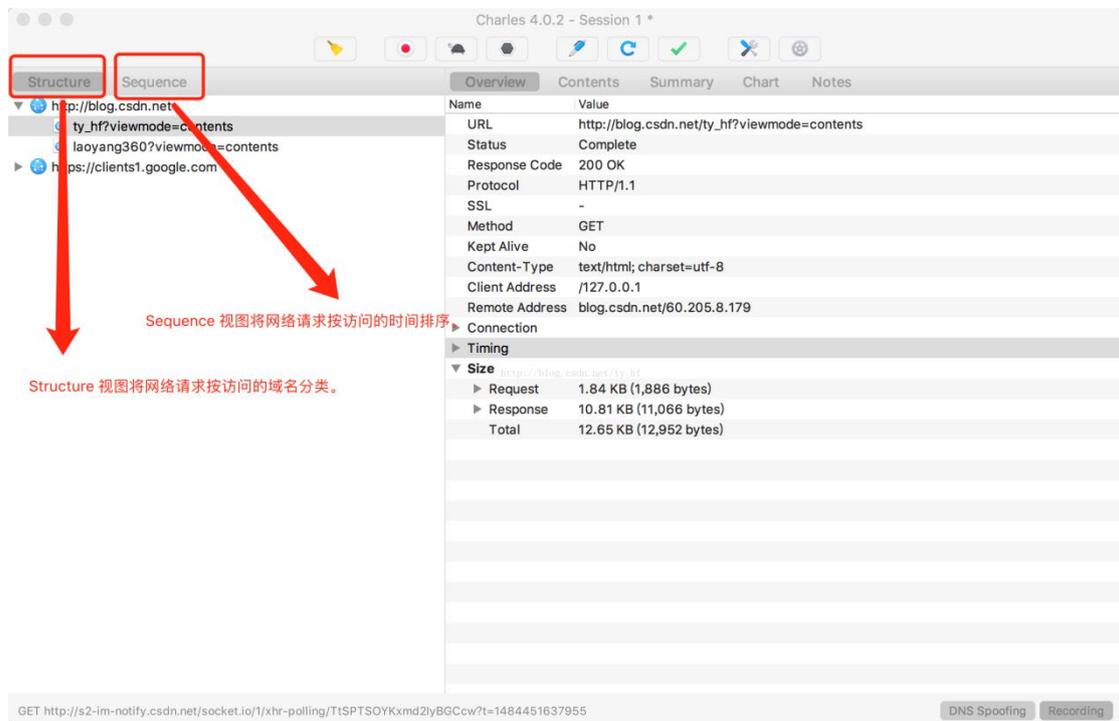


Charles 常用的功能总结

- 1.过滤网络请求
- 2.截取 iPhone 上的网络封包
- 3.截取 Https 通讯信息
- 4.模拟慢速网络
- 5.修改网络请求内容
- 6.给服务器做压力测试
- 7.修改服务器返回内容
- 8.Map 重定向功能
- 9.Rewrite 功能
- 10.Breakpoints 功能
- 11.设置 ssl 允许访问地址
- 12.查看连接本机 charle 代理的 ip
- 13.设置请求的黑名单
- 14.缓存请求返回的内容

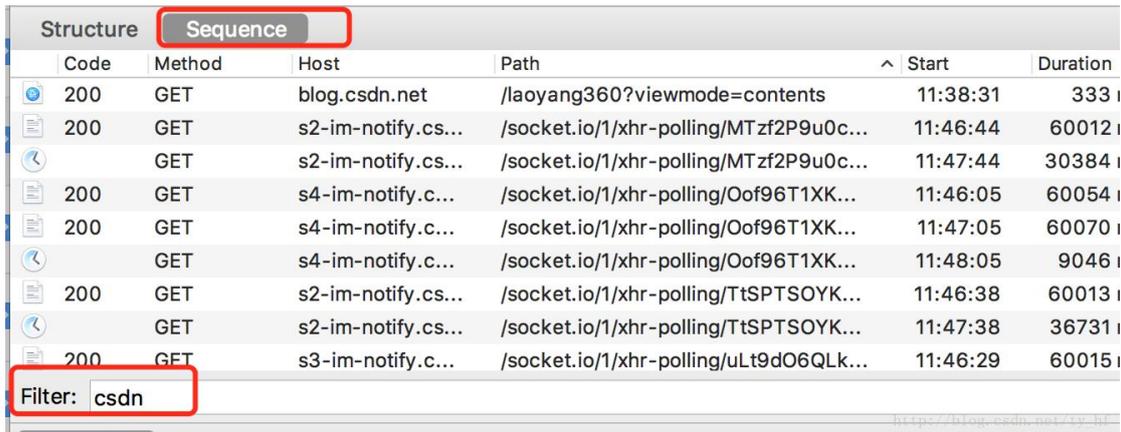


对于某一个具体的网络请求，你可以查看其详细的请求内容和响应内容。如果请求内容是 POST 的表单，Charles 会自动帮你将表单进行分项显示。如果响应内容是 JSON 格式的，那么 Charles 可以自动帮你将 JSON 内容格式化，方便你查看。如果响应内容是图片，那么 Charles 可以显示出图片的预览。

