when it comes to good security practicies, but we will pass that over for another day*)
- And finally, the UK based Credit Reference Agency who again allowed a DNS Zone Transfer to be left insecure. In this case, any party discovering this hole could dig down to multiple internal servers, one of which allowed the download of a scripted object that was hard coded with the User ID and Password to connect to an internal server (as it happens, when this was reported to their large Security Department, their Main Technical Board, and their Technical Operational Security Manager, not only could they not discover the hole for themselves, but when they did, they

struggled to apply the easy security configuration – here the issues may be skills related, but again, we will leave that for another day!).

The value of OSINT discoveries it, the methodology looks for *subliminal* and *unknown* data that has found its way into the eye of the public, in the form of Metadata, Relational Data, and Data Objects which have not considered the potential security exposure. This approach may also be extrapolated to inspect, and acquire data from Social Media Sites, and other such public forums to paint a picture of the target – providing what may be referred to as a *Cuckoos-Egg* style of attack. But the most beneficial factor of an OSINT activity is that it can be run over extended periods, allowing the analysis phase to be conducted completely off-line.

Post Incident Analysis

In some cases, taking the Tesco Bank Breach as an example, it was possible to use OSINT techniques as a reverse incident process to support a leg of post-event Forensic Analysis to discover how a successful attack, and subsequent compromise, may have been played out in real-time. In the case of Tesco, it was possible to look at their annual reports to understand their thirst for cost cutting, and outsourcing operations. From that juncture, the analyst may then move on to identify the platforms that were outsourced onto, and then investigate what the security profile of the supporting systems may have been at the time, allowing on occasions an indication as to how a successful breach/compromise may have occurred.

Conclusion

Penetration Testing has now been with us for at least three decades and along its path, it has evolved in many forms, being delivered by both internal and external operatives, and is today, in the age of insecurity, a *must* have to underpin robust level of *informed* security. It is, however, time for the basis of the Penetration Testing operation to move into what I call the level of *Minority Reporting* state, encompassing OSINT that can give both the testing team, and the organisation the power to look to the *current* state of security, and into the *future* to predict those extant *unknown-unknown* security exposures to accommodate a level of a wide-focus informed position, with the potential to uncover what were, up to that juncture, operational chinks in the logical armour awaiting exploitation.

It may also be concluded that, whilst there is the opportunity to outsource such a security activity to a Third Party, it is nevertheless incumbent on the owner organisation to do their part in the Security Lifecycle of such a testing activity to ensure that *they assess*, *manage*, and *secure* their *own* assets to the best of their ability, both *pre*, and *post* such an activity being commissioned – remember, at the end of the day, anyone can outsource the owned security obligations, but the contracting organisation nevertheless still owns them!

Automated Source Code Review with Fortify SCA

Muruganandam Chandrasekaran

Muruganandam Chandrasekaran is working as a Principal Security Engineer at Oracle India Pvt Ltd. He has published many security articles in magazines and blogs.

Sumalatha Chinnaiyan

Sumalatha Chinnaiyan is working as a Security Lead in Acalvio Technologies Ltd. She is performing various security activities in machine learning technologies.

Fortify could translate Java bytecode, and this bytecode analysis can find flaws introduced due to compiler bugs. This is used as a first step when the FOSS has been determined potentially unsafe, to understand the magnitude and criticality of potential security flaws found by SCA in the software. This high-level analysis can be followed by source code analysis if necessary.

Introduction

Every single line of code should get reviewed and be made vulnerability-free code.

It is essential that we write quality code from day one to deliver high-quality products. We need to educate and encourage developers about quality by continuously monitoring/promoting throughout the development life cycle with various security methods.

We can make a reliable source code review process with automation using script and integration tools. This makes the source code review effortless and vulnerability free.

"All we need to know is how to use the available tools/methods and how best we can leverage, integrate and orchestrate them to create an end-to-end system in place which we can call the <<Automated Review System>>" (Aravind Kashyap, https://dzone.com/articles/code-quality-continuous-integration)

About Fortify Audit Workbench

Fortify Static Code Analyzer (more commonly known as Fortify SCA) is a static analyzer. This means it processes the source code as the data exists on disk, and predicts what vulnerabilities the program would suffer if it were to be compiled and run. The analyzer first parses the supported source code into an intermediate model (an analysis model) which is stored on disk in the form of .NST (normalized syntax tree) files. Then, a completed model is scanned during an analysis phase. During this phase, many different analyzers compare the patterns found in the NST with corresponding patterns in a corpus of security rules. Any matches are potential vulnerabilities printed into a Fortify Project Result ("FPR") file.

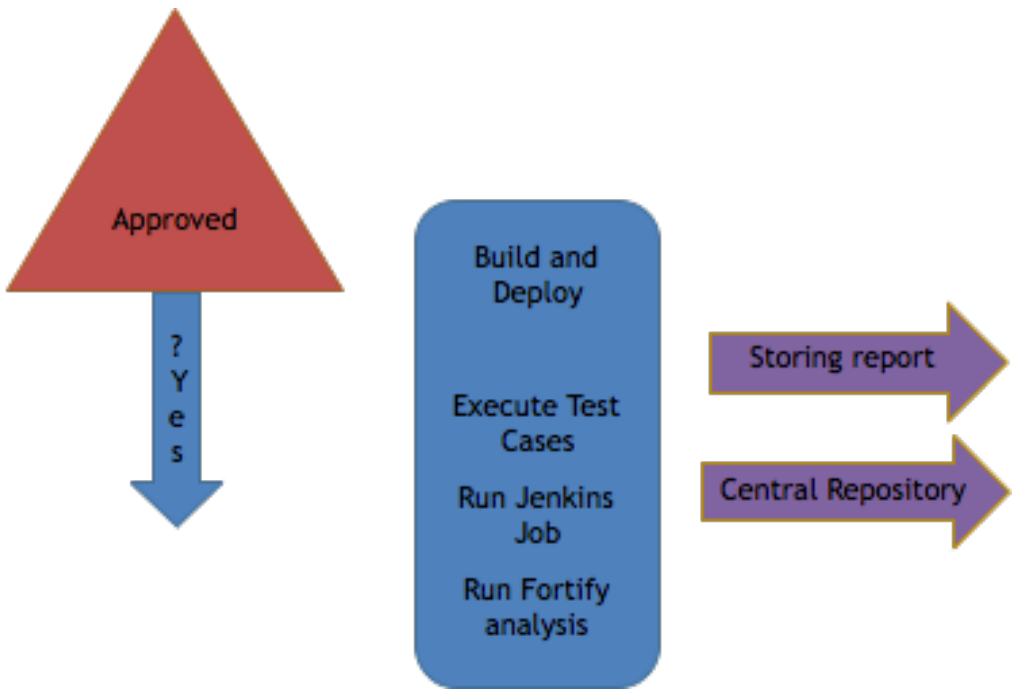
Fortify could translate Java bytecode, and this bytecode analysis can find flaws introduced due to compiler bugs. This is used as a first step when the FOSS has been determined potentially unsafe, to understand the magnitude and criticality of potential security flaws found by SCA in the software. This high-level analysis can be followed by source code analysis if necessary.

The various analyzers in Fortify SCA include:

- Semantic analyzer
- Structural analyzer
- Configuration (XPath expressions and key=value formatted text files) analyzer
- Control Flow analyzer
- Data Flow analyzer
- Buffer analyzer
- Many other, undocumented analyzers

Automation Process Life Cycle

So, there is a definite need for an automated review process to have a stable Review System.



