北京郵電大學

计算机网络

第二章 应用层
Application Layer

网络空间安全学院

liujy@bupt.edu.cn

主要内容

- 2.1 网络应用概述
- 2.2 DNS
- 2.3 WWW应用和HTTP
- 2.4 Email应用
- 2.5 FTP
- 2.6 远程登录协议：Telnet
- 2.7 应用层安全隐患

教学目标

- 掌握应用层的基本概念
  - 应用层的体系结构
    - C/S模型
    - P2P模型
  - 进程与端口号
- 掌握典型的应用层协议
  - DNS
  - HTTP
  - SMTP、POP3

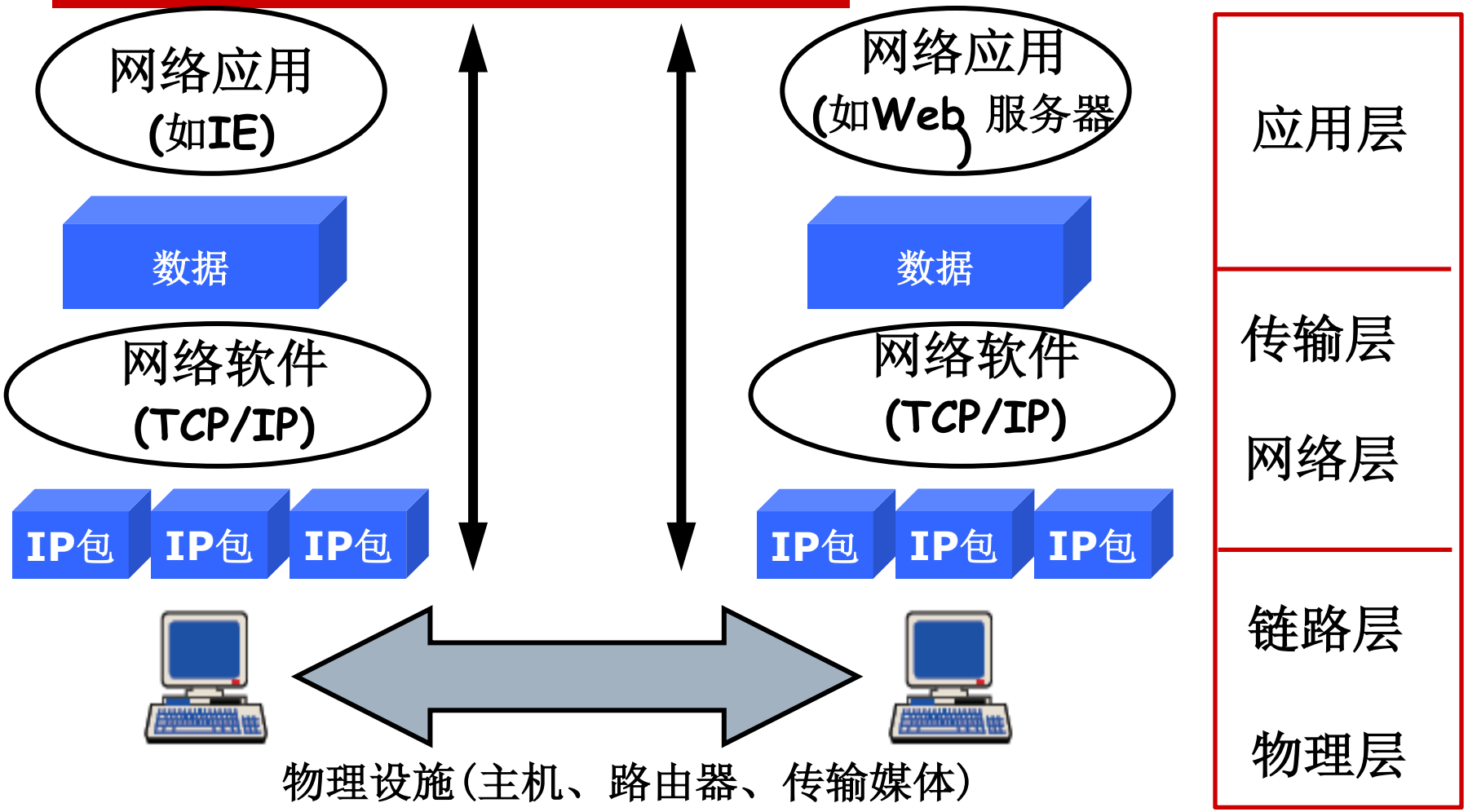
主要内容

- *2.1 网络应用概述*
  - 网络应用的发展史
  - 应用层的体系结构：C/S和P2P
  - 进程通信及端口号
  - 应用对于传输服务的要求
- 2.2 DNS
- 2.3 WWW应用和HTTP
- 2.4 Email应用
- 2.5 FTP
- 2.6 远程登录协议：Telnet
- 2.7 应用层安全隐患

网络应用的发展史

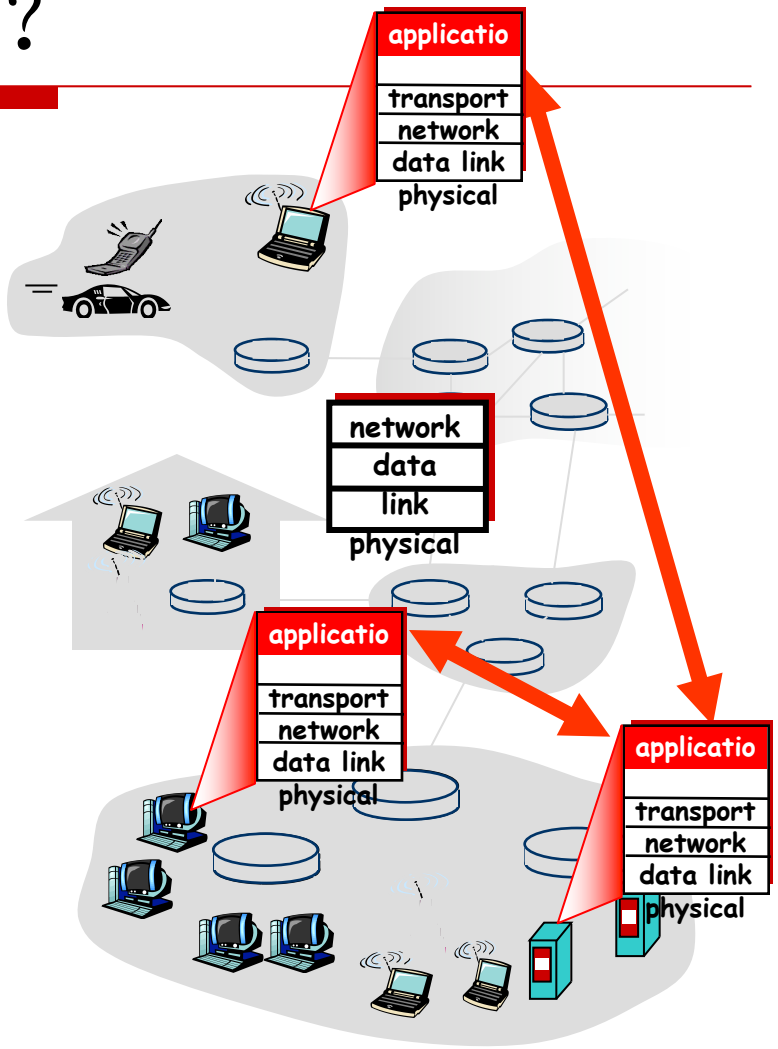
出现时间	应用	发明者	应用层协议
1969	远程登录、BBS	连接ARPANet的4所大学的学生	Telnet
1971	Email	Ray Tomlinson	SMTP、POP3
1971	文件传输	RFC114	FTP
1989	WWW	Tim Berners Lee	HTTP
1994	Web Blog	Justin Hall	HTTP
1995	Amazon（电子商务）	Jeff Bezos	HTTP
1995	IP电话（VoIP）	VocalTec公司	SIP、RTP
1995	音频/视频点播	Xing技术公司	RTP
1996	OICQ（即时消息）	Yair Goldfinger等	私有协议
1999	Napster（P2P文件共享）	Shawn Fanning	私有协议

网络应用的通信过程


网络应用
(如IE)
网络应用
(如Web 服务器)
数据
数据
网络软件
(TCP/IP)
网络软件
(TCP/IP)
IP包
IP包
IP包
IP包
IP包
IP包
物理设施(主机、路由器、传输媒体)
应用层
传输层
网络层
链路层
物理层

如何提供一个网络应用？

- *编写应用程序*
  - 在不同的端系统上运行
  - 通过网络和其他端系统通信
    - 遵守特定的协议
  - 示例：web server软件和浏览器软件通信
- *无需修改网络核心软件*
  - 网络核心设备不运行网络应用程序
  - 方便实现快速的应用开发和部署



应用层协议的内容

- 消息类型
  - 例如：请求、应答
- 语法（Syntax）
  - 消息中包含哪些字段，每个字段的长度
- 语义 (Semantics)
  - 每个字段信息的含义
- 时序：消息的顺序

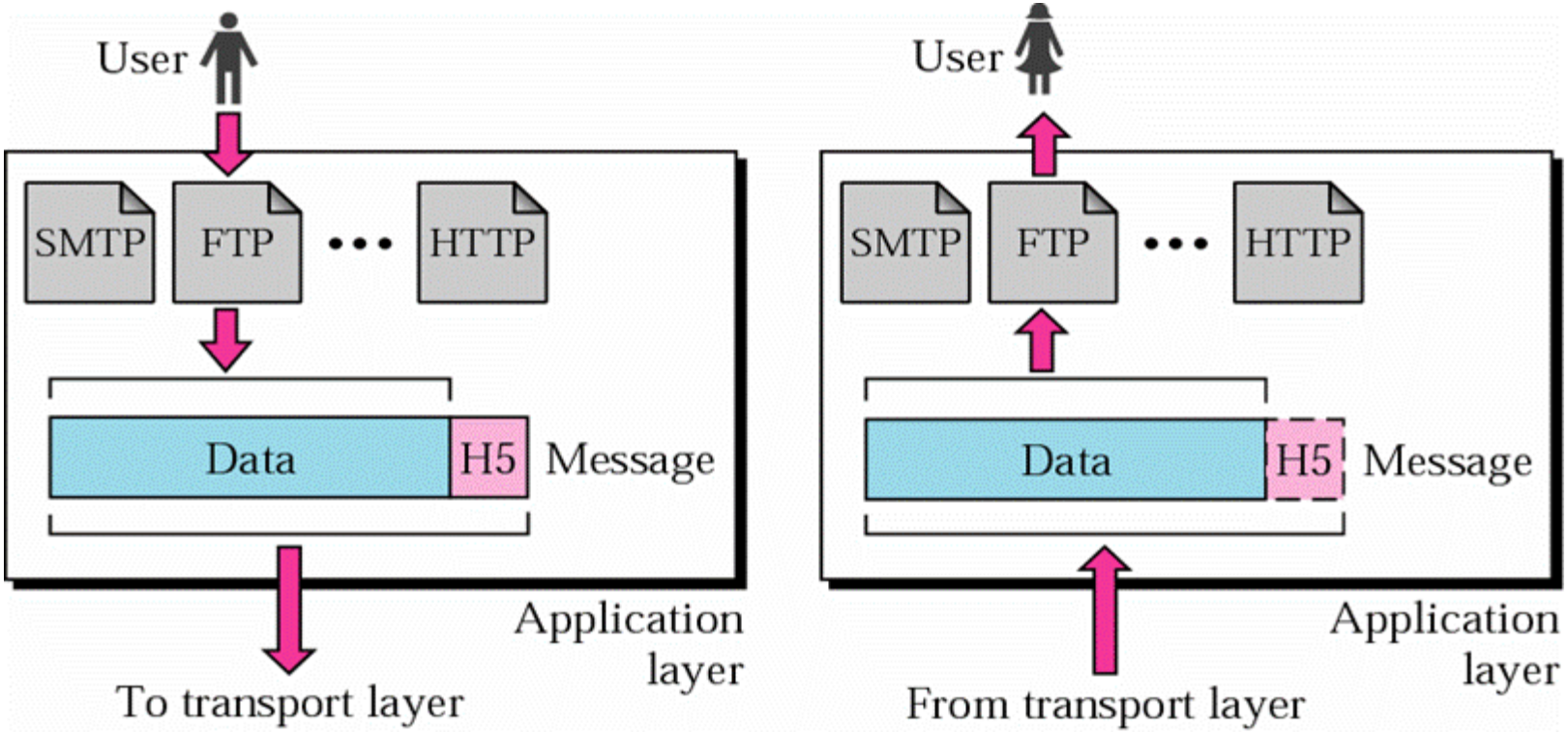
请求行 → `GET /somedir/page.html HTTP/1.1`

消息头（可选）

```
Host: www.abc.edu.cn
User-agent: Mozilla/4.0
Accept-language:zh-cn
Connection: Keep-Alive
```

应用层协议的特点

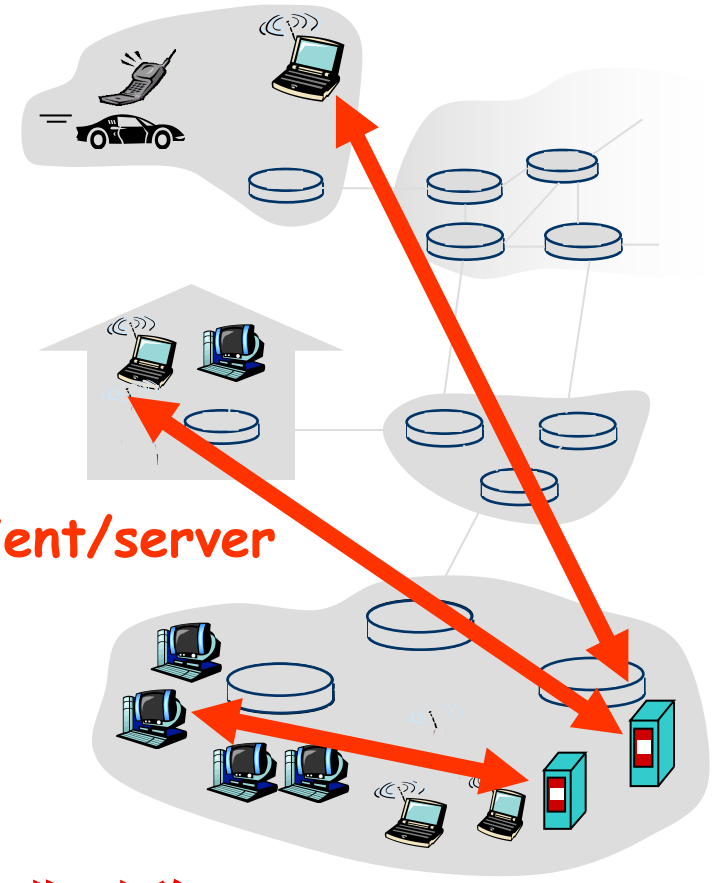
- 面向用户，支持特定的网络应用
  - *没有通用的协议，每个应用有专门的协议！*
  - *相比其他层，协议数量最多、最复杂*



网络应用的体系结构

- Client-server(C/S)
- Peer-to-peer (P2P)
- C/S和P2P混合结构

C/S 体系结构



- *服务器:*
  - 提供服务，同时处理多个客户请求
  - 一直在线
  - 地址(域名)公开，一般不变
  - 可能有多个服务器(server farms)
  - 一般需要高性能硬件支持
- *客户:*
  - 与服务器通信，使用服务
  - 临时连接到网络
  - IP地址可能是动态的
  - 客户之间不直接通信

几个概念

- 客户和服务器是计算机进程，即运行着的程序（软件）。
- 使用计算机的人是计算机的“用户”(user)，而不是“客户”(client)
- 有些外国文献把运行客户程序的机器称为client，把运行服务器程序的机器称为server，需要根据上下文判断。一般我们称为客户端（或客户机）和服务器端。
- 应如何理解“两个计算机进行通信”？

采用C/S模型的应用

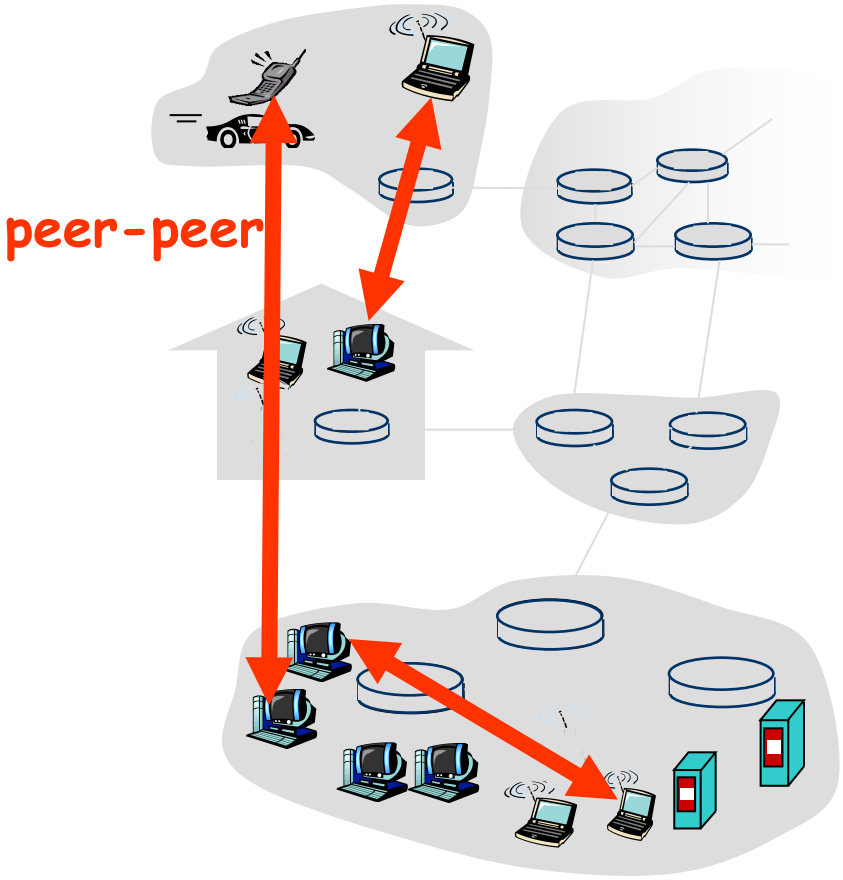
- WWW
- Email
- DNS
- 文件传输：FTP
- BBS、远程登录
- 网络管理：SNMP
- ……

P2P体系结构

- Peer –to-Peer
- 没有严格的服务器-客户机之分，Peer
- 没有一直在线的服务器，Peer在必要时充当服务器
- Peer之间直接通信
- Peer临时连接到网络，且IP地址可能动态变化

可扩展、成本低，

但难于管理、有安全隐患

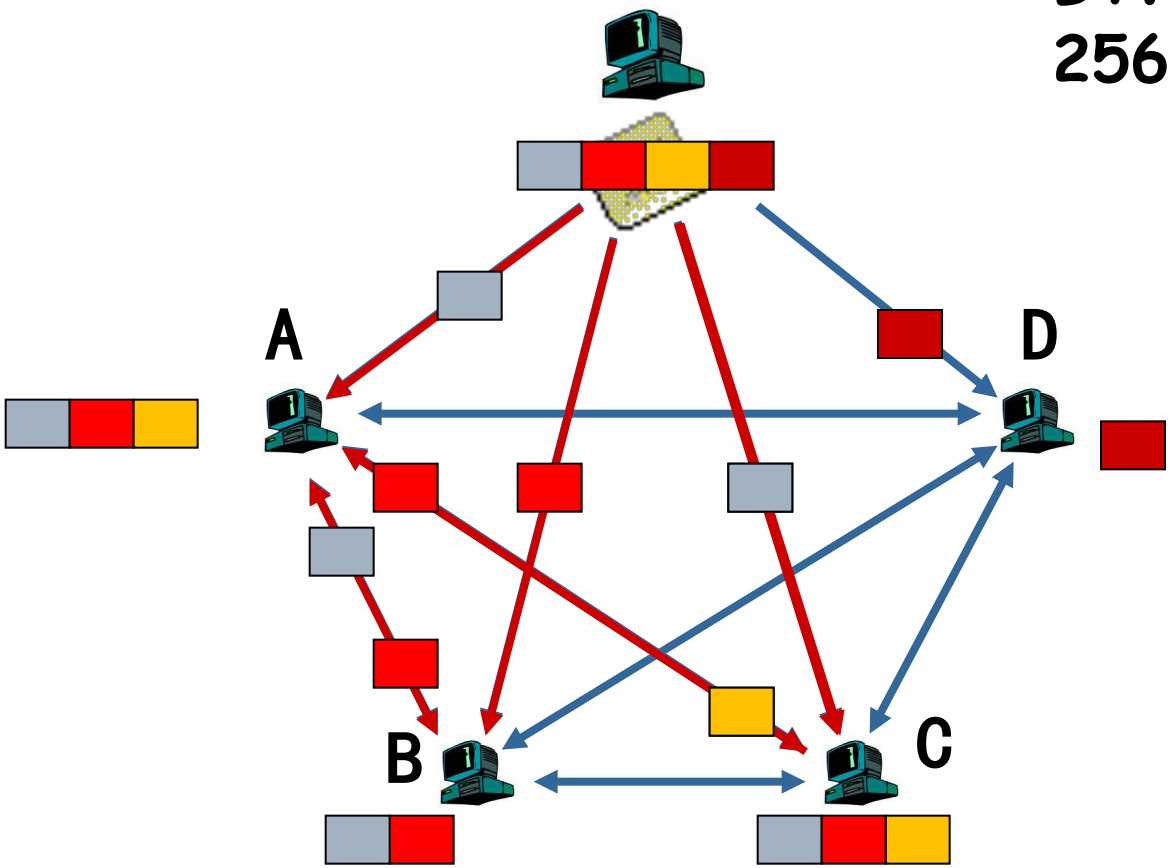


采用P2P模型的应用

- 文件下载（共享）
  - Napster、BT、迅雷....
- 流媒体
  - 视频点播：风行
  - IP电视：PPLive
- 网格计算（Grid Computing）
  - Distribute.net、SETI@Home
- 分布式存储
  - OceanStore

P2P文件下载：Peer之间互相传输文件

BT：将文件分为256KB的数据块



P2P文件下载示例：星球大战

d8:announce34:http://tracker.ydy.com:86/announce10:createdby13:BitComet/0.5813:creationdatei1117953113e8:encoding3:GBK4:infod6:lengthi474499162e4:name51:05.262005.StarWars Episode IV A New Hope-Rv9.rmvb10:name.utf-851:05.26.2005.Star WasEpisode IV A New Hope-Rv9.rmvb12:piecelengthi262144e6:pieces36220:XXXXXXXXXXXXXXXX(SHA1杂凑值)

表示了如下信息:

Tracker地址：http://tracker.ydy.com:86/announce

创建者：BitComet/0.58

创建时间:1970-1-1 00:00秒后1117953113秒.即Sun Jun 5 14:31:53 2005.

编码技术：是BitComet的扩展，实际上用了UTF-8就不需要GBK.

info: (这是单文件模式的代表)

大小:474499162(452Mb)

文件名:05.262005.Star Wars Episode IV A New Hope-Rv9.rmvb

name.utf-8:也是BitComet的扩展,指出文件名编码不是GBK而是UTF-8.

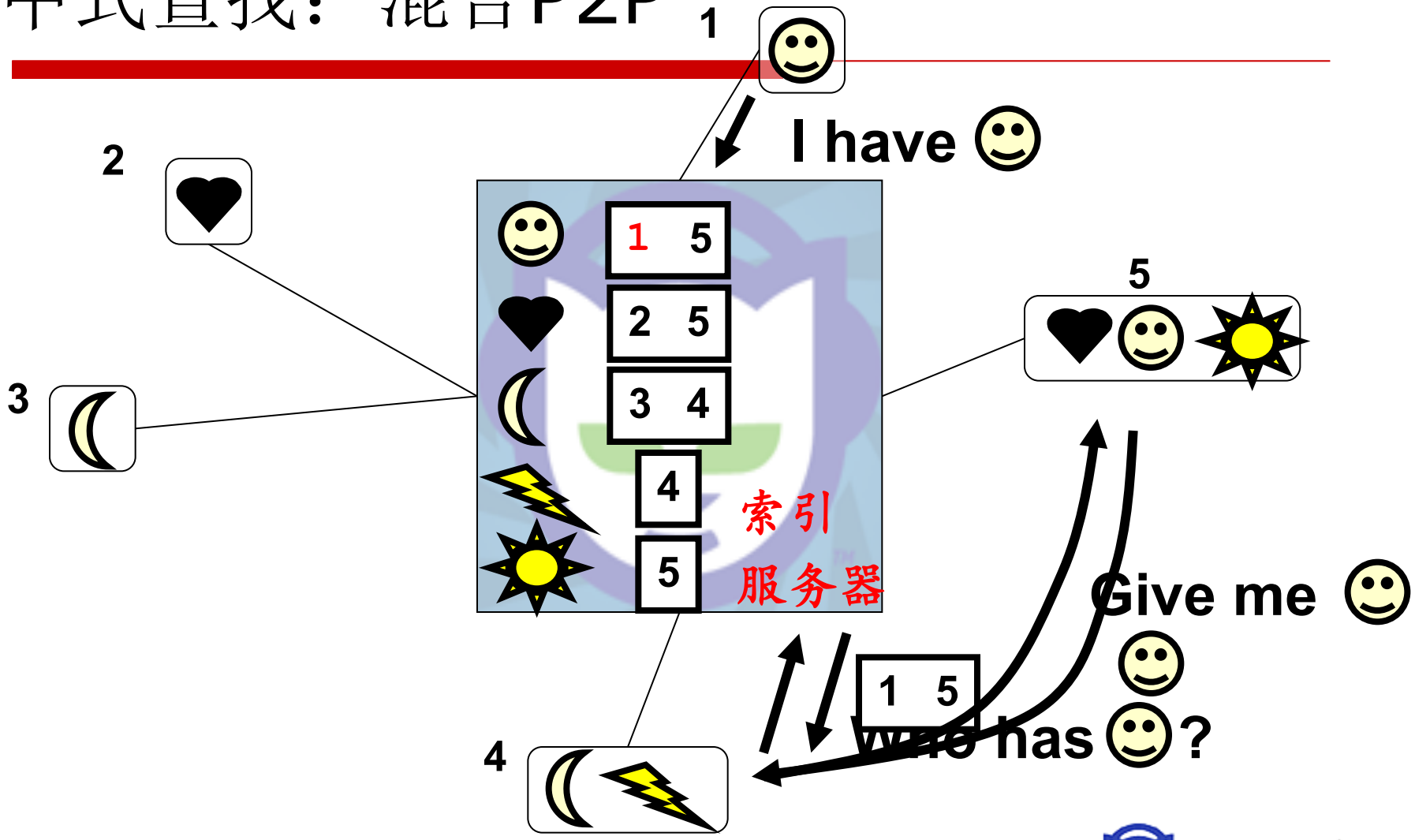
文件块大小:262144字节(256KB)

pieces:长度为366220的SHA1杂凑值（hash）内容，由于每一个文件块20字节SHA1杂凑值,可见文件块有36620 / 20 = 1831个

P2P: 如何查找资源？

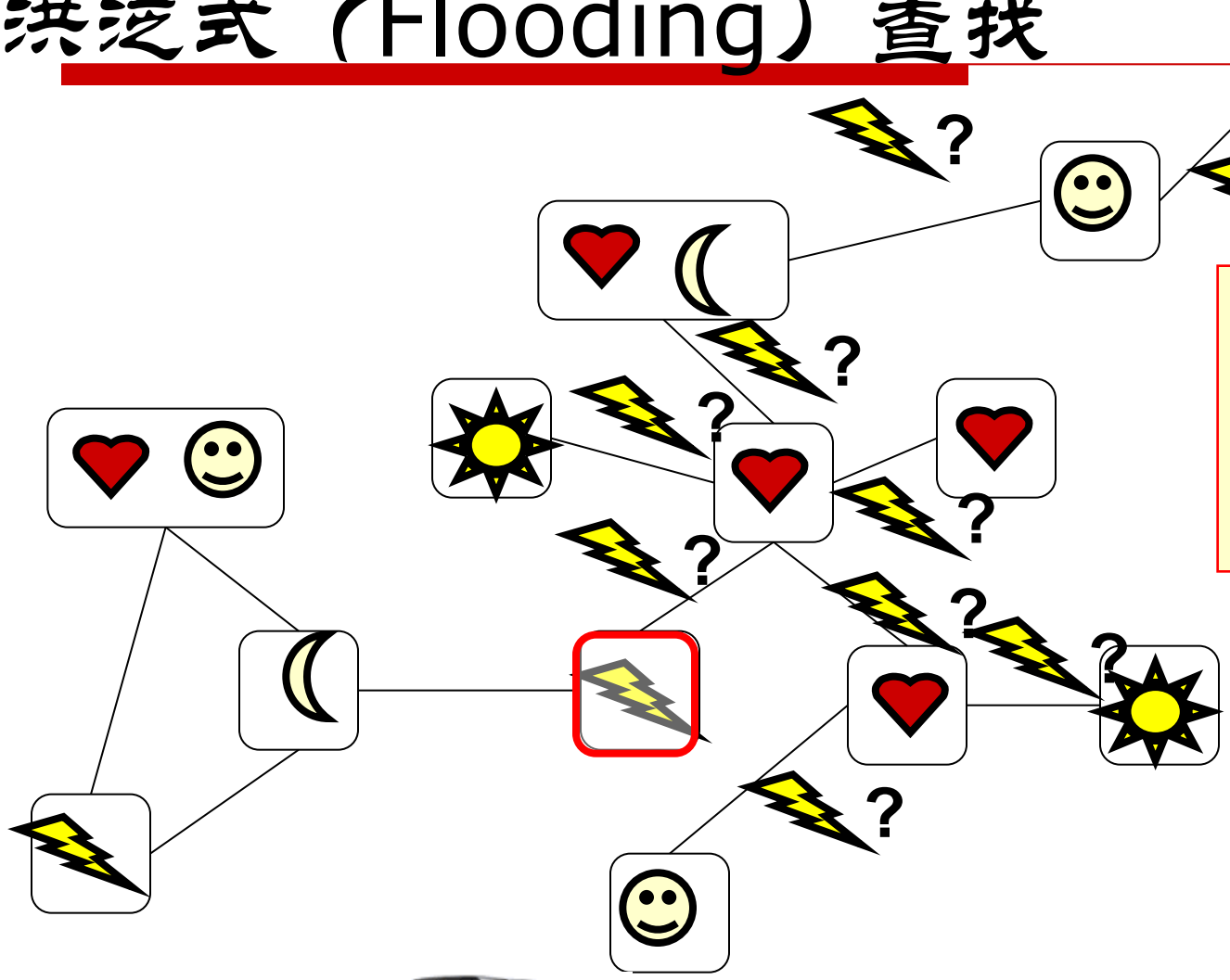
- 集中式：从索引服务器(Index Server)中查找资源，eg. Napster
- 分布式：洪泛查询(Query Flooding)，eg. Gnutella
- 分层叠加网，超级节点提供索引服务，eg.KaZaA, freenet,chord

