Penetration Testing on Android Device

Saranya Elangovan
Computer Science Department
College of Science
Illinois Institute of Technology
Chicago, IL, USA
srajagopalanelangova@hawk.iit.edu

Deepak Govind Mukundan
Computer Science Department
College of Science
Illinois Institute of Technology
Chicago, IL, USA
dmukundan@hawk.iit.edu

Vinita Nanavati
Computer Science Department
College of Science
Illinois Institute of Technology
Chicago, IL, USA
vnanavati@hawk.iit.edu

Abstract— **A Penetration test, is a simulated cyberattack against your computer system to check for exploitable vulnerabilities. Penetration testing helps to secure network and to determine security weakness. In this project we perform different penetration testing using private network, android device, virtualized system and tools. We predominantly use tools within the Kali Linux. The key focus of this paper is to establish connection between Kali Linux and android mobile's web browser. Further analysis involved finding SQL injection and XSS Scripting vulnerabilities. We demonstrate on attacking smartphone and summarize detail steps and methods while conducting these attacks.**

Keywords—Penetration testing, Kali Linux, SQL Injection ,XSS android (key words**)**

I. INTRODUCTION

Pen Test is a method of evaluating computer and network security by simulating an attack on a computer system or network from external and internal threats. Penetration Testing is used to find flaws in the system to take appropriate security measures to protect the data and maintain functionality. It is a proof-of-concept approach to explore and exploit vulnerabilities. This process confirms whether the vulnerability really exists and further proves that exploiting it can result in damage to the application or network. This process involves an analysis of the system for any potential vulnerabilities that could result for poor or improper system configuration, both known and unknown hardware or software flaws, and operational weaknesses in process or technical countermeasures.

Smartphones have become indispensable Mobile devices for many users with different types and versions. The Android platform most widely used systems in the smartphones and it involves more security challenges.

II. TYPES OF TESTING

There are five types of Penetration Testing:

A. External testing

External penetration tests target the assets of a company that are visible on the internet, e.g., the web application itself, the company website, and email and domain name servers (DNS). The goal is to gain access and extract valuable data.

B. Internal testing

External penetration tests target the assets of a company that are visible on the internet, e.g., the web application itself, the company website, and email and domain name servers (DNS). The goal is to gain access and extract valuable data. In an internal test, a tester with access to an application behind its firewall simulates an attack by a malicious insider.

C. Blind testing

In a blind test, a tester is only given the name of the enterprise that's being targeted. This gives security personnel a real-time look into how an actual application assault would take place.

D. Double blind testing

In a double-blind test, security personnel have no prior knowledge of the simulated attack.

E. *Targeted testing*.

In this scenario, both the tester and security personnel work together and keep each other appraised of their movements.

III. WHY PEN TEST IS IMPORTANT ?

There are a variety of reasons for performing a penetration test. One of the main reasons is to find vulnerabilities and fix them before an attacker does. Pen

test evaluates a system's ability to protect its networks, applications, endpoints and users from external or internal threats. It also attempts to protect the security controls and ensures only authorized access. The fact penetration testing provide an excellent view of the actual security state of an environment as well as the organization security state.

Penetration testers get to view security in an actual operational context, not merely on document or in discussions. Pen testers can concentrate on the most likely exploitable issues and see if an actual attacker could take advantage of them findings. The main objective of penetration testing is to determine the security weaknesses. A penetration test can also be used to: 1) test an organization's security policy compliance; 2) test employee security awareness; and 3) test an organization's ability to respond to security incidents.

IV. Pen Test Tools

A prime advantage of using open source Pen Testing tools is that they are constantly being refined by contributors and other kinds of Cyber security professionals to ensure that they stay at the forefront of the ever-changing threat landscape.

There is also several penetration testing software

- ***Metasploit*** - An Advanced Framework used for pen testing that contains command-line and GUI interfaces.
- ***Wireshark*** – A framework tradition understood for making arranged the tiniest experiences about your depiction, framework traditions and so on. The information that is got back by method for in this way can be seen through GUI.
- ***w3af* –** It is an instrument for assaulting and reviewing the web environment applications. It can be dynamic in a wide range of environment conditions with introduced python.
- ***John the Ripper*** – A password cracker.
- ***Nessus*** – A very robust vulnerability identifier. It is a helplessness scanner that contains immense library of vulnerabilities and tests to recognize them. It contains OS recognition and port checking, so Nessus call nmap for testing of these parts.
- ***Nmap*** – A network mapper, as the name suggest, that aids in understanding the characteristics of any target network.
- ***Dradis*** – An open source framework that helps with maintaining the information that can be shared among the participants of a pen-test.
- ***BeEF*** – BeEF is short for "Browser Exploitation Framework" and focuses on web browsers.
- ***Saint*** – Chairman's Network Tool (SAINT) recognize all the framework vulnerabilities remotely furthermore concentrate on different targets
- ***Core Impact*** - It abuses vulnerabilities in project without framework disappointments and runs tests for a system. It permits the analyzer to change one machine and run robotized checks for extra machines inside the system.
- ***Codenomicon*** - This device helps in finding the most concealed vulnerabilities in the framework and give the best setup to the mechanized entrance testing.
- ***Hydra*** - It is the finest login snap device. It yields dependability and support more than thirty conventions to the pen-analyzers.
- ***Burp suite* –** Burp Suite is an integrated platform for performing security testing of web applications. Burp Proxy is an intercepting proxy server for security testing of web applications. It operates as a man-in-the-middle between your browser and the target application, allowing you to intercept and modify all HTTP/S traffic passing in both directions.
- ***SqlMap*** – It is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers.

