1. 应用层的协议数据是如何经局域网传输的？
2. 结合已经学习过的各层次协议，试分析一下发送一个邮件或访问一个网站时可能会使用到的各个协议

一、 应用层功能概述

- 应用层是网络体系结构中的最高层，为最终用户提供服务
- 应用层并不是针对每一项网络应用服务制定的标准

p 常见网络应用

- E-mail
- Web
- QQ/MSN
- 远程登录
- P2P文件共享
- 联网游戏
- YouTune/土豆网
- 网络电话
- 实时视频会议
- 大规模并行计算

二、网络的计算和访问模式

- 以服务器为中心的计算模式-资源共享(resource-sharing)模式

  共享共同的应用，如文件服务器，打印服务器

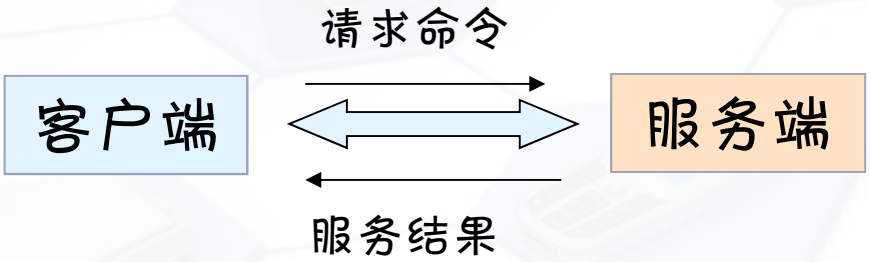
- 客户/服务器(Client/Server)模式

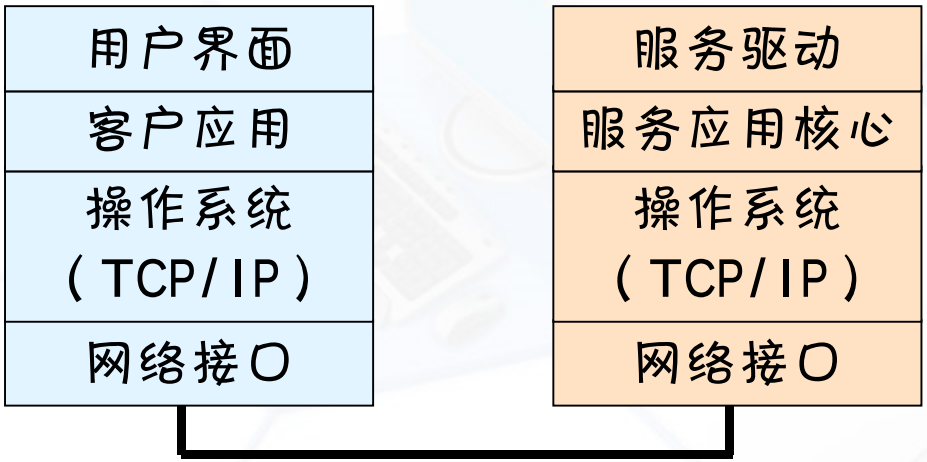
  系统使用了客户和服务器双方智能、资源和计算能力来执行一个特定的任务

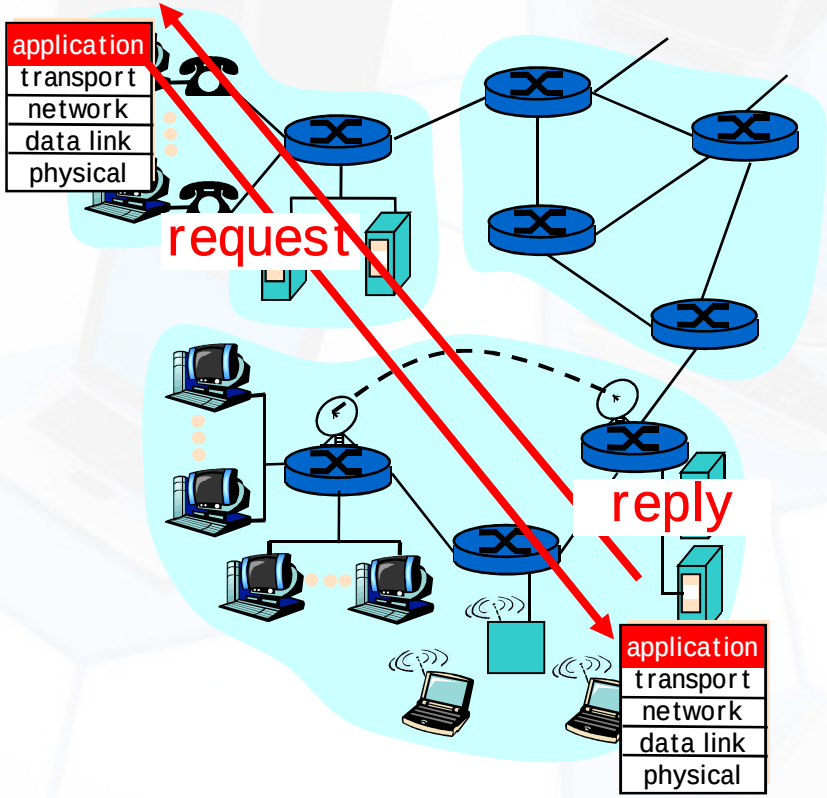
- 对等(Peer to Peer，P2P)模式

  用户和资源处于对等状态，分布式计算

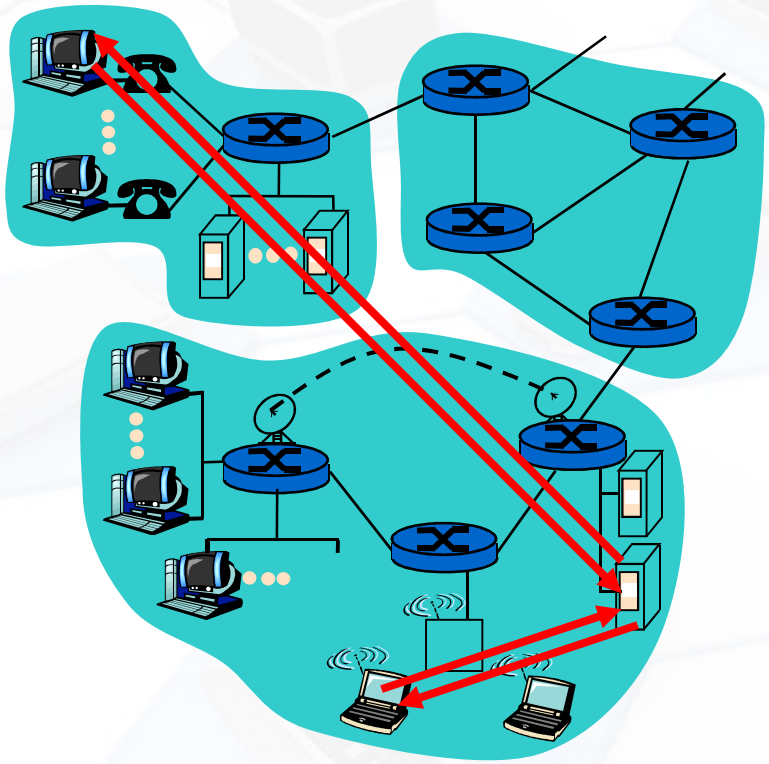
1. 客户/服务器模式(Client/Server)







p Client-server 结构的特征



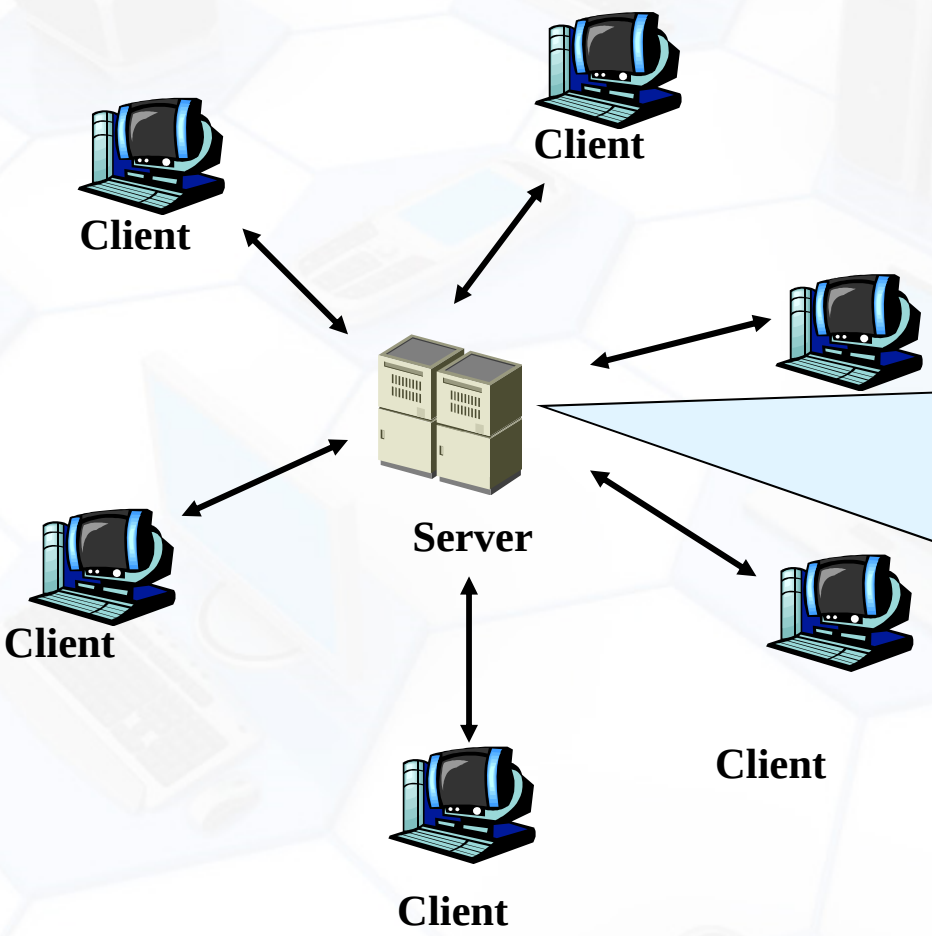
Server端:

- § 保持开机状态
- § 需要永久性的IP地址
- § 能够处理大规模的访问

Client端:

- § 与服务端进行通讯
- § 与服务的连接具有间歇性
- § 可能使用动态IP地址
- § 客户端之间没有直接通讯

p 客户/服务器模式的逻辑示意



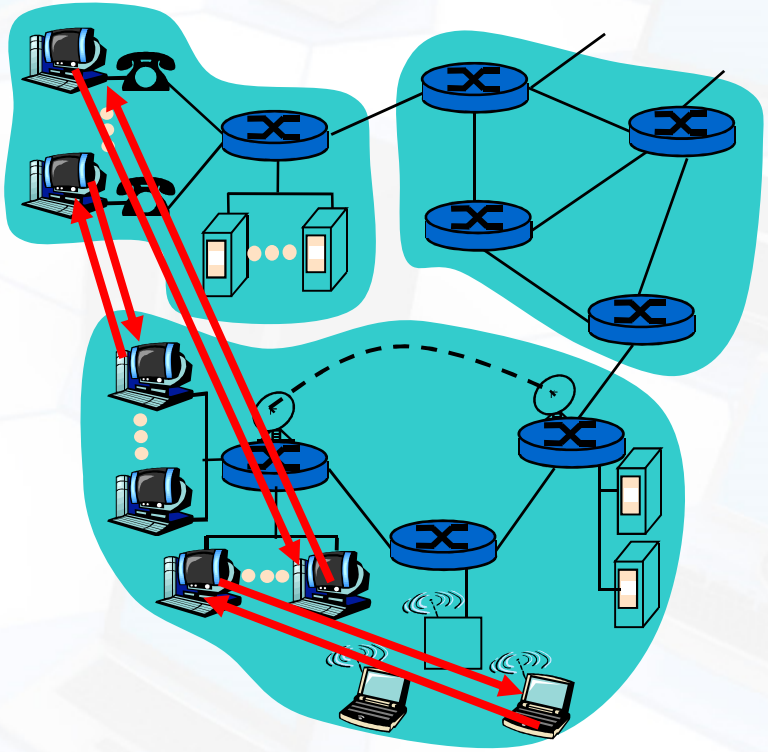
问题:
- 单点失效
- 性能瓶颈

(计算、存储资源受限，网络拥塞等)

2. 对等模式(Peer to Peer，P2P)

- P2P通信模式中各方都具有相同的能力，其中任何一方都可以发起一个通信会话
- 在P2P通信过程中，每个通信节点同时具有服务器和客户端的功能
- P2P网络中的节点间采用P2P通信模式，它是构筑在现有网络基础设施上的一个重叠网络(Overlay Network)

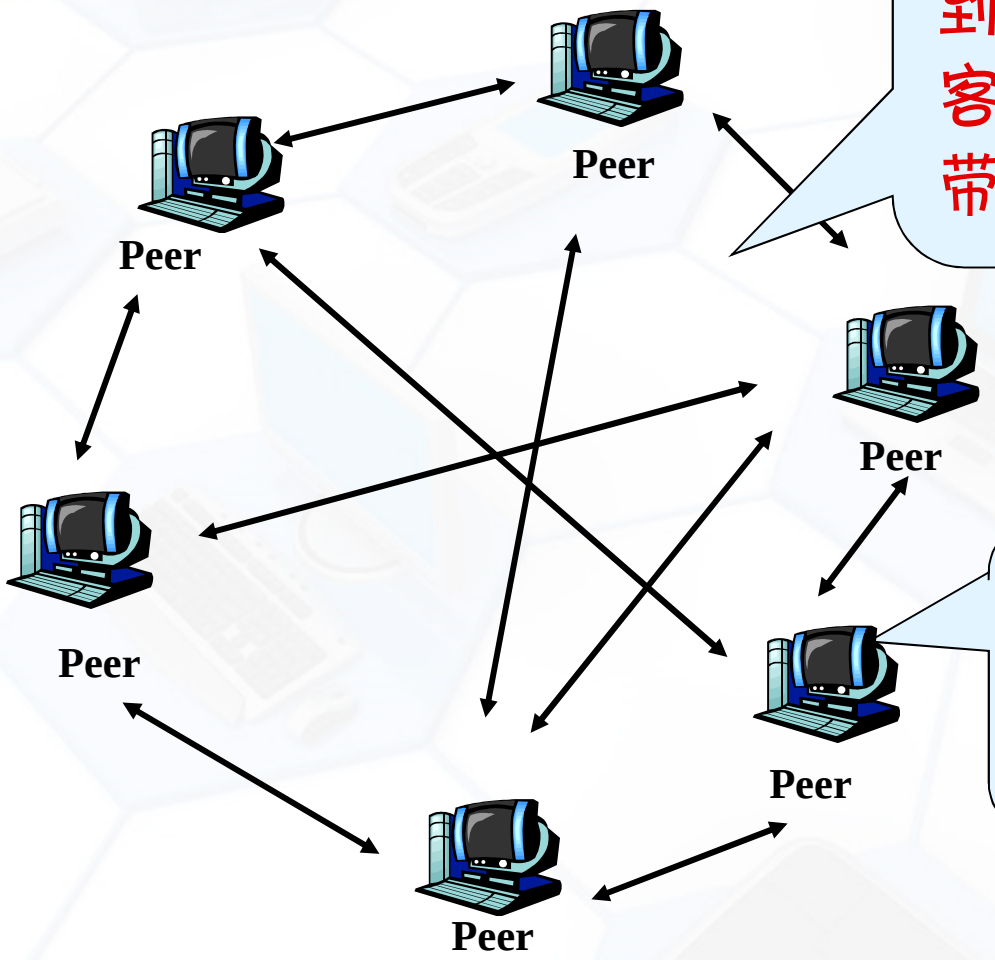
p P2P结构的特征



- § 不需要始终运行的主机
- § 终端系统间随意直接通信
- § 对等节点间歇性的相互连接和交换IP地址
- § 极具扩展性但难于管理

- § P2P网络是一个应用层网络，一般由网络边缘节点构成，充分利用资源
- § 网络的扩展性好
- § 资源分布在各个节点中，不是集中在一个服务器上，不存在单点瓶颈

p 对等模式的逻辑关系



将服务器的功能分散到客户端，充分利用客户端的计算、存储、带宽等资源

无中心服务器
Peer端既是客户端，又是服务器

p P2P连接资源的方式

过程：发布 à 定位 à 使用或者下载

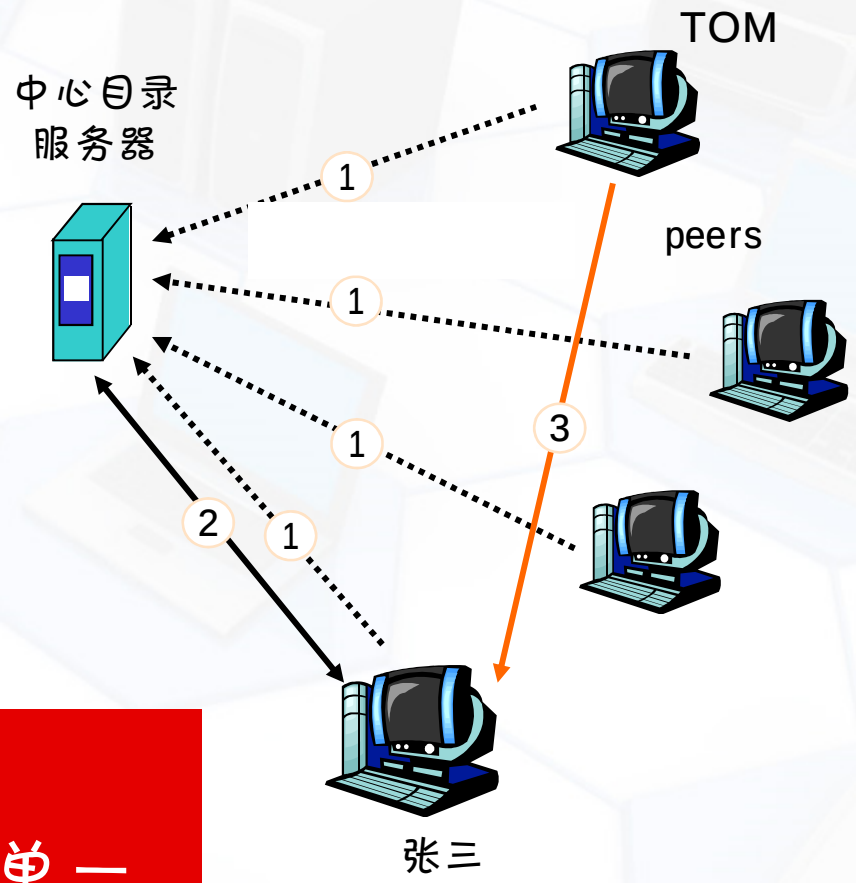
- § 基于目录服务器
  - ü 使用中心目录服务器用于资源发布和定位
  - ü SETI@home、Napster、Groove
- § 完全分布式
  - ü 无任何中心服务器，资源的定位使用泛洪
  - ü Gnutella
- § 层次结构
  - ü 将节点分为一般节点和超级节点，一般节点通过超级节点来发布和定位资源，超级节点之间采用泛洪方式来定位资源
  - ü KazaA、Skype
- § 结构化P2P
  - ü 将资源和其存储位置关联起来
  - ü Chord、CAN、Tapestry、Pastry

