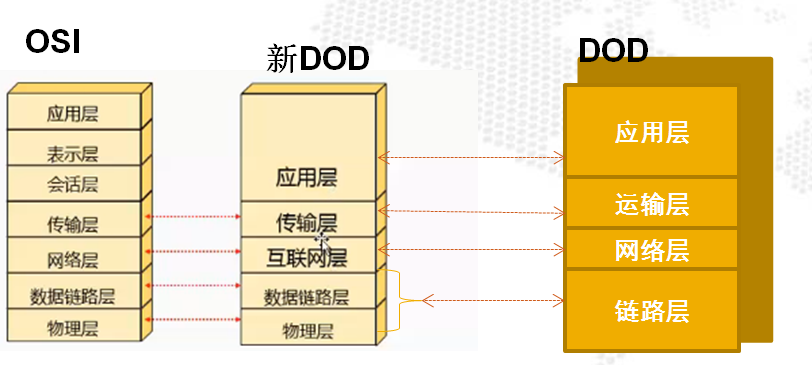
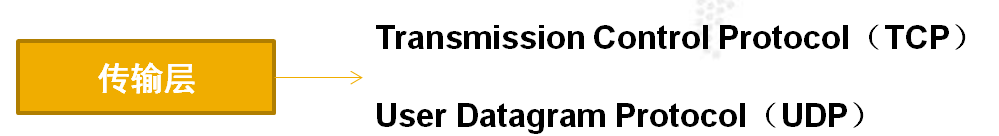
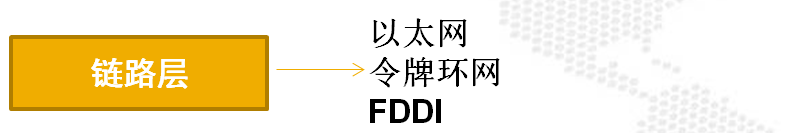
**TCP/IP协议分层**











**HTTP(***HyperText Transfer Protocol* **)即超文本传输协议，是互联网上应用最为**

**广泛的一种网络协议，所有的WWW文件都必须遵守这个标准。设计HTTP最初**

**的目的是为了提供一种发布和接收HTML页面的方法。**

**Web 浏览器、服务器和相关的Web 应用程序都是通过HTTP 相互通信的。HTTP**

**是现代全球因特网中使用的公共语言。**

**URI (Uniform Resource Identifier): 统一资源标识符， URL 实际上是它的一个**

**子集。URI 是一个通用的概念，由两个主要的子集URL 和URN 构成，URL 是通**

**过描述资源的位置来标识资源的，而URN则是通过名字来识别资源的，与它们当**

**前所处位置无关。如果URL 中没有给出abs\_path，那么当它作为请求URI 时，必须以“/”的形式给出，通常这个工作浏览器自动帮我们完成。eg:输入：[www.guet.edu.cn](http://www.guet.edu.cn/)，浏览器自动转换成：http://www.guet.edu.cn/**

**URL (Uniform Resource Location)：统一资源定位符，是浏览器寻找信息时所**

**需的资源位置，通过URL，才能找到、使用并共享因特网上大量的数据资源。它**

**提供了一种定位互联网上任意资源的手段，也是浏览器寻找服务资源所需的信息。**

**大多数URL 方案的URL 语法都建立在这个由9 部分构成的通用格式上：**

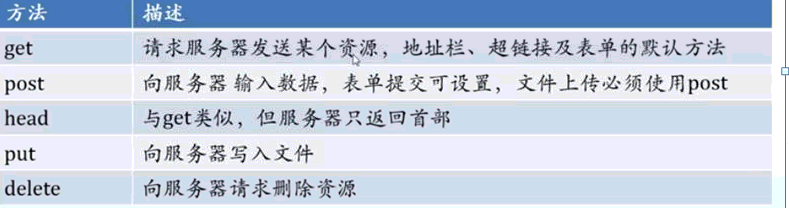
**<scheme>://<user>:<password>@<host>:<port>/<path>;<params>?<query>#<frag>**

|  |  |  |
| --- | --- | --- |
| **组件** | **描述** | **备注** |
| **方案(scheme)** | **访问服务器以获取资源时要使用哪种协议** |  |
| **用户(user)** | **某些方案访问资源时需要的用户名** | **一般在FTP方案下有用户名及对应密码** |
| **密码(password)** | **用户名后面可能要包含的密码，中间由冒号（:）分隔** |
| **主机(host)** | **资源宿主服务器的主机名或点分IP地址** |  |
| **端口(port)** | **资源宿主服务器正在监听的端口号。很多方案都有默认端口号（HTTP 的默认端口号为80）** |  |
| **路径(path)** | **服务器上资源的本地名，由一个斜杠（/）将其与前面的URL 组件分隔开来。路径组件的语法是与服务器和方案有关的（本章稍后会讲到URL 路径可以分为若干个段，每段都可以有其特有的组件。）** |  |
| **参数(parameter)** | **某些方案会用这个组件来指定输入参数。参数为名/ 值对。URL 中可以包含多个参数字段，它们相互之间以及与路径的其余部分之间用分号（;）分隔** |  |
| **查询(query)** | **某些方案会用这个组件传递参数以激活应用程序（比如数据库、公告板、搜索引擎以及其他因特网网关）。查询组件的内容没有通用格式。用字符“?”将其与URL 的其余部分分隔开来** | **动态web开发，如jsp,需要client传入参数，进行数据处理** |
| **片段(frag)** | **一小片或一部分资源的名字。引用对象时，不会将frag 字段传送给服务器；这个字段是在客户端内部使用的。通过字符“#”将其与URL 的其余部分分隔开来** | **锚点名** |

**HTTP常见方案**

|  |  |
| --- | --- |
| **方案** | **描述** |
| **http** | **除了无用户名与密码之外，与通用格式一致，省略端口则使用80** |
| **https** | **与http的区别在于使用了SSL， 默认端口是443** |
| **mailto** | **指向email地址，其格式与通用格式不一致， mailto:test@xxx.com** |
| **ftp** | **文件传输协议， 默认端口号21** |
| **file** | **一台主机上可直接访问文件** |
| **rtsp** | **实时流传输协议** |

**HTTP方法**



**POST与GET区别（WEB开发为例）**



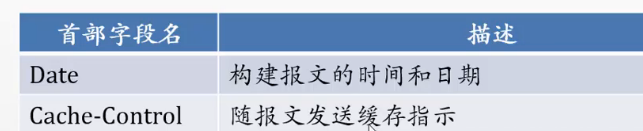
**HTTP状态码分类**



**常见状态码及意义**



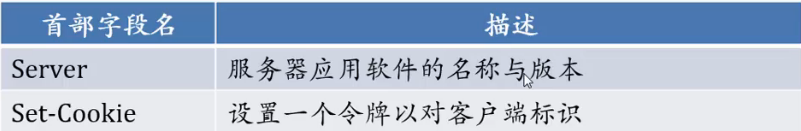
**首部分类**



**常见请求首部**



**常见响应首部**



**常见实体首部**





**常见MIME类型**



**HTTPS（全称：Hypertext Transfer Protocol over Secure Socket Layer）即HTTP下加入SSL层，用于安全的HTTP数据传输。HTTPS存在不同于HTTP的默认端口及一个加密/身份验证层（在HTTP与TCP之间）。这个系统的最初研发由网景公司进行，提供了身份验证与加密通讯方法，现在它被广泛用于万维网上安全敏感的通讯，例如交易支付方面。**