



传输层概述、UDP与DNS

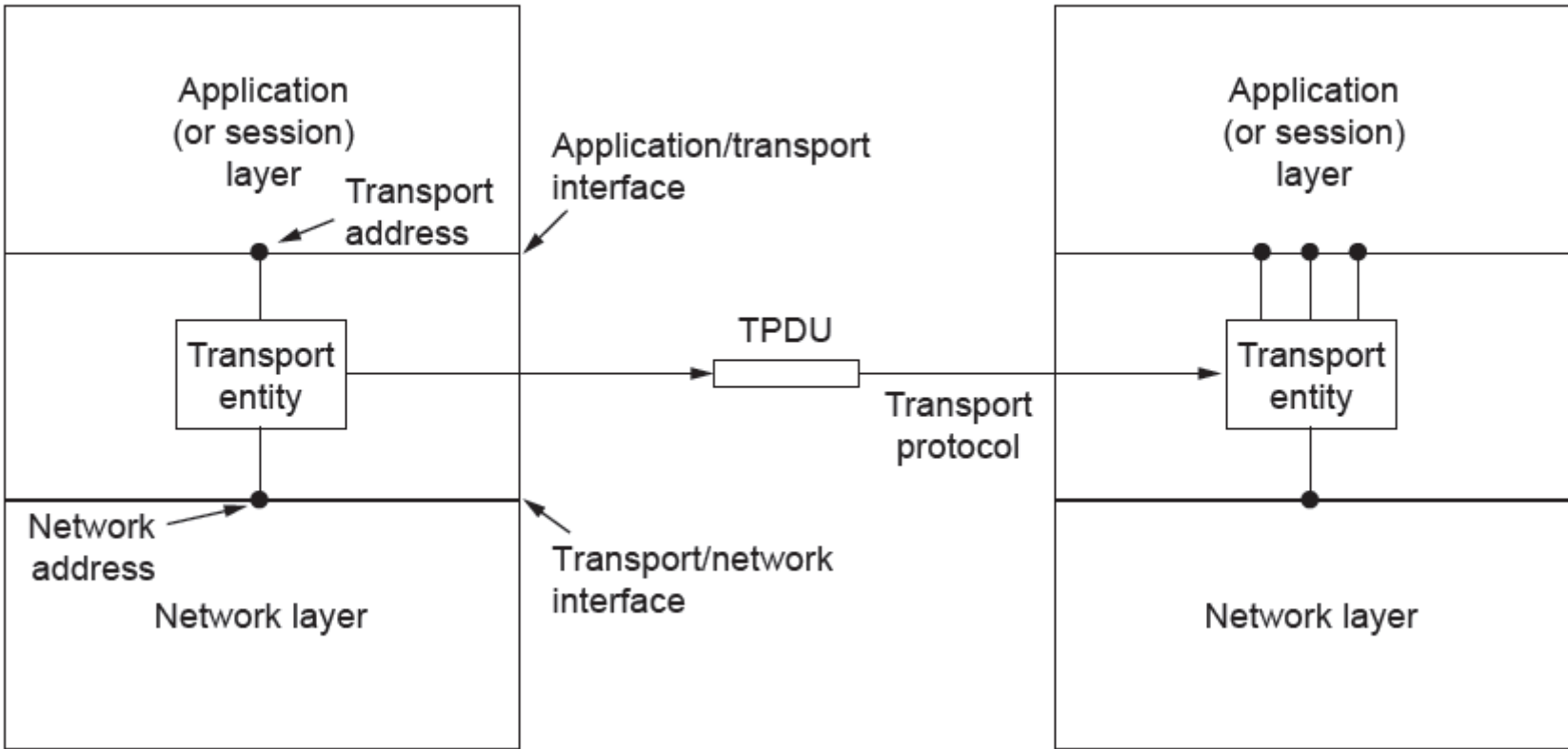
刘志敏

liuzm@pku.edu.cn



提纲

- 传输层的功能及服务
- 多路复用与并发操作
- 传输层协议
- UDP协议
- DNS系统：从应用层穿越整个协议栈



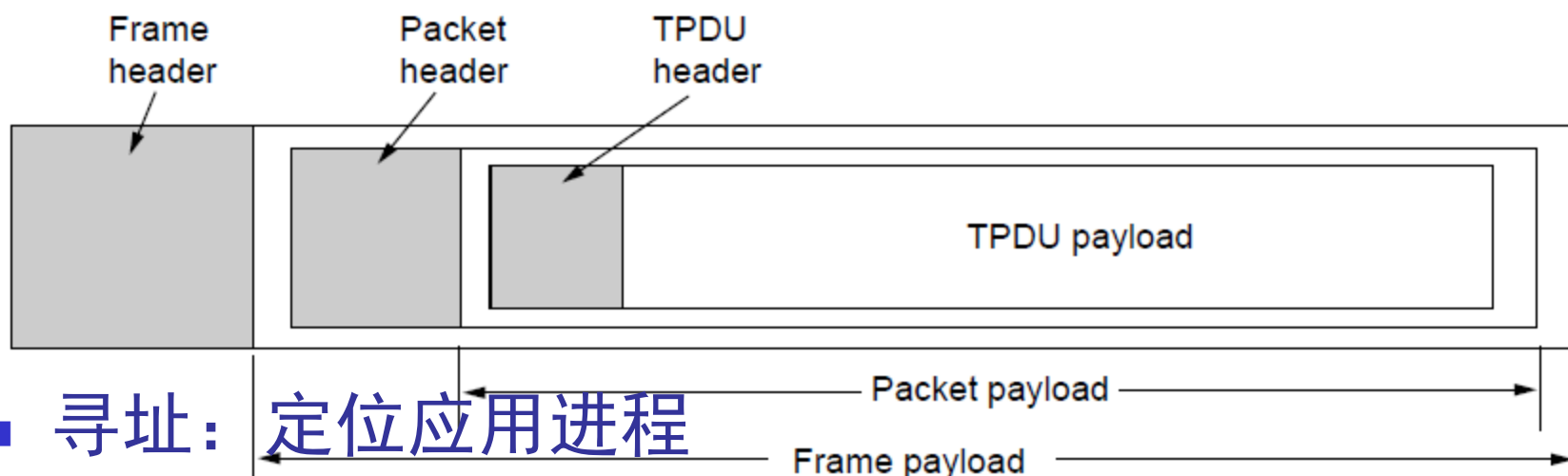
网络层：分组交换，数据报方式存在不可靠（丢失、乱序、延迟）

传输层：为应用层提供连接或无连接的服务，差错及流量控制

应用层：各类业务，或要求保证可靠性或实时性

传输层的功能及服务

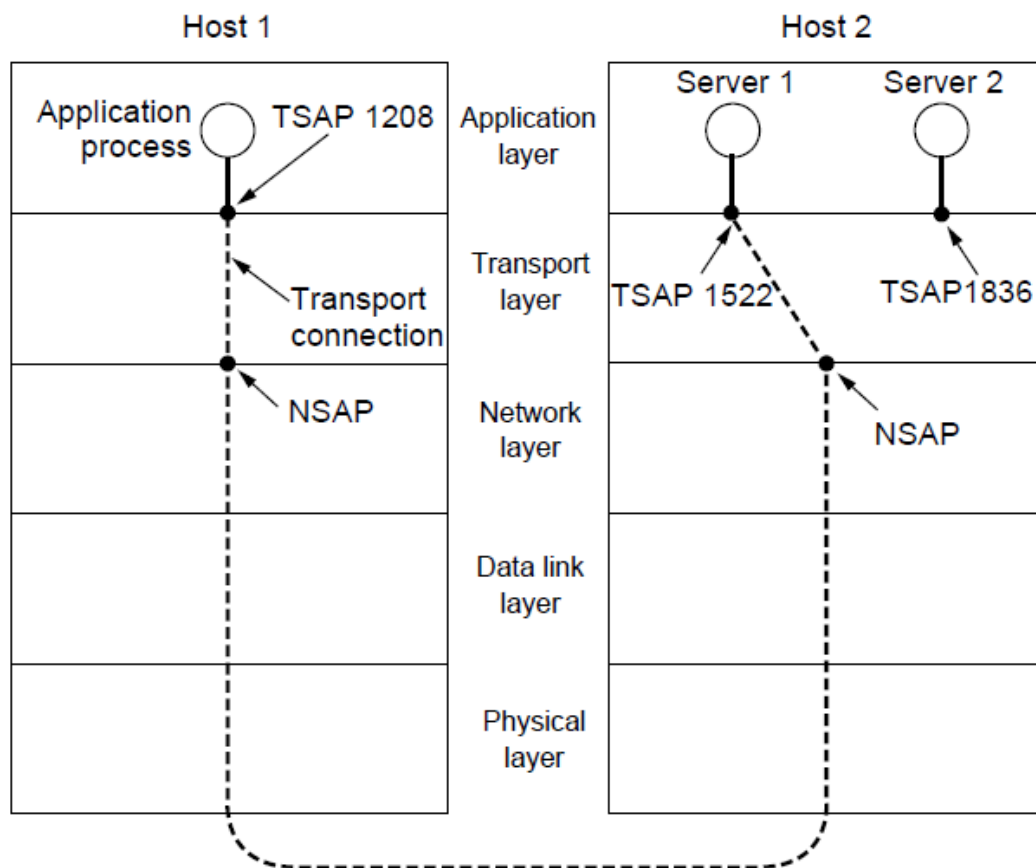
报文段TPDUs, 分组及帧



- 寻址：定位应用进程
- 复用：在一个主机上支持多个任务
- 连接建立与释放：提供面向连接的服务
- 差错控制与流量控制
- 拥塞控制：辅助网络层解决拥塞问题
- 崩溃恢复

