

# 第 1 章

## Web 基础与 HTTP 协议

### 技能目标

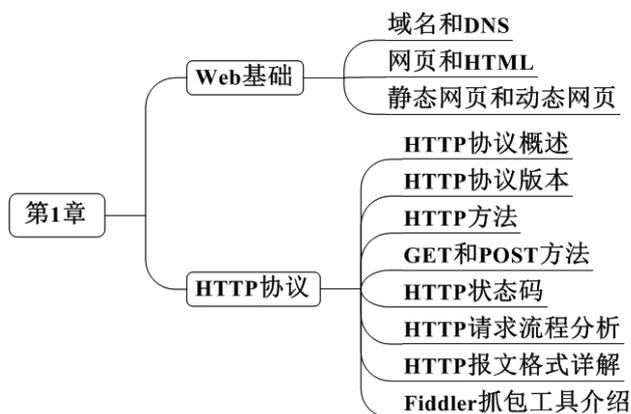
- 了解静态网页与动态网页
- 理解 HTTP 协议的 GET 方法和 POST 方法
- 理解 HTTP 协议请求流程
- 掌握 Fiddler 抓包工具的使用

### 本章导读

随着互联网的飞速发展，企业信息化应用大多已采用网页的形式构建，掌握网页的相关知识和 HTTP 的请求流程，是掌握互联网技术的第一步，本章将讲解相关内容。

APP 扫码看视频





## 1.1 Web 基础

本节先讲解 Web 基础知识，包括域名的概念、DNS 原理、静态网页和动态网页的相关知识。

### 1.1.1 域名和 DNS

#### 1. 域名的概念

网络是基于 TCP/IP 协议进行通信和连接的，每一台主机都有一个唯一的固定的 IP 地址，以区别于网络上成千上万个用户和计算机。网络在区分所有与之相连的网络和主机时，均采用了一种唯一、通用的地址格式，即每一个与网络相连接的计算机和服务器都被指派了一个独一无二的地址。为了保证网络上每台计算机的 IP 地址的唯一性，用户必须向特定机构申请注册、分配 IP 地址。网络中的地址方案分为两套：IP 地址系统和域名地址系统。这两套地址系统其实是一一对应的。IP 地址用二进制数来表示，每个 IP 地址长 32 比特，由 4 组 8 位的二进制数字组成，数字之间用点间隔，例如 100.10.0.1 表示一个 IP 地址。由于 IP 地址是数字标识，使用时难以记忆和书写，因此在 IP 地址的基础上发展出一种符号化的地址方案，来代替数字型的 IP 地址。每一个符号化的地址都与特定的 IP 地址对应，这样网络上的资源访问起来就容易得多了。这个与网络上的数字型 IP 地址相对应的字符型地址，就被称为域名。

通俗的说，域名就相当于一个家庭的门牌号码，别人通过这个号码可以很容易地找到你。

##### (1) 域名的结构

以一个常见的域名为例来说明，如图 1.1 所示，www.baidu.com 网址由两部分组成，“baidu”是这个域名的主体，而最后的“com”则是该域名的后缀，代表这是一个 com 国际域名，是顶级域名，而前面的 www 是主机名。

域名由英文字母和数字组成，每一组不超过 63 个字符，也不区分大小写字母，

除连字符 (-) 外不能使用其他的标点符号。级别最低的域名写在最左边，而级别最高的域名写在最右边。由多组组成的完整域名总共不超过 255 个字符。

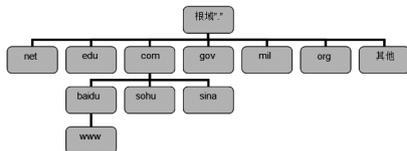


图 1.1 域名结构图

一些国家也纷纷开发采用本民族语言构成的域名，如德语、法语等。中国也开始使用中文域名，但可以预计的是，在今后相当长的一段时期内，以英语为基础的域名（即英文域名）仍然是主流。

## （2）域名结构类型

① 根域：指的是根服务器，用来管理互联网的主目录，全世界只有 13 个。1 个为主根服务器，放置在美国；其余 12 个均为辅根服务器，其中美国 9 个，欧洲 2 个，位于英国和瑞典，亚洲 1 个，位于日本。所有根服务器均由美国政府授权的互联网域名与号码分配机构（ICANN）统一管理，来负责全球互联网域名根服务器、域名体系和 IP 地址等的管理。