(1) 基于目录服务器--Napster

① 每个节点都会向中心服务器通告其IP地址，具备的资源

② 张三通过服务器查询所需要的MP3

③ 张三需要的文件由TOM的主机传输

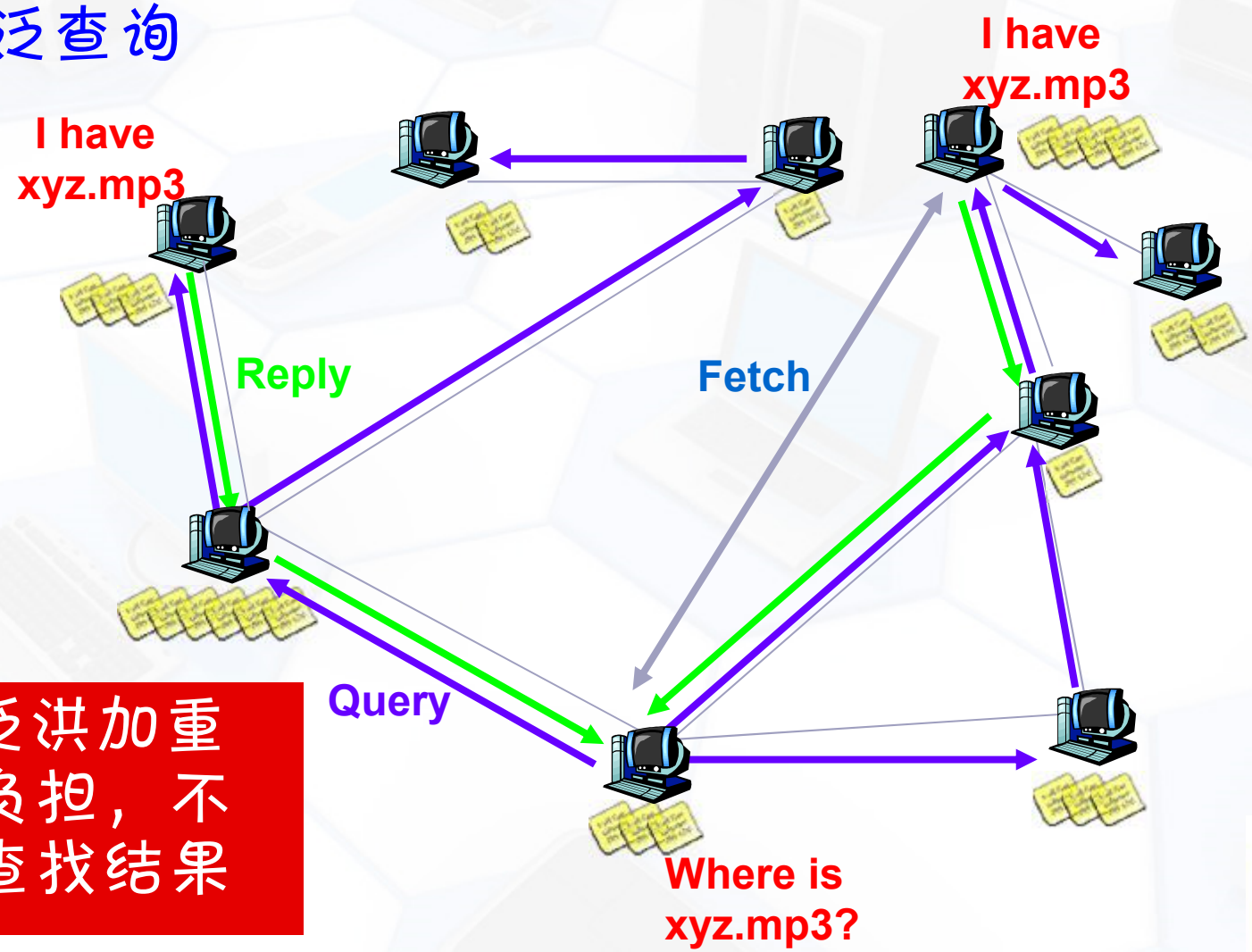


优点：查找简单，高效

缺点：目录服务器是瓶颈，单一故障点，不具可扩展性

(2) 完全分布式-- Gnutella

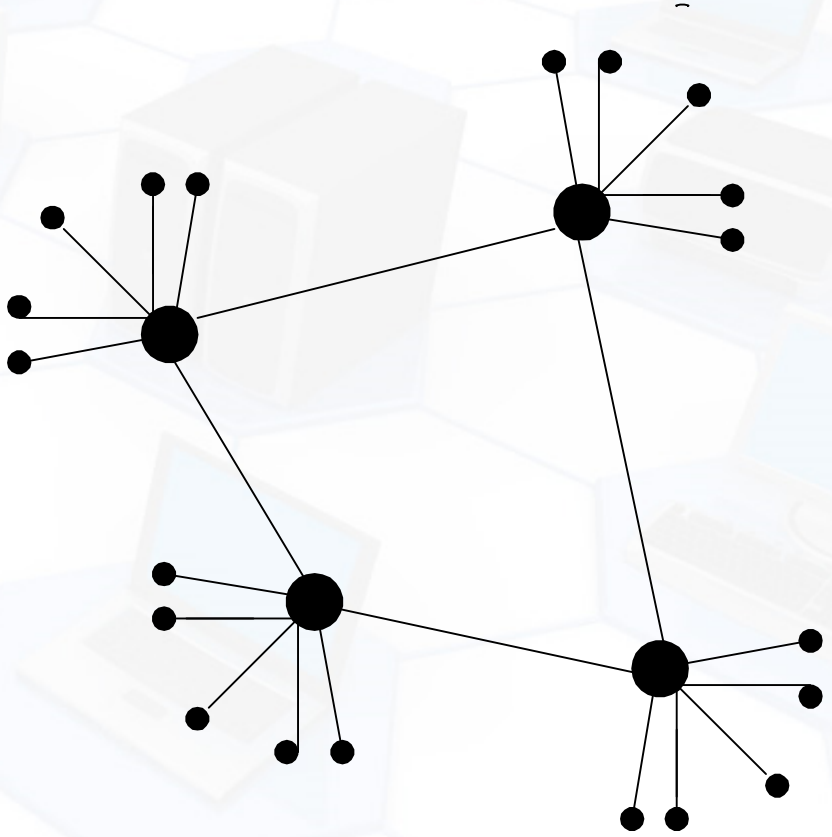
- 无中心服务器，无单点瓶颈
- 洪泛查询



缺点：泛洪加重网络负担，不保证查找结果

(3) 层次P2P网络-- KazaA

- p 每个peer 要么是组长，要么连接到组长
- ü Peer和其小组长之间通过TCP连接
- ü 某些小组长间也是由TCP连接
- p 组长负责收集其组员的信息



考虑了节点能力的不同，对泛洪有所改善

查找时间和范围具有不确定性

(4) 典型的P2P应用


easyMule


ERYCD
分享 互联网


BitComet
c++ bittorrent client


BitTorrent™

(5) client-server与P2P的混合结构

Skype

- 通过中心服务器查找远端节点的地址
- 节点之间直接连接 (不通过服务器)

即时消息

- 两个用户之间的交谈是P2P方式
- 用户在线的检测和定位则通过中心服务端:
  - 用户在线时向中心服务器登记其IP地址
  - 用户从中心服务器查找其同伴的IP地址

二、 TCP/IP的应用层

典型的应用层协议的功能和作用

协议名称	全　称	功　能
DNS	Domain Name System	实现域名和IP地址之间的相互转换
HTTP	Hypertext Transfer Protocol	用于在在器和WWW服务器之间传送超文本文件
SMTP	Simple Mail Transfer Protocl	实现电子邮件传输
FTP	File Transter Protocol	实现文件传输
SNMP	Simple Network Management Protocol	提供了一种监控和管理计算机网络的有效方法
TFTP	Trivial File Transfer protocot	建立在UDP协议之上用于提供小而简单的文件传输服务
DHCP	Dynamic Host Configuration Protocol	给网络客户机分配动态的IP地址

1. DNS服务

- 域名系统DNS (Domain Name System)的作用: 将主机名映射为IP地址
- 域名:用容易记忆的ASCII串符号来指代IP地址，方便用户访问网络

1) DNS结构

- DNS包括域名、主机和域名服务器三大要素
- 任何一个连接在因特网上的主机或路由器，都有一个惟一的层次结构的名字，即域名(domain name)
- 域名结构:采用层次型命名机制的命名方法，由若干分量组成,由点隔开:

…….三级域名.二级域名.顶级域名

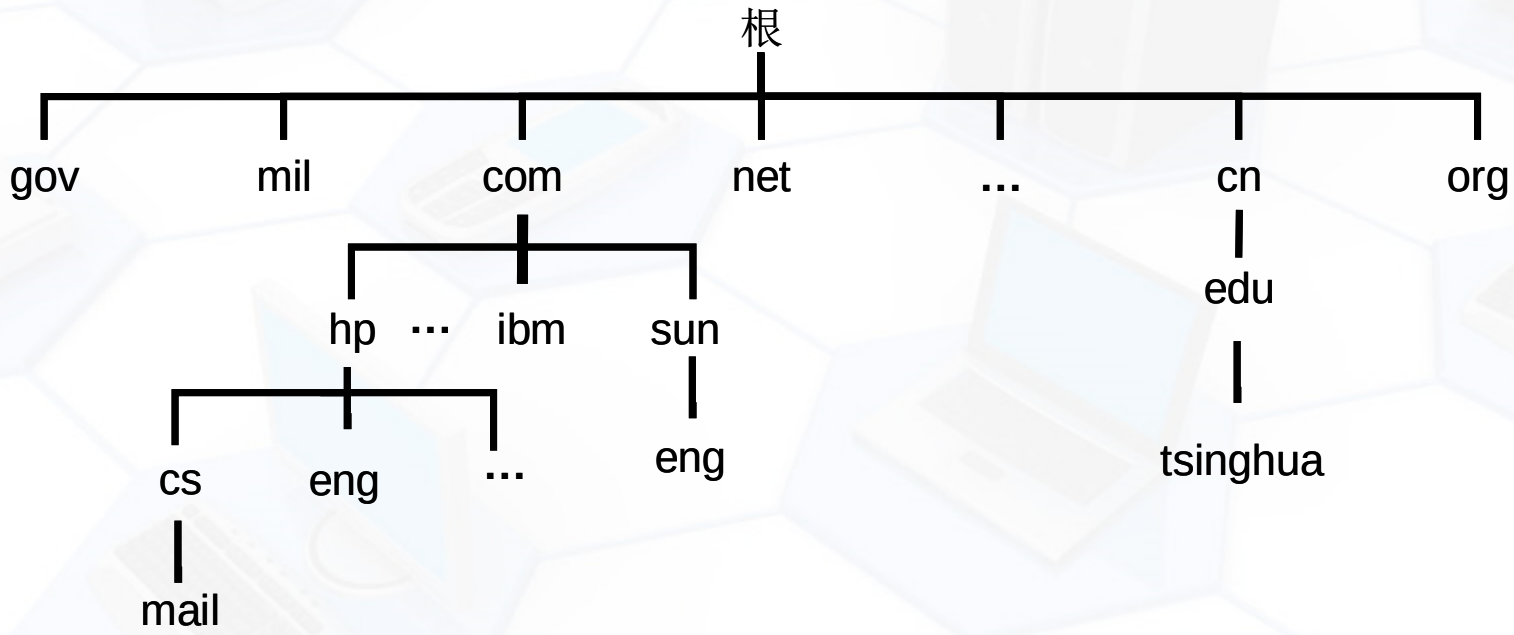
- 顶级域名

由因特网Internet的域名管理机构管理，现在的顶级域名可分为三大类

类　　别	常 用 域 名 举 例
国家顶级域名	.cn—中国、.us—美国、.uk—英国
国际顶级域名	.int—国际性的组织
通用顶级域名	.com—公司、.net—网络服务机构、 .edu—教育部门、.gov—政府部门、 .org—非赢利组织、.mil—军事部门、 .biz—商业组织、.info—信息服务、 .name—个人域名、.coop—商业合作组织 .pro—律师、医生等专业人员、 .museum—博物馆及文化遗产组织

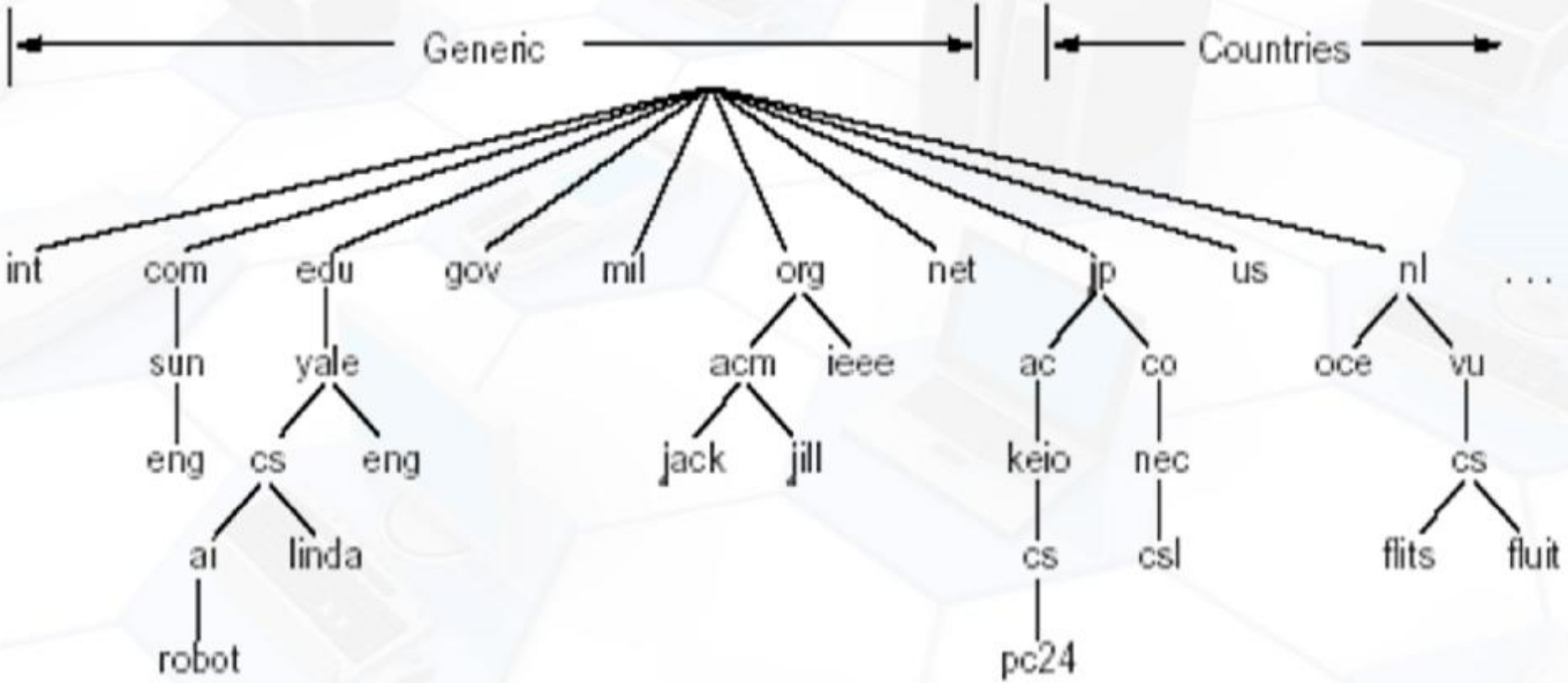
p 因特网名字空间的结构

采用分级管理的方式，整个域名空间形状类似一棵倒立的树



域名只是逻辑概念，并不反映出计算机所在的物理网络

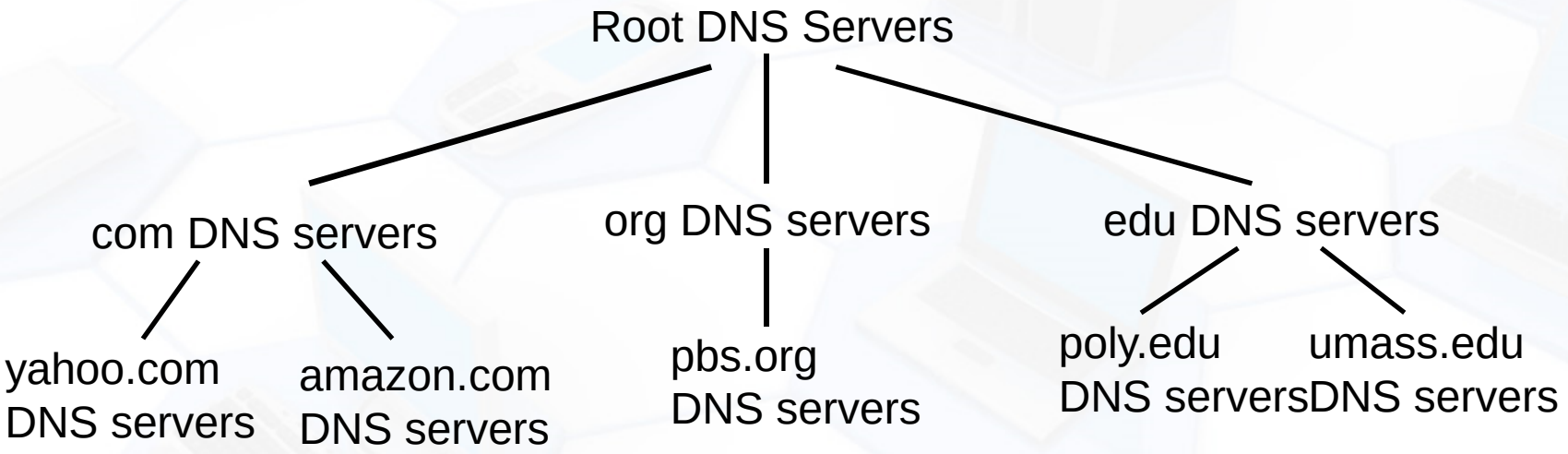
p Internet的域名空间



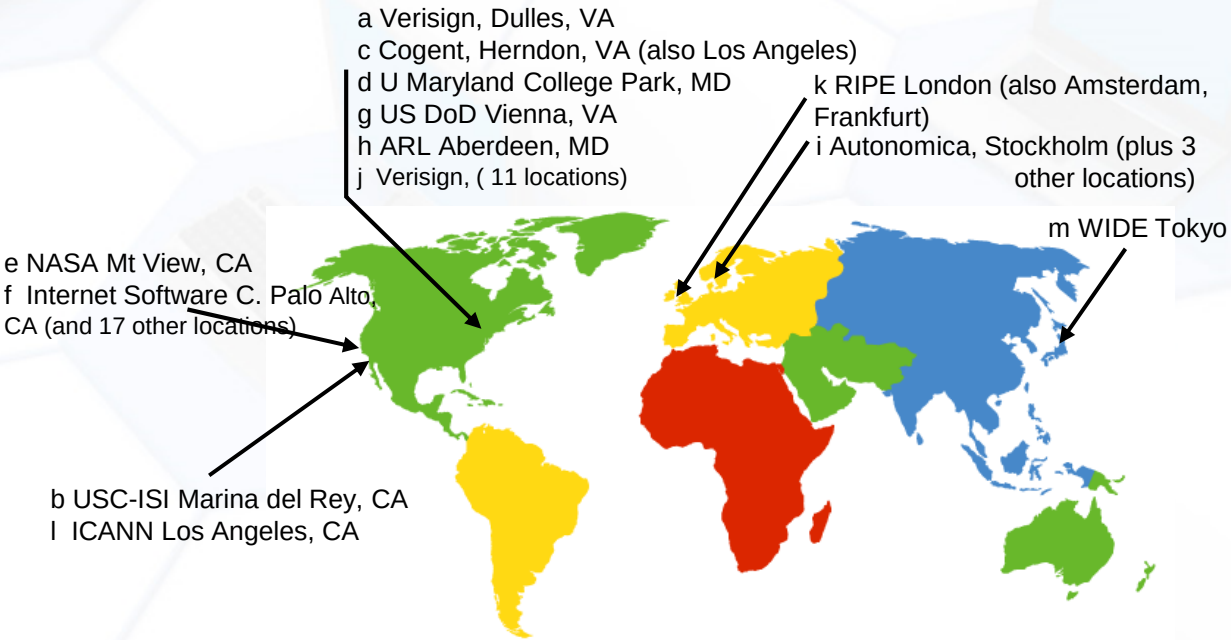
2) 域名解析

- § DNS系统采用了与分级结构的域名空间相对应的方式，即层次化模式，类似于分布式数据库查询系统
- § 提出DNS解析请求的主机与域名服务器之间采用客户机-服务器模式工作
- § 域名解析的查询过程可分为两部分：
  - ü 一是客户端本身及客户端对服务器的查询
  - ü 二是服务器与服务器之间的查询

p DNS—分布式、层次化的数据库



- DNS: 根域名服务器
  - 为本地域名服务器提供解析服务
  - 根域名服务器的任务
    - 向权威域名服务器寻求尚不知晓的域名映射
    - 获取映射信息
    - 将结果反馈给本地域名服务器