寻址

每个主机仅有一个NSAP但有多多个应用，需要标识TSAP；
不同TSAP的端口不同，应用程序接收不同端口上的消息

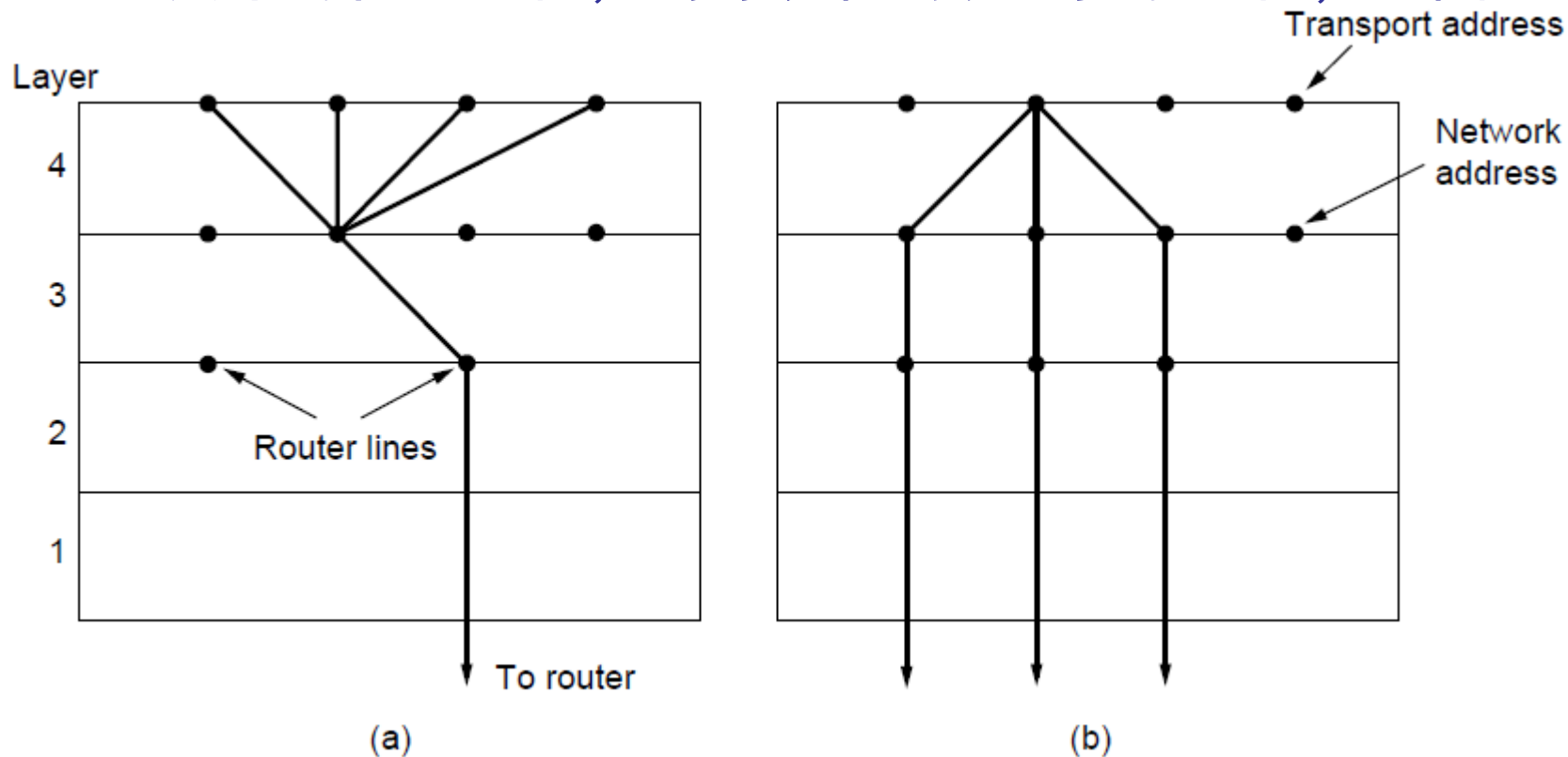


TSAP, NSAP和传输连接

多路复用

复用：主机只有一个地址，但有多多个应用；这些应用都使用一个地址；如图a，4个独立的连接公用一个地址

反向复用：若一个主机有多条网络路径，而一个应用需求高于其中的任一路径，则可以轮询使用多条路径，如图b





多路复用与并发操作

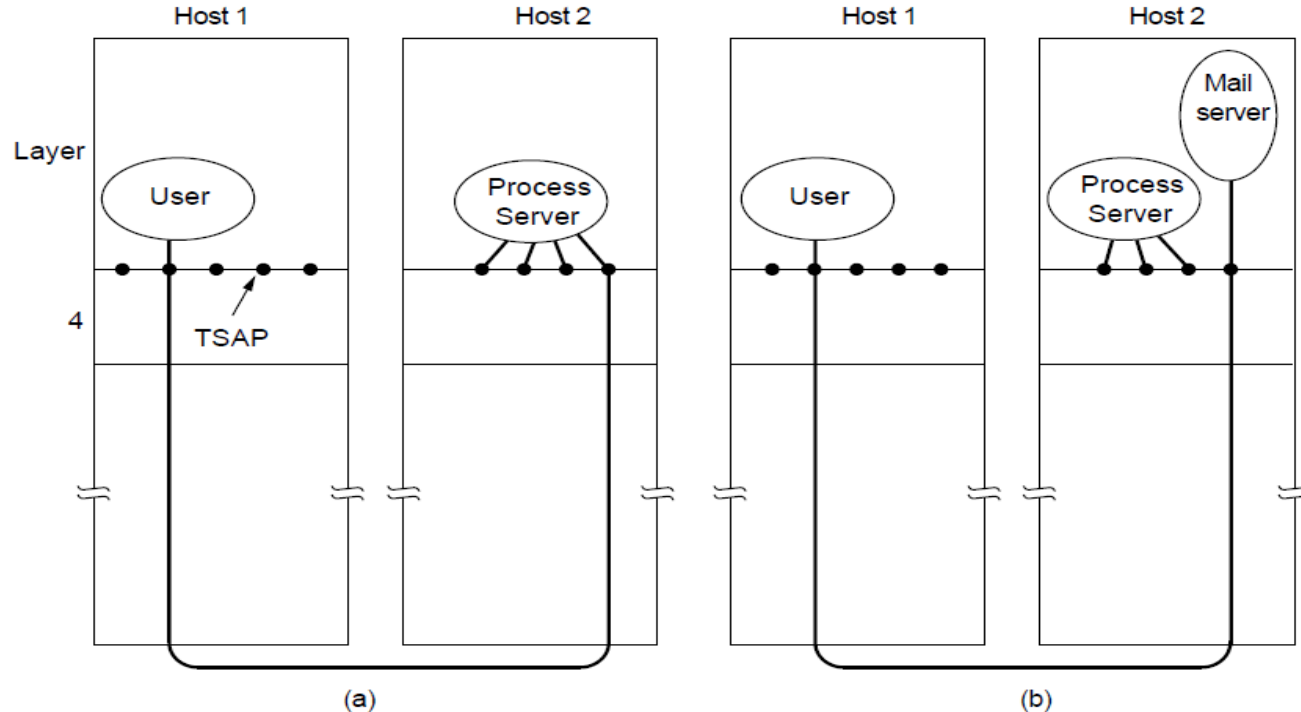
- 服务器如何为多个用户提供服务？

方案1：服务器在固定端口上监听，不同端口提供不同的服务，例如/etc/services

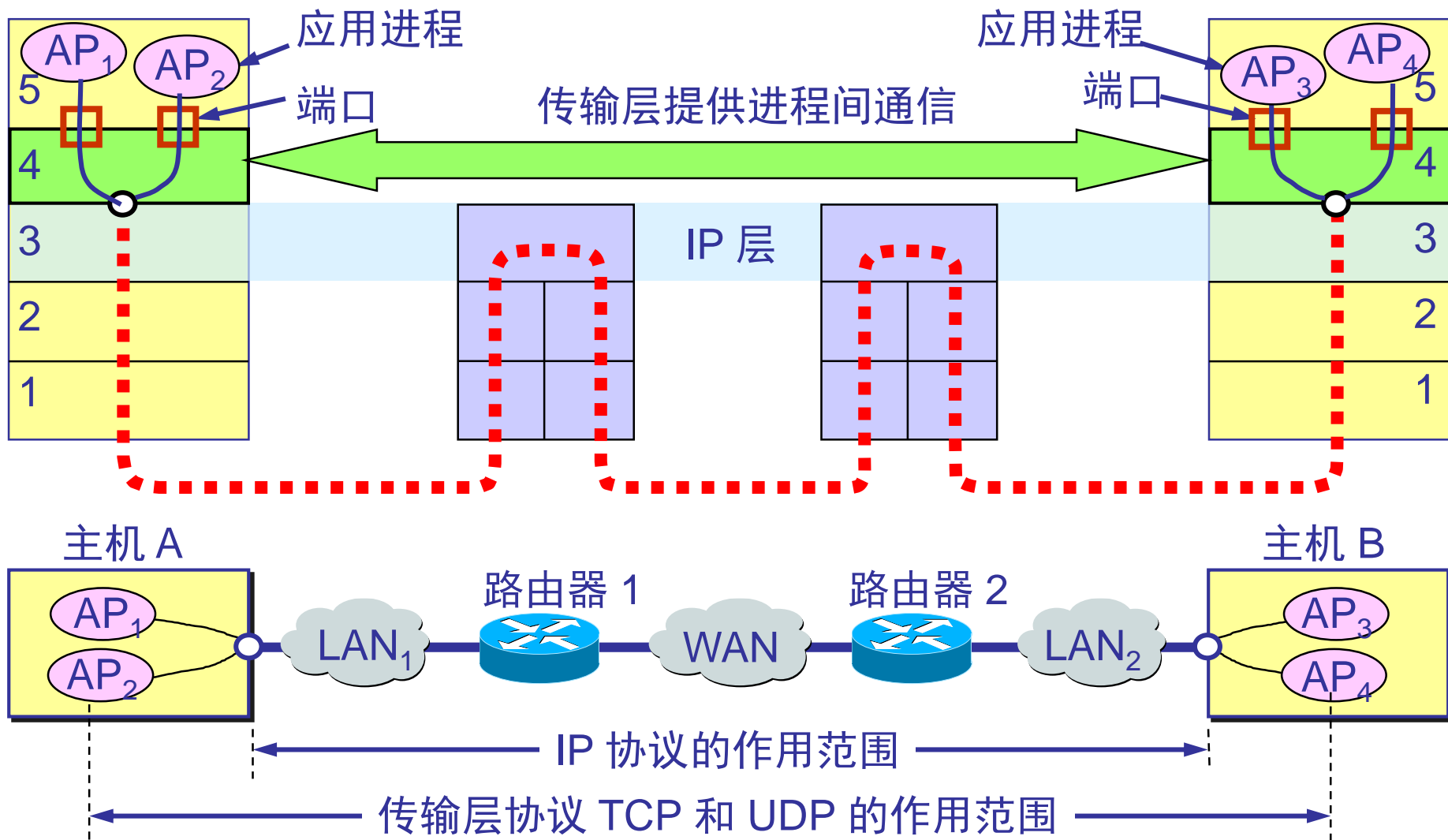
方案2：端口映射器；用户与端口映射器（其端口已知）建立连接，发送消息指定其所需的服务器名，得到服务器名对应的TSAP；之后，再与所需的服务器建立连接；类似于查号台

