Fiddler抓包

2018.06

Fiddler工具介绍

HTTP协议

Fiddler的使用

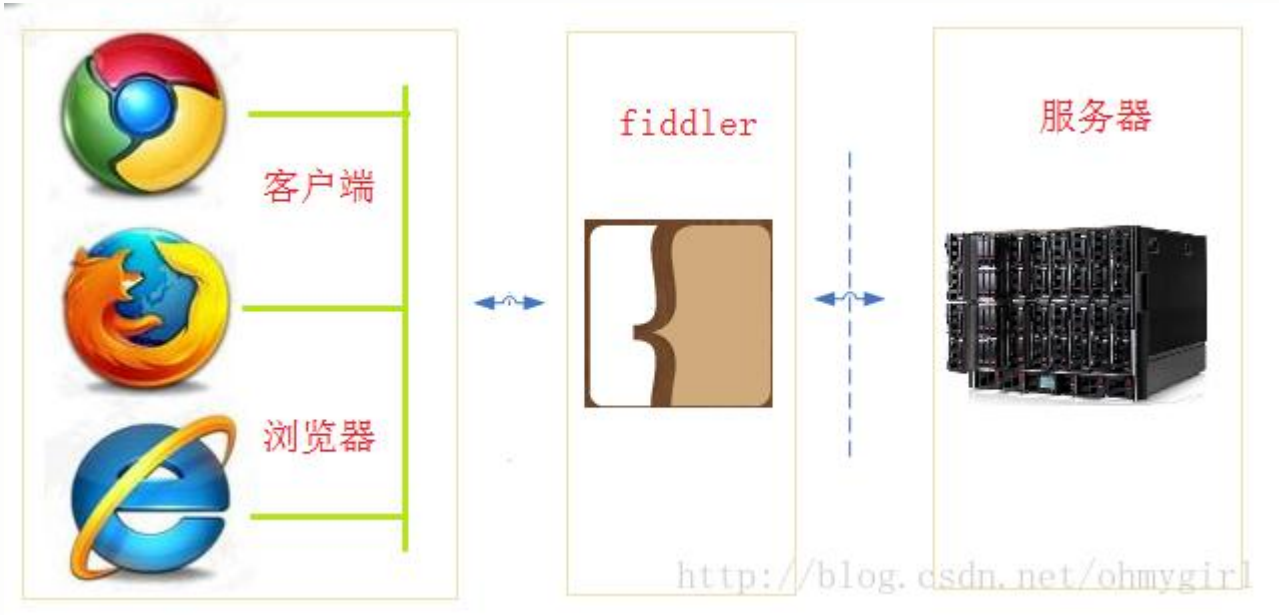
Fiddler图标含义

Fiddler的日志如何解析

一、Fiddler工具介绍

- **Fiddler是强大的抓包工具，它的原理是以web代理服务器的形式进行工作的，使用的代理地址是：127.0.0.1，端口默认为8888，我们也可以通过设置进行修改。**
- **代理就是在客户端和服务器之间设置一道关卡，客户端先将请求数据发送出去后，代理服务器会将数据包进行拦截，代理服务器再冒充客户端发送数据到服务器；同理，服务器将响应数据返回，代理服务器也会将数据拦截，再返回给客户端。**
- **Fiddler可以抓取支持http代理的任意程序的数据包，如果要抓取https会话，要先安装证书。**