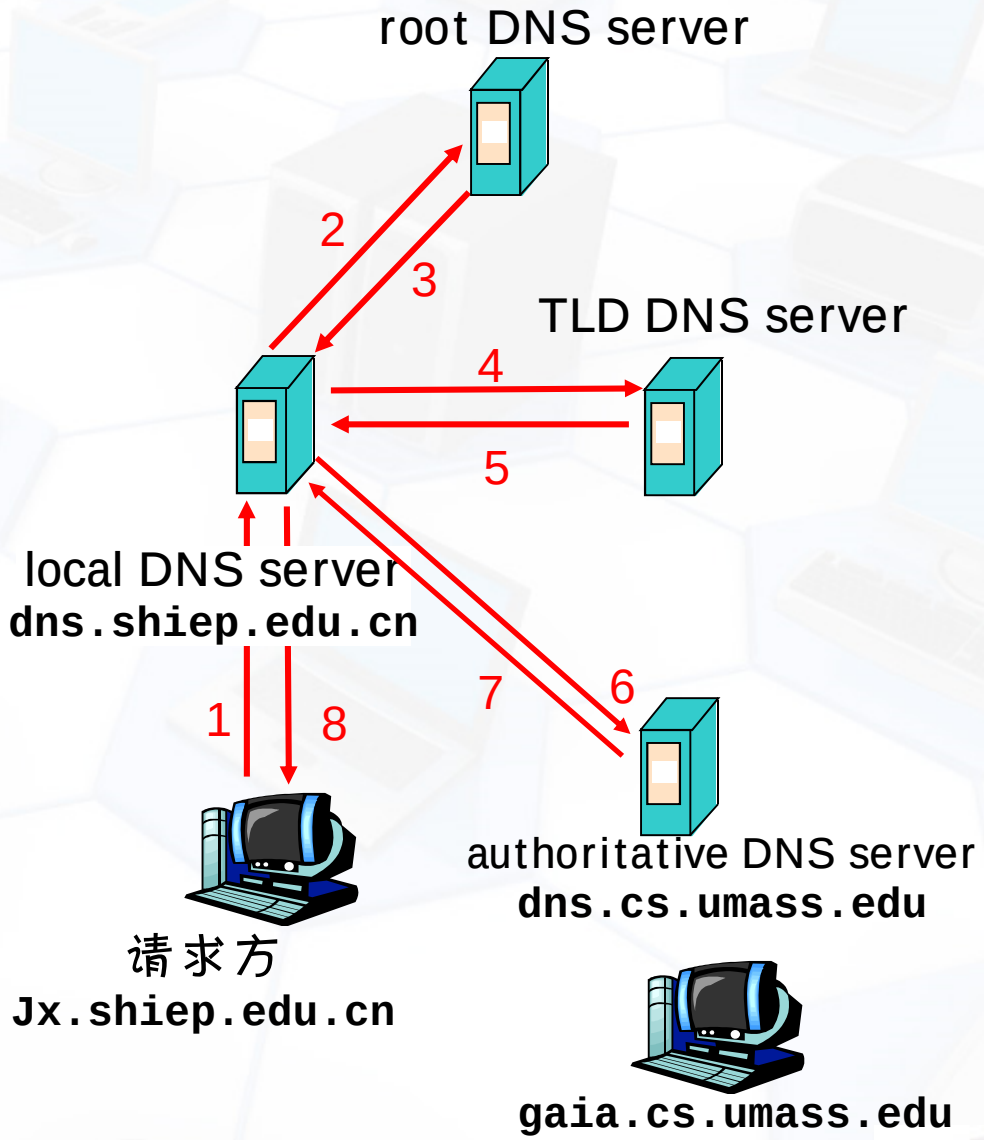


13 root name servers worldwide

- TLD和权威域名服务器
  - 顶级域名服务器(Top-level domain)

    负责顶级域名(如: com, org, net, edu) 和所有国家的顶级域名(如: fr, ca, cn)
  - 权威DNS服务器: 组织机构的DNS服务器, 为组织机构的服务器(如: Web and mail)提供权威的主机名到IP地址的映射
  - 本地域名服务器
    - 严格上看不属于DNS服务的层次范围
    - 每个ISP都有自己的本地域名服务器(也称默认名字服务器)
    - 主机的DNS请求被先送往本地DNS服务器
    - 起到代理的作用，将DNS请求送往DNS层次结构中

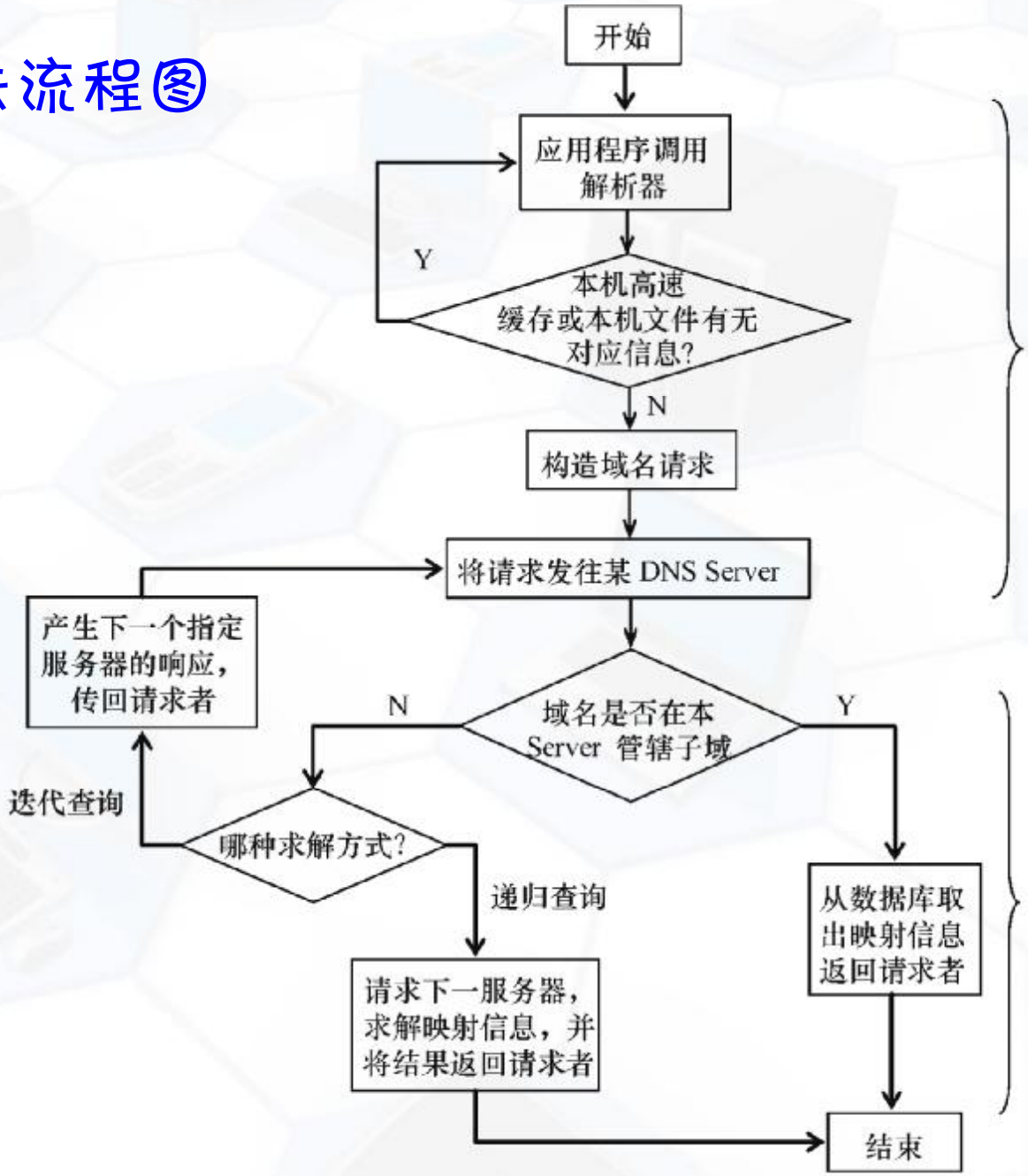
- 电力学院的某主机想要知道位于gaia.cs.umass.edu主机的IP地址



p 域名查询类型

- § 递归查询（recursive query）：每一个被请求的名字服务器如果没有该记录，它就会向其它域名服务器查询，并沿着查询的路径逐个返回记录
- § 迭代查询（iterative query）：本地服务器如果没有该记录，就向高级域名服务器请求，被请求的服务器如果没有该记录就会返回一个可供查询的名字服务器地址

p 解析算法流程图


开始
应用程序调用解析器
本机高速缓存或本机文件有无对应信息?
Y
N
构造域名请求
将请求发往某 DNS Server
产生下一个指定服务器的响应，传回请求者
域名是否在本 Server 管辖子域
N
Y
迭代查询
哪种求解方式?
递归查询
请求下一服务器，求解映射信息，并将结果返回请求者
从数据库取出映射信息返回请求者
结束
请求者操作
服务器操作

域名服务器缓存污染

维基百科，自由的百科全书

（重定向自域名劫持）

汉 漢▼

域名服务器缓存污染（DNS cache poisoning），又名**域名服务器快取侵害**（DNS cache pollution），是指一些刻意制造或无意中制造出来的域名服务器封包，把域名指往不正确的IP地址。一般来说，外间在互联网上一般都有可信赖的域名服务器，但为减免网络上的交通，一般的域名都会把外间的域名服务器资料暂存起来，待下次有其他机器要求解析域名时，可以立即提供服务。一但有关网域的局域域名服务器的缓存受到污染，就会把网域内的电脑导引往错误的服务器或服务器的网址。

域名服务器缓存污染可能是透过域名服务器软件上的设计错误而产生，但亦可能由别有用心者透过研究开放架构的域名服务器系统来利用当中的漏洞。

为防止局域的域名服务器缓存污染除了要定时更新服务器的软件以外，可能还需要人手改动某些设定，以提高服务器对可疑的域名封包作出筛选[1]。

透过改动Windows 2003的某些域名封包设定，可以摒除有可疑的封包。[1]

目录 [隐藏]

缓存污染攻击 [编辑]

一般来说，一部连上了互联网的电脑都会使用互联网服务供应商(ISP)提供的域名服务器。这个服务器一般只会为供应商的客户来服务，通常都会储蓄起部份客户曾经请求过的域名的缓存。缓存污染攻击就是针对这一种服务器，以影响服务器的用户或下游服务。

防火长城的缓存污染攻击 [编辑]

在中国，对于所有经过防火长城的在UDP的53端口上的域名查询进行IDS入侵检测，一经发现与黑名单关键词相匹配的域名查询请求，其会马上伪装成目标域名的解析服务器给查询者返回虚假结果。由于通常的域名查询查询没有任何认证机制，而且域名查询通常基于的UDP协议是无连接不可靠的协议，查询者只能接受最先到达的格式正确结果，并丢弃之后的结果。[2]

对于不了解相关知识的网民来说也就是，由于系统默认使用的ISP提供的域名查询服务器查询国外的权威服务器时即被防火长城被污染，使其缓存受到污染，因而默认情况下查询ISP的服务器就会获得虚假IP地址；而用户直接查询境外域名查询服务器（比如 Google Public DNS）又可能会被防火长城污染，从而在没有任何防范机制的情况下仍然不能获得目标网站正确的IP地址。[3]

- 验证DNS污染的办法

  使用nslookup命令查找一个不存在的域名，比如 www.test.net，由于此域名不存在，应没有任何返回。

  但如果得到了一个错误的IP（不确定），即可证明域名缓存已经被污染了

- 电信DNS劫持

  一般是指电信在对其所属的DNS服务器做了某些明显违背互联网规范的行为,导致出现将客户浏览器导向其自身所辖的网页。比如使用电信宽带接入时，经常在浏览网站时转向114查询的页面

2. Web服务

- 万维网WWW(World Wide Web)是一个大规模的、联机式的信息集合，简称为Web
- 发展过程：
  - WWW研制者：CERN（the European parical physics Laboratory）欧洲瑞士日内瓦的离子物理实验室。
  - 1984年，Time Berners-Lee提出了超文本（Hypertext)语言（HTML）；1989年研制成功WWW（world wide web）,1991年公布
  - 1993年，美国伊利诺依大学国家超级计算机中心NCSA青年科学家 Marc Andreeason开发成功了浏览工具Mosaic
  - 1994年进而发展成为Netscape
  - 1996年，又出现了IE（Internet Explorer）

1) Web的基本概念

WWW 是一个庞大的、世界范围的、分布式超媒体系统，由遍布在Internet中的WWW 服务器组成

超媒体是指除了文本信息，还包含图形、声音、动画、视频等

2) 统一资源定位符URL(Uniform Resource Locator)

URL描述了web页面的“名称”、“在什么地方”以及“怎样访问”三方面的内容，格式如下：

scheme：//host：port/path

①　②　③

①— 服务方式或协议，除了WWW 用的HTTP 协议之外，还可以是FTP、TELNET 等

②— 存有该资源的主机地址，有时也包括端口地址

③— 路径，指出服务器上某资源的具体地址

p URL举例

www.someschool.edu/someDept/pic.gif

host name　　path name

http://www.shiep.edu.cn:80/network/index.html

协议，除了http之外，还可以是https/ftp 等

存放该资源的主机名，也可包括端口

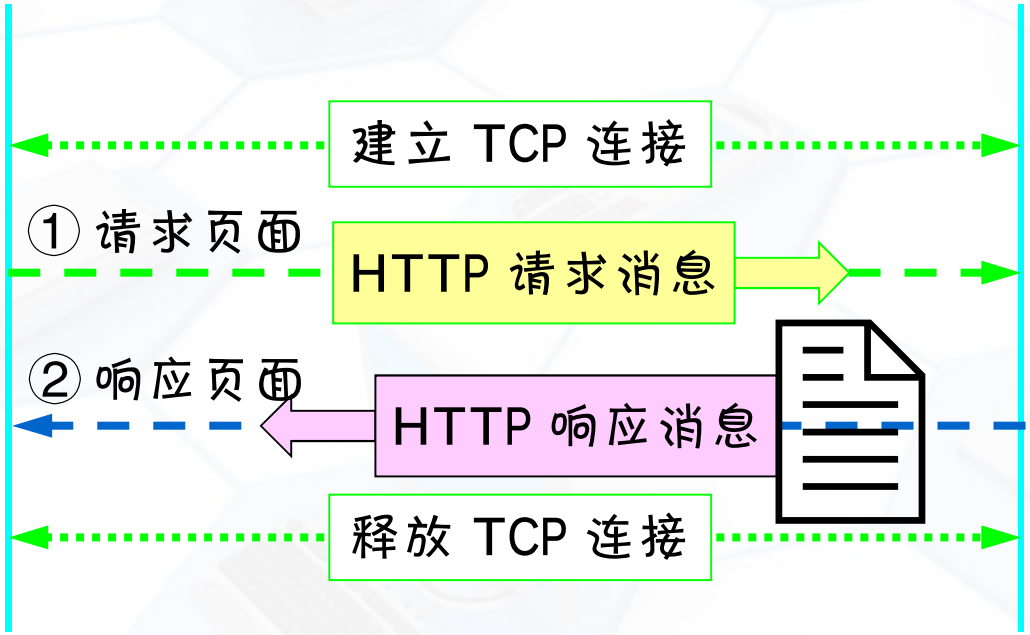
指出该主机上最终访问的文件名称

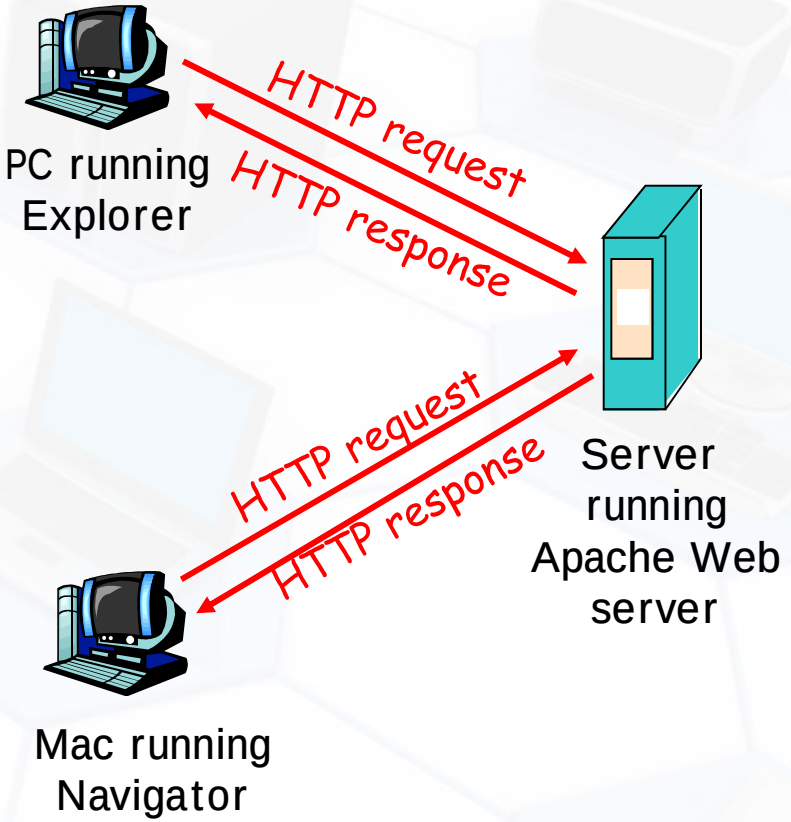
指出该主机上某资源的具体地址/路径

3）超文本传输协议HTTP（Hypertext Transfer Protocol）

- HTTP是web服务的核心，诞生于1990年，是用来在浏览器和WWW 服务器之间传送超文本的协议
- HTTP协议基于客户机/服务器工作方式
- 由两个相当明显的项组成
  - 从浏览器到服务器的请求集
  - 从服务器到浏览器的应答集
- HTTP 会话过程包括以下四个步骤：连接、请求、应答、关闭

p HTTP消息传输过程





4) 超文本标记语言（HTML）

- HTML 是一种标记语言，描述如何格式化文档。它使用一些约定的标记对WWW上各种信息、格式以及超级链接进行描述
- 任何一个Web浏览器都可以读取任何Web页面，并且对页面重新格式化
- HTML文档是以纯ASCII文件存在的，由“控制语句”与“显示内容”两部分组成
- 静态HTML与动态HTML