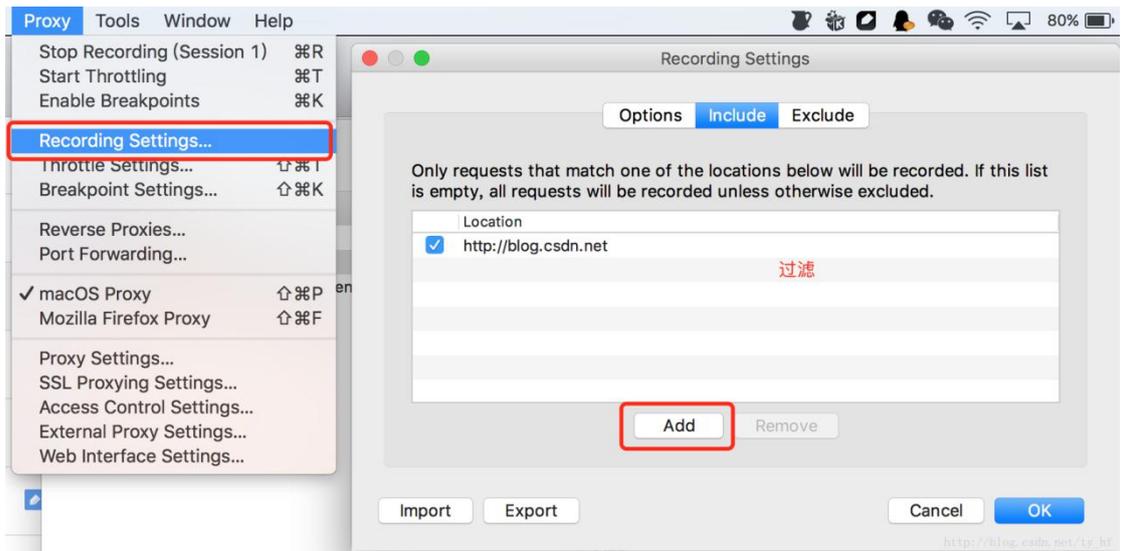
1.过滤网络请求

通常情况下，我们需要对网络请求进行过滤，只监控向指定目录服务器上发送的请求。对于这种需求，以下几种办法：

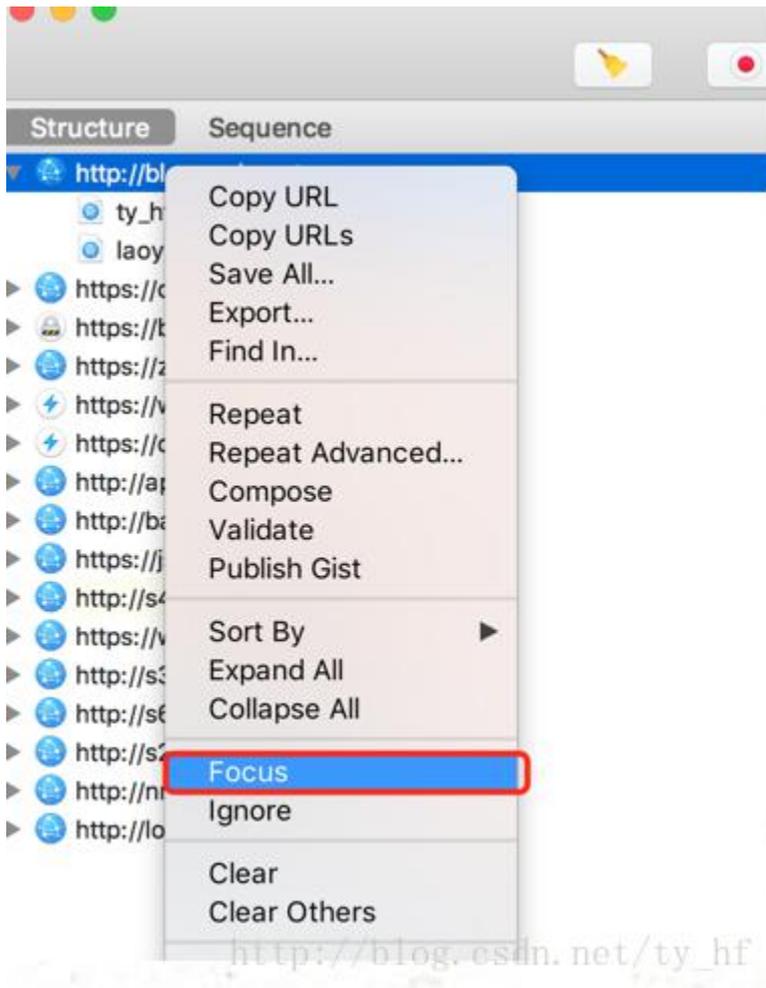
方法一：在主界面的中部的 Filter 栏中填入需要过滤出来的关键字。例如我们的服务器的地址是：http://blog.csdn.net, 那么只需要在 Filter 栏中填入 csdn 即可。



方法二：在 Charles 的菜单栏选择 “Proxy” -> ”Recording Settings”，然后选择 Include 栏，选择添加一个项目，然后填入需要监控的协议，主机地址，端口号。这样就可以只截取目标网站的封包了。如下图所示：



方法三：在想过滤的网络请求上右击，选择 “Focus”，之后在 Filter 一栏勾选上 Focussed 一项，如下图所示：



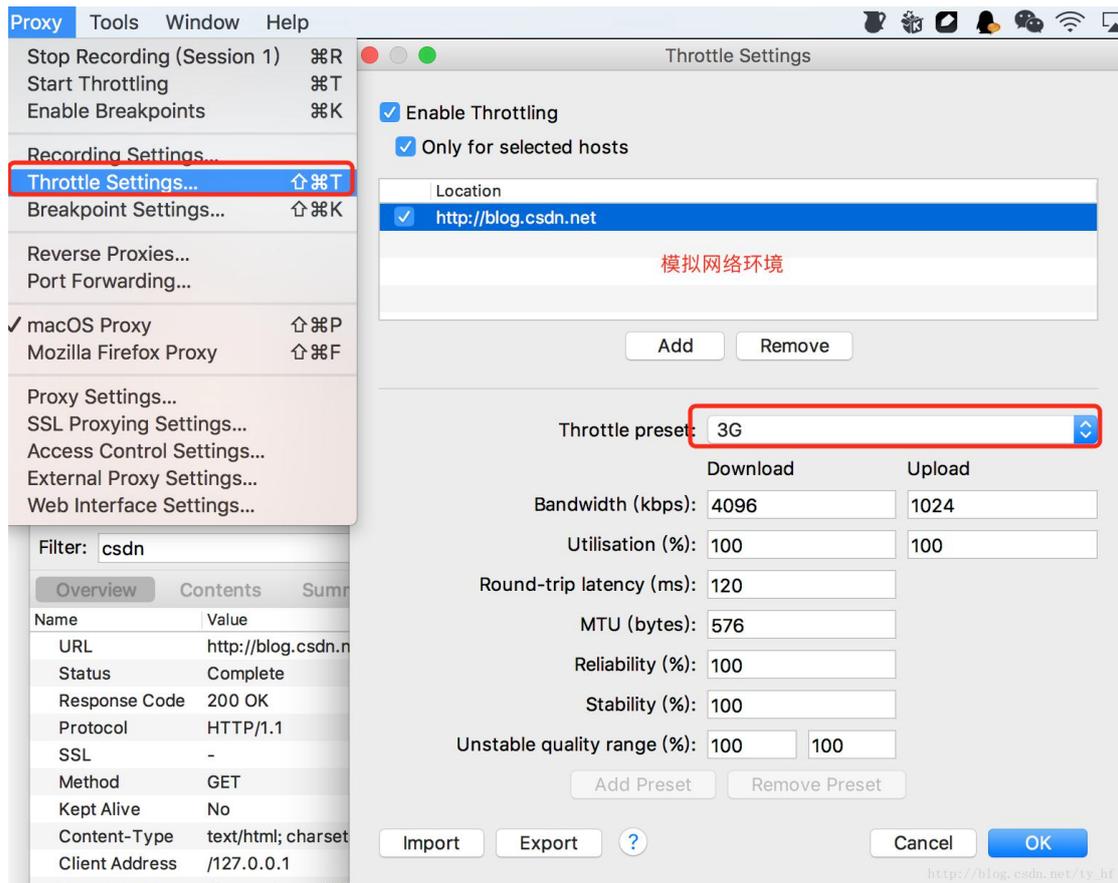
2. 截取 iPhone 上的网络封包
charles 如何抓取 http/https 请求