集中式查找：混合P2P



napster

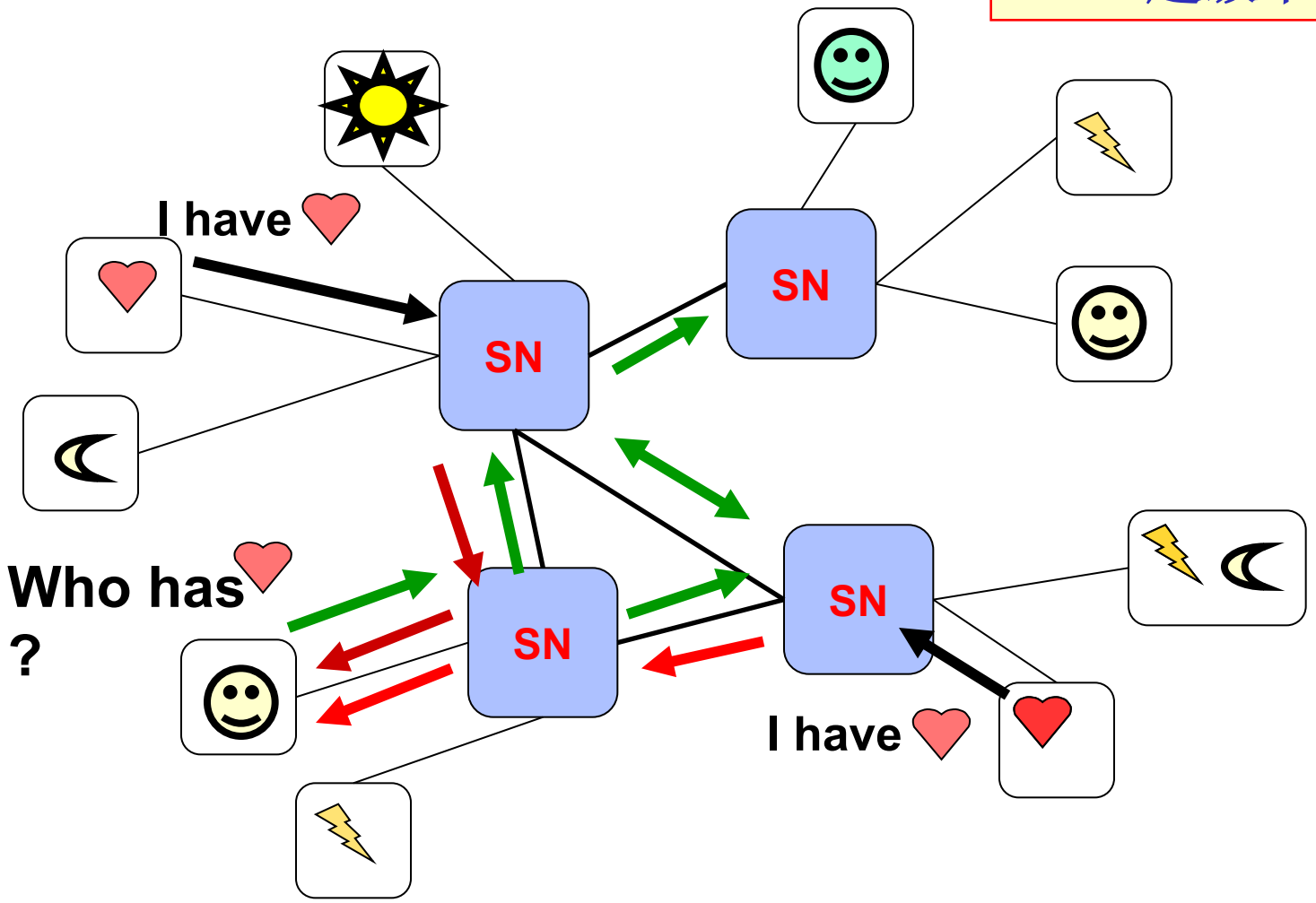
洪泛式（Flooding）查找



洪泛：每个Peer都将查询请求转发给相邻的全部Peer

分层叠加网（Overlay Network）查找

SN：超级节点



C/S和P2P混合结构

- *即时消息QQ*
  - 用户之间聊天：P2P
  - 用户注册、朋友列表、当前状态：C/S
    - 用户到中央服务器注册
    - 用户从中央服务器获得朋友的IP地址

- *Skype*
  - VoIP应用
  - 中央服务器：获得对端用户的地址
  - P2P：用户直接通话

进程通信

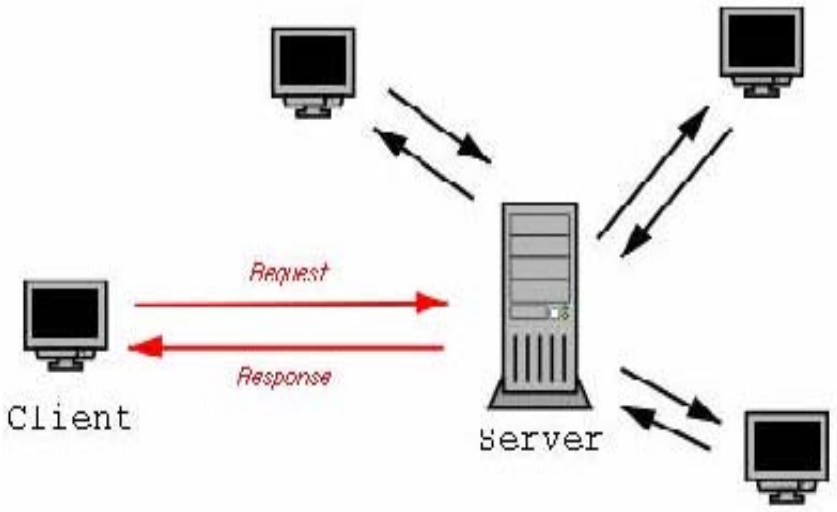
进程(Process)：主机内程序的一次执行

- 在主机内部，进程使用进程间通信机制 (由OS决定)
- 不同主机之间，通过交换消息进行通信

- 注：**P2P应用既运行 client进程也运行 server 进程**

Client进程：发起通信

Server进程：在Client进程之前运行；等待Client进程的请求



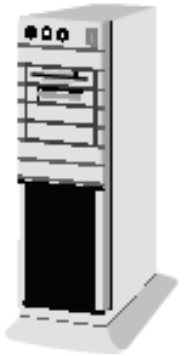
C/S模式的主机间进程通信

client

Client主机

server

Server
主机



IP 地址:
10.2.81.103

Client 应用

应用层协议

IP 地址:
202.112.96.2

Server 应用

client 程序

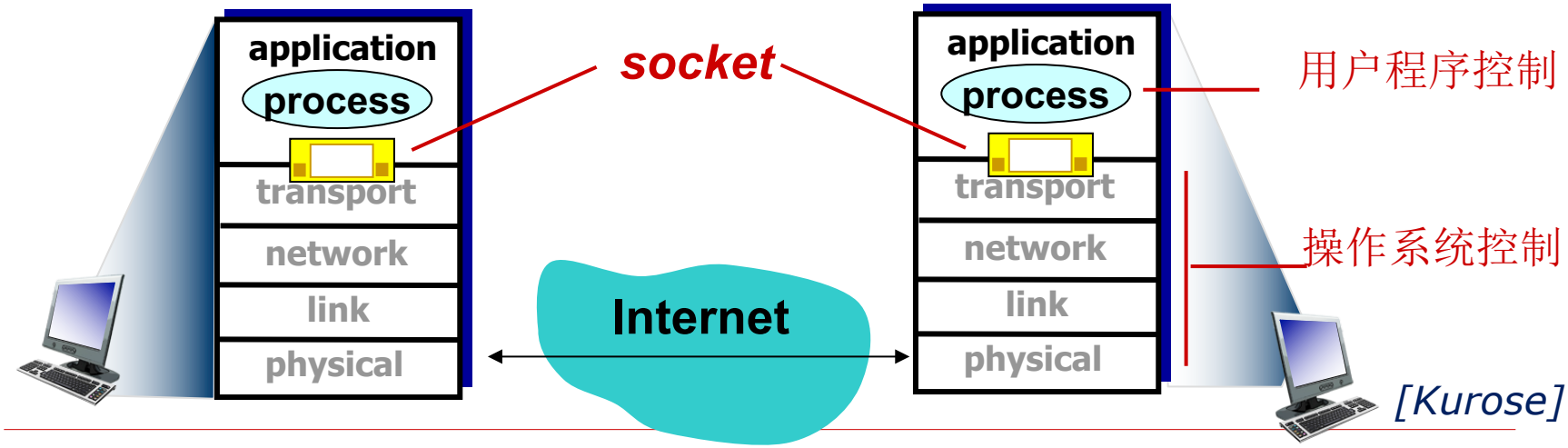
Server程序

Client进程

Server进程

不同主机间通信的接口：Sockets

- 套接字
- 应用进程使用socket来发送/接收消息
- 应用编程接口（API）
  - 位于应用层和传输层 之间
  - 帮助应用访问系统内核（TCP/IP协议软件）
  - 设置相关参数（如最大报文段长度MSS等）



进程的地址

- Internet 上的每台主机都有一个唯一的IP地址
- *Q:* IP地址能唯一标识进程吗?

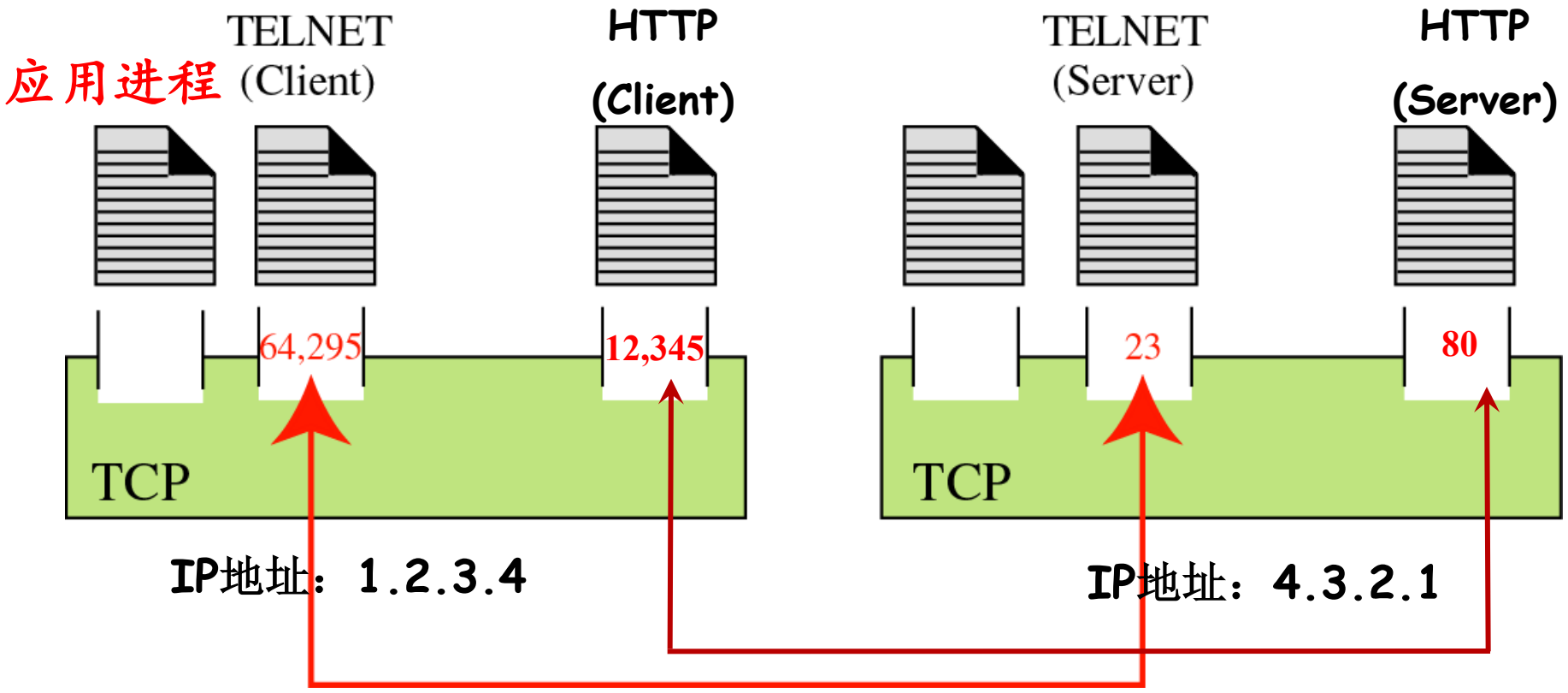
A: 当然不行

一台主机中可能有多个进程在运行

- **进程标识:**

  主机IP地址+进程端口号
- 常用端口号
  - **www Server: 80**
  - **Mail server: 25**
  - **DNS server: 53**
- 进程标识示例:
  - 要访问百度服务器 **www.baidu.com**
  - **IP地址:** 61.135.169.125
  - **端口号: 80**

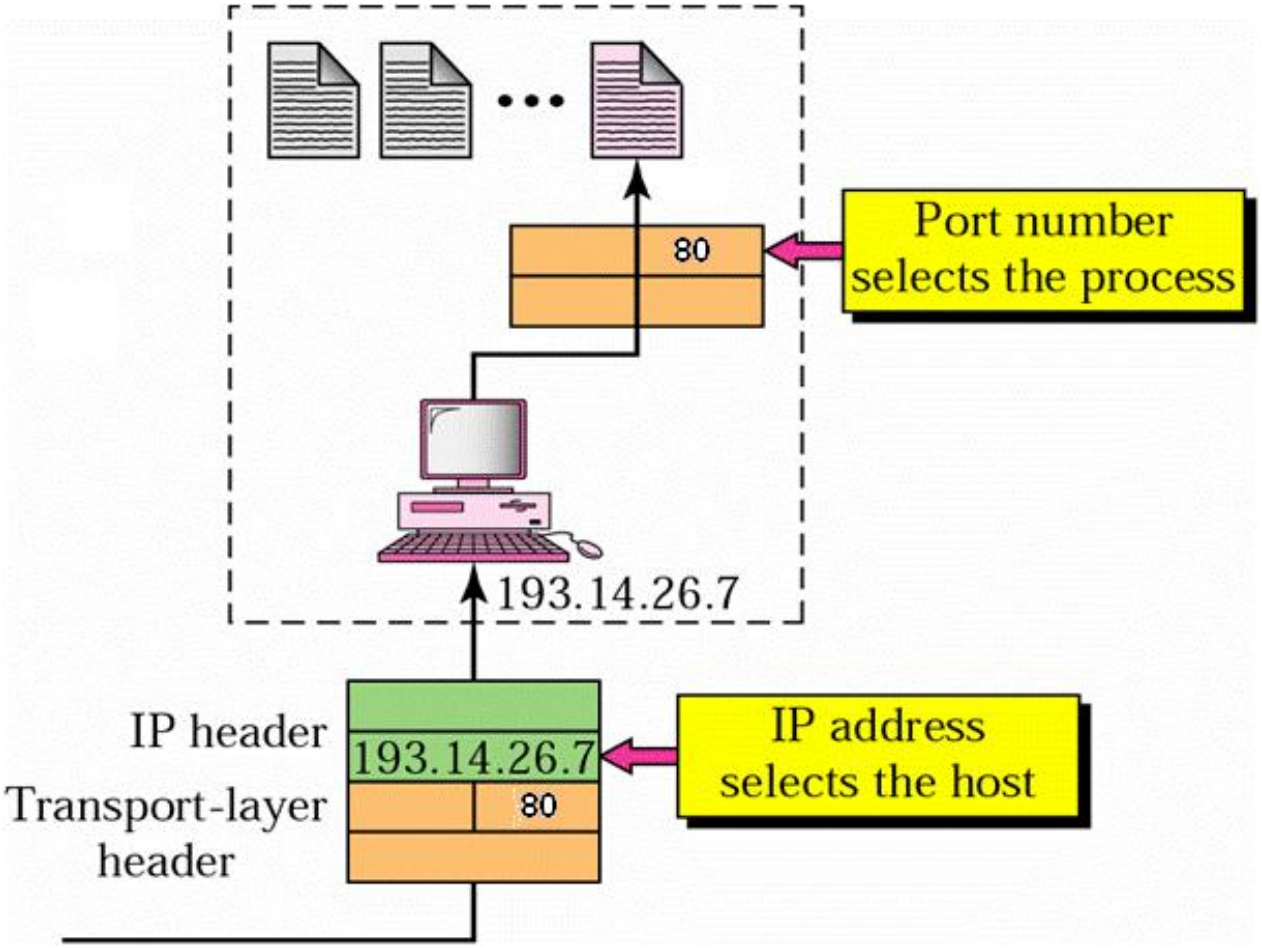
通信关系与端口号



通信关系的标识：

（源主机IP地址，源端口号，目的主机IP地址，目的端口号）

IP地址与端口号示例


80
Port number
selects the process
193.14.26.7
IP header
193.14.26.7
Transport-layer
header
80
IP address
selects the host

网络应用对传输服务的要求：衡量指标

可靠性：数据丢失率

- 可容忍差错，如话音
- 不能容忍差错，如Email

实时性

- 端到端的时延

吞吐量 (Throughput)

- 端到端的带宽
- 音频数据：5kbps-1Mbps
- 视频数据:10kbps-5Mbps

安全性

- 是否加密、是否验证用户身份、能否保证数据一致性

因特网应用的分类（ITU-T G.1010）

可容忍差错	可视电话、视频会议	多媒体即时消息	流媒体	传真
不容忍差错	命令/控制类（Telnet，交互式游戏）	事务类（电子商务、网页浏览、E-mail）	即时消息类、下载类（FTP、图像传输）	背景类（Usenet、论坛、SMS）
时延要求	交互式（1秒内）	响应式（2秒左右）	及时性的（10秒左右）	时延不敏感的（远超过10秒）

因特网的传输层服务

TCP 服务

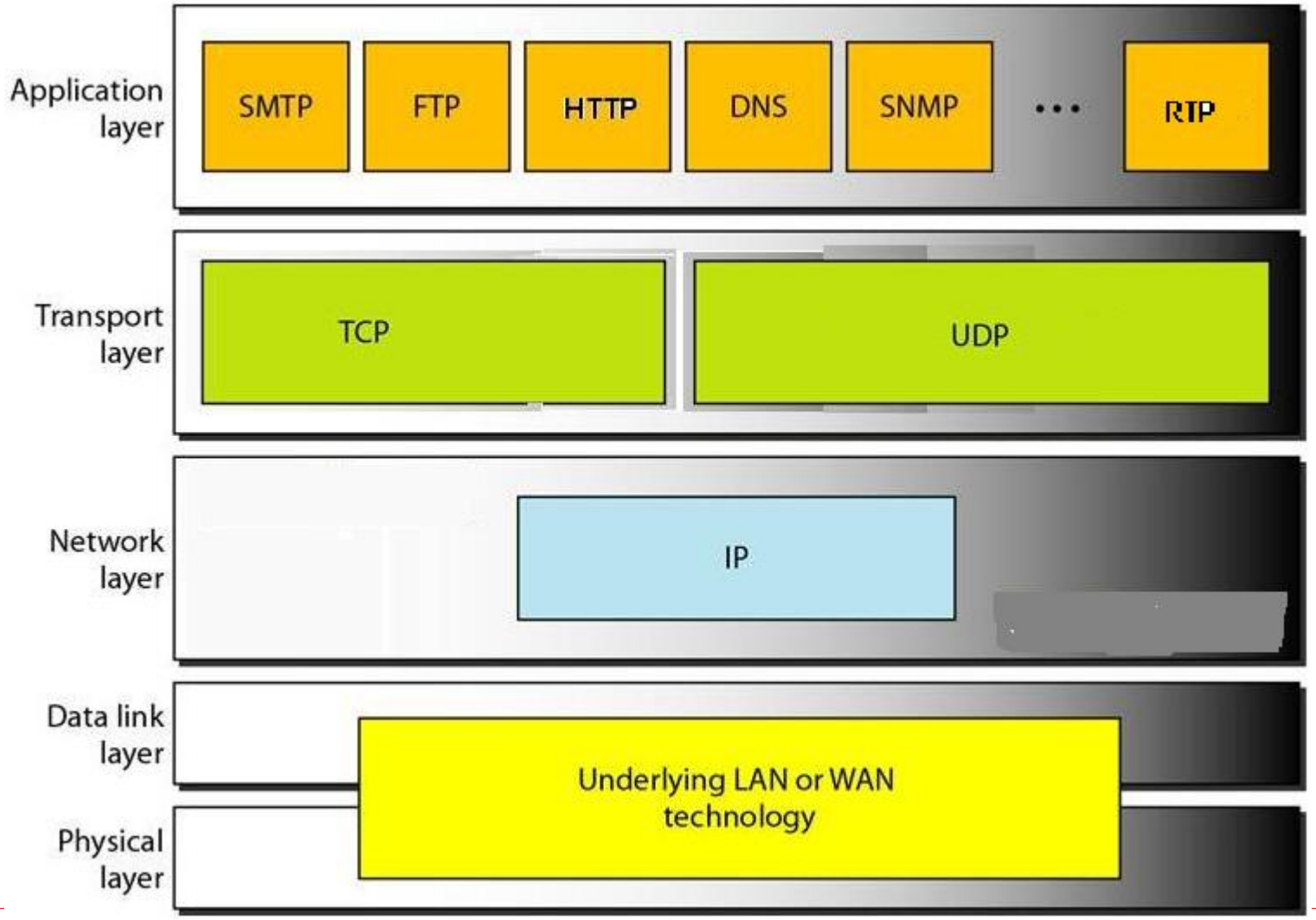
- *面向连接：*传输数据之前需要建立连接
- *可靠传输：*无差错、按序交付
- *流量控制：*可以限制发送速度，不会淹没接收方
- *拥塞控制：*在网络过载时可以限制发送速度
- *不保证：*实时性、最低吞吐量和安全性

Q: 为什么要使用UDP?