  动态HTML又分为服务器端动态web页面和客户端动态web页面

5) WWW服务的实现过程

客户端的工作过程举例：假如有用户要访问http://www.shiep.edu.cn/main.html，则浏览器的工作过程如下：

(1) 浏览器分析URL

(2) 浏览器向DNS询问web服务器www.shiep.edu.cn的IP 地址

(3) DNS的应答是202.17.57.180

(4) 浏览器和IP 地址为202.17.57.180的80 端口建立TCP 连接

(5) 浏览器执行HTTP 协议，发送GET main.html 命令，请求读取该文件

(6) www.shiep.edu.cn 服务器返回main.html文件到客户端

(7) 释放TCP 连接

(8) 浏览器显示main.html页面

- 服务器端的工作过程

(1) 接受来自客户(浏览器)的TCP连接
(2) 获取所需文件的名字
(3) 获取文件(从磁盘上)
(4) 将文件返回给客户
(5) 释放该TCP连接

6) HTTP消息类型

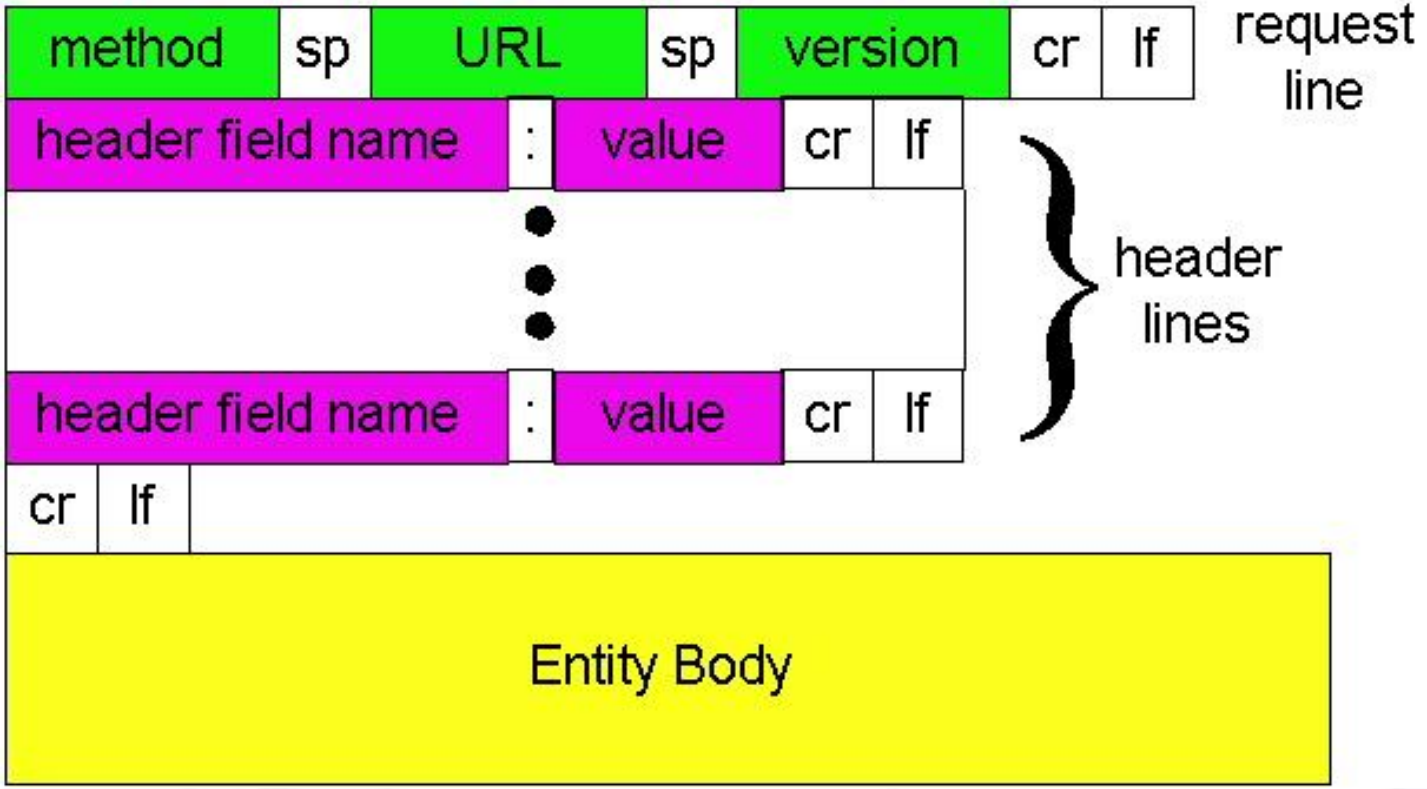
- HTTP 有两类消息：
  - 请求消息——客户端发向web服务器
  - 响应消息——从web服务器发往客户端

- HTTP请求消息

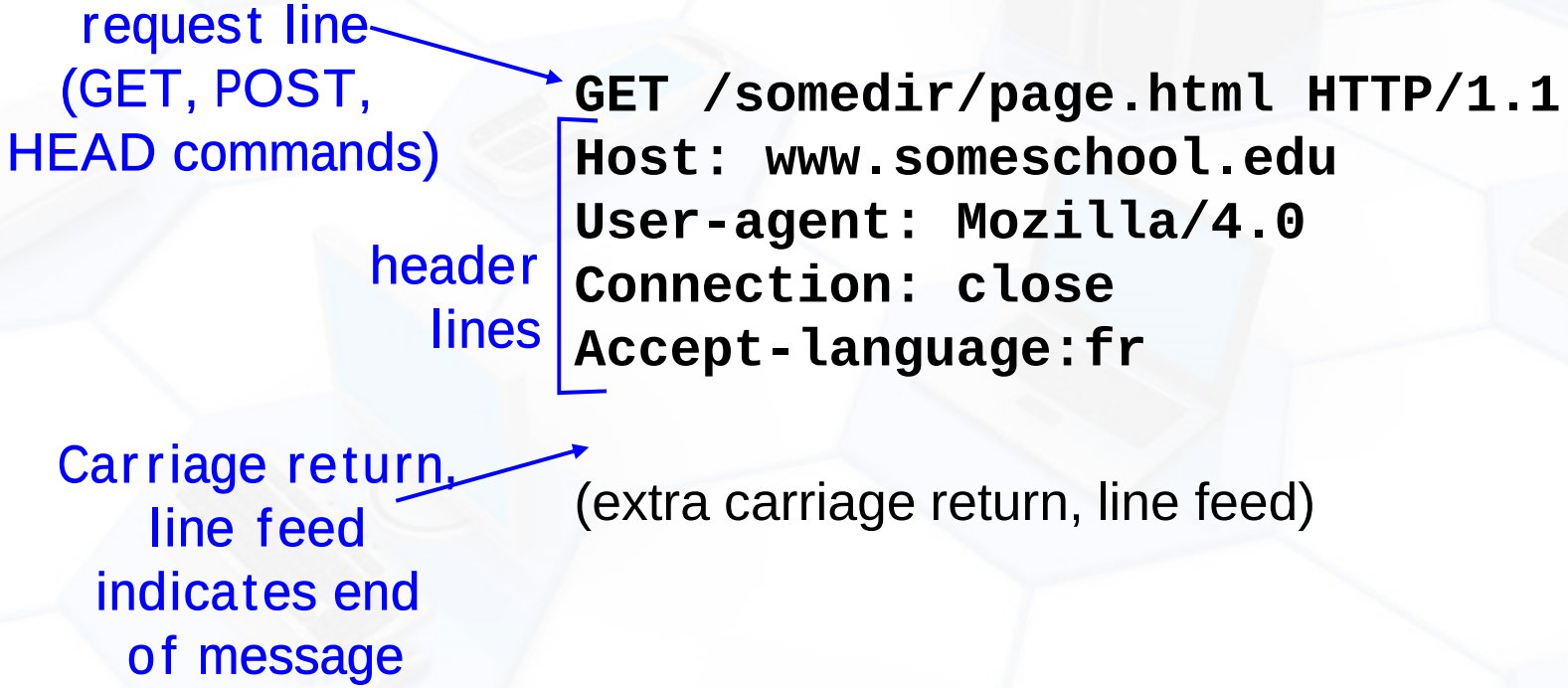
方法(操作)	意义
OPTION	查询特定选项
GET	请求读取一个web页面
HEAD	请求读取由一个web页面的头部
POST	给服务器添加信息（例如，注释）
PUT	请求存储一个web页面
DELETE	删除web页面
TRACE	送回收到的请求

- 响应消息的状态码
  - 1xx 表示通知信息，如服务器同意处理客户请求(100)
  - 2xx 表示成功，如请求成功(200)或没有内容存在(204)
  - 3xx 表示重定向，如页面移动(301)或者缓存的页面仍然有效(304)
  - 4xx 表示客户错误，例如禁止页面(403)或者页面没有找到(404)
  - 5xx 表示服务器错误，如服务器内部错误(500)或者以后再试(503)

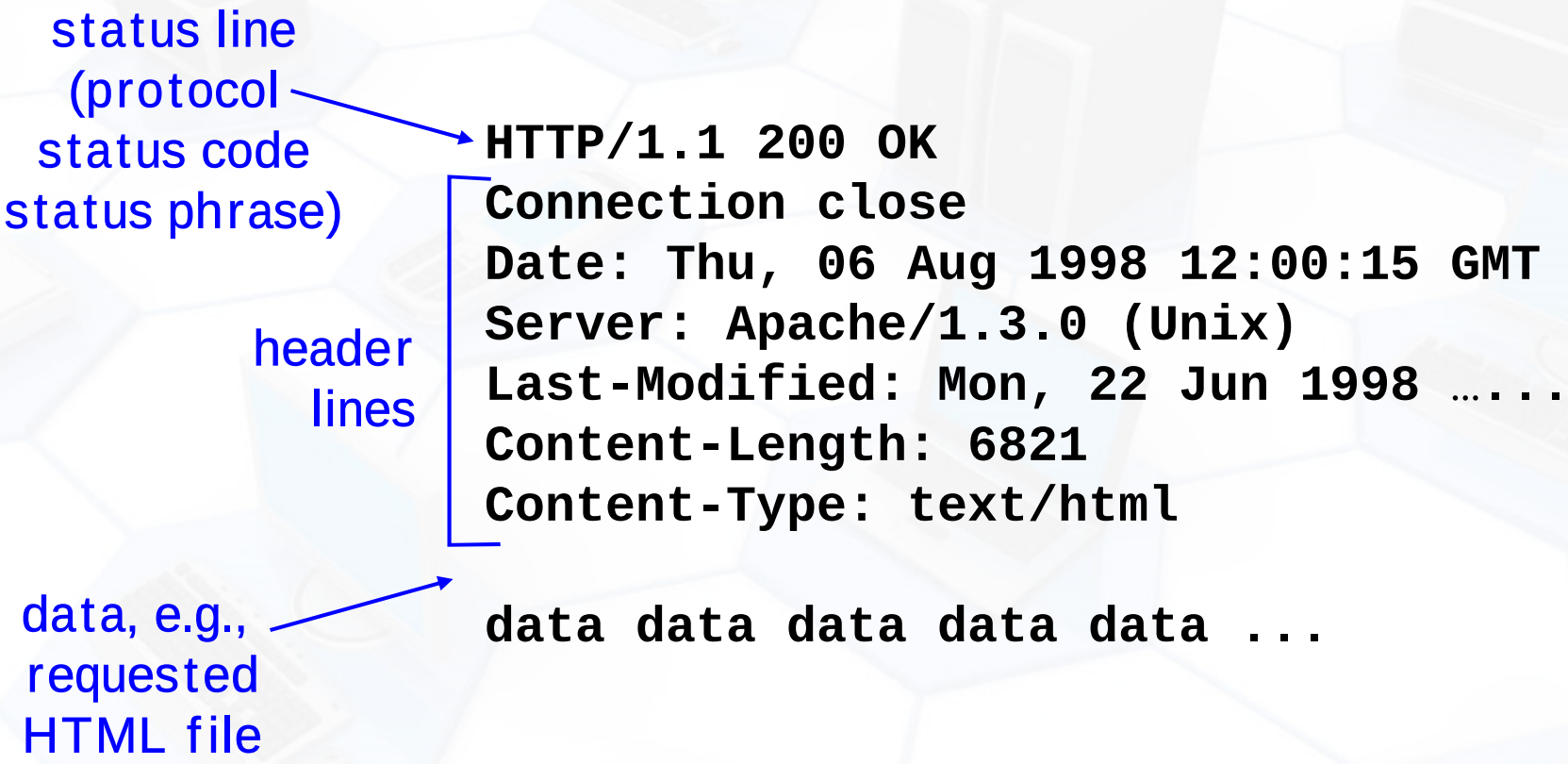
7) HTTP 请求信息的格式



p HTTP 请求消息举例



p HTTP 响应消息举例



p 在客户端尝试HTTP

①使用 Telnet连接到 Web server

```
telnet cis.poly.edu 80
```

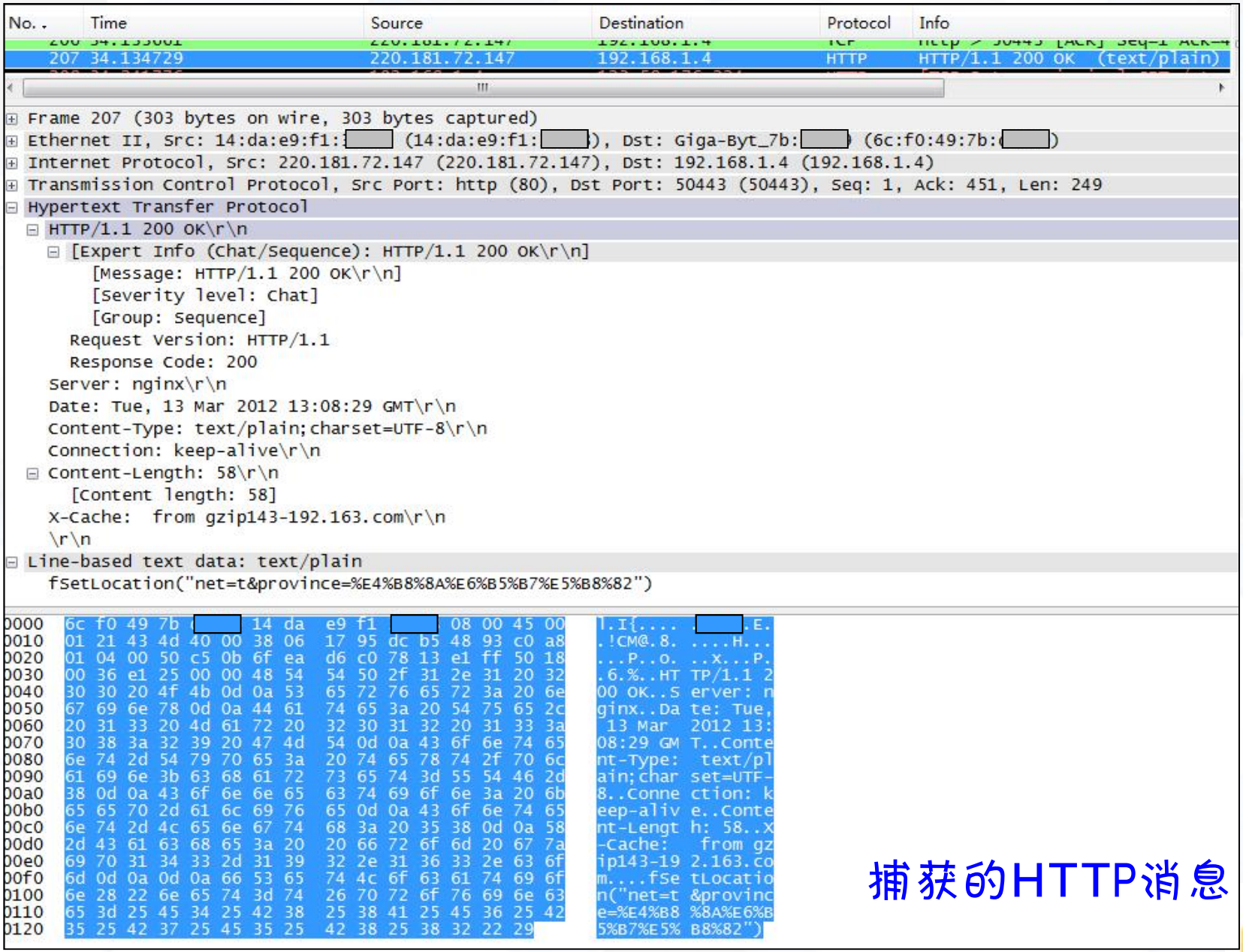
通过80端口打开cis.poly.edu 的TCP连接

② 使用 GET HTTP 请求

```
GET /~ross/ HTTP/1.1
Host: cis.poly.edu
```

键入HTTP请求，
向HTTP服务端发出GET 请求

③ 观察HTTP server回送的响应消息


No. Time Source Destination Protocol Info
207 34.134729 220.181.72.147 192.168.1.4 HTTP HTTP/1.1 200 OK (text/plain)
Frame 207 (303 bytes on wire, 303 bytes captured)
Ethernet II, Src: 14:da:e9:f1:[redacted] (14:da:e9:f1:[redacted]), Dst: Giga-Byt_7b:[redacted] (6c:f0:49:7b:[redacted])
Internet Protocol, Src: 220.181.72.147 (220.181.72.147), Dst: 192.168.1.4 (192.168.1.4)
Transmission Control Protocol, Src Port: http (80), Dst Port: 50443 (50443), Seq: 1, Ack: 451, Len: 249
Hypertext Transfer Protocol
HTTP/1.1 200 OK\r\n
[Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
[Message: HTTP/1.1 200 OK\r\n]
[Severity level: Chat]
[Group: Sequence]
Request Version: HTTP/1.1
Response Code: 200
Server: nginx\r\n
Date: Tue, 13 Mar 2012 13:08:29 GMT\r\n
Content-Type: text/plain;charset=UTF-8\r\n
Connection: keep-alive\r\n
Content-Length: 58\r\n
[Content length: 58]
X-Cache: from gzip143-192.163.com\r\n
\r\n
Line-based text data: text/plain
fSetLocation("net=t&province=%E4%B8%8A%E6%B5%B7%E5%B8%82")
0000 6c f0 49 7b [redacted] 14 da e9 f1 [redacted] 08 00 45 00 l.I{.... [redacted].E.
0010 01 21 43 4d 40 00 38 06 17 95 dc b5 48 93 c0 a8 .!CM@.8. ....H...
0020 01 04 00 50 c5 0b 6f ea d6 c0 78 13 e1 ff 50 18 ...P..o. ..x...P.
0030 00 36 e1 25 00 00 48 54 54 50 2f 31 2e 31 20 32 .6.%..HT TP/1.1 2
0040 30 30 20 4f 4b 0d 0a 53 65 72 76 65 72 3a 20 6e 00 OK..S erver: n
0050 67 69 6e 78 0d 0a 44 61 74 65 3a 20 54 75 65 2c ginx..Da te: Tue,
0060 20 31 33 20 4d 61 72 20 32 30 31 32 20 31 33 3a 13 Mar 2012 13:
0070 30 38 3a 32 39 20 47 4d 54 0d 0a 43 6f 6e 74 65 08:29 GM T..Conte
0080 6e 74 2d 54 79 70 65 3a 20 74 65 78 74 2f 70 6c nt-Type: text/pl
0090 61 69 6e 3b 63 68 61 72 73 65 74 3d 55 54 46 2d ain;char set=UTF-
00a0 38 0d 0a 43 6f 6e 6e 65 63 74 69 6f 6e 3a 20 6b 8..Conne ction: k
00b0 65 65 70 2d 61 6c 69 76 65 0d 0a 43 6f 6e 74 65 eep-aliv e..Conte
00c0 6e 74 2d 4c 65 6e 67 74 68 3a 20 35 38 0d 0a 58 nt-Lengt h: 58..X
00d0 2d 43 61 63 68 65 3a 20 20 66 72 6f 6d 20 67 7a -Cache: from gz
00e0 69 70 31 34 33 2d 31 39 32 2e 31 36 33 2e 63 6f ip143-19 2.163.co
00f0 6d 0d 0a 0d 0a 66 53 65 74 4c 6f 63 61 74 69 6f m....fSe tLocatio
0100 6e 28 22 6e 65 74 3d 74 26 70 72 6f 76 69 6e 63 n("net=t &provinc
0110 65 3d 25 45 34 25 42 38 25 38 41 25 45 36 25 42 e=%E4%B8 %8A%E6%B
0120 35 25 42 37 25 45 35 25 42 38 25 38 32 22 29 5%B7%E5% B8%82")

