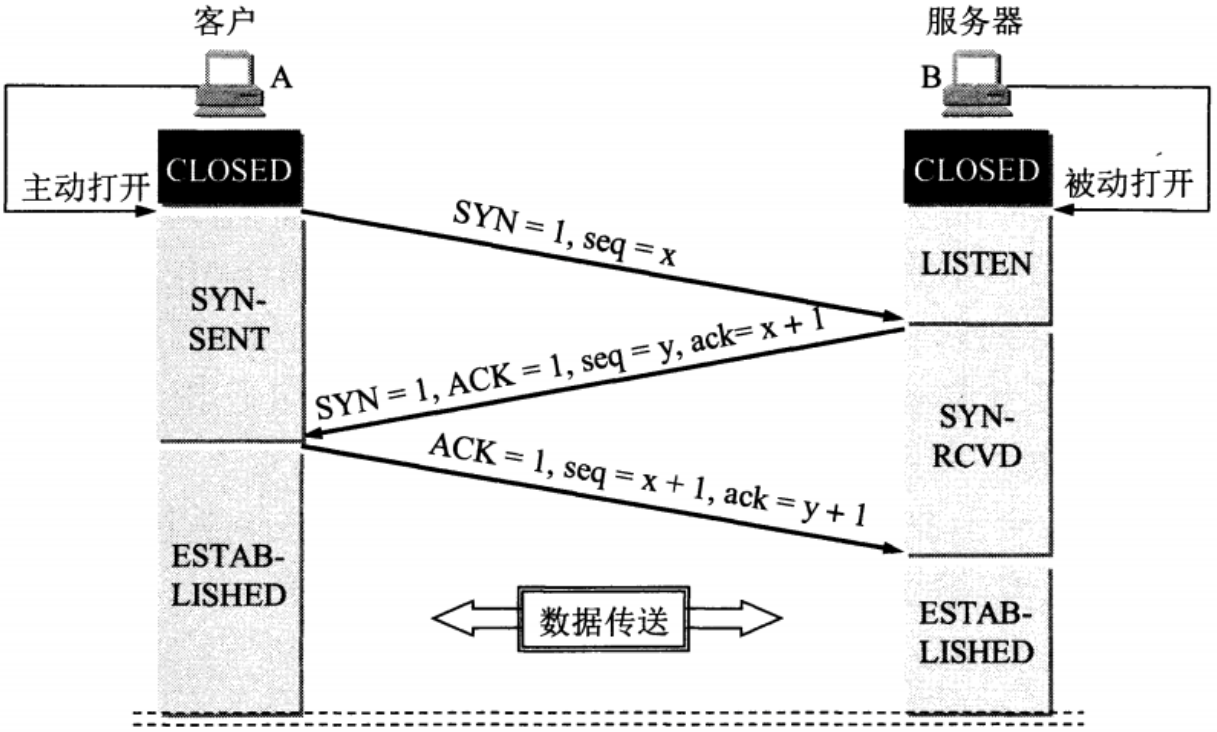
计算机网络

常见题目整理

常见的问题包括:TCP三次握手, 四次挥手, 网页访问的全过程, 拥塞控制, https, 几个协议的字段及其基本含义

1. 请讲一下TCP的三次握手:

    - 1 客户端发含SYN，序列号位x的报文到服务器. [客 -> SYN_SEND]
    - 2 服务器收到客户端发来的SYN请求,并回复SYN+ACK, 确认号为x+1, 序列号为y的报文给客户端(同意建立连接) [服 -> SYN_RECV]
    - 3 客户端收到服务器的SYN+ACK后,回复ACK, 确认号为y+1的报文给服务器(表示客户端收到了服务器端发的同意报文) [客 -> ESTABLISH]
    - 4 服务器收到客户端的ACK,连接已建立,可以数据数据传输 [服 -> ESTABLISH]