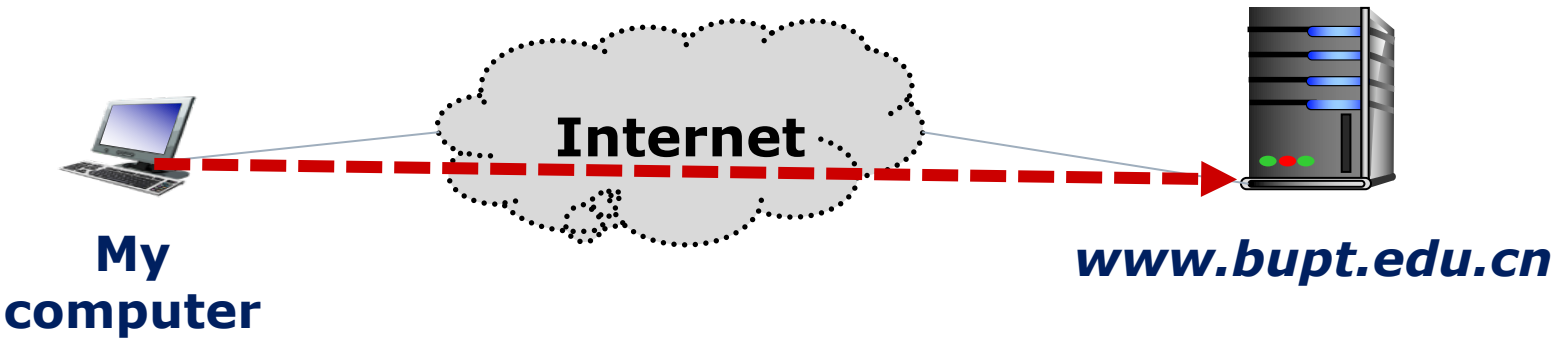
UDP 服务

- *不可靠传输*
  - 无连接
  - 数据可能传错、丢失、重复
  - 可能接收顺序与发送顺序不同
  - 无流量控制
  - 无拥塞控制
  - 不保证实时性、最低吞吐量和安全性

网络应用对传输层协议的选择



用户的应用需求：访问北邮主页



- 这是一个WWW应用
  - 性能要求：可靠性
  - 传输层：TCP
  - 第一步：握手——建立到Web服务器的TCP连接
  - 但是，*www.bupt.edu.cn*对应哪个*IP地址*？

➡ **DNS帮你查**

主要内容

- 2.1 网络应用概述
- *2.2 DNS*
  - DNS的功能和名字空间
  - 递归查询与迭代查询
  - DNS资源记录
  - DNS的消息格式
- 2.3 WWW应用和HTTP
- 2.4 Email应用
- 2.5 FTP
- 2.6 远程登录协议：Telnet
- 2.7 应用层安全隐患

DNS(域名系统)：基本功能

- 因特网上的主机的ID：
  - 名字：域名，如www.bupt.edu.cn，供用户识别

Q：如何完成两者之间的转换？

DNS

  - IP地址：32位，用于传输IP包，由主机和路由器识别
  - *Q：如何完成两者之间的转换？*

ARP

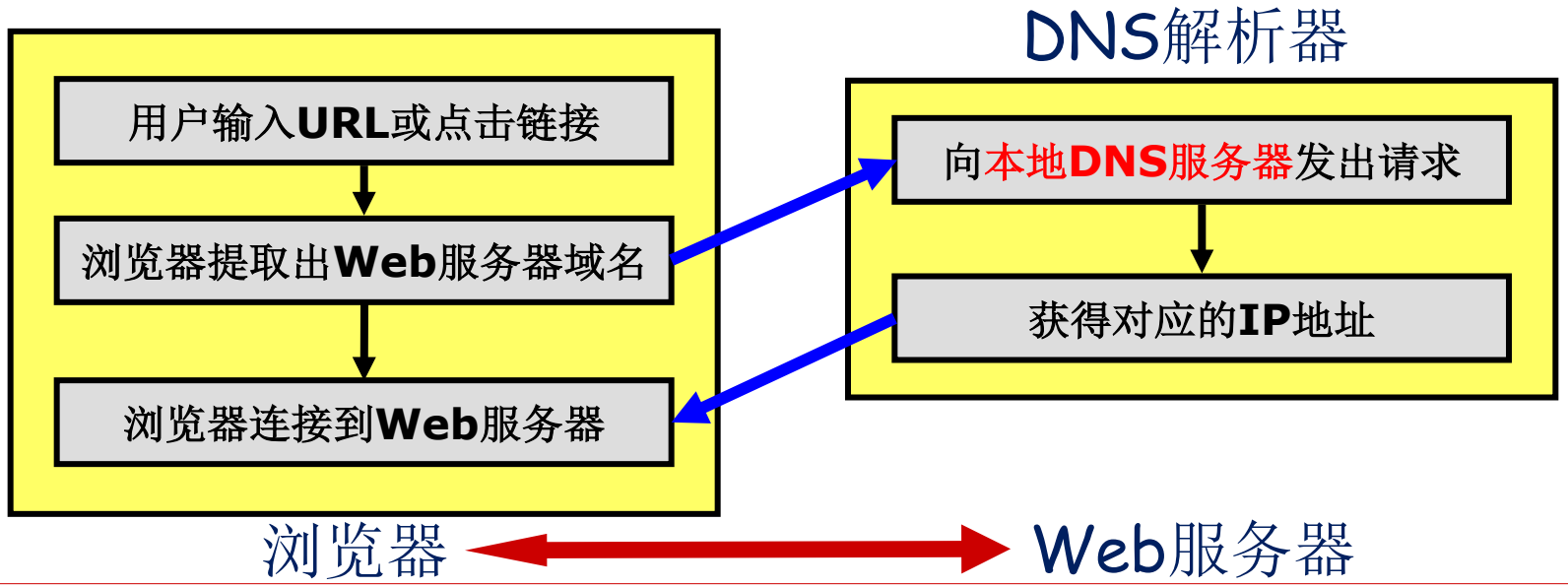
  - MAC地址：48位，用来在局域网中识别主机/路由器

DNS(域名系统)：基本功能

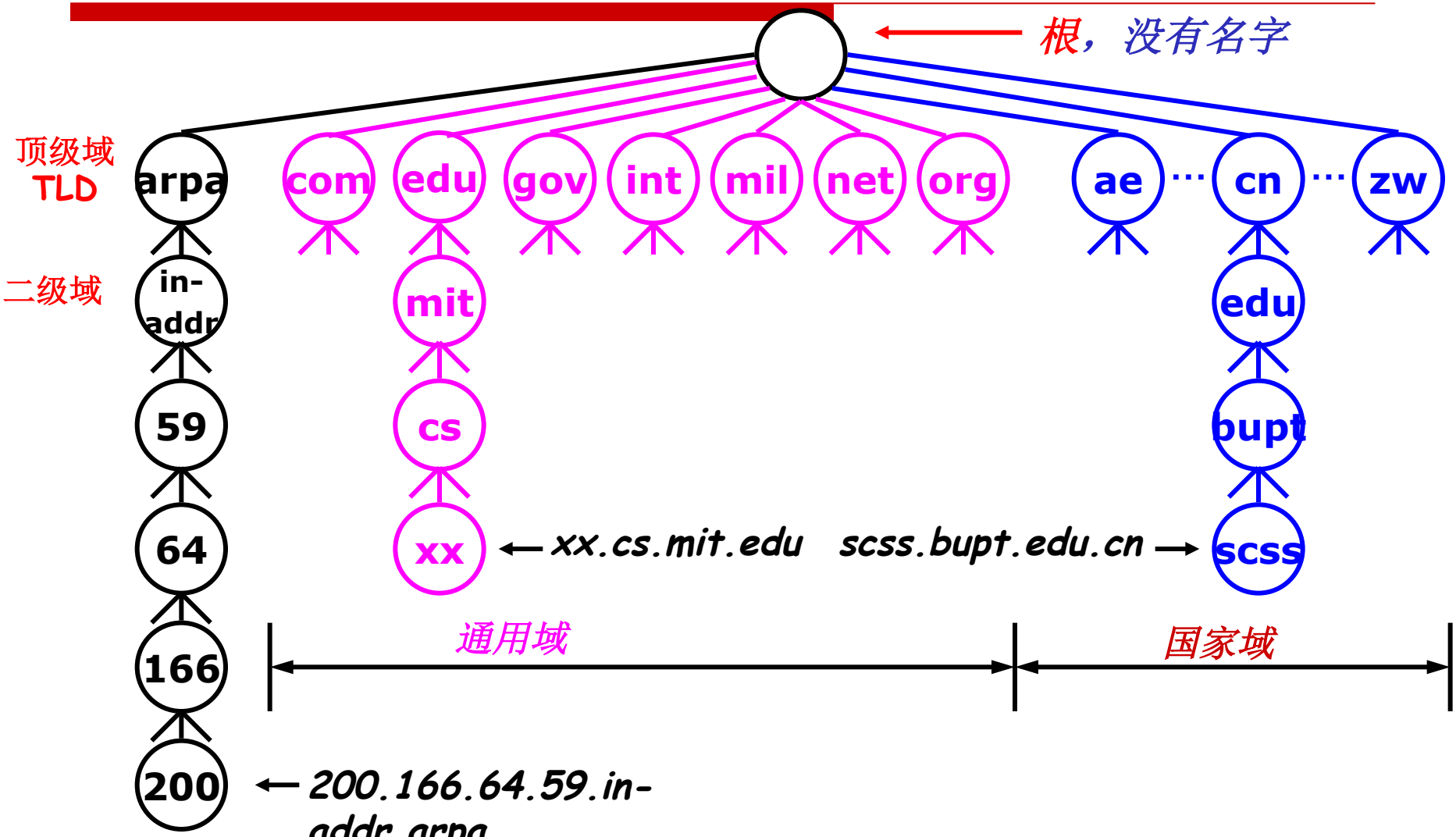
- IP地址的优点和缺点
  - IP地址更适合计算机处理，包含足够的路由信息。
  - IP地址不适合人们记忆
  - 无法通过IP地址猜测主机的用途
    - 如一个主机到底是www服务器还是FTP服务器？
- 如何取长补短——域名系统
  - 优点：
    - 使用方便，易于记忆。
    - 一致性好，不会随IP地址的改变而改变。
  - 采用分层结构的分布式数据库，提供主机名和IP地址映射的目录服务
  - 允许主机查询分布式DNS数据库的应用层协议

DNS协议要点

- 采用C/S模型，Client请求，Server响应
- 客户端程序：resolver(解析器)
- 传输层采用UDP
- DNS服务器的访问端口号是53
- DNS是其他应用层协议的支撑协议



层次化的名字空间

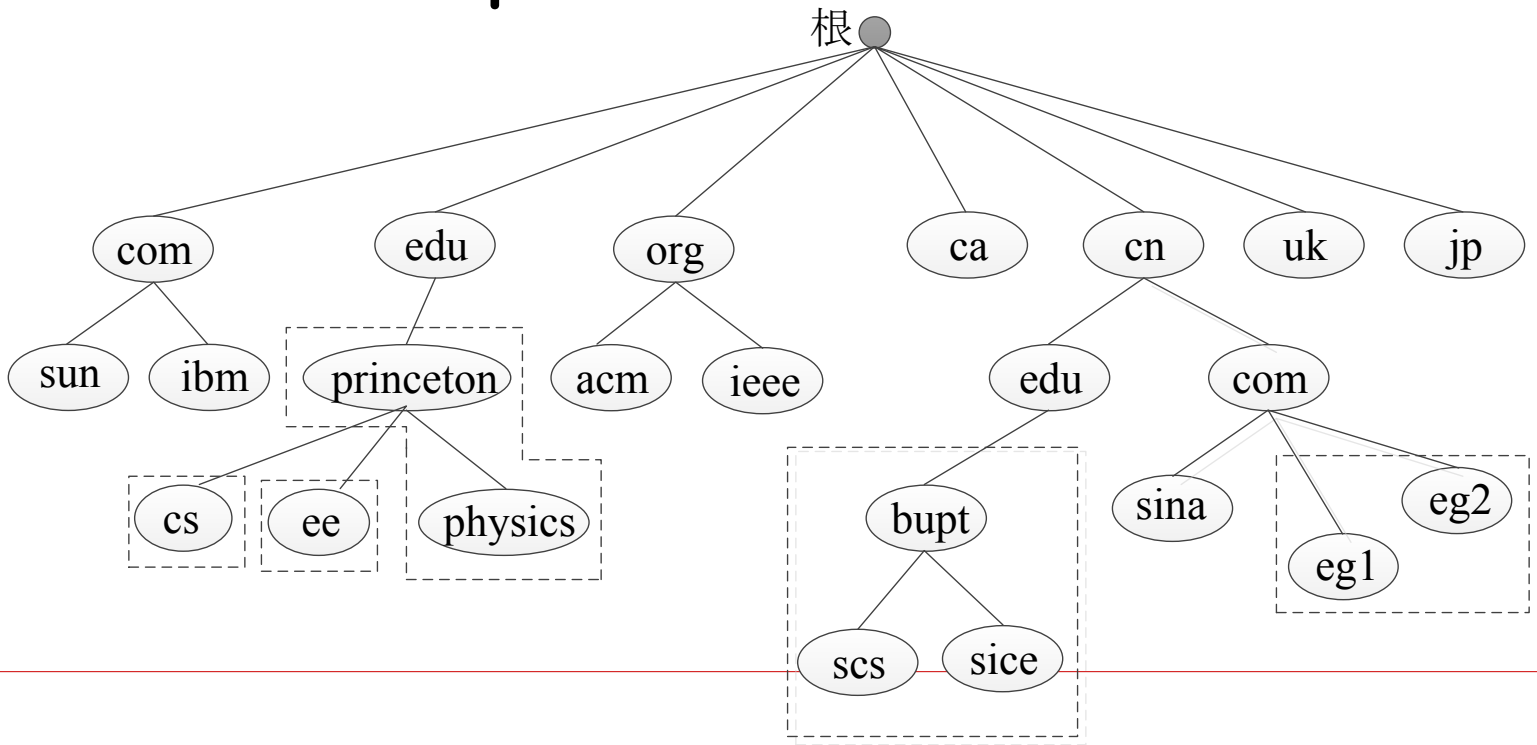

根，没有名字
顶级域
TLD
arpa
com
edu
gov
int
mil
net
org
ae
cn
zw
二级域
in-addr
mit
edu
59
cs
bupt
64
xx
xx.cs.mit.edu
scss.bupt.edu.cn
scss
166
通用域
国家域
200
200.166.64.59.in-addr.arpa

主机域名的构成

- 主机名是由一系列由“.”分开的标签组成：
  - 标签由英文字母和数字组成；
  - 每个标签不能超过63个字符；
  - 全部的标签不能超过255个字符；
  - 书写顺序是从主机开始直到域名树的根域为止。例如：scss.bupt.edu.cn
- 域名是一个组织在域名空间中的名字
  - 例如，北京邮电大学的域名为bupt.edu.cn。
  - 该组织中已注册的主机都以组织的域名为后缀。

区域

- 是域名空间中的一部分（子树），从域名管理的角度来划分的，即DNS服务器的管辖范围是“区”
  - 每一个区设置相应的权威服务器，用来保存该区中所有主机的域名到ip地址的映射



DNS服务器

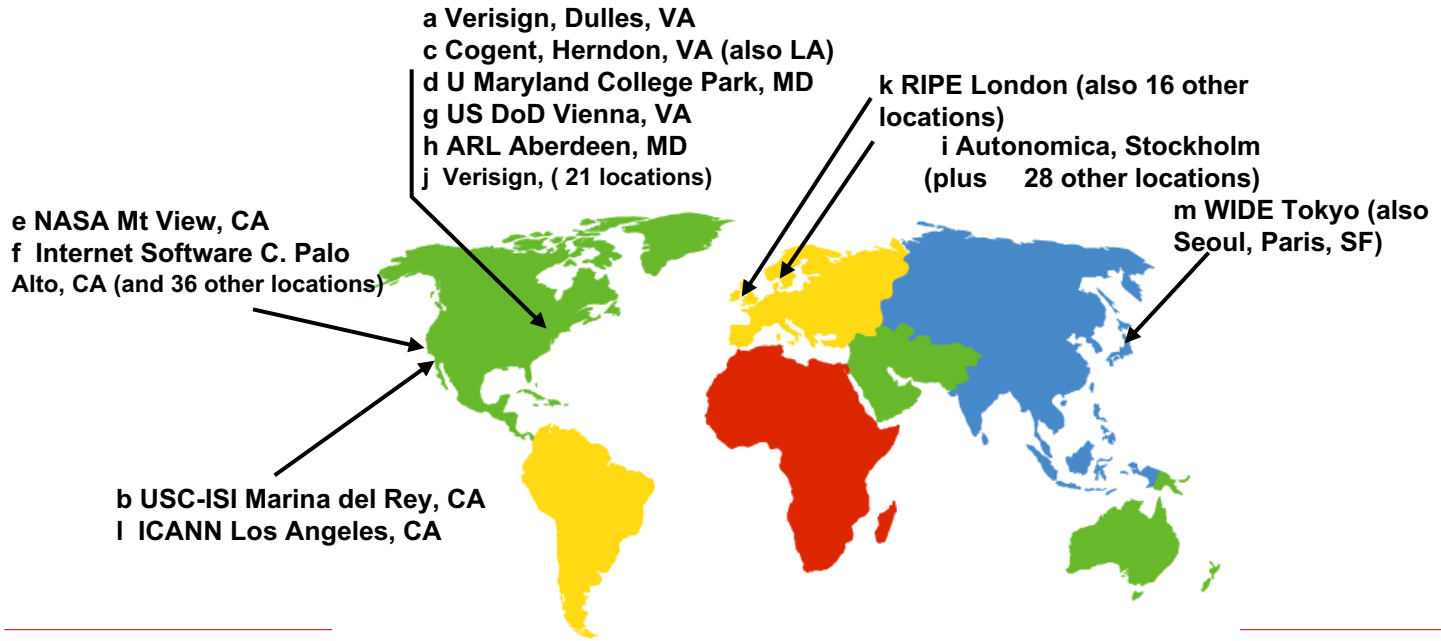
- 根据DNS服务器的作用，可分为四类：
  - **根域名服务器**（Root Name Server）
    - 为下级域名服务器提供域名解析服务；
      - 它需要知道全部顶级域名服务器的地址。
    - 数量很少，由于历史原因，主要分布在北美地区。
  - **顶级域名服务器**（Top level Name Server）
    - 负责管理该顶级域名服务器注册的所有二级域名。
  - **权威域名服务器**（Authoritative Name Server）
    - 每台因特网中的主机都应该在所在域的域名服务器中注册，提供注册的域名服务器就是该主机的认证域名服务器。
  - **本地域名服务器**（Local Name Server）
    - 当一个主机发出 DNS 查询请求时，这个查询请求报文就发送给本地域名服务器。

根域名服务器

名字解析起始：

解析器请求本地名字服务器

- 全球共有13个(a-m)
- 当用户无法在本地完成名字解析时，即访问根名字服务器
- 根名字服务器知道所有顶级名字服务器的IP地址
  - 每个名字服务器均知道其下级名字服务器的地址
  - 最终能访问到权威名字服务器，完成名字解析

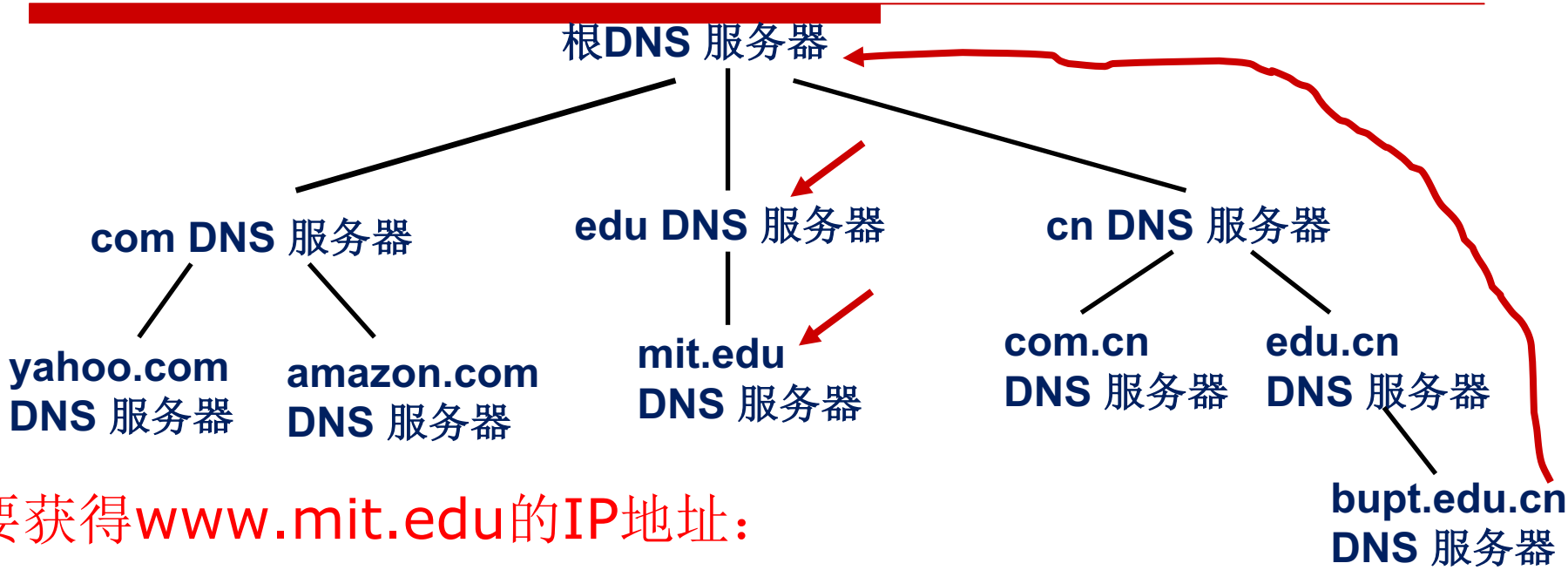


[Kurose]

权威名字服务器和本地名字服务器

- 顶级域名服务器
  - 负责管理之下注册的所有二级域名
- 权威名字服务器：
  - 企业网/校园网的DNS 服务器，提供主机名(如Web服务器、Email 服务器的域名)与IP地址的权威映射
  - 由企业网/校园网自己维护或ISP维护
- 本地名字服务器：
  - 用户resolver将DNS请求发送给本地DNS 服务器
  - 如果本地DNS 服务器找不到对应的映射关系，则将代理resolver去请求其他的名字服务器
  - 本地DNS 服务器可能是企业网/校园网的DNS 服务器或ISP 的DNS 服务器

DNS解析



要获得www.mit.edu的IP地址：

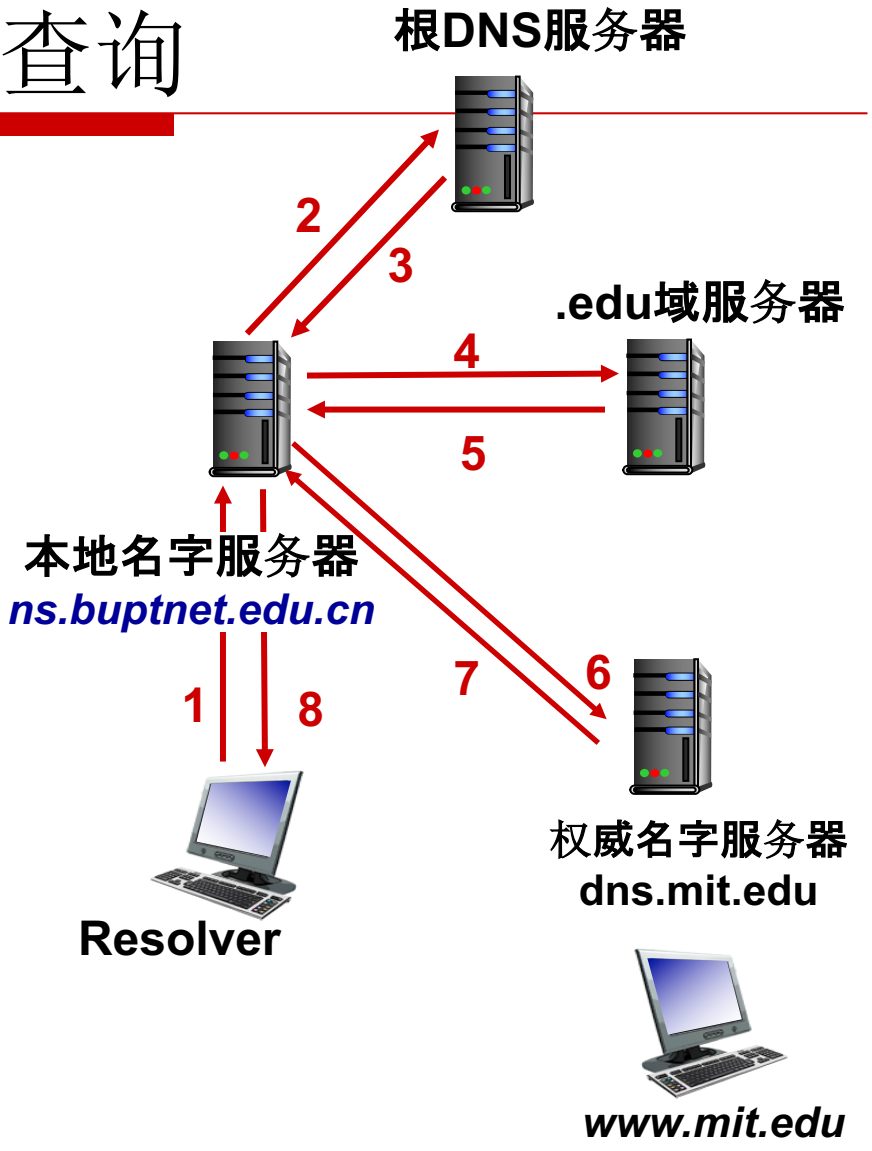
- 首先请求某个根服务器以找到.edu域的DNS 服务器
- 然后请求.edu域的DNS 服务器(TLD名字服务器)以找到.mit.edu域的DNS 服务器
- 请求mit.edu的DNS 服务器(权威服务器)以获得www.mit.edu的IP地址

DNS名字解析：迭代查询

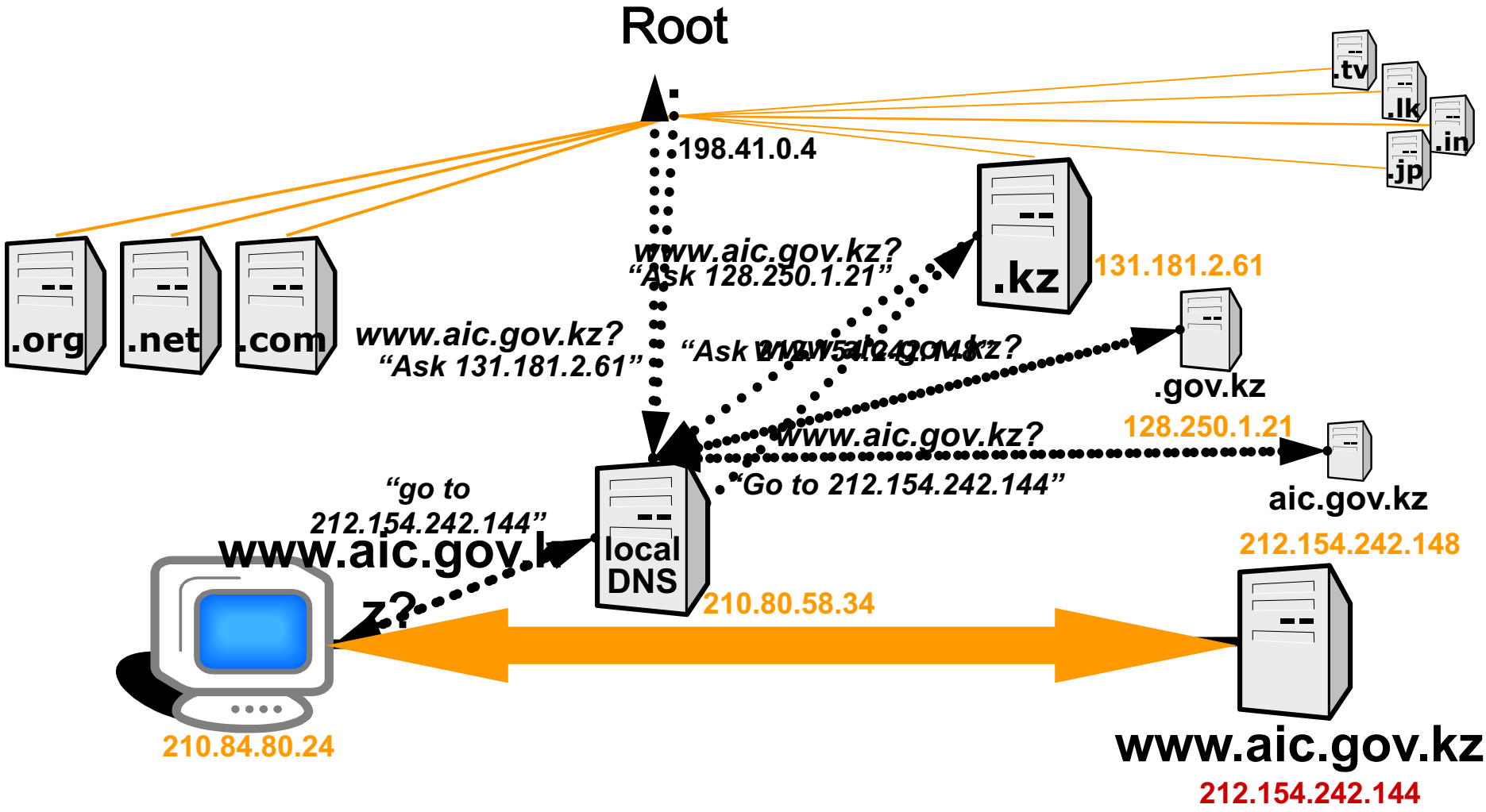
- 目前因特网采用的方式
- 示例：北邮的主机想查找www.mit.edu的IP地址

