```
功能:决定了向用户提供应用服务时通信的活动
                           应用层 が议:Http、FTP、POP3、STMP
                           功能:提供处于网络连接中的两台计算机之间的数据传输
                  四层模型 🏻
                                  协议:TCP、UDP
                           网络层 ⊙ 协议:IP
                  ※ WHE、用來处增连接网络的硬件部分
客户端:发送带SYN标志的数据包,作为请求表示希望进行同步连接。服务端:接收到请求后,发送带SYN标志和ACK标志的数据包,作为响应表示可以进行同步连接。
客户端:接收到响应后,发送带ACK标志标志的数据包,表示收到服务铺的响
TCP协议 ◎ 三次握手 ◎ 应.
(TCP/IP)◎ 基本的
                  URI-统一资源标识符;URL-统一资源定位符
          无状态协议、TCP/IP应用层协议 ○ Cookie技术
                           特点:只要任意一端没有明确提出断开连接,则保持TCP连接状
          持久连接(HTTP/1.1) 

好处:减少建立TCP连接的额外开销、减轻服务器端的负载、Web页面的显示速度也相应提高了

Connection:keep-alice
          管线化 ○ 客户端可以并行发送多个Http请求
          传输前:内容编码
过程中:分块传输编码
                                                                                   GET ○ 获取资源、没有报文主体、幂等的
                                                                                  POST ⊙ 传输实体主体、非幂等
                                                      请求方法 URI 协议版本
                                                     请求首部字段
通用首部字段
实体首部字段
                                                                                  PUT ○ 传输数据取代指定的文档的内容
                                             请求行
                                             请求头部 <sub>①</sub>
                                                                                   DELETE ○ 删除文件
                                                                                  HEAD ◎ 获取报文头部
                                     请求报文 ② 报文主体
                                                      报文主体
                                                                          请求方法 ◎
                                                                                  OPTIONS © 询问支持的请求方法
                                                                                   要求在代理服务器通信时建立隧道,实现用隧道协议进行TCP通
CONNECT ○ 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
                                                                                   追踪路径,让Web服务器将之前的请求通信环回给客户端、易受
TRACE ② XST政击
                                                                                             1XX:信息性状态码,接收的请求正在处理
                                                                                                                        HTTP报文:用于HTTP协议交互的信息
                                                                                                                       204 No Content:请求正常处理,但没有资源(实体主体)返回
                                                                                             2XX:成功状态码,请求正常处理完毕 ©
                                                                                                                        206 Partial Content:表示执行了范围请求,请求正常处理,返回了这部分实体主体
                                                                                                                       301 Moved Permanently:永久性重定向,请求的资源永久的
                                                      协议版本 状态码 状态码的原因短语 (如OK
                                                                                                                       改变了uri
                                                                                                                       302 Found:临时性重定向,请求的资源被分配了新的URI,但这个
                                              响应头部 ⊙
                                                                                                                       URI是临时性的
                                                      实体首部字段
                                                                                      状态码 。 3XX:重定向状态码,需要附加操作 。
                                                                                                                       303 See Other:类似302,但希望用户以Get方法去请求资源
                                     响应报文 ⊙
                                             报文主体
                                                                                                                      304 Not Modified:表示未满足请求的条件
                                                                                                                      307 Temporary Redirect:同302,但禁止重定向时将原本POST请求变成GET
                                                                                                                              400 Bad Request:请求报文存在语法错误
                                                                                                                              401 Unauthorized:表示请求没有通过HTTP认证或认证失败
                                                                                             4XX:客户端错误状态码,服务器无法处理请求《
                                                                                                                             403 Forbidden:服务器拒绝了你的请求
                                                                                                                             404 Not Found:不存在的资源
                                                                                                                              500 Internal Server Error:服务器执行请求时出错
                                                                                            SXX:服务端错误状态码,服务器处理请求出错 533 Service Unavailable:服务器暂时处于超负载或进行停机维护中
                                 告知服务器,客户端能够处理的媒体类型,以及媒体类型的相对优先级,一次 Accept \odot 可指定多种媒体类型
                                 Accept-Charset:告知服务器,客户端接受的字符集
                                 Accept-Encoding:告知服务器,客户端支持的内容编码
                      请求首部字段 ⊙
                                 Accept-Language:告知服务器,客户端支持的自然语言
                                 Host: 请求资源所在的服务器
                                 User-Agent:用来传达浏览器的种类
                                 Cookie:服务器将会接到的Cookie信息
                                 Accept-Ranges:可接受的字节范围
                                 Age:告知客户端,源服务器在多久前创建了响应,单位是秒
 Http
                      响应首部字段 O Location:配置重定向状态码,令客户端重定向到指定的URI
                                 Server: Http服务器的安装信息
                                 SetCookie: 开始状态管理所使用的Cookie信息
          Http报文首部(
                                                             Cache-Control: no-cache
目的:防止从缓存中返回过期的资源
                                 请求中:表示不接受缓存
Cache-Control:通过命令进行缓存控制 ② 响应中:可以缓存,但要求每次使用前向源服务器进行校验?
                                 Connection:管理持久连接;控制不再转发给代理的首部字段
                      通用首部字段 ③
                                 Date: 创建报文的时间
                                 Transfer-Encoding: 规定报文主体传输时的编码格式
                                 Allow:资源可支持的HTTP方法
                                 Content-Type: 实体主体内对象的媒体类型
                                 Content-Encoding: 实体主体内选用的内容编码方式
                      实体首部字段 ©
                                 Content-Language: 实体主体使用的自然语言
                                 Content-Length: 实体主体的字节数
                                 Content-Range:针对范围请求,实体主体的位置范围
                     通信使用明文(不加密),内容可能会被窃听
                    不验证通信方的身份,有可能遭遇伪装
                    无法证明报文的完整性,可能遭到篡改
          Http的缺点
                                Ajax
                    HTTP/1.1瓶颈 O SPDY协议:应用层和传输层间新加会话层
                                                      目的:解决XHR附带的缺陷引起的问题
                                WebSocket协议:全双工通信 O Http连接建立后一次握手,Upgrade:websocket
                                                    通信时使用websocket自己的数据帧
                       HTTP + 加密 + 认证 + 完整性保护
                 组成 OHTTP + SSL + TLS
                       共享密钥加密(对称) © 加密和解密采用同一个密钥
                       公开密钥加密(非对称)○ 采用一对非对称的密钥,私有密钥由发布方自己保管,公开密钥可随意发布
                      数字证书认证机构 ○ 对公开密钥做数字签名,将该公开密钥和公钥证书绑定
                               首先采用公开密钥加密,服务器向数字证书认证机构发送自己的
公开密钥,认证机构用自己的私有密钥问服务器的公开密钥做数
字签名并排上数字证书。 席户端拿到服务器的公钥证书后,向数
字证书认证机构进行验证,确认后使用公钥对报文加密后发送。
服务器使用私钥对报文解密
          Https
                 采用混合加密机制 · 确认交换的密钥是安全后,使用共享密钥加密的方式进行通信
                 SSL ② 安全套接字层协议,采用数据加密技术,保证数据在网络传输时不会被截取
                 TLS ○ 传输层安全协议,用于两个应用程序之前提供保密性和数据完整性
                BASIC认证 ② Authorization首部 + Base64编码
                质询响应:
认证要求-->
<-- 质询码
DIGEST认证 ② 响应码 -->
          认证 🤄
                SSL客户端认证
                FormBase认证
                XSS:跨站脚本攻击
                SOL注入政击
          安全 🤄
                OS命令注入攻击
```

HTTP头部注入攻击 ○ 如:Location、Set-Cookie