② 顶级域：包括组织域名和国家 / 地区域名。域名的最右侧是国家 / 地区域名，国家代码由两个字母组成，如 .cn、.de 和 .jp，其中 .cn 是中国专用的顶级域名。在国家 / 地区域名左侧是组织域名，常见的 .com 用于商业机构，.net 用于网络组织，.org 用于各种组织（包括非盈利组织）。

③ 二级域名：在顶级域名之前的域名，它是指域名注册人的网上名称，例如 baidu、ibm、yahoo、microsoft 等。

④ FQDN：即主机名 .DNS 后缀，是指主机名加上全路径，全路径列出了序列中的所有域成员。全域名可以从逻辑上准确地表示出主机在什么地方，也可以说全域名是主机名的一种完全表示形式。从全域名中包含的信息可以看出主机在域名树中的位置。

## 2. Hosts 文件

Hosts 文件是一个用于存储计算机网络中节点信息的文件，它可以将主机名映射到相应的 IP 地址，实现 DNS 的功能，它可以由计算机的用户进行修改控制。

### （1）Hosts 文件的作用

在网络上访问网站，要首先通过 DNS 服务器把要访问的域名解析成 IP 地址后，计算机才能对这个网络域名作访问。

要是对于每个域名请求我们都要等待域名服务器解析后返回 IP 信息，这样访问网络的效率就会降低，因为 DNS 做域名解析和返回 IP 都需要时间。为了提高对经常访问的网络域名的解析效率，可以通过在 Hosts 文件中建立域名和 IP 的映射关系来达到目的。根据系统规定，在进行 DNS 请求以前，系统会先检查自己的 Hosts 文件中是否

有这个网络域名映射关系。如果有，则调用这个 IP 地址映射，如果没有，再向已知的 DNS 服务器提出域名解析，也就是说 Hosts 的请求级别比 DNS 高。

### (2) 修改 Hosts 文件

Windows 系统中 Hosts 文件存储在目录 `c:\windows\system32\drivers\etc\` 下面，用记事本可以对其进行修改，CentOS 系统中存储在目录 `/etc/hosts` 下面，用 `vi /etc/hosts` 可以对其进行修改，如在文件中加上以下代码。

```
127.0.0.1 www.baidu.com
```

当访问 `www.baidu.com` 时，发现本机 `hosts` 文件中有映射的 IP 地址，则访问这个 IP 地址。

## 3. DNS

在互联网上域名与 IP 地址之间是一一对应的，域名虽然便于人们记忆，但机器之间只能互相认识 IP 地址，它们之间的转换工作称为域名解析，域名解析要由专门的域名解析系统来完成，DNS 就是进行域名解析的系统。

主机名到 IP 地址的映射有两种方式。

(1) 静态映射。每台设备上配置主机到 IP 地址的映射，各设备独立维护自己的映射表，而且只供本设备使用。

(2) 动态映射。建立一套域名解析系统 (DNS)，只在专门的 DNS 服务器上配置主机到 IP 地址的映射，网络上需要使用主机名通信的设备，首先需要到 DNS 服务器查询主机所对应的 IP 地址。

通过主机名，最终得到该主机名对应的 IP 地址的过程叫作域名解析 (或主机名解析)。在解析域名时，可以首先采用静态域名解析的方法，如果静态域名解析不成功，再采用动态域名解析的方法。将一些常用的域名放入静态域名解析表中，可以大大提高域名解析效率。

在 Windows 命令行模式中输入网络查询命令 `nslookup www.baidu.com`，可以查询到域名对应的 IP 地址。

## 4. 域名注册

域名注册是 Internet 中用于解决地址对应问题的一种方法。域名注册遵循先申请先注册原则，管理机构对申请人提出的域名是否违反第三方的权利不进行任何实质性审查。每个域名都是独一无二的、不可重复的。

域名注册的所有者都是以域名注册提交人填写的域名订单的信息为准的，注册成功 24 小时后，即可在国际 (ICANN)、国内 (CNNIC) 管理机构查询 whois 信息 (whois 信息就是域名所有者等信息)。