迭代查询过程：

- 在无法完成域名解析时，本地DNS服务器询问根DNS 服务器
- 根DNS 服务器返回.edu域DNS 服务器的地址
- 本地DNS 服务器询问.edu域DNS 服务器
- 以此类推，最终请求权威DNS服务器



DNS名字解析：迭代查询示例(2)

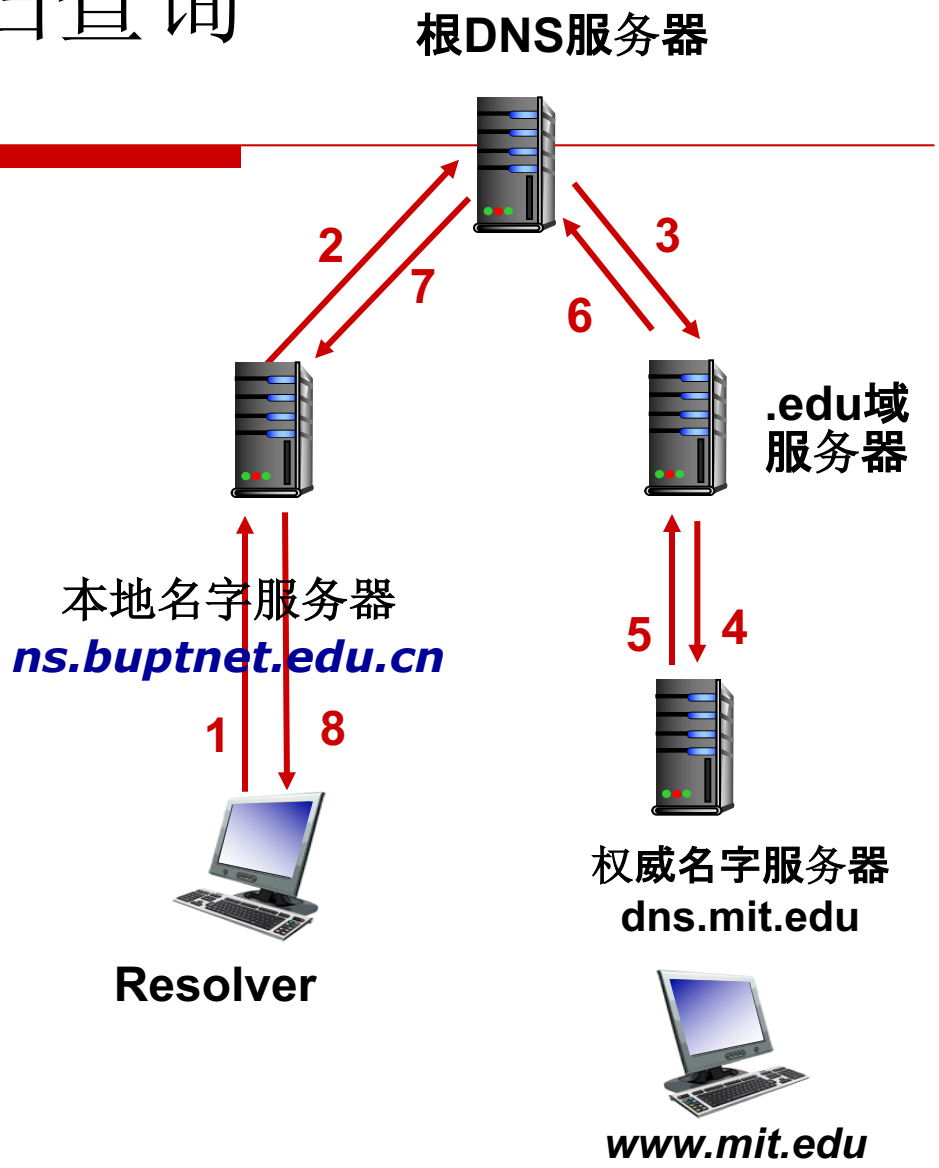


DNS名字解析：递归查询

- 示例：北邮的主机想查找 **www.mit.edu的IP地址**

递归查询

- 由被请求的名字服务器代理进行DNS查询
- 因特网中可选采用
- Q：有什么缺点吗？



[Kurose]

DNS性能改进措施

- 主、备用服务器
  - 定期把数据复制到备用服务器中，主服务器发生故障时，备用服务器代替工作
- 当DNS服务器收到一个新的DNS响应，会将域名和地址的映射关系缓存在本地
  - 提高后续查找效率
  - 缓存的映射数据在一段时间后过期
  - 本地DNS服务器通常可以直接访问TLD 服务器，而不必访问根名字服务器
- 映射数据更新/通知机制
  - RFC 2136
  - https://datatracker.ietf.org/doc/rfc2136/

DNS资源记录（RR）（1）

资源记录保存在DNS服务器中

RR 格式：(name, TTL, class, type, value)

Time-to-live,
有效期

IN，
因特网

- Type=A——地址资源记录
  - **Name= 主机的域名，Value= IP地址**

ns.buptnet.edu.cn 86400 IN A 202.112.10.37

- Type=NS——权威DNS服务器资源记录
  - **Name= 域** (如 bupt.edu.cn)
  - **value=** 此域的权威DNS 服务器的域名

bupt.edu.cn 86400 IN NS ns.buptnet.edu.cn

DNS资源记录（RR）(2)

RR 格式: (name, TTL,, class, type, value)

- Type=CNAME——标准名称资源记录
  - **Name= 域名(别名)**
  - **Value= 规范名**

  www.bupt.edu.cn 86400 IN CNAME vn46.bupt.edu.cn

- Type=MX——邮件服务器资源记录
  - **Name= 域**
  - **Value= 此域的邮件服务器的规范名字**

  bupt.edu.cn 86400 IN MX mxbiz1.qq.com

DNS资源记录（RR）(3)

RR 格式: (name, TTL, class, type, value)

Owner name	TTL	Class	RRType	RData
bupt.edu.cn	86400	IN	NS	ns.buptnet.edu.cn
ns.buptnet.edu.cn	86400	IN	A	202.112.10.37
www.bupt.edu.cn	86400	IN	CNAME	vn46.bupt.edu.cn
bupt.edu.cn	86400	IN	MX	mxbiz1.qq.com
www.bupt.edu.cn	86400	IN	A	10.3.9.161

查询资源记录的命令：**nslookup**

C>nslookup –query=MX bupt.edu.cn

nslookup示例

```
C:\Users\Administrator>nslookup -query=MX bupt.edu.cn
服务器:  UnKnown
Address:  10.3.9.4

非权威应答:
bupt.edu.cn     MX preference = 5, mail exchanger = mxbiz1.qq.com
bupt.edu.cn     MX preference = 10, mail exchanger = mxbiz2.qq.com

C:\Users\Administrator>nslookup www.bupt.edu.cn
服务器:  UnKnown
Address:  10.3.9.4

DNS request timed out.
    timeout was 2 seconds.
DNS request timed out.
    timeout was 2 seconds.
非权威应答:
名称:    vn46.bupt.edu.cn
Addresses:  2001:da8:215:4038::161
          10.3.9.161
Aliases:  www.bupt.edu.cn


C:\Users\Administrator>
```

DNS协议消息

DNS协议包含**请求(query)**和**应答(reply)**两个消息，其消息格式相同

消息头(12字节)

- 标识（ID）：16位，一对请求和应答消息使用同样的标识
- 标志（Flag）：
  - 消息是*请求*或*应答*
  - 是否使用递归查询
  - 递归查询是否可行
  - 应答是否是权威的

2字节	2字节
标识	标志
问题个数	回答个数
权威回答个数	附加回答个数
问题段（由Resolver添加在请求消息中）	
回答段（由服务器添加在应答消息中）	
权威回答段（由服务器添加在应答消息中）	
附加回答段（由服务器添加在应答消息中）	

在DNS数据库中增加资源记录

- 示例：一个新创建的公司“Network Utopia”
- 需要向DNS注册机构（如CNNIC，中国互联网络信息中心）注册域名networkuptopia.com.cn
  - 应提供域名、公司的权威DNS服务器的IP地址
  - 注册机构将在.com.cn名字服务器中增加两条资源记录：

```
(networkutopia.com.cn, dns1.networkutopia.com.cn, NS)
(dns1.networkutopia.com.cn, 212.212.212.1, A)
```

- 同时需要在该公司的权威DNS服务器中为www.networkuptopia.com.cn创建类型为A的资源记录；为邮件服务器mail.networkuptopia.com.cn创建类型为MX的资源记录和类型为A的资源记录

DNS的安全性问题

- 名字服务器易受攻击，名字数据库被非法主机修改
  - DNS伪冒(spoofing)
  - DNS缓存毒化(Cache poisoning)
  - DOS、缓存溢出(Buffer overrun)、重放(Replay)攻击…
- 改进策略：DNSSEC
  - RFC 3090
  - 数字签名、公共密钥
    - 对DNS请求/应答消息进行身份认证

DNS概要

- 层次化、分布式的域名系统
- 实现名字解析（域名←→IP地址）的协议，因特网各应用的支撑协议
  - 采用C/S模型
    - Client：Resolver
    - Server：名字服务器
  - 一般使用UDP，服务器端口号为53
  - 请求/应答消息
  - 迭代查询vs 递归查询
- 资源记录（RR）

主要内容

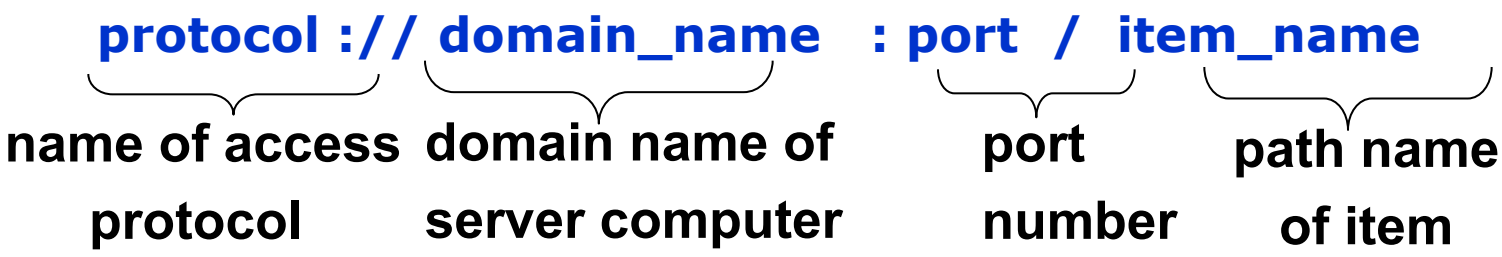
- 2.1 网络应用概述
- 2.2 DNS
- *2.3WWW应用和HTTP*
  - WWW的体系结构
  - URL
  - 网页类型
  - HTTP：操作过程、消息格式
  - 持久连接与非持久连接
  - Cookies和Proxy
- 2.4 Email应用
- 2.5 FTP
- 2.6 远程登录协议：Telnet
- 2.7 应用层安全隐患

WWW概述（1）

- 万维网，WWW, World Wide Web
- 不是一个网络，而是基于因特网的信息服务系统
  - “universe of network-accessible information, an embodiment of human knowledge” *(Tim Berners-Lee)*
- 分布式超媒体系统(Hyper-media)
  - 超媒体包括文本、图形、图像、音频、视频等多种媒体
- 以C/S模式工作
  - 客户程序向服务器程序提出请求
  - 服务器程序向客户程序返回客户需要的万维网文档，即页面

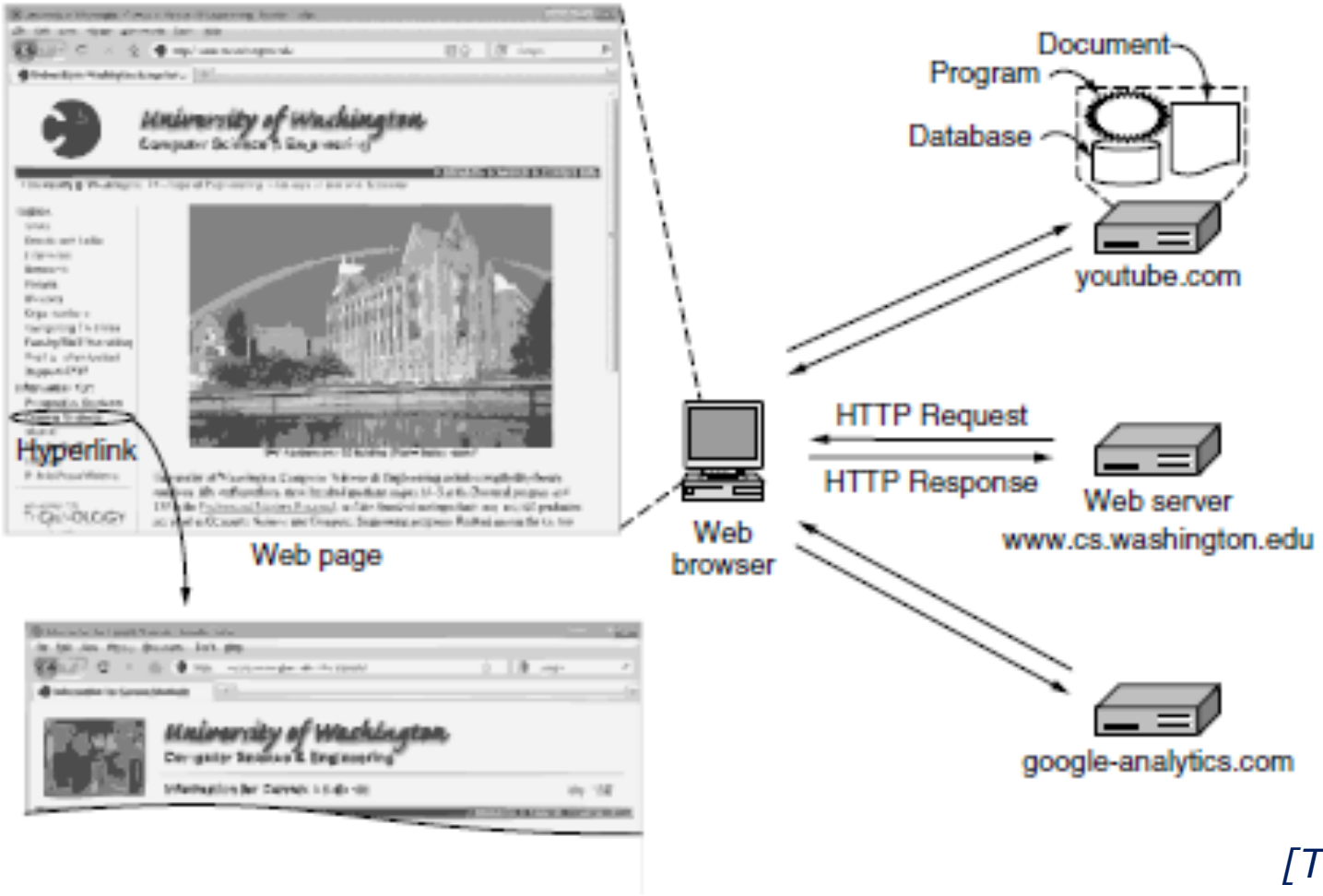
WWW概述（2）

- 如何标识分布在整个因特网上的页面
  - URL(Universal Resource Locater)唯一标识



- 用什么协议实现通过WWW上的各种链接获取信息
  - HTTP(Hyper-Text Transfer Protocol)
- 如何使不同风格的页面在不同主机上显示，并标识超链
  - HTML(Hyper-Text Markup Language)
- 如何使用户能方便找到所需的信息
  - 搜索引擎

WWW的体系结构



[Tanenbaum]

WWW应用的客户：浏览器

- WWW应用的用户接口
- 功能
  - 向Web服务器发送请求消息
  - 接收Web服务器的应答消息
  - 解释网页文档的源代码，将网页呈现给用户
  - 通用的客户端，支持Email、文件传输、BBS、电子商务等应用
- 不同的浏览器对于同一网页文件的呈现结果不同

WWW应用的服务器：Web服务器

- 保存Web网页文档
- 接收客户端的请求消息
- 返回响应消息：状态信息和网页数据
- 可选功能
  - 保存访问用户信息
  - 对于访问用户进行身份认证
  - 对于网页数据进行权限管理
  - ……

WWW应用的编址：URL

- 统一资源定位符，唯一标识一个资源，对资源进行定位和访问

  协议名 :// 主机：端口号 /文件路径及文件名

  例如：http://www.abc.com:8080/example/example.html

- 和网页上的链接相关联
- 缺省值：
  - 协议名：http
  - 端口号：80
  - 文件名：index.html

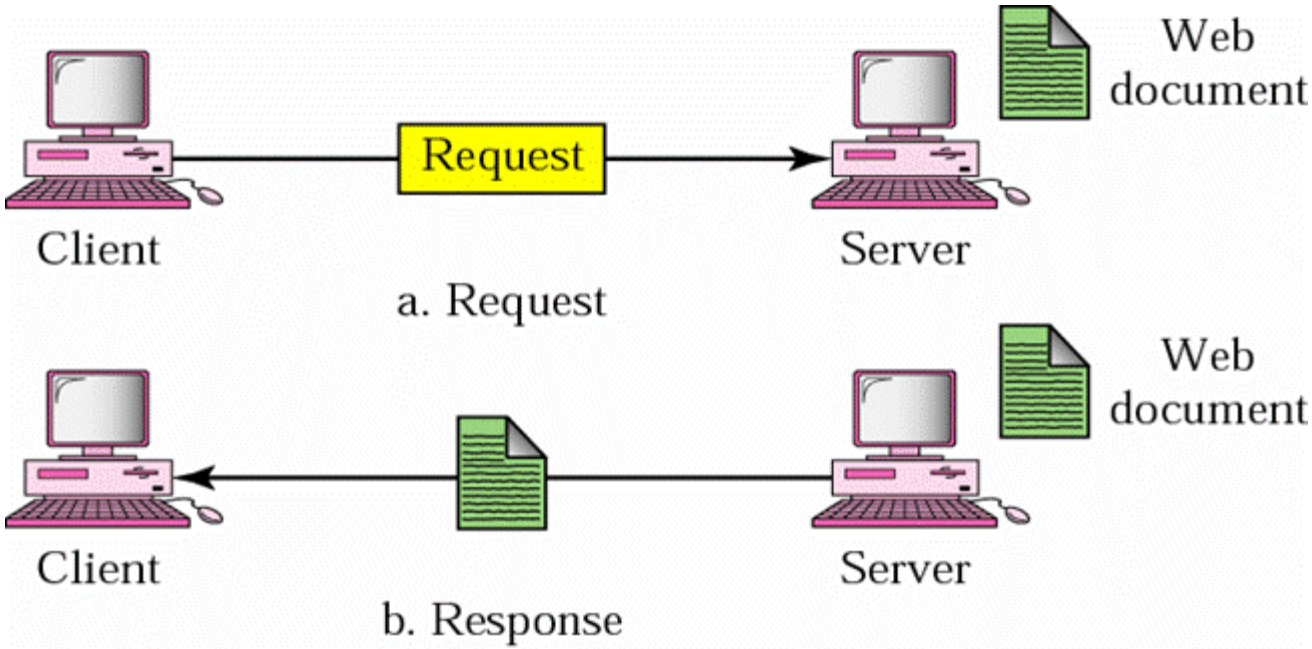
  例如：www.abc.com 等同于

  http://www.abc.com:80/index.html

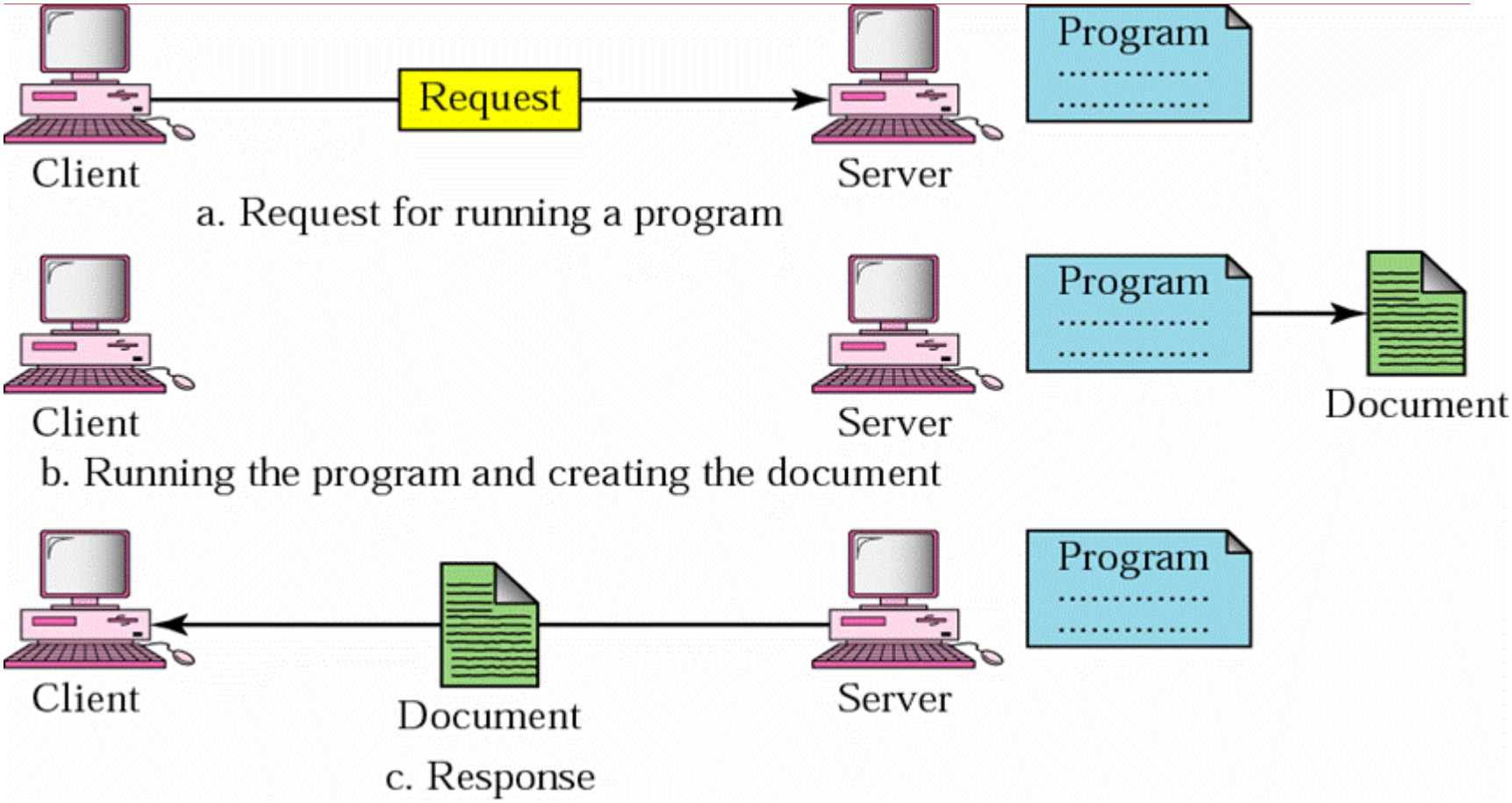
网页类型

- 静态网页（Static web page）
  - 以文件形式保存
  - 不同的用户、不同时间访问，返回结果均一样
- 动态网页（Dynamic web page）
  - 收到浏览器请求后，服务器端动态生成网页
  - 示例：CGI脚本
- 活跃网页（Active web page）
  - 收到服务器响应后，客户端动态生成网页
  - 示例：Java Applet

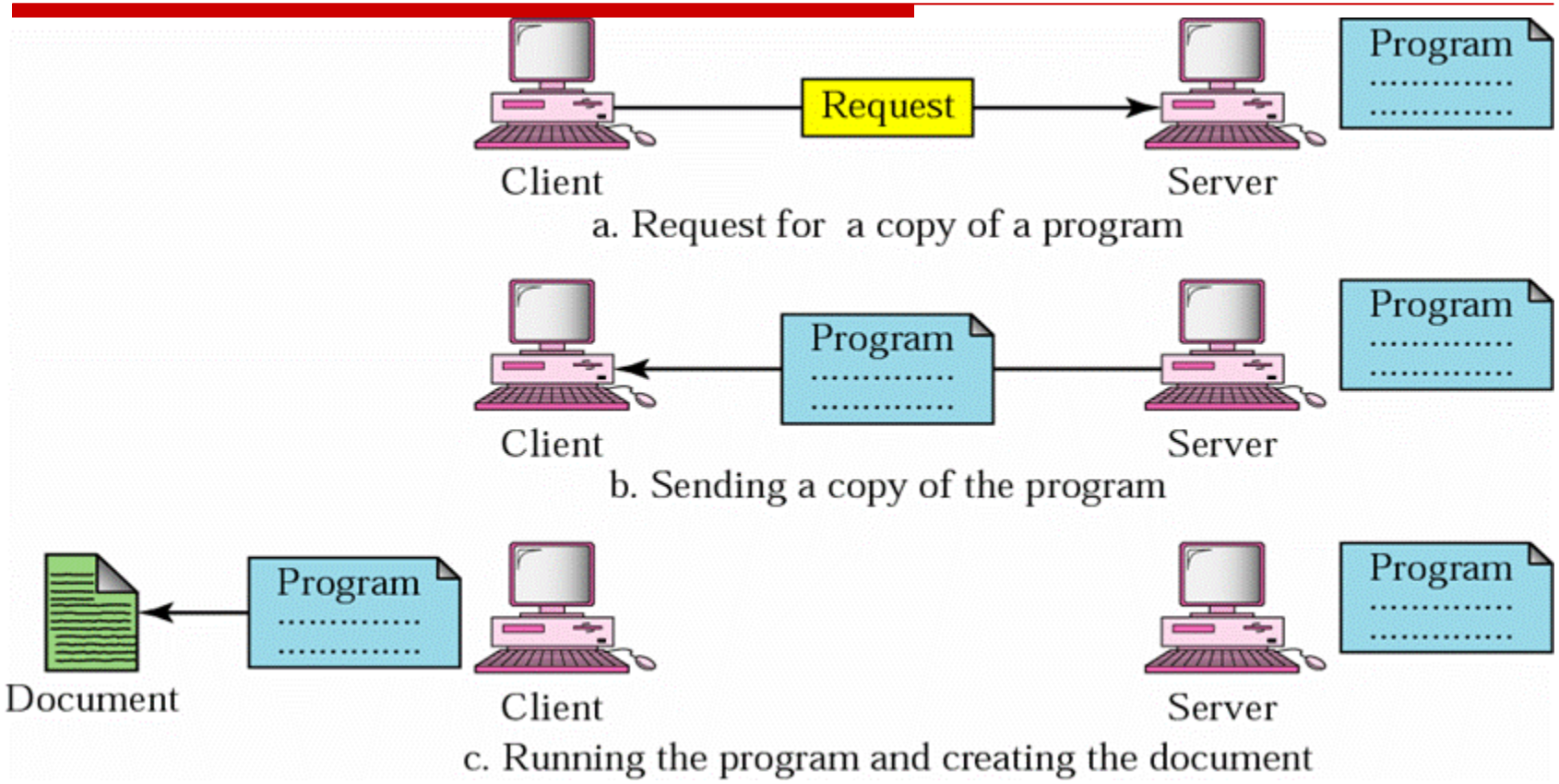
静态网页原理


Web
document
Request
Client
Server
a. Request
Web
document
Client
Server
b. Response