V. Pen Testing Phases

Penetration testing is a combination of techniques that considers various issues of the systems and tests, analyzes, and gives solutions. It is based on a structured procedure that performs penetration testing step-by-step.

A. *Planning and Preparation*

Planning and preparation starts with defining the goals, the testing method used and objectives of the penetration testing. The common objective was to identify the vulnerability and improve the security of the technical systems.

B. *Reconnaissance*

This stage includes:

- Gathering intelligence (e.g., network and domain names, mail server) to better

understand how a target works and its potential vulnerabilities.

- The tester starts by analyzing the available information and, if required, requests for more information such as system descriptions, network plans, etc. from the client.

C. *Scanning*

The next step is to understand how the target application will respond to various intrusion attempts. Scanning is the process of finding openings in the target organization, such as wireless access points, internet gateways, available systems, vulnerability lists, and port listening. This is typically done using:

- ***Static analysis*** – Inspecting an application's code to estimate the way it behaves while running. These tools can scan the entirety of the code in a single pass.

- ***Dynamic Analysis*** –Inspecting an application's code in a running state. This is a more practical way of scanning, as it provides a real-time view into an application's performance.

D. *Gaining Access*

This stage uses web application attacks, such as cross-site scripting, SQL injection and backdoors, to uncover a target's vulnerabilities. Testers then try and exploit these vulnerabilities, typically by escalating privileges, stealing data, intercepting traffic, etc. to understand the damage they can cause.

E. *Maintaining Access*

The goal of this stage is to see if the vulnerability can be used to achieve a persistent presence in the exploited system.

F. *Analysis*

The results of the penetration test are then compiled into a report detailing:

- Specific vulnerabilities that were exploited.
- Sensitive data that was accessed.
- The amount of time the pen tester was able to remain in the system undetected.

G. *Report*

Report preparation must start with overall testing procedures, followed by an analysis of vulnerabilities and risks. The high risks and critical vulnerabilities must have priorities and then followed by the lower order.

VI. Existing Approach

Android is the largest installed base of any mobile platform and is growing rapidly. Every day another millions of users power up their Android devices for the first time and start looking for apps, games, and services. Android provides a world class platform for various usage such as creating apps and games for users everywhere around the world.

Android is an actualizing program input qualities and occasion driven framework which is for the most part and profoundly in view of client activities. Android frameworks contains LINUX working frameworks, GUI interface, framework libraries and introduced applications. Each android framework must have a show record which is a sort of XML document required to keep up the life cycle of applications in android cell phones. Show document contains data about android goal informing, administration, format, and movement. Numerous specialist and viewer learning about android-based cell phones since this is a standout amongst the most developing field currently innovation.

G1-Android is the first commercially available phone that features Google's Android Platform SDK. This is a partnership project with Taiwan Based HTC Corp and supported by the United States Service provider T-Mobile. Thus, the phone is available in the market as T-Mobile HTC Android. Android is the first truly open source platform for mobile devices with a fully integrated software stack that consists of an operating system, middleware, user-friendly interface and applications, and allows the users to develop additional software and change or replace functionality without limitations. To achieve the unlimited functionalities, Android uses Linux operating system as its core OS and ensures that users experience same Internet activities equal to what they can experience on a desktop PC. Several smartphone features and functions help to increase usage of data and services, but it is also open to the risk of introducing new vulnerabilities.

VII. IMPLEMENTATION - PHASE I

The implementation phase of penetration test will be based on the scenario of hacking Android phone.

A. *Planning and Preparation*

Goal: Perform System and Port Scanning on mobile device
Objective: Extract all basic data from the user

B. *Tools Used:*

Attacker:
Operating System: Kali Linux 2.0
RAM: 4GB
CPU: i5 7th gen processor.

Victim:
Android Device: OnePlus2
Operating System: Android 6.0.1 (Marshmallow)
RAM: 4GB
CPU: Octa-core 4x1.56 GHz Cortex-A53

C. *Scanning*

System Scanning
The System Scan allows to run a complete check of your computer system to see if there are any malicious threats that can be found on your PC.

Msfvenom:
To Proceed with system scanning, we first establish a connection between Kali Linux system with the mobile device through **hack.apk.** Send the file to the device via mail. Once a port has been set, the application listens to the port from the mobile and a connection is established. This is done through msfvenom command.

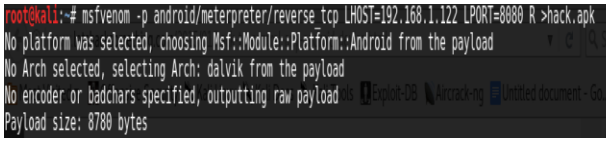

Figure 1: Msfvenom command to establish connection to the mobile device.