多路复用与并发操作

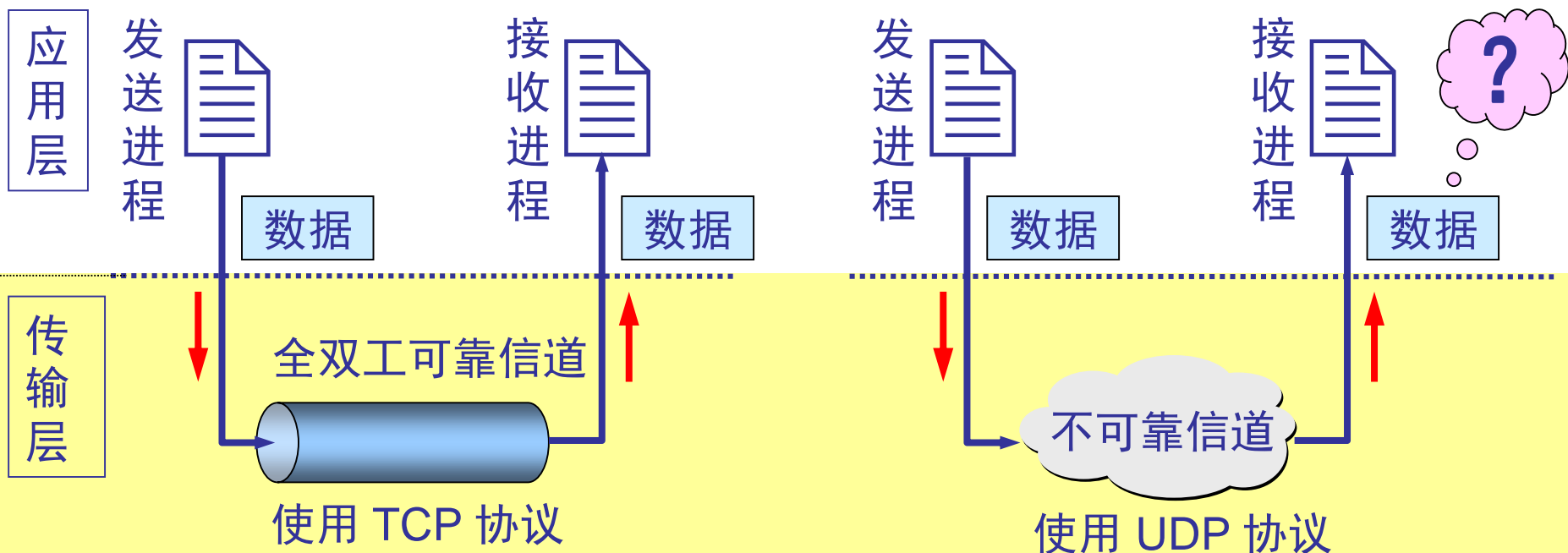
- 服务器如何有效地提供多个服务？
 - a: 一个进程服务器同时监听一组端口，等待连接请求；如 LINUX 系统的进程服务器 `inetd`
 - b: 当有请求时，进程服务器才启动新的服务器，而进程服务器继续监听请求——按需创建服务，节省服务器资源



传输层协议

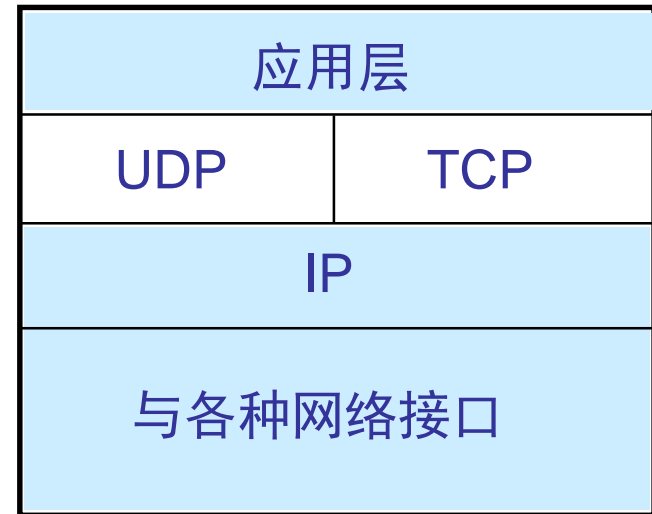


传输层向上提供可靠和不可靠逻辑信道



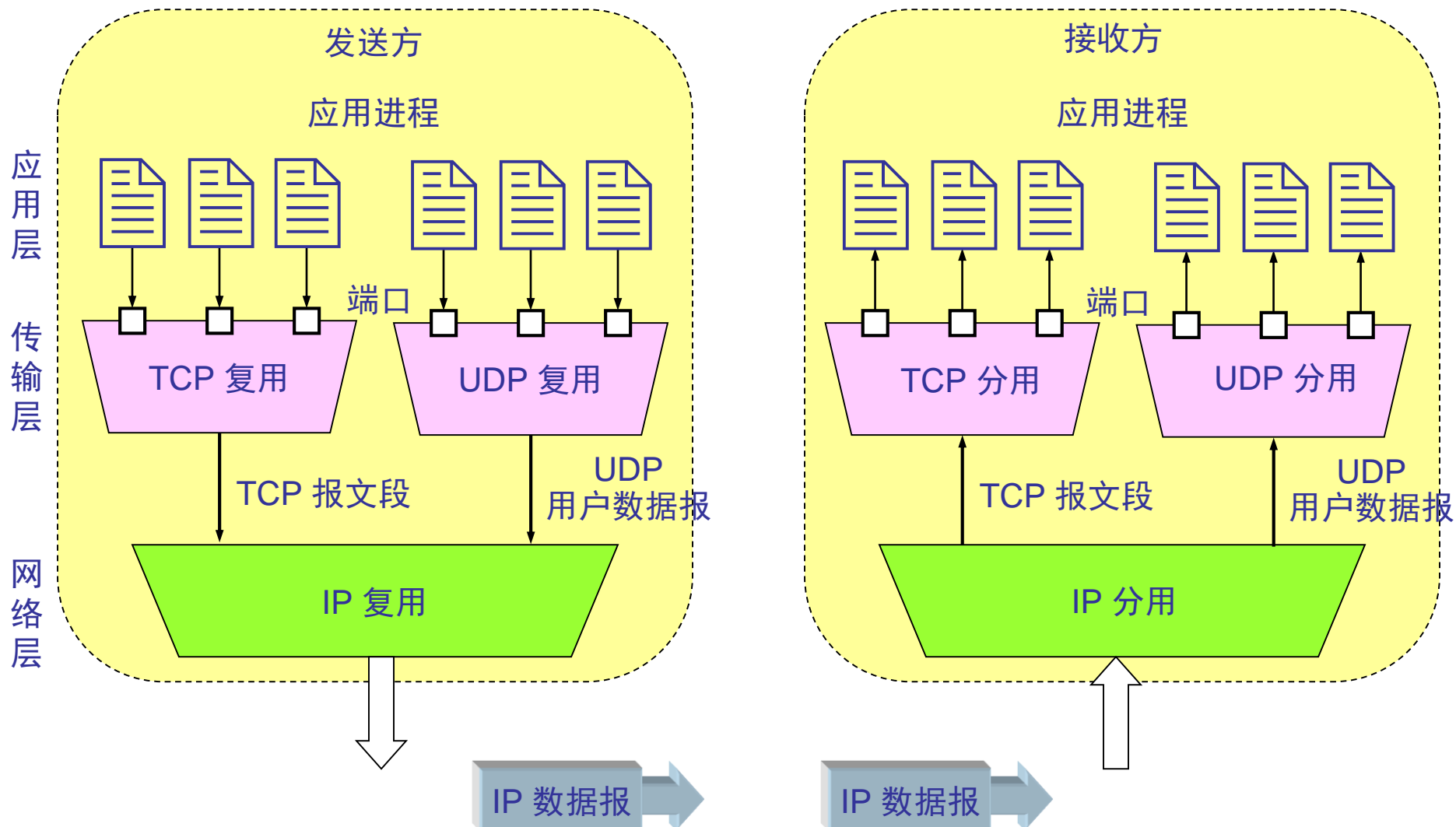
UDP与TCP

- UDP 在传送数据之前不需要建立连接。传输层在收到 UDP 报文后，也不发送确认。
- TCP 提供面向连接的可靠的服务。TCP增加了许多开销，如协议数据单元的首部更长，采用拥塞窗口控制、超时重传机制等，算法更复杂。



传输层

端口在进程间通信的作用：标志进程





端口

- 端口号占 16 bit 。
- 两类端口：一类是熟知端口，其数值一般为 0~1023，另一类是一般端口，用来随时分配给请求通信的进程。

应用程序	FTP	TELNET	SMTP	DNS	TFTF	HTTP	SNMP
熟知端口	21	23	25	53	69	80	161

- 提供的服务及其端口号：在Unix系统中
/etc/services，在Windows 系统中
[/windows/system32/drivers/etc/services](#)



套接字(socket)

