

# HTTP协议

## 1 Http简介

### 什么是Http协议

HTTP协议：超文本传输协议（HTTP，HyperText Transfer Protocol）是互联网上应用最为广泛的一种网络协议。用于定义WEB浏览器与WEB服务器之间交换数据的过程。

传输协议：在客户端和服务端通信时，规范了传输数据的格式



必须先有请求,才会有响应

HTTP协议的作用

- HTTP协议是学习JavaWEB开发的基石，不深入了解HTTP协议，就不能说掌握了WEB开发，更无法管理和维护一些复杂的WEB站点。

HTTP协议的特点

- 基于请求/响应模型的协议。
  - 请求和响应必须成对；
  - 先有请求后有响应。
- 简单快捷
  - 因为发送请求的时候只需要发送请求方式和请求路径即可
- HTTP协议默认的端口:80
  - 例如：`http://www.lagou.com:80`
- 无状态协议
  - 多次请求之间相互独立，不能交互数据

### Http协议的版本

- HTTP/1.0，发送请求，创建一次连接，获得一个web资源，连接断开。
- HTTP/1.1，发送请求，创建一次连接，获得多个web资源，连接断开。

### HTTP协议有两种报文格式：

- 请求报文：由客户端向服务器端发出的报文。
- 响应报文：从服务端到客户端的报文。

## 2 Http协议详解

## 2.1 Http请求报文详解

HTTP请求报文：由客户端向服务器端发出的报文。

HTTP请求报文格式：包含请求行、请求头、空行、请求体 四个部分



### 请求行

请求行格式：请求方式 资源路径 协议/版本

例如：POST /web01/login.html HTTP/1.1

请求行必须在HTTP请求格式的第一行。

请求方式：协议规定7种，常用两种：GET和POST

GET请求：

将请求参数追加在URL后面，不安全。例如：login.html?username=tom&password=1234

URL长度限制GET请求方式的数据大小。

没有请求体

POST请求

请求参数显示请求体处，较安全。

请求数据大小没有限制。

只有表单设置为method="post"才是post请求。其他的都是get请求。

常见GET请求：地址栏直接访问、<a href="">、<img src=""> 等

### 请求头

请求头：描述了客户端向服务器发送请求时使用的http协议类型，所使用的编码，以及发送内容的长度，referer，等等。

请求头也是用的键值对key:value

常见请求头	描述
Referer	浏览器通知服务器，当前请求来自何处。如果是直接访问，则不会有这个头。常用于：防盗链
Cookie	与会话有关技术，用于存放浏览器缓存的cookie信息。
User-Agent	浏览器通知服务器，客户端浏览器与操作系统相关信息

## 请求体

通常情况下，只有**post**请求方式才会使用到请求体，请求体中都是用户表单提交的数据，每一项数据都使用键值对**key=value**，多组值使用**&**相连。