3. 截取 Https 通讯信息
charles 如何抓取 http/https 请求

4. 模拟慢速网络

在做移动开发的时候,我们常常需要模拟慢速网络或者高延迟的网络,以测试在移动网络下,应用的表现是否正常。Charles 对此需求提供了很好的支持。

在 Charles 的菜单上,选择“Proxy”->“Throttle Setting”项,在之后弹出的对话框中,我们可以勾选上“Enable Throttling”,并且可以设置 Throttle Preset 的类型。如下图所示:

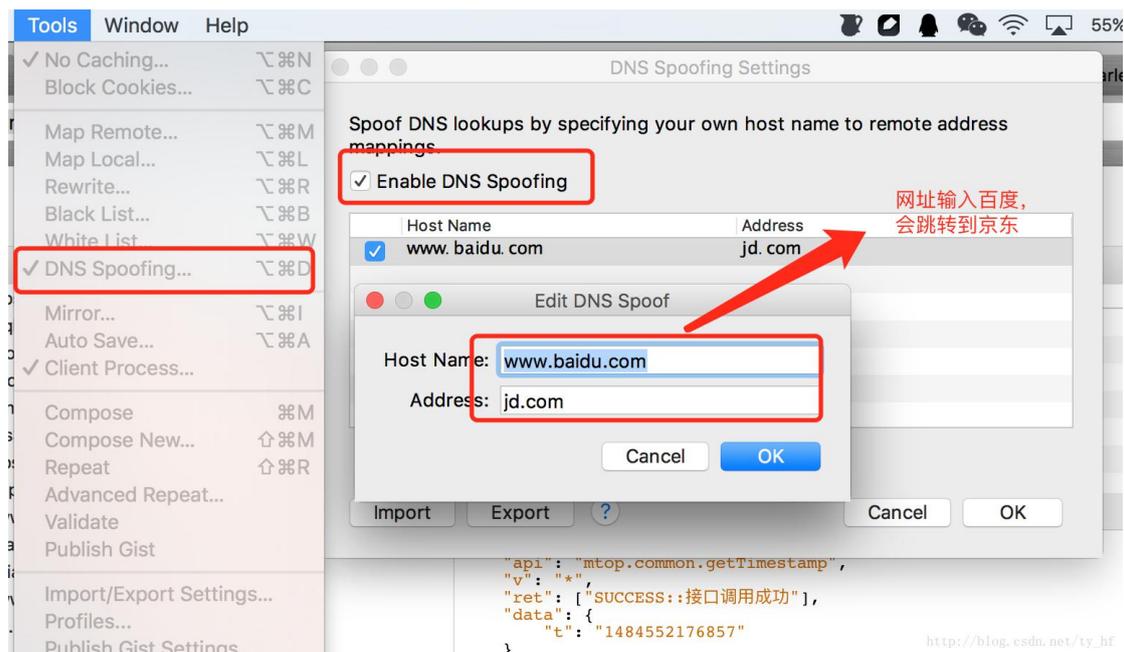


5. 修改网络请求地址

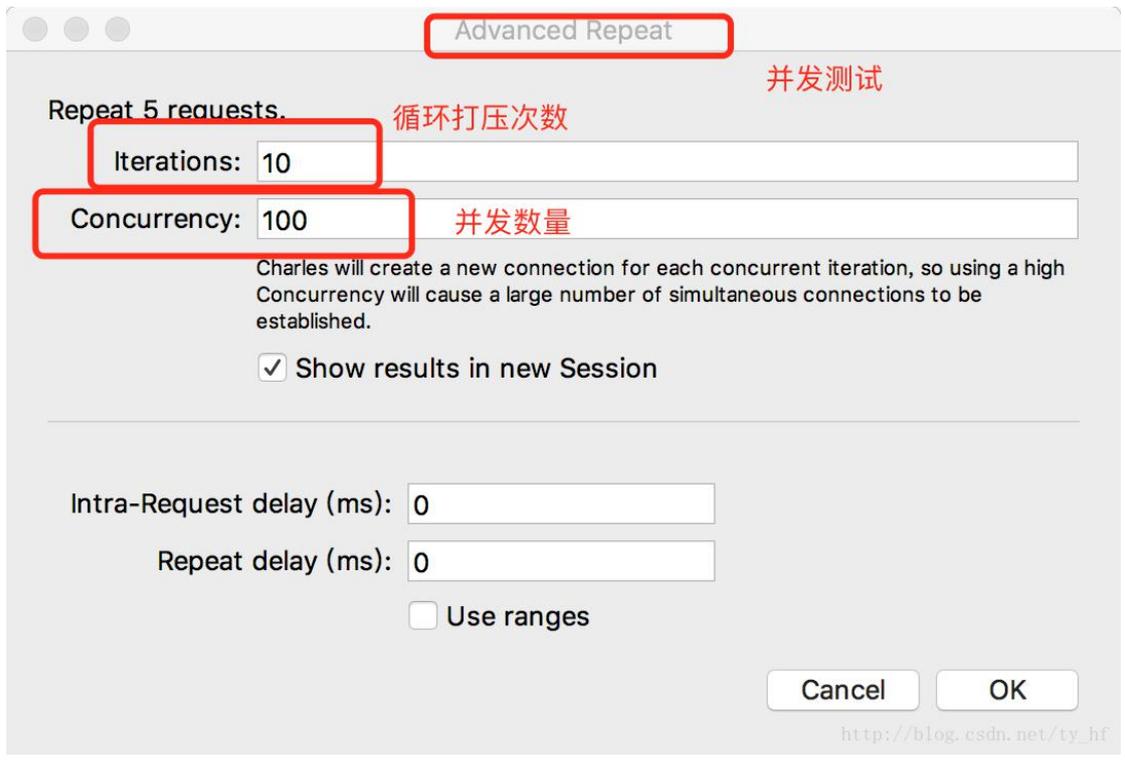
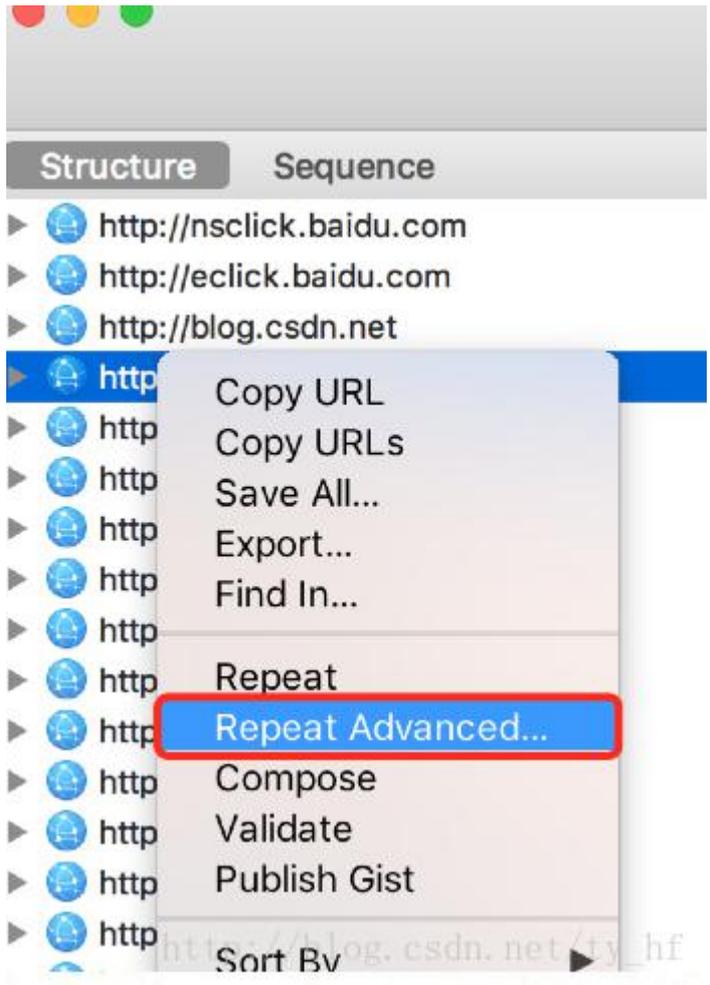
dns 欺骗，说简单点就是把域名解析到一个假的 ip，