动态网页原理

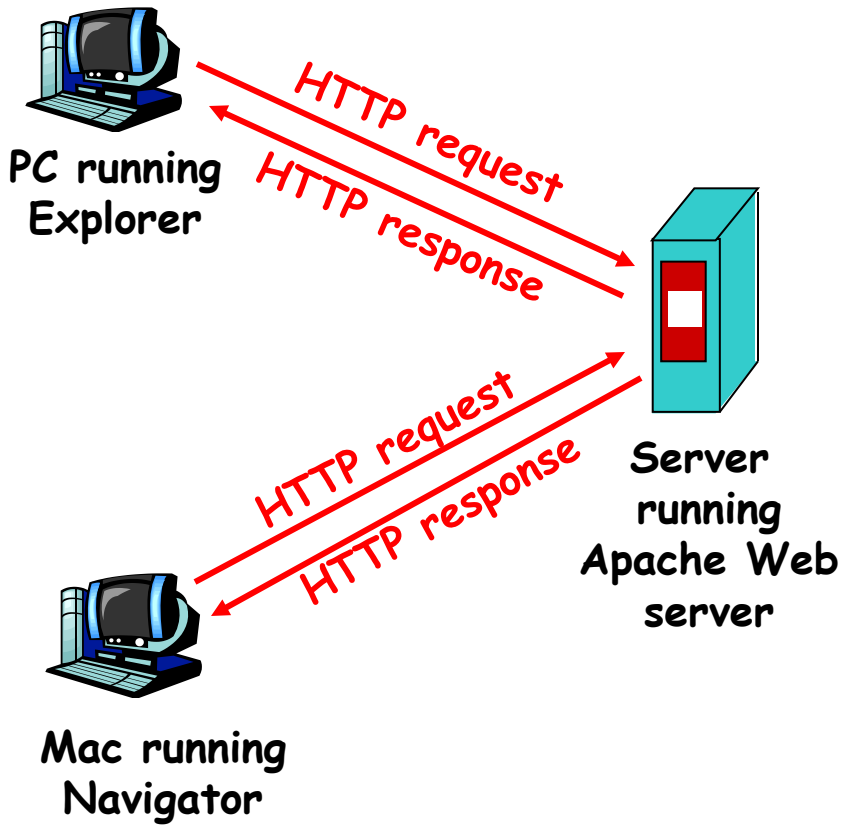

Request
Client
Server
Program
a. Request for running a program
Client
Server
Program
Document
b. Running the program and creating the document
Client
Document
Server
Program
c. Response

活跃网页原理

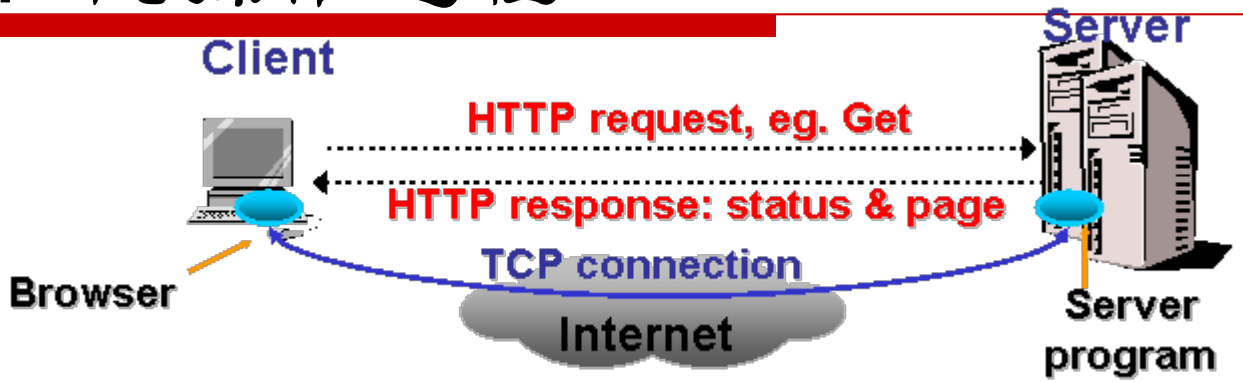

Request
Client
Server
Program
a. Request for a copy of a program
Program
Client
Server
Program
b. Sending a copy of the program
Document
Program
Client
Server
Program
c. Running the program and creating the document

HTTP概述 Hypertext Transfer Protocol

- WWW的应用层协议
- 采用C/S模型
  - *客户端浏览器：* 发出HTTP请求，接收HTTP响应，向用户呈现网页
  - *Web服务器:* 接收HTTP请求，返回HTTP响应（包含请求的网页数据）
- 传输层采用TCP
- 无状态（Stateless）协议
  - 服务器端不保存以前的访问记录，对于每个请求都是单独处理



HTTP 的操作过程



(1) 浏览器分析超链接，如 http://www.abc.com/example.html

(2) 浏览器使用通过DNS获得服务器(*www.abc.com*)的IP地址

(3) 浏览器建立到服务器的TCP连接

(4) 浏览器发送HTTP 请求： GET /example.html HTTP/1.0

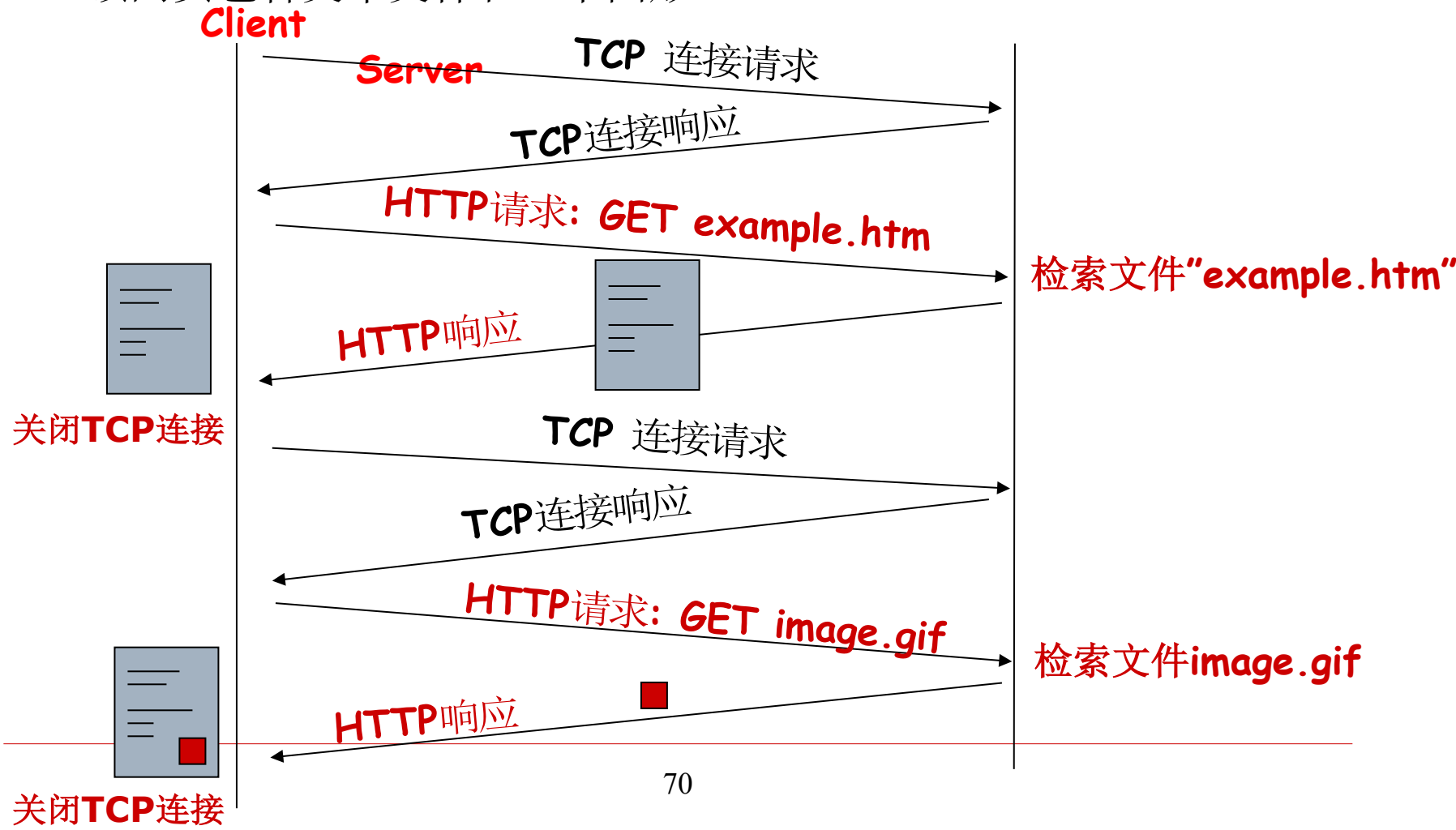
(5) 服务器发送HTTP响应

(6) 释放TCP连接

(7) 浏览器显示网页example.html

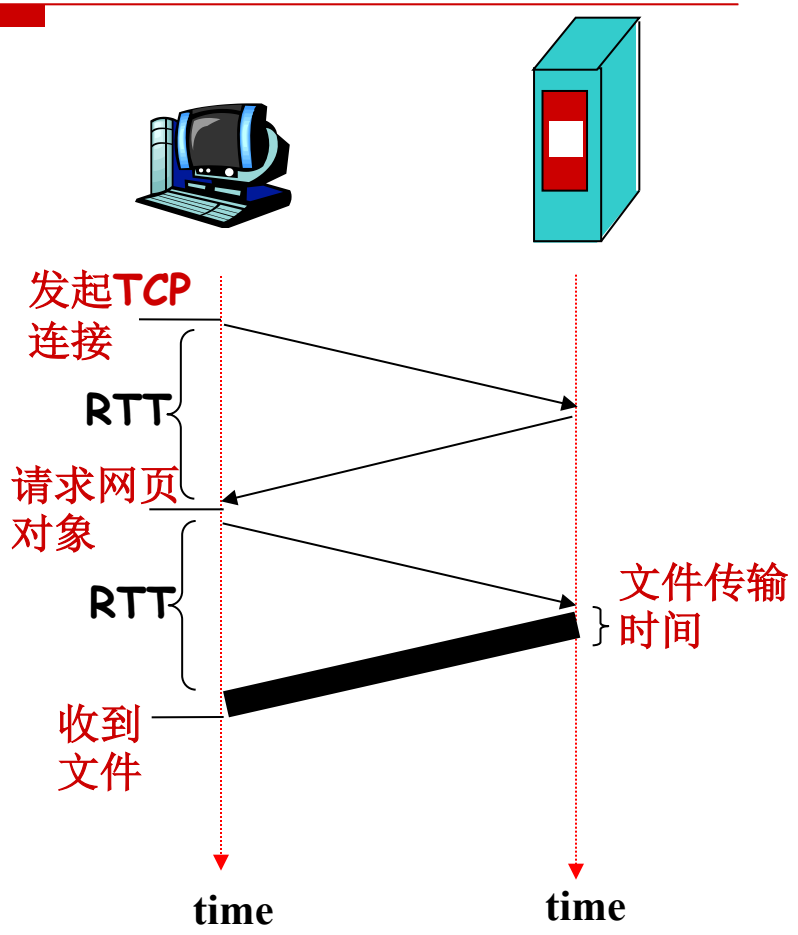
非持久连接

- 假定用户在浏览器地址栏输入：www.abc.com/example.htm
- 该网页包含文本文件和一个图形



非持久连接：响应时间

RTT: Round Trip Time, 往返时间，time for a small packet to travel from client to server and back.(不包含发送时延)



响应时间：

- 1 ×RTT：建立TCP连接
- 1 ×RTT： 发送HTTP请求、返回HTTP响应
- 网页对象文件传输时间

响应时间 = 2 ×RTT+文件传输时间

HTTP连接

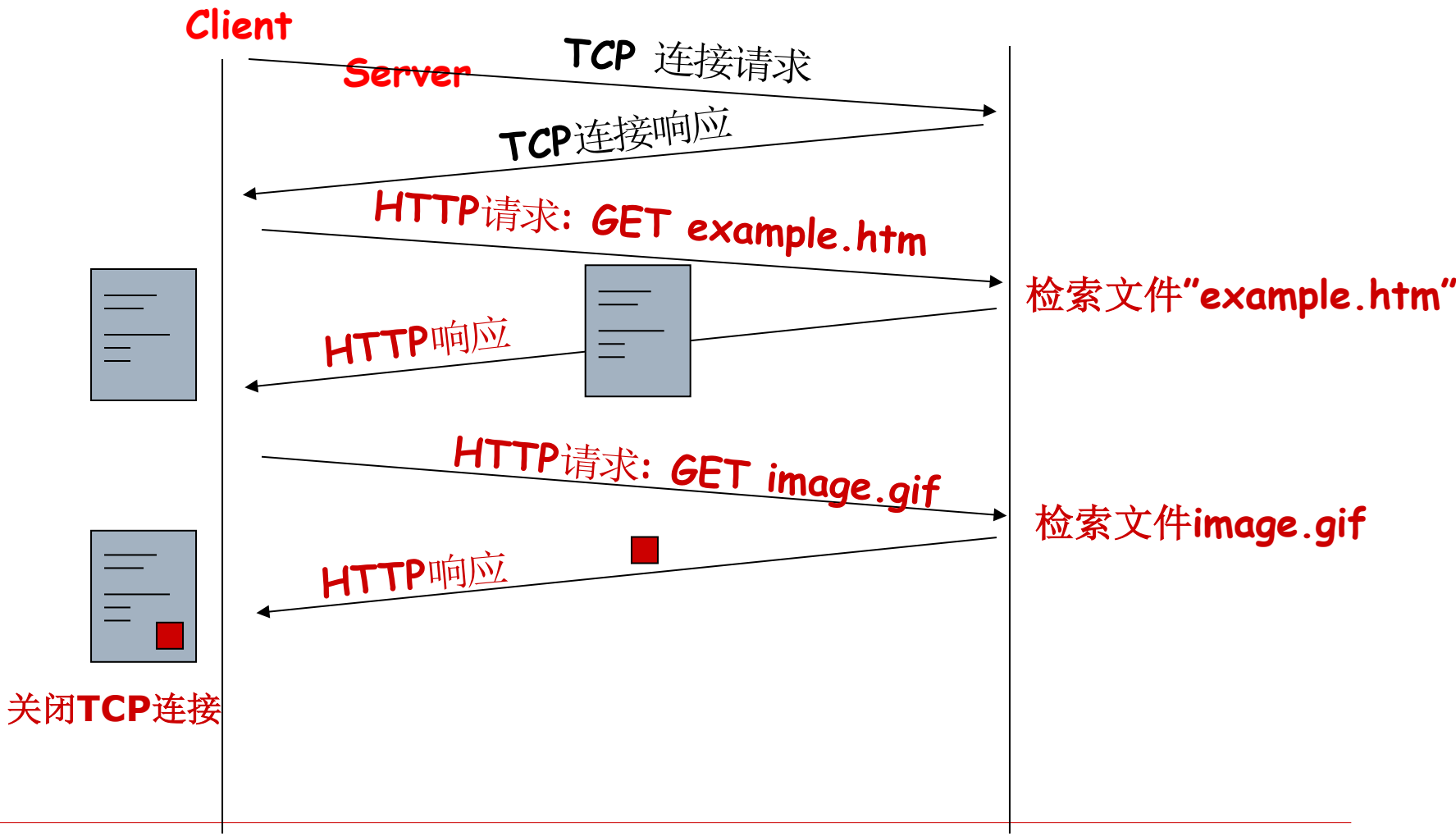
Nonpersistent HTTP (非持久连接)

- 一个连接只传输一个对象
- Server发送响应后，即关闭连接
- 优点：简单
- 缺点：
  - 每个对象都会有2× RTT
  - 建立TCP连接需要OS开销
  - 用户经常需要同时建立多个TCP连接
- HTTP 1.0

Persistent HTTP (持久连接)

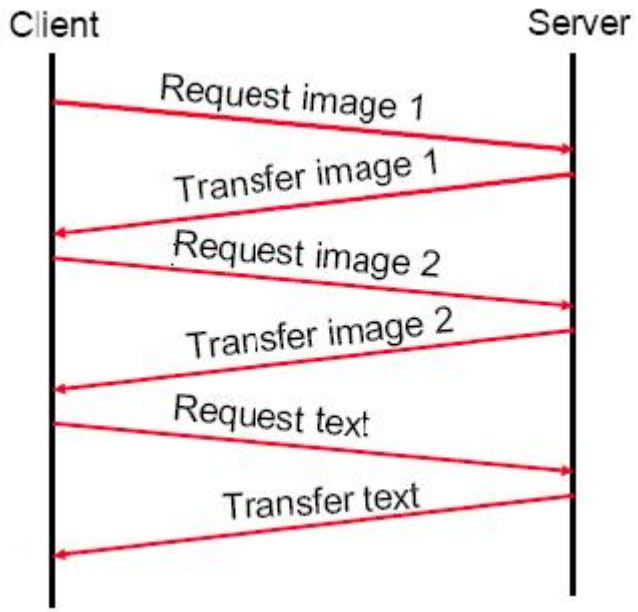
- 一个TCP连接可以传输多个对象
- 优点：发送完响应后，Server不关闭连接，在一个TCP连接上可以传输多个对象
- HTTP 1.1

持久连接：示例

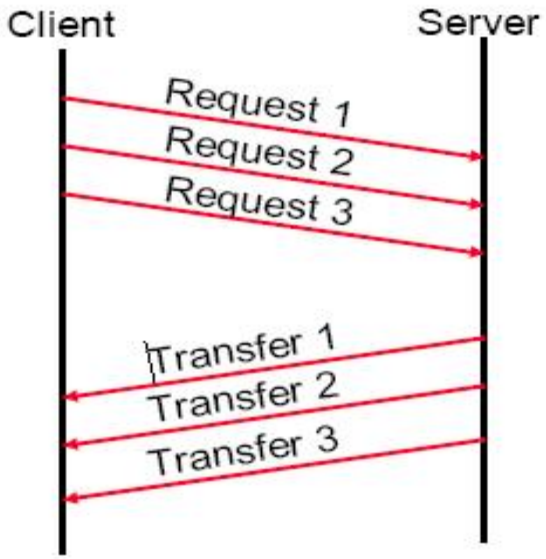

Client
Server
TCP 连接请求
TCP连接响应
HTTP请求: GET example.htm
检索文件"example.htm"
HTTP响应
HTTP请求: GET image.gif
检索文件image.gif
HTTP响应
关闭TCP连接

持久连接：流水线(pipelining)

- Client可以连续发出多个请求，而不必等待前一个请求的响应完成



非流水线机制



流水线机制

HTTP消息：请求

- HTTP有两类消息：请求（*request*）,响应（ *response* ）
- 消息格式：ASCII，（易读）
- HTTP请求消息

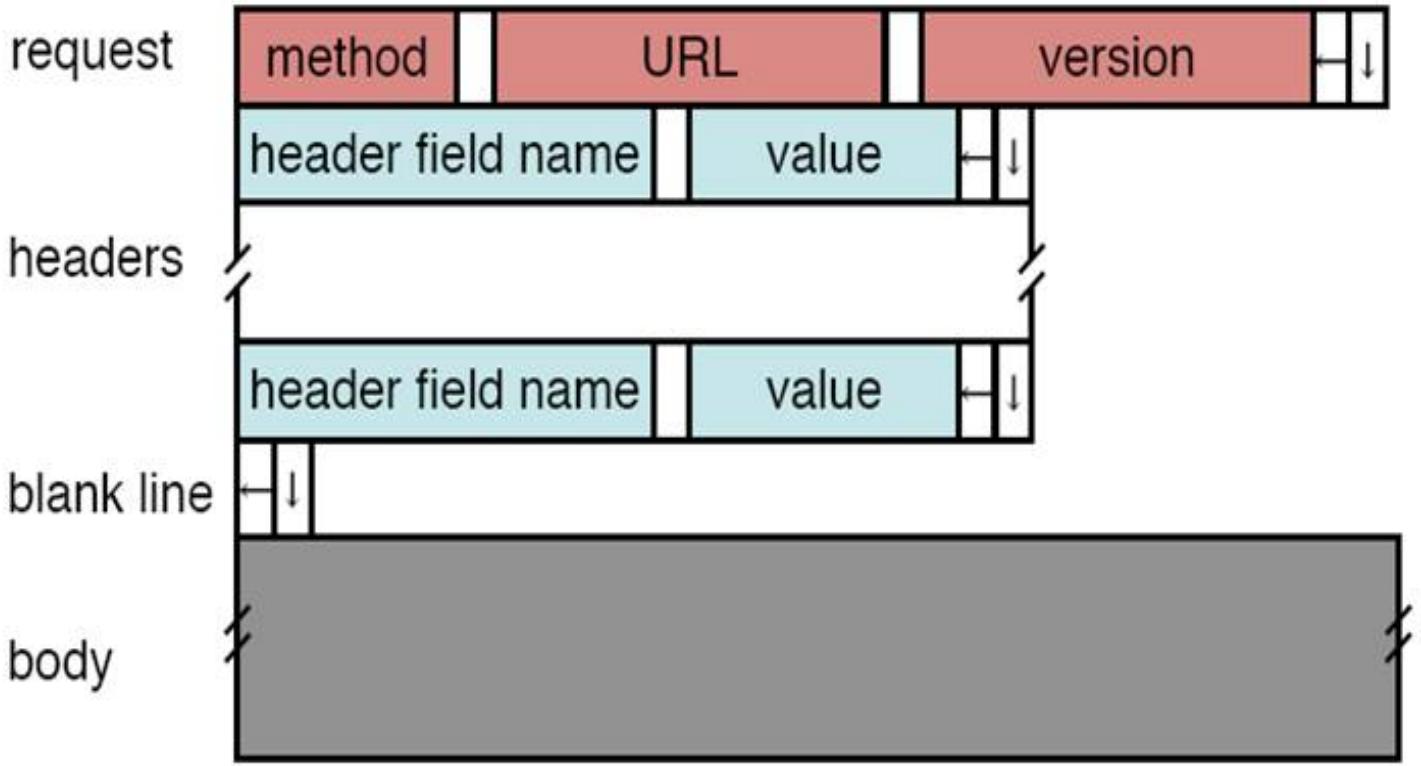
请求行 → `GET /somedir/page.html HTTP/1.1`

消息头（可选）

```
Host: www.abc.edu.cn
User-agent: Mozilla/4.0
Accept-language:zh-cn
Connection: Keep-Alive
```

一个空行（回车符+换行符），表示消息结束

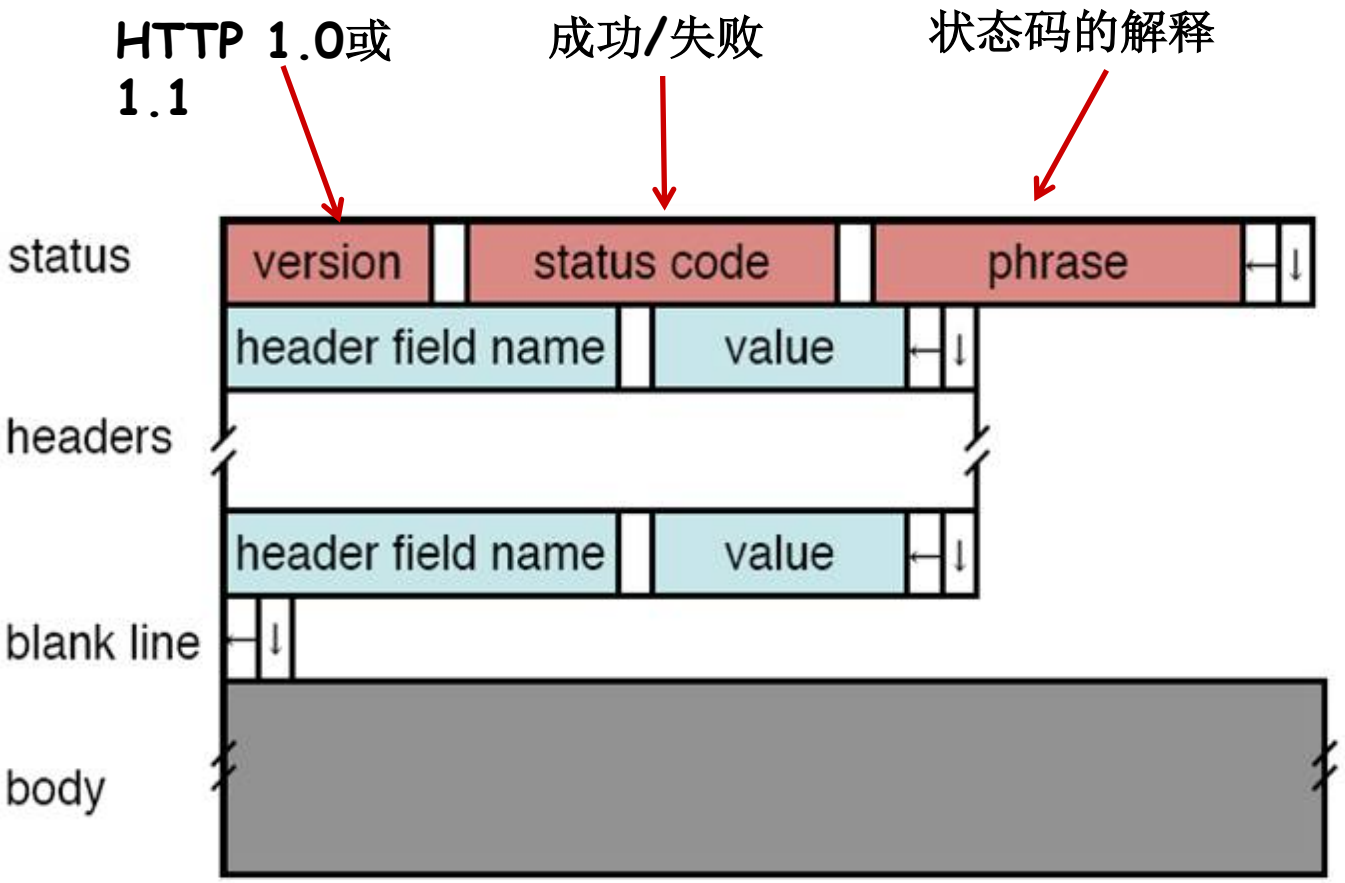
HTTP请求消息的格式



HTTP方法 (Methods)

HTTP方法	功能描述	支持的版本号
GET	从服务器下载URL对应的网页	HTTP1.0
PUT	将网页上传到URL指定的位置	HTTP1.1
POST	对URL指定的资源进行操作，如对文档进行注释、提交表格、在数据库中追加信息等	HTTP1.0
HEAD	向服务器请求URL对应的网页的消息头，不返回网页内容	HTTP1.0
DELETE	请求服务器删除URL对应的网页	HTTP1.1
TRACE	要求服务器将请求消息原样返回，用于错误诊断	HTTP1.1

HTTP 响应消息的格式



HTTP响应消息示例

状态行 → HTTP/1.1 200 OK

消息头（可选）
Date: Thu, 03 Mar 2011 06:28:24 GMT
Server: Microsoft-IIS/6.0
Content-Length: 1819
Content-Type: text/html;charset=gb2312
Connection: Keep-Alive

空行 →

data data data data data ...