域名注册步骤如下。

(1) 准备申请资料：`com` 域名无需提供身份证、营业执照等资料，`cn` 域名已开放个人申请注册，所以申请时需要提供身份证或企业营业执照。

(2) 寻找域名注册网站：由于 `com`、`cn` 等不同后缀域名均属于不同注册管理机

构所管理，如要注册不同后缀域名则需要从注册管理机构寻找经过其授权的顶级域名注册服务机构。如 com 域名的管理机构为 ICANN，cn 域名的管理机构为 CNNIC（中国互联网络信息中心）。若注册商已经通过 ICANN、CNNIC 双重认证，则无需分别到其他注册服务机构申请域名。

(3) 查询域名：在域名注册查询网站注册用户名成功后查询域名，选择您要注册的域名，并点击注册。

(4) 正式申请：查到想要注册的域名，并且确认域名为可申请的状态后，提交注册，并缴纳年费。

(5) 申请成功：正式申请成功后，即可开始进行 DNS 解析管理、设置解析记录等操作。

### 1.1.2 网页与 HTML

网页是构成网站的基本元素，是承载各种网站应用的平台，网页是由 HTML（超文本标记语言）编写的。通俗地说，网站就是由网页组成的，如果您只有域名和虚拟主机而没有制作任何网页的话，客户仍旧无法访问您的网站。

#### 1. 网页概述

网页是一个文件，它存放在世界上某个角落的某一部计算机中，而这部计算机必须是与互联网相连的。网页经由网址（URL）来识别与存取，是互联网中的一“页”。

网页可以包括如下内容。

(1) 文本：文本是网页上最重要的信息载体与交流工具，网页中的主要信息一般都以文本形式为主。

(2) 图像：图像在网页中具有提供信息并展示直观形象的作用。

静态图像：在网页中可能是图片或矢量图形。通常为 GIF、JPEG 或 PNG，或矢量格式，如 SVG 或 Flash。

动画图像：通常为 GIF 和 SVG。

(3) Flash 动画：动画在网页中的作用是有效地吸引访问者更多的注意。

(4) 声音：声音是多媒体和视频网页重要的组成部分。

(5) 视频：视频文件的采用使网页效果更加精彩且富有动感。

(6) 表格：表格用来在网页中控制页面信息的布局方式。

(7) 导航栏：导航栏在网页中是一组超链接，其连接的目的端是网页中重要的页面。

(8) 交互式表单：表单在网页中通常用来连接数据库并接受用户在浏览器端输入的数据，利用数据库为客户端与服务器端提供更多的互动。

网页相关概念如下：

(1) 域名：是浏览网页时输入的网址。

(2) HTTP：用来传输网页的通信协议，使用浏览器访问网址时，在域名前面要加上 http://，表示使用 http 协议传输网页。

- (3) URL：是一种万维网寻址系统，表示网络上资源的位置路径。
- (4) HTML：是编写网页的超文本标记语言。
- (5) 超链接：将网站中不同网页链接起来的功能。
- (6) 发布：将制作好的网页上传到服务器供用户访问的过程。

## 2. HTML 概述

HTML 叫作超文本标记语言，是一种规范，也是一种标准，它通过标记符来标记要显示的网页中的各个部分。网页文件本身是一种文本文件，通过在文本文件中添加标记符，可以告诉浏览器如何显示其中的内容（如：文字如何处理，画面如何安排，图片如何显示等）。浏览器按顺序阅读网页文件，然后根据标记符解释和显示其标记的内容，对书写出错的标记将不指出其错误，且不停止其解释执行过程，编制者只能通过显示效果来分析出错原因和出错部位。但需要注意的是，不同的浏览器，对同一标记符可能会做出不完全相同的解释，因而可能会有不同的显示效果。

HTML 文件可以使用任何能够生成 txt 文件的文本编辑器来编辑，生成超文本标记语言文件，只用修改文件名后缀为“.html”或“.htm”即可。

## 3. HTML 基本标签

### (1) HTML 语法规则

HTML 标签采用双标记符的形式，前后标记符对应，分别表示标记开始和结束，标记符中间的内容被标签描述。前标记符由“<XXX>”表示，结尾标记符多了一个“/”，由“</XXX>”表示。

### (2) HTML 文件结构

HTML 文件最外层由<html></html>表示，说明该文件是用 HTML 语言来描述的。在它里面是并列的头标签（<head>）和内容标签（<body>），最基本的 HTML 文件结构如下：

```
<html>
  <head> 网页的内容描述信息 </head>
  <body> 网页显示的内容 </body>
</html>
```