可以不必一定要用 localhost,127.0.0.1,装个 B 把百度对应到京东来调试~

选择 Tools->DNS Spoofing，设置如下图：



6.给服务器做压力测试



7.修改服务器返回内容

有些时候我们想让服务器返回一些指定的内容，方便我们调试一些特殊情况。例如列表页面为空的情况，数据异常的情况，部分耗时的网络请求超时的情况等。如果没有 Charles，要服务器配合构造相应的数据显得会比较麻烦。这个时候，使用 Charles 相关的功能就可以满足我们的需求。

根据具体的需求，Charles 提供了 Map 功能、Rewrite 功能以及 Breakpoints 功能，都可以达到修改服务器返回内容的目的。这三者在功能上的差异是：

△Map 功能适合长期地将某一些请求重定向到另一个网络地址或本地文件。

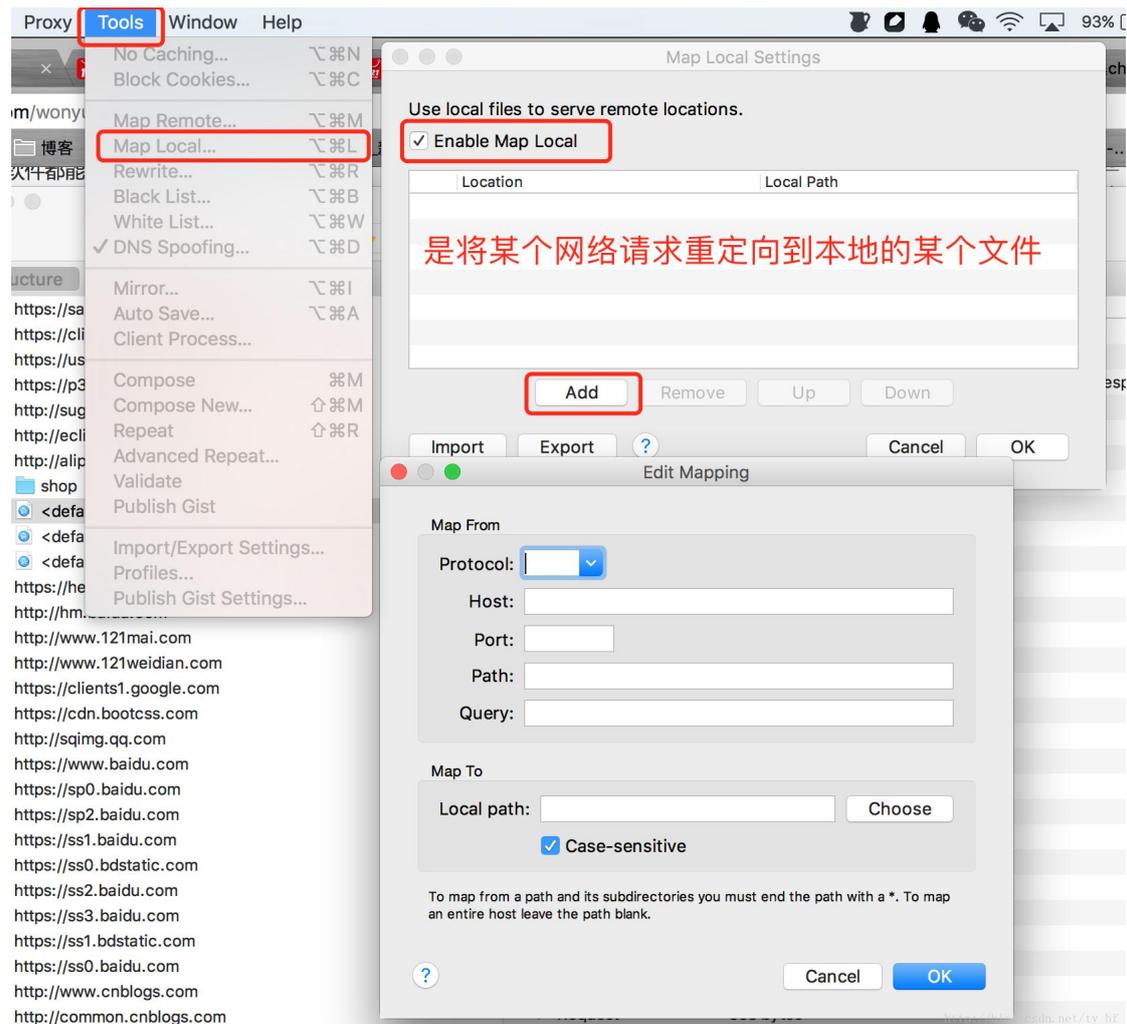
△Rewrite 功能适合对网络请求进行一些正则替换。