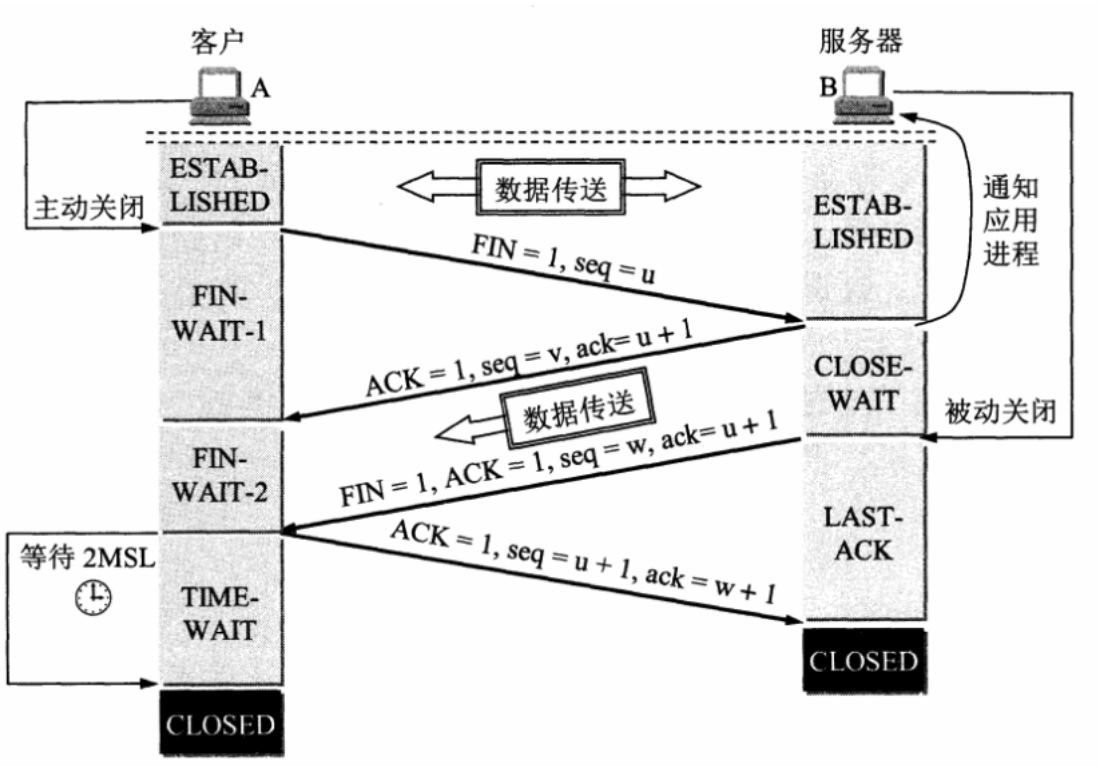


2. 讲一下四次挥手的过程:

    - 1 客户端发送FIN给服务器, 说明客户端不必发送数据给服务器了(请求释放从客户端到服务器的连接) (这个时候如果没发送完可以断开么,断开会把数据先发送完么???) [客-> FIN-WAIT-1]

- 2 服务器接受到客户端发送的FIN,回复ACK给客户端(同意释放从客户端到服务器的连接) [服->CLOSE-WAIT]
- 3 客户端收到服务器回复的ack,此时从客户端到服务器的连接已释放(但服务器到客户端的连接还未释放,并且客户端还可以接受数据); [客->FIN-WAIT-2]
- 4 服务器继续发送之前没发完的数据给客户端;
- 5 服务器发送FIN+ACK给客户端,说明客户端发送完了数据(请求释放从服务器到客户端的连接)[服->LAST-ACK]
- 6 客户端收到FIN+ACK,并回复ACK给客户端(同意释放从服务器到客户端的连接)[客->TIME-WAIT]
- 7 服务器收到客户端的ACK后,释放从服务端到客户端的连接.[服->CLOSED]
- 8 客户端等待2个MSL后,释放连接.[客-> CLOSED]

TCP 四次挥手释放连接



3. 讲一下网页是如何访问的:

- DHCP获取主机ip地址, 利用ARP请求获取网关的MAC地址.
- DNS域名解析,浏览器查找域名所对应的IP地址

        - 检查DNS缓存,如果之前登陆过这个网站, 那么DNS缓存中就会存有该链接对应的ip地址
        - DNS缓存中没有找到的话,就给本地DNS发送一个查询请求
        - 本地DNS服务器向根DNS服务器发送查询请求(根DNS服务器是域名解析的起点)
        - 根DNS服务器告诉本地DNS服务器,顶级DNS服务器是谁
        - 本地DNS向顶级DNS服务器发送查询请求
        - 顶级DNS返回权威DNS服务器的地址
        - 本地DNS向权威DNS服务器发送查询请求
        - 权威DNS服务器告诉本地DNS服务器最终的ip地址
    - TCP建立连接, TCP三次握手
    - 发送HTTP请求, 客户端发送请求报文
    - 服务器处理请求并返回HTTP响应报文
    - 浏览器解析渲染页面,客户端处理数据
    - 连接释放, 四次挥手