使用了Fiddler之后web客户端和服务器的请求如右图所示：



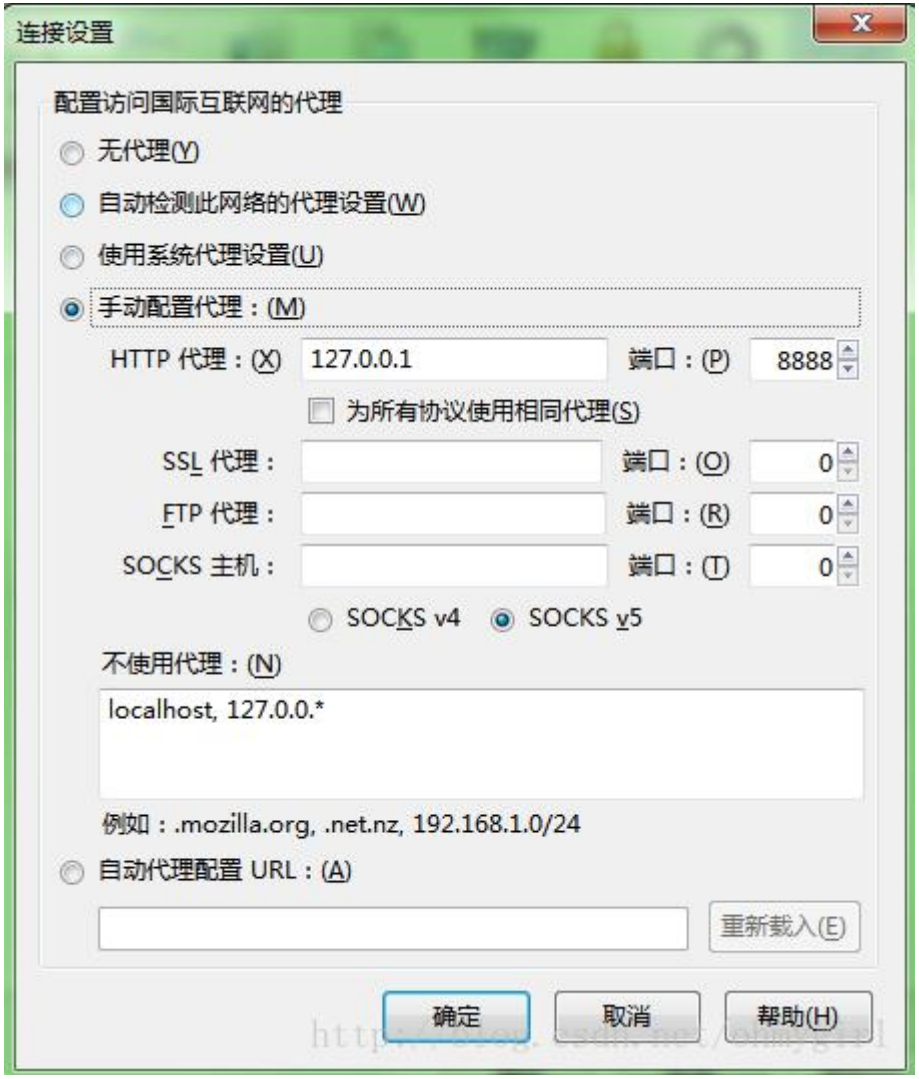
浏览器代理设置

- Fiddler 作为系统代理，当启用 Fiddler 时，IE 的PROXY 设定会变成 127.0.0.1:8888，因此如果你的浏览器在开启fiddler之后没有设置相应的代理，则fiddler是无法捕获到HTTP请求的。

- 如下是启动Fiddler之后，IE浏览器的代理设置：

浏览器代理设置

- 以Firefox为例，默认情况下，firefox是没有启用代理的
  在firefox中配置http代理的步骤
  工具->选项->高级->网络->设置。并配置相应的代理如下



Fiddler工具介绍

HTTP协议

Fiddler的使用

Fiddler图标含义

Fiddler的日志如何解析

二、HTTP协议

1.分析Fiddler抓取的数据包，我们首先要熟悉HTTP协议。
HTTP即超文本传输协议，是一个基于请求与响应模式的、无状态的、应用层的协议，绝大多数的Web开发，都是构建在HTTP协议之上的Web应用。

2.HTTP的工作过程：当我们请求一个超链接时，HTTP就开始工作了，客户端先发送一个请求到服务器，请求内容包括：协议版本号、请求地址、请求方式、请求头和请求参数；服务器收到请求后做相应的处理，并将响应数据返回到客户端，响应内容包括：协议版本号、状态码和响应数据。前端根据响应数据做相应的处理，就是最终我们看到的内容。这些过程是HTTP自动完成的，我们只是输入或点击请求地址，然后查看前端给我们展示的内容。更多关于HTTP协议的介绍请参考：
http://www.cnblogs.com/li0803/archive/2008/11/03/1324746.html

3.请求方式常用的有：GET、PUT、POST、DELETE。

HTTP状态码

1.HTTP状态码主要分为5类：以1开头的代表请求已被接受，需要继续处理；以2开头的代表请求已成功被服务器接收、理解、并接受；以3开头的代表需要客户端采取进一步的操作才能完成请求；以4开头的代表了客户端看起来可能发生了错误，妨碍了服务器的处理；以5开头的代表了服务器在处理请求的过程中有错误或者异常状态发生，也有可能是服务器意识到以当前的软硬件资源无法完成对请求的处理。

2.常见的主要有：200：服务器成功处理了请求；404：未找到资源；500：内部服务器错误；503：服务器目前无法为请求提供服务；302：请求的URL已临时转移；304：客户端的缓存资源是最新的，要客户端使用缓存。
每个状态码的详细介绍请参考：
https://baike.baidu.com/item/HTTP%E7%8A%B6%E6%80%81%E7%A0%81/5053660?fr=aladdin

