（1）加载本地的资源文件。（假设在你手机assets，或者sdcard存在以下index.html文件），如我们使用WebView的loadUrl方法加载本地的index.html代码。

WebView mWebview= (WebView)findViewById(R.id.wv1);

WebSettings wSet = wView.getSettings();

wSet.setJavaScriptEnabled(true);

//mWebview.loadUrl("file:///android\_asset/index.html"); //工程目录assets index.html文件

//mWebview.loadUrl("content://com.android.htmlfileprovider/sdcard/index.html");//本地sdcard文件。

如使用WebView的loadUrl方法加载本地的一张图片。

String str = "file:///android\_asset/icon.png";

mWebview.loadUrl(str);

如我们也可以用webView的loadDataWithBaseURl来完成.

String data = "<HTML>在模拟器 2.1 上测试WebView,这是<IMG src=\"APK'>file:///android\_asset/zhangluyue.jpg\"/>张小月的图片";

// SDK1.5本地文件处理(不能显示图片)

//mWebview.loadData(URLEncoder.encode(data, encoding), mimeType, encoding); // SDK1.6及以后版本 //mWebview.loadData(data, mimeType, encoding); // 本地文件处理(能显示图片) mWebview.loadDataWithBaseURL(null, data, mimeType, encoding, null);

备注：

static final String mimeType = "text/html";

static final String encoding = "utf-8";

（2）加载远程资源文件。

直接网页显示google：

mWebview.loadUrl("http://www.google.com");

比较安全的Java和Js交互方式：

可以借鉴Android Intent的思路，Java和Js定义一个url格式如js://,Java调用Js方法，在Js方法中通过window.location.href=’js://?key=value#key1=value1’模拟跳转，被Java的shouldOverrideUrlLoading捕获，函数的返回值可以放在url的参数中。（Js调用Java方法原理相同）

这样的交互方式是异步的，如果你想知道调用一个Js方法是否返回了值怎么办？一般Java调用Js方法是在onPageFinished方法中，获得Js返回值是在shouldOverrideUrlLoading方法中，两个方法有个共同的参数webview,所以可以首先webview.setTag(false)，如果捕获到返回结果，则webview.setTag(true),postDelayed在很短时间比如300毫秒后，webview.getTag()检查是否有变化即可。

Android Webview漏洞

① 成名已久的任意命令执行漏洞，通过addJavascriptInterface方法，Js可以调用Java对象方法，通过反射机制，Js可以直接获取Runtime，从而执行任意命令。Android 4.2以上，可以通过声明@JavascriptInterface保证安全性，4.2以下不能再调用addJavascriptInterface，需要另谋他法。

② 之前爆出来的UXSS漏洞，可以越过同源策略，获得任意网页的Cookie等信息，Android 4.4以下都有此问题，基本无解，只能重新编译浏览器内核解决。

Java和Javascript安全交互

首先要说明几点：

1.Android Webview中Java调用Js方法很容易，loadUrl(“javascript:isOk()”)就可以调用isOk这个Js方法，但不能直接获取Js方法的返回结果。