△Breakpoints 功能适合做一些临时性的修改。

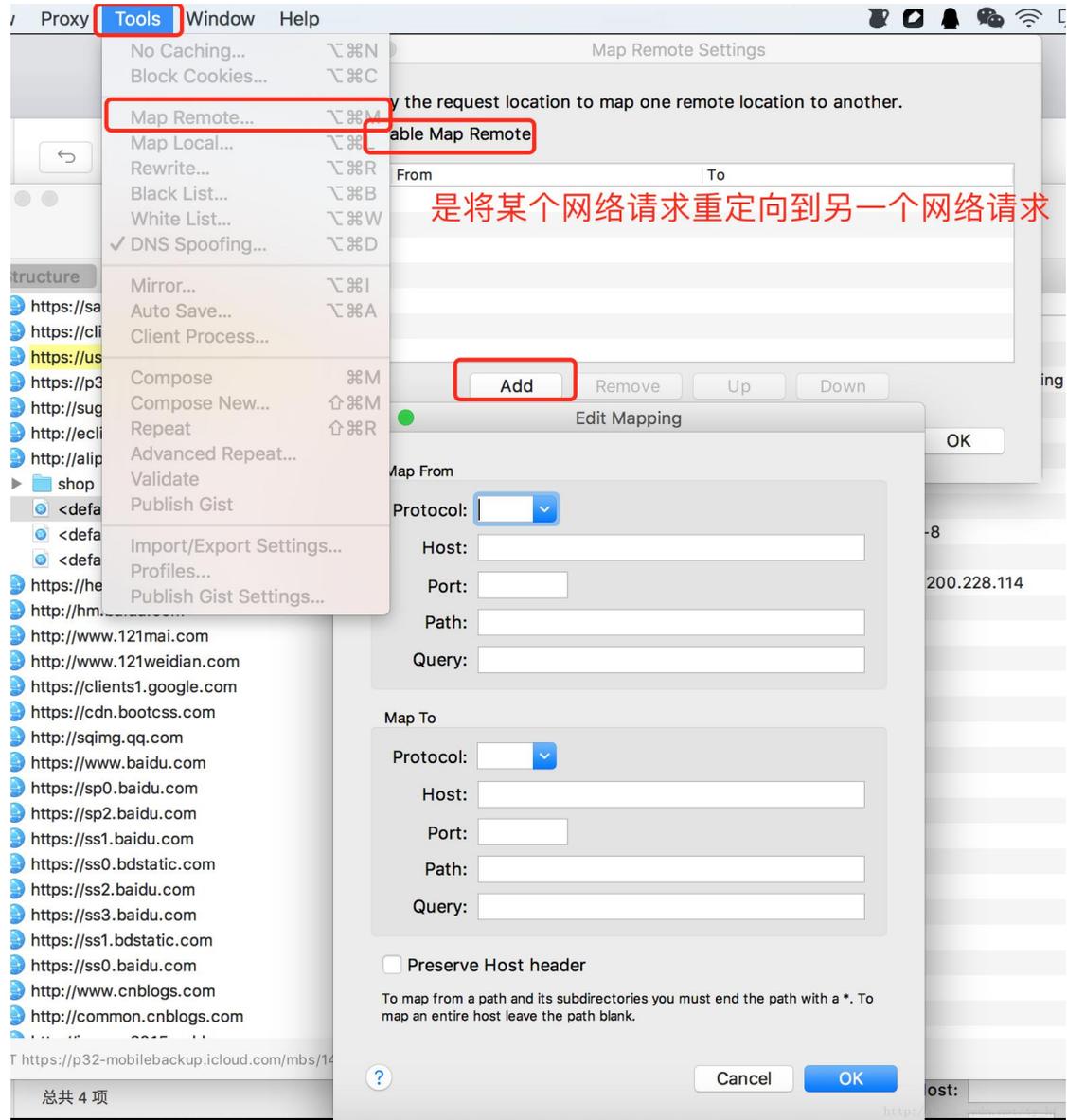
8.Map 重定向功能

Charles 的 Map 功能分 Map Remote 和 Map Local 两种，顾名思义，Map Remote 是将指定的网络请求重定向到另一个网址请求地址，Map Local 是将指定的网络请求重定向到本地文件。

在 Charles 的菜单中，选择 “Tools” ->” Map Remote” 或 “Map Local” 即可进入到相应功能的设置页面。

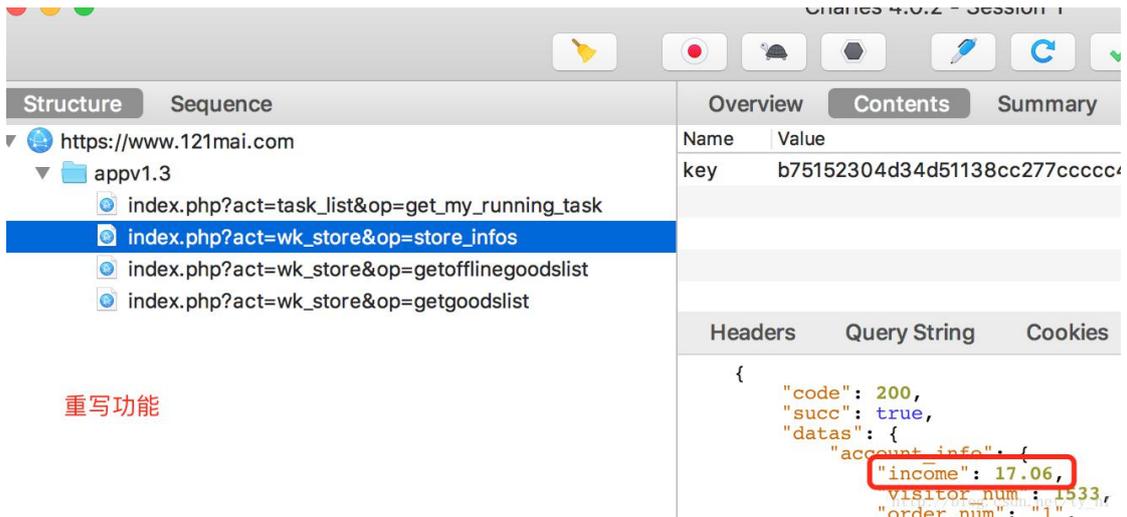


△对于 Map Local 功能，我们需要填写的重定向的源地址和本地的目标文件。对于有一些复杂的网络请求结果，我们可以先使用 Charles 提供的 “Save Response...” 功能，将请求结果保存到本地（如下图所示），然后稍加修改，成为我们的目标映射文件。