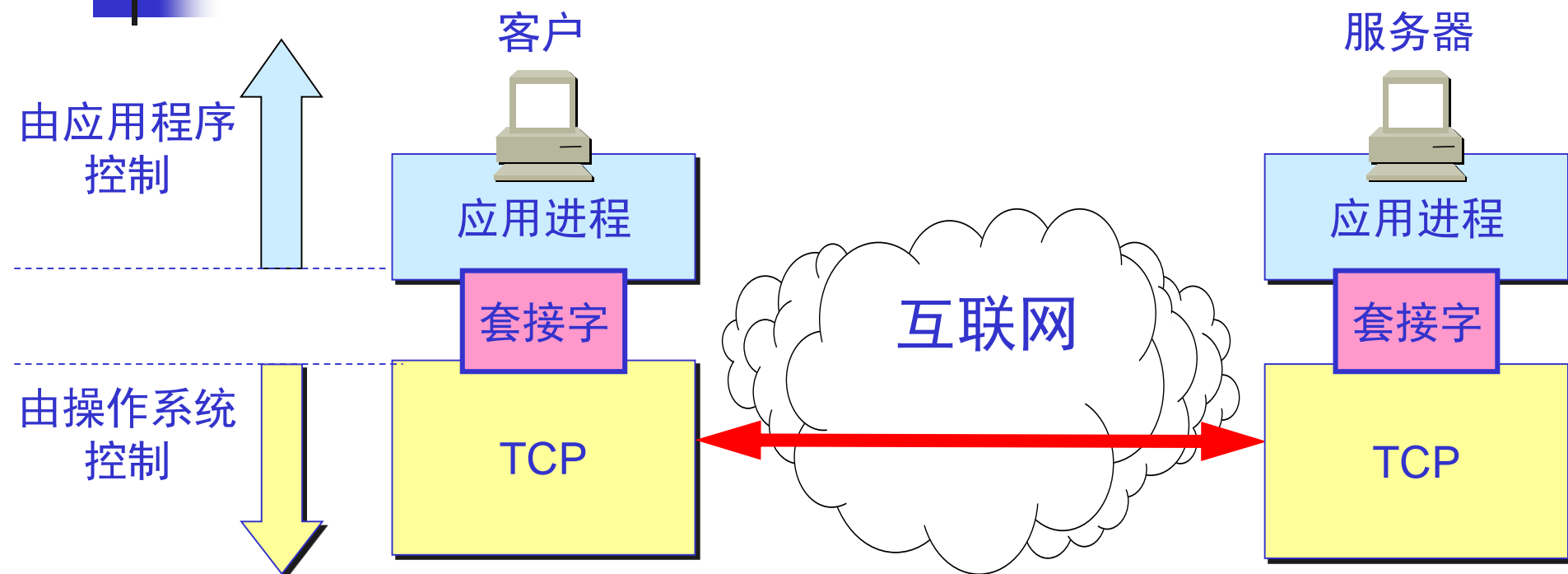
- IP地址与端口组成 socket或套接字，网络编程时使用。

套接字 socket = (IP地址: 端口号)

- 每一条 **TCP** 连接唯一地被通信两端的两个端点（即两个套接字）所确定。即：

TCP 连接 ::= {socket1, socket2}
= {(IP1: port1), (IP2: port2)}

进程通过套接字接入网络



■ 问题：

- 采用TCP或UDP，要考虑哪些因素？
- 采用socket编程，主要步骤？服务器、客户机有何不同？

调用 socket 创建套接字

操作系统

套接字描述符表
(每一个进程一个描述符)

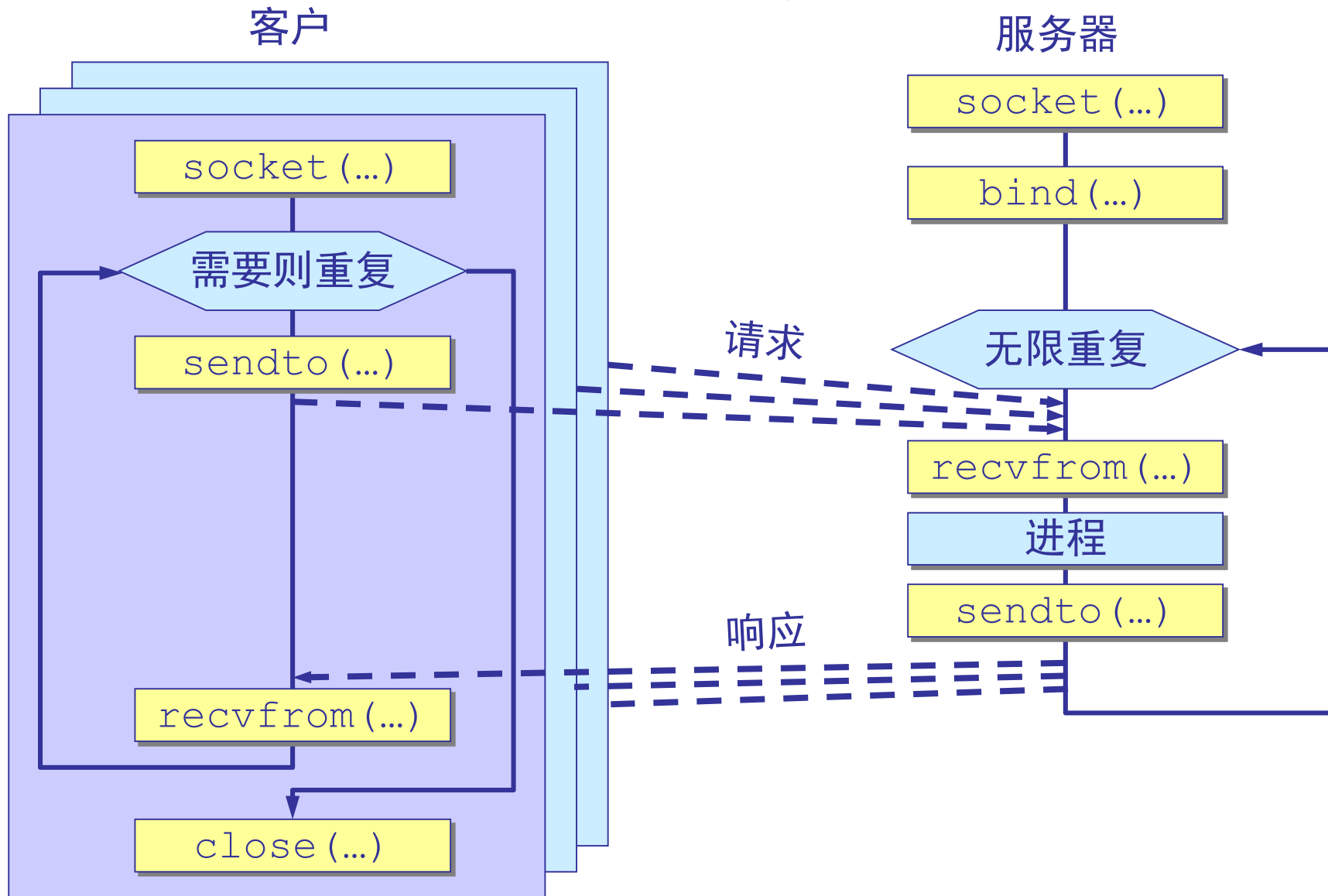
0:	
1:	
2:	
3:	
4:	
	⋮

套接字的数据结构

协议族: PF_INET
服务: SOCK_STREAM
本地 IP 地址:
远地 IP 地址:
本地端口:
远地端口:
⋮

用套接字实现进程间通信

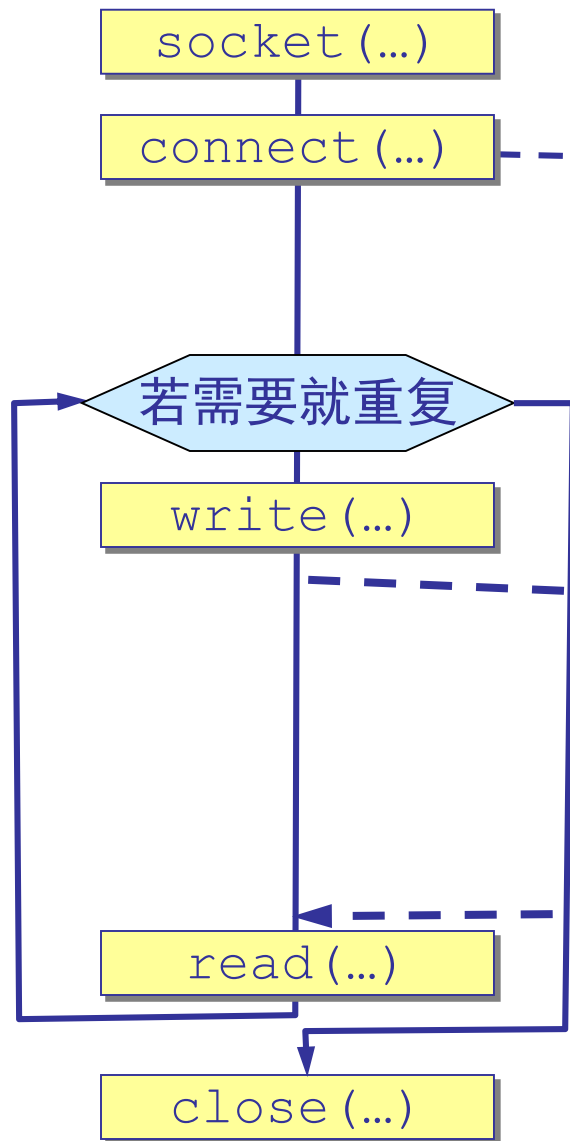
无连接服务



用套接字实现进程间通信

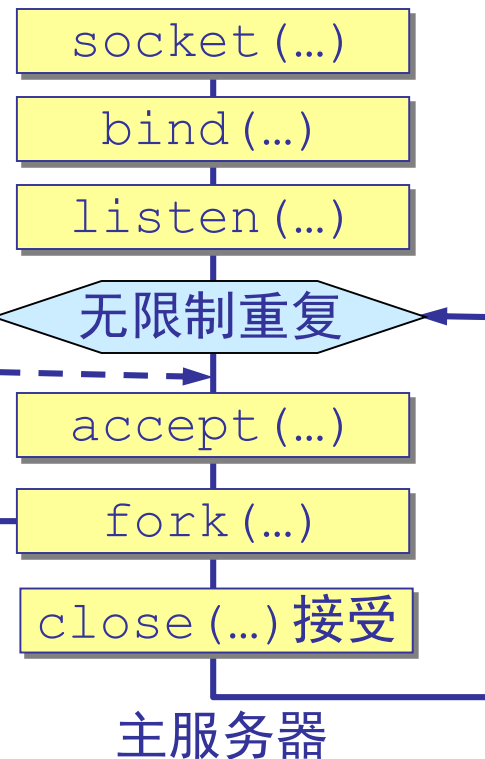
面向连接的服务

客户



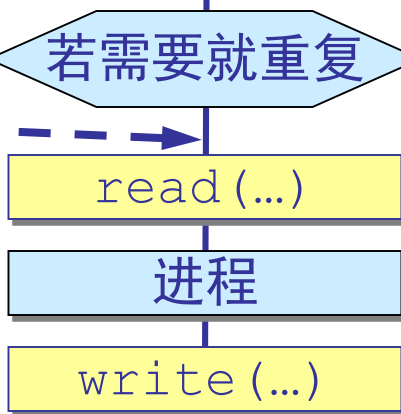
连接请求

服务器

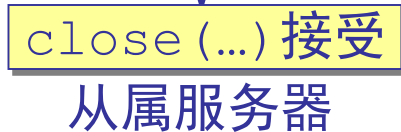


主服务器

请求



响应



从属服务器

Berkeley Sockets (2)