4. 讲一下拥塞控制?流量控制和拥塞控制的最主要的区别是什么?发送窗口的大小取决于流量控制还是拥塞控制?
    - 拥塞控制是为了防止过多的数据注入到网络, 整个网络的吞吐量随输入负荷增加而下降.
    - 具体的tcp拥塞控制算法分为几个主要的算法: 慢启动, 拥塞避免, 快速重传and快速恢复
        - (以TCP报文段的个数为讨论问题的单位，而不是以字节为单位)
        - 慢启动将拥塞窗口大小cwnd设置为1, 而发送窗口swnd等于cwnd, 因此发送方当前只能发送一个数据报文段,接收放收到该数据报后, 给发送方回复一个确认字段,发送方收到后,将拥塞窗口的值*2. 当拥塞窗口cwnd的值等于慢启动的门限值时, 改用拥塞避免算法.
        - 拥塞避免每个轮次cwnd就只会+1, 直到发送方判断出现了拥塞导致超时, 更改cwnd=1, ssthresh=cwnd/2, 并重新开始慢启动算法;

        - 如果出现了3次冗余ack,就会立即重传, 而不是等到超时重传计数器超时再重传, 并开始执行快速恢复算法, ssthresh=cwnd/2, cwnd=sshresh+3.

    - 而流量控制是接收方返回tcp报文时, 流量窗口的值, 是为了抑制发送端发送数据的速率, 以便是接收端来得及接收.
    - 发送窗口的大小等于Min[拥塞窗口，接收窗口]，因此是两种控制共同作用.

5. 简述一下https和http的区别:
    - https提供了端到端加密.
    - 安全套接字层SSL是运输层的安全协议。SSL作用在端系统应用层的HTTP和运输层之间，在TCP之上建立起一个安全通道，为通过TCP传输的应用层数据提供安全保障。应用层协议HTTP调用SSL进行加密，HTTPS（超文本传输安全协议）可以看作是提供安全服务的HTTP协议。
    - 区别：1、HTTPS协议要申请CA认证；2、HTTP是明文传输，HTTPS是加密的安全传输；3、HTTP使用80端口，HTTPS使用443端口；

6. 简述以下https的通信过程:
    - SSL握手:
        - 0 TCP三次握手
        - 1 客户机发送它支持的密码算法列表 + 客户机的不重数;
        - 2 从该列表中, 服务器选择1对称算法 + 1公钥算法 + 1MAC算法, + 证书 + 服务器不重数;
        - 3 客户机验证证书, 提取服务器公钥, 生成主密钥MS, 用服务器公钥加密, 发送;
        - 4 服务器使用相同的密钥导出函数, 客户机和服务器独立地从MS和不重数中计算加密和MAC密钥,
        - 5 客户机发送所有握手报文的一个MAC
        - 6 服务器发送所有握手报文的一个MAC
    - 连接关闭

- 1 发送TCP FIN, 并且ssl记录的类型字段指出该记录是否是用于种植该SSL会话的.

7. 主要的协议:

   (需要能指出字段的含义, 常用的取值, 长度)

