Beijing Forest Studio

北京理工大学信息系统及安全对抗实验中心

http协议概览

硕士研究生 张睿智

2020年04月25日

内容提要

- 背景介绍
- 相关概念
- HTTP特点
- HTTP工作原理
- HTTP报文
- HTTP方法
- HTTP状态码
- HTTPS

背景介绍

- HTTP诞生于互联网的黎明期，CERN（欧洲核子研究组织）的蒂姆·伯纳斯-李博士提出了一种能让远隔两地的研究者们共享知识的设想
- 最初研究者设想的基本概念是：借助多文档之间的相互关联形成的超文本（HyperText），形成可相互参阅的WWW（World Wide Web，万维网）
- 现在已提出了3项 WWW 构建技术
  - 把 SGML（标准通用标记语言）作为页面文本标记语言的HTML
  - 作为文档传递协议的HTTP
  - 指定文档所在地址的URL

背景介绍

- HTTP的发展
  - HTTP/0.9
    - HTTP问世之初，没有作为正式的标准被建立
  - HTTP/1.0
    - HTTP于1996年5月正式作为标准被公布，该版本被命名为HTTP/1.0，并记载于RFC1945
    - 该协议标准至今仍被广泛使用在服务器端
  - HTTP/1.1
    - 于1997年1月公布，是目前最主流的HTTP协议版本
    - 修订版RFC2616是当前的最新版本
  - HTTP/2.0
    - 下一代HTTP协议
    - 2013年8月进行首次合作共事性测试

背景介绍

- TCP/IP协议族
  - 通常使用的网络是在TCP/IP协议族的基础上运作的，HTTP属于其内部的一个子集
  - 在五层协议模型结构中，HTTP属于应用层，该层中其他熟知的协议还有FTP（File Transfer Protocol，文件传输协议）、DNS（Domain Name System，域名系统）等

相关概念

- URL（Uniform Resource Locator）
  - 统一资源定位符
  - 使用浏览器等访问web页面时需要输入的网页地址
- URI（Uniform Resource Identifier）
  - 统一资源标识符
  - 由某个协议方案表示的资源的定位标识符（协议方案指访问资源所使用的协议类型名称），如采用HTTP协议时，协议方案就是http
- URI用字符串标识某一互联网资源，URL表示资源的地点（互联网上的位置）
- URL是URI的子集

HTTP特点

- 无状态（stateless）
  - 无状态协议自身不对请求和响应之间的通信状态进行保存
  - 每当有新的请求发送时，就会有对应的新响应产生
  - 能够快速处理大量事务，确保协议的可伸缩性
  - 后引入Cookie管理状态
- 简单快速
  - 客户向服务器请求服务时，只需传送请求方法和路径
- 灵活
  - 允许传输任意类型的数据对象
  - 正在传输的类型由Content-Type加以标记。