捕获的HTTP消息

p 密码泄露

No.	Time	Source	Destination	Protocol	Length	Info
29	9.828892	192.168.43.172	220.181.15.149	HTTP	654	GET /js4/main.jsp?sid=JBbIFouyvhuIBfrCSuyyRdvGPOdvaipf HTTP/1.1
30	9.984308	220.181.15.149	192.168.43.172	TCP	54	http > messageservice [ACK] Seq=1 Ack=1461 Win=8760 Len=0

```
⊞ Internet Protocol Version 4, Src: 192.168.43.172 (192.168.43.172), Dst: 220.181.15.149 (220.181.15.149)
⊞ Transmission Control Protocol, Src Port: messageservice (2311), Dst Port: http (80), Seq: 1461, Ack: 1, Len: 600
⊞ [2 Reassembled TCP Segments (2060 bytes): #28(1460), #29(600)]
⊟ Hypertext Transfer Protocol
  ⊟ GET /js4/main.jsp?sid=JBbIFouyvhuIBfrCSuyyRdvGPOdvaipf HTTP/1.1\r\n
    ⊟ [Expert Info (Chat/Sequence): GET /js4/main.jsp?sid=JBbIFouyvhuIBfrCSuyyRdvGPOdvaipf HTTP/1.1\r\n]
        [Message: GET /js4/main.jsp?sid=JBbIFouyvhuIBfrCSuyyRdvGPOdvaipf HTTP/1.1\r\n]
        [Severity level: Chat]
        [Group: Sequence]
      Request Method: GET
      Request URI: /js4/main.jsp?sid=JBbIFouyvhuIBfrCSuyyRdvGPOdvaipf
      Request Version: HTTP/1.1
    [truncated] Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/vnd.ms-excel, application/vnd.ms-pow
    Accept-Language: zh-cn\r\n
    Accept-Encoding: gzip, deflate\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; GTB5; .NET CLR 1.1.4322; .NET CLR 2.0.50727; .NET CLR 3.0.
    Host: twebmail.mail.126.com\r\n
    Connection: Keep-Alive\r\n
    [truncated] Cookie: SID=d3f9d444-ce31-4e0f-a62e-ce426e781d50; S_INFO=1333544162|0|#2&30#|net1305161@126.com; NTES_SESS=1N
    \r\n
    [Full request URI: http://twebmail.mail.126.com/js4/main.jsp?sid=JBbIFouyvhuIBfrCSuyyRdvGPOdvaipf]
```

p 钓鱼网站

钓鱼式攻击

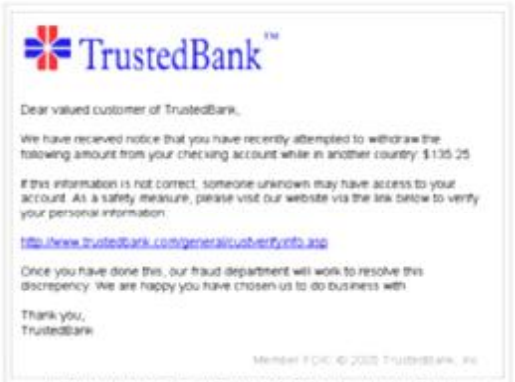
维基百科，自由的百科全书

（重定向自钓鱼网站）

钓鱼式攻击（Phishing，与钓鱼的英语fishing发音一样，又名“网钓法”或“网络网钓”，以下简称网钓）是一种企图从电子通讯中，透过伪装成信誉卓著的法人媒体以获得如用户名、密码和信用卡明细等个人敏感信息的犯罪诈骗过程。这些通信都声称（自己）来自于风行的社交网站（YouTube、Facebook、MySpace）、拍卖网站（eBay）、网络银行、电子支付网站（PayPal）、或网络管理者（雅虎、互联网服务供应商、公司机关），以此来诱骗受害人的轻信。网钓通常是透过e-mail或者即时通讯进行[1]。它常常导引用户到URL与接口外观与真正网站几无二致的假冒网站输入个人资料。就算使用强式加密的SSL服务器认证，要侦测网站是否仿冒实际上仍很困难。网钓是一种利用社会工程技术来愚弄用户的实例[2]。它凭恃的是现行网络安全技术的低亲和度。[3]种种对抗日渐增多网钓案例的尝试涵盖立法层面、用户培训层面、宣传层面、与技术保全措施层面。

网钓技术最早于1987年问世，而首度使用“网钓”这个术语是在1996年。该辞是英文单词**钓鱼**（fishing）的变种之一[4]，大概是受到“飞客”（phreaking）一词影响[5][6]，意味着放线钓鱼以“钓”取受害人财务资料和密码。

目录 [隐藏]



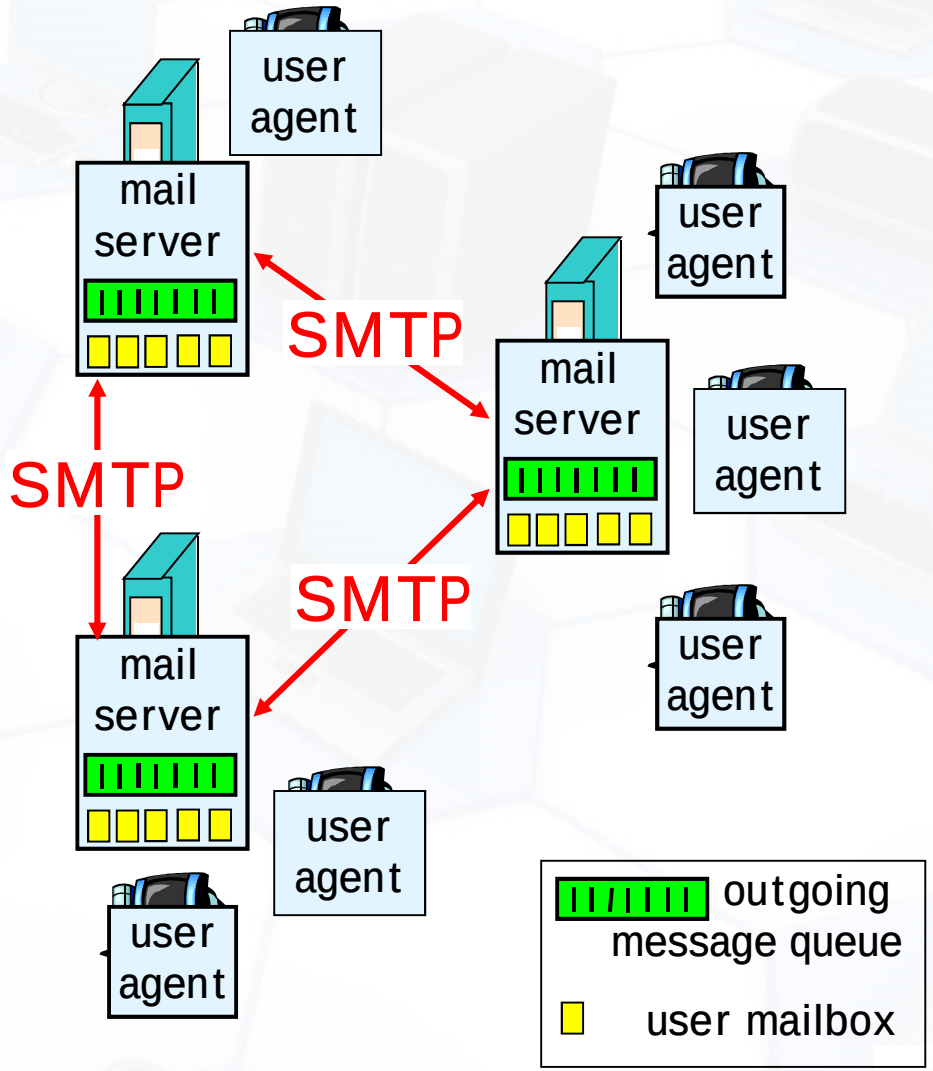
一个假装成来自某（虚构）银行官方电子邮件的网钓电子邮件范例。发件人透过要求收件人在网钓站点“确认”其身分试图骗取其保全信息。注意received与discrepancy两字的拼写错误。这种错误在大部分网络网钓电子邮件十分常见。

3. E-mail服务

1) 电子邮件系统的组成

三个主要组成构件

- 用户代理
- 邮件服务器
- 协议



2) 用户代理(user agent)

- 用户代理是运行在客户机上的一个本地程序，负责让用户撰写、阅读和处理电子邮件
- 常用的有：sendmail、Netscape 、 IE、foxmail等

3) 邮件服务器(mail server)

- 邮箱 存放给用户的信息(邮件)
- 消息队列 准备发送的邮件
- SMTP协议 邮件服务器间发送邮件信息使用的协议

4) 协议

§ 发送邮件：如SMTP(Simple Mail Transfer Protocol)协议、MIME(Multipurpose Internet Mail Extensions)协议。前者只能传输文本信息，后者则可以传输包括文本、声音、图像等在内的多媒体信息

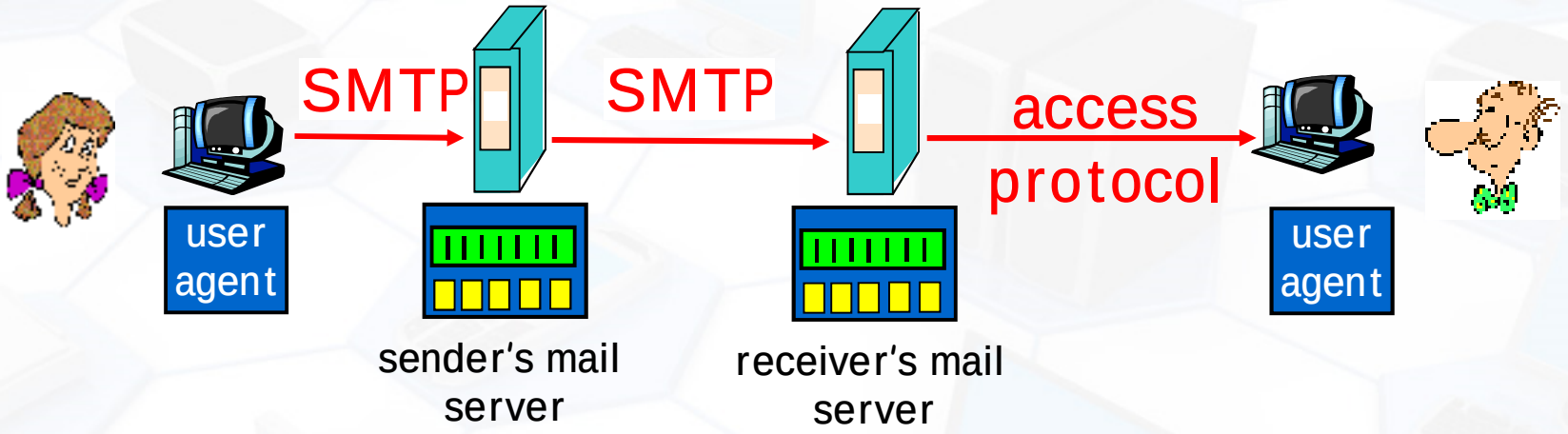
§ 接收邮件，如：POP(Post Office Protocol)

p 邮件首部有严格的格式要求，尤其是E-mail地址，E-mail 地址的标准格式为：

<收信人信箱名>@主机域名

- 相关协议
  - 消息交换的协议
    - 发送email：简单邮件传输协议 SMTP（Simple Mail Transfer Protocol）
    - 接收email：邮局协议第3版POP3（Post Office Protocol version 3），Internet消息访问协议IMAP（Internet Message Access Protocol）
  - 消息格式的协议
    - RFC 822：基本的ASCII的文本邮件
    - 多用途Internet邮件扩展 MIME（Multipurpose Internet Mail Extensions）：RFC 822的多媒体扩展

p Mail 访问协议



- p SMTP: 传递/存储邮件到接收服务器
- p Mail 访问协议: 从服务器获取邮件
  - ü POP: Post Office Protocol [RFC 1939]
    - ¡ 需认证然后下载邮件
  - ü IMAP: Internet Mail Access Protocol [RFC 1730]
    - ¡ 比POP有更多特色 (也更复杂)
    - ¡ 适合在服务器上操控存放的消息
  - ü HTTP: Hotmail , Yahoo! Mail, etc.

p SMTP协议

SMTP基本命令集

命令	描述
HELO	向服务器标识用户身份，发送者能欺骗，但一般情况下服务器都能检测到
MAIL	初始化邮件传输 mail form :
RCPT	标识单个的邮件接收人；常在MAIL命令后面，可有多个rcpt to:
DATA	在单个或多个RCPT命令后，表示所有的邮件接收人已标识，并初始化数据传输结束
VRFY	用于验证指定的用户/邮箱是否存在；由于安全方面的原因，服务器常禁止此命令
EXPN	验证给定的邮箱列表是否存在，扩充邮箱列表，也常被禁用
HELP	查询服务器支持什么命令
NOOP	无操作，服务器应响应OK
QUIT	结束会话
RSET	重置会话，当前传输被取消

p SMTP通信过程实例

```
S: 220 hamburger.edu
C: HELO crepes.fr
S: 250  Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr... Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Do you like ketchup?
C: How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection
```

- 手工尝试SMTP连接
  - telnet *servername* 25
  - 服务端会给出 220 开头的回应
  - 接下来可以尝试输入 HELO, MAIL FROM, RCPT TO, DATA, QUIT 等命令
  - 以上方法可以不使用客户端进行邮件发送

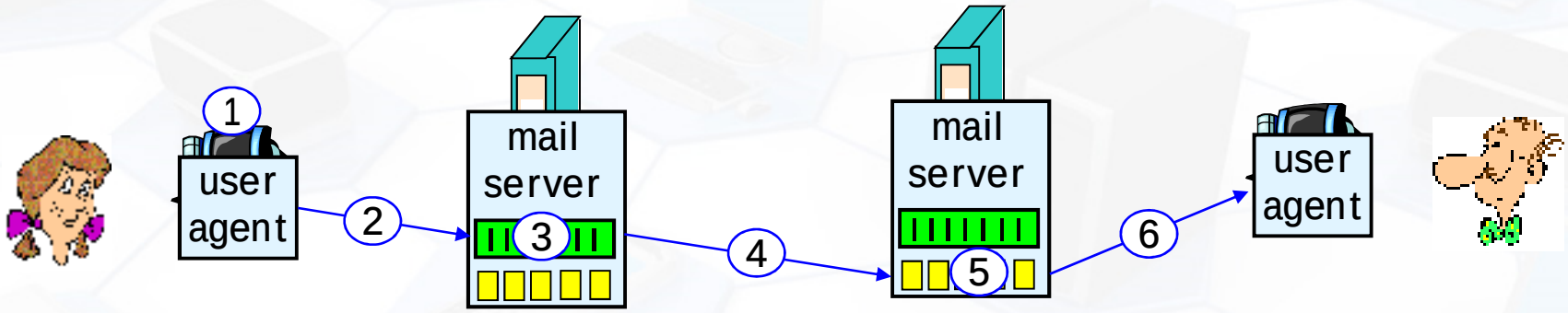
p POP3协议

- § 邮局协议 POP 是一个非常简单、但功能有限的邮件读取协议，现在使用的是它的第三个版本 POP3
- § POP 也使用客户/服务器的工作方式
- § 在接收邮件的用户主机中必须运行 POP 客户端进程，而在用户所连接的邮件服务器中则运行 POP 服务器进程
- § POP3协议一般假设用户从服务器上把邮件存储到本地主机上，同时删除保存在邮件服务器上的邮件

p IMAP协议

- § IMAP 也是按客户/服务器模式工作，现在较新的版本是 IMAP4
- § 用户在自己的主机上就可以操纵邮件服务器的邮箱，就像在本地操纵一样
- § IMAP 是一个联机协议。当用户主机上的 IMAP 客户程序打开 IMAP 服务器的邮箱时，用户就可看到邮件的头部。若用户需要打开某个邮件，则该邮件才传到用户的计算机上

p 举例: Alice 发邮件给Bob



① Alice使用她的UA写了一封邮件给 bob@someschool.edu

② Alice端的UA将消息发往她的邮件服务器; 之后该消息进入消息队列

③ 客户端的 SMTP与Bob的邮件服务器建立TCP连接

④ SMTP 客户端将Alice的消息通过TCP连接发出

⑤ Bob端的邮件服务器将此消息放入Bob的邮箱

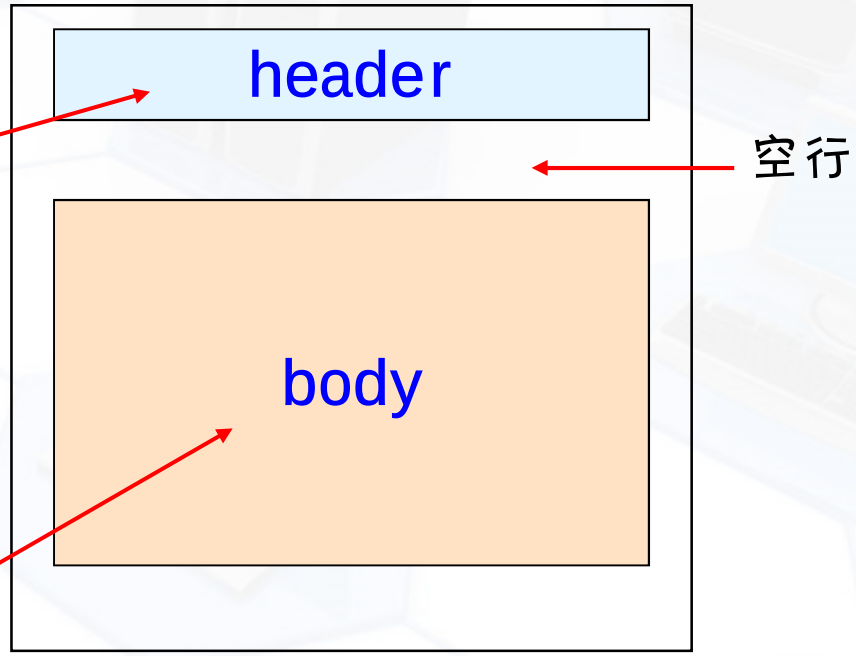
⑥ Bob 打开他的UA来读取邮件

5) 邮件消息格式

标注的文本消息格式:

- § header lines
  - ü To:
  - ü From:
  - ü Subject:

  不同于SMTP命令
- § body
  - ü 消息正文，只能是ASCII字符



第一封电子邮件内容 [编辑]

历史上第一封E-mail确切的发出时间、地点、人物有所争议。

- 《互联网周刊》报道为："1969年10月，世界上的第一封电子邮件是由计算机科学家Leonard K.教授发给他的同事的一条简短消息。"
- 1971年由为阿帕网工作的麻省理工学院博士Ray Tomlinson测试软件（SNDMSG）时发出的，并且首次使用"@"作为地址间隔标示。

中国第一封电子邮件内容 [编辑]

1987年9月14日[1]中国第一封电子邮件是由"**德国互联网之父**"**维纳·措恩**与**王运丰**在北京的计算机应用技术研究所发往德国卡尔斯鲁厄大学的，其内容为英文，大意如下。

- 原文：
  - **Across the Great Wall we can reach every corner in the world.**
- 中文大意
  - **跨越长城，走向世界。**

这是中国通过北京与德国卡尔斯鲁厄大学之间的网络连接，向全球科学网发出了第一封电子邮件。

p 带有多媒体扩展的消息格式

- § MIME: multimedia mail extension [RFC 2045, 2056]
- § 消息头部有附加说明 MIME 内容的类型

MIME 版本

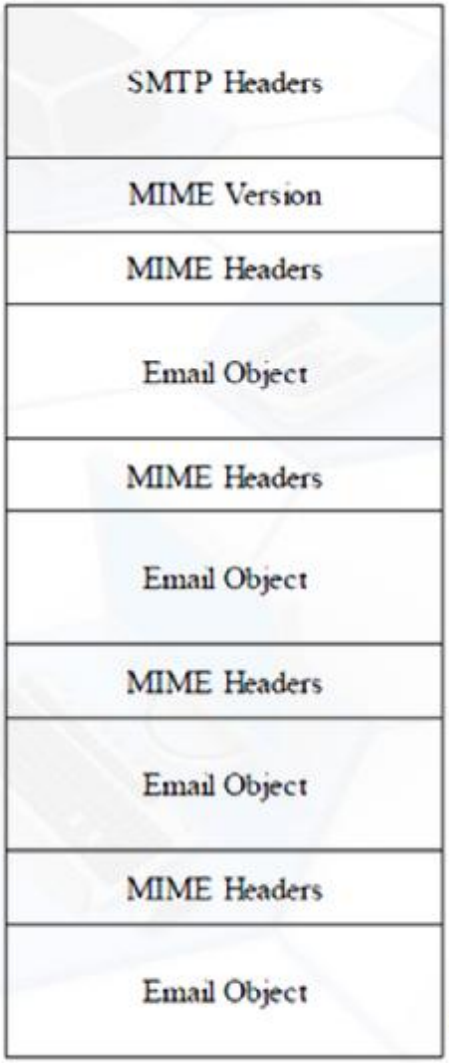
数据编码方法

多媒体数据类型，
子类，
参数说明

编码后的数据

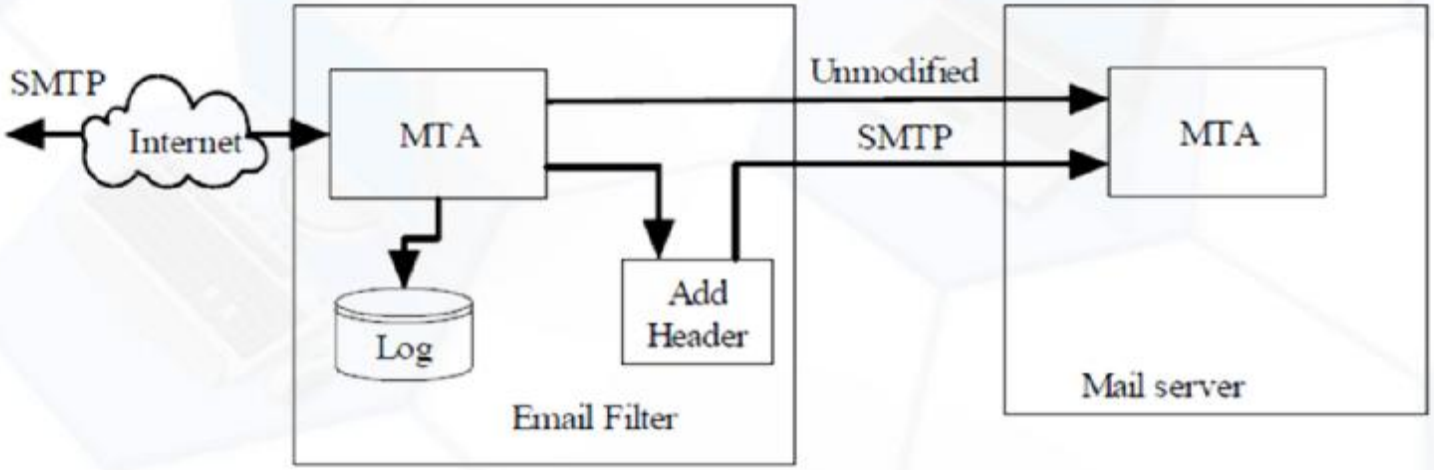
```
From: alice@crepes.fr
To: bob@hamburger.edu
Subject: Picture of yummy crepe.
MIME-Version: 1.0
Content-Transfer-Encoding: base64
Content-Type: image/jpeg

base64 encoded data .....
.........................
......base64 encoded data
```

MIME Structure



6) 邮件过滤

- 基于邮件地址
- 基于域名地址
- 基于恶意信息（敏感信息）



主题：FW_研ü发人ù员シ的°考ü核激ù励 7gbq3

```
Received: from www.ad456.xicp.net (unknown [121.228.216.42])
        by mx10 (Coremail) with SMTP id KMmowEAJIkEpv2pPhTorAA--.1823S2;
        Thu, 22 Mar 2012 13:57:03 +0800 (CST)
Received: from nklek.com by www.ad456.xicp.net (MDaemon PRO v11.0.3 trial)
        with ESMTP id md50005682930.msg
        for <733488@126.com>; Thu, 22 Mar 2012 11:44:01 +0800
X-Spam-Processed: www.ad456.xicp.net, Thu, 22 Mar 2012 11:44:01 +0800
        (not processed: message was sent from localhost)
X-MDMailLookup-Result: pass smtp.mail=qpjye@ad456.xicp.net (ip=127.0.0.1) (www.ad456.xicp.net)
X-Return-Path: qpjye@ad456.xicp.net
X-Envelope-From: qpjye@ad456.xicp.net
X-MDaemon-Deliver-To: 733488@126.com
Message-ID: <20120322114358337624@ad456.xicp.net>
From: " " <qpjye@ad456.xicp.net>
To: <733488@126.com>
Subject: =?utf-8?B?Rlc656CUw7zlj5HkurrHnOWRmOOCt+eahMKw6ICDw7zmoLjmv4DHnOWKsSAg?==?utf-8?B?IDdnYnEz?=
Date: Thu, 22 Mar 2012 11:43:51 +0800
MIME-Version: 1.0
Content-Type: multipart/mixed;
        boundary="----=_NextPart_000_0F11_01E3461A.13358E90"
X-mailer: Qwqpkxg 1
Disposition-Notification-To: sgcs001@sina.com
X-Coremail-Antispam: 1Uf129KBjDUn29KB7ZKAUJUUjxr529EdanIXcx71UUUUU7v73
        VFW2AGmfu7bjvjm3AaLaJ3D34fWan5Wa4Y9rZ7Zwn3GrZ8Z3WfXay7Zr47WrZ8W397YxBI
        daVFxhVjvjDU0xZFpf9x07bzmiiUUUUU=
```

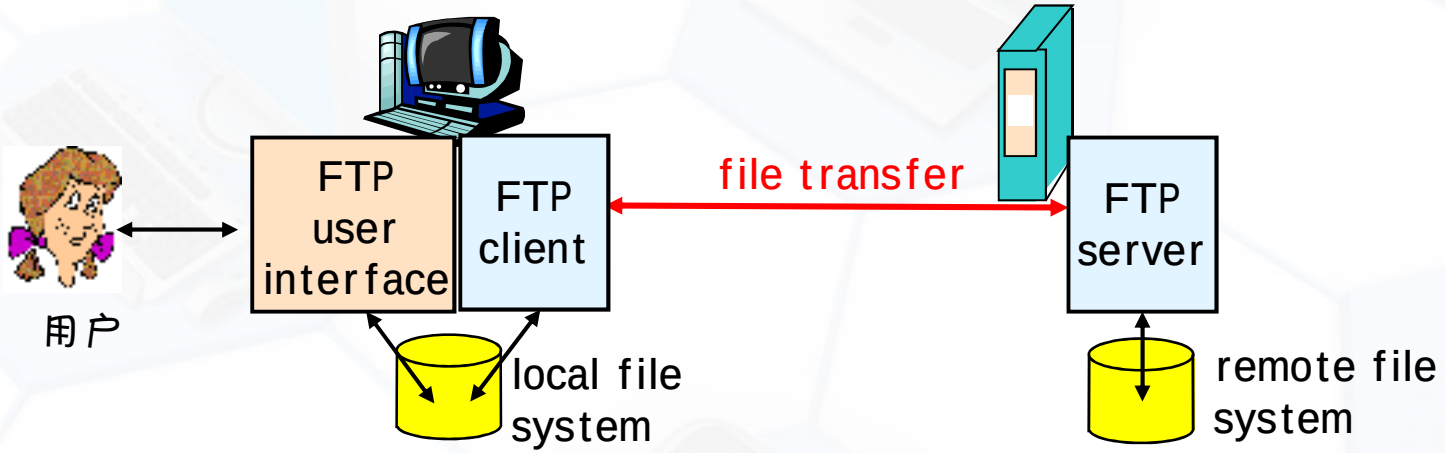
垃圾邮件

4. FTP服务

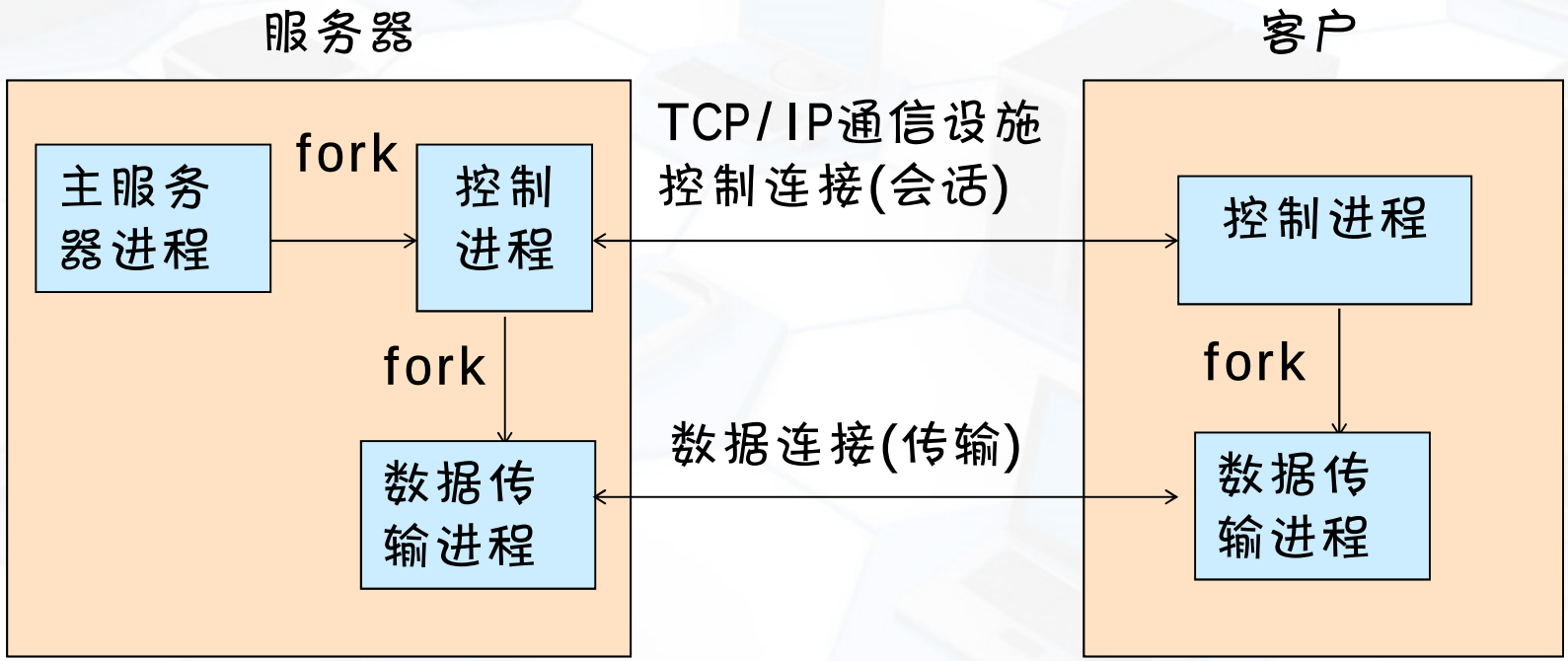
§ 文件传输协议FTP（File Transfer Protocol）是用于在TCP/IP 网络上两台计算机间进行文件传输使用得最广泛的协议

§ FTP提供文件传送的基本服务，减少或消除在异构系统下处理文件的兼容问题

§ Client/Server模式：客户端发起传输



p FTP的客户/服务器模型



§服务器进程由一个主服务器进程与若干个从属进程组成

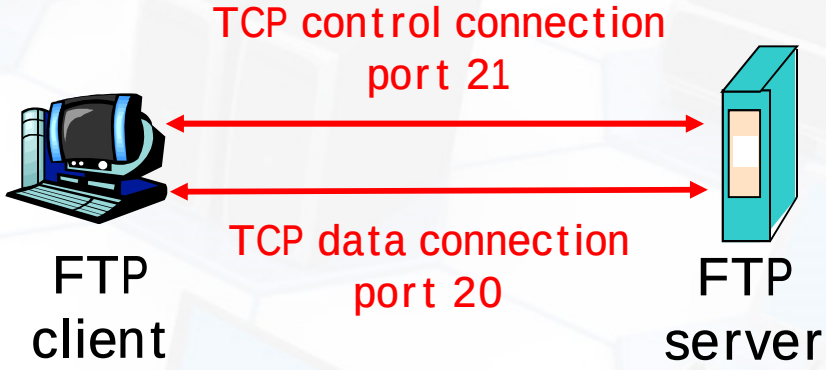
§从属进程有两个：一个是“控制进程”，另一个是“数据传输进程”

§控制进程传送命令，告诉服务器将传送哪个文件

§数据传输进程传送所有数据

p FTP: 分离的控制与数据连接

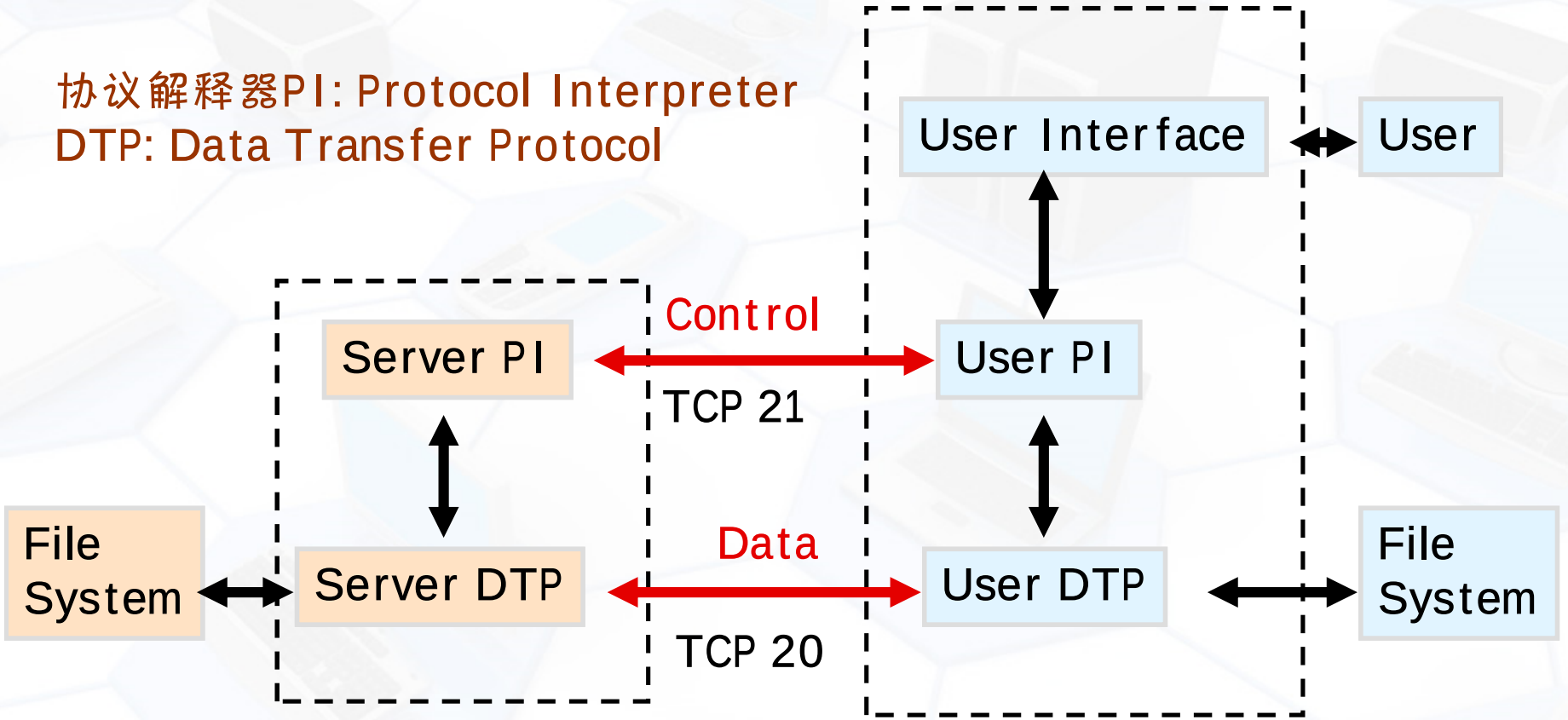
- § FTP客户端联系FTP服务端的21号端口，使用TCP作为传输协议
- § 客户端通过控制连接获取授权
- § 客户端通过控制连接发送命令来浏览远端文件系统的目录
- § 当服务端收到文件传输命令，服务端与客户端间打开第二个TCP连接 (用于文件传输)
- § 当文件传输完成，服务端关闭数据传输连接