Steps

Here are the steps that could help you build an automated system for code reviews:

- *"Choose a Repository (downloaded source code) to store the updated report for your system.*
- *Configure Jenkins job and set <<Build Trigger>> as per the project requirements. For example, Unit test executions can be a daily and performance/security can be weekly frequencies.*
- *Configure Git in Jenkins under <<Source Code Management>> to pull the code."* (Aravind Kashyap, https://dzone.com/articles/code-quality-continuous-integration)

This shell script is to invoke Fortify audit process against each pull.

```
export PATH=$PATH: $PATH

export WORKSPACE="$WORKSPACE"

THUMPER_LOC=$LOC

PROJECT_NAME=oasg-11

FPR_LOC=$PATH/ FPR_FILES

REPORT_LOC=$PATH/report_loc

TARGET_DIR="latest_fortify_reports"

for D in `find $WORKSPACE -maxdepth 1 -mindepth 1 -type d -not -path '*/\.*'`

do

sourcedir=`basename $D`

echo $sourcedir

cd ${WORKSPACE}/

if [ -d "$sourcedir.fpr" ]

then

rm -f $sourcedir.fpr

fi

sourceanalyzer -b $sourcedir -clean

sourceanalyzer -b $sourcedir -verbose $FPR_LOC/$D

sourceanalyzer -b $sourcedir -scan -f "$FPR_LOC/$sourcedir".fpr BIRTReportGenerator -
template "Developer Workbook" -format "HTML" -output ""$REPORT_LOC/
$sourcedir"_fortify_report_`date +%F`.html" -source "$FPR_LOC/$sourcedir".fpr --
IncludeDescOfKeyTerminology
```

```
if [ -e ""$REPORT_LOC/$sourcedir"_fortify_report_`date +%F`.html" ]; then

echo ""$REPORT_LOC/$sourcedir"_fortify_report_`date +%F`.html"

fi

scp -i /opt/HP_Fortify_SCA_and_Apps_4.40/fortifykey ""$REPORT_LOC/$sourcedir"_fortify_report_`date +%F`.html"
"droot@$THUMPER_LOC/$PROJECT_NAME/$TARGET_DIR"

done
```

$WORKSPACE - Working Directory for this source code repository.

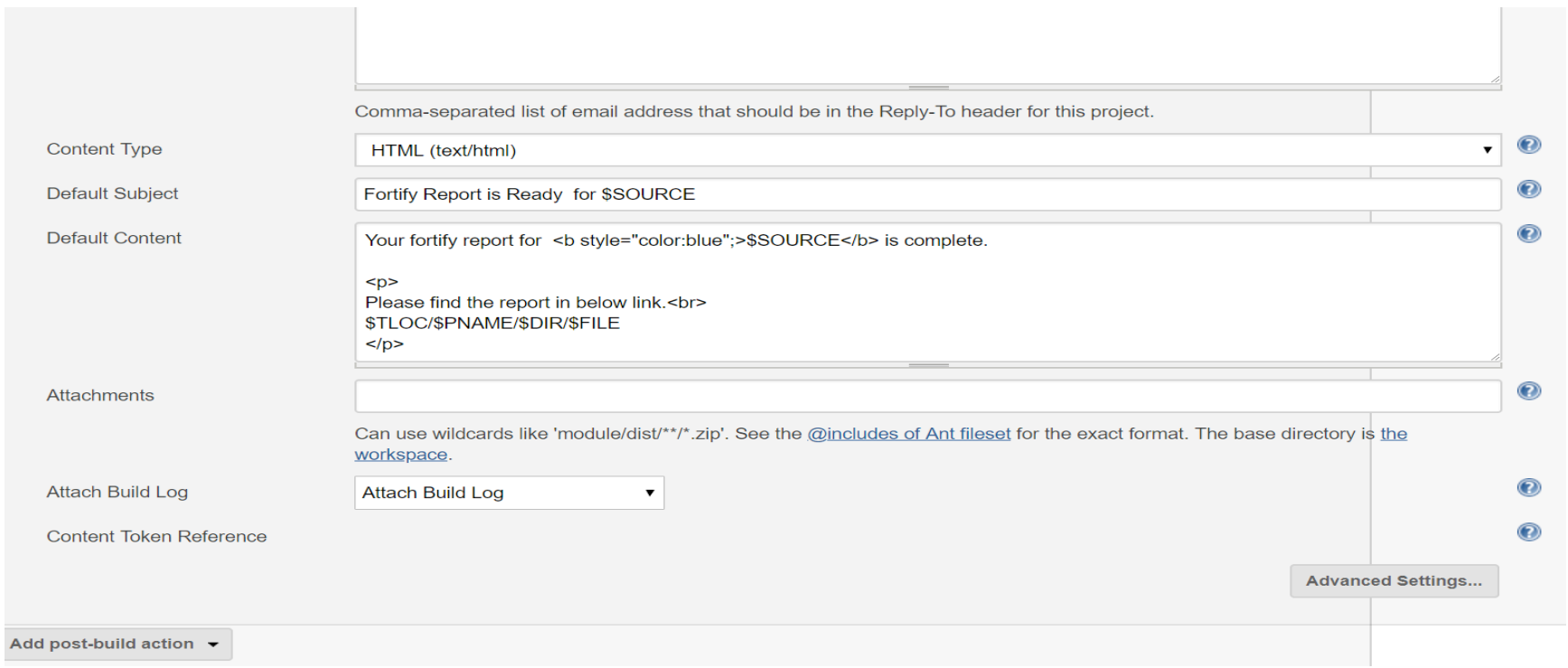
$REPORT_LOC – the Local repository for the newly generated Fortify reports.

$TARGET_DIR – Central Remote repository for the freshly made Fortify reports.

$SOURCE_DIR - Working Directory for the updated source code.

Configure "Editable Email Notification" to get the build result mails with build log attached and the test results published as part of mail subject with the module name using “Inject environment variables” value.

Mail Body is published with the module name and Report location as shown in the picture below.



On completion of the job, Jenkins sends the mail to the "Recipient List" and "Reply-To" List.

Conclusion

In conclusion, we can see that a review system with solid automated source code review will help with continuous tasks reviews, controlled check-ins, and share reports via mailing systems with all shareholders. This will reduce the security tested job on compiling, executing test suites, deployments on Jenkins, and this will further shorten QA cycles and avoid repeated manual tests comparatively.

References

Aravind Kashyap, https://dzone.com/articles/code-quality-continuous-integration

The Commoditization of Penetration Testing

Haydn Johnson

Haydn is an Information Security Specialist with over 5 years' experience of security program initiation and management, strategy, design, policy, operations, incident response, and implementation experience. He advocates Purple Teaming principles as a powerful methodology for improving intra-organizational security and relationships. Having recently moved to internal security, Haydn uses the offsec mindset to create impactful change within his organization. Committed to learning and sharing his skills, he has spoken at multiple conferences in America and Canada, and has published multiple online articles on security.

Haydn has a Masters in Information Technology, the OSCP and GXPN certifications.

With the commoditization of the Pentest, it seems all the data is copy-and-pasted in, with the only thing changing being the client name. I can understand from a business perspective the need to standardize the methodology and reporting to provide a better service and ensure a certain quality of Pentest – and all the various benefits that come with this. However, the standard of quality takes a gigantic hit in the process.

Introduction

The understanding of the Penetration Testing definition seems to be quite ambiguous among business clients. Different companies will provide different answers when they are asked about what they actually mean by using this and associated terms. The distinction between Vulnerability Assessments and Red Teaming, or PCI Penetration Testing is definitely not clear enough. In fact, Penetration Testing is a term misused and abused, ultimately watering down the work that involved people simply love to do. This article is going to discuss the trend to water pentesting down, why it was watered down and two approaches attempting to buck the trend.

The Commoditization of Penetration Testing

Let's start off with what Pentesting is sold as, or, more correctly, the commoditization of Penetration Testing. I have seen many reports that are titled a Penetration Test and a methodology that seems to be a penetration test. However, after looking deeply, it is merely a Vulnerability Assessment, or at worst just a scan. The idea of commoditization is to take something and make it easily repeatable, requiring low entry-level skills to complete the project and highest profit per test possible. So a Penetration test is watered down into its smallest form, requiring minimal technical skills, as much automation as possible, and the reporting is a standard format for each project. We have all seen this happen over time.

Big budget firms attempting to make money in 'cyber' security have high costs per man hour, each employee has to meet a quota of billable hours. As such, they have come into the market with five day pentests (three days testing and two days for reporting). This has resulted in an influx of low quality pentests. Penetration Testing by companies that commoditize it, become what is known as a 'pentest puppy mill', they reduce value adding tests to just a check box tick for management. The issue is that these companies forget the reality of what a Penetration Test is - a simulated attack. They forget that penetration tests are unique to each company based on their market, number of employees, political/ religious ties (if any). They also forget that Penetration Testing requires a high degree of knowledge in multiple disciplines and that each environment is different.

Commoditization of the Report

Penetration Tests are executed to help the customer's organization find vulnerabilities, understand the impact and remediate them. They are meant to help improve the security posture of the customer.