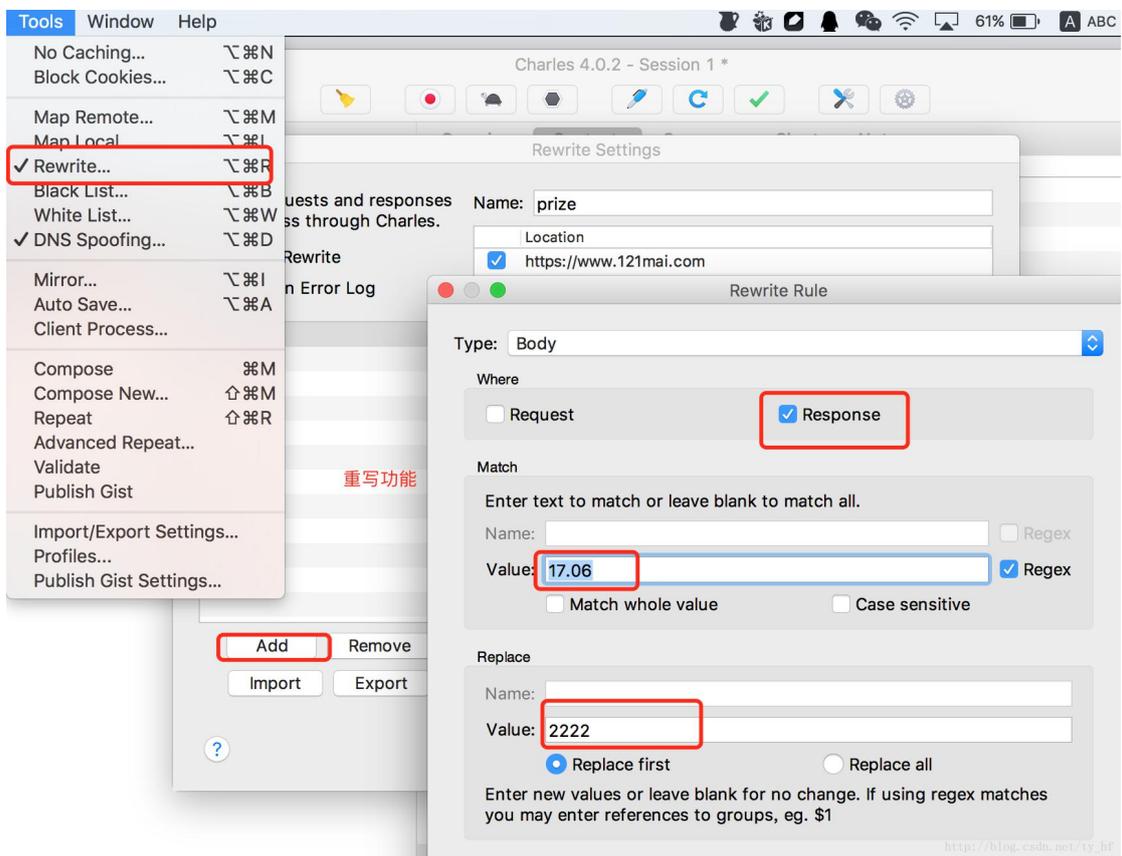


9.Rewrite 功能

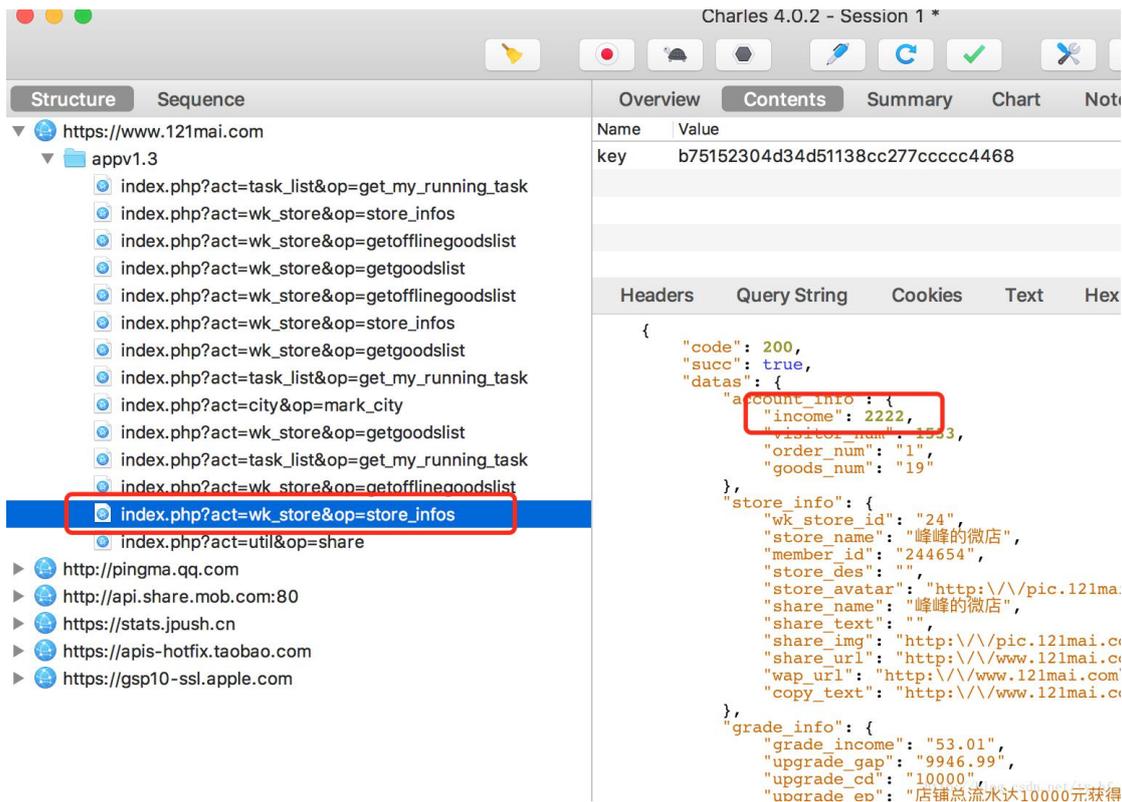
△Rewrite 功能功能适合对某一类网络请求进行一些正则替换，以达到修改结果的目的。例如，我们的客户端有一个 API 请求是获得用户余额，而我当前的余额是 “17.06”，如下所示：



重写设置:



再次请求，余额变成了 2222,重写成功。

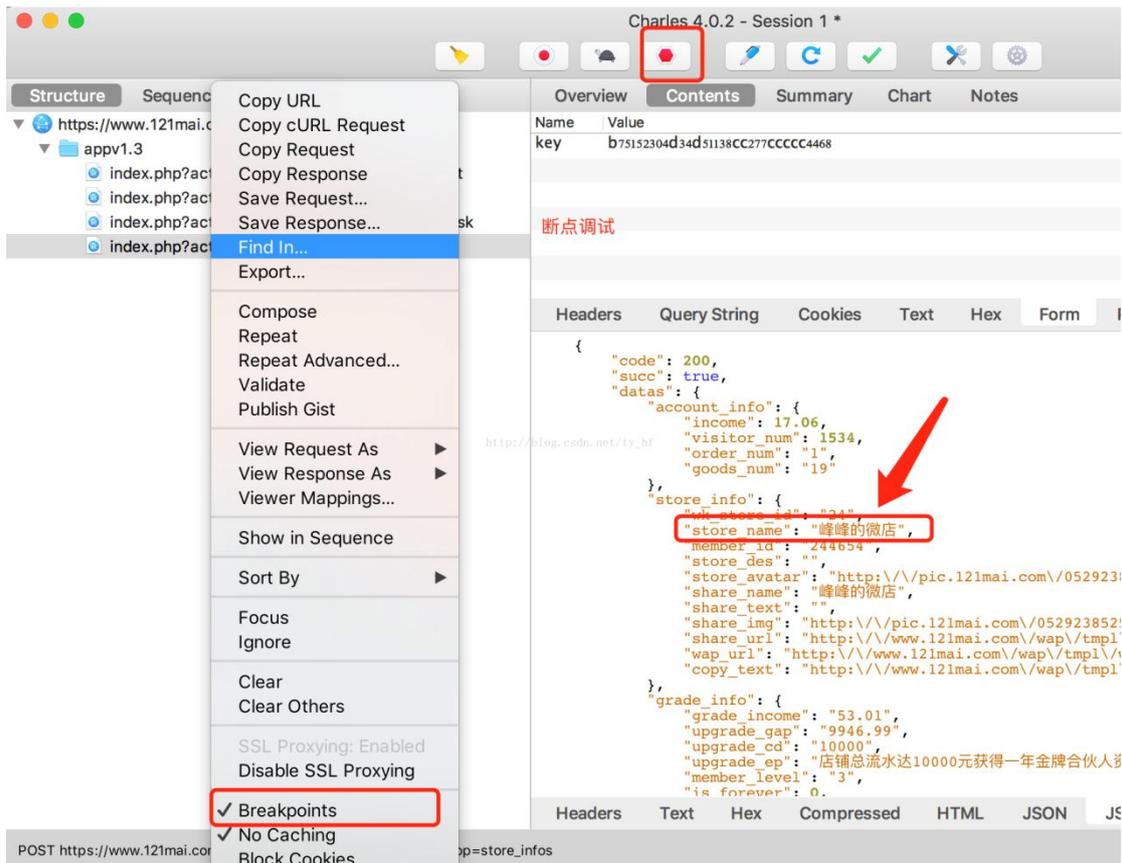


10. Breakpoints 功能

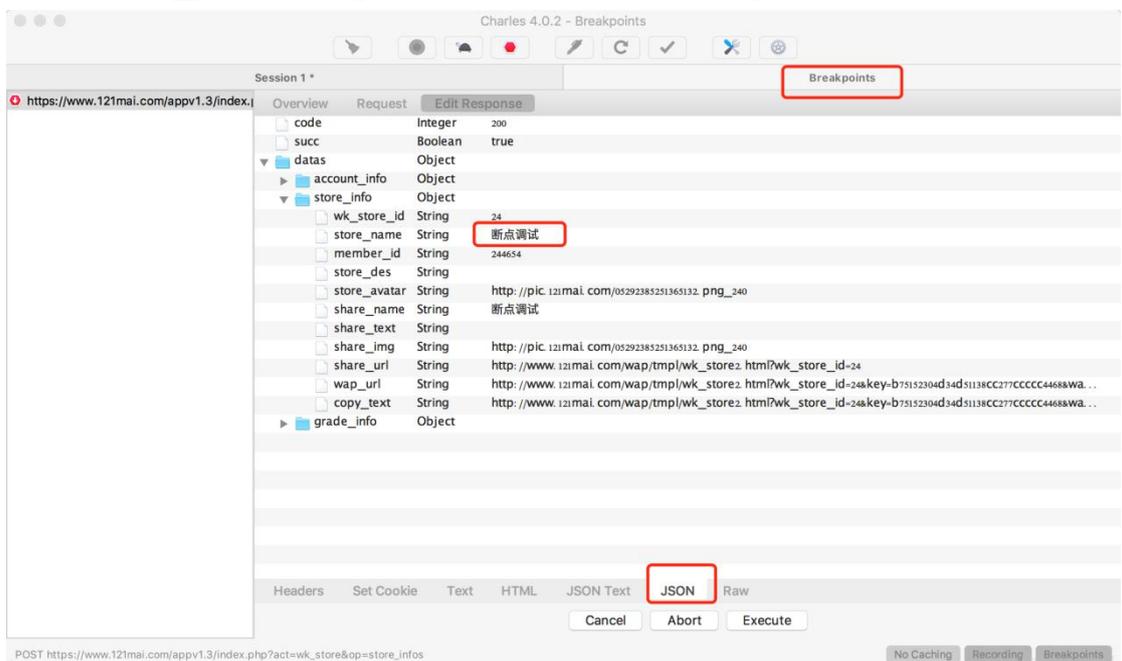
上面提供的 Rewrite 功能最适合做批量和长期的替换，但是很多时候，我们只是想临时修改一次网络请求结果，这个时候，使用 Rewrite 功能虽然也可以达到目的，但是过于麻烦，对于临时性的修改，我们最好使用 Breakpoints 功能。

Breakpoints 功能类似我们在 Xcode 中设置的断点一样，当指定的网络请求发生时，Charles 会截获该请求，这个时候，我们可以在 Charles 中临时修改网络请求的返回内容。

下图是我们临时修改获取用户信息的 API，将用户的昵称进行了更改，修改完成后点击“Execute” 则可以让网络请求继续进行。



按照上边设置好以后，再次访问接口将会有断点提示，进入编辑返回的内容即可（我在运用时很奇怪，设置成功后，得退出一次 charles 程序，再启动才会有断点界面）



http://cgi.connect.qq.com
 https://www.121mai.com
 appv1.3
 index.php?act=util&op=judge_version
 index.php?act=wk_store&op=store_infos
 index.php?act=index&op=search_key_list
 index.php?act=wk_store&op=getofflinegoodslist
 index.php?act=task_list&op=get_my_running_task
 index.php?act=wk_store&op=getgoodslist
 index.php?act=city&op=getCityId
 index.php?act=city&op=actinfo_by_cityId
 index.php?act=city&op=location
 index.php?act=city&op=mark_city
 https://api.map.baidu.com
 https://apis-hotfix.taobao.com
 http://api.exc.mob.com
 http://m.data.mob.com

```

{
  "code": 200,
  "succ": true,
  "datas": {
    "account_info": {
      "income": 17.06,
      "visitor_num": 1534,
      "order_num": 1,
      "goods_num": 19
    },
    "store_info": {
      "wk_store_id": "24",
      "store_name": "断点调试",
      "member_id": 244634,
      "store_des": "",
      "store_avatar": "http://pic.121mai.com/0529238525",
      "share_name": "断点调试",
      "share_text": "",
      "share_img": "http://pic.121mai.com/0529238525136",
      "share_url": "http://www.121mai.com/wap/tmpl/wk_s",
      "wap_url": "http://www.121mai.com/wap/tmpl/wk_s",
      "copy_text": "http://www.121mai.com/wap/tmpl/wk_s"
    }
  }
}
  
```

断点调试成功

11. 设置 ssl 允许访问地址

Proxy Tools Window Help

Stop Recording (Session 1) ⌘R
 Start Throttling ⌘T
 Enable Breakpoints ⌘K

Recording Settings...
 Throttle Settings... ⌘T
 Breakpoint Settings... ⌘K

Reverse Proxies...
 Port Forwarding...

macOS Proxy ⌘P
 Mozilla Firefox Proxy ⌘F

Proxy Settings...
 SSL Proxying Settings...
 Access Control Settings...
 External Proxy Settings...
 Web Interface Settings...

users
 <unknown>
 https://adashbc.ut.taobao.com
 http://api.weibo.cn
 https://configuration.apple.com
 https://p32-mobilebackup.icloud.com
 https://p32-content.icloud.com
 https://us-std-00001.s3.amazonaws.com

Charles 4.0.2 - Session 1 *

SSL Proxying Settings

SSL Proxying Client Certificates Root Certificate

Charles can show you the plain text contents of SSL requests and responses. Only the locations listed below will be proxied. Charles will issue and sign SSL certificates, please press the Help button for more information.

