# 1、环境配置

## 1.1 下载与安装

### 1.1.1 网络抓包工具Fidder

下载地址 <https://www.telerik.com/download/fiddler>

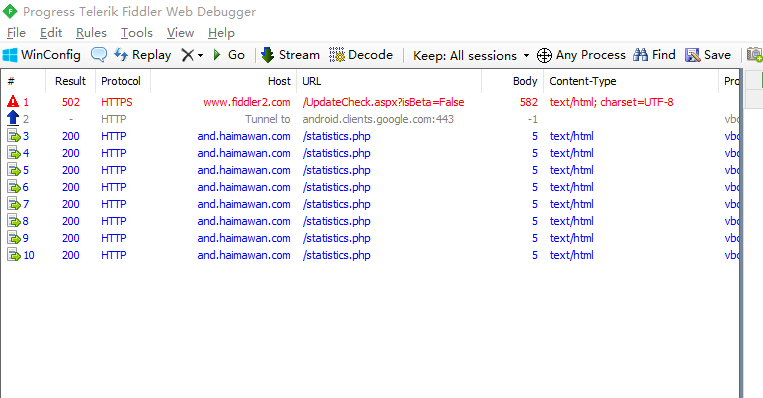


图 1 Fidder软件界面

### 1.1.2 海马玩安卓模拟器

下载地址 <http://droid4x.haimawan.com/>

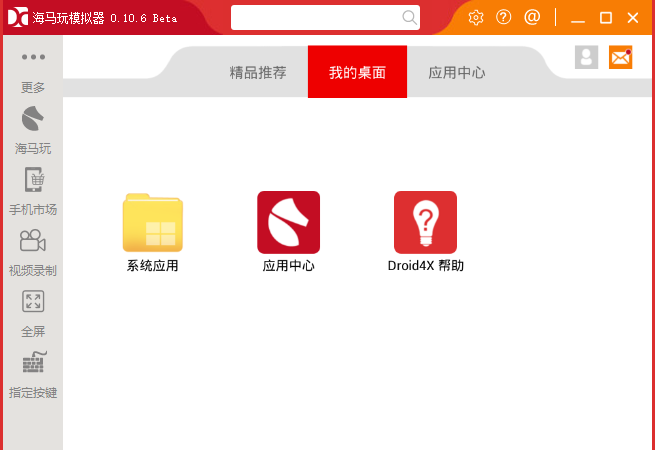


图 2 海马玩软件界面

### 1.1.3 数据源APP（春雨医生）

点击“应用中心”



搜索“春雨医生”，下载“春雨医生”官方版



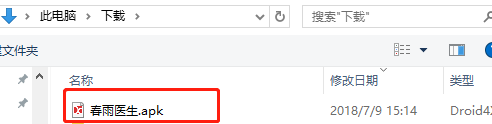
顺利下载完成后会自动安装。当模拟器中的网络不好时，也可以先从其他渠道下载APK文件，然后手动安装到模拟器中。



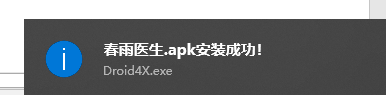
下载APK文件后，打开模拟器，依次点击“设置”、“其他设置”、“Apk安装”



选中APK文件进行安装







安装完毕后，可以再“我的桌面”看到APP

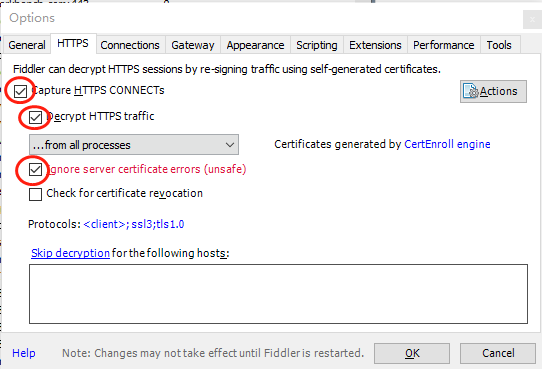


## 1.2 配置Fidder

### 1.2.1 设置Fidder请求监听规则

Tools -> Telerik Fiddler Options -> HTTPS选项卡

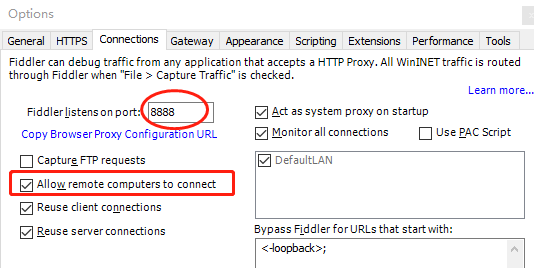
勾上Capture HTTPS CONNECTs和Decrypt HTTPS traffic这两项，建议勾上Ignore server certificate erros (unsafe)



Connections选项卡

勾上Allow remote computers to connect这项

记住Fiddler listens on port监听端口8888（等下会用到）



设置完必须重启Fiddler

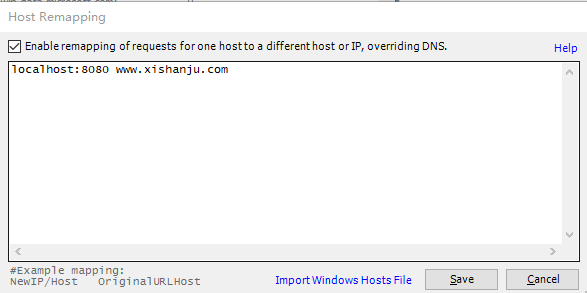
### 1.2.2 设置Fiddler请求转发规则

Tools -> HOSTS

勾上Enable remapping of requests for one host to a different host or IP, overriding DNS.