Primitive	Meaning
SOCKET	Create a new communication end point
BIND	Associate a local address with a socket
LISTEN	Announce willingness to accept connections; give queue size
ACCEPT	Passively establish an incoming connection
CONNECT	Actively attempt to establish a connection
SEND	Send some data over the connection
RECEIVE	Receive some data from the connection
CLOSE	Release the connection

The socket primitives for TCP

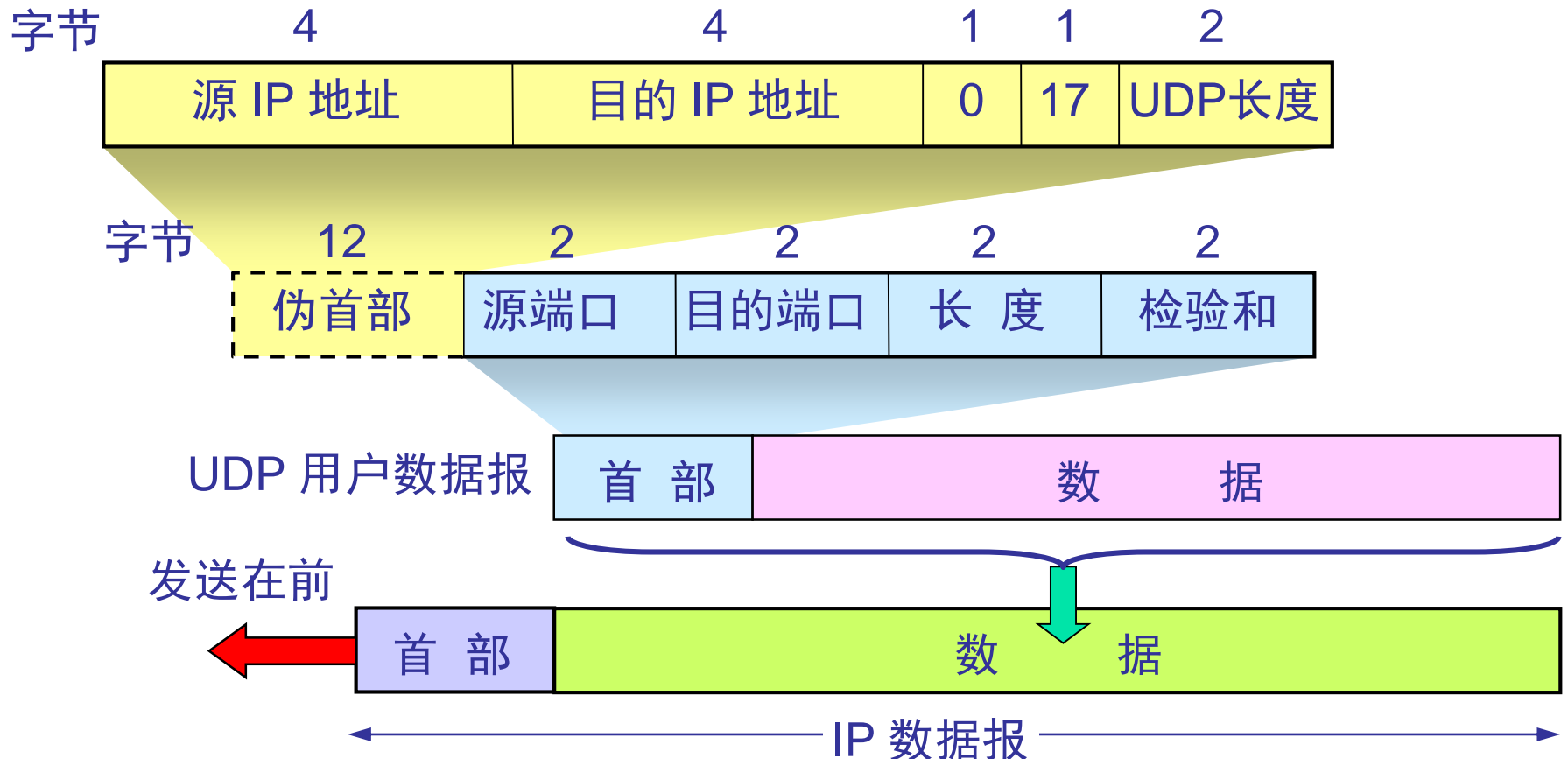


用户数据报协议 UDP

- UDP在IP数据报上增加端口和差错检测功能；没有流量控制、拥塞控制、重传机制
- UDP只提供不可靠交付，优点是：
 - 不需要建立连接，简单、易于实现
 - 只有8个字节的首部开销
 - 网络拥塞时也不降低发送速率，适于传输实时业务，短时突发数据
- 基于UDP的应用：DNS、TFTP、RIP、DHCP、SNMP、NFS、RTP、IGMP

UDP用户数据报格式

检验和：在计算检验和时，临时把“伪首部”和UDP用户数据报连接在一起，伪首部仅仅是为了计算检验和





提纲

- 传输层的功能及服务
- 多路复用与并发操作
- 传输层协议
- UDP协议
- DNS系统：从应用层穿越整个协议栈



DNS系统

- IP地址不宜记忆，开始用hosts.txt，但需要集中，很难保证唯一；1983年发明DNS，1998年ICANN管理域名，定义250个顶级域名，申请二级域名付费使用
 - 域名抢注、域名拍卖（例如：.tv \$50万）
- 域名便于理解并记忆
 - 例如：各大学网站的域名 www.pku.edu.cn, 美国大学的网站 www.mit.edu
- 当更换一个服务器时，其IP地址可能需要改变（如因位置的改变），但域名不变，方便用户使用
- 可以为服务器及主机定义别名：一个主机有多个名字
- 负载分配：多个服务器共用一个域名
 - 例如各大门户网站、搜索引擎网站



DNS系统

- **域名系统** DNS (Domain Name System)

- 域名是主机的别名，采用层次结构

... . 三级域名 . 二级域名 . 顶级域名

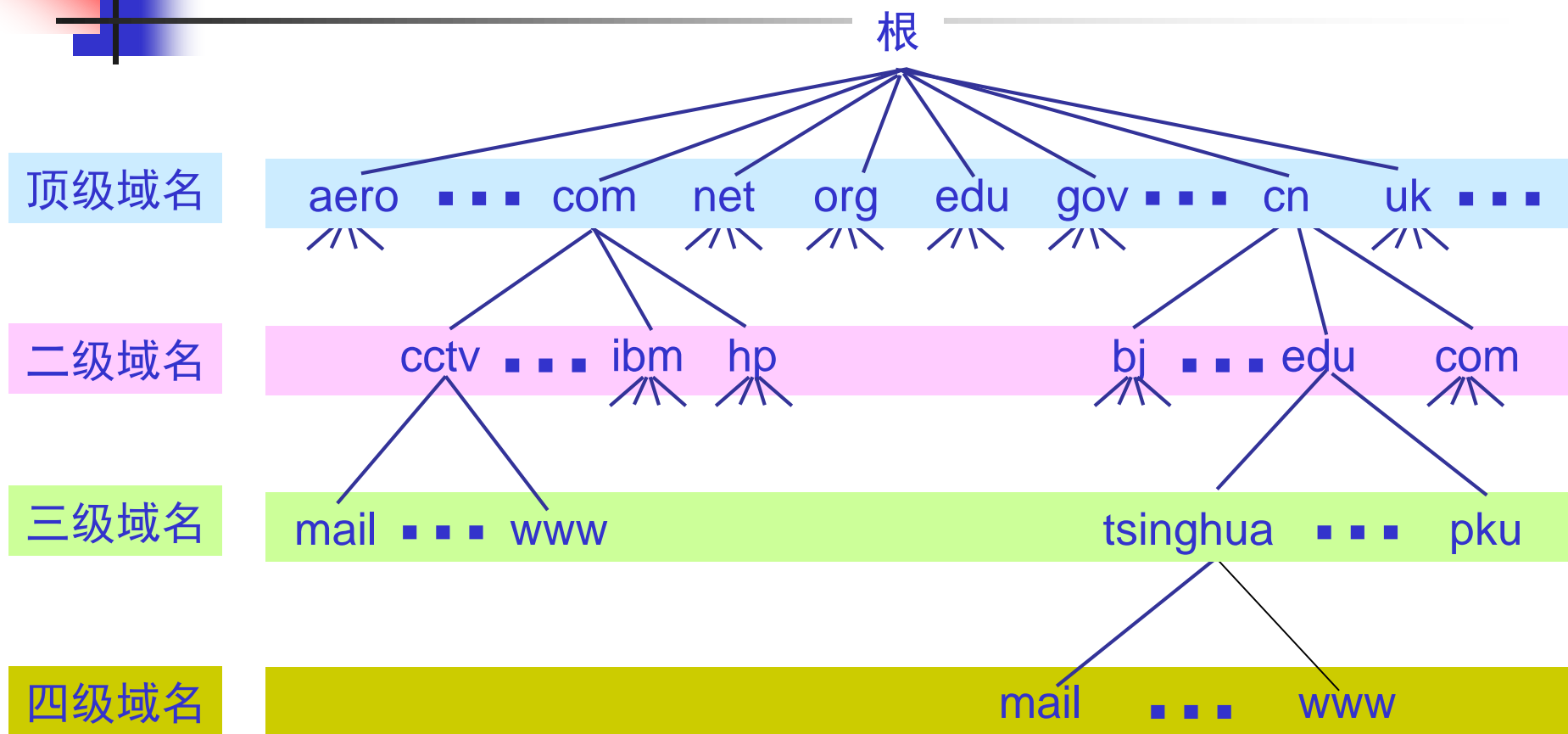
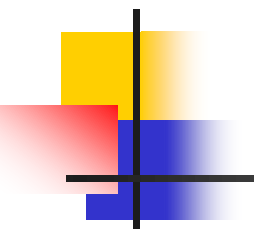
如 www.pku.edu.cn

邮件服务器的命名，邮件采用用户名@邮件服务器

- **域名服务器**负责域名到 IP 地址的解析

- 采用**分布式**结构，由若干个域名服务器负责名字解析

互联网的域名空间





顶级域名 TLD(Top Level Domain)