A blue team's view of a Penetration Test goes something like this:

- Organize scope of testing and sign off
- Testing begins
- Lots of alerts that are ignored
- Testing finishes
- Given a list of problems on your network and work to do

This is not helpful to a blue team. A report at the end is basically a slap in the face; here is everything you are doing wrong and work to fix it. Many commoditized penetration tests do this, it's static and easy to repeat.

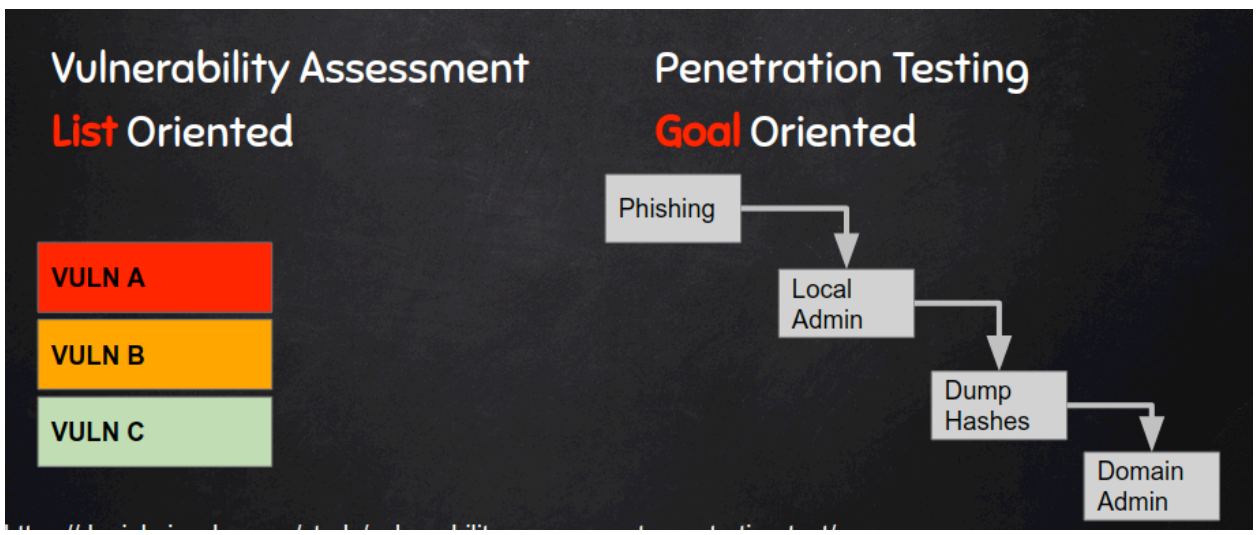
Why businesses commoditize

It is not a secret that businesses are in business to make money, specifically make a profit, as much profit as possible. As a result they try to commoditize as much as possible. Reducing a complicated task such as Penetration Testing allows businesses to hire cheaper resources because they are less knowledgeable. Many projects for penetration testing are based on the number of hours each employee will work to complete the project. The fewer hours it takes, due to 'simplifying' penetration testing, allows the company to save money, increasing profits. Repeatability allows more projects more often. Being able to use less knowledgeable employees means businesses have more employees to conduct commoditized Penetration Testing. Expensive, unique tools do not have to be created or bought. By commoditizing something as technical like Penetration Testing, businesses don't have to invest in research and development time for customized tools sets and or they don't have to spend money on advanced tools to help with their projects. This means businesses don't have to worry about as much lead up time or money to invest before receiving their money. Standardized reporting helps businesses to automate the reporting process. This can save businesses money by not having to create a completely new report each project. This makes sense, however, with the commoditization of the Pentest, it seems all the data is copy-and-pasted in, with the only thing changing being the client name. I can understand from a business perspective the need to standardize the methodology and reporting to provide a better service and ensure a certain quality of Pentest – and all the various benefits that come with this. However, the standard of quality takes a gigantic hit in the process.

So what is a 'Penetration Test'?

A quick Google search for 'what is a Penetration Test' brings a Wikipedia definition that is very succinct: "A penetration test, colloquially known as a pen test, is an authorized simulated attack on a computer system, performed to evaluate the security of the system." The key point being a simulated attack, this instantly means any type of scanning or

vulnerability assessment is NOT a Penetration Test. Sure these have their place in a Pentest, but are not, on their own, a Pentest.



The above image I made to create a visual demonstration of the difference between a vulnerability assessment and a most basic pentest. A vulnerability assessment is list oriented with ALL the possible risks found from the assessment listed in order of criticality. A pentest, for example, is goal oriented to show the impact of a successful simulated attack. In this image, it is a pentest that successfully gains Domain Admin access through phishing.

There are clear differences in the activities conducted. The Vulnerability Assessment is more of a scan and see what potential vulnerabilities there are, whereas the pentest has conducted a simulated attack and shown through each step how they have gained domain administrator access.

Pentest Methodologies

There are a few different methodologies for penetration testing. Some more well-known than others. My favorite is the Penetration Testing Execution Standard (http://www.pentest-standard.org/index.php/Main_Page), created in collaboration with the industries best pentesters/red teamers/attackers/tool makers.

PTES list the stages of a Penetration Test as:

- Pre-engagement Interactions
- Intelligence Gathering
- Threat Modeling
- Vulnerability Analysis
- Exploitation
- Post Exploitation
- Reporting

Even the list of tools from PTES shows that the skill level required in understanding the tools for a pentest is not trivial. Many people in the industry, such as Daniel Miessler, have commented on what a penetration test is. Daniel has a great article that can be found here: https://danielmiessler.com/study/vulnerability-assessment-penetration-test/ The idea is

that the Pentest methodology should have steps after a 'vulnerability' analysis, which then begins more of the active part of a simulated attack. If not, then it is most likely not a penetration test.

Attempting to buck the trend of commoditization

As a result, the various understandings of a Pentest and the difference in quality and costings, some approaches have come out to buck the trend. The easiest way has been to name the testing something completely different from Penetration testing with a different model, whereas another is still called penetration testing but provides much more value in a clearer way.

Bug Bounty Programs

Bug bounty programs are aimed at web applications. They generally have experienced pentesters known as 'bug hunters' attack the application to find vulnerabilities also known as 'bugs'. Bug bounty programs differ from Penetration Tests in two big ways:

- Bug hunters are only paid when they are the first to find a vulnerability.
- Bug hunters are what's known as crowdsourced, they are individual security researchers. They are not a team from a pentest company.

There are other items that differentiate from a pentest, such as a longer time frame is given for a bug bounty. (three months+).

Iterative Modern Day Penetration Testing

If you know Chris Nickerson and his company Lares Consulting, or have seen any of his talks, you will have heard about iterative Penetration Testing. Iterative Penetration Testing is defined by:

- **Incrementing the testing difficulty**

The aim here is to start an attack, such as network scanning, in its most detectable form (Nmap with no flags) and if that is detected slowly increase the technique. So in this case, assuming a standard Nmap scan was detected, a pentester would then conduct a fragmented Nmap scan in an attempt to not be detected. As soon as the attack is not detected the finding is written up and sent to the client, hoping the client can fix/improve and more advanced testing can be done again.

- **Providing results as they are found. This allows a company to fix findings in a staggered way instead of being dump results at the very end.**

Chris gave an amazing talk at HackFest that explains iterative pentesting and the value. The talk is mostly on how pentesters and red teams should and can provide real value to blue teamers. The talk is called "Adversarial Simulation: Why your defenders are the Fighter Pilots" and can be found here (there is swearing).

Notable 3rd option

Vetted Crowd Sourced Iterative Pentesting

This is a combination of bug bounty and iterative Penetration Testing ideology. Quality pentesters are vetted from around the world similar to a Bug Bounty platform. Instead of being paid per finding they are paid for their time and effort. The pentesters do this as a hobby or second job and are vetted for quality on each pentest. As a result, they provide much better quality than a standard Penetration Test. The iterative Penetration Testing comes in the form of being available via live chat during the testing, so companies paying for the service can understand the thought process of the testers, ask

questions and what not. Additionally findings are reported as they are found. One company that offers something like this is Cobalt, known as *Pentesting-as-a-service*.