网页数据 →

HTTP 响应：状态码（RFC2616）

状态码	简短解释	含义
200	OK	网页请求成功，消息体包含所请求的数据
204	No content	网页请求成功，但无需返回消息体，网页不用刷新，例如对用户提交信息的确认
301	Moved Permanently	请求内容已经移到另一个服务器，响应的消息头中将包含该服务器的域名
304	Not Modified	网页没有修改，用户可以继续使用缓存的网页，此响应中不包含消息体
403	Forbidden	用户没有权限访问请求的网页
404	Not found	没有找到请求网页，可能是由于用户提供的URL错误
500	Internal Server Error	服务器内部故障
503	Service Unavailable	服务器由于临时过载，不能响应用户的请求
505	HTTP Version Not Supported	服务器不支持请求消息中的HTTP版本

HTTP: cookie

- Web服务器对于访问用户的标识信息，可用于识别用户、记录用户信息和访问情况

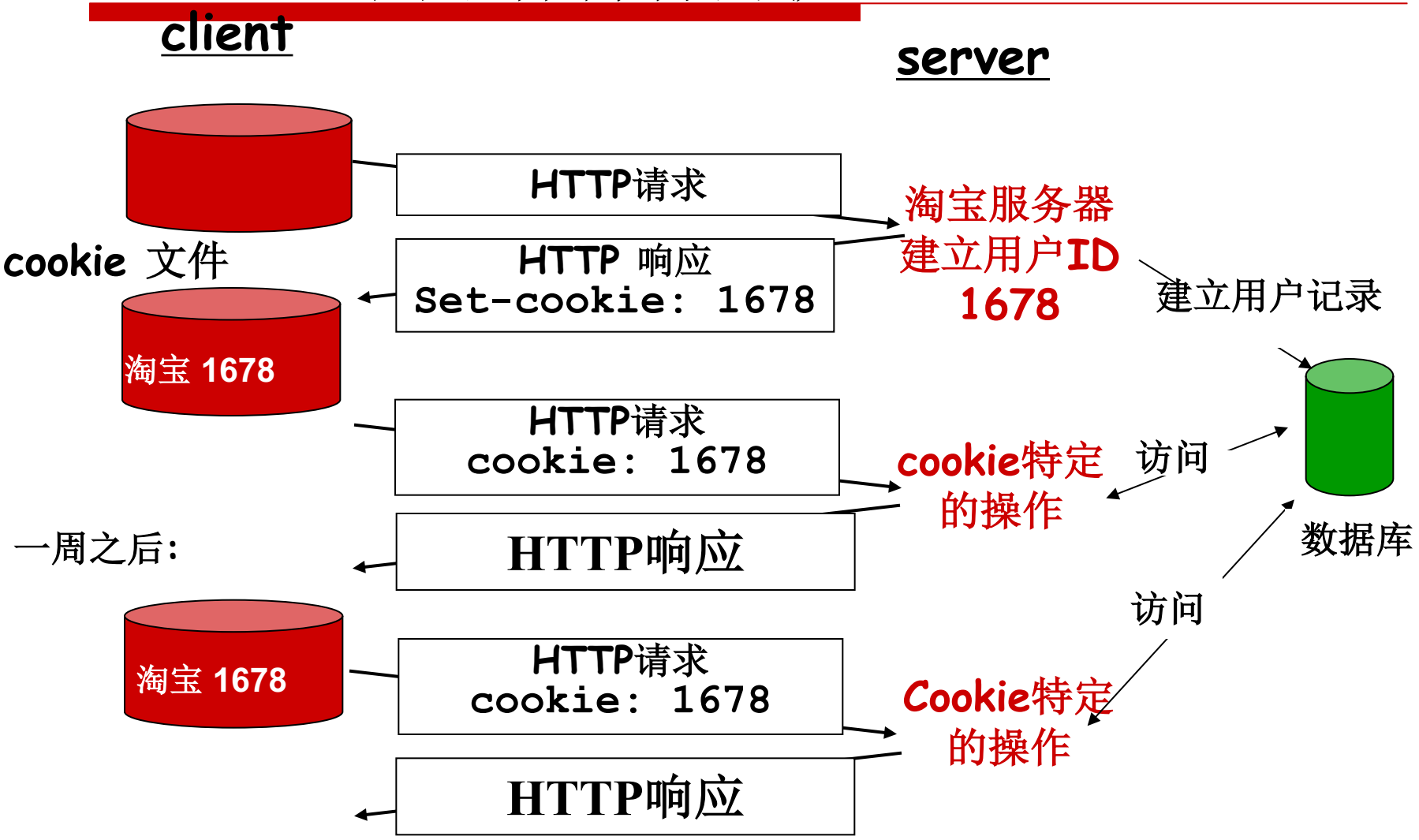
Cookie涉及下列内容：

1) 在HTTP响应中包含一行set-cookie头信息
2) 在HTTP请求中包含一行cookie头信息
3) Client主机中保存一个cookie文件，由浏览器管理
4) Server端的后台数据库

示例：

- 小王访问淘宝网站
- 服务器收到初次访问请求时，将增加小王的相关信息
  - 唯一的ID
  - 数据库的记录

Cookie: 用于保持访问状态



Cookie的特性

Cookie的应用：

- 身份认证
- 购物车
- 个性化的推荐信息
- 用户会话状态

  (Web e-mail)

禁用cookie？

- 可以在浏览器中限制cookie使用

Cookie 的隐私问题:

- **cookie** 让网站得到用户的很多个人信息
  - 姓名、联系方式、 住址、工作单位...

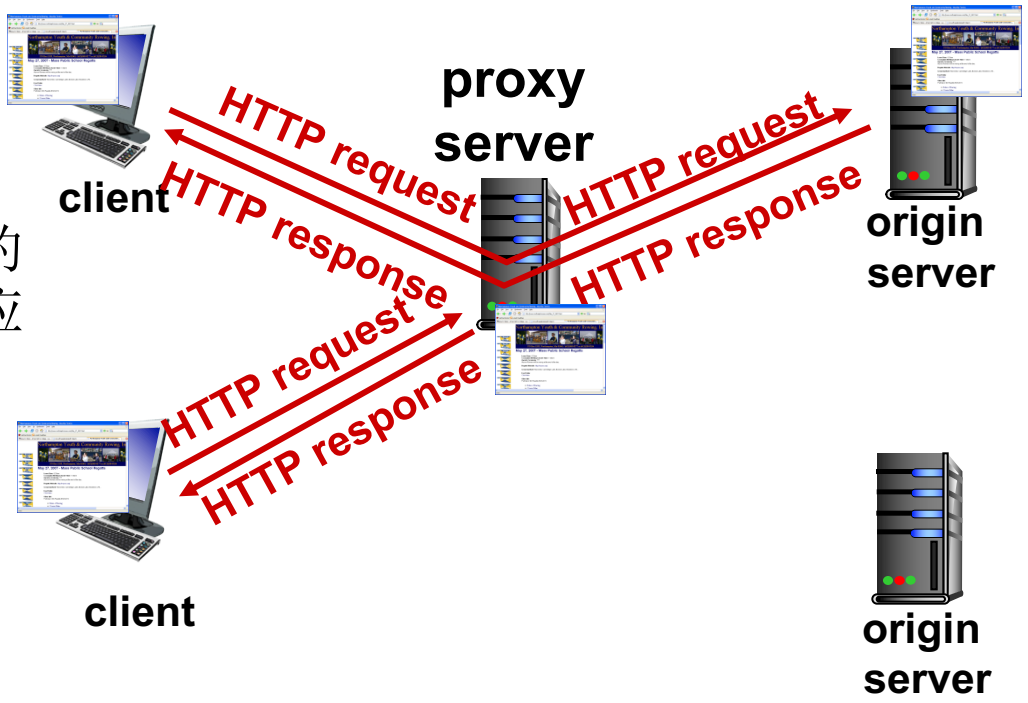
Cookie文件内容

Domain	Path	Content	Expires	Secure
toms-casino.com	/	CustomerID=497793521	15-10-02 17:00	Yes
joes-store.com	/	Cart=1-00501;1-07031;2-13721	11-10-02 14:22	No
aportal.com	/	Prefs=Stk:SUNW+ORCL;Spt:Jets	31-12-10 23:59	No
sneaky.com	/	UserID=3627239101	31-12-12 23:59	No

代理服务器 (Proxy Server)

目的：减轻Web服务器的负担，加快访问速度

- 在浏览器程序中设置通过代理访问
- 浏览器将所有HTTP请求均发送给代理
  - 如果在代理中找到请求的对象，则由代理返回响应
  - 否则代理将请求转发给要访问的Web Server，由该Server 响应
  - 代理缓存来自Web Server的网页对象



[Kurose]

Proxy(Web caching)

- 客户和原服务器之间的中转设备
- 既是客户端，又是服务器端
- 通常由ISP、公司或学校提供

- Proxy的优点
- 缩短了网页访问的响应时间
- 减轻了链路的负载
- 减轻了服务器的负载

- **浏览器*Caching*：**

浏览器缓存访问过的网页，进一步减少网络负载

缓存的更新：条件GET

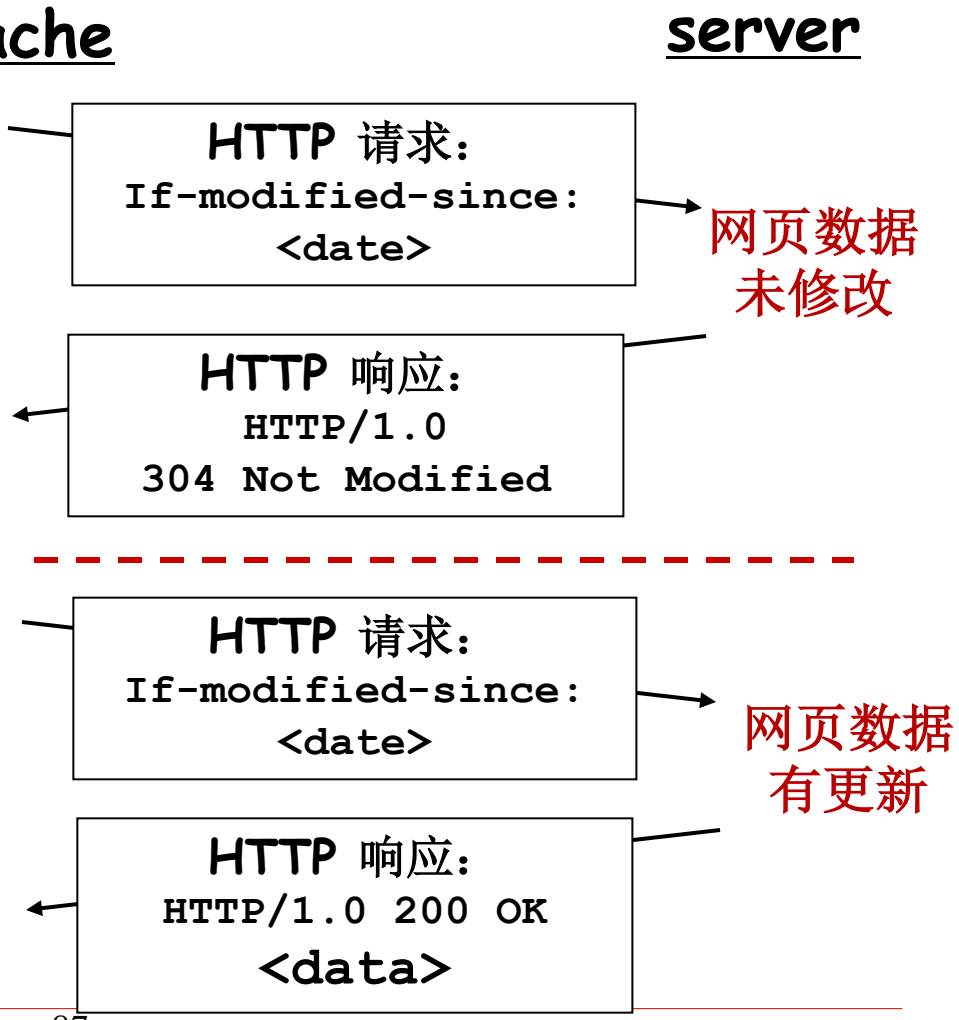
- 问题：如果cache中的对象过期了（不是最新版本）怎么办？
- Cache在HTTP请求中包含对象的最后更新时间：

  If-modified-since: <date>

  - 该时间包含在上次访问时Server的HTTP响应中

  Last-Modified: <date>
- Server如果没有更新对象，则返回

  HTTP/1.0 304 Not Modified



HTTP的安全性：HTTPS [RFC 2818]

- HTTP协议的安全隐患
  - 数据保密性问题：明文传输
  - 数据完整性问题：可能被中间节点篡改
  - 身份校验问题：可能遭受中间人攻击
- 将HTTP与SSL（Secure Socket Layer）协议相结合
- 对HTTP消息加密、对Web服务器/客户进行身份认证
- 在不安全的网络上建立一条安全的通道，防止窃听和途中攻击（MITM，Man-in-the-middle attack）

HTTP的安全性：HTTPS [RFC 2818]

- HTTP隧道（Tunneling）
  - 将一个协议数据封装在另一个协议内透明传输的方法
  - HTTP隧道：将SSL数据（加密方式、认证信息等）封装在HTTP消息内，代理对于消息中的数据不作处理，直接转发

小结：WWW概要

- 分布式、超媒体系统
- 采用C/S模型
  - Client：浏览器Browser（通用客户端）
  - Server：Web服务器
- 协议：HTTP
  - 使用TCP，服务器默认端口号为80
  - 基于ASCII的请求/响应消息
  - 无状态
- 增强机制
  - Cookie（无状态→有状态）
  - Proxy（缓存、条件Get）

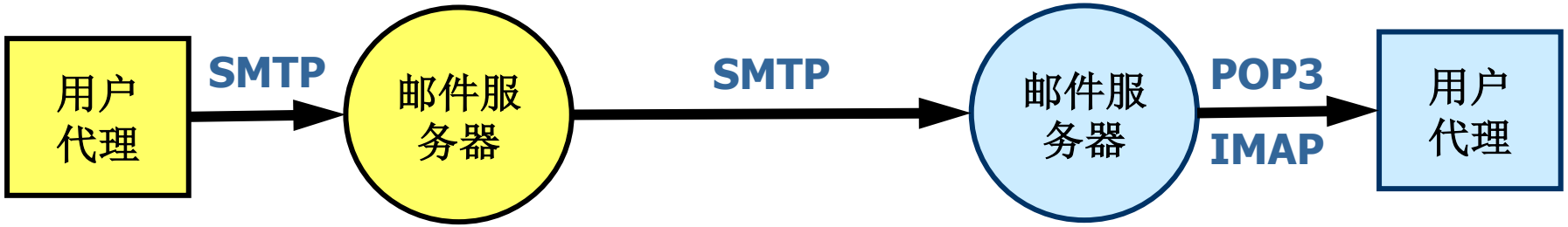
主要内容

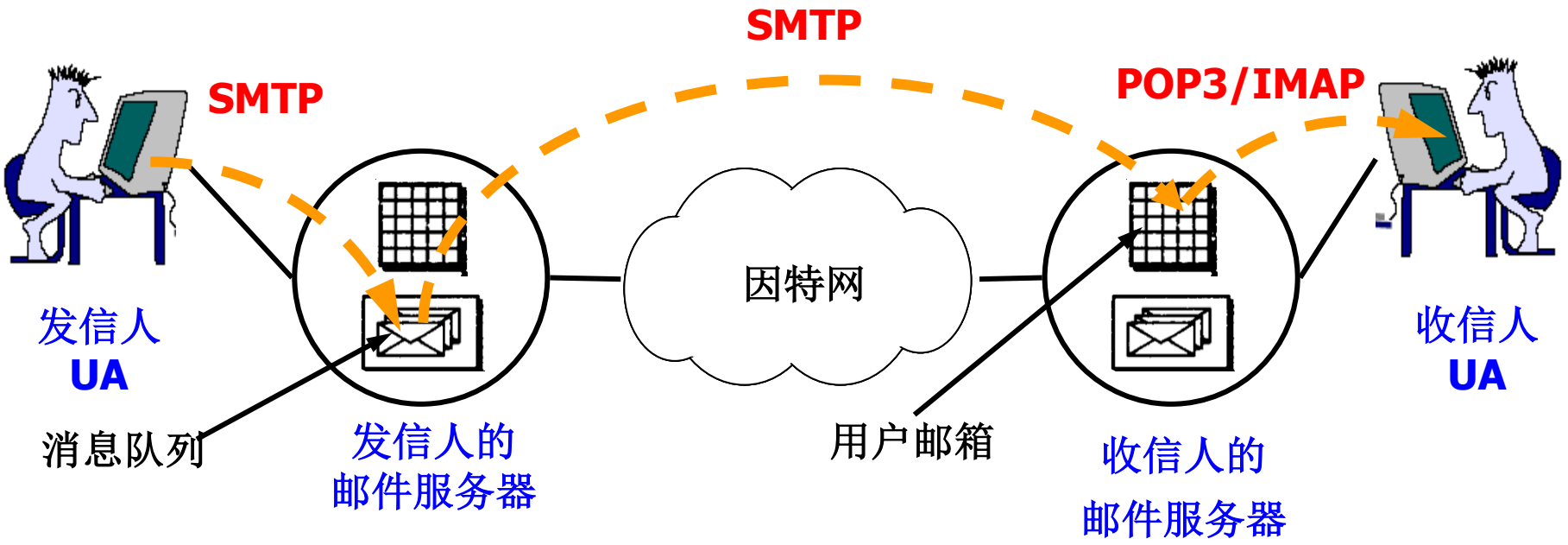
- 2.1 网络应用概述
- 2.2 DNS
- 2.3WWW应用和HTTP
- *2.4 Email应用*
  - 电子邮件系统的结构
  - 邮件地址和邮件格式
  - 邮件发送协议SMTP
  - 邮件访问协议POP3
- 2.5 FTP
- 2.6 远程登录协议：Telnet
- 2.7 应用层安全隐患

Email应用的特性

- 电子邮件
  - 两个计算机上的用户通过网络交换邮件消息（Mail Exchange）
- 异步应用，方便用户
- 提供一对多通信
- 价格低廉
- 谁提供Email服务？
  - ISP，如@163.com，@sina.com
  - 公司、学校、机构组织，如@bupt.edu.cn
  - 和其他应用一起，如@qq.com、@139.com

电子邮件系统的构成(1)


用户
代理
SMTP
邮件服
务器
SMTP
邮件服
务器
POP3
IMAP
用户
代理


SMTP
SMTP
POP3/IMAP
因特网
发信人
UA
消息队列
发信人的
邮件服务器
用户邮箱
收信人的
邮件服务器
收信人
UA

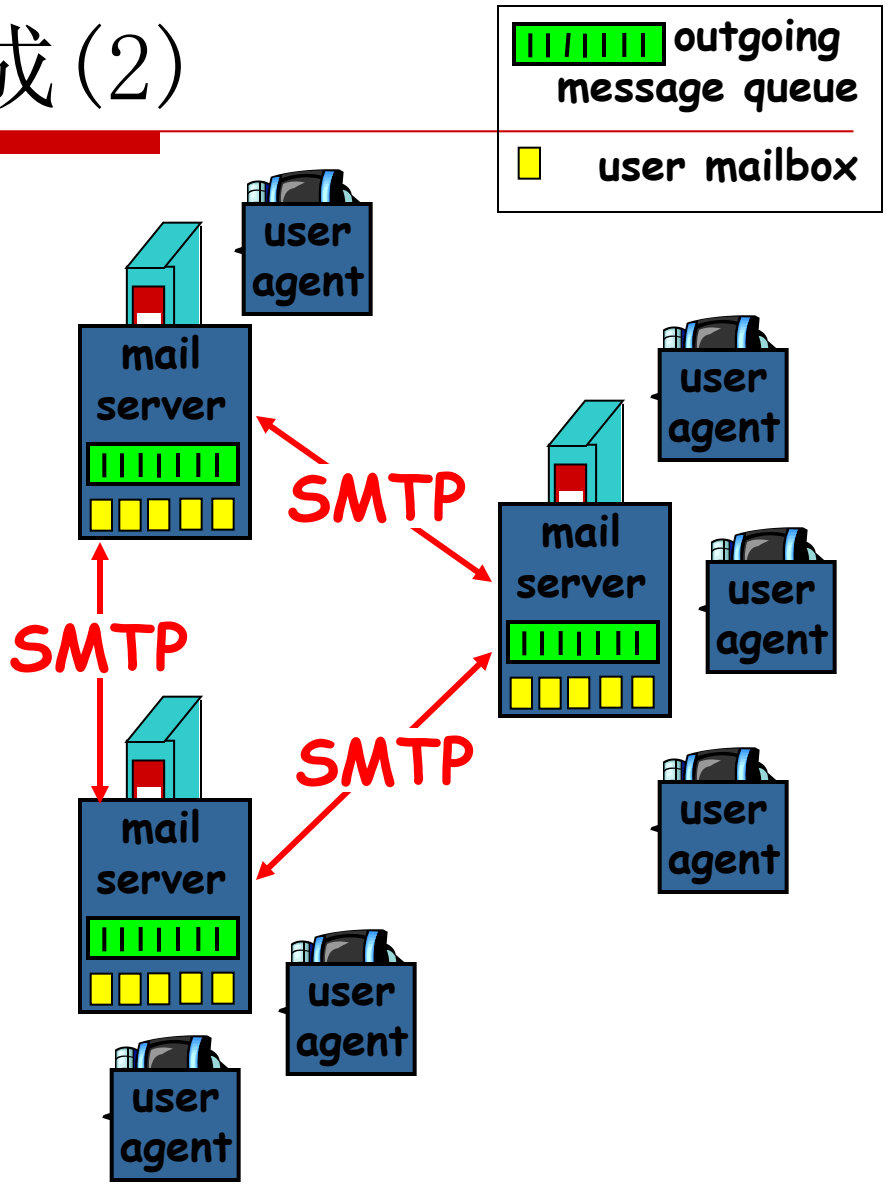
电子邮件系统的构成(2)

主要包含：

- 用户代理(user agents, UA)
- 邮件服务器(mail servers)
- 邮件传输协议：SMTP
- 邮件访问协议：POP3或IMAP

用户代理

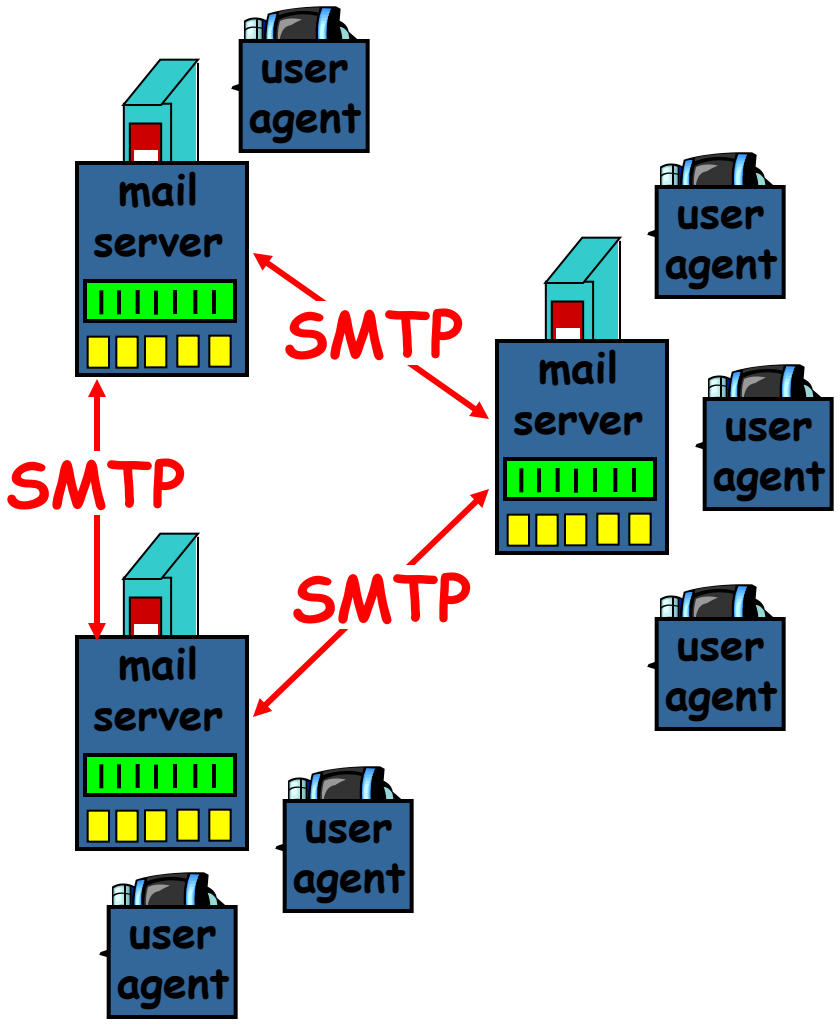
- 客户端程序
- 提供编辑、发送、显示邮件和邮件发送情况报告等功能
- 示例：Eudora, Outlook, Foxmail



电子邮件系统的构成(3)

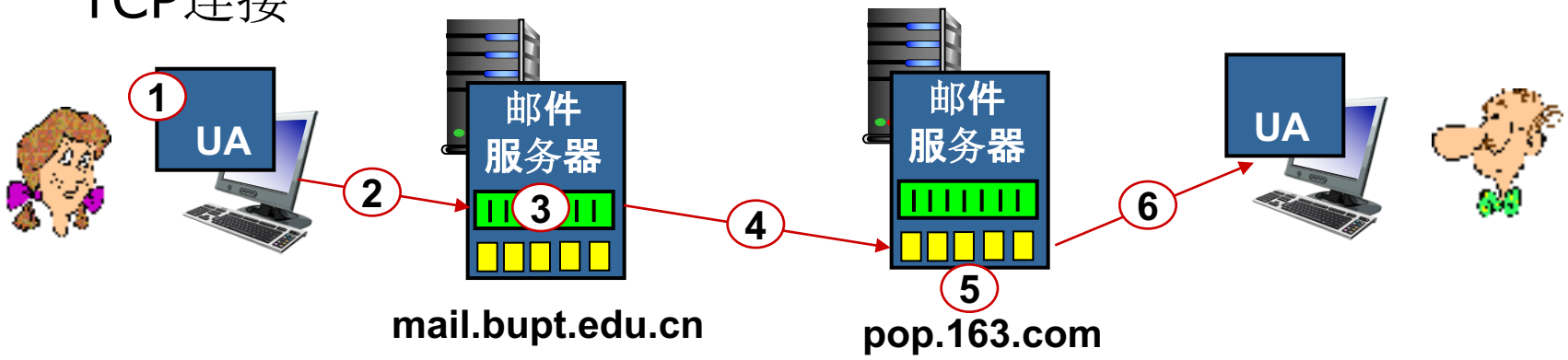
邮件服务器

- 功能：类似“邮局”，接收和转发电子邮件，向发信人报告邮件发送状态
- 邮箱：保存收到的邮件
- 消息队列：暂存待发邮件
- 使用SMTP协议传输邮件
- 采用C/S模式
- 可能是Server，也可能是Client
  - client: 发送邮件时
  - server: 接收邮件时



电子邮件的传输过程示例

1) Alice在我邮校园网内用UA (如Foxmail) 编辑邮件，收件人是 bob@163.com
2) Foxmail把邮件发送到我邮的邮件服务器，邮件被放入消息队列
3) 我邮的邮件服务器建立到163.com邮件服务器的TCP连接
4) 我邮的邮件服务器把邮件转发给pop.163.com
5) 163邮件服务器把邮件保存到Bob的邮箱
6) Bob使用UA(如outlook)下载和查看邮件