(1) 国家顶级域名 nTLD: 如.cn, .us, .uk等

(2) 通用顶级域名 gTLD: 如

.com (公司和企业)

.net (网络服务机构)

.org (非赢利性组织)

.edu (美国专用的教育机构)

.gov (美国专用的政府部门)

.mil (美国专用的军事部门)

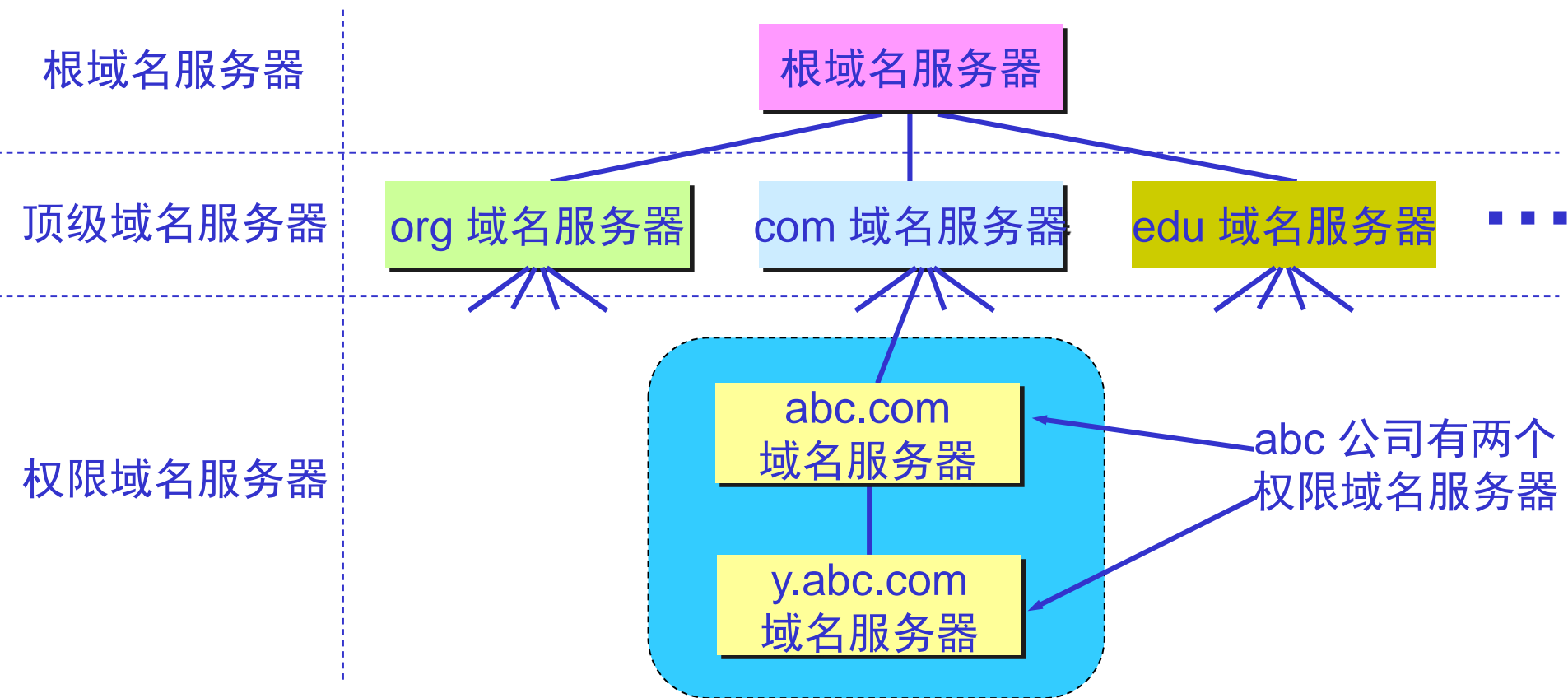
.int (国际组织)



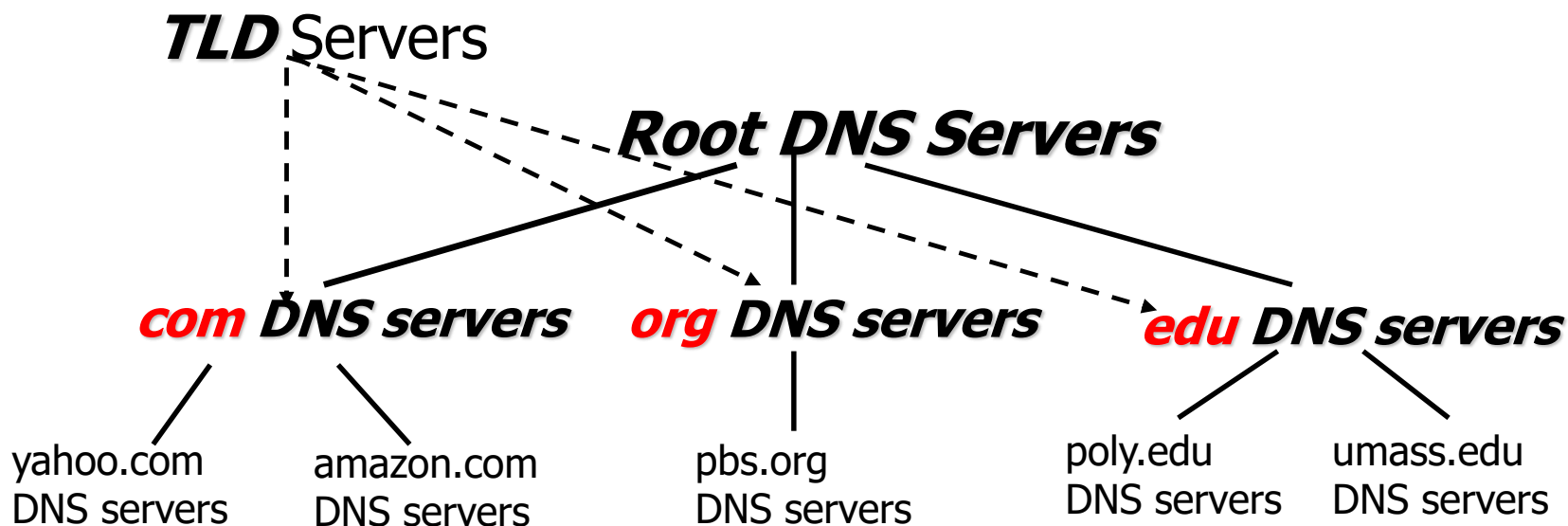
DNS 如何工作？

- DNS为何采用分布式结构？因集中式结构存在如下问题：
 - 单点失效问题；
 - 性能可扩展问题，大量请求使DNS服务器成为瓶颈
 - 服务器与客户机距离远，引入延迟较大；
 - 单个数据库的容量巨大，频繁更新不便于管理
- DNS 采用分布式层状结构， DNS分类
 - *Root* DNS 根服务器
 - *Top-Level* 顶层DNS服务器
 - *Authoritative* 授权DNS服务器
 - 本地服务器

DNS 域名服务器的层状结构



分布式层状结构的数据库



为了提供更好的可扩展性，数量巨大的DNS服务器组成了遍布全球的一个树型结构。一个DNS服务器并没有包含所有的域名到IP地址的映射。全球13个rootDNS，下设TLD



DNS 如何工作？

■ 顶级域名服务器（即TLD服务器）

- 负责顶级域名 *com*, *org*, *net*, *edu*等，及国家域名如cn, uk, fr, ca, jp的解析
- 当收到DNS查询请求时，就给出相应的回答

■ 权限域名服务器

- 负责一个区的域名服务器；保存DNS记录；
- 当一个权限DNS不能给出最后的查询回答时，就会响应一个下一步应当找查询的权限DNS

■ 本地域名服务器

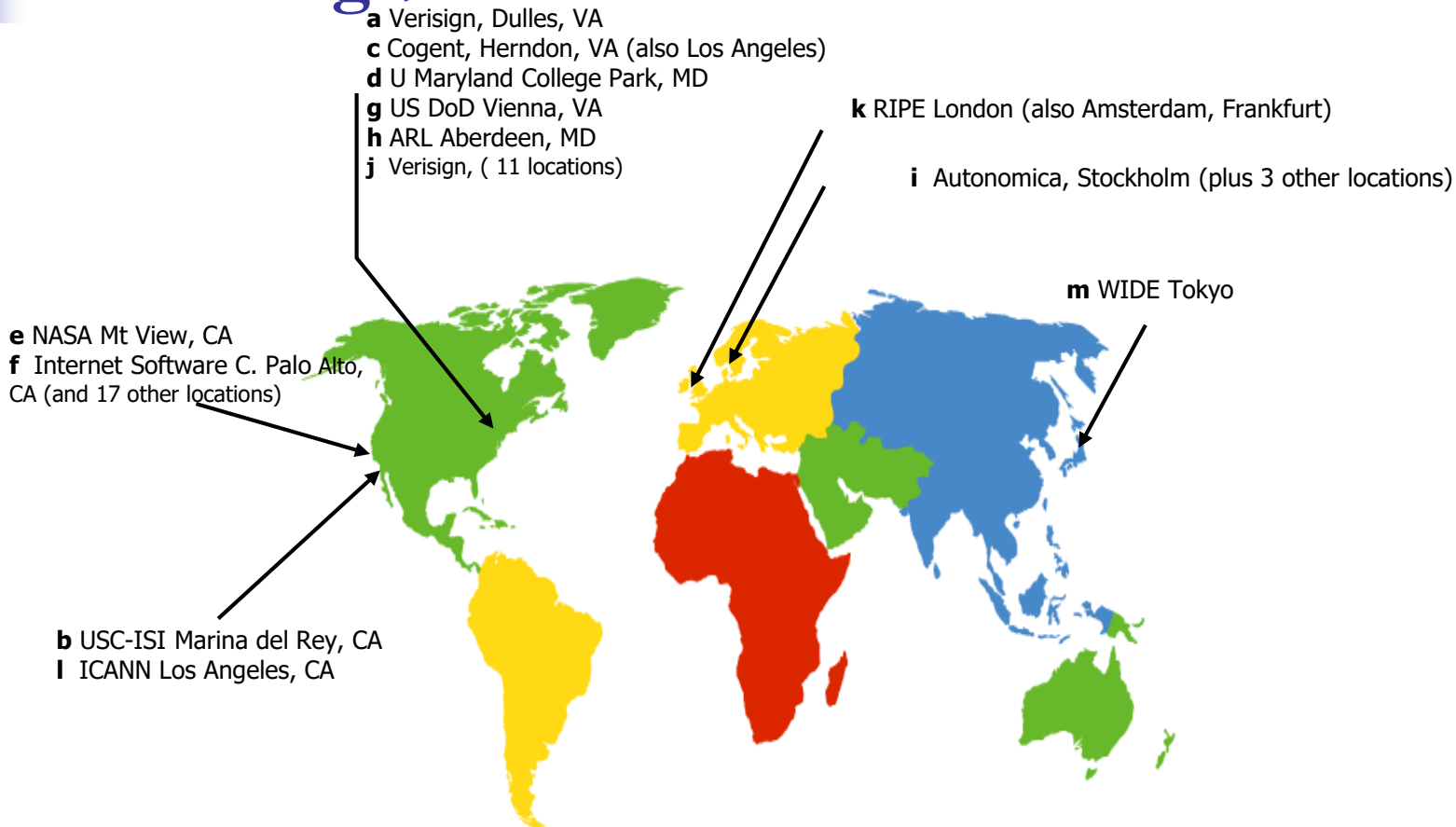
- 当一个主机发出DNS查询请求时，首先发送给本地DNS
- 每个ISP，都可以拥有一个本地DNS，称为缺省的DNS
- 实质上是一个代理，负责转发查询请求给分布式DNS