Fixing that "Report"

As a security practitioner working with the IT operations team and developers, I have learned how their processes work. Continuous integration and deployment are actual things that help companies produce quality code and applications. They use ticketing systems, scrum and sprints to complete work. A new welcomed trend from Penetration Testing is connecting seamlessly into the current way teams work. This started with iterative testing, providing the findings as they are found. The biggest game changer I have seen is the 'ticketing' system that connects to the customers 'ticketing system'. The pentesters act like internal employees in that they create the findings in a system that emulates software development, such as planning, tracking, and managing a project. This allows a 'security' issue to become like any other software update; the developers open the ticket and follow their normal process. It's a fantastic improvement. This also works the same with network findings. The operations team is always having fires to deal with and closing 'tickets'. If a patch needs to be applied, it no longer comes across as a separate 'security' finding that needs to be resolved. It joins the queue (based on criticality) of an operations team member's normal workflow.

The close out

Penetration Testing has and will continue to be commoditized. Commoditization of the Penetration Test is not helping companies improve their security posture. There are, however, a few different approaches that are attempting to bring value back into the Penetration Test and this is extremely important.

If you are a Penetration Tester, or looking to become a Penetration Tester, take these points and provide real value to the world. If you are a company looking for Penetration Testing take these points to evaluate your vendors and ensure you get the best bang for your buck.

References

- **Wikipedia Penetration Testing https://en.wikipedia.org/wiki/Penetration_test**
- **Daniel Miessler https://danielmiessler.com/**
- **Vulnerability Assessment and Penetration Test definition https://danielmiessler.com/study/vulnerability-assessment-penetration-test/**
- **Cobalt: https://cobalt.io/**
- **Lares Consulting http://laresconsulting.com/**
- **Adversarial Simulation: Why your defenders are the Fighter Pilots https://www.youtube.com/watch?v=flmxbKflAE4**

How Does Python Affect Pentesters?

Nouha Ben

Nouha Ben is 15-years-old Computer Science student. She is a Junior Python Coder, interested in Robotics, AI and High Tech. Nouha codes in multiple languages - C++, JavaScript+, HTML+CSS, Python, SQL, Java. With nearly 1,5-year-experience she has recently worked on AI project and has a good level of knowledge on Neural Networks, AI, Machine Learning, Cyber Security, Ethical Hacking and General Computer science.

Nouha's social media profiles:

- Twitter: https://twitter.com/QueenOfCode3?lang=en
- Google+: https://plus.google.com/u/0/+thequeeneagle
- YouTube Channel: The Queen Of Eagles

The article covers the most useful information about Python, especially why it is worth using it as a pentester, white hat or IT professional in general. It presents a few of the most used Python libraries recommended by the author. There is also a basic approach to web scraping using Python and requests and a quick guide for Urlib. Moreover, she covers the basic usage of Scapy including simple code examples, and scanning ports and networks using Libnmap.

Introduction

In this article, I'm going to talk about how Python programming language can affect your job as a Pentester or a White Hat hacker. I'm going to explain why you should use Python in your work.

What you will need: All you need is to be familiar with some programming terms. It will be also great if you have some experience with any programming language such as Python.

What you will learn: This article will familiarize you with Python's programming terms and you will have a good knowledge of Python programming and you will move on from Python basics to more advanced concepts and discover more about Python's world.

Why Python?

Python is a general purpose, interpreter programming language. It is object oriented. It also supports limited functional programming but multiple programming paradigms including OOP, imperative and functional, Python is very popular nowadays. It has multiple uses. Python has a very large and comprehensive standard library but Python is best known for its simplicity and straightforward syntax. It is also used in multiple domains, such as robotics, machine learning, big

data, hacking, AI, etc., but you might not use it in Application Developments because Python cannot be as fast as fully-compiled programming languages like C++, for example.

There are a lot of big famous companies nowadays that are using Python in their projects. I will name a few:

- Google.
- YouTube, the most popular video sharing service.
- Dropbox.
- NSA uses Python for cryptography and intelligence analysis.
- NASA.
- Netflix.
- IBM uses Python for hardware testing.

Python for White Hat hackers/Pentesters?

As we saw in the list above, Python is used widely, but you're asking yourself how Python would affect your job as a Pentester/Coder/Hacker? It's relatively easy, compared to other languages and it's supported by all major platforms: Linux, Mac OS X, and Windows. It has a deep set of native libraries that, in turn, reduces the number of lines of code required. It has a highly active development community but it also includes tools and libraries for hacking and penetration testing.

Are there any Python libraries for pentesters?

The answer is yes. I'll name the top three most useful Python libraries with an explanation of what each of them does. These libraries are the most used libraries by Pentesters who code in Python. For sure there are more than three but these are my favorites, so here are my top three libraries for Pentesters:

Mona: Mona.py is a plugin for Immunity Debugger, which is developed by Corelan Team. Corelan Team is a group of IT Security researchers/enthusiasts/professionals/hobbyists who share the same interest.

Python Nmap: Makes it easier to programmatically parse Nmap scan results and every Pentester uses Nmap. Python-Nmap provides an easy method to analyze scan results, and execute custom attacks against specific hosts.

Impact: Basically, impact is a group or collection of Python classes for working with network protocols and provides low-level programmatic access to the packets.

Bonus: Urllib; we will work with it later.

Web Scraping with Python and requests

If you already have a background of HTML and Networking, that would be great. With Python, you can do web scraping. There are actually a bunch of Python modules used for web scraping, like beautifulsoup4 and requests or HTTPLib. Now I'm going to show you a simple, quick example of using requests in Python.

We're going to write a simple code that allows us to log into our website or any website you want to log into just by using Python and requests Urllib is very useful but it would be great if you use HTTPLib too, they're both great.

First you need to understand how data is handled at the HTML page level.

The login prompt on a web page is just a simple HTML form. For example, when you enter your data (e.g. username, email address, etc.) and you click submit, you're sending your data to the authentication application behind the page. This is called a POST. You're pushing, or *Posting* your data.

Example Code:

```
1.  import requests
2.  from requests import *
3.  Post-Login-URL = https://my.website_name.com/login'
4.  REQUEST-url='https://my.website_name.com/home/posts'
5.  load = {
6.      'username': 'your_username',
7.      'password': 'your_password'
8.  }
9.  with requests.Session() as session:
10.    post = session.post(Post-Login-URL, data=payload)
11.     r = session.get(REQUEST-URL)
12.     print(r.text)
```

Getting Started with Urllib

When it comes to "Fetching Data" across the web or the World Wide Web, Urllib is well known and used widely; in that subject, Urllib and HTTPLib are the most useful in web scraping but for now I'm going to talk about Urllib.

If you already heard of this module before you might know that the Urllib module has changed. What do I mean by that actually? Well, to be more clear, the Urllib module has been split into parts or small modules and it has been renamed in Python 3; there is Urllib.request , Urllib.parse and Urllib.error. The Urllib.request.urlopen() function in Python 3 is the same urllib2.urlopen() and the urllib.urlopen has been removed. So this is important, that's why I mentioned it because you might be working with Python 2 and then you might change and work with Python 3 so this is a really good thing to notice because you know the two major versions of Python are not the same and the new version of Python supports newer features.

Note: *For Python versions earlier than 2.7.9, Urllib does not attempt to validate the server certificates of HTTPS URIs. Use at your own risk!*

You can always read the documentations and read about that more in the Urllib docs.

Working with Urllib

So this module provides a high-level interface for fetching data across the World Wide Web and it has a bunch of useful functions that we're going to learn about.

Urllib is a native Python module/library so you don't need to install it using pip.

For more documentations check out:

https://docs.python.org/2/library/urllib.html

Now I'll show you a quick example, for sure you have heard of *GET* and *POST* methods and for now we are going to use Urllib and use the GET method to retrieve a URL containing parameters:

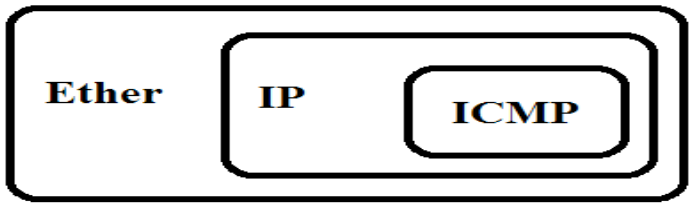
```
1. >>> import urllib
2. >>> params = urllib.urlencode({'spam': 1, 'fizz': 2, 'eggs': 0})
3. >>> f = urllib.urlopen("http://www.musi-cal.com/cgi-bin/query?%s" % params)
4. >>> print f.read()
```

Find this code and examples at https://docs.python.org/2/library/urllib.html

Using Python and Scapy for packet sniffing

There is one more Python module that I really prefer and it's the most used by pentesters. Scapy is a powerful interactive packet manipulation program. It is able to forge or decode packets of a wide number of protocols, send them on the wire, capture them, match requests and replies, and much more. Scapy can also easily handle most classic tasks, like scanning, trace routing, probing, unit tests, attacks or network discovery. It can replace hping, arpspoof, arp-sk, arping, p0f and even some parts of Nmap, tcpdump, and tshark.