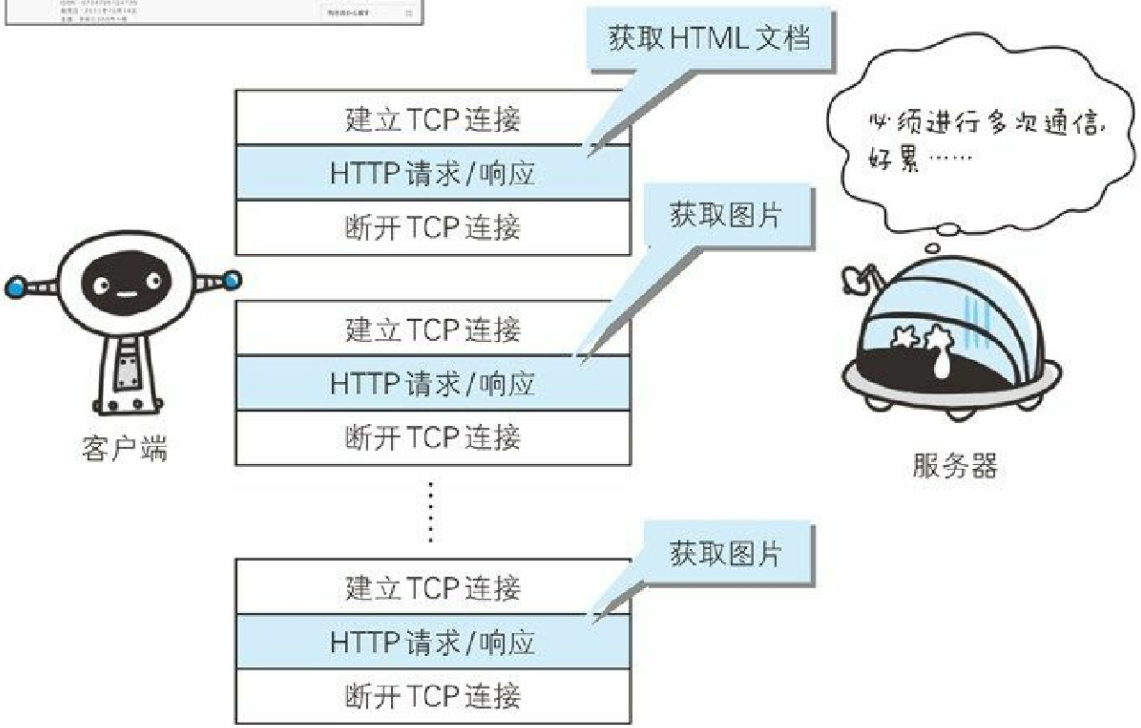
HTTP特点

- 无连接（非持续连接）HTTP/1.0
  - 每次连接只处理一个请求，服务器处理完客户的请求，并收到客户的应答后，即断开连接
  - 客户端再次发送请求时，要建立一个新的连接
  - HTTP/1.0版本的主要缺点
- 持续连接 HTTP/1.1
  - 不用为每个web对象创建一个新的连接，一个连接可以传送多个对象