Fiddler工具介绍

HTTP协议

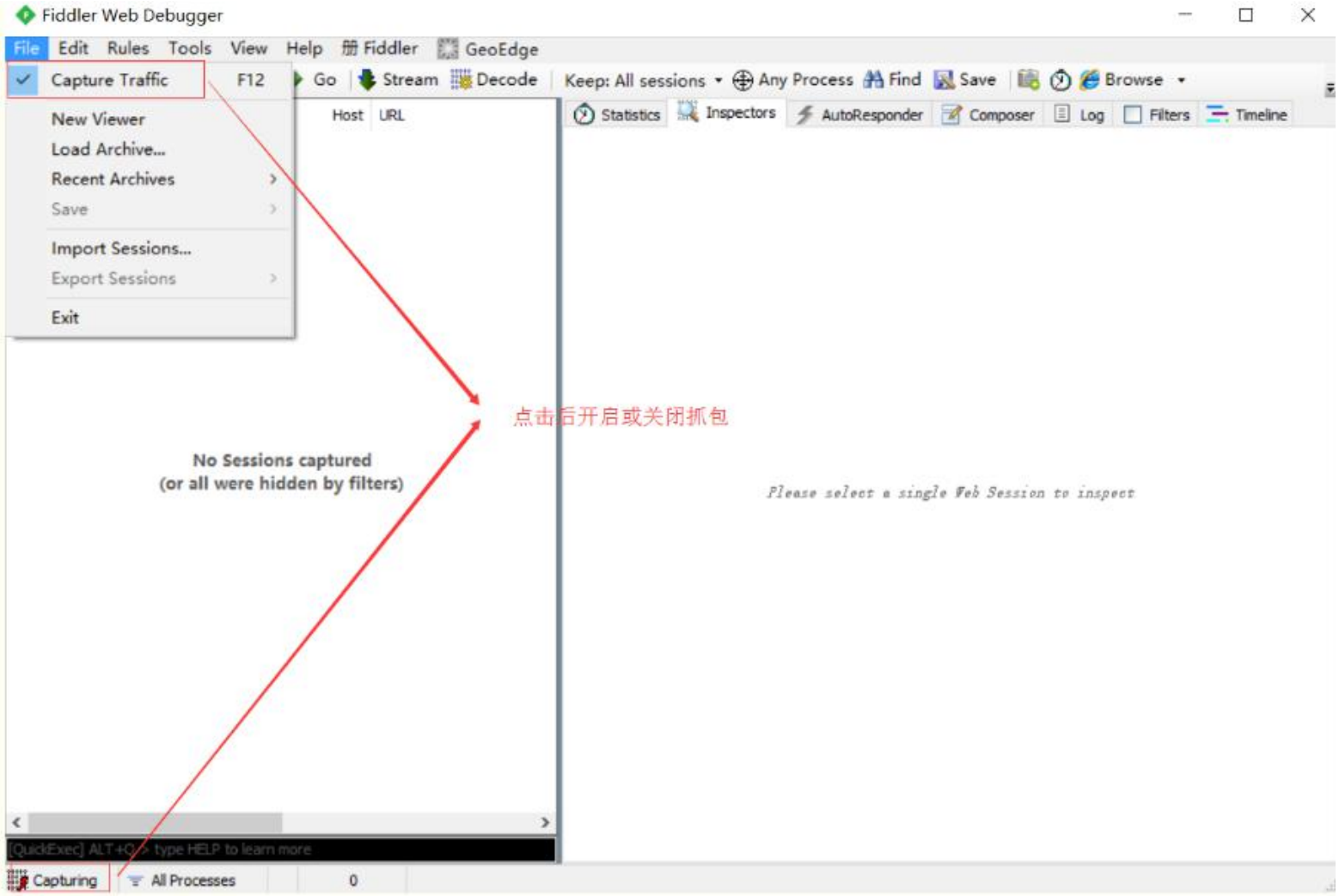
Fiddler的使用

Fiddler图标含义

Fiddler的日志如何解析

三、Fiddler的使用

3.1 要使用Fiddler进行抓包，首先需要确保Capture Traffic是开启的（安装后是默认开启的），勾选File->Capture Traffic，也可以直接点击Fiddler界面左下角的图标开启和关闭抓包。