Scapy uses Python dictionaries as the data structure for packets. Each packet is a collection of multiple or nested dictionaries with each layer being a child dictionary of the previous layer, built from the lowest layer up. If we visualize the nested packet layers would look something like this:



Getting started with Scapy:

Now that we have a good understanding about Scapy, we can take an example about what you can do with Scapy. Now I'm going to show you an example code.

First of all, in order for your program/script to format and return the packet info as you wish, the `sniff ()` function passes the packet object as the one and only argument into the function. We're going to perform a simple custom action with each sniffed packet.

Here's an example code that uses Scapy. What does this code actually do? This code is an example of keeping track of the number of packets sniffed.

Example code:

```
1. import __future__
2. from  scapy.all import *
3.
4. # Creating a Packet Counter
5. P_counter = 0
6.
7. #Define our Custom Action function
8. def  custom_action(packet):
9.     global  P_counter
10.     P_counter += 1
11.     return 'Packet #{}: {} ==> {}'.format(P_counter, packet[0][1].src,
12.         packet[0][1].dst)
13.# Setup sniff, filtering for IP traffic
14.sniff(filter="ip", prn=custom_action)
```

For more documentation about Scapy, I recommend checking out the Scapy Docs on: https://media.readthedocs.org/pdf/scapy/latest/scapy.pdf

Here is one more example on creating a simple Custom Formatted ARP Monitor. ARP stands for Address Resolution Protocol. It scans for active devices on the local network segment but we're going to use the same `prn()` Function that we've seen above in the first example code.

Example code:

```
1. Import __future__
2. from  scapy.all  import *
3. def  monitor_display(pkt):
4.     if  pkt[ARP].op == 1:
5.         return 'Request: {} is asking about {}'.format(pkt[ARP].psrc, pkt[ARP].pdst)
6.     if  pkt[ARP].op == 2:
7.         return '*Response: {} has address {}'.format(pkt[ARP].hwsrc, pkt[ARP].psrc)
8. sniff(prn=arp_display, filter="arp", store=0, count=10)
```

Creating another sample program that is similar to the previous example code:

```
1. import logging
2. import scapy
3. Logg=logging.getLogger("scapy.runtime").setLevel(logging.ERROR)
4. from scapy.all import *
5. dst_ip = " 192.168.56.1"
6. src_port = RandShort()
7. dst_port=80
8.
9. tcp_connect_scan_resp = sr1(IP(dst=dst_ip)/
   TCP(sport=src_port,dport=dst_port,flags="S"),timeout=10)
10.if(str(type(tcp_connect_scan_resp))=="<type 'NoneType'>"):
11.    print "Closed"
12.elif(tcp_connect_scan_resp.haslayer(TCP)):
13.    if(tcp_connect_scan_resp.getlayer(TCP).flags == 0x12):
14.send_rst = sr(IP(dst=dst_ip)/
   TCP(sport=src_port,dport=dst_port,flags="AR"),timeout=10)
15.        print "Open"
16.elif (tcp_connect_scan_resp.getlayer(TCP).flags == 0x14):
17.    print "Closed"
```

Output:

```
1. Begin emission:
2. Finished sending 1 packets.
3. Received 0 packets, got 0 answers, remaining 1 packets
4. Closed
```

Scanning Ports/Networks using Libnmap

So now I'm going to talk about Libnmap but there a few things you should keep in mind before getting started. I'll not show you example codes on using Libnmap but for sure you will get familiarized well with Libnmap and how it works. Actually, you'll be able to know where to start.

About Libnmap:

Libnmap is a Python toolkit for manipulating Nmap. It currently offers the following modules:

- *Process:* enables you to launch Nmap scans
- *Parse:* enables you to parse Nmap reports or scan results (only XML so far) from a file, a string...
- *Report:* enables you to manipulate a parsed scan result and de/serialize scan results in a json format

What is Port Scanning or Networks Scanning?

It's a process that sends client requests to a range of server port addresses on a host, with the goal of finding an active port. This is one of the most popular techniques in hacking or pentesting. The attackers uses that technique to discover services that they can exploit to break into systems. By that, the attackers can find information about the target system; for example, what services are running or the users who own those services. That process is very important to network security technicians and one of the most important things that you should learn how to protect yourself or how to defend against attackers.

Types of Scanning:

1. Address Resolution Protocol (ARP that we have seen earlier)

2. The Vanilla TCP connect

3. The TCP SYN (Half Open)

4. The TCP FIN

5. The TCP Reverse Indent

6. The TCP XMAS

7. The TCP NULL

And much more.

Is port scanning legal?

This is the most important part - here you should be careful. Port scanning is legal but it can be illegal; it depends on your case.

Scanning other's systems would get you in trouble. Also, scanning without permission leads to trouble, too, but you can always protect yourself and avoid getting in trouble if you scan a port or a system that isn't illegal. Port scans are illegal only if you use the information from a port scan to exploit a vulnerability or open a port on the system.

But you should always pay attention before doing anything that needs you to ask for permission first; you won't get in trouble for sure.

How to protect yourself?

Another way to protect yourself is to minimize the impact of your scans on the client's environment.

- Ask for permission before scanning
- For students or someone who is training, you can scan your IP address on your machine or set up a virtual machine
- Do not scan IP ranges beyond those for which you need information
- Do not perform vulnerability scans when a simple ping scan will do

References:

I recommend visiting the following websites for more information about ethical hacking and penetration testing and more documentation:

https://www.cybrary.it/course/ethical-hacking/

http://libnmap.readthedocs.io/en/latest/

https://www.owasp.org/index.php/Main_Page

https://www.calyptix.com/top-threats/port-scanning-legal-answers-companies/

https://thepacketgeek.com/scapy-sniffing-with-custom-actions-part-1/

https://www.sans.org/reading-room/whitepapers/auditing/port-scanning-techniques-defense-70

Automation In Penetration Testing Nowadays Is As Important As RAM In Any System

An Interview with Falgun Rathod

Falgun Rathod

Mr. Falgun Rathod, Managing Director of the Cyber Octet Private Limited based in Ahmedabad, has nearly one decade of experience in Industry. Mr. Rathod is specialized in Cyber Crime Investigation & Infrastructure Security. He is renowned Cyber Security Consultant and listed in Top ten Ethical Hackers of India & Top Ten Cyber Cops of India. Mr. Rathod had delivered 100+ Seminars & Trainings in various Colleges and Corporates across India. He has been known for his distinguished contribution to the society for cyber security awareness. He assisted State & Central Government for many projects & cases.

Automation in this service will minimize the manpower and the companies will stop outsourcing for this kind of project. Automation of penetration testing will just have a simple role as any other tool or service provider company will have. They will understand the infrastructure or SOW or deploying the tool in your environment. They will configure it for you and train your in-house team to use such tools. Hence, there will be a huge impact on the outsourcing of penetration testing, especially for manual testing.

[PenTest Magazine]: Some time ago, penetration testing was said to be in a definitional crisis. The difference between the terms "penetration testing" and "vulnerability assessment" appeared to be not clear enough, not only amongst customers but also within the pentesting environment itself. How does the situation look currently and what are your reflections upon this matter?

[Falgun Rathod]**:** Yeah, there has been lots of conflict between VA (Vulnerability Assessment) & PT (Penetration Testing), as you said, but awareness always brings the change and so did this. Most companies and technical people are able to differentiate between these two buzzwords. I had even published an article on Slideshare where the difference and whole framework for conducting VA and PT is given.

Still there are a few upcommers in the industry as well as companies from another domain that are not aware of this. But time and experience are the best teachers.

[PT]: What is the role of automation in penetration testing nowadays and how do you evaluate this trend?