- § 服务端会为新的文件传输打开新的 TCP连接
- § 控制连接可以看作是“带外连接(out of band)”
- § FTP服务器保持 用户的“状态”: 当前访问的目录，先前的授权

p FTP的工作逻辑模型

协议解释器PI: Protocol Interpreter
DTP: Data Transfer Protocol



控制连接一直保持到客户服务器连接的全过程，数据连接根据需要打开

p FTP的工作过程如下

- § 在FTP的服务器上，只要启动了FTP服务，主服务器进程就打开熟知端口(端口号为21)，使客户进程能够连接上，等待客户机的连接请求
- § 当主服务器进程监听到客户端发出的建立连接的请求时，就启动从属进程处理客户进程发来的请求，自己则回到等待状态，继续接受其它客户进程的请求
- § 从属进程有两类，其中控制进程进行用户名密码及权限的验证，验证通过后，相当于在客户机与FTP服务器之间打开了一个命令传输的控制连接，该连接在整个FTP会话期间一直存在，所有与文件管理有关的命令将通过该连接被发送至服务器端执行。并且创建“数据传输进程”与“数据连接”，用于连接客户端和服务器端的数据传输进程，完成文件的传送

§ 数据连接有两种传输模式：

- ü 主动传输模式
- ü 被动传输模式

§ 使用FTP的两种方式：

- ü 用户可以使用FTP命令来进行文件传输，这种称为交互模式
- ü 使用工具软件如NetAnts、CuteFTP等，另外Internet Explorer和Netscape Navigator也提供FTP客户软件的功能

p ftp 命令及响应结果

模拟命令:

- 通过控制通道发送ASCII字符命令
- ftp 202.38.75.79
- USER *username*
  - ü *anonymous*
- PASS *password*
- LS *return list of file in current directory*
- get filename
- put filename
- help

模拟的返回代码

- 由状态码和短语组成
- 331 Username OK, password required
- 125 data connection already open; transfer starting
- 425 Can't open data connection
- 452 Error writing file

5. 简单网络管理协议SNMP

网络管理的功能

- ü 故障管理

  故障检测、故障定位、故障报告

- ü 配置管理

  识别网上的设备和用户，维护网上软硬件和电路的精确清单

- ü 性能管理

  资源利用率分析

- ü 计费管理

  对用户使用的各种资源进行跟踪，统计时间

- ü 安全管理

  对用户使用的各种资源进行跟踪，统计时间

- 网络管理系统一般要包含以的元素

① 若干个需要被管理的网络设备节点，每个节点上都运行着一个称为设备代理(agent)的应用进程

② 至少一个管理工作站，它是整个网络管理系统的核心，由网络管理员直接操作和控制

③ 管理协议，用来定义调用网管服务的接口原语以及在网管系统之间进行信息和命令交换的协议数据单元

p 网络管理协议

- § 最有影响的网络管理协议是SNMP(Simple Network Management Protocol)
- § SNMP通常用来管理网络设备并获取设备信息
- § SNMP是TCP/IP协议族中的一员，但它并不依赖于IP。它被设计成与协议无关，所以它可以在IP、IPX、AppleTalk、OSI以及其他用到的传输协议上被使用
- § 现有三个版本： SNMPv1、 SNMPv2和SNMPv3

p SNMP的三个版本

§ SNMPv1的优点是非常简单，但是不能有效地传送大块的数据，在安全性方面还存在着缺欠，从而在某些领域妨碍了SNMP协议的使用

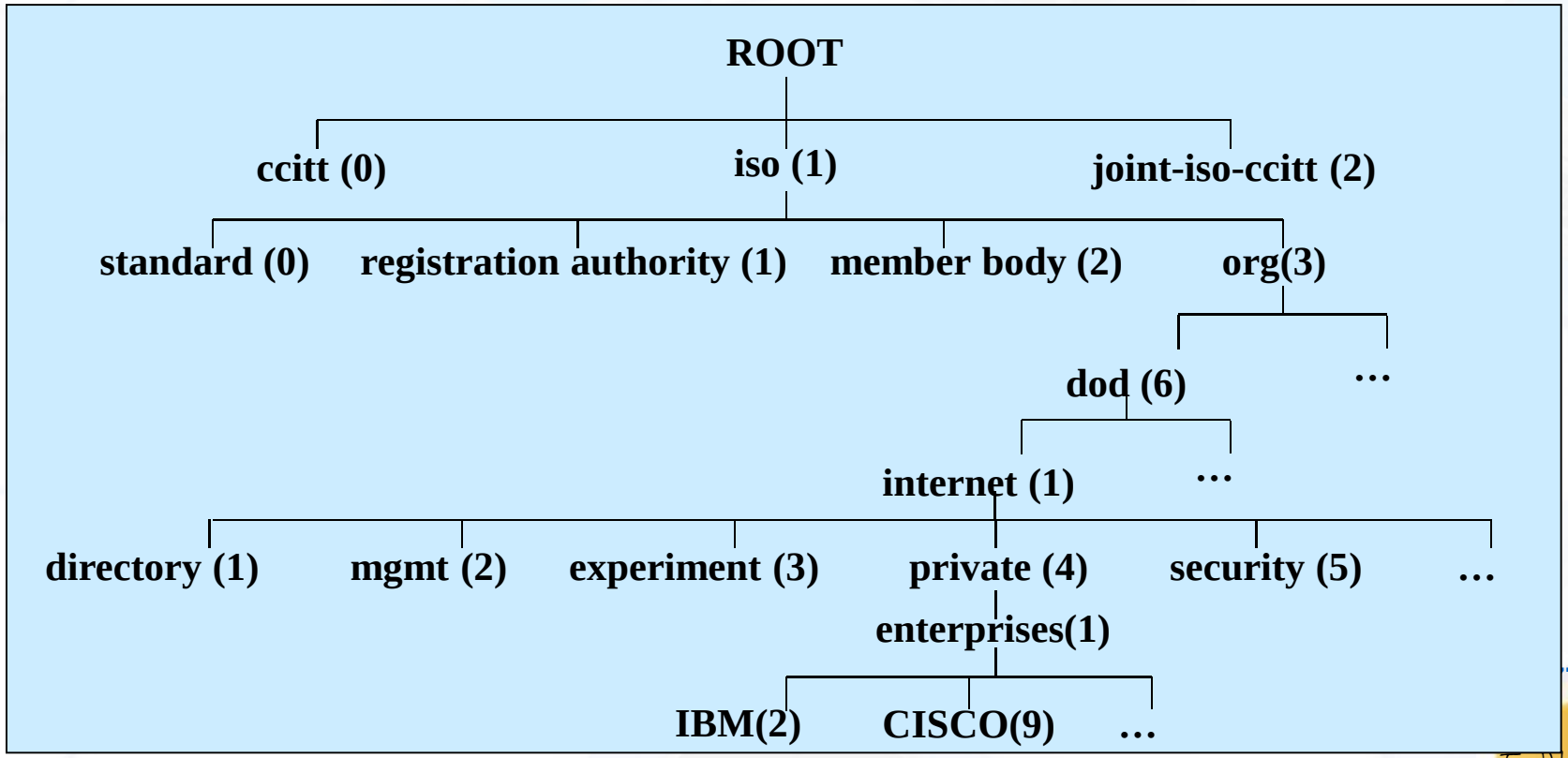
§ SNMPv2在SNMPv1的基础上增加了一些新的功能。对于服务器来说，最方便的命令为get-bulk操作。此外，SNMPv2比SNMPv1有更强的安全性

§ SNMPv3最大的改进就是安全特性

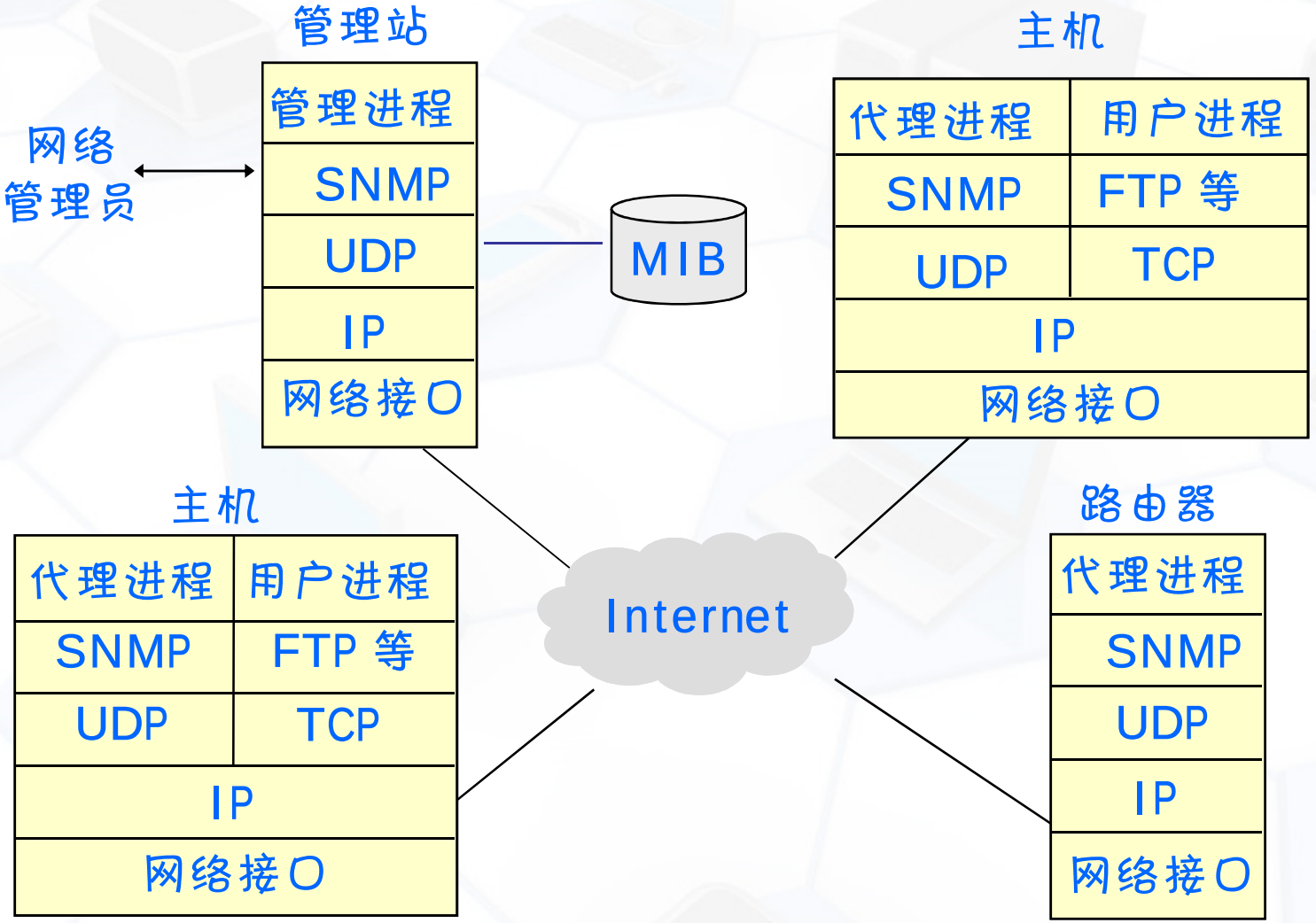
- SNMP的组成
  - 管理信息基库(MIB：Management Information Base)：包含状态信息的数据库
  - 管理信息的结构(SMI:Structure of Management Information)：定义MIB的入口
  - 简单网络管理协议(SNMP)：受管理的对象与服务器间的通信方法

1) MIB及SMI

- MIB是对通过管理协议可以访问的信息的精确定义，是一个网络中所有可能的被管理对象的集合的数据结构
- MIB是一个按照层次结构组织的树状结构(类似于域名系统)



2) SNMP管理模型


管理站
网络
管理员
管理进程
SNMP
UDP
IP
网络接口
MIB
主机
代理进程
用户进程
SNMP
FTP 等
UDP
TCP
IP
网络接口
主机
代理进程
用户进程
SNMP
FTP 等
UDP
TCP
IP
网络接口
Internet
路由器
代理进程
SNMP
UDP
IP
网络接口

p SNMP通信

§ SNMP报文使用UDP来传送，端口采用161/162

- ü 161端口被设备代理监听，等待接受管理进程发送的管理信息查询请求消息
- ü 162端口由管理进程监听，等待设备代理进程发送的异常事件报告陷阱消息，如Trap

§ 设备与服务器有两种通信方式

- ü 投票
- ü 中断

§ 直接自陷投票(trap-directed polling)

- ü 将投票与中断结合使用以弥补各自的缺陷

6. 动态主机配置协议DHCP

动态主机配置协议DHCP(Dynamic Host Configuration Protocol)被称作是引导程序协议BOOTP的扩展

- BOOTP也称为自举协议，使用UDP使一个无盘工作站自动获取配置信息。但是它是一个静态配置协议而不是动态的
- DHCP增加了许多功能
- 与BOOTP协议格式仍然可以兼容

p DHCP特点

§ DHCP网络运行环境包括三部分软件

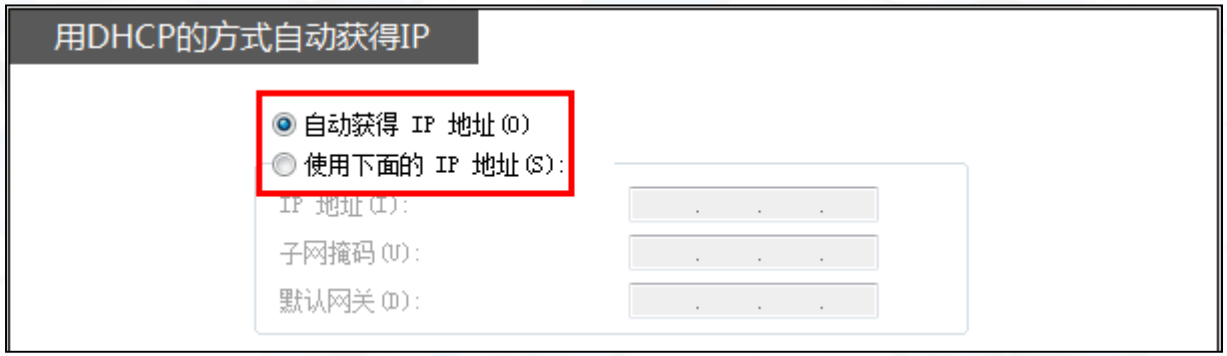
- ü DHCP客户机软件
- ü DHCP服务器
- ü DHCP中继代理

§ DHCP服务器可以根据请求向客户机分配所有现有的未分配地址，每次这样的分配称为一次地址出租

1） DHCP的工作流程


DHCP数据发送过程
1
客户端广播DHCP Discover消息
服务器提供地址租约（Offer）
2
3
客户端选择并请求地址租用（Request）
服务器确认将地址租用给客户端（ACK）
4
DHCP客户端
DHCP服务器

DHCP的客户端把网卡的IP地址配置为动态获取方式，就会发送DHCP请求，来寻找DHCP服务器申请地址



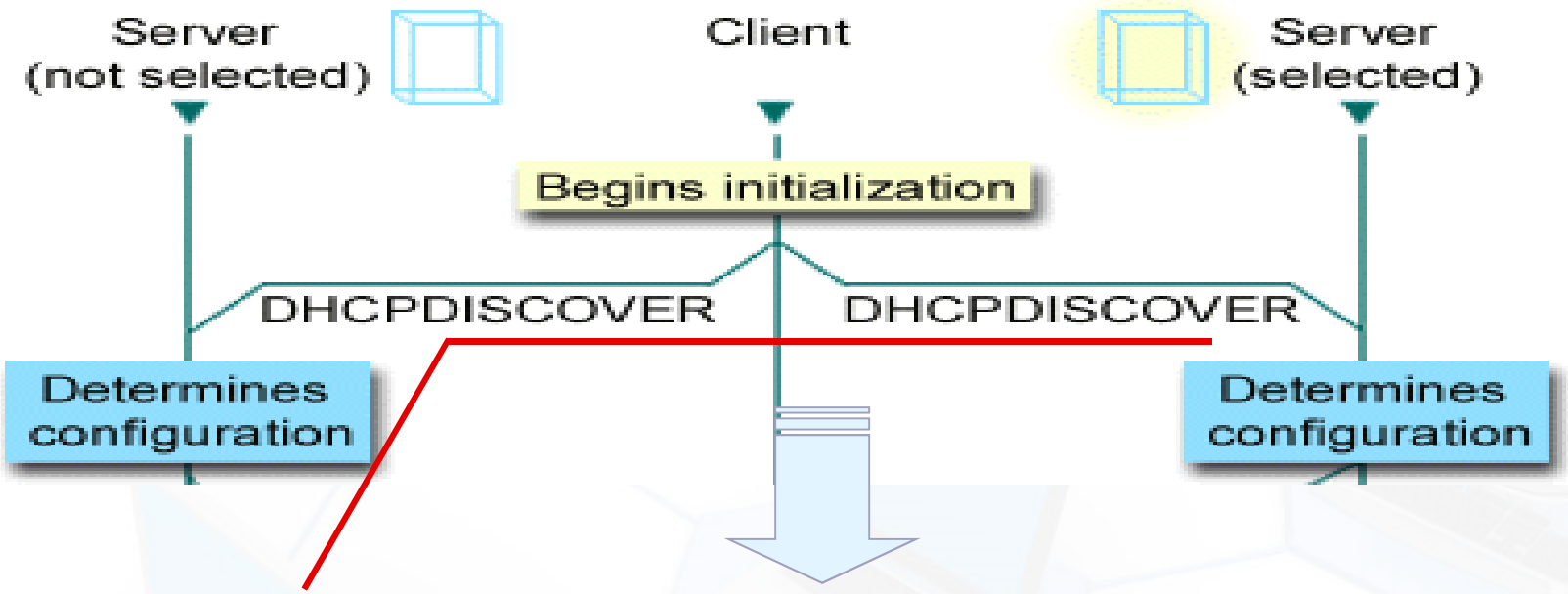
DHCP的在客户端申请、服务器下发地址的过程中，一共会产生四个数据包：

DHCP的数据包

Filter: bootp Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Length	Info
7	9.940756	0.0.0.0	255.255.255.255	DHCP	618	DHCP Discover - Transaction ID 0x71f
12	11.965171	100.1.1.1	255.255.255.255	DHCP	342	DHCP Offer - Transaction ID 0x71f
13	11.980845	0.0.0.0	255.255.255.255	DHCP	618	DHCP Request - Transaction ID 0x71f
14	12.014926	100.1.1.1	255.255.255.255	DHCP	342	DHCP ACK - Transaction ID 0x71f