HTTP特点

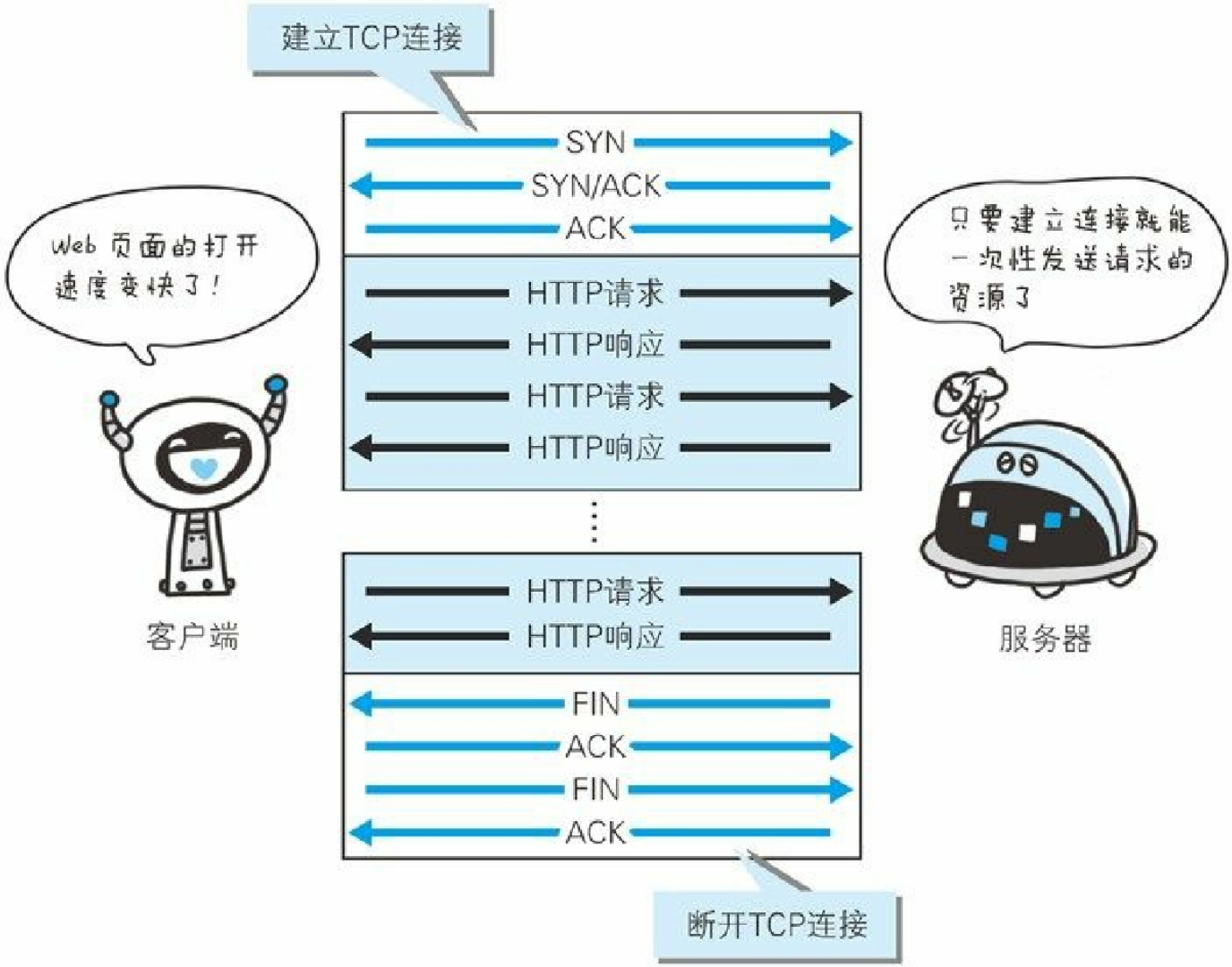
- 非持续连接

发送请求一份包含多张图片的HTML文档对应的Web页面，会产生大量的通信开销。



HTTP特点

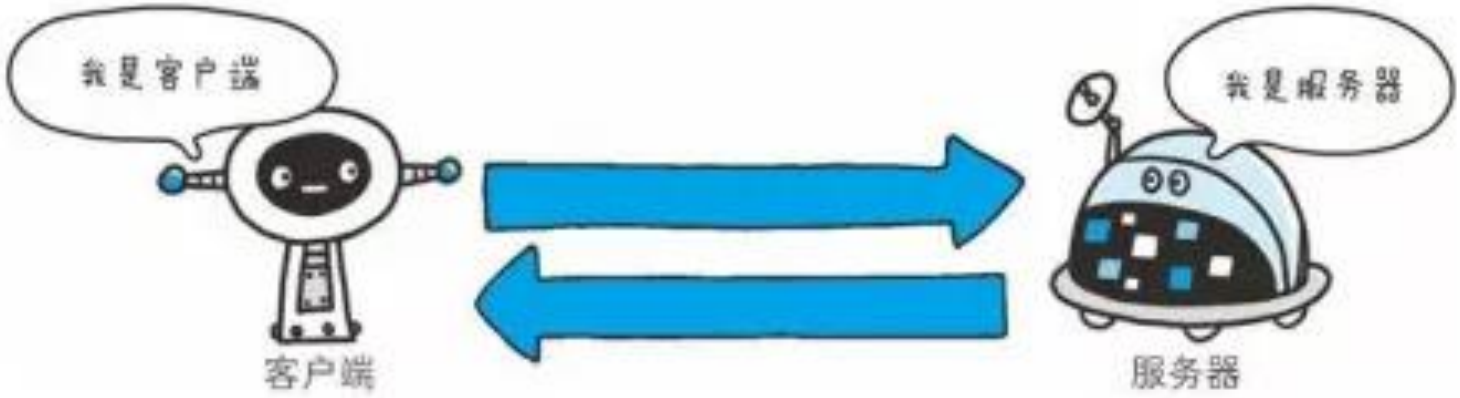
- 持续连接（HTTP keep-alive）



HTTP工作原理

HTTP工作原理

- 应用HTTP协议时，必定是一端担任客户端角色，另一端担任服务器角色
- 请求访问资源的一端称为客户端，提供资源响应的一端称为服务器端
- 通过请求和响应的交互达成通信，通信必定是从客户端开始建立的，服务端在没有接收到请求之前不会发送响应