常用的头标签中的标签如表 1-1 所示。

表 1-1 头标签中常用标签

标签	描述
<title>	定义了文档的标题
<base>	定义了页面链接标签的默认链接地址
<link>	定义了一个文档和外部资源之间的关系
<meta>	定义了 HTML 文档中的元数据
<script>	定义了客户端的脚本文件
<style>	定义了 HTML 文档的样式文件

内容标签中常用的标签如表 1-2 所示。

表 1-2 内容标签中常用标签

标签	描述
<table>	定义了一个表格
<tr>	定义了表格中的一行
<td>	定义了表格中某一行的一列
<a>	定义了一个超链接
 	定义了换行
<font>	定义了字体

#### 4. 网站和主页

网站是由网页组成的，包含多个网页页面，是具有独立域名、独立存放空间的内容集合，这些内容可能是网页，也可能是程序或其他文件。

主页（首页）是用户打开浏览器时默认打开的网页。

当一个网站服务器收到一台电脑上网络浏览器的消息连接请求时，便会向这台电脑发送这个文档。当在浏览器的地址栏输入域名，而未指向特定目录或文件时，通常浏览器会打开网站的首页。网站首页往往会被编辑得易于了解该网站提供的信息，并引导互联网用户浏览网站其他部分的内容。这部分内容一般是一个目录性质的内容。

#### 5. Web1.0 与 Web2.0

Web1.0 是指早期互联网模式，以门户网站为主，内容由网站运营商提供，以巨大的点击量和增值服务为主要盈利手段。

(1) Web1.0 基本采用的是技术创新主导模式，信息技术的变革和使用对于网站的新生与发展起到了关键性的作用。新浪最初就是以技术平台起家，腾讯以即时通讯技术起家，盛大以网络游戏起家，在这些网站的创始阶段，技术性的痕迹相当之重。

(2) Web1.0 的盈利都基于一个共通点，即巨大的点击流量。无论是早期融资还是后期获利，依托的都是为数众多的用户和点击率，以点击率为基础上市或开展增值服务，以及受众群众的基础，决定了盈利的水平和速度，充分地体现了互联网的眼球经济色彩。

Web2.0 是相对于 Web1.0 的新的时代，指的是一个利用 Web 的平台，由用户主导生成内容的互联网产品模式，如博客、社交网站等。为了区别传统由网站雇员主导生成内容而定义为第二代互联网，即 Web2.0，是一个全新的时代。其特征如下：

(1) 用户分享。在 Web2.0 模式下，可以不受时间和地域的限制分享各种观点。用户既可以得到自己需要的信息，也可以发布自己的观点。

(2) 以兴趣为聚合点的社群。在 Web2.0 模式下，聚集的是对某个或者某些问题感兴趣的群体，可以说，在无形中已经产生了细分市场。

(3) 开放的平台，活跃的用户。平台对于用户来说是开放的，而且用户因为兴趣而保持比较高的忠诚度，他们会积极地参与其中。

### 1.1.3 静态网页与动态网页

#### 1. 静态网页

在网站设计中,纯粹 HTML 格式的网页通常被称为“静态网页”,静态网页是标准的 HTML 文件,它的文件扩展名是 .htm、.html。静态网页是网站建设的基础,早期的网站一般都是由静态网页制作的。静态网页也可以呈现各种动态的效果,如 .GIF 格式的动画、Flash、滚动字幕等,但这些“动态效果”只是视觉上的,与下面将要介绍的动态网页是不同的概念。

#### 2. 动态网页

所谓动态网页,是指跟静态网页相对的一种网页编程技术。随着 HTML 代码的生成,静态网页的内容和显示效果就基本上不会发生变化了——除非你修改页面代码。而动态网页则不然,页面代码虽然没有变,但是显示的内容却可以随着时间、环境或者数据库操作的结果而发生改变。动态网页 URL 的后缀不是 .htm、.html、.shtml、.xml 等静态网页的常见网页制作格式,而是以 .aspx、.asp、.jsp、.php、.perl、.cgi 等形式为后缀,并且在动态网页网址中有一个标志性的符号——“?”。

动态网页是基本的 html 语法规范与 Java、PHP、C# 等高级程序设计语言、数据库编程等多种技术的融合,以期实现对网站内容和风格的高效、动态和交互式的管理。因此,从这个意义上来讲,凡是结合了 HTML 以外的高级程序设计语言和数据库技术进行的网页编程生成的网页都是动态网页。