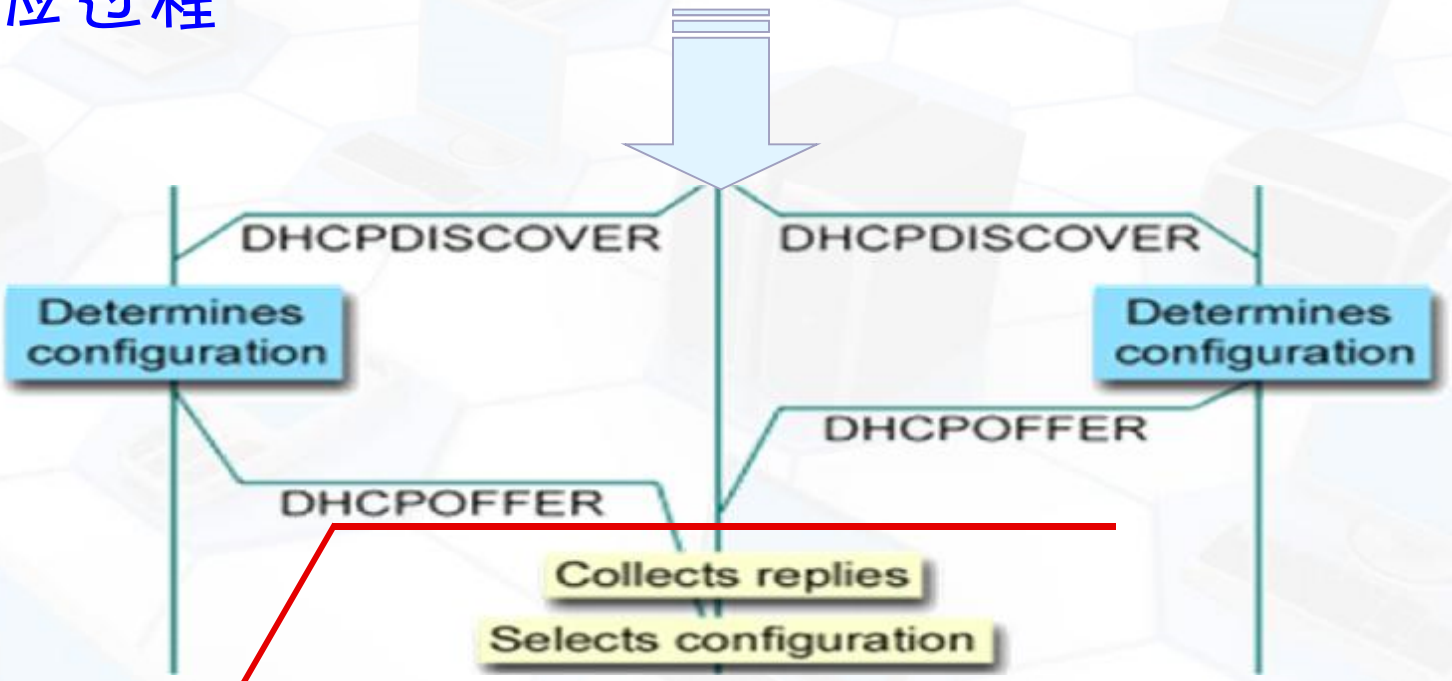
① DHCP发现过程



客户端在网络上广播DHCPDISCOVER消息，DHCPDISCOVER消息使用MAC地址来标识客户端，并且可能包含IP地址和租期的建议的值

源MAC是自己的MAC地址，目的MAC是FFFF.FFFF.FFFF广播；源IP是0.0.0.0（现在还没有IP，就用全0地址），目的IP是255.255.255.255的三层广播，因为DHCP服务器在哪里还不知道，所以使用广播来寻找

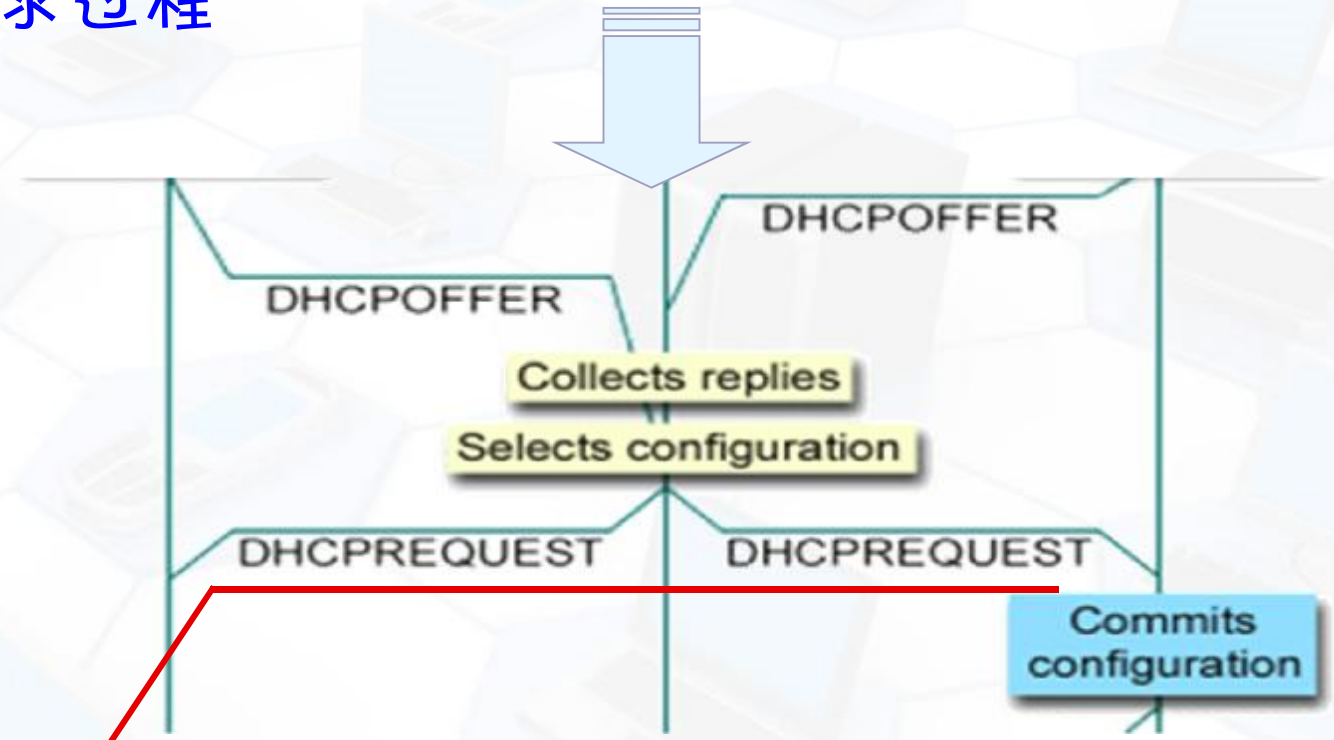
② DHCP回应过程



DHCP服务器回应DHCPOFFER消息给客户端，其中包含分配的IP地址、租用期限和其它配置参数，服务器应该确保分配的IP地址没有被使用

DHCP Offer数据包的地址如下：源MAC是DHCP服务器的MAC，目的MAC是FFFF.FFFF.FFFF广播；源IP是DHCP服务器的IP，目的IP是255.255.255.255广播；这时客户端还没有获得IP，DHCP服务器端现在还无法定位客户端，所以用广播来回应

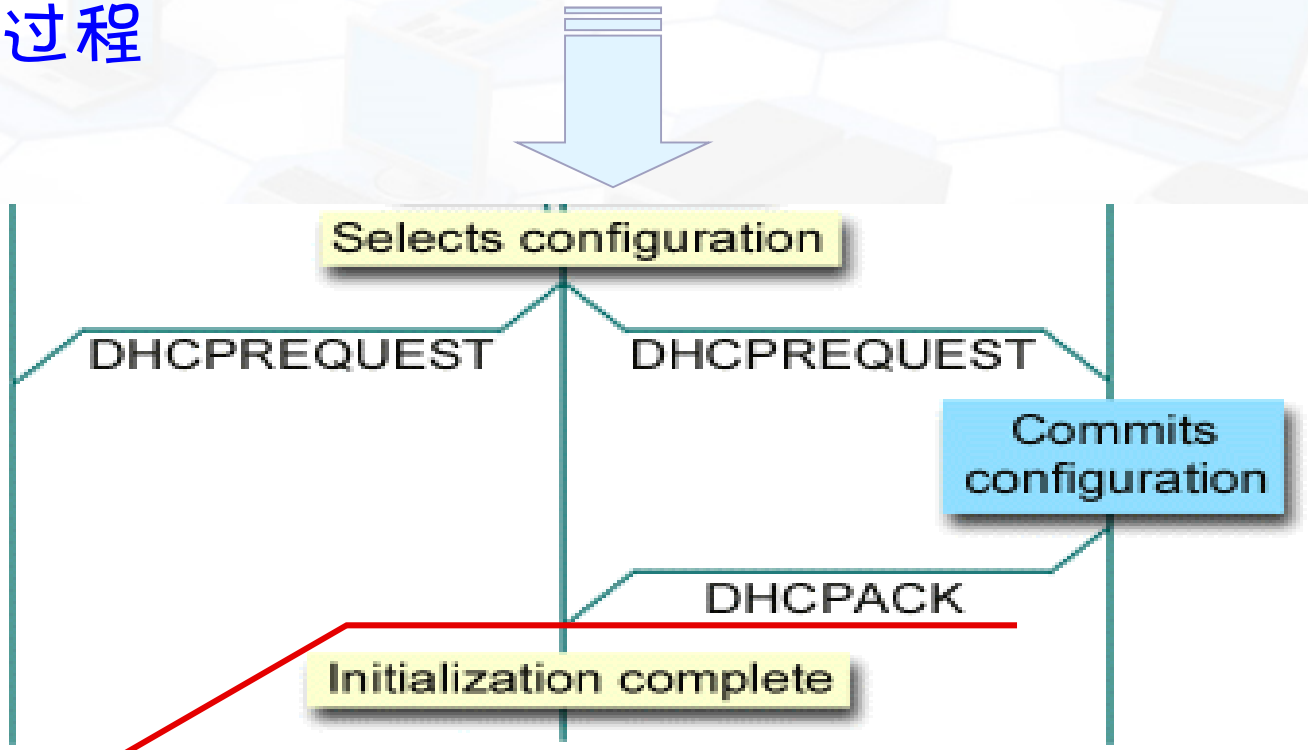
③ DHCP 请求过程



客户端选择一个DHCP服务器并且在整个网络上广播DHCPREQUEST消息，告诉其它DHCP服务器它选择哪个服务器提供的IP地址

客户端还没有正式拿到地址，所以还需要向DHCP服务器申请。这时客户端的源IP还是0.0.0.0，目的IP还是255.255.255.255；源MAC是客户端的MAC，目的MAC是FFFF.FFFF.FFFF广播包

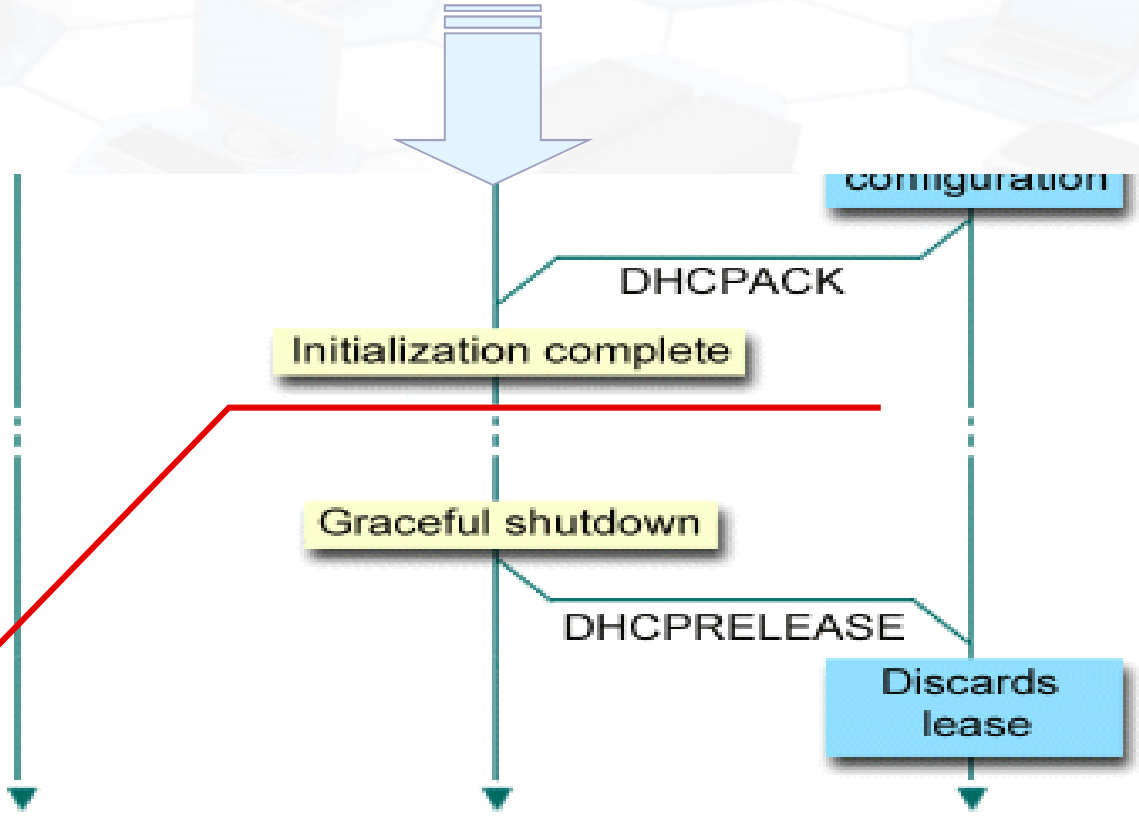
④ DHCP确认过程



DHCPREQUEST选定的服务器给客户端回应DHCPACK，其中包含有配置参数

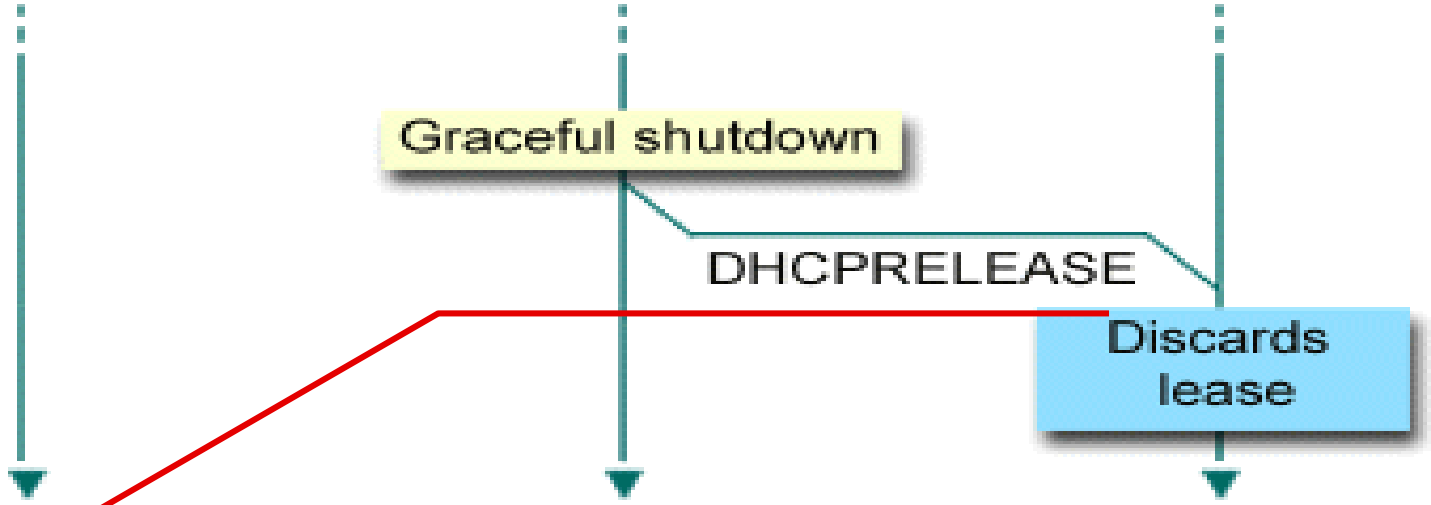
服务器收到客户端的请求后，会发出一个DHCP ACK用来确认这个IP地址可以分配给客户端。客户端收到第四个DHCP ACK数据包才算正式得到这个IP

⑤ 客户端完成配置



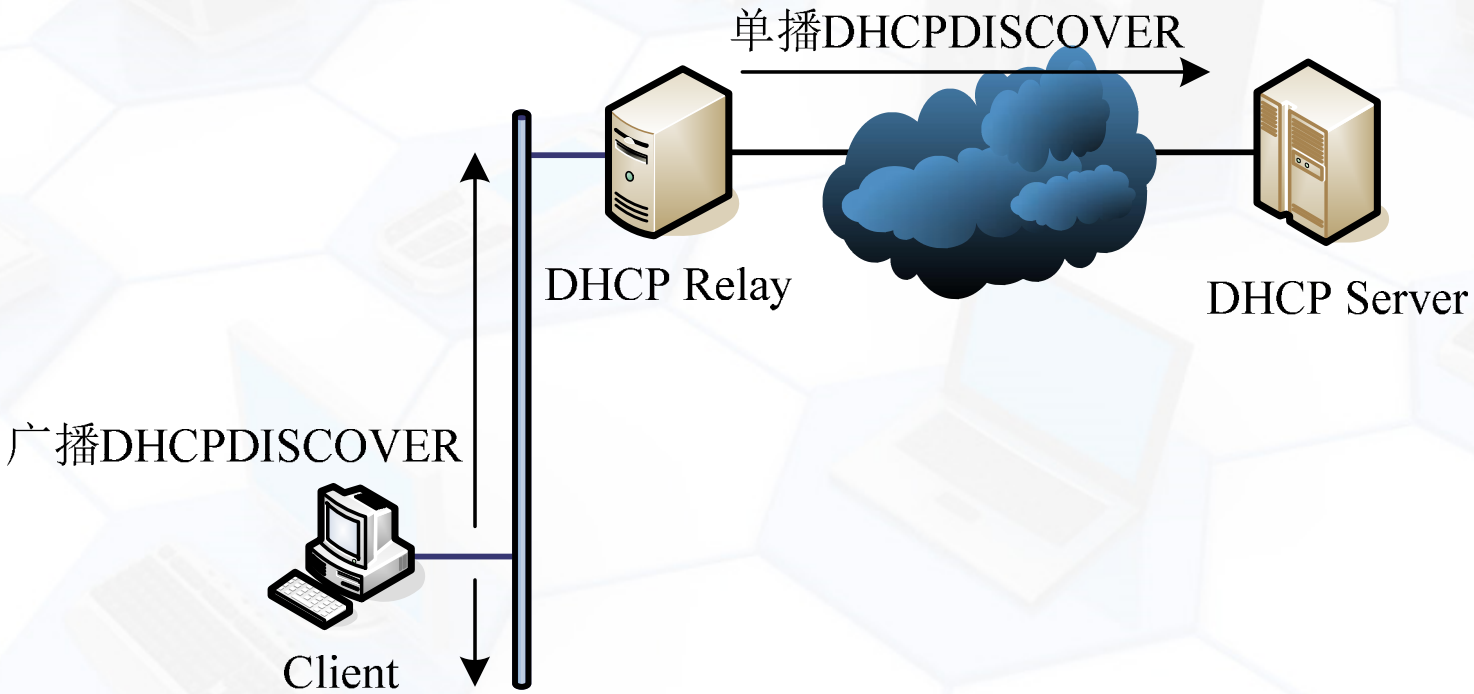
客户端接收到DHCPACK后，应该对参数进行最后的检查，例如对分配的IP地址进行ARP过程，然后完成相应的配置

⑥ 地址的续租与释放过程



客户端可以通过向DHCP服务器发送DHCPRELEASE消息来释放对IP地址的租用，其中包含有客户端的MAC地址和租用IP地址

2）DHCP中继


单播DHCPDISCOVER
DHCP Relay
DHCP Server
广播DHCPDISCOVER
Client