Enable SSL Proxying

Location

- *.121mai.com:443
- api.weibo.cn:443
- *.m.jd.com:443
- *.com:443
- *.cn:443

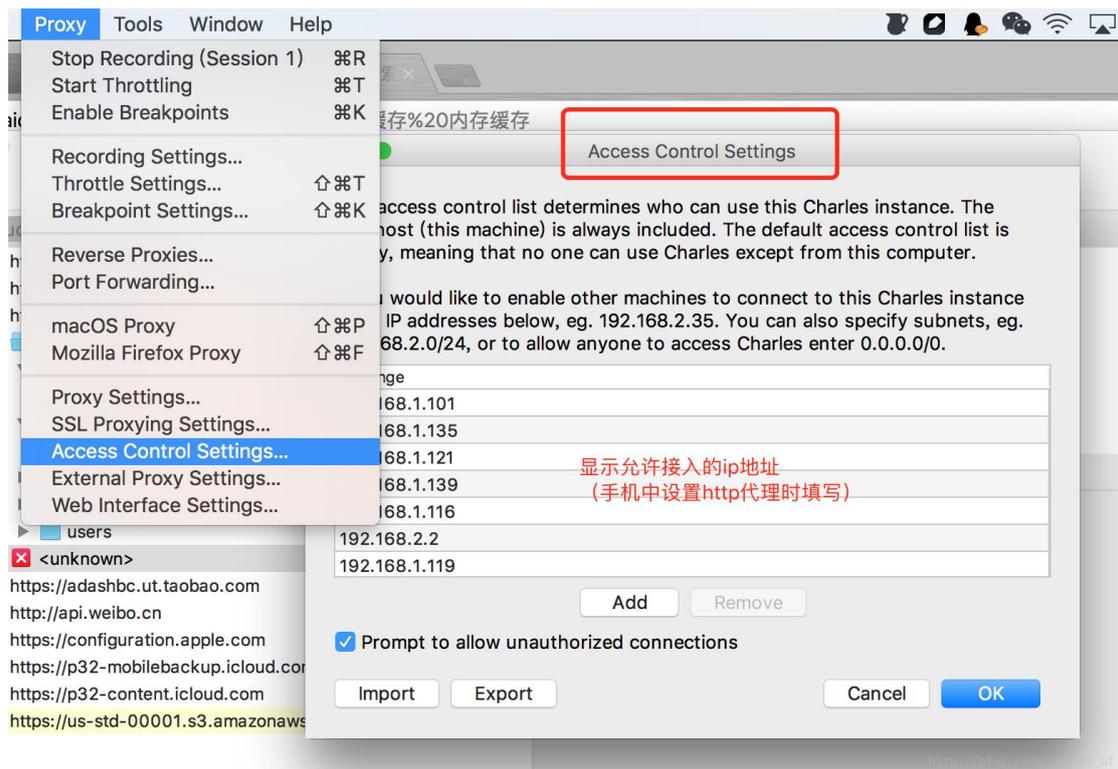
Add Remove

Cancel OK

https访问可以显示的域名

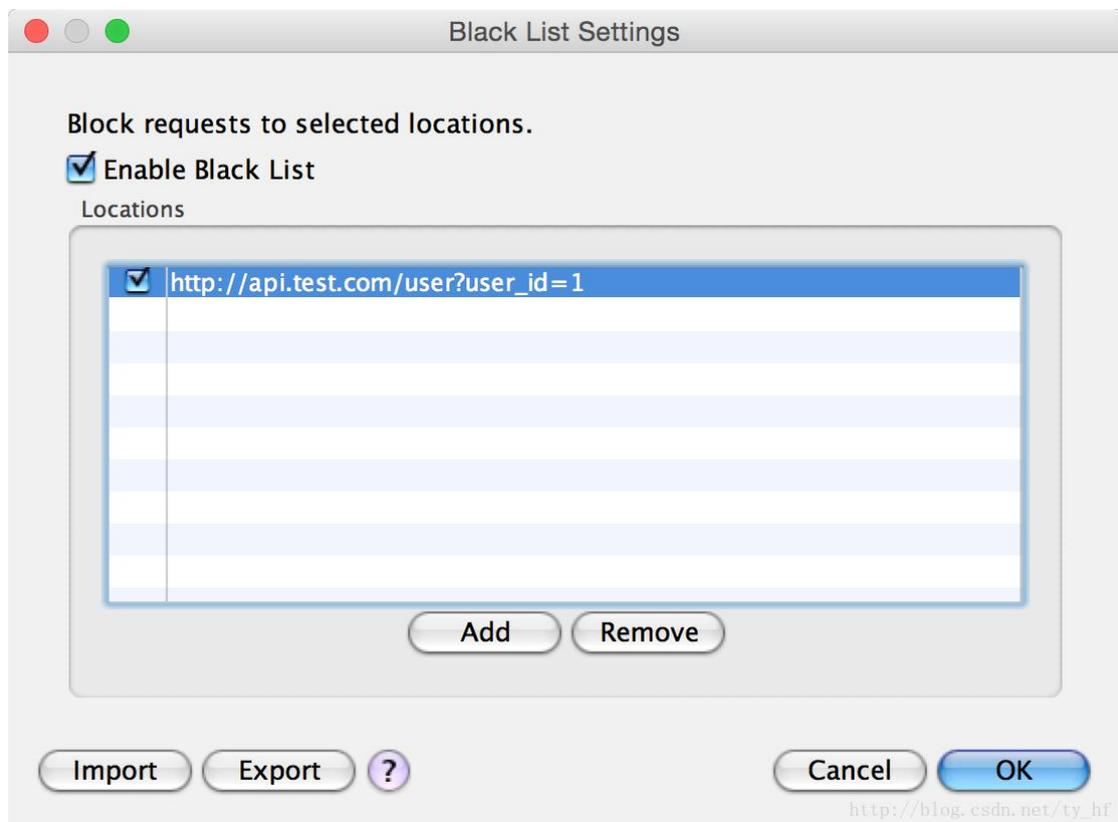
有他俩能允许大部分

12. 看连接本机 charle 代理的 ip 有非法的 remove 掉即可



13. 设置请求的黑名单

不想某些请求发起，直接返回 404，可以用黑名单
选择 Tools->Black List，设置如下图：



14.缓存请求返回的内容

这个我用来干过做缓存数据用，让 app 在没有 server 的时候也能跑，选择 Tools->Mirror，设置如下图：