Fiddler工具介绍

HTTP协议

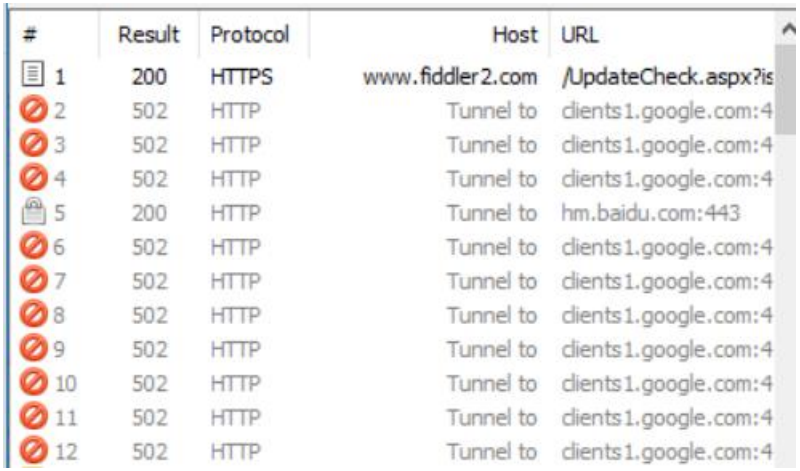
Fiddler的使用

Fiddler图标含义

Fiddler的日志如何解析

四、Fiddler图标含义

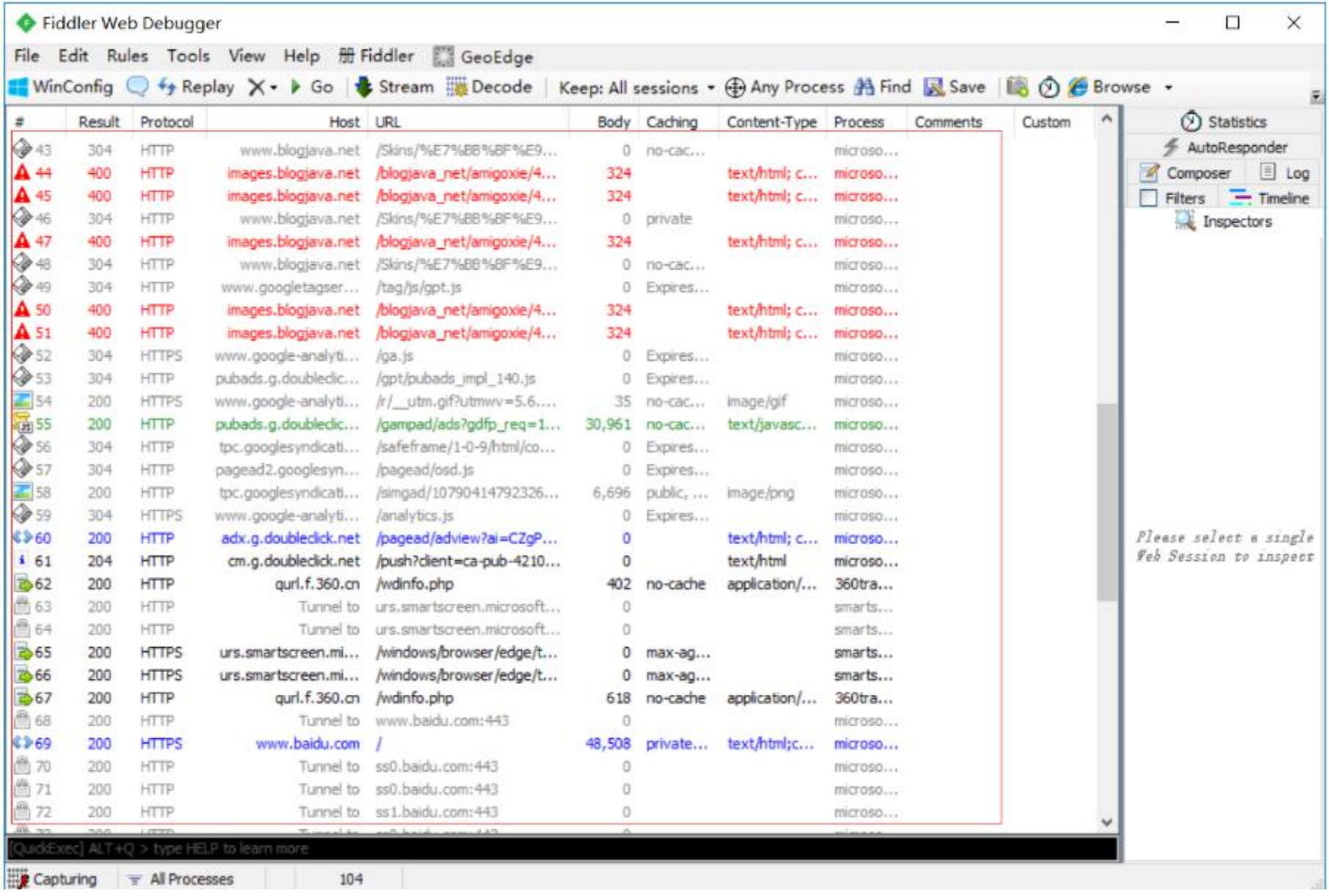
Fiddler 的 session 都是按照顺序展示的，在序号前面则是 session 的快捷图标，如下图所示：