   - HTTP

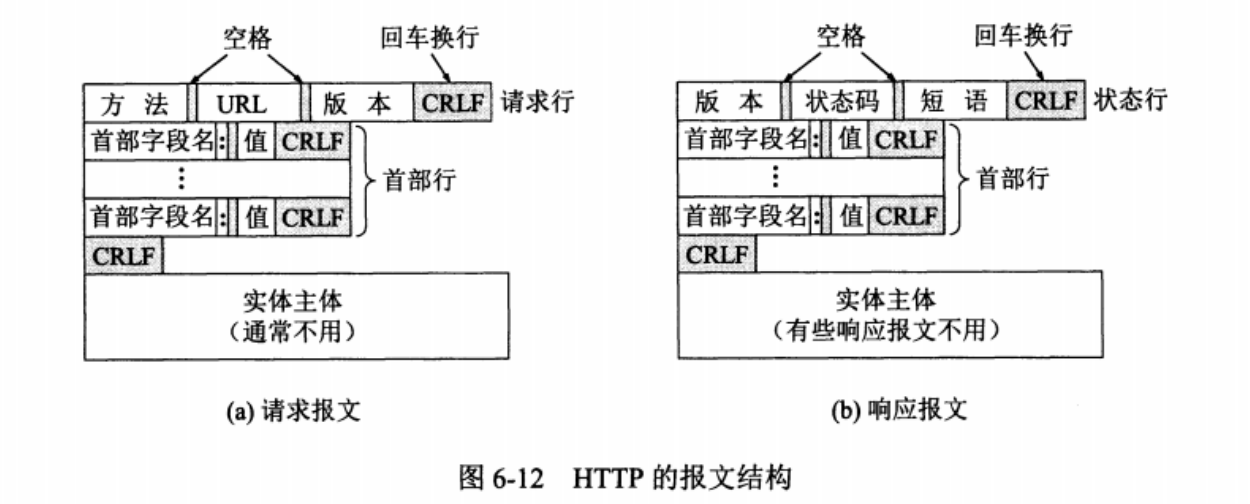


图 6-12 HTTP 的报文结构

   - TCP

图 3-3 TCP 头部结构

- UDP

图3-7 UDP报文段结构

- IPv4

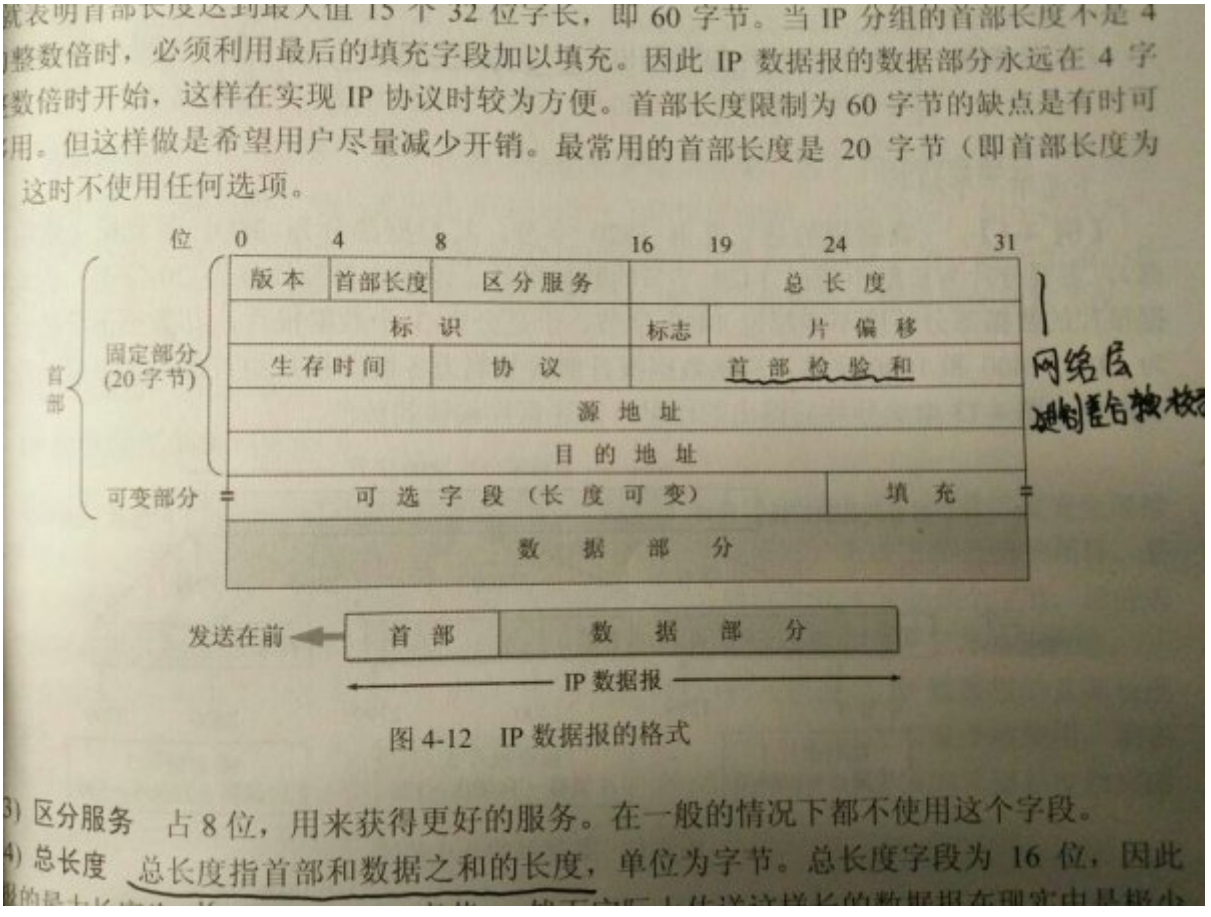


图 4-12 IP 数据报的格式

3) 区分服务 占 8 位，用来获得更好的服务。在一般的情况下都不使用这个字段。

4) 总长度 总长度指首部和数据之和的长度，单位为字节。总长度字段为 16 位，因此

- IPv6

计网知识细节整理

(这一块主要是个人的实际情况整理,略去了熟悉的内容,可参考.)