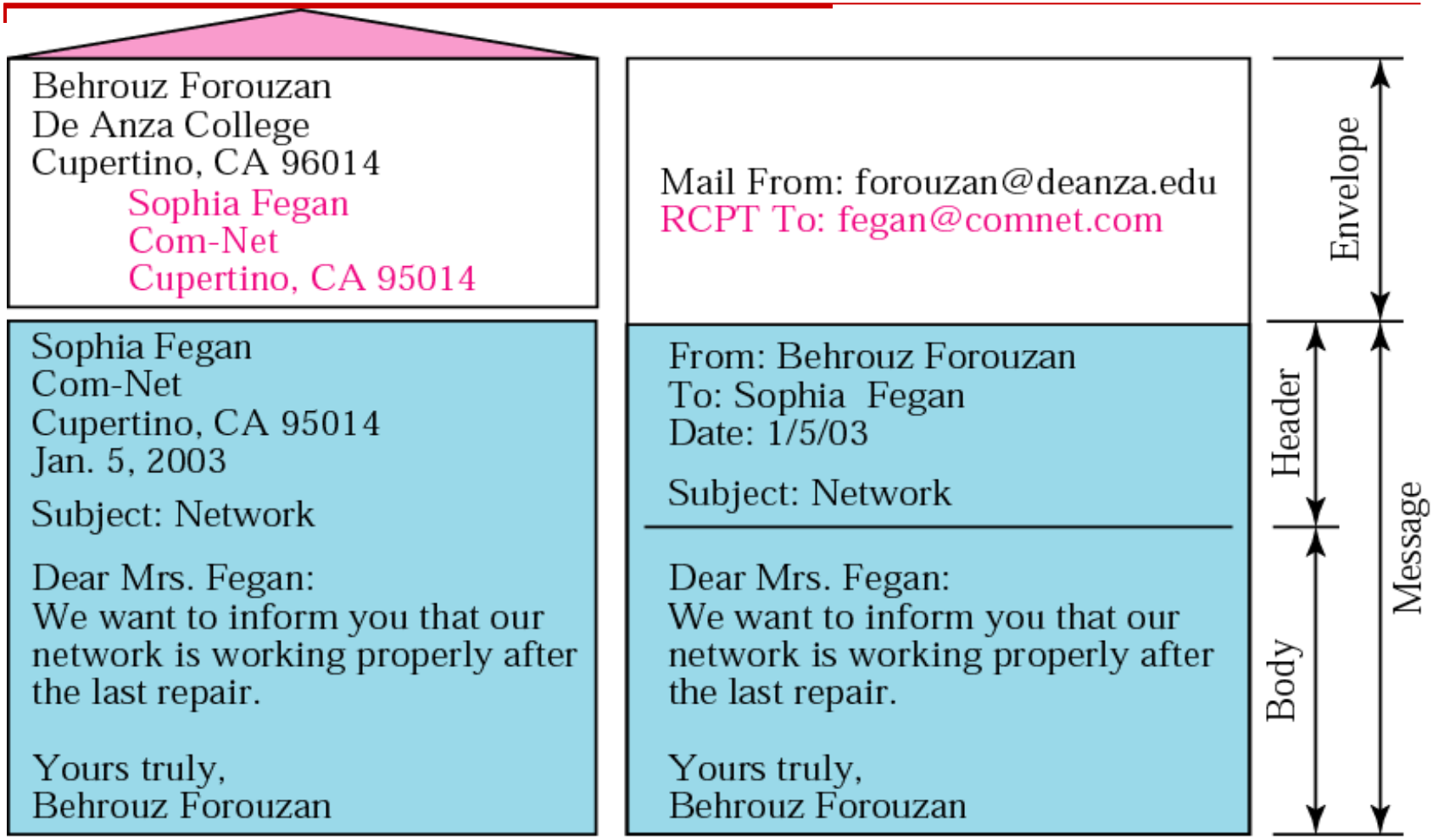
电子邮件地址

- 电子邮件用户必须有一个电子邮件地址
  - 许多网站提供免费电子邮件服务，需要的话可到这些网站上申请一个邮箱（电子邮件地址）。
- 电子邮件地址由两部分组成：
  - 用户名
  - 邮箱所在的邮件服务器的主机域名
  - 全球唯一性
- 用户名和邮件服务器域名之间用“@”隔开

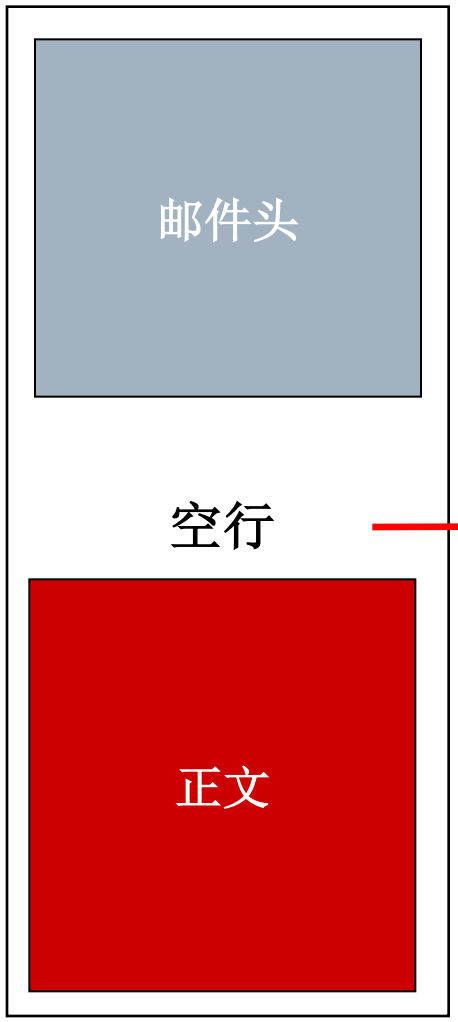
  用户名@邮件服务器域名

  - 例如：
    - liujy@bupt.edu.cn

传统信件 vs. Email



邮件消息格式(RFC5322)




```
Date: Tue, 8 Mar 2024 13:15:53 +0800
From: "chengli" chengli@bupt.edu.cn
To: "Alice" <alice@example.com.cn>
Subject: Test
Message-ID: <20240308131553659234 6
            @bupt.edu.cn>

This is test message…
```

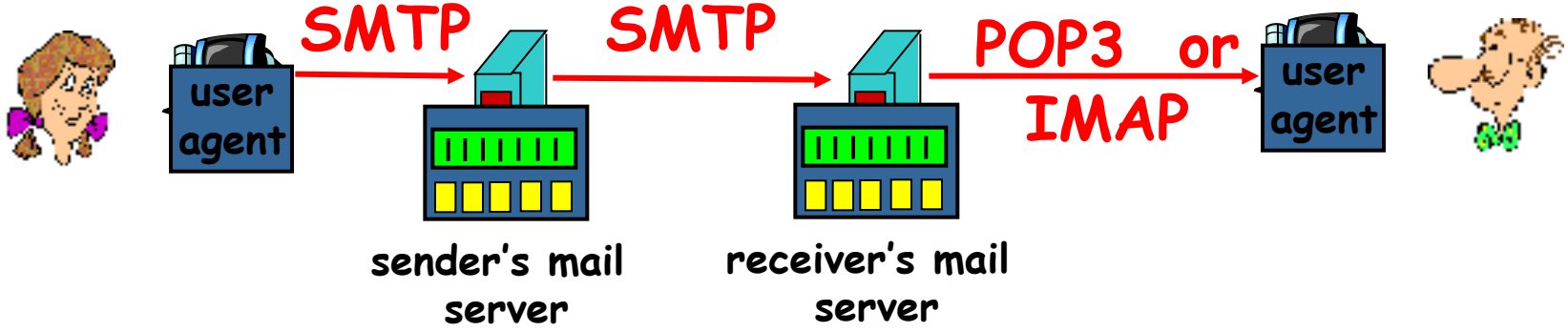
只支持***ASCII***文本邮件！

主要的邮件头

邮件头字段分类	字段	含义描述
发信人相关字段	From:	建立邮件的用户Email地址
	Sender:	发送邮件的用户Email地址
	Reply-to:	回邮地址
收信地址相关字段	To:	一个或多个收信人地址
	Cc:	抄送地址
	Bcc:	暗送地址
发信时间	Date:	邮件发送的日期和时间
标识字段	Message-Id:	邮件的唯一标识
	References:	其他相关标识
信息字段	Subject:	邮件摘要
	Keywords:	发信人设置的关键字
跟踪字段	Received:	邮件所经过的邮件服务器列表
	Return-Path:	回邮路径

Email的通信协议

- SMTP：发送/转发邮件——PUSH服务
- POP3或IMAP：从邮箱收邮件到本地计算机——PULL服务



SMTP [RFC 2821]

- 简单邮件传输协议，Simple Mail Transfer Protocol
- 采用C/S模式
- 传输层协议使用TCP，Server端口号是25
- 直接传输：邮件直接从发信人的服务器传输到收信人的服务器，而不由中间服务器转发
- 命令/响应方式交互（类似HTTP ）
  - 命令(Command): ASCII文本
  - 应答(Reply): 状态码和短语
- 只支持7位ASCII文本邮件
  - 格式由RFC5322确定

SMTP交互示例

Client建立到Server的TCP连接

```
S: 220 AnyMacro Email System
C: EHLO mail.bupt.edu.cn
S: 250-mx4.bupt.edu.cn
C: MAIL FROM: <chengli@bupt.edu.cn>
S: 250 Ok
C: RCPT TO: <wangc@bupt.edu.cn>
S: 250 Ok
C: RCPT TO: <noname@bupt.edu.cn>
S: 550 Recipient address rejected: User unknown in
  local recipient table
C: Data
S: 354 End data with <CR><LF>.<CR><LF>
C: 邮件（消息头+正文）
C: .
S: 250 Ok: queued as E75A2B644B
C: QUIT
S: 221 Bye
```

SMTP 总结

- SMTP使用持久连接
- SMTP只支持传输ASCII文本消息
- SMTP服务器使用CRLF.CRLF(只包含".的一行)来判断邮件结束

与HTTP比较：

- HTTP: pull
- SMTP: push
- 均使用持久连接
- 均采用ASCII 命令/响应进行交互，响应均包含状态码

邮件访问协议

- 邮件访问协议：从邮件服务器收/读邮件
  - POP：Post Office Protocol [RFC 1939]
    - 服务器端口号：110
    - 对用户进行身份认证、下载邮件到用户计算机
  - IMAP：Internet Mail Access Protocol [RFC 1730]
    - 服务器端口号：143
    - 比POP功能强、复杂
    - 可以在邮件服务器上处理邮件
  - Webmail：使用浏览器读邮件

POP3协议示例

身份认证阶段

- 客户端命令：
  - **user**：指定用户名
  - **pass**：指定访问密码
- 服务器的响应：
  - **+OK**
  - **-ERR**

事务处理阶段

- 客户端命令：
  - **list**：显示邮件个数
  - **retr**：下载邮件
  - **dele**：删除邮件
  - **quit**

```
S: +OK Hello there
C: user test
S: +OK Password required
C: pass 123456
S: +OK logged in

C: stat
S: +OK 2 4686
C: list
S: 1 2268
S: 2 2418
S: .
C: retr 1
S: +OK 2268 octets follow
S: 邮件（消息头+正文）
S: .
C: quit
S: +OK Bye-bye
```

POP3 vs. IMAP

POP3

- 用户使用客户端程序访问邮件服务器
- 用户下载邮件时可以选择
  - 下载后从服务器删除，或
  - 在服务器上保留副本
- 服务器上一个用户只有一个文件夹：inbox
- 会话状态信息不保留

IMAP

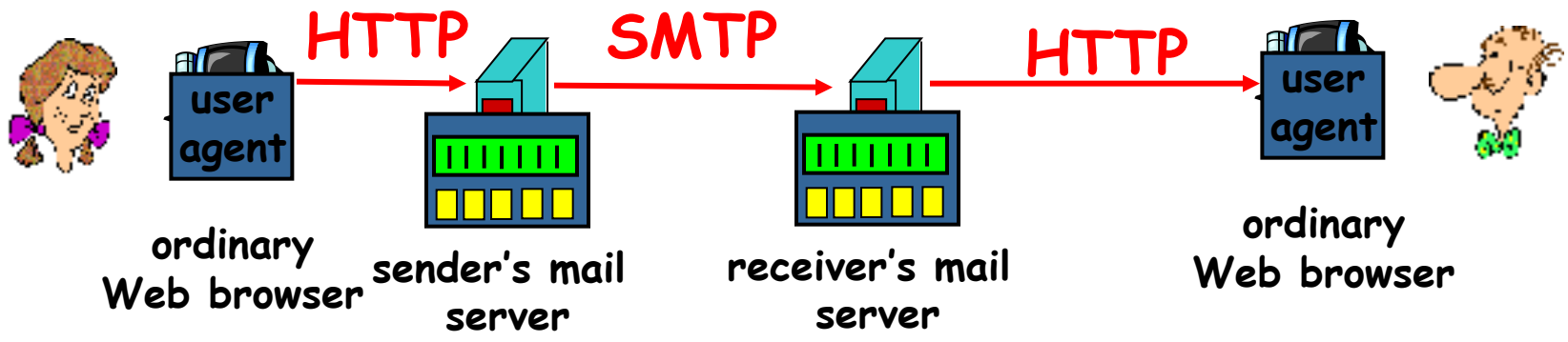
- 用户可以使用客户端或浏览器访问服务器
- 邮件一直保留在服务器
- 用户可以在服务器上创建并管理多个文件夹
- 用户可以只读取邮件头，或部分邮件（如文本）
- 示例：gmail

示例：bupt.edu.cn的邮件服务器，可以选择使用POP3或者IMAP访问邮箱

POP3 vs. IMAP

操作位置	操作内容	IMAP	POP3
收件箱	阅读、标记、移动、删除邮件等	客户端与邮箱更新同步	仅客户端内
发件箱	保存到已发送	客户端与邮箱更新同步	仅客户端内
创建文件夹	新建自定义的文件夹	客户端与邮箱更新同步	仅客户端内
草稿	保存草稿	客户端与邮箱更新同步	仅客户端内
垃圾文件夹	接收误移入垃圾文件夹的邮件	支持	不支持
广告邮件	接收被移入广告邮件夹的邮件	支持	不支持

Web mail



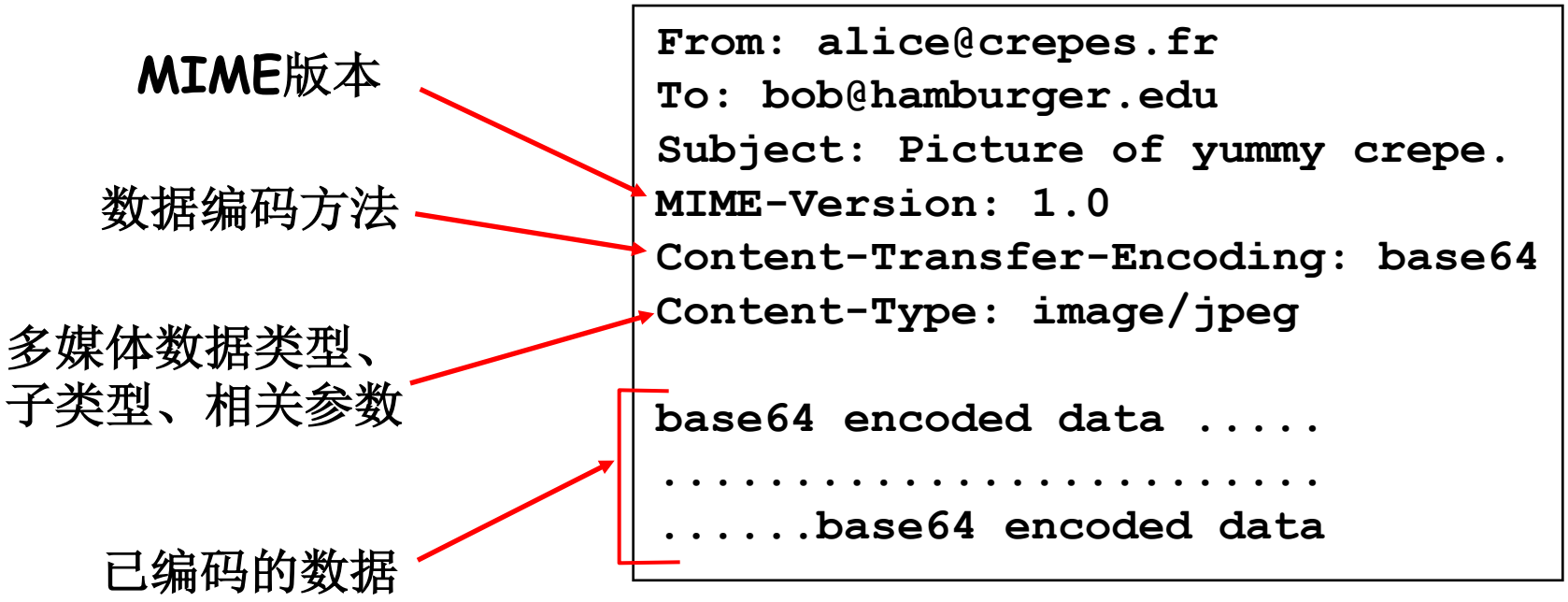
- 用户通过浏览器访问邮件服务器
- 用户可以在服务器上创建和管理自己的文件夹

邮件格式：多媒体扩展

- SMTP的缺点：
  - 不能传送可执行文件或其他的二进制对象
  - 限于传送7位的ASCII码
  - 会拒绝超过一定长度的邮件
  - 某些SMTP的实现没有完全按照SMTP的因特网标准
- 通用因特网邮件扩充（MIME）
  - 并未改动或取代SMTP
  - 定义了传送非ASCII码的规则

邮件格式：多媒体扩展

- MIME: multimedia mail extension, RFC 2045, 2056
- 在邮件头增加字段，说明媒体数据的类型



MIME邮件头字段

消息头字段	含义描述
MIME-Version:	标识MIME的版本
Content-Id:	邮件消息内容的唯一标识符
Content-Type:	邮件中的数据类型
Content-Transfer-Encoding:	邮件传输的编码方式
Content-Description:	关于邮件内容的英文描述

MIME支持的数据类型

类型	子类型	含义描述
Text	Plain	无格式文本
	Richtext	有简单格式的文本
Image	Gif	GIF格式静态图像
	Jpeg	JPEG格式静态图像
Audio	Basic	音频
Video	Mpeg	MPEG格式视频
Application	octet-stream	字节序列
	Postscript	PostScript文档
Message	RFC5322	RFC 5322格式的消息
	Partial	部分消息（消息被拆分，以便于传输）
	External-body	访问外部数据的机制（访问方法、URL等）
Multipart	Mixed	消息的多个独立部分按顺序组织在一起
	Alternative	相同消息的另一种格式
	Parallel	消息的多个部分必须同步
	Digest	每个部分都是完整的RFC 5322消息

E-mail的三种编码标准

- 7位ASCII码
- QP(Quote-Printable)
  - QP的规则是对于信件中的7位数据无须重复编码，仅将8位的数据转成7位。QP编码适用于非ASCII码的文字内容，将每个字节用两个16进制数值表示，然后在前面加“=”。所以经过QP编码后的文字通常是这个样子：：=B3=C2=BF=A1=C7=E5=A3=AC=C4=FA=BA=C3=A3=A1
- Base64
  - 其编码规则是将整个文件重新编码成7位，通常用于传送二进制文件。
  - 先把二进制代码划分为一个个24位长的单元，然后把每一个24位单元划分为4个6位组。每一个6位组按以下方法转换成ASCII码
    - A-Z表示0-25，a-z表示26-51，0-9表示52-61，+表示62，/表示63，==表示最后一组只有8位的，=表示最后一组只有16位的。
  - Base64编码后的文字通常是这个样子：pGquYaZuoUmn2qxPseepc6dnoUGr3LCqv70ms
- 具有MIME功能的Email软件大都能自动判别邮件是采用何种编码，然后自动选择用QP或Base64来解码。

E-mail的三种编码标准

- Base64
  - 其编码规则是将整个文件重新编码成7位，通常用于传送二进制文件。
  - 先把二进制代码划分为一个个24位长的单元，然后把每一个24位单元划分为4个6位组。每一个6位组按以下方法转换成ASCII码
    - A-Z表示0-25，a-z表示26-51，0-9表示52-61，+表示62，/表示63，==表示最后一组只有8位的，=表示最后一组只有16位的。
  - Base64编码后的文字通常是这个样子：
    pGquYaZuoUmn2qxPseepc6dnoUGr3LCqv70ms
- 例：

24位二进制代码　01001001 00110001 01111001
划分为4个6位组　010010 010011 000101 111001
对应的base64编码　S　T　F　5
用ASCII码发送 01010011 01010100 01000110 00110101

MIME
邮件示例

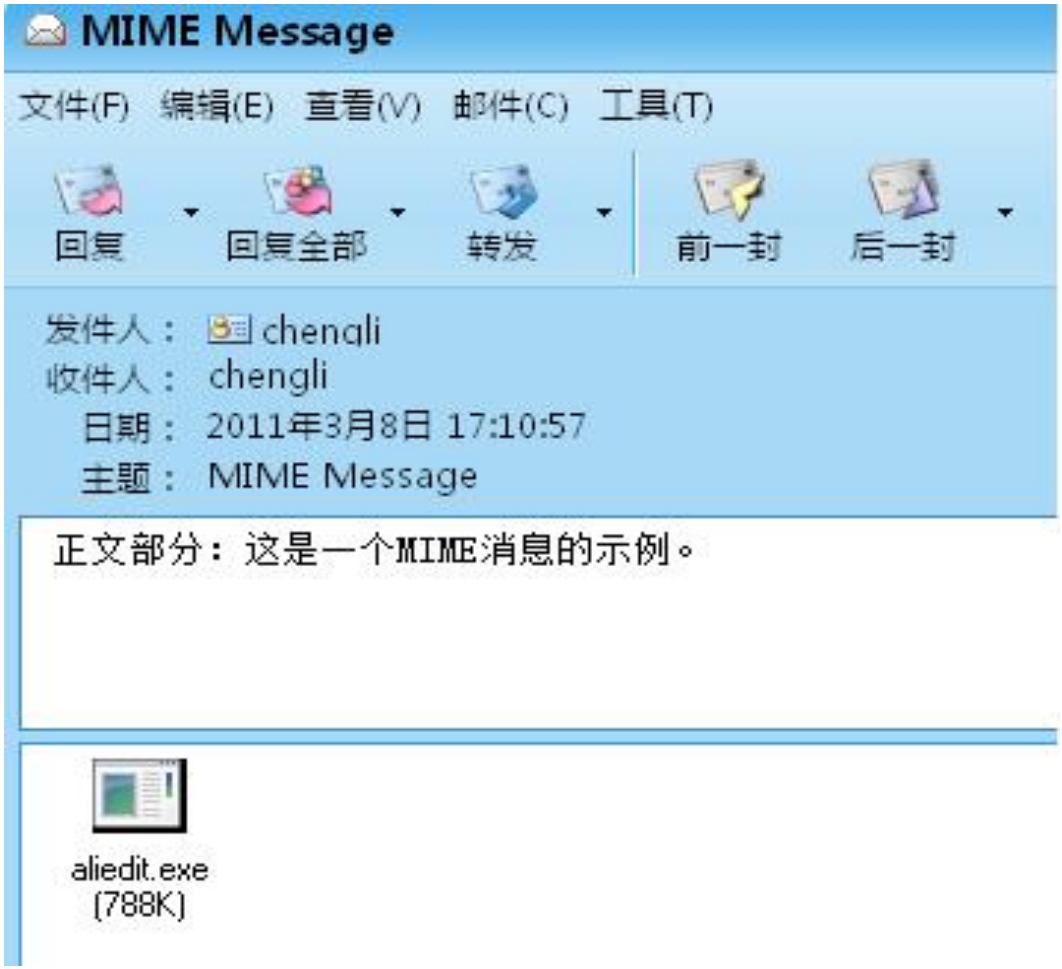
```
Date: Tue, 8 Mar 2011 17:10:37 +0800
From: “chengli” <chengli@bupt.edu.cn>
To: "chengli" <chengli@bupt.edu.cn>
Subject: MIME Message
Message-ID: <201103081710371330466@ebupt.com>
X-mailer: Foxmail 6, 15, 201, 23 [cn]
Mime-Version: 1.0
Content-Type: multipart/mixed;
        boundary="=====001_Dragon736704835806_====="

This is a multi-part message in MIME format.

--=====001_Dragon736704835806_=====
Content-Type: text/plain;
        charset="gb2312"
Content-Transfer-Encoding: base64

1f3OxLK/t9ajutXiysfSu7j2TUlNRc/7z6K1xMq+wP2how0K
--=====001_Dragon736704835806_=====
Content-Type: application/octet-stream;
        name="aliedit.exe"
Content-Transfer-Encoding: base64
Content-Disposition: attachment;
        filename="aliedit.exe"
```

Foxmail显示的邮件



Email的安全性问题

- SMTP和POP3对于安全性考虑不足
  - 未强制要求对发信人的身份进行认证
    - 伪造发信人
  - 发信人用户名和密码没有加密
  - 邮件明文传输
- SSL协议为SMTP和POP3提供了安全的传输通道，所有数据加密后再传输
  - SMTP：服务器端口465
  - POP3：服务器端口995

小结：Email概要

- 异步式通信
- 多个应用层协议/标准
  - 发邮件：SMTP
  - 收邮件：POP3/IMAP
  - 邮件格式：MIME（用于Email和Web）
  - Webmail
- 传输层：TCP
  - SMTP服务器端口25
  - POP3服务器端口110
- 增强安全性：SSL

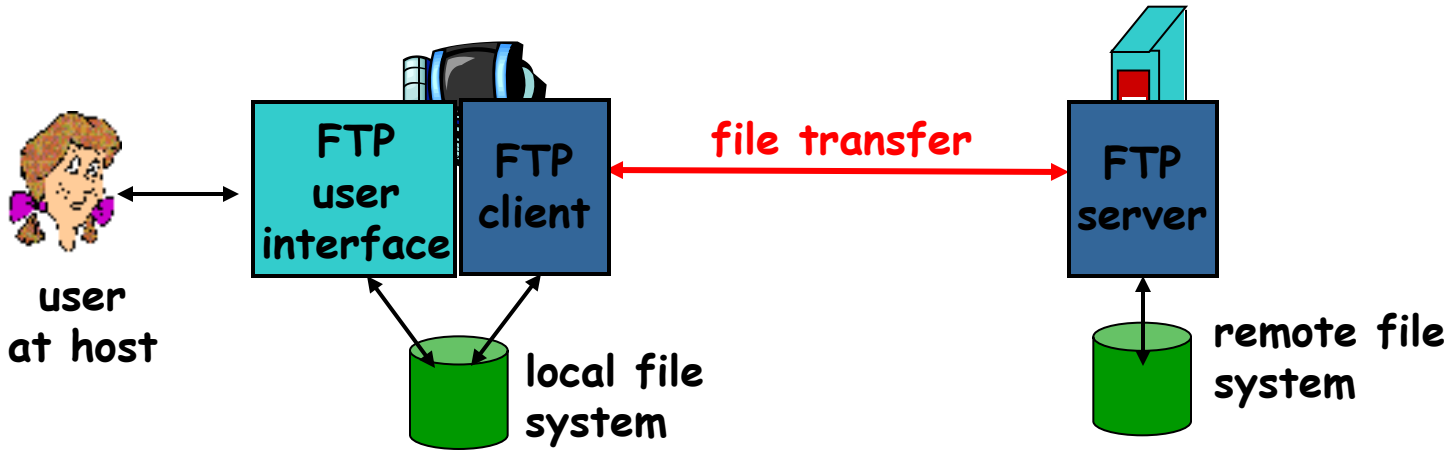
主要内容

- 2.1 网络应用概述
- 2.2 DNS
- 2.3WWW应用和HTTP
- 2.4 Email应用
- *2.5 FTP*
- 2.6 远程登录协议：Telnet
- 2.7 应用层安全隐患

FTP的特性

- 文件传输协议
  - 用于将文件从一台计算机传输到另一台计算机上
- 最早的应用层协议
  - 1971年4月，RFC114，使用NCP（网络控制协议）传输数据
  - 1980年6月，RFC765，使用TCP/IP传输数据
  - 1985年10月，RFC959，FTP的基本规范