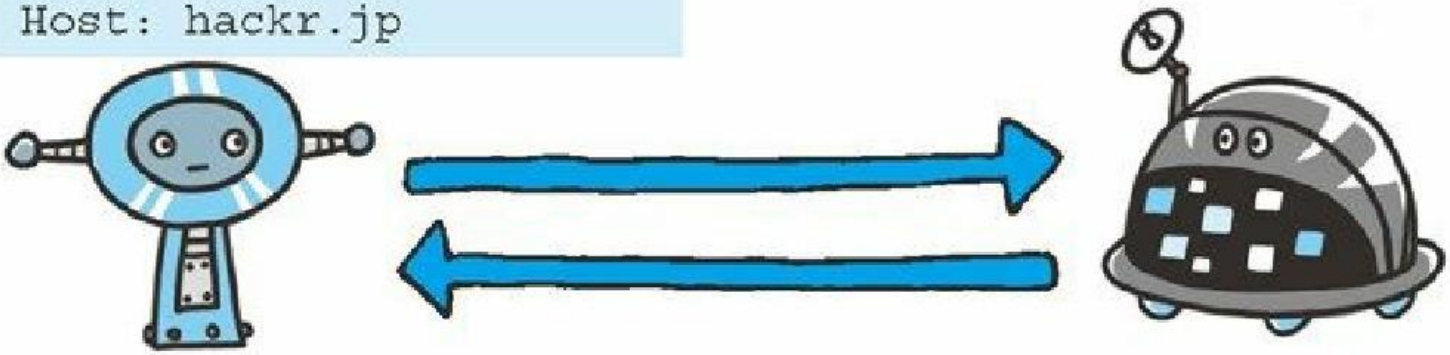


HTTP工作原理

- 请求访问某台 HTTP 服务器上的页面资源

①发送请求

```
GET / HTTP/1.1
Host: hackr.jp
```



客户端　　服务器

②发送响应

```
HTTP/1.1 200 OK
Date: Tue, 10 Jul 2012 06:50:15 GMT
Content-Length: 362
Content-Type: text/html
<html>
...
```

HTTP工作原理

- 在web浏览器输入某网站URL
  - 浏览器向 DNS 服务器请求解析该 URL 中的域名所对应的 IP 地址
  - 解析出 IP 地址后，根据该 IP 地址和默认端口 80，与服务器建立连接
  - 浏览器发送 HTTP 请求给服务器
  - 服务器对浏览器请求作出响应
  - 释放 TCP连接
  - 浏览器解析html 内容并显示

HTTP方法

HTTP方法

- HTTP/1.0和HTTP/1.1支持的方法

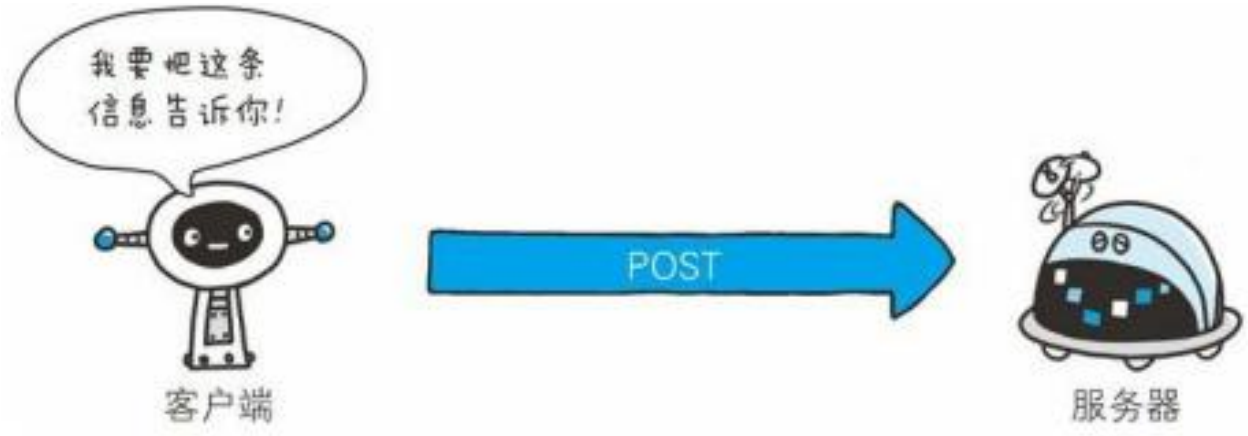
方法	说明	支持的 HTTP 协议版本
GET	获取资源	1.0、1.1
POST	传输实体主体	1.0、1.1
PUT	传输文件	1.0、1.1
HEAD	获得报文首部	1.0、1.1
DELETE	删除文件	1.0、1.1
OPTIONS	询问支持的方法	1.1
TRACE	追踪路径	1.1
CONNECT	要求用隧道协议连接代理	1.1
LINK	建立和资源之间的联系	1.0
UNLINE	断开连接关系	1.0