#### 3. 动态网页语言

早期的动态网页主要采用通用网关接口(Common Gateway Interface, CGI)技术,虽然 CGI 技术已经发展成熟而且功能强大,但由于编程困难、效率低下、修改复杂,所以有逐渐被新技术取代的趋势。

目前常用的动态网页编程语言介绍如下:

(1) PHP 即 Hypertext Preprocessor(超文本预处理器),它是当今 Internet 上使用最为火热的脚本语言,其语法借鉴了 C、Java、PERL 等语言,但只需要掌握很少的编程知识就能使用 PHP 建立一个真正交互的 Web 站点。

(2) ASP 即 Active Server Pages(动态服务器页面),它由微软开发,是一种类似于 HTML、Script 与 CGI 的结合体,但并没有提供自己专门的编程语言,而是允许用户使用许多已有的脚本语言编写 ASP 的应用程序。ASP 的程序编制比 HTML 更方便且更灵活,它在 Web 服务器端运行,运行后再将运行结果以 HTML 格式传送至客户端的浏览器。

(3) ASP.NET 的前身是 ASP 技术,是在 IIS2.0 上首次推出,在 IIS3.0 上成为服务器端应用程序的热门开发工具,微软还特别为它量身打造了 Visual InterDev 开发工具。ASP.NET 的简单以及高度可定制化的能力,也是它迅速崛起的原因之一;不过它的缺点也逐渐地浮现出来:面向过程的程序开发方法,让程序维护的难度提高很多,尤其是大型的 ASP 应用程序;采用解释型的 VBScript 或 JScript 语言,让性能无法完全发挥。

(4) JSP 即 Java Server Pages (Java 服务器页面), 它是由 Sun Microsystem 公司于 1999 年 6 月推出的新技术, 是基于 Java Servlet 以及整个 Java 体系的 Web 开发技术。

## 1.2 HTTP 协议

超文本传输协议 (HyperText Transfer Protocol, HTTP) 是互联网上应用最为广泛的一种网络协议, 所有的网页文件都必须遵守这个标准。HTTP 最初的设计目的是为了提供一种发布和接收 HTML 页面的方法。

### 1.2.1 HTTP 协议概述

HTTP 协议采用了请求 / 响应模型。客户端向服务器发送一个请求, 请求头包含请求的方法、URL、协议版本, 以及包含请求修饰符、客户信息和内容的类似于 MIME 的消息结构。服务器以一个状态行作为响应, 响应的内容包括消息协议的版本、成功或者错误编码, 再加上服务器信息、实体元信息以及可能的实体内容。

HTTP 已经演化出了很多版本, 它们中的大部分都是向下兼容的。

(1) HTTP/0.9——已过时。只接受 GET 一种请求方法, 没有在通信中指定版本号, 且不支持请求头。由于该版本不支持 POST 方法, 所以客户端无法向服务器传递太多信息。

(2) HTTP/1.0——这是第一个在通信中指定版本号的 HTTP 协议版本, 至今仍被广泛采用, 特别是在代理服务器中。

(3) HTTP/1.1——当前版本。持久连接被默认采用, 并能很好地配合代理服务器工作; 还支持以管道方式同时发送多个请求, 以便降低线路负载, 提高传输速度。

### 1.2.2 HTTP 方法

HTTP 支持几种不同的请求命令, 这些命令被称为 HTTP 方法 (HTTP method), 每条 HTTP 请求报文都包含一个方法, 告诉服务器要执行什么动作, 包括获取一个页面、运行一个网关程序、删除一个文件等, 最常用的获取资源的方法是 GET、POST。HTTP 常用方法如表 1-3 所示。

表 1-3 HTTP 方法

HTTP 方法	描述
GET	请求获取 Request-URI 所标识的资源
PUT	请求服务器存储一个资源, 并用 Request-URI 作为其标识
DELETE	请求服务器删除 Request-URI 所标识的资源
POST	在 Request-URI 所标识的资源后附加新的数据
HEAD	请求获取由 Request-URI 所标识的资源的响应消息报头