#	Result	Protocol	Host	URL
1	200	HTTPS	www.fiddler2.com	/UpdateCheck.aspx?is
2	502	HTTP	Tunnel to	clients1.google.com:4
3	502	HTTP	Tunnel to	clients1.google.com:4
4	502	HTTP	Tunnel to	clients1.google.com:4
5	200	HTTP	Tunnel to	hm.baidu.com:443
6	502	HTTP	Tunnel to	clients1.google.com:4
7	502	HTTP	Tunnel to	clients1.google.com:4
8	502	HTTP	Tunnel to	clients1.google.com:4
9	502	HTTP	Tunnel to	clients1.google.com:4
10	502	HTTP	Tunnel to	clients1.google.com:4
11	502	HTTP	Tunnel to	clients1.google.com:4
12	502	HTTP	Tunnel to	clients1.google.com:4

五、对抓取的日志解析

如图所示的区域为数据包列表，要分析这些数据包，首先要了解各字段的含义。



例

#：顺序号，按照抓包的顺序从1递增

Result：HTTP状态码

Protocol：请求使用的协议，如HTTP/HTTPS/FTP等

HOST：请求地址的主机名或域名

URL：请求资源的位置

Body：请求大小

Caching：请求的缓存过期时间或者缓存控制值

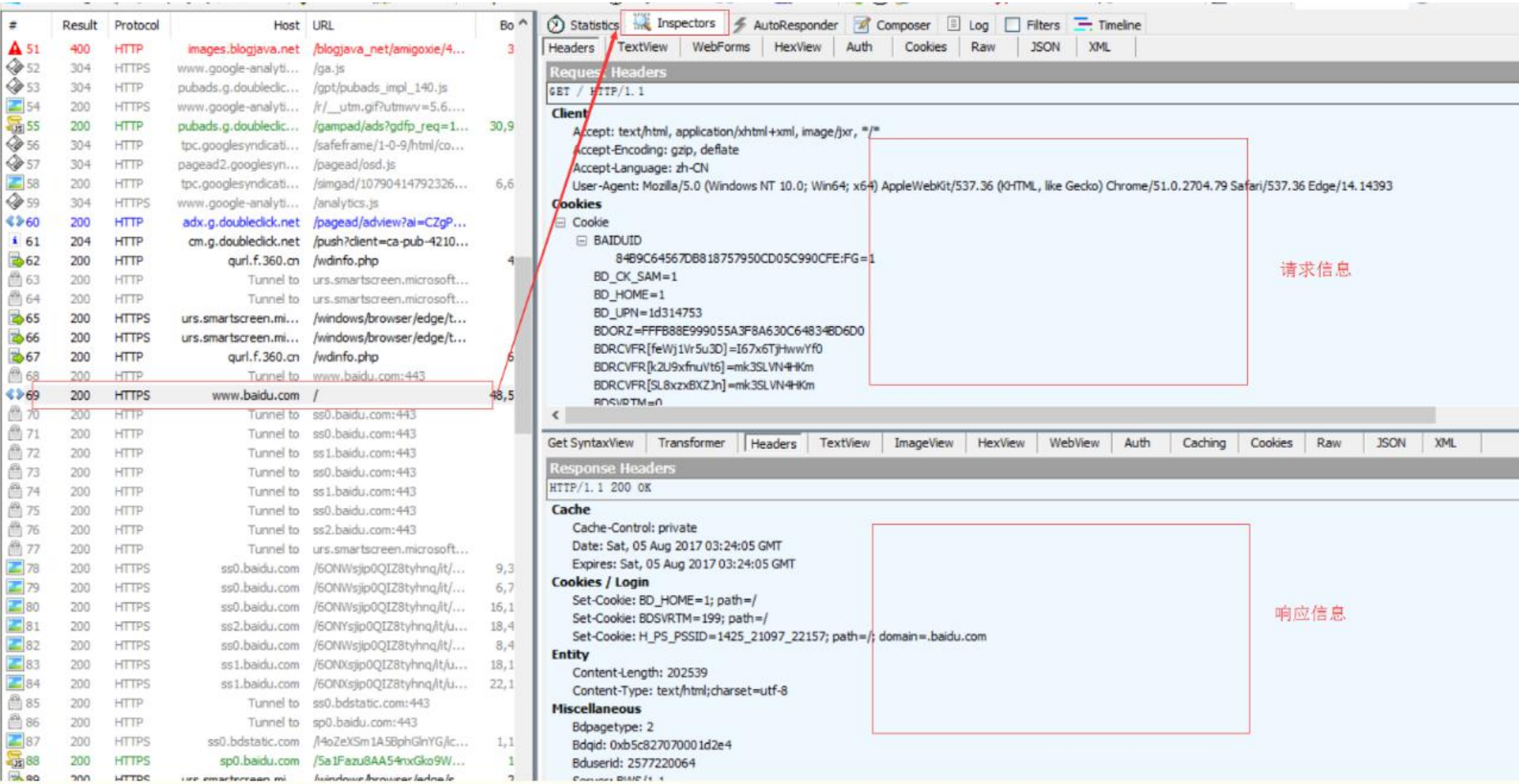
Content-Type：请求响应的类型

Process：发送此请求的进程ID

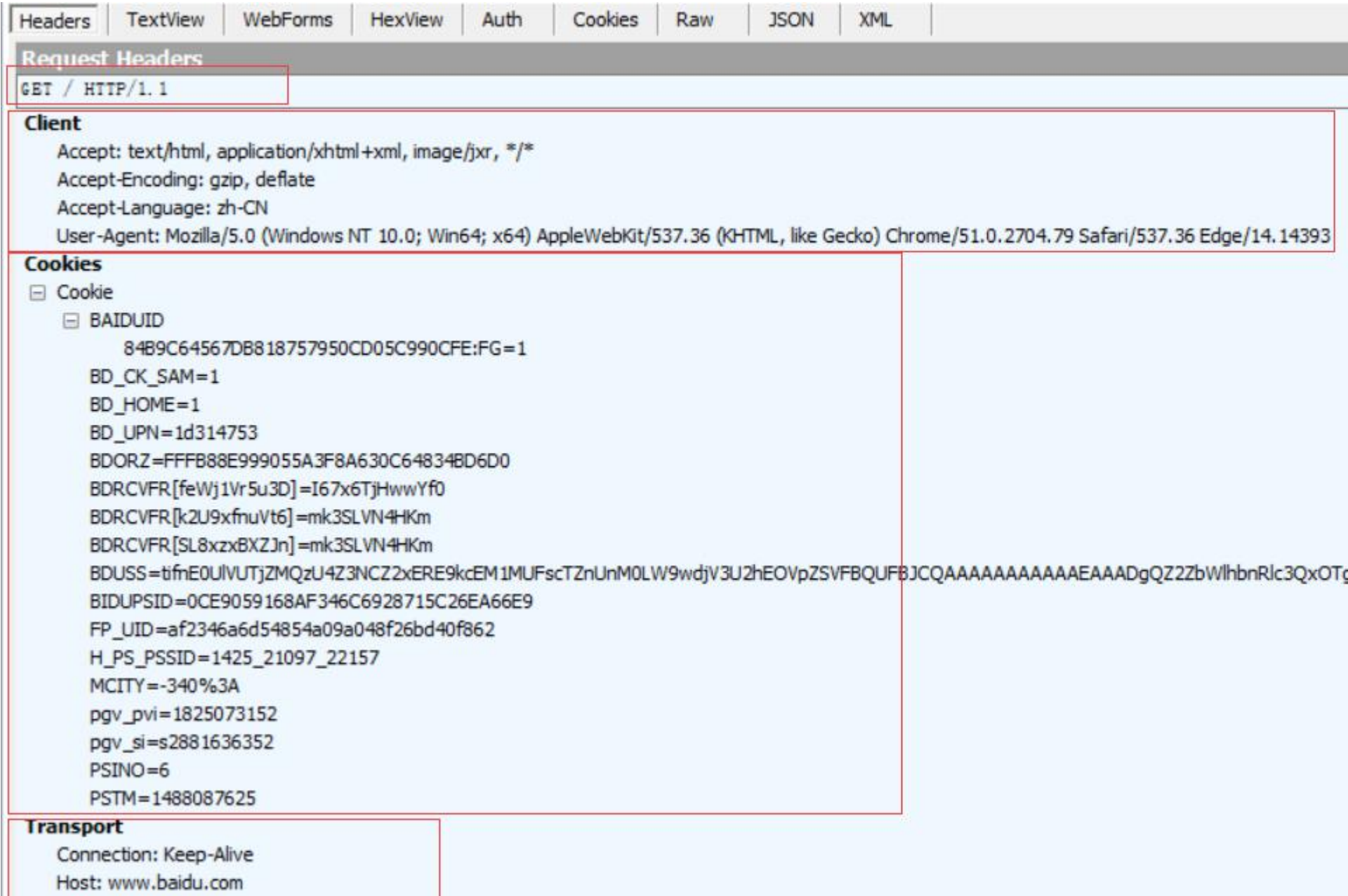
Comments：备注

Custom：自定义值

每个Fiddler抓取到的数据包都会在该列表中展示，点击具体的一条数据包可以在右侧菜单点击Insepector查看详细内容。主要分为请求（即客户端发出的数据）和响应（服务器返回的数据）两部分。