HTTP方法

- GET
  - 请求访问已被URI识别的资源
  - 指定的资源经服务器解析后返回响应内容

HTTP方法

- POST
  - 向指定资源提交数据进行处理请求（如提交表单）
  - 功能与GET很相似，但POST的主要目的并不是获取响应的主体内容

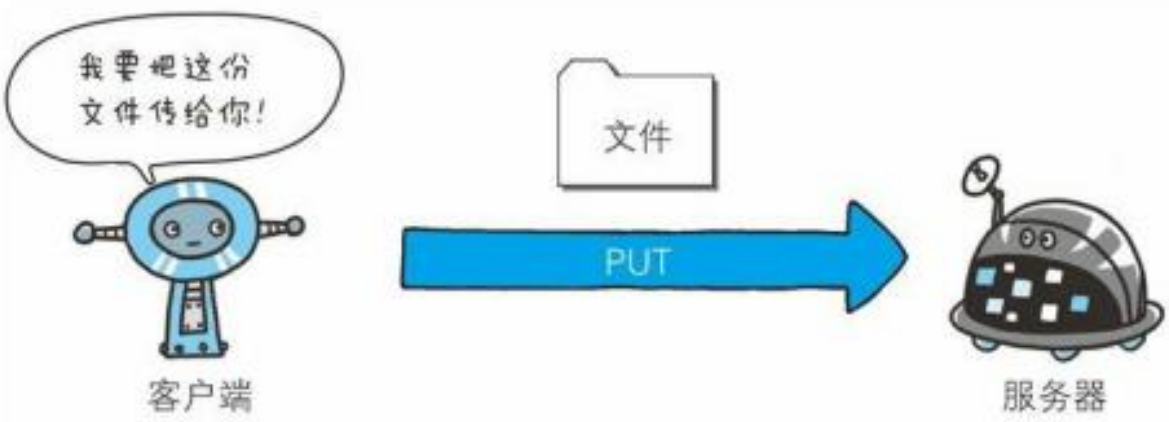


HTTP方法

- GET与POST的区别
  - GET提交的数据会放在URL之后，以?分割URL和传输数据，参数之间以&相连；POST方法是把提交的数据放在HTTP包的Body中。
  - GET提交的数据大小有限制，最多只能有1024字节（因为浏览器对URL的长度有限制）；POST方法提交的数据没有限制