1. ARP协议流程:
   - 当主机 A 欲向本局域网上的某个主机 B 发送 IP 数据包时，就先在其 ARP 高速缓存中查看有无主机B 的 IP 地址。如有，就可查出其对应的硬件地址，再将此硬件地址写入 MAC 帧，然后通过局域网将该 MAC 帧发往此硬件地址。如没有， ARP 进程在本局域网上广播发送一个 ARP 请求分组。主机 B 单播发送一个 ARP 响应分组给主机 A,其他主机不会理睬这个请求分组，主机 A 将得到的 IP 地址和路由的下一跳硬件地址写入 ARP 高速缓存

2. http webservice socket的区别:
   - Socket是基于TCP/IP的传输层协议。WebService是基于HTTP协议传输数据的，HTTP是基于TCP的应用层协议。
   - WebService采用了基于HTTP的SOAP协议传输数据，接口支持面向对象，最终WebService会将对象进行序列化后通过流传输。Socket接口直接通过流进行传输，不支持面向对象。Socket在传输层，WebService在应用层。http 协议基于socket，此外，web service基于http协议和soap。

3. TCP RST标志位的作用:
    - RST表示复位，用来异常的关闭连接，在TCP的设计中它是不可或缺的。就像上面说的一样，发送RST包关闭连接时，不必等缓冲区的包都发出去（不像上面的FIN包），直接就丢弃缓存区的包发送RST包。而接收端收到RST包后，也不必发送ACK包来确认。
    - TCP处理程序会在自己认为的异常时刻发送RST包。例如，A向B发起连接，但B之上并未监听相应的端口，这时B操作系统上的TCP处理程序会发RST包。
    - 又比如，AB正常建立连接了，正在通讯时，A向B发送了FIN包要求关连接，B发送ACK后，网断了，A通过若干原因放弃了这个连接（例如进程重启）。网通了后，B又开始发数据包，A收到后表示压力很大，不知道这野连接哪来的，就发了个RST包强制把连接关了，B收到后会出现connect reset by peer错误。

4. socket客户端关闭时,缓冲区的数据还会传输么?
    - close只有在引用计数为0时，才会真正调用close（ ），否则只是引用计数减1 。调用close（ ），系统会尝试发送完内核缓冲区内所有数据，然后才会发送FIN。
    - shutdown不理会引用计数和内核缓冲区内的剩余待发数据包，直接发送FIN。shuwdown可以只关闭套接字某个方向的连接，例如关闭发送，关闭接收，或两者都关闭。
    - 让套接字直接发送RST，从而没有FIN的发送，接收方返回ECONRESET错误，连接直接关闭。RST在TCP协议中表示复位，用来异常的关闭连接。发送RST包关闭连接时，不必等缓冲区的包都发出去，直接就丢弃缓冲区的包发送RST包，接收端接收到RST包以后，也不必发送ACK包确认。

5. https通信过程(概要版)
    - 认证服务器: 一些主流浏览器会内置一个受信任的CA机构列表，并会保存相关CA机构的https证书。当用户在访问部署了https证书的网站时，服务器会提供经CA机构颁发的https证书，如果认证该服务器证书的CA机构是存在于浏览器的受信任CA机构列表当中，并且该https证书中的所有信息均与当前证在访问的网站所有信息一致，那么浏览器就会认为

服务端是可信的，并从https证书中取得公钥（也就是CSR文件），用于后面的流程。

- 协商会话秘钥: 在服务器认证完获取公钥之后，利用公钥与服务器进行加密通信，协商出两个会话秘钥，用于加密客户端和加密服务端互发数据时的会话秘钥。这个秘钥是随机生成的，每一次协商产生的结果都不一样，所以安全性也是比较高的。
- 加密传输: 当客户端和服务器端都拥有了协商的会话密钥之后，进行数据传输时，都是以密文的方式进行传输。这样的传输方式，保证了数据的私密性和完整性，再也不用担心数据在传输过程中被第三者窃取和篡改了。

6. session和cookie的区别
    - 储存位置：Cookie是客户端会话技术，数据保存在客户端，Session是服务器端会话技术，数据保存在服务器端。
    - 存储容量：Cookie一般<=4KB，Session无限制。
    - 跨域支持：Cookie支持跨域，Session不支持。

7. 广播, 单播和多播: 广播是1对所有, 单播是1对1, 多播是1对一组.