Msfconsole:
To start exploiting data from the mobile, we use msfconsole command. Once the session has started, we can exploit data like call log, contacts list, messages etc.

Figure 2: Msfconsole to exploit data from mobile device.

D. *Exploiting Data*

Android Commands:
Various android commands are available on kali linux to exploit data from the user. Some of them are:

a. activity start: Start an android activity from the uri string
b. check root: Check whether the device is rooted or not.
c. dump_calllog: Get call log
d. dump_contacts: Get contacts list
e. dump SMS: Get SMS messages
f. geolocate: Get current location of the user.
g. send_sms: Send sms from target session.
h. set_audio_mode: Set Ringer mode.
i. Wakelock: Enable/Disable wakelock
j. wlan_geolocate: Get current lat-long using WLAN information.

Dump Contacts:
In Order to obtain all contacts available on the mobile device, we use the command dump_contacts. Now all contacts are copied into the local system in a .txt file.

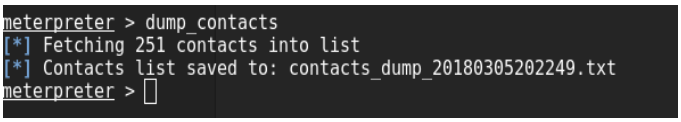

Figure 3: dump_contacts Command

Send SMS from Kali Linux:
Use send_sms command to send messages by specifying the number to which the message must be sent along with the content of the message.

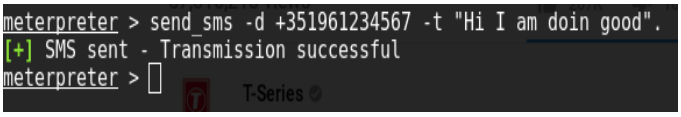

Figure 4: send_sms Command

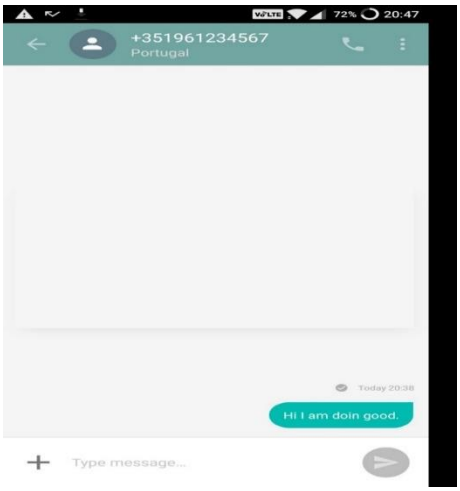

Figure 5: Screenshot from the mobile device.

Port Scanning:

Port scanning is a method which sends out a request to the target on each port and makes a note of status of ports - Open or close. SYN scan will tell which ports are listening and which are not depending on the type of responses generated.

We use Nmap command to scan for open ports on the android device. If an open port is found, then the attackers can inject data onto them and attack the system. In this case, there is only one open port available. It took almost 2.3 seconds to scan the entire android device to search for open ports.

```
root@kali:~# nmap 192.168.1.130
Starting Nmap 7.40 ( https://nmap.org ) at 2018-03-05 20:53 EST
Nmap scan report for 192.168.1.130
Host is up (0.027s latency).
Not shown: 999 closed ports
PORT     STATE SERVICE
9080/tcp open  glrpc
MAC Address: C0:EE:FB:5A:56:B4 (OnePlus Tech (Shenzhen))

Nmap done: 1 IP address (1 host up) scanned in 2.35 seconds
root@kali:~#
```

Figure 6: Nmap Command.

VIII. IMPLEMENTATION - PHASE II

WebView is a view that display web pages inside the application.Android WebView is a system component for the Android operating system (OS) that allows Android apps to display content from the web directly inside an application. WebView is powerful because it not only provides the app with an embedded browser, it also allows the developer's app to interact with web pages and other web apps.

Our first phase of experiment is to connect the Web application of Android Device with Kali Linux.Kali Linux takes the control of the android device and interpret the running session. A separate website was created to show the connections. The connection was established using Metasploit tool.

Steps to connect Android WebView and Kalilinux:

1) Run Metasploit and type "**msfconsole**" at a terminal prompt.

2) Type,"**use exploit /android/browser/webview_addjavascriptinerface"**

3) Then type, "**show options**" to see what needs to be set

4) Enter set "lhost 192.168.1.140".(Android device)

5) Enter, "set URIPATH Security".