[FR]: Automation in Penetration Testing nowadays is as important as RAM in any system. An automation system will try to design architecture that will automate all the tasks of the penetration tester. It will give the real time scenario to interpret the automation systems for data formatting, implement proof of concept as per the objectives and scope. There is a high cost of time, money and manpower for manual penetration testing so bringing automation to the industry will be beneficial.

[PT]: Which institutional and business areas do you consider to be getting the most benefits from automation in pentesting?

[FR]**:** Most of the industries that are storing and processing important business data are getting the most benefit from the automation in penetration testing. AI based automation would be simplifying the lives of testers/developers and administrators.

[PT]: Could you please indicate the most important advantages of automation in penetration testing according to your own experience?

[FR]**:** Industries are now getting reliant on complex IT infrastructure, hence most of the services and protocols are being used and made accessible that leverage the attacker to deploy malware or perform an attack. As discussed, to reduce the time of the manual and testing, more manpower needs to be implemented. There are automated tools available but by creating an automation system in identifying the vulnerabilities and creating proof of concept without false positive results will get penetration testers and analysts to think out of the box and minimize the timings. Even the reports generated would be much easier to understand for the developers and administrator or the management.

[PT]: Speaking about matters from a purely technical perspective, what do you consider to be the main drawbacks of the automation?

[FR]**:** There are various drawbacks. Every new technology has pros and cons; some of them lack security configuration management and skill sets of the analyst or tester using the automated tool, vulnerabilities in the tool itself, business logical testing issues, as my answer in the previous question, false positive results comes as advantage and disadvantage as well, less intelligent analysis of the sensitive data, stability issues in different environments and there can be various other drawbacks of the automation depending upon the scope of work but meanwhile we can consider above mentioned points.

[PT]: Let's turn our conversation a bit more into the business aspect. Would you agree with a statement that commoditization of pentesting, for instance, offering packages, can potentially cause a lack of confidence between business partners, or evoke some problems in communication?

[FR]**:** Yes, I have seen there are various tools in the industry that are open source and widely used by the professionals, meanwhile there are companies who offer the tools of penetration testing at a huge cost, even though many tools do not perform and few are very good. I would not be taking name of any but there are various things that could be added as features or taken as unique selling points and there are various tools at the same time that say they provide such features and good results but actually they are not, which misleads the customer and the whole industry and trust level decreases from using such tools or the company providing services.

[PT]: Is it true that more companies tend to choose price instead of security?

[FR]**:** Yes. It’s a serious issue. Nowadays, there are several companies providing services in penetration testing and a few of them are charging very high amounts (10,000-20000 USD), which seems to looting the customers, as this is a somewhat new industry and management is not aware about this about the criticality of the data and penetration testing. Then there are few companies, or you can say freelancers, that are charging a very low cost (250-500 USD). And companies will be going for conducting security testing from the companies offering lowest price; it is as simple as compared to compromising security of the system intentionally.

[PT]: Could you explain the correlation between the increasing role of automation and outsourcing of penetration testing? Is there any? If so, what is the character of this relation?

[FR]: Automation in this service will minimize the manpower and the companies will stop outsourcing for this kind of project. Automation of penetration testing will just have a simple role as any other tool or service provider company will have. They will understand the infrastructure or SOW or deploying the tool in your environment. They will configure it for you and train your in-house team to use such tools. Hence, there will be a huge impact on the outsourcing of penetration testing, especially for manual testing.

[PT]: According to common opinion, the increase of automation has lead to the spread of 'Pentest Puppy Mills'. What are your observations on this matter? Is there any way out of this professional cul-de-sac on the job market?

[FR]: As mentioned in the last answer, everything is correlated with each other. There are so called startups and freelancers or ‘pentest puppy mills’ that are in the industry that just run the tool, take the report, format it as per their document and deliver it to you.

This is something serious that management should try to understand, learn and then evaluate on various factors like clientele, skills of resource person, profile of the company, rates, scope of work and a few others, and then finalize.

[PT]: The dominance of automation seems to diminish the role of the promising exploit developers, bug hunters and researchers. Would you agree that increasing level of the automation in penetration testing might impoverish infosecurity talent landscape?

[FR]: No, I don’t think so. This is kind of two sides of the coin. There are changes periodically in technology, hence there would be vulnerabilities, so we will require good exploit developers, bug hunters and researchers at the same time to make us understand the bugs and where the products are lagging behind. Yes, once it is found, then it will impact the roles of such talented guys in near future. Hence, as it is said, there will always be pros and cons of new technology. New talent or researchers get better then think of something out of the concept as the attackers are never going to stop themselves so researchers can’t stop either.

[PT]: What are your predictions for the future of penetration testing? Do you think that the role of automation will gradually increase and if so, where is the potential climax of this situation? Perhaps the individual approach to penetration testing will then become more valuable?

[FR]: It will not be a climax but the automation/AI in penetration testing is the future, undoubtedly. The next five years are going to be the years for the new technology in the industry to be evolved and then the next five years to stabilize it. So this is the start where one can boost himself into learning AI or automation in penetration testing. Still, it’s got a broad scope. Every individual from IT should have this skill set to understand the vulnerabilities, understand the impact of the attacks, identifying risk and threats in their Infrastructure. There have been standards proposed for automation in PT. So, now we just need to stay calm and wait for the right time and see how the things will get change and what it will actually impact.

Adopting Automated Pentest Within Your Company

Zinedine Boudegna

Zinedine is an architect consultant in CyberSecurity. He is managing several information security projects in his careers in order to align security objectives with business objectives, with the speed necessary in terms of CyberSecurity growth. Zinedine offers his expertise as a consultant for the various companies internationally in the following areas: Security Compliance, Security Operations, Security & Risks Management, Security Assesement & Testing, Identity & Access Managment, User Education.

Setting up a vulnerability management system will require the intervention of several stakeholders, whether in the security team or in the other operational teams. It is important to document the stakeholders, and to define their roles and responsibilities in relation to the program. It is recommended to define OLAs with internal teams to minimize the waiting times for exposure of the risks found.

Introduction

The importance of pentests for organizations

More and more standards and laws, such as PCI DSS, ISO 27001 GDPR (Article 32), require organizations to perform intrusion tests at regular intervals, to assess the robustness of security systems and to enable continuous improvement of their security.

By putting in place regular pentests, organizations benefit from:

1. Avoiding losses in down time systems

An attack happens at any time without warning, and usually when you least expect it, sometimes the cyberattacks target one safety pin (Confidentiality, Integrity, Availability) by targeting the availability of information via DDOS; for example, large losses may be directly related to the latter, however, higher losses can be incurred for the restarting of the systems.

2. Meeting regulatory requirements and avoiding fines

Detailed reports that slopes will manage, can avoid significant fines for corporate non-compliance and allow companies to demonstrate due diligence to evaluators by maintaining required audit controls.

3. Preserving the corporate image and building customer loyalty

Security incidents do not go without leaving a significant impact and a negative term on the sales and the public image of an organization. Penetration tests help organizations avoid incidents by simulating them in advance.

4. Helping evaluate investment in terms of cyber security

Decision-makers need accurate indicators and mapping to illustrate weaknesses in their infrastructure and security systems, intrusion tests help to provide this view, highlighting priorities investment, which is decided in relation to the criticality of the vulnerabilities identified by crossing it with the criticality of the systems in which it is located.

5. Managing risks correctly

One of the most popular benefits of intrusion testing is to provide a structured and optimal basis for risk handling, as the slopes will show the list of vulnerabilities in the target environment as well as the associated risks. This information will provide concrete indicators for better management of cyber risks.

Philosophy of Pentest

The goal of a pentest is to put yourself in the shoes of a hacker, therefore, following his approach is essential to bring out real risks on the target infrastructure, regardless of the type of pentest, he is subject to steps defined from the recognition of the target to the exploitation of vulnerabilities and the presentation of the reports. Here we will present the steps as defined in the penetration testing standard execution (PTES), which consists of seven main sections that cover, in terms of pentest:

- Pre-engagement Interactions
- Intelligence Gathering
- Threat Modeling
- Vulnerability Analysis
- Exploitation
- Post Exploitation
- Reporting

Manual Pentest

Whether pentests are manual or automated, they are subject to the same philosophy. In the manual pentest, the company must call on a security professional specializing in the security of the systems it wants to test, so it is manual work that must rely on the skills of the engineer and its creativity.