DNS 如何工作？

- 根域名服务器：知道所有的顶级域名服务器及其IP地址
- 互联网上共有13 个根DNS，其名字是英文字母a~m，域名分别是 a.rootservers.net, b.rootservers.net, ... m.rootservers.net
- 本地DNS在对域名进行解析时若无法解析，就首先求助于根根域名服务器。
- 根域名服务器并不直接把域名转换成IP地址，而是在使用迭代查询时，把下一步要找的顶级域名服务器的IP地址告知本地DNS

Root DNS 在全球的分布 (<http://www.root-servers.org/>)



- 根域名服务器并不直接把域名转换成 IP 地址
- 在使用迭代查询时，根域名服务器把下一步应当找的顶级域名服务器的 IP 地址告诉本地域名服务器



提高DNS的可靠性

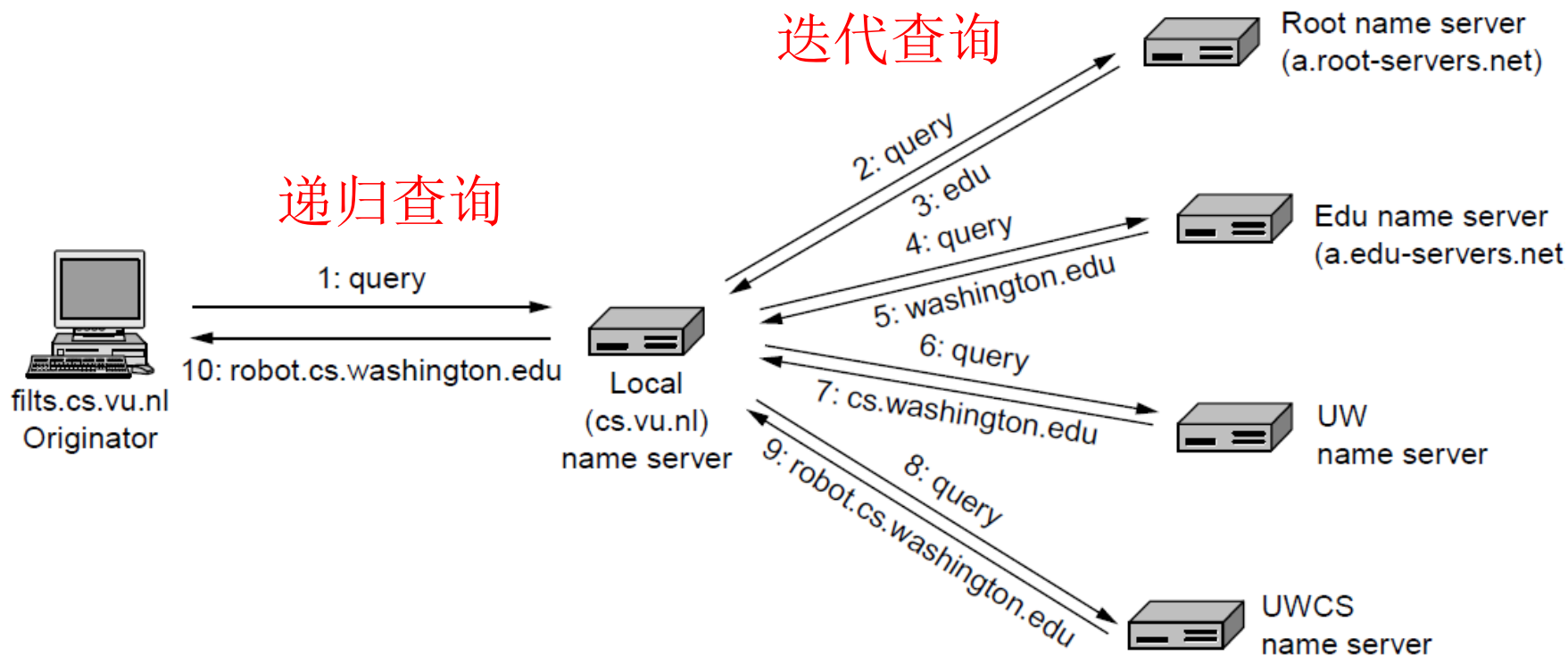
- 设置主域名服务器、辅助域名服务器
- 当主域名服务器出故障时，辅助域名服务器可以保证 DNS 的查询工作不会中断
- 主域名服务器定期把数据复制到辅助域名服务器中，而更改数据只在主域名服务器中进行。这样就保证了数据的一致性。



域名解析过程

- 主机向本地DNS的查询一般是**递归查询**。如果本地DNS无法解析域名，则以DNS 客户的身份，向其根DNS继续发出查询请求报文。
 - **递归查询**：本地DNS代替主机处理域名解析工作，直到返回完整的答案。
- 本地DNS向根DNS的查询通常是**迭代查询**。当根DNS收到本地DNS的迭代查询请求报文时，给出IP地址或“下一步要查询的域名服务器”
 - **迭代查询**：根DNS的查询只是反馈部分答案并移动到下一次查询过程，而由本地DNS继续发起下一次请求

域名解析过程



解析器查询一个远程名字的步骤

域名解析：增加了延迟及网络流量；

如何降低？本地DNS的DNS缓存



DNS缓存

- 每个域名服务器维护一个高速缓存，存放最近用过的名字以及从何处获得名字映射信息的记录
- 减轻根域名服务器的负载，使DNS查询请求和应答报文的数量大为减少
- 为保持高速缓存中的内容正确，DNS为每项内容设置计时器，并处理超过时限的项（例如，每个项目只存放两天）
- 当权威域名服务器回答一个查询请求时，在响应中都指明绑定有效的时间值。
 - 增加时间值可减少网络开销，而减少时间值可提高域名转换的准确性



DNS 记录与报文

DNS: 分布式数据库存储的资源记录 (RR)

(name, value, type, ttl)

- Type=**A**
 - name 为主机名
 - Value 为 IP 地址
- Type=**NS**
 - name 为一个域名
 - Value 为该域的授权服务器的域名
- Type=**CNAME**
 - name 为别名
 - value 为规范化名字

例如: Name=www.ibm.com
Value=servereast.backup2.ibm.com
- Type=**MX**
 - Name 为邮件服务器的别名
 - Value 为规范化名字



DNS报文传输

- DNS的查询及响应，使用UDP， 端口为53
 - 包括用户主机以及域名服务器
- DNS客户机未收到响应，则重新查询；若仍然失败，则向另一个域名服务器发送查询
 - 域名服务器采用主备方式
- DNS的查询报文含有16位的标识符，响应报文含有对应的标识符

DNS 协议，消息

DNS协议的消息有两种，**query**和**reply**，两者的格式相同

头部（12字节）

- ❑ **标识符: 16 位**，由客户机确定，服务器返回响应时以相同的标识
- ❑ **标志: 16位**
 - ❑ 1位: **query**(0)/**reply**(1)
 - ❑ 1位: 授权**DNS**服务器(1)
 - ❑ 1位: 递归查询(1)
 - ❑ 1位: 可以递归查询(1)(在reply中设置)

identification	flags
number of questions	number of answer RRs
number of authority RRs	number of additional RRs
questions (variable number of questions)	
answers (variable number of resource records)	
authority (variable number of resource records)	
additional information (variable number of resource records)	