6) Finally, type "exploit":

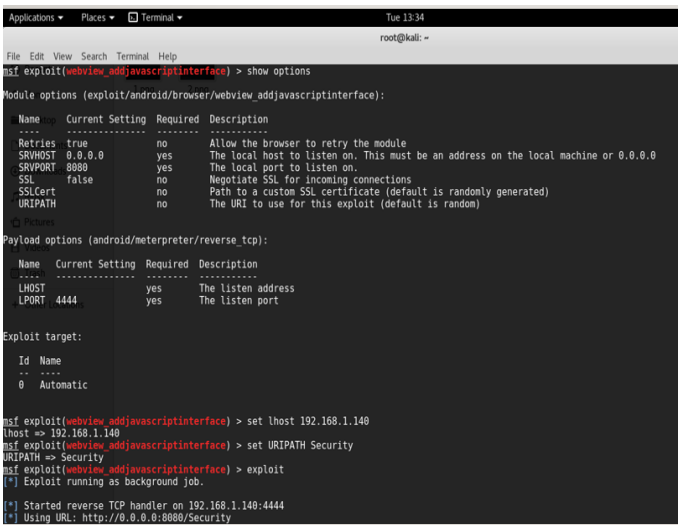

Figure 7: Connection establishment with Android Web View and kaliLinux

A server is started on the Kali system that hosts a webpage containing the exploit. A URL is provided including the URI path.

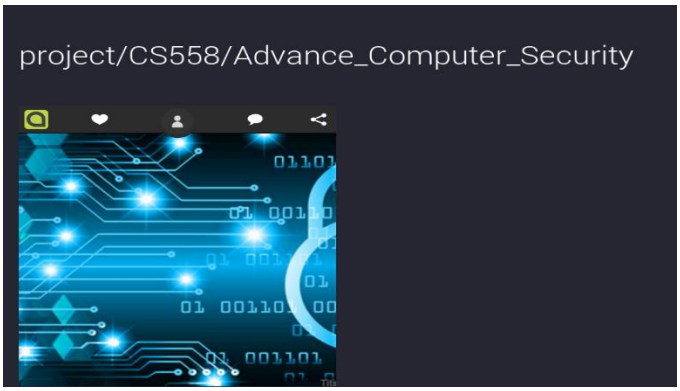

Figure 8: Website created for connection establishment

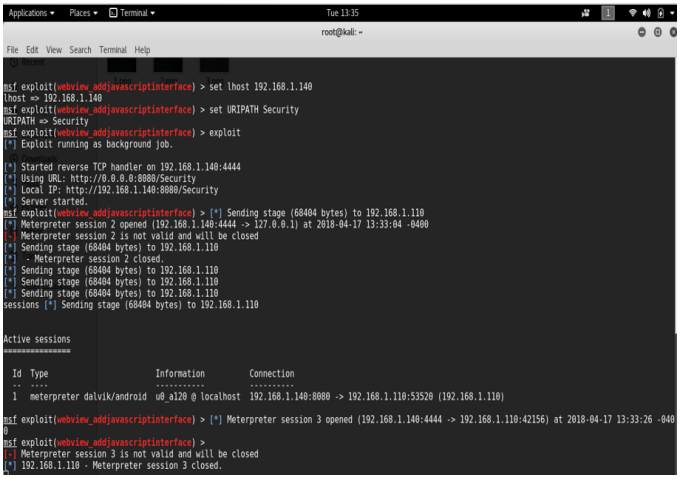

Figure 9: Active sessions web page displayed on mobile phone whose IP address is 192.168.1.140.

IX. IMPLEMENTATION - PHASE III

SQL INJECTION

SQL injection is the process of exploiting interfaces that read input from the user and directly interact with an SQL database. The attacker tries to craft a payload consisting of a custom SQL statement fragment as an input. The exploitation happens when the input interface does not have any mechanism to clean the input before sending it to the database. A successful SQL injection attack can result in disclosure of sensitive data, and some cases can even destroy the database completely. An attacker can use it to bypass a web application's authentication and authorization mechanisms and retrieve the contents of an entire database. SQL Injection can also be used to add, modify and delete records in a database, affecting data integrity. SQL Injection can provide an attacker with unauthorized access to sensitive data including, customer data, personally identifiable information, trade secrets, intellectual property and other sensitive information.

In order to run malicious SQL queries against a database server, an attacker must first find an input within the web application that is included inside of an SQL query. An attacker can then insert a payload that will be included as part of the SQL query and run against the database server.

An attacker do the following features :

- An attacker can use SQL Injection to bypass authentication or even impersonate specific users.
- One of SQL's primary functions is to select data based on a query and output the result of that query. An SQL Injection vulnerability could allow the complete disclosure of data residing on a database server.
- Since web applications use SQL to alter data within a database, an attacker could use SQL Injection to alter data stored in a database. Altering data affects data integrity and could cause repudiation issues, for instance, issues such as voiding transactions, altering balances and other records.
- SQL is used to delete records from a database. An attacker could use an SQL Injection vulnerability to delete data from a database. Even if an appropriate backup strategy is employed, deletion of data could affect an application's availability until the database is restored.

SQL MAP - A Brief Introduction

Sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester and a broad range of switches lasting from database fingerprinting, over data fetching from the database, to accessing the underlying file system and executing commands on the operating system via out-of-band connections.

Steps for SQL Injection:

Our experiment main focus is to identify vulnerable websites and introduce a SQL injections map is used to detect and exploiting SQL injection flaws.

1).Enter the vulnerable website “sqlmap -u www.sallatykka.com/web/index.php?id=31 –dbs

-u dbs reveals the web technology, scripting language ,the available database hosted on the website.