Benefits and limitations of manual pentest:

Advantages

- *Following the business logic*

Automated tools are weak when it comes to testing for business specific vulnerabilities. Manual efforts are required to perform specific business tests.

- *Updating the knowledge base and zero days*

When a new vulnerability/exploit is released, most automated tools must wait until the next update to be able to use it in their tests, while a human can learn a new technique and implement it the next day. However, this again requires a qualified expert.

- *False positives*

With automated tools, the false positive rate is considered high. A manual scan is required to confirm the reported vulnerabilities.

[Source: https://resources.infosecinstitute.com/automated-tools-vs-a-manual-approach/]

Limits of the manual pentest

- *Variable results*

The results will be different from one tester to another, even if the target is the same, which may not give the real state of the weaknesses of the target infrastructure.

- *Time consumption*

The manual penetration test is a long and complex process, which takes a long time, knowing that time is a very important factor when dealing with cyberattacks. In a manual penetration testing, the engineer needs to write his exploits by adapting to the situation with which it is associated, a great team is usually required for testing large sizes.

- *Must be done by someone skilled*

Generally manual slopes are complex processes that require highly qualified and experienced people, who are relatively rare and expensive.

- *Remembering all the details*

The engineer needs to memorize all the steps that required the success of his manual pentest for reasons of reporting or roll back in the event of an incident, which is not obvious for a human being.

- *Reporting*

Reports must be handwritten, which is a long process with much room for error.

- *HR flexibility*

It is very difficult to recruit a new test pen because of non-standardization. In fact, each test pen follows its own methodology.

Automated Pentest

It is automating the process end in order to exceed the limits of the manual pentest. Before going into the details of the description, the pentest (regardless of the type) must go through four major phases:

1. Data gathering

2. Vulnerability Analysis

3. Exploit

4. Reporting

In order to ensure pentest automation, and minimize human intervention, it is necessary to automate the previous phases as follows:

- *Data gathering*

The data collection phase consists of clearly defining the scope of work; it is the reference phase on which all the other phases will be based.

The automatic system must be able to import a list of assets to test a lot of information behind it, and this information will have to reflect the details of each asset in the scope, namely IP address, application, type of the asset, version, criticality of data. CMDB tools can help with this task, especially when it comes to automatic recognition tools.

- *Vulnerability Analysis*

Based on the information from the previous step, there are tools that will work with the data collection output to automatically perform vulnerability scans. Also, there are details about the asset versions better the analyzers vulnerabilities will customize the plugins to use for its specific versions, which will encourage more concrete results and dismantle false positives. Its platforms include tools that will compare the different versions tested with a centralized database.

There are many vulnerability analysis tools in the market such as:

- NESSUS

- Rapid7 Nexpose

- OpenVAS

The output of this phase is a set of what are called Common Vulnerabilities and Exposure (CVE) with the different assets tested, and optionally a list of recommendations.

- *Exploit*

By associating an automatic exploit tool with the outputs of the previous phase, we can tell the system to automatically exploit the vulnerabilities of the assets to which VECs have been associated.

Of course, with these tools we can add conditions, for example, we would not want, if the system finds a CVE related to a vulnerability to DDOS, that the agent puts down our system?

- *Reporting*

Thanks to automated reporting tools, the state of the pentest process can be reflected to answer the following questions:

- *What is the number of vulnerable systems?*

- *On the whole of the vulnerabilities what is the number of the exploited systems?*

- *What is the vulnerability classification in order of criticality?*

Scheduled reports can be configured and sent automatically to the people concerned with the need-to- know principle.

Benefits of automated pentest

- *Execution speed*

Automated tools work faster. In a penetration test, a complete manual approach would be followed in an automated approach.

- *Number of tests*

Automated tools are perfect for testing a large number of payloads. Therefore, automated tools can cover a wider scope .

- *Coverage*

Manual tests can not cover everything from A to Z.

- *Required skills*

A manual penetration test requires an expert or a team of experts, while a automated pentest requires the mastery of the concepts and tools behind it.

- *Reporting*

A nice and clean report with one click is a great advantage with automated tools to save a lot of time. The manual creation of a penetration test report is obviously a tedious task. Most automated tools now provide a way to produce the final report as needed.

[Source: https://resources.infosecinstitute.com/automated-tools-vs-a-manual-approach/]

How to adopt the automated pentest within your company

Identify motivation goals

After all, why set up an automated pentest system? You must have a clear objective, which can be:

- The alignment at a given standard, ISO 27001, PCI DSS

- Simply, the control of the risks related to the faults

- Test the robustness of the systems of the future suppliers

Define a clear process

Once the objectives are defined, and before proceeding to the technique, it is very important to define a documented process on which to rely during the automation of the pentest process, the latter can be comprised of the following steps:

- Preparation

- Scan vulnerability

- Reporting
- Exploit
- Reporting
- Define Remediation Actions
- Implement remediation actions
- Rescan
- Reporting

Each company applies the previous phases according to its environment.

Identify and be aware of internal & external stakeholders

Setting up a vulnerability management system will require the intervention of several stakeholders, whether in the security team or in the other operational teams. It is important to document the stakeholders, and to define their roles and responsibilities in relation to the program. It is recommended to define OLAs with internal teams to minimize the waiting times for exposure of the risks found.

Implementation of an asset inventory tool

An automated pentest program will require the automation of the company's inventory of information assets, a quality inventory must also include higher levels of information such as the criticality of the asset that will improve the quality of the reports provided thereafter.

Definition of a VAPT schedule according to the needs of conformities

A VAPT calendar (Vulnerability Assessment & Penetration Test) is essential to have a visibility on the future tests to be carried out automatically by the systems. This will make it possible to know how to take the right decision if ever there is a system to exclude in a given moment.

Set up a vulnerability management tool

Once the various steps have been successfully completed, it is now time to move to the choice of the vulnerability management solution. The choice may depend on the organization and the decision may depend on one or more of the following criteria:

- Budget allocated
- Number of guests
- Competence of safety engineers
- Quality of the support provided
- Integrability with exploit tools
- Quality of reports provided

Setup an exploit tool

Choosing the right tools to integrate with the Vulnerability Management Systems is a very important task, because it is during this phase that we will connect the two systems in such a way as to allow the automatic exploit.

Setting gateway rules between the analysis tools and exploits

Once the exploit tools are set up and connected with the vulnerability management system, it is now time to define the rules of automatic exploits, which can be as follows:

- Do not exploit when it comes to vulnerability on the X server
- Do not exploit in the X days, Y
- Avoid exploiting DDOS threat types
- Avoid each exploit on the X databases, Y, Z

Identify the needs in terms of reporting

It is important to choose the report templates to be created automatically or manually, compared to the need for compliance or otherwise, and to assign access according to the different levels, for example to ensure the patching of their systems teams operators can access a report specific to their systems.

Tests and continuous improvement

Although we manage to automate the pentest process, that does not mean that we are not going to do manual work, on the contrary. It is important to closely monitor the system at all levels, and to suggest areas for improvement in order to align with new threats and good practices.

References:

- https://resources.infosecinstitute.com/automated-tools-vs-a-manual-approach/
- https://www.coresecurity.com/content/penetration-testing

Cryptocurrency and Regulations

Sikkandar Sha

Sikkandar has over seven years of experience in IT risk consulting services with primary focus on pen testing, social engineering and cyber-crime investigation along with other IT security assessments and process audits. Overall responsibilities also include client relationship management, business development, proposal preparation, project delivery, stakeholder management and market analysis. Extensive industry experience in various verticals, combined with comprehensive technical knowledge and assess threats to client environments in order to provide solutions that add value to business.

In every booming industry, there will be scams. Why do scams exist? If we analyze the basic root cause of this issue, ignorance and greed are the major drivers of such scams. If you think you can buy Bitcoin today and become rich overnight, you are WRONG. It does not work that way and it never will. You have to do your homework thoroughly before you make any decision.

Introduction

Let's talk about the most Googled word after Jesus in the second half of last year - BITCOIN. Back in 2008, after the financial collapse in US due to irresponsible lending and bundled mortgages, Satoshi Nakamoto (original name unknown) decided to publish a "White Paper" and invented Bitcoin - a Peer to Peer digital currency that could establish transparency and bring decentralization to the financial sector, which basically removes the middle men aka banks. Crypto currencies have seen a massive rise over the past few years with currencies like Bitcoin leading the way.