↑
12 bytes
↓

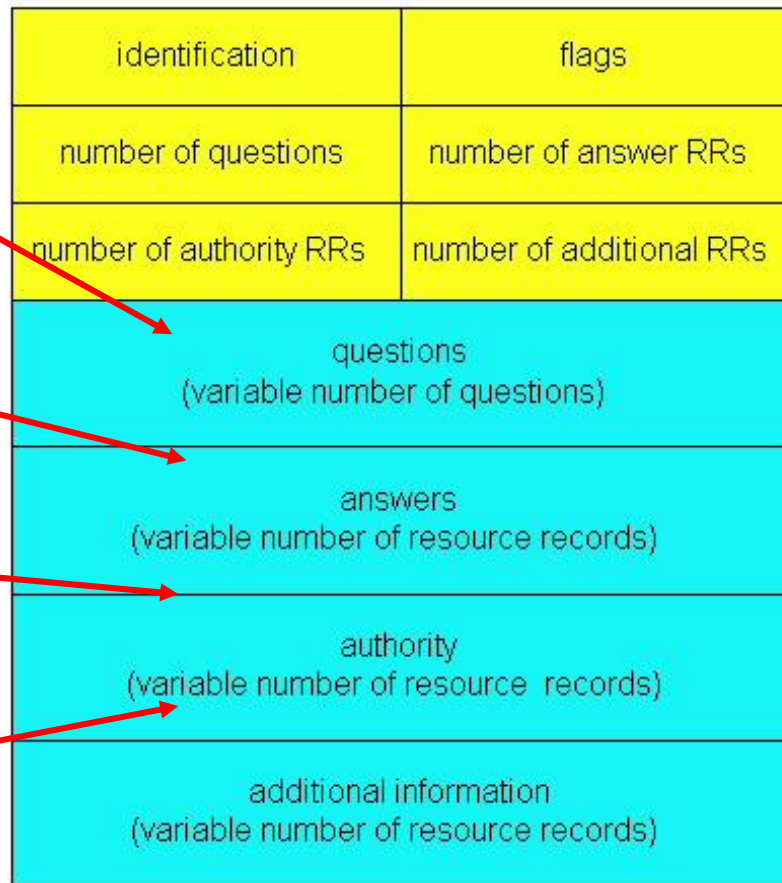
DNS 协议, 消息

问题：一个 **name** 域，为待查询的域名；**type** 域指示希望查询的类型

查询结果：**RR**，可以有多个结果

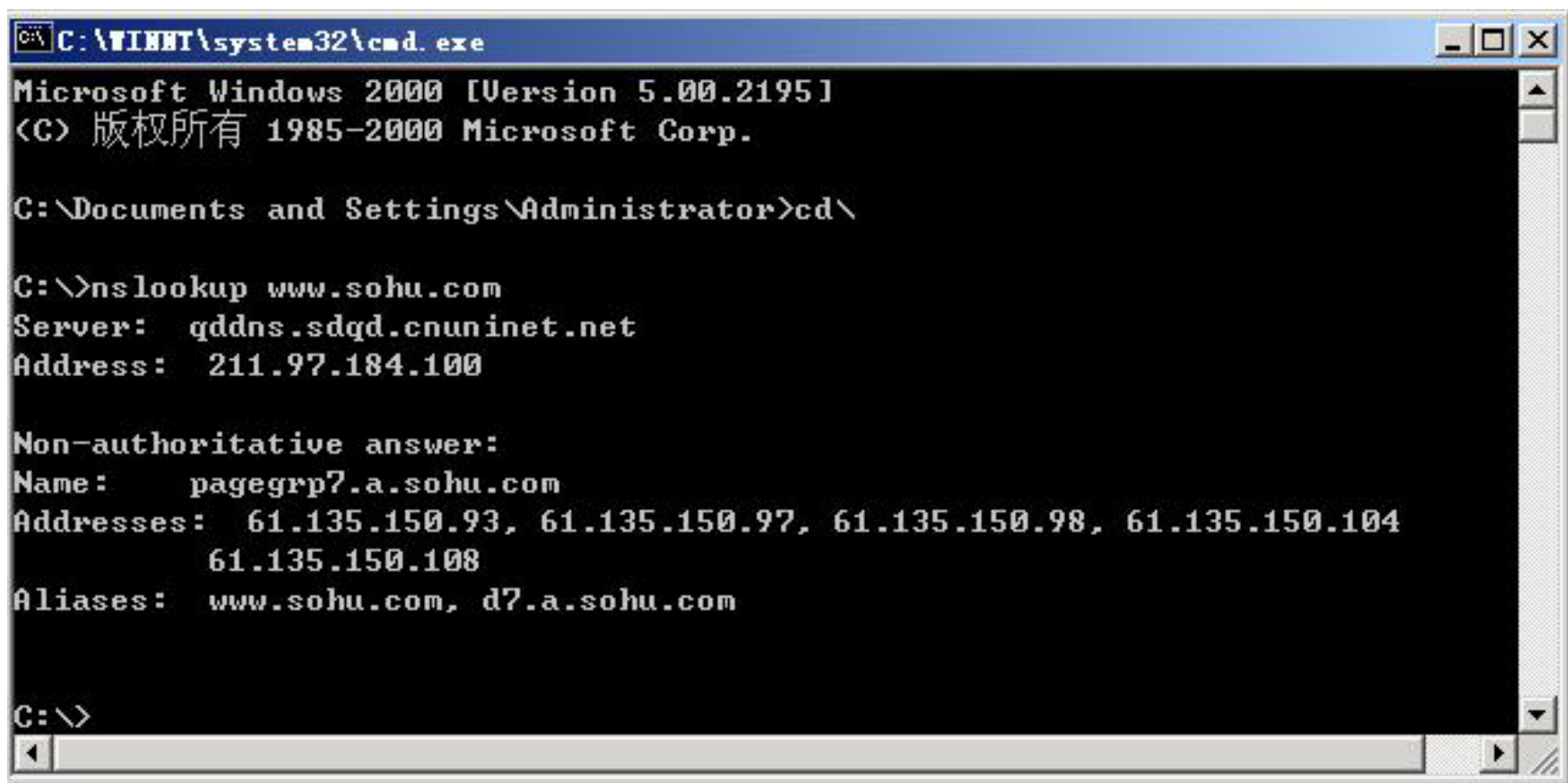
授权部分：包含其他授权 **DNS** 服务器的信息

附加信息（资源记录的变量数）



使用 *nslookup* 命令

- Nslookup 是一个实现域名解析的工具，可以监测网络中DNS服务器是否正确



```
C:\WINNT\system32\cmd.exe
Microsoft Windows 2000 [Version 5.00.2195]
(C) 版权所有 1985-2000 Microsoft Corp.

C:\Documents and Settings\Administrator>cd\

C:\>nslookup www.sohu.com
Server:  qddns.sdqd.cnuninet.net
Address:  211.97.184.100

Non-authoritative answer:
Name:     pagegrp7.a.sohu.com
Addresses: 61.135.150.93, 61.135.150.97, 61.135.150.98, 61.135.150.104
          61.135.150.108
Aliases:  www.sohu.com, d7.a.sohu.com

C:\>
```



在DNS数据库中插入记录

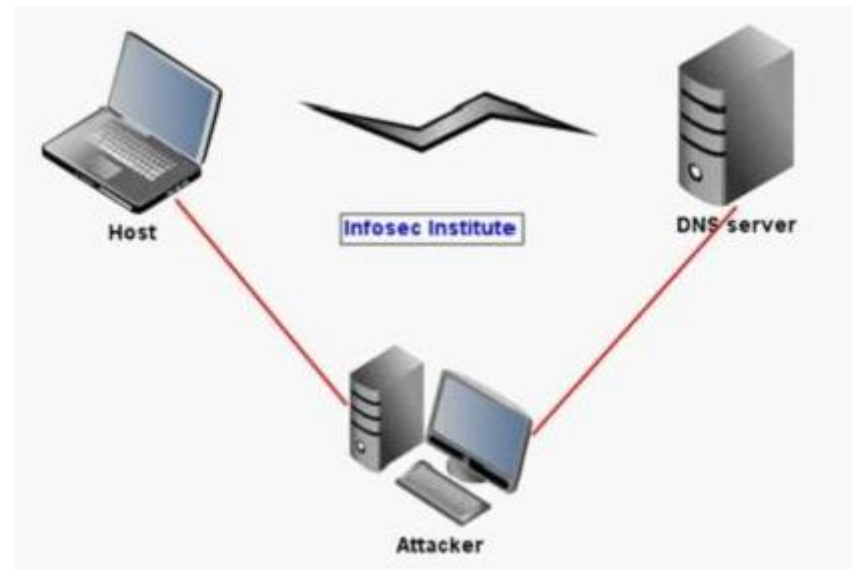
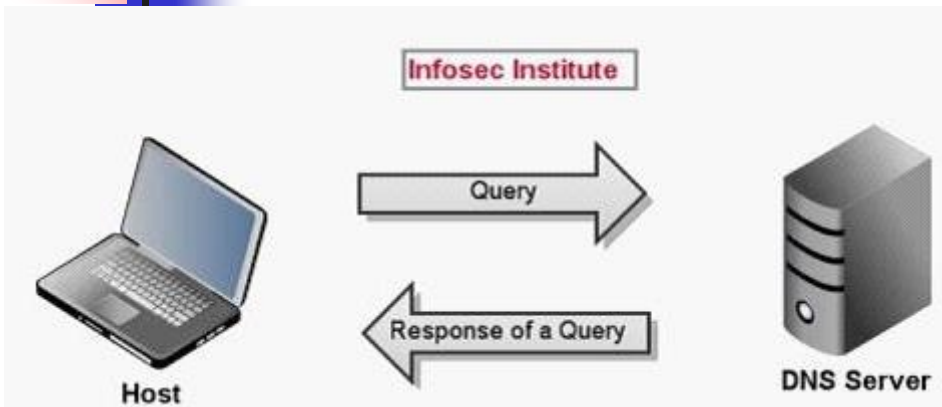
- ICANN向各注册登记机构授权
- cn顶级域域名，由信产部中国互联网络信息中心（China Internet Network Information Center，简称CNNIC）管理
- 来自互联网研究认证中心的消息：2014年7月25日，互联网名称与数字地址分配机构（ICANN）发布消息，在中国增设L根域名服务器镜像节点。此举将提升DNS的安全性、稳定性以及容错性，缩短中国互联网用户查询域名的响应时间，并有助于抵御网络攻击、域名劫持和网络瘫痪等网络威胁。
<http://www.zgydhlw.cc/dongtai/yenei/145.html>
- 域名注册：通过域名注册网站提交申请并付费



针对DNS的攻击

- 通过发送ICMP ping报文攻击13个根服务器（2002年10月21日）；
- 发送大量DNS请求报文给顶级域名服务器
- 伪造目标主机（如邮件服务器）向权威DNS发送DNS请求；导致目标主机接收大量的响应信息
- 中间人攻击：截获主机请求报文，伪造应答，使主机重定向到错误的网址。因DNS查询没有认证机制，容易被篡改，通过对UDP端口53上的DNS查询进行检测，发现与关键词相匹配的请求，则伪装成目标DNS向查询者返回虚假结果

DNS欺骗中间人攻击



攻击者冒充DNS服务器向主机提供错误的DNS信息。

ARP：查询及响应，实现IP--MAC的映射

ARP缓存中毒：攻击者对目标设备进行ARP缓存中毒攻击，之后将拦截DNS查询请求，然后就能够发送欺骗性的DNS响应报文，导致主机访问了错误的网站；根源在于未对目标主机认证



问题

- UDP是不可靠的数据报传输协议。UDP依据什么实行对上层应用的多路复用和分用？UDP的校验和是如何计算的？校验的范围包括什么？
- 为何只能有13 个根DNS？中国为什么没有根域名服务器？
- 根域名服务器与根域名服务器镜像有何区别？
- 如果一个恐怖分子破坏了全球的DNS服务器，问这将如何改变一个人使用互联网的能力？



问题

- DNS采用UDP，若出现数据报文丢失，会出现问题吗？如果会，是如何解决的？
- 随着Web服务器的急剧增长，注册在.com域的公司成千上万，导致该域的顶级服务器负载繁重，请提出一种缓解该问题的方案，但不用修改域名，可以修改客户代码
- 若要禁止某些用户访问某些网站，有何办法？