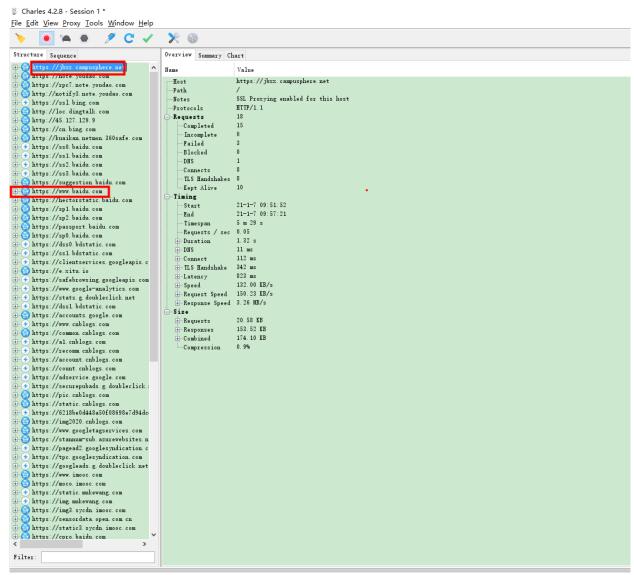
一、Structure/结构视图

将网络请求按访问的域名分类,

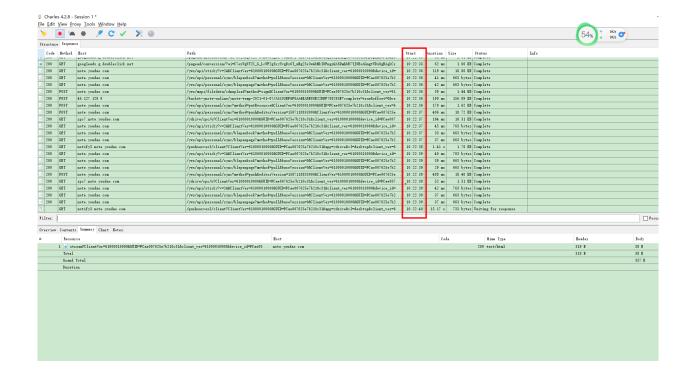
比如某个域名下有n个资源请求,那么所有此域名下的请求都会汇总到这个域名

下



二、Sequence/序列视图

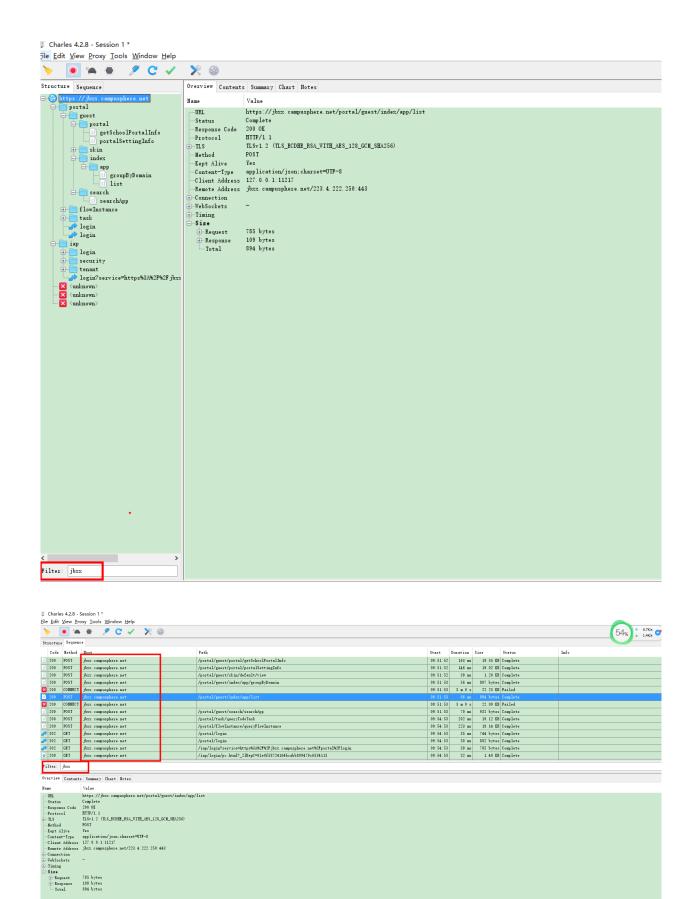
将网络请求按访问的时间排序,按照发送请求先后顺序进行以此排列。 如下图 红框部分 就是请求发出去的具体时间,按先后顺序排列展示



三、请求筛选过滤

请求很多的时候,我们只想关注我们要抓取的域名,怎么办,可以进行筛选过滤。

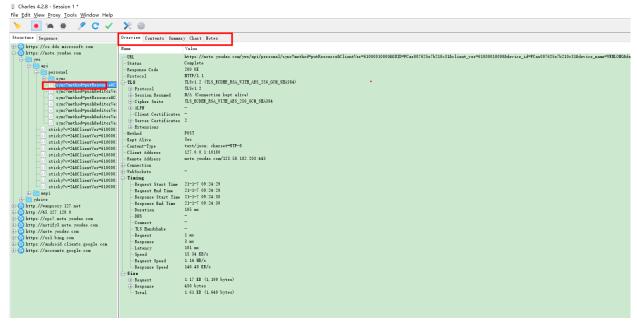
结构/序列两种视图都可以进行filter输入框就行关键字筛选过滤,且互不影响即两种视图的过滤是独立的,可以在结构视图中过滤关键字A,在序列视图过滤B