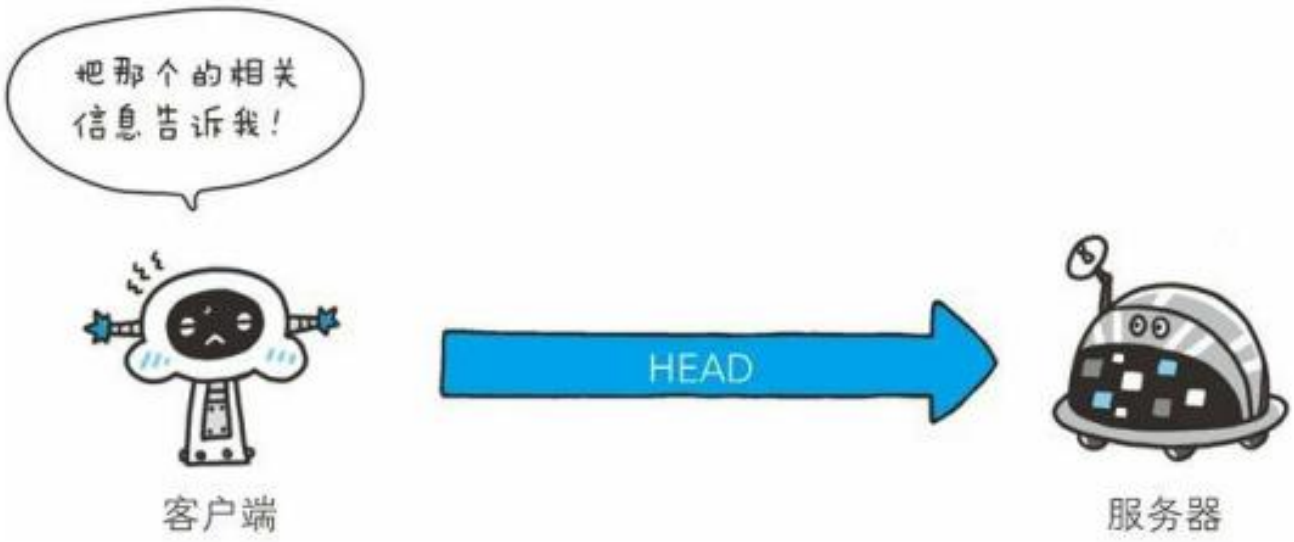
HTTP方法

- PUT
  - 用于传输文件，在请求报文的主体中包含文件内容，然后保存到请求URI指定的位置
  - 自身不带验证机制，任何人都可以上传文件，存在安全性问题，一般的web网站不使用该方法



HTTP方法

- HEAD
  - 和GET方法一样，但不返回报文的主体部分
  - 用于确认URI的有效性及资源更新的日期时间等



HTTP方法

- DELETE
  - 请求服务器删除指定的页面，与PUT方法相反
  - 不带验证机制，一般网站不使用
- OPTIONS
  - 查询针对请求URI指定的资源支持的方法
- TRACE
  - 追踪路径，让web服务器端将之前的请求通信环回给客户端，可以查询发送出去的请求是怎样被加工修改的
- CONNECT
  - 在与代理服务器通信时建立隧道，实现用隧道协议进行TCP通信
  - 主要使用SSL和TLS协议把通信内容加密后经网络隧道传输