FTP：文件传输协议



- 和远程主机之间上传/下载文件
- C/S模式
  - *client:* 连接到服务器，发起文件传输
  - *server:* 远程主机，一般用于保存大量文件
- 传输协议使用TCP

FTP：两个连接

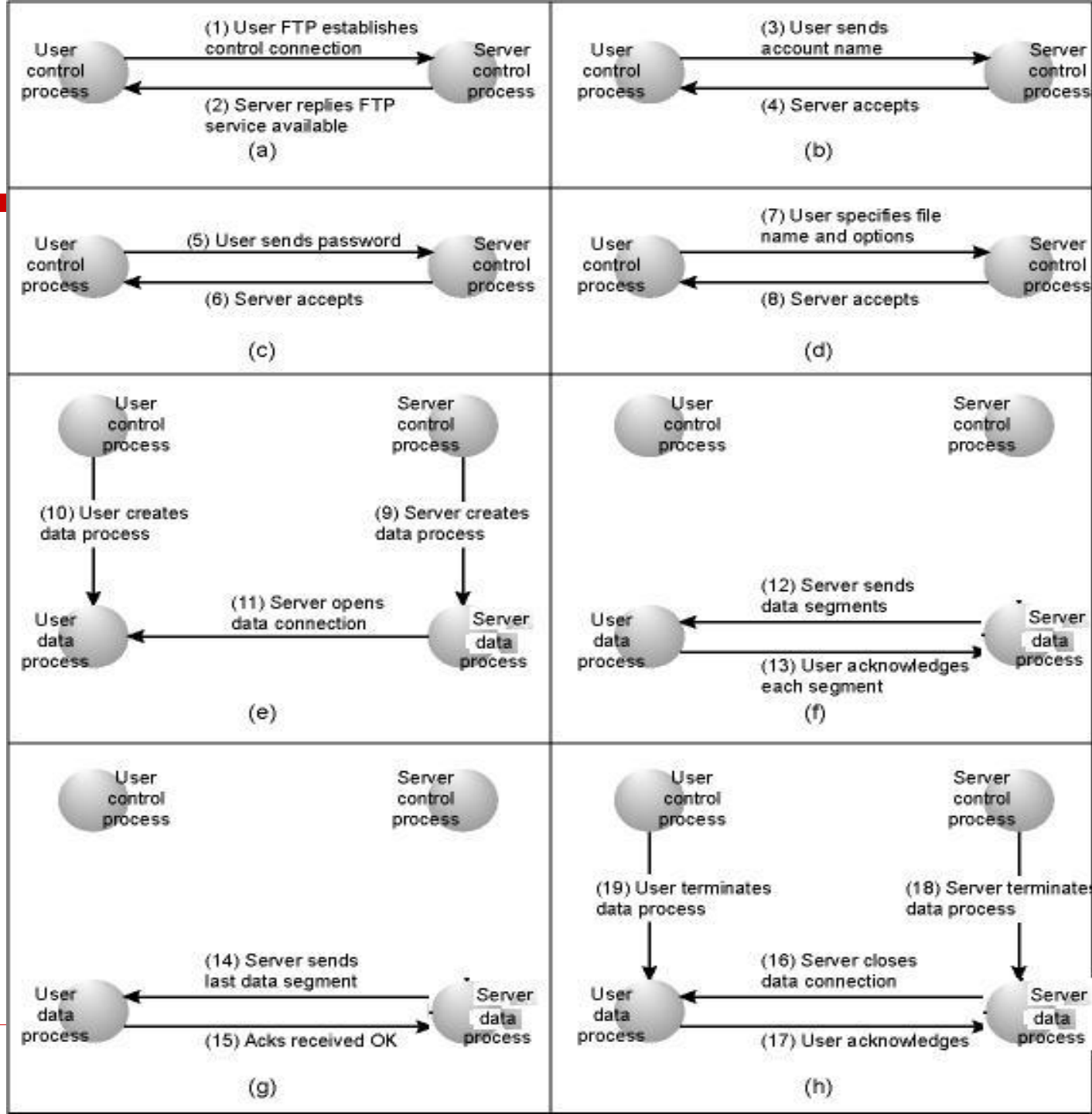
- **控制连接**：
  - FTP client首先与FTP server（端口21）建立控制连接
  - 进行身份验证
  - 传输命令和响应
  - 持久连接：在访问期间，控制连接一直存在
- **数据连接**：
  - 当需要传输数据（如显示目录、上传文件、下载文件）时，由Server（或者Client）建立数据连接
  - 数据传输结束后，连接关闭
  - 临时连接

- 如果要传输下一个文件，需要重新建立数据连接
- **FTP server** 维护“交互状态”：当前目录、身份验证情况等

FTP
文件传输过程示例

Control process:
传输和处理FTP命令

Data process:
传输数据



FTP命令和返回码

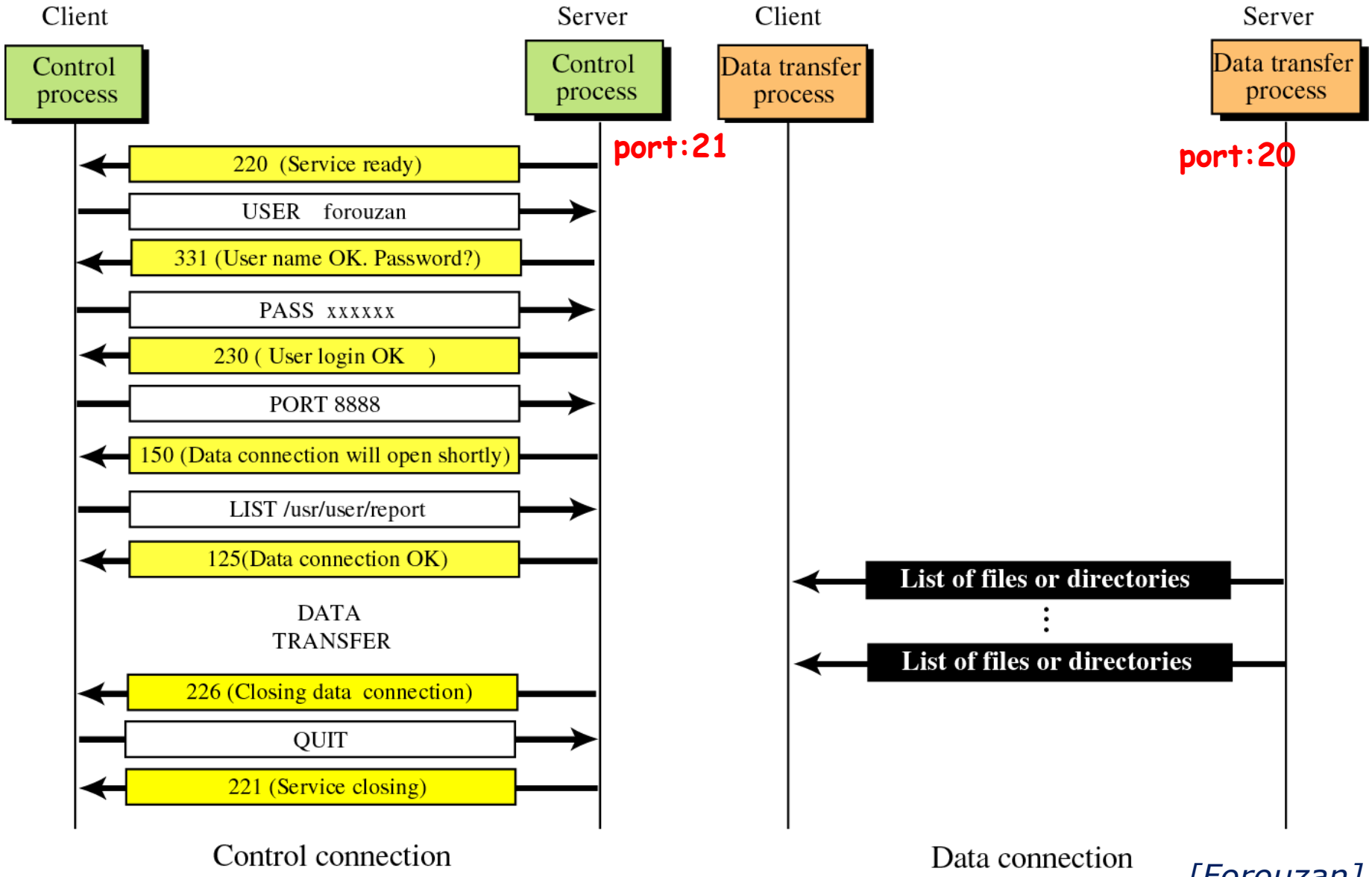
主要FTP命令：

- USER *username*
- PASS *password*
- LIST 请求目录和文件列表
- RETR filename 请求下载文件
- STOR filename 请求上传文件
- DELE filename 请求删除文件
- QUIT 请求结束FTP会话
- …

主要的状态码：

- 331 Username OK, password required
- 125 data connection already open; transfer starting
- 150 File status okay; about to open data connection
- 250 Requested file action okay, completed.
- 221 Service closing control connection.
- …

FTP交互示例：显示目录列表



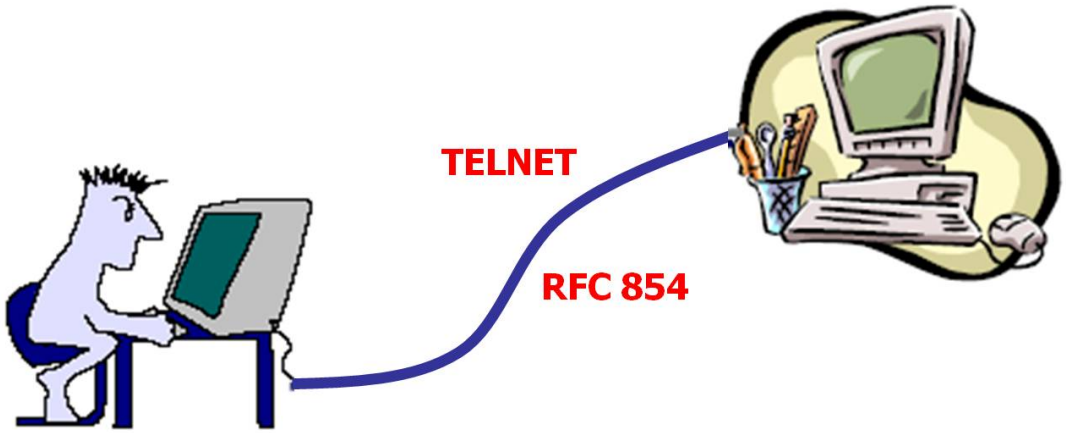
[Forouzan]

主要内容

- 2.1 网络应用概述
- 2.2 DNS
- 2.3WWW应用和HTTP
- 2.4 Email应用
- 2.5 FTP
- *2.6 远程登录协议：Telnet*
  - Telnet的功能和特点
  - NVT和选项协商
- 2.7 应用层安全隐患

远程登录应用与TELNET

- 远程登录
  - 从一台计算机通过网络登录到远程另一台计算机上进行操作
  - 应用：远程维护、BBS
- TELNET：终端仿真协议
  - 终端与计算机之间的通信协议
  - 终端：只有输入输出设备，没有CPU、内存

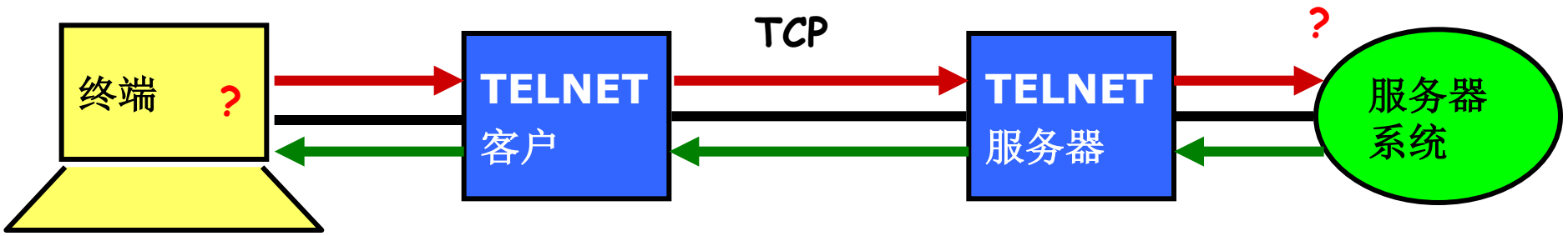


TELNET的特性

- C/S模型
  - 终端：客户端
  - 远程主机：服务器
- 基于TCP，服务器端口号：23
- 通用、双向、基于8位字符的通信协议

TELNET的特性：NVT

- 问题：终端和主机数据格式不统一

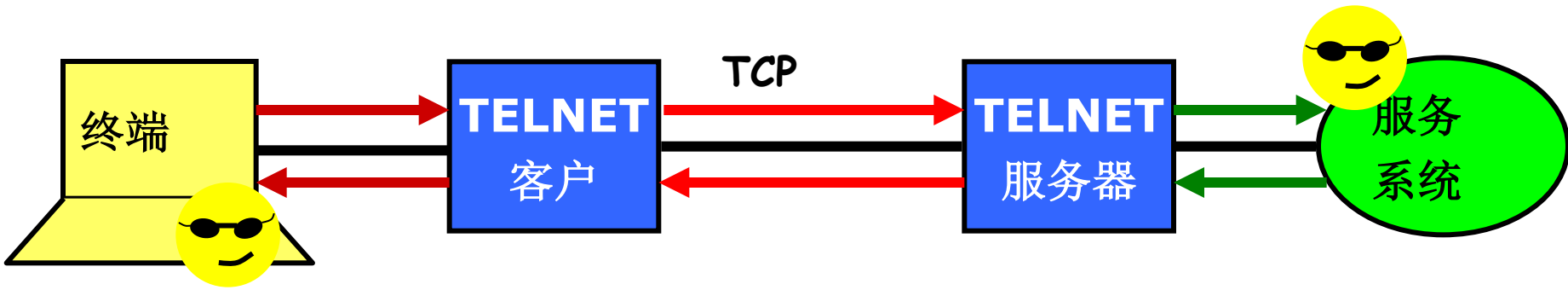


客户系统的格式

服务器系统的格式

NVT的功能

- NVT：网络仿真终端
  - 标准的数据格式
  - 实现异构设备的互联



客户系统的格式

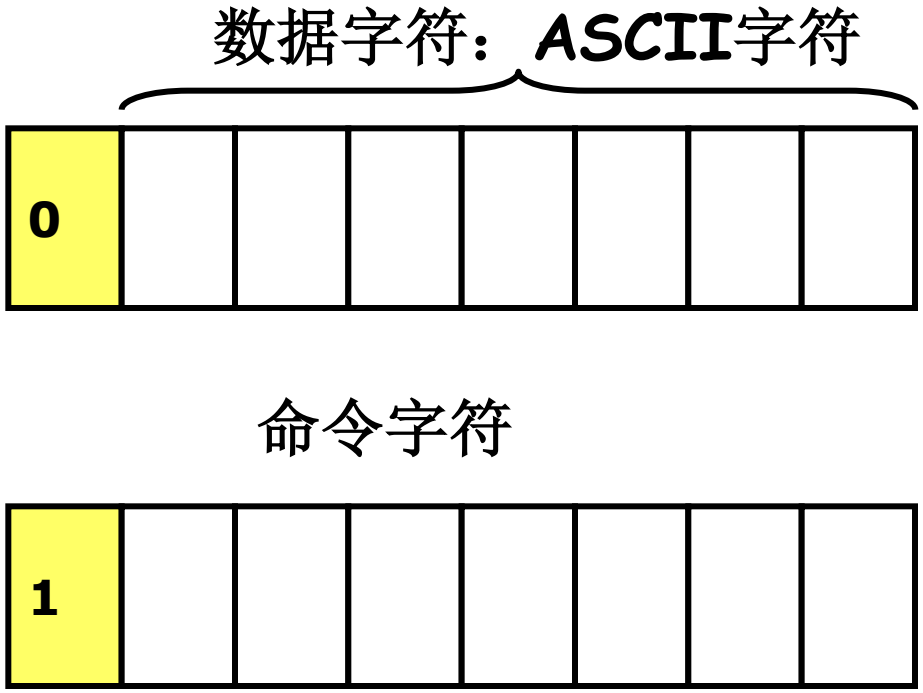
服务器系统的格式

NVT格式

TELNET 客户和服务器完成**NVT**格式和本系统格式之间的转换

NVT格式

- 8位字符



主要的NVT控制字符

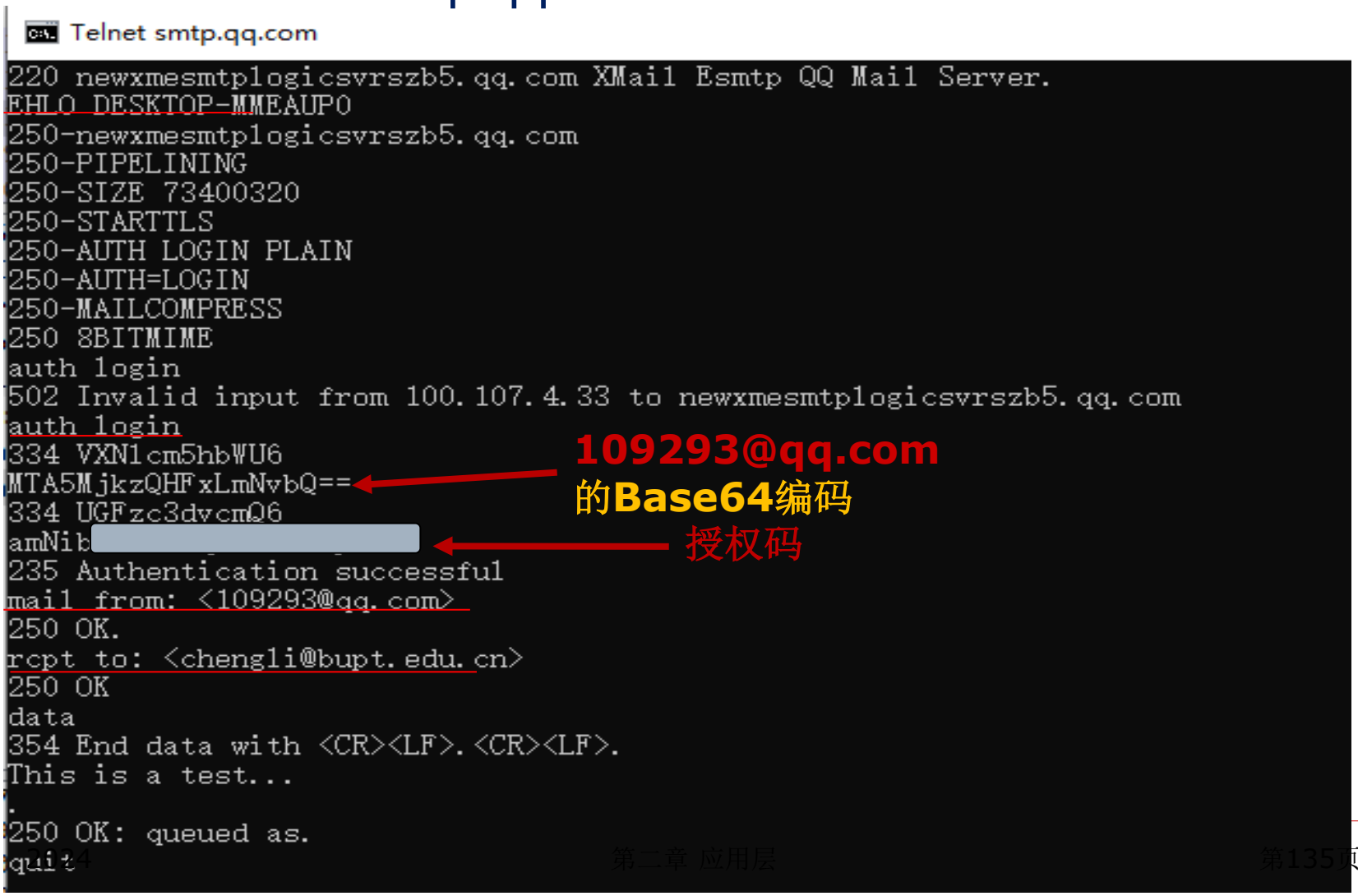
控制字符	十进制值	含义描述
IAC	255	解释字符，表示随后的字符是命令字符
DONT	254	要求对端不采用某个选项
DO	253	要求对端采用某个选项
WONT	252	本端不采用某个选项
WILL	251	本端采用某个选项
SB	250	开始子选项协商
EL	248	删除一行
EC	247	删除一个字符
SE	240	子选项协商结束
…		

选项协商

- 最初，NVT只支持最基本的能力
  - 单色显示器、字符终端、ASCII字符集
- 选项协商：
  - 终端和远程主机协商可支持的能力集
  - 便于扩充能力，而不必修改原有协议
- 可协商的选项示例
  - 远程主机是否采用Echo（回传收到的数据）
  - 终端类型，如ANSI、VT100等
  - 窗口大小，如80*25
  - 传送模式：每次发送一个字符，还是一行字符
  - ……

telnet程序：辅助了解其他应用层协议

- 使用TELNET协议连接到远程主机
- C:> telnet smtp.qq.com 25



telnet程序

- 使用TELNET协议连接到远程主机
- C:> telnet pop.126.com 110

```
+OK Welcome to coremail Mail Pop3 Server (126coms[f7aa0ed0270c139a0807240dd70090
6bs])
user test0318
+OK core mail
pass 123456
+OK 2 message(s) [14732 byte(s)]
list
+OK 2 14732
1 13403
2 1329
.
retr 2
+OK 1329 octets
Received: from mail.bupt.edu.cn (unknown [211.68.71.7])
        by mx1 (Coremail) with SMTP id H8mowLDbb_JKLoJNhybUCA--.37S2;
        Thu, 17 Mar 2011 23:52:42 +0800 (CST)
Received: from localhost (localhost.localdomain [127.0.0.1])
        by mx3.bupt.edu.cn (Postfix) with SMTP id F1D5A1BD77
        for <test0318@126.com>; Thu, 17 Mar 2011 23:52:48 +0800 (CST)
```

TELNET的安全性问题

- 明文传输、易被窃取
- 1995年，替代协议SSH（Secure SHell）
  - 对通信双方进行身份认证，
  - 对所有数据加密
  - 基于TCP，服务器端口号22

主要内容

- 2.1 网络应用概述
- 2.2 DNS
- 2.3WWW应用和HTTP
- 2.4 Email应用
- 2.5 FTP
- 2.6 远程登录协议：Telnet
- *2.7 应用层安全隐患*

对传统应用协议的攻击（1）

- Sniffing：嗅探
  - 捕获网络上传输的数据
  - 如：Wireshark软件
- Spyware：间谍软件
  - 驻留在用户计算机内，监控用户的操作，收集其个人信息、控制用户计算机（如下载恶意软件）
  - 如：Keyloggers监控键盘和鼠标操作
- Phishing：钓鱼邮件
  - 伪装成银行发送的邮件，让用户访问假冒的银行网站以窃取用户的账户和密码

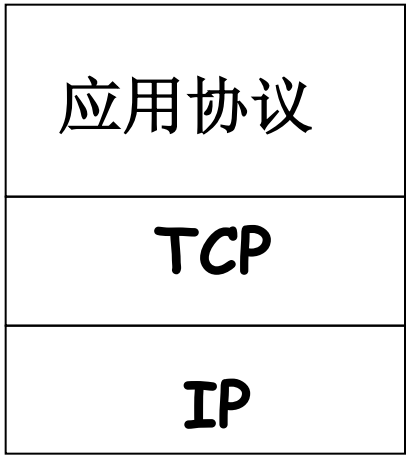
对传统应用协议的攻击（2）

- DOS(Denial of Service，拒绝服务)
  - 耗尽服务器资源，使服务器不能正常提供服务
  - 如，缓存溢出攻击
- Virus：病毒
  - 恶意程序，破坏计算机资源
  - 感染+传播
- Trojan: 特洛伊木马
  - 隐藏在合法程序内
  - 在满足条件时发作
  - 不复制

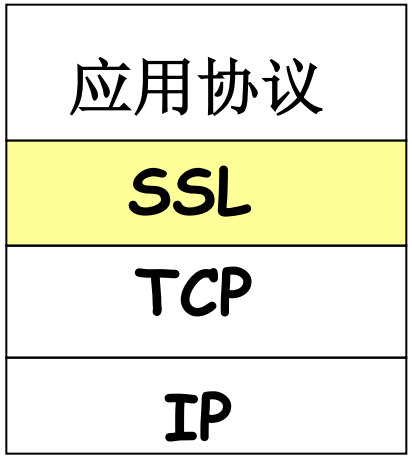
SSL概述

- Secure Socket Layer
- 为基于TCP的应用提供安全的传输通道
- 1993年由 Netscape设计
- 类似协议：TLS (RFC2246)
- 安全性支持
  - 机密性（Confidentiality）
  - 一致性（Integrity）
  - 身份认证（Authentication）

SSL的位置



普通应用



基于SSL的应用

第二章总结

重点：掌握DNS、WWW、Email的应用层协议的要点和原理、FTP和Telnet的功能及主要概念

- 客户和服务器使用请求/应答消息进行交互
  - 客户发送请求
  - 服务器返回响应
- 消息格式
  - 消息头：协议要处理的信息
  - 数据：交付给用户的信息

重要概念:

- 集中式/分布式
  - C/S vs P2P
- 传输要求：可靠/不可靠
  - 选择TCP或UDP
- 控制信息vs. 数据信息
  - 是否在同一条连接上传输
- 有状态/无状态

版权说明

- 本讲义中部分图片来源于下列教材所附讲义：
  - Jim Kurose, Keith Ross，Computer Networking: A Top Down Approach, 7th Edition，Pearson/Addison Wesley, April 2016，引用时标记为*[Kurose]*;
  - Andrew S. Tanenbaum, Computer Networks, Fourth Edition, 清华大学出版社（影印版），2004，引用时标记为*[Tanenbaum]*;
  - 谢希仁，计算机网络，第五版，电子工业出版社，2008年1月,引用时标记为*[谢];*
  - Behrouz A. Forouzan，Data Communications and Networking，Fourth Edition, McGraw-Hill Higher Education, 2007年1月，引用时标记为*[Forouzan]*

本章勘误表(1)

页码	位置	原文	更正
31	第2段		段尾增加下列文字：“其中Web客户进程和Web服务器进程之间的通信关系由四元组（1.2.3.4,12345,4.3.2.1,80）来唯一标识”
31	图2-3		去掉图中左侧第一个进程下面的“54321”以及Web客户进程下面的“64321”
33	第1段最后一句	HTTP、FTP和Telnet都使用TCP	HTTP、FTP、SMTP和Telnet都使用TCP
37	第3段第一句	DNS系统的域名地址映射信息以资源记录（Resource Records）	DNS系统的域名地址映射信息保存在资源记录（Resource Records）中

本章勘误表(2)

页码	位置	原文	更正
37	第15行		删去“规范名”之后的“即Owner Name”
37	表2-4		删去表中第4行
33	最后一行	C:\>nslookup-query=nsbupt.edu.cn	C:\>nslookup -query=ns bupt.edu.cn
39	表2-6中第一行	授权响应	权威响应
39	表2-6中第四行	授权字段	权威记录字段
40	图2-12	TCP链接	TCP连接
45	最后一行	主要的消息头如表2-9所示	主要的消息头如表2-8所示
51	倒数第8行	POP3	邮局协议（Post Office Protocol，POP3）

本章勘误表(3)

页码	位置	原文	更正
54	表中第二行	字节序列	字节流
60	图2-33(a)	Server repies FTP service available	Server replies FTP service available
60	图2-33(g)	Acks recewed OK	Acks received OK
63	2.6标题下第7行	Telnt	Telnet