下面填写示例：NewIP/Host（去向） OriginalURLHost（来源）

这里相当于把来自www.xishanju.com的所有请求都转发到localhost:8080这里去，就达到我们本地调试的目的



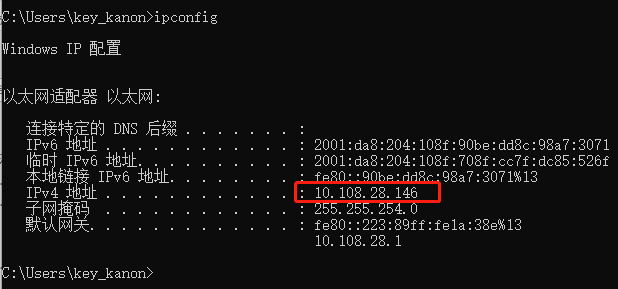
## 1.3 代理设置

### 1.3.1 查询主机内网IP

cmd -> ipconfig

记下我的内网IP是10.20.72.179（等下有用）

提示：电脑要和模拟器（手机）在同一个局域网内



### 1.3.2 设置模拟器网络代理

以海马玩浏览器为例，我的桌面 -> 系统应用 -> 设置 -> 无线和网络 -> Wi-Fi -> WiredSSID 长按 -> 修改网络

勾上显示高级选项，代理设置为手动，代理服务器主机名填刚才的内网IP，代理端口填写8888，其他保持默认。

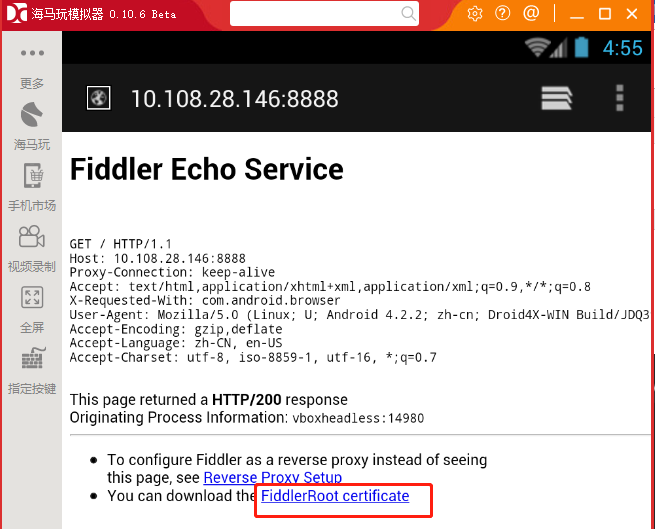






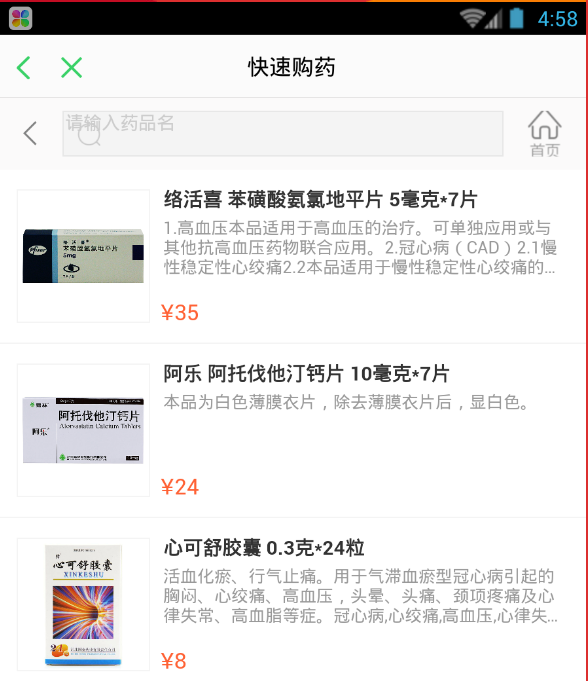
### 1.4 下载fiddler安全证书

在手机上打开浏览器输入一个上面ip地址和端口号组成的url，然后点击FiddlerRoot certificate下载fiddler证书



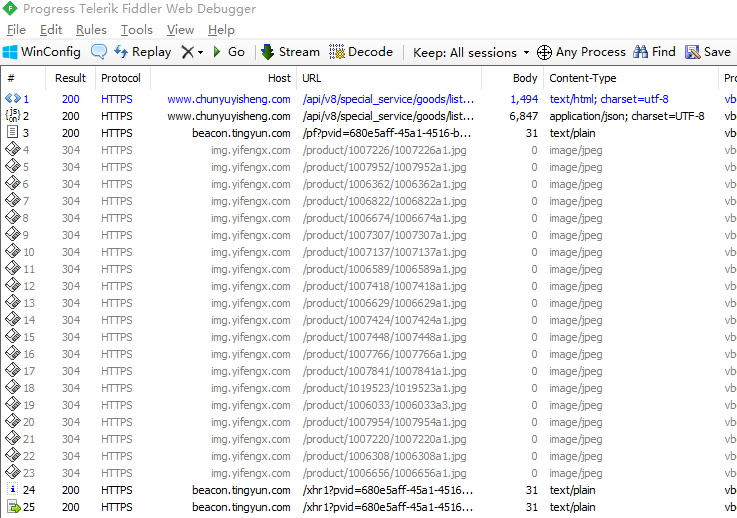
# 2、数据源识别

## 2.1 确定目标数据

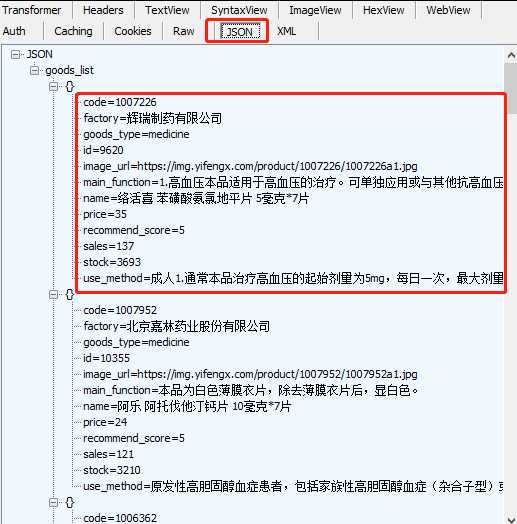


## 2.2 页面重载与监听数据传输

清空Fidder记录，然后重新进入目标数据所在的页面，便于在Fidder捕捉到的大量数据包中定位到目标数据



在载入新页面过程中，Fidder一共捕捉到25个数据包，其中有20张图片，可能为药物图片。我们需要的目标数据格式一般为json文件，且数据量相对较大，因此第2个数据包符合初步筛选条件，通过进一步查看可视化内容，最终确定为目标数据源。



## 2.3 目标URL获取

通过分析发现，当往下滑动时，页面每次会新载入20种药物，而我们需要至少50种，因此需要找到目标URL的规律，进而自动生成URL列表。

**1-20：**

https://www.chunyuyisheng.com/api/v8/special\_service/goods/list/?is\_json=1&**page=1**&goods\_category=%E5%BF%83%E8%A1%80%E7%AE%A1%E7%96%BE%E7%97%85&query=