HTTP报文

HTTP报文

报文首部

服务器端或客户端需处理的请求或响应的内容及属性

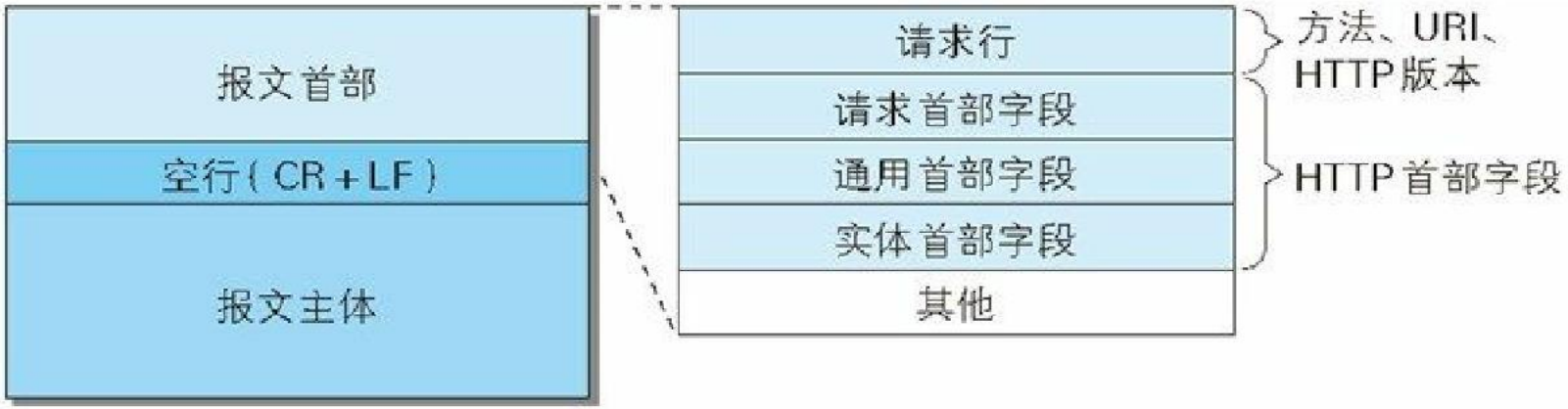
报文主体

应被发送的数据

- 报文首部与报文主体之间以空行（CR+LF)划分
- 不一定要有报文主体

HTTP报文

- 请求报文结构

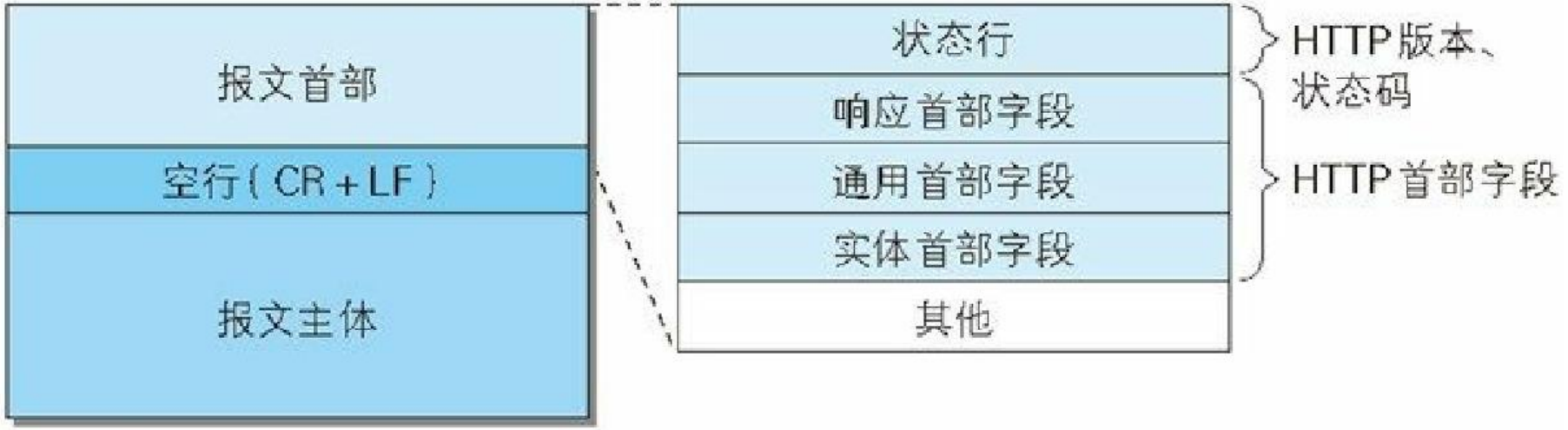


- 请求报文实例

```
GET / HTTP/1.1                                              请求行
Host: hackr.jp
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:13.0) Gecko/20100101 Firefox/13.0.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-us;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Pragma: no-cache
Cache-Control: no-cache                                     各种首部字段
空行（CR+LF）
```

HTTP报文

- 响应报文结构



HTTP报文

- 响应报文实例

```
HTTP/1.1 200 OK                                          状态行
Date: Fri, 13 Jul 2012 02:45:26 GMT
Server: Apache
Last-Modified: Fri, 31 Aug 2007 02:02:20 GMT
ETag: "45bae1-16a-46d776ac"
Accept-Ranges: bytes
Content-Length: 362
Connection: close
Content-Type: text/html                                  各种首部字段
空行(CR+LF)
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>hackr.jp</title>
</head>
<body>
<img src="hackr.gif" alt="hackr.jp" width="240" height="84" />
</body>
</html>                                                  报文主体
```

