

# Android Mobile Pentest 101

*© tsug0d, September 2018*

# Lecture 10.2 – Creating Exploit: Making HTTP Request

Mục tiêu: Gửi được truy vấn lên 1 trang web và lấy kết quả về

# Introduction

- Bài này giúp các bạn hiểu và code được truy vấn lên server 😊
- *Tại sao? Ah thì lúc hack được dữ liệu của user khác rồi thì phải gửi ra ngoài đúng hông, gửi lên web server là chuẩn bài rồi còn gì 😊*

# Let's dev

- Rồi, đầu tiên tạo project mới, chọn Empty Activity nhé
- Giờ chúng ta sẽ code chủ yếu trong 2 file **MainActivity.java** & **AndroidManifest.xml**

# Let's dev

- Nói sơ qua một chút về “**Permission**” (<https://developer.android.com/guide/topics/permissions/overview>)
- Đại khái là ban đầu app android sẽ không cho chúng ta tương tác với các vùng khác (như camera, internet, disk... vv), nếu bạn muốn thì phải tự request permission.
- Để gửi được HTTP Request, Cần phải định nghĩa trong **AndroidManifest.xml** 2 permissions:

```
android.permission.INTERNET  
android.permission.ACCESS_NETWORK_STATE
```

- Sử dụng <uses-permission> tag:

```
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

# Let's dev

- Xong phần cấu hình, giờ đi code nè
- Included mấy cái thư viện dưới đây vào MainActivity.java:

```
import androidx.appcompat.app.AppCompatActivity;  
import java.io.BufferedReader;  
import java.io.InputStreamReader;  
import java.net.URL;  
import java.net.URLConnection;  
import android.os.StrictMode.ThreadPolicy.Builder;  
import android.os.StrictMode;  
import android.os.Bundle;  
import android.util.Log;
```

# Let's dev

- Trong onCreate của MainActivity.java, chèn dòng này vào:

```
StrictMode.setThreadPolicy(new Builder().permitAll().build());
```

- Nhiều bạn sẽ tự hỏi dòng trên là clgt nên mình sẽ giải thích:

**StrictMode** là 1 class được load default bởi android, chặn các tương tác với disk I/O, Network access từ UI thread.

Khi tạo HTTP request, chúng ta sử dụng **URLConnection** class, mà cái này được chạy trên UI thread, do đó kiểu gì cũng dính cái error **"NetworkOnMainThreadException"**

Cho nên để né nó, thì add cái line trên vào để ghi đè cái policy cũ, allow all action là ok 😊

## Let's dev

- Tiếp theo thì gán giá trị vào url để truy vấn nào, ở đây là : <https://tsug0d.com/present/tsu.txt>

```
String url = "https://tsug0d.com/present/tsu.txt";  
StringBuilder url_holder = new StringBuilder();  
url_holder.append(url);
```



# Let's dev

- Tạo 1 connection (Nhớ để trong try – catch statement để khỏi error):

```
URLConnection conn = new URL(url_holder.toString()).openConnection();
```

- Set header cho connection:

```
conn.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");  
conn.setRequestProperty("charset", "utf-8");  
conn.setUseCaches(false);
```

- Tạo buffer để lấy dữ liệu về:

```
BufferedReader buffer = new BufferedReader(new InputStreamReader(conn.getInputStream()));
```

# Let's dev

- Xong goy, giờ đọc response trả về nữa thôi

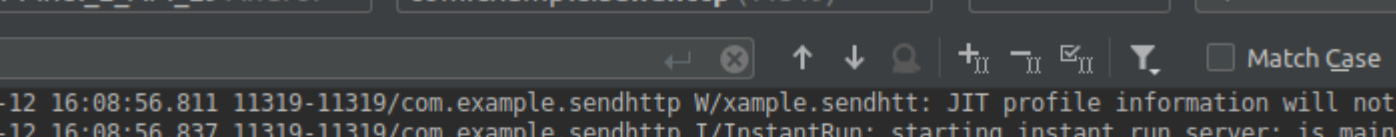
```
String response;  
String data_from_stream;  
for (response = new String(); true; response += data_from_stream)  
{  
    String stream = buffer.readLine();  
    data_from_stream = stream;  
    if (stream == null)  
    {  
        break;  
    }  
}
```

- Log ra xem nào 😊

```
Log.i("tsu", response);
```

# Let's dev

- Ngon! Hàng về.



Emulator Pixel\_2\_API\_29 Android com.example.sendhttp (11319) Verbose

tsu

2019-08-12 16:08:56.811 11319-11319/com.example.sendhttp W/xample.sendhtt: JIT profile information will not be recorded  
2019-08-12 16:08:56.837 11319-11319/com.example.sendhttp I/InstantRun: starting instant run server: is main process  
2019-08-12 16:08:56.972 11319-11319/com.example.sendhttp W/xample.sendhtt: Accessing hidden method Landroid/view/View;-  
2019-08-12 16:08:56.972 11319-11319/com.example.sendhttp W/xample.sendhtt: Accessing hidden method Landroid/view/ViewGr  
2019-08-12 16:08:57.000 11319-11319/com.example.sendhttp D/NetworkSecurityConfig: No Network Security Config specified,  
2019-08-12 16:08:57.477 11319-11319/com.example.sendhttp I/tsu: Oh? You're Approaching Me? MUDA MUDA MUDA x3.14  
2019-08-12 16:08:57.554 11319-13378/com.example.sendhttp W/OpenGLRenderer: Failed to choose config with EGL\_SWAP\_BEHAVI  
2019-08-12 16:08:57.556 11319-13378/com.example.sendhttp D/eglCodecCommon: setVertexArrayObject: set vao to 0 (0) 0 0  
2019-08-12 16:08:57.556 11319-13378/com.example.sendhttp D/EGL\_emulation: eglCreateContext: 0xeb7c8c40: maj 2 min 0 rcv  
2019-08-12 16:08:57.558 11319-13378/com.example.sendhttp D/EGL\_emulation: eglMakeCurrent: 0xeb7c8c40: ver 2 0 (tinfo 0x  
2019-08-12 16:08:57.566 11319-13378/com.example.sendhttp W/Gralloc3: mapper 3.x is not supported  
2019-08-12 16:08:57.581 11319-13378/com.example.sendhttp D/EGL\_emulation: eglMakeCurrent: 0xeb7c8c40: ver 2 0 (tinfo 0x  
2019-08-12 16:08:57.590 11319-13378/com.example.sendhttp D/eglCodecCommon: setVertexArrayObject: set vao to 0 (0) 1 0

- Trong trường hợp bạn quá sida thì vào đây tham khảo code này

<https://github.com/tsug0d/AndroidMobilePentest101/blob/master/lab/AndroidManifest.xml> <https://github.com/tsug0d/AndroidMobilePentest101/blob/master/lab/MainActivity.java> <https://github.com/tsug0d/AndroidMobilePentest101/blob/master/lab/MainActivity.java> <https://github.com/tsug0d/AndroidMobilePentest101/blob/master/lab/MainActivity.java>