Figure 10 : Website technology and the scripting language used

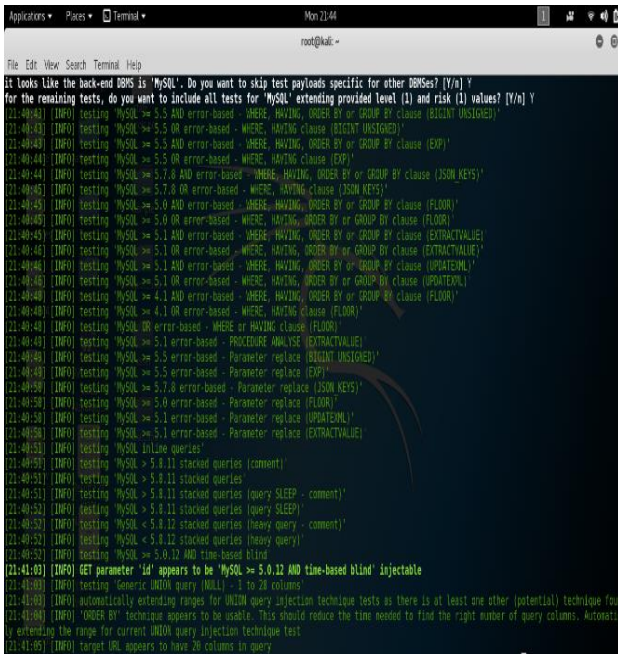

Figure 11: The backend database – My SQL.

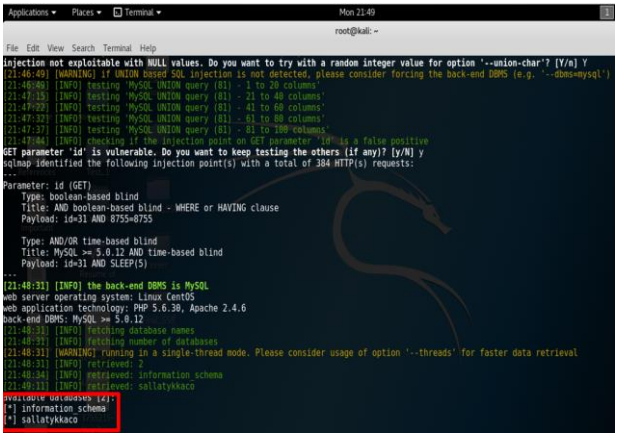

Figure 12 :Available database is displayed

2) Enter the vulnerable website “sqlmap -u www.sallatykka.com/web/index.php?id=31 –D sallatykka –tables

--tables -displays the available tables in the database.

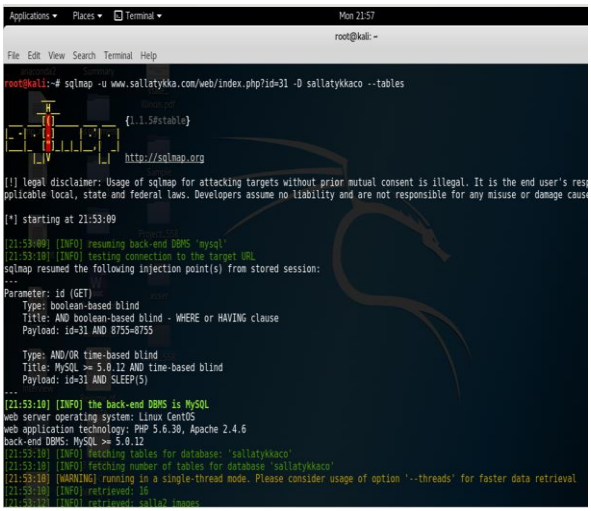

Figure 13: The available tables are displayed.

REMEDIATION

The best way to guard against such an attack is to put an `intermediate layer between the input interface and the database. We can do this by:

- using parameterized queries
- using stored procedures
- scanning user input for any escape characters

- Principle of Least privilege – Restricting admin type access to database servers

- Input Validation

- Blacklist Validation – Test the malicious input against a set of known malicious inputs

- Whitelist Validation – Test the malicious input against a set of known, approved inputs. The application knows exactly what's desired and rejects other inputs.

- Use testing tools to ensure deployed codes are secure. Enterprises and organizations may invest in testing tools such as web application scanners, vulnerability scanners, and static code analyzers. These tools help IT teams test and evaluate codes before, during, and after deployment

- Consider using web application firewalls. These provide firewall protection at the web application level.

- Practice secure coding. Companies with websites must employ and implement secure coding standards. The Open Web Application Security Project (OWASP) is a not-for-profit organization that helps web developers, administrators, and owners practice safe coding via community feedback.

- Patch systems and networks accordingly. IT administrators should take special care in making sure ALL systems in the network are patched, because one unpatched system may spell disaster. This prevents cybercriminals from exploiting vulnerabilities in unpatched/outdated software. Scan web applications for vulnerabilities: Enterprises need to check their web apps for vulnerabilities as these can lead to SQL injection and cross-site scripting attacks.