HTTP Request Header：以百度为例，查看请求百度主页这条数据包的请求数据，从上面的Headers中可以看到如下内容：



请求方式：GET
协议： HTTP/1.1
Client 头域：
Accept: text/html, application/xhtml+xml, image/jxr, */* ---------浏览器端可以接受的媒体类型

Accept-Encoding: gzip, deflate ---------压缩方法

Accept-Language: zh-CN --------语言类型

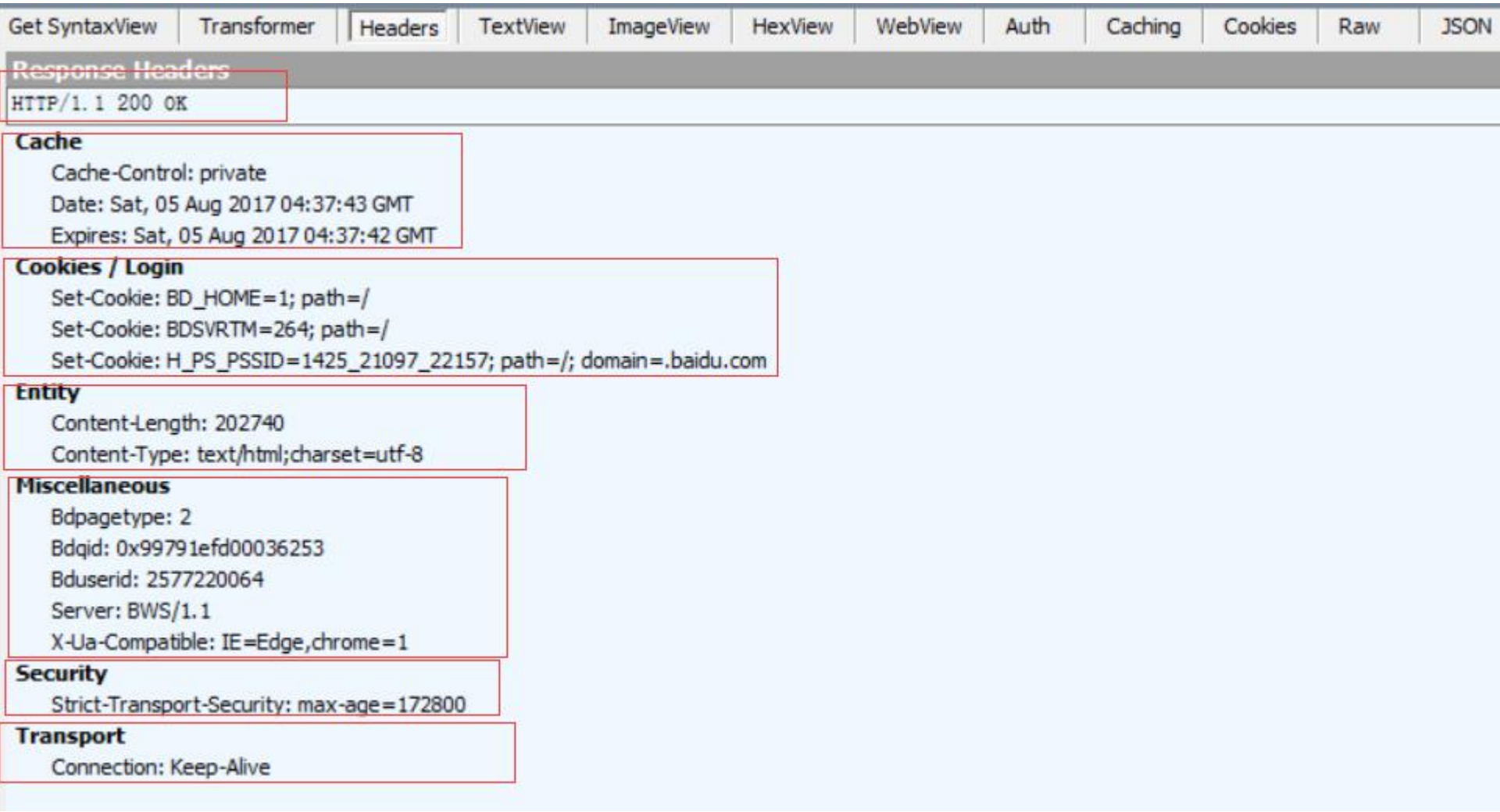
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.79 Safari/537.36 Edge/14.14393
---------客户端使用的操作系统和浏览器的名称和版本