GET 方法采用的是 URL 后缀的形式，比如 `http://www.test.com/a.php?Id=123` 就是一个 GET 请求，服务器接收后可以解析出 `Id=123`，而 POST 方法不需要在 URL 中显示“`?Id=123`”，参数作为内容进行了隐藏的提交，因此提交表单类或者有用户名、密码等内容时建议使用 POST 方法。

GET 方法在 URL 上显示参数，而 URL 有长度限制，故不适合提交过大的数据。GET 方法可以被浏览器缓存，当请求已经被请求过一次的 URL 时，浏览器不需要向服务器再次发出请求，可以直接在本地缓存中获得页面。GET 和 POST 方法的对比如表 1-4 所示。

表 1-4 GET 和 POST 方法比较

	GET 方法	POST 方法
对数据长度的限制	URL 的长度是受限制的（URL 的最大长度是 2048 个字符）	无限制
缓存	能被缓存	不能被缓存
安全性	与 POST 相比，GET 的安全性较差，因为所发送的数据是 URL 的一部分。在发送密码或其他敏感信息时绝不要使用 GET	POST 比 GET 更安全，因为参数不会被保存在浏览器历史或 Web 服务器日志中
历史	参数保留在浏览器历史中	参数不会保存在浏览器历史中
后退按钮 / 刷新	无害	数据会被重新提交（浏览器应该告知用户数据会被重新提交）
书签	可收藏为书签	不可收藏为书签

### 1.2.3 HTTP 状态码

HTTP 状态码（HTTP Status Code）是用以表示网页服务器 HTTP 响应状态的 3 位数字代码，当浏览器请求某一 URL 时，服务器会根据处理情况返回相应的处理状态。HTTP 状态码可以分为五大类，如图 1.2 所示。其中 2XX、3XX 表示请求正常，4XX、5XX 表示出现异常情况。

状态码首位	已定义范围	分类
1xx	100-101	信息提示
2xx	200-206	成功
3xx	300-305	重定向
4xx	400-415	客户端错误
5xx	500-505	服务器错误

图 1.2 HTTP 状态码分类

生产环境常见的状态码如表 1-5 所示。

表 1-5 生产环境常见的 HTTP 状态码

消息	描述
200 OK	请求成功（其后是对 GET 和 POST 请求的应答文档）
301 Moved Permanently	请求的永久页面跳转
403 Forbidden	禁止访问该页面
404 Not Found	服务器无法找到被请求的页面
500 Internal Server Error	内部服务器错误
502 Bad Gateway	无效网关
503 Service Unavailable	当前服务不可用
504 Gateway Timeout	网关请求超时

### 1.2.4 HTTP 请求流程分析

用户在浏览器输入 URL 访问时，发起 HTTP 请求报文，请求中包括请求行、请求头、请求体；服务器收到请求后返回响应报文，包括状态行、响应头、响应体，如图 1.3 所示。

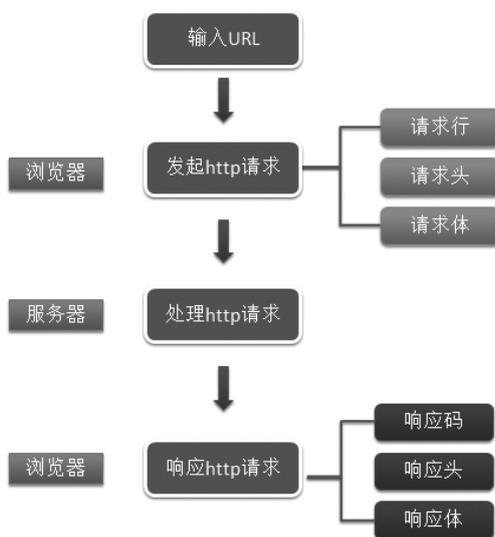


图 1.3 HTTP 请求流程

#### 1. 请求报文

##### (1) 请求行

由三部分组成，分别为：请求方法、URL 以及协议版本。

## (2) 请求头

请求头为请求报文添加了一些附加信息，由“名/值”对组成，每行一对，名和值之间使用冒号分隔，常用的请求头如表 1-6 所示。

表 1-6 HTTP 请求头

请求头	描述
Host	接受请求的服务器地址，可以是 IP: 端口号，也可以是域名
User-Agent	发送请求的应用程序名称
Connection	指定与连接相关的属性，如 Connection:Keep-Alive
Accept-Charset	通知服务端可以发送的编码格式
Accept-Encoding	通知服务端可以发送的数据压缩格式
Accept-Language	通知服务端可以发送的语言