X. IMPLEMENTATION - PHASE IV

CROSS SITE SCRIPTING

Cross site scripting (XSS) is a type of attack that can be carried out to compromise the users of a website. XSS allows the attackers to inject malicious client-side script codes into web pages viewed by the users. By leveraging XSS, an attacker need not target a victim directly. Instead, an attacker would exploit a vulnerability within a website or web application that the victim would visit, essentially using the vulnerable website as a medium to inject a malicious script to the victim's browser. In order to run malicious JavaScript code in a victim's browser, an attacker must first find a way to inject a payload into a web page that the victim visits. This can happen if the website directly includes user input in its pages, because the attacker can then insert a string that will be treated as code by the victim's browser. The script assumes that a comment consists only of text. However, since the user input is included directly, an attacker could submit this comment: "<script>...</script>". When the user's browser loads the page, it will execute whatever JavaScript code is contained inside the <script> tags. The attacker has now succeeded with his attack. Here, we achieve this by utilizing Burp Suite Tool in Kali Linux on webpages through Android WebView.

BURP SUITE – A Brief Introduction

Burp Suite is an integrated platform for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.

Tools Used By Burp Suite

- ***HTTP Proxy*** - It operates as a web proxy server, and sits as a man-in-the-middle between the browser and destination web servers. This allows the interception, inspection and modification of the raw traffic passing in both directions.
- ***Scanner*** - A web application security scanner, used for performing automated vulnerability scans of web applications.
- ***Intruder*** - This tool can perform automated attacks on web applications. The tool offers a configurable algorithm that can generate malicious

HTTP requests. The intruder tool can test and detect SQL Injections, Cross Site Scripting, parameter manipulation and vulnerabilities susceptible brute-force attacks.

- ***Spider*** - A tool for automatically crawling web applications. It can be used in conjunction with manual mapping techniques to speed up the process of mapping an application's content and functionality.
- ***Repeater*** - A simple tool that can be used to manually test an application. It can be used to modify requests to the server, resend them, and observe the results.
- ***Decoder*** - a tool for transforming encoded data into its canonical form, or for transforming raw data into various encoded and hashed forms. It is capable of intelligently recognizing several encoding formats using heuristic techniques.
- ***Comparer*** - A tool for performing a comparison (a visual "diff") between any two items of data.
- ***Extender*** - allows the security tester to load Burp extensions, to extend Burp's functionality using the security testers own or third-party code.
- ***Sequencer*** - a tool for analyzing the quality of randomness in a sample of data items. It can be used to test an application's session tokens or other important data items that are intended to be unpredictable, such as anti-CSRF tokens, password reset tokens, etc.

Steps For Cross Site Scripting -Attack In Website :

In our work, we input client side malicious JavaScript codes into "voices.iit.edu" website to perform cross site scripting using burp suite. All we've got to do are some simple tasks to carry out the attack.

1) First, we have to make the burp suite tool listen to the web browser at all times. Now we have to set the target to the website which we are going to attack. Set the host to the website "voices.iit.edu". Now the burp suite tool will be listening to all the sessions of the browser where the user navigates to that particular website.

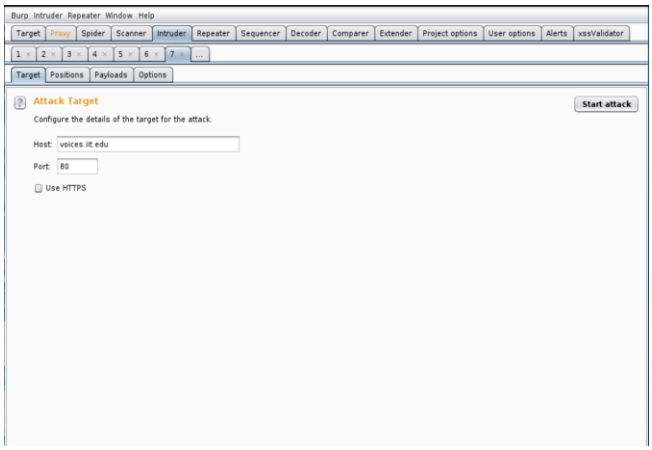

Figure 13: Set the target to "voices.iit.edu" under the Intruder Tab in the tool.

2) There will be a search box in that webpage prompting the user to search for any items available in the website's database. Once the user searches for anything using that search box, the burp suite tool will be immediately notified and the attack can be set. Now under the Positions tab, the attacker can identify where to inject the malicious script codes. We can see in the below figure that there is a highlighted portion stating "name" in it. The attacker will utilize that parameter to inject malicious code. The api fetched is under GET parameter so injecting a value into it is as easy as it gets.