COOKIE头域：将cookie值发送给服务器

Transport 头域：
Connection：当网页打开完成后，客户端和服务器之间用于传输HTTP数据的TCP连接是否关闭。keep-alive表示不会关闭，客户端再次访问这个服务器上的网页，会继续使用这一条已经建立的连接；close表示关闭，客户端再次访问这个服务器上的网页，需要重新建立连接。

HOST：主机名或域名，若没有指定端口，表示使用默认端口80.

HTTP ResponseHeader：继续以百度为例，如图所示：



协议：HTTP/1.1

状态码：200

Cache头域：
Cache-Control: private ---------此响应消息不能被共享缓存处理，对于其他用户的请求无效

Date: Sat, 05 Aug 2017 04:37:43 GMT ---------生成消息的具体时间和日期

Expires: Sat, 05 Aug 2017 04:37:42 GMT ---------浏览器会在指定过期时间内使用本地缓存

Cookie/Login 头域：

Set-Cookie: BDSVRTM=264; path=/ ---------把cookie发送到客户端
Set-Cookie: BD_HOME=1; path=/
Set-Cookie: H_PS_PSSID=1425_21097_22157; path=/; domain=.baidu.com

Entity头域

Content-Length: 202740 ---------正文长度
Content-Type: text/html;charset=utf-8 ---------告知客户端服务器本身响应的对象的类型和字符集

Miscellaneous 头域：
Bdpagetype: 2

Bdqid: 0x99791efd00036253

Bduserid: 2577220064
Server: BWS/1.1 ---------指明HTTP服务器的软件信息

X-Ua-Compatible: IE=Edge,chrome=1
Security头域：
Strict-Transport-Security: max-age=172800 ---------基于安全考虑而需要发送的参数，关于这个参数的解释，请参考：
http://www.freebuf.com/articles/web/66827.html

Transport头域：

Connection: Keep-Alive

6）TextView：显示请求或响应的数据。

7）WebForms：请求部分以表单形式显示所有的请求参数和参数值；响应部分与TextView内容是一样的。

8）Auth：显示认证信息，如Authorization

9）Cookies：显示所有cookies

10）Raw：显示Headers和Body数据

11）JSON：若请求或响应数据是json格式，以json形式显示请求或响应内容

12）XML：若请求或响应数据是xml格式，以xml形式显示请求或响应内容

13）上面是以百度主页为例，百度主页采用的是GET请求，在TextView中没有请求body，我们再以无忧行网站登录接口为例，它是一个POST请求，除了请求头外，在TextView中多了请求数据。这也是GET请求和POST请求的一个区别。GET请求是将请求参数放在url中，而POST请求一般是将请求参数放在请求body中。


Request Headers
POST /portal/api/usr/v1/login HTTP/1.1
Cache
Pragma: no-cache
Client
Accept: application/json, text/javascript, */*; q=0.01
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.79 Safari/537.36 Edge/14.14393
X-Requested-With: XMLHttpRequest
Cookies
Cookie
Hm_lpvt_a6f2de7e2c14b9597500778aea6c3e8c=1501911312
Hm_lvt_813bd30a7823288c414f5ae99aaa8ba4=1495246928,1495340588
Hm_lvt_a6f2de7e2c14b9597500778aea6c3e8c=1501911312
JSESSIONID=EE9BDFD440EB05E1A845EDC6D531C7C1


Headers
TextView
WebForms
HexView
Auth
Cookies
Raw
JSON
XML
account=[redacted]&password=13002df31a5196657b4c35203a1fe218&countryCode=86&clientType=4