Notes on using Wireshark for Monitoring Mobile HTTP Network traffic

You can find wireshark on line - it is a free tool.

https://www.wireshark.org/

Note that you may not be able to capture the mobile traffic on Windows because of WinPCap limitations. You may need to buy an additional adapter to do this. I'm using Mac to implement this functionality.

http://wiki.wireshark.org/CaptureSetup/WLAN#windows

What is it?

Wireshark is a tool for monitoring network traffic. Unlike an HTTP proxy server where you have to configure your machine to point to the HTTP proxy server in order to monitor the traffic. With Wireshark, you tell it to capture traffic from your network card, and it can then capture any traffic going through that network.

So if your mobile device is on the same wifi network as your Wireshark machine's wifi card. Then you can capture the wifi traffic, filter it, and then monitor the HTTP traffic from your mobile device.

Why would I want to do that?

Because sometimes the mobile app you are testing does not honour the proxy settings of the device and goes direct, so you don't see the traffic.

And because you can start learning more about the network traffic layers being used by your application and your device in general.

(It's also fun to hook into hotel wifi and airport lounge wifi - but don't tell anyone.)

But the serious point, is that we know we want to observe the http traffic. If we have the issue that we can't because we can't configure the app to point to the proxy then we need other options. We need to increase our flexibility to approaching the observation. So we have a new option - work at the network traffic level, rather than proxy.

Our aim is to keep looking for new ways of achieving our outcomes. Not finding tools, for the sake of tools. But finding new approaches.

Installing Wireshark

On Windows

The Windows install is simple. Just download and run.

https://www.wireshark.org/download.html

On Mac

Mac install was a little harder for me and it didn't work out the box so I had to do the extra steps to add the application to XQuartz

- Install X-Windows - X11 XQuartz - http://xquartz.macosforge.org/landing/
- Download the img for wireshark
- install Wireshark
- Start wireshark - it might take a while, but should work

If it doesn't work then you could try, start xquartz In the Applications menu of Xquartz, customize it and "Add Item" with the command:

- "open /Applications/Wireshark.app/Contents/MacOs/Wireshark"
- or "open wireshark"
- or "wireshark"

Then you could try, running wireshark from the Applications menu in XQuartz, or from the application icon directly.

You might find these links helpful if you are on a mac:

- https://ask.wireshark.org/questions/12140/cant-run-wireshark-in-mac-os-x-mountain-lion

On Linux

I haven't tried the install on linux - I imagine the instructions on the Wireshark website work fine.

First Usage

Wireshark, can seem intimidating initially to work with.

It is a complicated tool and there is a lot to learn about it.

Start a Capture

On the main page, select your network card hooked to the wifi network. Then click "Capture Options".

In Capture options table. Check to see that "Mon. Mode" says enabled, for the interface you want to use. If it doesn't, you'll only see your own traffic.

To change "Mon. Mode", double click the item in the table, and choose "Capture packets in promiscuous mode" and "Capture packets in monitor

mode", press [OK].

Then [Start] the capture.

If you are on an encrypted network then you might need to decrypt the traffic.

http://wiki.wireshark.org/HowToDecrypt802.11

I sometimes have to fiddle with the IEEE 802.11 preferences: changing them, hitting apply, changing them, etc. Until I see the actual http traffic.

I also have to disconnect the android device from the network, and then reconnect it, so that it sends the initial network connection and decryption packets. Feel free to test on open networks where you don't have these type of issues because they are insecure if you want to.

Filter the capture

At this point your going to start seeing a lot of traffic flowing through your network.

So you want to filter it.

in the filter text editor type "ip.addr eq 192.168.1.143" or whatever the ip of your device is, to start seeing that traffic.

then if you just want to see the http traffic you can do

"ip.addr eq 192.168.1.143 and http"

or, to see just the GET requests:

"ip.addr eq 192.168.1.143 and http.request.method eq "GET""

This is a useful tool to have in your toolbox, for those moments where you have less control over the application under test, but still want to observe the traffic in your testing.