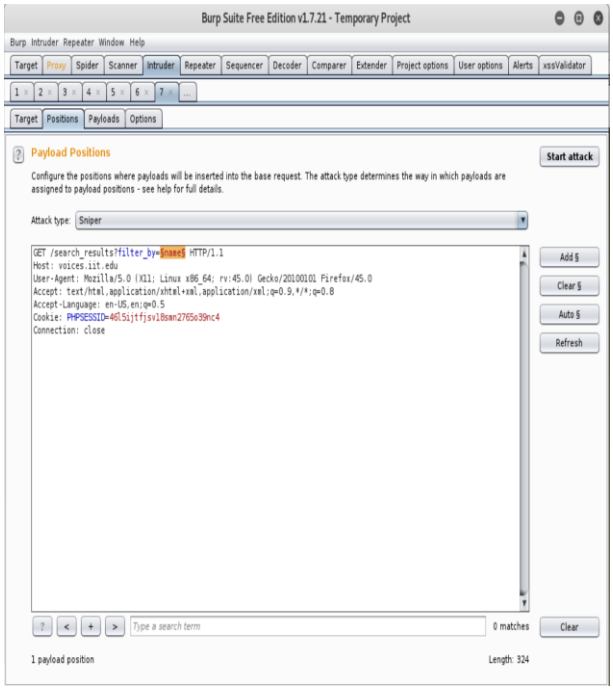

Figure 14: Burp suite session of the webpage. The highlighted parameter "name" is the attack parameter for the attacker.

3) Now that the parameter to attack has been identified, the attacker can move on to the next phase (i.e) to inject payloads into the parameter. These payloads contain malicious script code which can help the attacker retrieve data about the user. All the attacker needs to do is insert malicious script code, say, "<script>alert(document.cookie)</script>" into the 'Enter a new Item' textbox and then click add. The payload will be added into the list. These payloads will be overwritten at the place of the highlighted "Name" in the GET parameter which we saw earlier. Therefore each time a user tries to search for anything in that website, the attack will occur and will display an alert on the web page providing the session cookie. Thus the user can identify that an attack has happened on the web page in this web browser.

4) Once all this has been completed, all the attacker needs to do is to start the attack on the site. The attacker can select which payload to insert among multiple payloads added to the web page. Once a payload has been selected, the trap has been set and once the user tries to search for anything, the burp suite tool will do its work and the attacker can exploit all the information about the user through the attack and the tool.

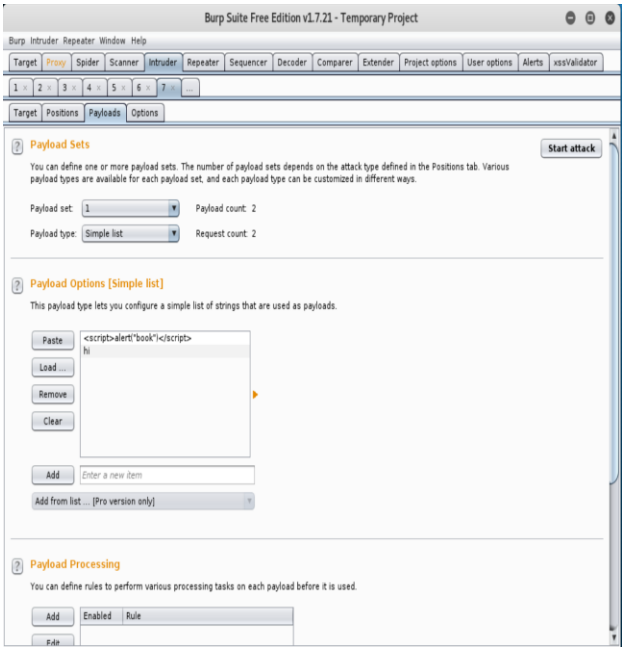

Figure :15 Set Payload (i.e) malicious scripts to be injected into the website.

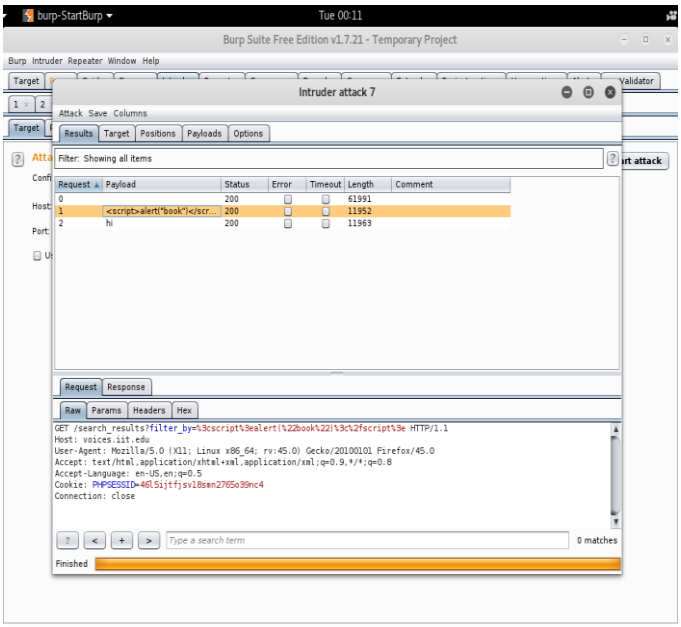

Figure 16 : The selected payload gets inserted into the GET parameter and the attack commences. The attacker can look into the attack under the responses tab

Thus by this method the attacker can perform Cross site scripting on websites and extract useful information about the user without a snitch.

REMEDIATION

The remediation of XSS vulnerabilities is heavily context-dependent and the patches vary. Here are some general tips (where UNTRUSTED is where user supplied data). Here are some ways describing what the developers can do the make their web site less vulnerable for the attackers.