例如：**username=tom&password=1234**

## Http请求报文演示

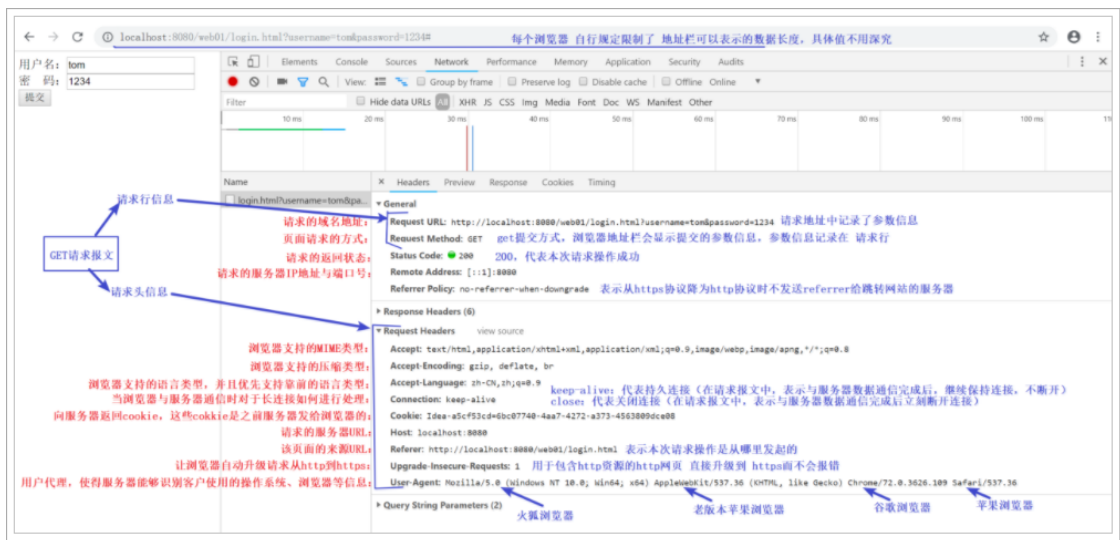
- 创建页面，编写“login.html”，并提供表单，分别设置表单的提交方式为：get和post。将表单提交位置设置成#，表示提交到当前表单。



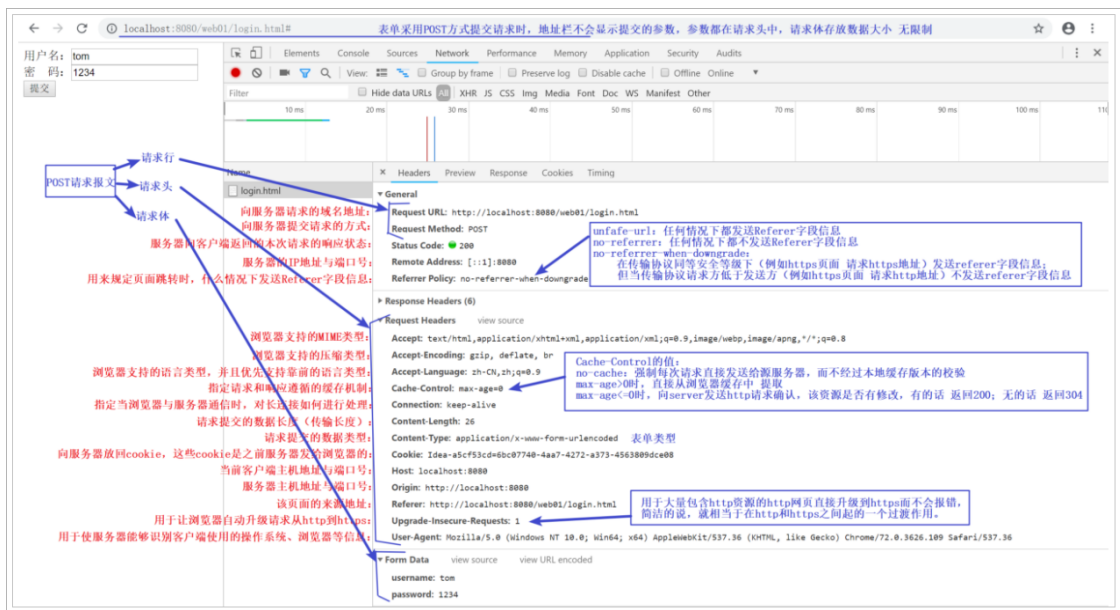
```
<form action="#" method="post">
  用户名: <input type="text" name="username" value=""/> <br/>
  密 码: <input type="text" name="password" value=""/> <br/>
  <input type="submit" />
</form>
```

观看下图，我们采用GET请求方式、POST请求方式的抓包结果：（Chrome浏览器）

- GET请求抓包数据：



- POST请求抓包数据:



## 2.2 Http响应报文详解

响应报文: 从服务端到客户端的报文。

HTTP响应报文格式: 响应行、响应头、空行、响应体 四个部分



## 响应行

例如: HTTP/1.1 200 OK

格式: 协议/版本 状态码 状态码描述

状态码: 服务器与浏览器用于确定状态的固定数字号码

200 : 请求成功。

302 : 请求重定向。

304 : 请求资源没有改变, 访问本地缓存。

404 : 请求资源不存在。通常是用户路径编写错误, 也可能是服务器资源已删除。

500 : 服务器内部错误。通常程序抛异常。

## 响应头

响应头: 用来描述服务器回给客户端浏览器的content的一些描述, 例如: 我是什么服务器, 我返回的是啥编码, 我返回的内容有多长等等

响应头也是用的键值对 key:value

常见响应头	描述
Location	指定响应的路径, 需要与状态码302配合使用, 完成跳转。
Content-Disposition	文件下载的时候使用。通过浏览器以下载方式解析正文
Set-Cookie	与会话相关技术。服务器向浏览器写入cookie
Refresh	定时刷新

## 响应体

响应体, 就是服务器发送给浏览器的正文。

```
<!DOCTYPE html>
<html >
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
  <form action="#" method="post">
    用户名: <input type="text" name="username" value=""/> <br/>
    密 码: <input type="text" name="password" value=""/> <br/>
    <input type="submit"/>
  </form>
</body>
</html>
```

## Http响应报文演示

如下图, 我们提供的响应的抓包结果 (Chrome浏览器)

← → ↻ localhost:8080/web01/login.html#

用户名:   
密码:   
提交

Elements Console Sources Network Performance Memory Application Security Audits

Filter Hide data URLs XHR JS CSS Img Media Font Doc WS Manifest Other

10 ms 20 ms 30 ms 40 ms 50 ms 60 ms 70 ms 80 ms 90 ms 100 ms

Name X Headers Preview Response Cookies Timing

login.html

General

Request URL: http://localhost:8080/web01/login.html  
Request Method: POST  
Status Code: 200  
Remote Address: [::1]:8080  
Referrer Policy: no-referrer-when-downgrade

Response Headers view source

Accept-Ranges: bytes  
Content-Length: 349  
Content-Type: text/html  
Date: Fri, 22 Feb 2019 09:45:44 GMT  
ETag: W/"349-1550827501147"  
Last-Modified: Fri, 22 Feb 2019 09:25:01 GMT

Headers Preview Response Cookies Timing

1 <!DOCTYPE html>  
2 <html>  
3 <head>  
4 <meta charset="UTF-8">  
5 <title>Title</title>  
6 </head>  
7 <body>  
8 <form action="#" method="post">  
9 用户名: <input type="text" name="username" value=""> <br/>  
10 密码: <input type="text" name="password" value=""> <br/>  
11 <input type="submit"/>  
12 </form>  
13 </body>  
14 </html>

响应行  
响应头  
响应体

Http响应报文

Web服务器表明自己是否接受获取某个实体的一部分的请求:  
响应给客户端浏览器的内容长度:  
响应给客户端浏览器的内容类型:  
服务器做出响应的时间:  
表示一个对象的标志值:  
Web服务器记录的对象最后修改时间:

服务器给客户端响应的状态码:  
200 : 请求成功。  
302 : 请求重定向。  
304 : 请求资源没有改变, 访问本地缓存。  
404 : 请求资源不存在。通常是用户路径编写错误, 也可能是服务器资源已删除。  
500 : 服务器内部错误。通常程序抛异常。

Web服务器表明自己是否接受获取某个实体的一部分 (比如表单的一部分) 的请求  
bytes: 表示接受  
none: 表示不接受

Etag: 就是一个对象的标志值, 就一个对象而言, 如何一个html文件, 如果被修改了, 其Etag也会别修改, 所以, Etag的作用跟Last-Modified的作用差不多, 主要供web服务器判断一个对象是否改变了。