**21-40：**

https://www.chunyuyisheng.com/api/v8/special\_service/goods/list/?is\_json=1&**page=2**&goods\_category=%E5%BF%83%E8%A1%80%E7%AE%A1%E7%96%BE%E7%97%85&query=

**41-60：**

https://www.chunyuyisheng.com/api/v8/special\_service/goods/list/?is\_json=1&**page=3**&goods\_category=%E5%BF%83%E8%A1%80%E7%AE%A1%E7%96%BE%E7%97%85&query=

可以发现在URL中，只有“page=”部分发生改变，因此可以通过修改数值来获取更多的数据。

# 3、文档解析

以前20种药物的信息为例，展示文档解析与数据抽取过程，若需要更多的数据，可以编写循环程序进行数据抓取，然后再将所获得的数据进行合并。

## 3.1 文档解析

R代码：

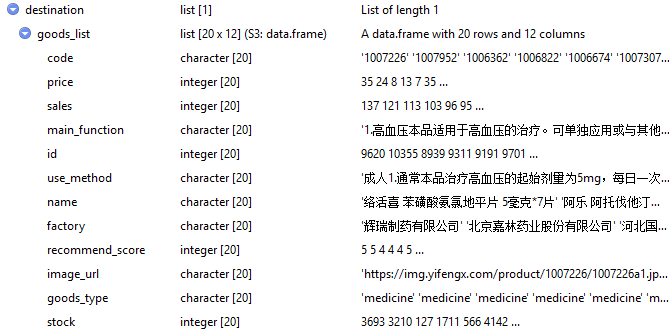
library(jsonlite)

url\_1 <- "https://www.chunyuyisheng.com/api/v8/special\_servi

ce/goods/list/?is\_json=1&page=1&goods\_category=%E5%BF%83%E8%

A1%80%E7%AE%A1%E7%96%BE%E7%97%85&query="

destination <- fromJSON(url\_1)



## 3.2 数据抽取

从goodslist数据框中选择所需要的数据，在这里我们选择price、sales、name、recommend\_score数据项。

R代码：

goods <- destination$goods\_list

goods[,c("name","recommend\_score","price","sales")]



## 3.3 数据输出

将目标数据输出为CSV文件进行保存

R代码：

data <- goods[,c("name","recommend\_score","price","sales")]

write.csv(data,"data\_goods.csv")