- ***HTML Body*** – Convert special characters used in html pages to HTML entities. For example, convert & to &. By this way, the attackers cannot easily identify script codes to enter in the text boxes and the attacks can reduce.

- ***HTML Attribute*** - Convert the untrusted user input to HTML entities to prevent the creation of other attributes and never let any user data into the "id", "class" or "name" parameters in HTML tags. The developer needs to be very cautious when providing

user data into DOM event handlers (e.g. onclick), at they are made to execute JavaScript.

- ***Untrusted URL*** – The developers can URL encode the user data, whitelist known URLs and run the user data through a proper URL library in your language. Take notice to the protocol specified and if you expect HTTP or HTTPS links, whitelist those. Prevent JavaScript from running by using a protocol handler.

- ***GET Parameter*** - URL encode the user data and prevent the use of ampersand as it may lead to parameter pollution issues. This can be done by developing Application Programmable Interface (APIs). APIs encode the URL data and the developers can build it in such a way that user identifying data does not get displayed at the URL component of the browser. Often GET parameters can be easily targeted and utilized to inject malicious script codes into the website.

- ***CSS Values*** – Hex encode CSS values. For example, instead of specifying the background color of the web page to be 'white', the developers can use hexadecimal code like '#ffffff' or use 'rgb(255,255,255)'.

- ***JavaScript Variable*** – Add quotes around a JavaScript variable and hex encode their values so that the users cannot easily get the values of parameters used in the site easily without a key. Also prevent line breaks in code. Line breaks may lead to special characters getting added to the page.

- ***DOM XSS*** - Sanitize using a library written in the language you use. Enforce the use of safer functions whenever applicable (e.g. inner Text instead of inner HTML). Be very careful when determining what data are allowed to be printed. It's better to have a whitelist of allowed characters than a blacklist.

XI. Analysis And Lesson Learnt

Our main idea firstly was to establish connection between kali Linux and android mobile device. After this step we moved ahead with the insertion of vulnerabilities like SQL injection and Cross site scripting. After insertion was the remediation step that suggests the steps to prevent such flaws in future.

However, after scanning for open ports (while there was connection between kali Linux and android device apps)only 1 open port was found which was not sufficient to insert vulnerabilities. Hence, we took control of android mobile's web browser with the help of kali Linux. This made android device as web app server thus found more open ports that ease the purpose of insertion of vulnerabilities defined above.

XII. Conclusion

After conducting the penetration test on Android phone by it can be summarized the findings of this tool which are the Android system is a vulnerable and can be hacked through Wi-Fi. The attacker can access to sensitive data Thus it was found of Android Operating system can easily attacked. It is very interesting to note that Android use on smartphones has increased significantly over the last few years because of the open source platform it adopted with so many features available. Not everyone knows initially the type of vulnerabilities in specific system. However, checking the OWASP(Open web application security project)top 10 vulnerabilities may ease the process of vulnerability assessment and thus prevents risk to systems.

XIII. Future Work

Our next step in future will be to find Blind SQL injection like Boolean based and time based vulnerability along with the remediation steps using kali Linux. The cross site session hijacking prevention can be further enhanced, based on complex encryption techniques.

XIV. References

[1] Denis, Matthew, Carlos Zena, and Thaier Hayajneh. "Penetration testing: Concepts, attack methods, and defense strategies." 2016 IEEE Long Island Systems, Applications and Technology Conference (LISAT), 2016. doi:10.1109/lisat.2016.7494156

[2].Xiong, Pulei, and Liam Peyton. "A model-driven penetration test framework for Web applications." 2010 Eighth International Conference on Privacy, Security and Trust, 2010. doi:10.1109/pst.2010.5593250.

[3].Bau, Jason, Elie Bursztein, Divij Gupta, and John Mitchell. "State of the Art: Automated Black-Box Web Application Vulnerability Testing." 2010 IEEE Symposium on Security and Privacy, 2010. doi:10.1109/sp.2010.27.

[4].Piscitello, David. “Your First Penetration Test”. WatchGuard LiveSecurity.URL:http://www.corecom.com/external/livesecurity/pentest.html (retrieved 5 December, 2015).

[5].OUSPG Glossary of Vulnerability Testing Terminology.
URL:
http://www.ee.oulu.fi/research/ouspg/sage/glossary/
(retrieved 5 December 2015)

[6] F. Alisherov, and F. Sattarova, *Methodology for Penetration Testing*, International Journal of Grid and Distributed Computing, *2(2), 2009*, 44-49.

[7] https://blog.appknox.com/understanding-owasp-top-10-mobile-client-side-injection/

[8] https://hub.packtpub.com/knowing-sql-injection-attacks-and-securing-our-android-applications-them/.

[9]Sql Injection by Roger bisson MBCS

[10] W. G. Halfond, J. Viegas and A. Orso, “A Classification of SQL Injection Attacks and Countermeasures,” College of Computing Georgia Institute of Technology IEEE, 2006, pp. 1-7.

[11] W. G. Halfond and A. Orso, “Detection and Prevention of SQL Injection Attacks,” Advances in Information Security, Malware Detection, vol. 27, 2007, pp. 85–112.