## (3) 空行

请求头的最后会有一个空行，表示请求头结束，接下来为请求体，这一行非常重要，必不可少。

## (4) 请求体

请求体是请求提交的参数，GET 方法已经在 URL 中指明了参数，所以提交时没有数据；而 POST 方法提交的参数在请求体中。

## 2. 响应报文

### (1) 状态行

由三部分组成，分别为：协议版本、状态码、状态码描述。

### (2) 响应头

与请求头类似，为响应报文添加了一些附加信息。常见响应头如表 1-7 所示。

表 1-7 HTTP 响应头

响应头	描述
Server	服务器应用程序软件的名称和版本
Content-Type	响应正文的类型（是图片还是二进制字符串）
Content-Length	响应正文长度
Content-Charset	响应正文使用的编码
Content-Encoding	响应正文使用的数据压缩格式
Content-Language	响应正文使用的语言

### (3) 空行

响应头的最后会有一个空行，表示响应头结束。

#### (4) 响应体

服务器返回的相应 HTML 数据，浏览器对其解析后显示页面。

## 1.2.5 Fiddler 抓包工具

### 1. Fiddler 简介

Fiddler 是一款抓取 HTTP 数据包的工具软件，用于分析 HTTP 报文非常方便。

### 2. Fiddler 使用方法

安装 Fiddler 后，打开浏览器请求某一网址，请求的 URL 会以列表方式显示，如图 1.4 所示。

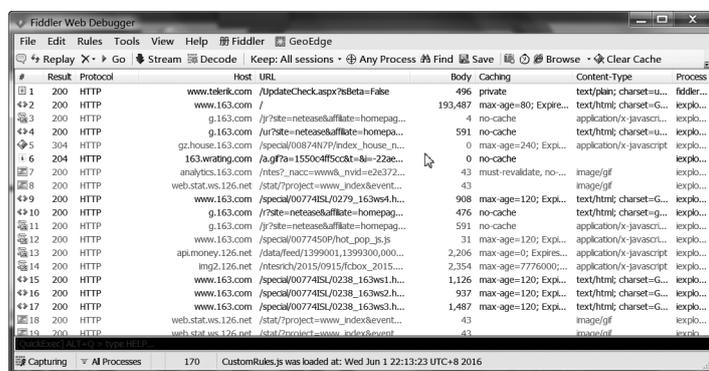


图 1.4 Fiddler 抓包

选中某个 URL，可以看到对应的请求报文和响应报文，选择 Inspectors 中的 Headers，可以看到请求头和响应头的详细信息，如图 1.5 所示。

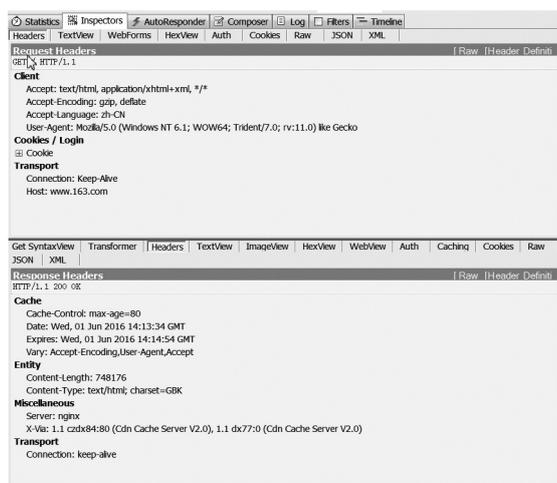


图 1.5 Fiddler 查看请求头和响应头

## 本章总结

- 网页内容由网站内部采集发布的是 Web1.0，内容由使用者提供的是 Web2.0。
- POST 和 GET 方法在缓存、安全性、长度限制等方面有区别。
- HTTP 协议请求响应以报文形式传递。
- Fiddler 是一款抓取 HTTP 数据包的工具软件，用于分析 HTTP 报文非常方便。

## 本章作业

1. 简述静态网页和动态网页的区别。
2. 简述 Web1.0 和 Web2.0 的区别。
3. 简述 GET 和 POST 方法的区别。
4. 结合使用 Fiddler 工具掌握 HTTP 协议的请求报文和响应报文包含的内容。
5. 用课工场 APP 扫一扫完成在线测试，快来挑战吧！