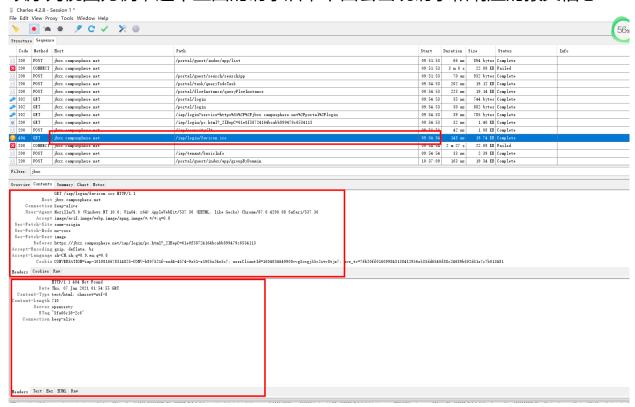
四、报文解析

4.1、选择要解析的请求

以结构视图为例,选中左边的某个请求后,右边出现该请求和响应的报文信息



以序列视图为例,选中上面的请求后,下面会出现请求和响应的报文信息



4.2、通过不同窗口解析报文

解析报文常用的就是OverView窗口和Contents窗口

4.2.1、OverView窗口

该窗口显示的是请求报文的概要总览,分name和value两列展示,如下图所示



URL 请求地址

Status 响应状态

Response Code 状态code

Protocol 协议

TLS https协议

Method 请求方法

Kept Alive 长链接

Content-Type 内容类型

Client Address 客户端地址

Remote Address 远端地址

Connection 链接

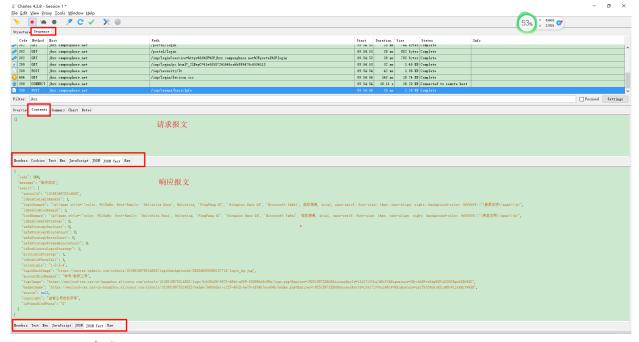
Timing 耗时

Size 请求响应大小

4.2.2、Contents窗口

不管是序列视图还是结构视图, Contents窗口都分为上下两部分子窗口(windows为例 MAC另说)

上面窗口是请求报文子窗口,展示的是浏览器等客户端发给服务器的请求报文下面窗口是响应报文子窗口,展示的是服务器回复给客户端的响应报文下图以序列视图为例,查看Contents窗口



根据不同请求

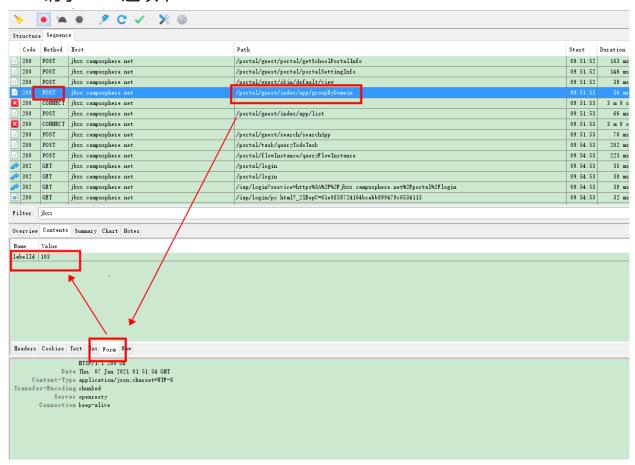
请求报文子窗口,又有Header、Cookies、Text、JSON、Form、Raw等选项卡

响应报文子窗口,又有Header、Text、JavaScript、JSON、Raw等选项卡 Raw是未经处理的请求信息,左侧的所有项目都是对raw信息的拆分和美化 其中常用的选项卡截图:

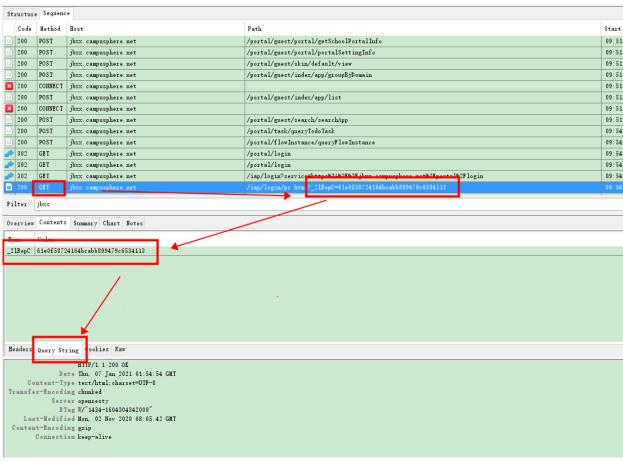
Post请求JSON/JSON Text选项卡:



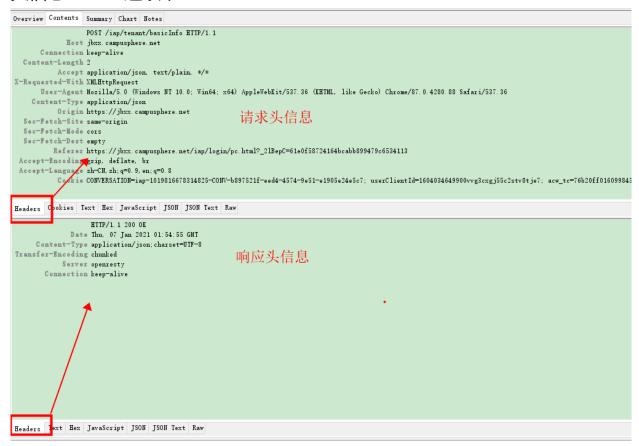
Post请求Form选项卡:



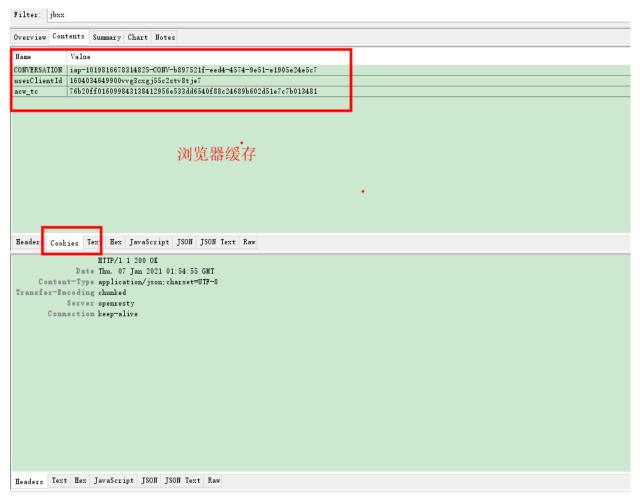
Get请求 QueryString选项卡: ?后面的查询参数



头信息Header选项卡:



缓存Cookie选项卡:



cookie一般会保存用户身份信息,所以可以通过抓包获取到登陆后的用户 cookie,就可以在postman/httpclient等工具/代码中直接访问登陆后才能访问 的服务/资源

4.2.3、其他窗口简介

Summary窗口:展示的是资源概要信息,类似Chrome Dev的Network

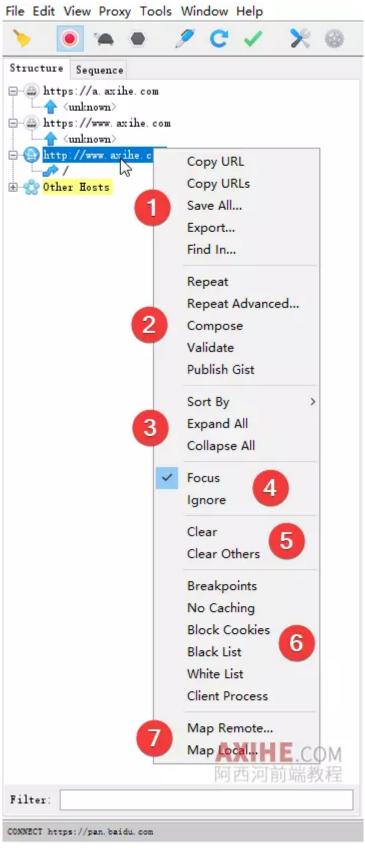
Chart窗口:展示的是timeline的文件加载图形化信息

Notes窗口:展示的是来源之类的说明信息

这几个窗口一般不会用到,只做简单介绍。

五、右键操作

不论在结构视图还是序列视图,当选中某个请求/选中某个域名时,都可以右键点击选择后续操作,具体如下图:



区域 1 基本操作 : 基本的URL复制, 文件保存, 以及选中文件内搜索

• 区域 2 **重写操作**: 重写发送请求(调用接口合适), 或者发到gitlist

里

• 区域 3 **查看区**:排序之类的

• 区域 4 设置关注:可以设置哪些你关心的或者需要忽略的

• 区域 5 清除区: 垃圾清理使用

• 区域 6 工具区:对应常见工具的入口

• 区域 7 映射区: 把文件映射到本地或者原创URL

当然也可以在OverView窗口和Contents窗口,右键选取操作,其操作选项是上面的子集,就不再赘述了。

后面文章会对几种重要的右键操作进行讲解,敬请期待。