HTTP状态码

HTTP状态码

- HTTP状态码是由服务端返回给客户端的
- 表示客户端请求的返回结果、标记服务器端的处理是否正常、通知出现的错误等工作
- 以 3 位数字和原因短语组成
- 数字中的第一位指定了响应类别

	类别	原因短语
1XX	Informational（信息性状态码）	接收的请求正在处理
2XX	Success（成功状态码）	请求正常处理完毕
3XX	Redirection（重定向状态码）	需要进行附加操作以完成请求
4XX	Client Error（客户端错误状态码）	服务器无法处理请求
5XX	Server Error（服务器错误状态码）	服务器处理请求出错

HTTP状态码

- 2xx 成功
  - 200 OK
    - 表示从客户端发来的请求在服务器端被正常处理
  - 204 No Content
    - 表示服务器接收的请求已成功处理，但在返回的响应报文中不含实体的主体部分
    - 一般在只需要从客户端往服务器发送信息，而对客户端不需要发送新信息内容的情况下使用
  - 206 Partial Content
    - 表示客户端进行了范围请求，服务器成功执行
    - 响应报文中包含由 Content-Range 指定范围的实体内容

HTTP状态码

- 3xx 重定向
  - 301 Moved Permanently
    - 永久性重定向
    - 请求的资源已被分配了新的 URI，以后应使用资源现在所指的URI
  - 302 Found
    - 临时性重定向
  - 303 See Other
    - 由于请求对应的资源存在着另一个 URI，应使用GET方法定向获取请求的资源
  - 304 Not Modified
    - 表示客户端发送附带条件的请求时，服务器端允许请求访问资源，但未满足条件的情况
  - 307 Temporary Redirect

HTTP状态码

- 4xx 客户端错误
  - 400 Bad Request
    - 请求报文中存在语法错误
  - 401 Unauthorized
    - 发送的请求需要有通过 HTTP 认证的认证信息
    - 浏览器初次接收到401响应，会弹出认证用的对话窗口
    - 若之前已进行过 1 次请求，则表示用户认证失败
  - 403 Forbidden
    - 对请求资源的访问被服务器拒绝
  - 404 Not Found
    - 服务器上无法找到请求的资源
    - 也可以在服务器端拒绝请求且不想说明理由时使用

HTTP状态码

- 5xx 服务器错误
  - 500 Internal Server Error
    - 服务器端在执行请求时发生了错误
    - 也可能是 Web 应用存在的bug或某些临时的故障
  - 503 Service Unavailable
    - 服务器暂时处于超负载或正在进行停机维护，现在无法处理请求

HTTPS

- HTTP 缺点
  - 通信采用明文，不具备加密功能，内容可能被窃听
  - 不验证通信方的身份，因此有可能遭遇伪装
    - 无法确定请求发送至目标的 Web 服务器是否是已伪装的
    - 无法确定响应返回到的客户端是否是已伪装的
    - 无法确定正在通信的对方是否具备访问权限
    - 即使是无意义的请求也会照单全收，无法阻止海量请求下的 DoS 攻击
  - 无法证明报文的完整性，有可能已遭到篡改
    - 中间人攻击（Man-in-the-Middle attack，MITM）

HTTPS

- HTTPS = HTTP+加密+认证+完整性保护
- HTTP通信接口部分用SSL和TLS协议替代
- 安全可靠
- 开销较大，处理速度较慢
- 成本较高
- 非敏感信息一般使用HTTP通信，只有在包含个人信息等敏感数据时，才利用 HTTPS 加密通信
- 在进行加密处理时，并非对所有内容都进行加密处理，而是仅在那些需要信息隐藏时才会加密，以节约资源

道德经

大成若缺，其用不弊。
大盈若冲，其用不穷。
大直若屈。大巧若拙。
大辩若讷。静胜躁，寒胜热。清静为天下正。

谢谢！