Most of these cryptocurrencies (unlike the name would suggest) are not actually regulated like fiat currencies throughout the world but are treated as commodities. Although they are traded worldwide, cryptocurrencies like Bitcoin take most of their trade in East Asia, in countries like China, with the world largest community of Bitcoin traders and South Korea (who have just regulated Bitcoin) taking the vast majority of this. India as well has seen a massive use of cryptocurrency over the past year but the waters are still murky on where its future will lie.

Crypto as of today is still at its infancy stage. A similar analogy would be to go back to 1994 and talking about the internet and everyone would discard the idea, although without the same discarded idea, it would be almost impossible to live today. Bitcoin did not attract much attention until the price climbed past $8k USD this past November. Then it went mainstream. Nobody batted an eye when it was $300 back in 2013 or $900 in 2016. And price is the same problem that also puts Bitcoin down. Bitcoin is not your conventional stock. There is an underlying technology called blockchain that is an open source code that deploys Bitcoin and all its variants. A few centuries ago when fiat currency replaced gold, it was laughed at and mocked. Then came the internet and then the Y2K problem and we can go on. Without a shadow of a doubt, blockchain will revolutionize every industry as we know it, due to the transparency it brings to the

table. You see some seniors today struggling to adapt to smart phones. That will pretty much be the case when blockchain takes over every industry. Get with the program or be left behind, whether you are an investor or not, you don't have a choice.

In India, there are now a number of unregulated cryptocurrency exchanges, but at this time there are no clear regulations in the country for virtual currencies.

Bitcoin and other cryptocurrencies are not illegal but RBI does advise against investing in them due to the risks involved. Since South Korea's recent regulation of cryptocurrencies to ban all unregulated exchanges, their value has plummeted and it's estimated investors have lost millions across the globe.

India's major financial institutions, including the SEBI & RBI, do not recognize currencies like Bitcoin as legal tender.

Role of Media

The media plays a very important role in the crypto industry. The extra hype caused by the media has a drastic impact on people. It's a field day for media when Bitcoin slumps from $10K USD to $8k USD and only a few weeks earlier it jumped from $5K to $8K. How is that a crash? In the blockchain community, it is called FUD (Fear, Uncertainty and Doubt). Bitcoin was invented in 2009 and for anyone calling it a bubble, have a look at Bitcoin's price graph and then decide for yourself. In countries like Canada, one can tap his/her phone at stores to pay for bills in cryptocurrency. If this was back in 2006, would you believe the concept? Of course not. That's the scenario with blockchain today. We predict mass adoption of blockchain by 2025. This adoption would disrupt every industry as we know it, due to the transparency and data accuracy compared to the Enterprise solutions we have today. We would never ask people to invest their money into something that they're uncertain of, which is fair enough. But from a technology perspective, blockchain is here to stay. As quoted by Satoshi Nakamoto in his white paper, "Welcome to the Future".

Initial Coin Offerings (ICOs) & Scams

In every booming industry, there will be scams. Why do scams exist? If we analyze the basic root cause of this issue, ignorance and greed are the major drivers of such scams. If you think you can buy Bitcoin today and become rich overnight, you are WRONG. It does not work that way and it never will. You have to do your homework thoroughly before you make any decision.

ICOs (Initial Coin Offerings) have created a lot of waves last year taking advantage of the crypto boom and scamming people. If you are reading about a company that you want to invest in on Facebook, then you are doing it wrong. Reddit is a great source for information and other significant platforms include Telegram and Slack, which are both available on Android and iOS. Mark Zuckerberg announced on 30th of January 2018 that he will be banning all advertisements and ad channels regarding cryptocurrency on Facebook, which is a great move so people don't get misinformed or manipulated in any sense. An interesting fact that most people don't know about, or did not pay attention to, is that Tyler and Cameron Winklevoss were the twins who accused Zuckerberg of stealing their idea and Mark had to settle with them for $65M USD. Now the Winklevoss brothers, who were already millionaires before they went to Harvard, took the $65M USD and bought Bitcoin back in 2012 and are the first officially "recognized" Bitcoin billionaires of the world and operate the most successful crypto exchange, Gemini, based in US.

Trading Cryptocurrencies

Trading crypto is not advisable for beginners. You will need to study Technical Analysis (TA) and chart coins based on several parameters. Otherwise, it's all speculative gambling. If you are an investor, choose good projects and leave it for some time and you'll be amazed by the ROI and this applies only to the investors who do their homework. Trading takes quite a while as you should position your entry and exit wisely and the key is not to be greedy. Remember, this is not a slot machine!

Regulations in different countries and their current stance on crypto

- China likes everything to be in house and does not like Westerners disrupting their business ecosystem. They have officially chosen two projects so far, Neo and Walton. More cryptos can be built on top of Neo as it's a platform.
- Canadian government is currently considering using Ethereum Blockchain for all government funding which will completely eradicate corruption.
- In August 2017, Vitalik Buterin, the founder of Ethereum was invited by Vladimir Putin to discuss the possibilities for developing an in-house Russian coin that would serve the purpose that Russia intends to use it for.
- Omisego, an Ethereum smart contract token, has been in talks with the banks of Thailand to deploy it as a payment gateway.
- Ripple (XRP) is a transaction protocol that has already made partnerships with most of the banks across the world including State Bank of India (SBI).
- In politically disturbed countries, like Zimbabwe and Venezuela, where their fiat currency literally crashed to 0, Bitcoin and Dash are being used as alternate payment systems.
- Amid Venezuela's current political situation and economic condition, Venezuela launched its own oil backed cryptocurrency called "Petro", in hopes that it will circumvent the US financial sanctions and revive the country's economy.
- Power Ledger, which is an Australian project that intends to disrupt the energy industry promoting solar panels, are currently in talks with multiple countries, including India, and Tech Mahindra is doing the test phase of the project.

There are several projects that you can actually research by going to coinmarketcap.com and you can do a simple Google search on all the facts stated in this article.

Indian Government – Plans for the future

More and more people are starting to trade cryptocurrencies in India every day, with India's biggest exchange, Zebpay, adding around 200,000 new users every month. With these large numbers and the vast amount of wealth being traded, the Indian Government feels it needs to take action. In April last year, the Indian government took the first step towards possible regulations by setting up an interdisciplinary committee of government and central bank officials to propose regulations on digital currencies.

Later in November, the Supreme Court filed a petition and ordered the government to respond, regarding the regulation of the flow of digital currencies and to make sure that they would be made accountable to the exchequer.

Whether a ban or to impose regulations to come into place is yet to be seen but plans have started to be set in motion. Most traders, whilst unhappy with a ban, are really hoping that some clarity regarding cryptocurrencies in India will come in the near future.

Currently, the Finance Ministry is looking into the possibility of putting into place regulations on cryptocurrency exchanges but this is still some time away as the Parliament would need to legislate any new regulations. There is also concern about protecting investors who are using off-shore exchanges. Many cases have been brought to light of investors complaining of fraudulent transactions.

It is likely that if regulations come into place, they will be concerning specific currencies and not the technology behind them, which is blockchain. Many of India's largest banks have started using the blockchain technology and RBI is even looking at the possibility of creating its own cryptocurrency.

Role of Income Tax Department in India

Whilst there are no current regulations in place, the government has sent out tens of thousands of tax notices to individuals involved in the virtual exchanges on the basis of recent reports that showed that over $3.5 billion worth of cryptocurrency transactions had taken place in India over a 17-month time frame. Under the 2002 Prevention of Money Laundering act, currencies like Bitcoin face many challenges. Due to the anonymity of the technology behind cryptocurrencies, it is impossible to trace distribution, and the current lack of regulation makes it easy for people to use illegally obtained money and convert it into 'clean' legal cash, which is something the government wants to prevent. Countries like Japan and Russia have already put a full-time ban in place to prevent these types of transactions. Many investors are not reflecting their cryptocurrency earnings on their tax return.

The main tax concern is that those dealing with the currency are not paying a capital gains tax to the government, which could have a disastrous effect on the economy. The future for Bitcoins is still unclear in India but it's likely that in the time to come, steps will be taken to regulate it but it is still unclear what these regulations will be. Other parts of the world have taken different stances on cryptocurrencies with some having put stringent bans in place while others have put in various levels of regulation. We will have to wait and see how the Indian government chooses to move forward on this tricky issue.