

# 今日干饭背诵 (1.8)

## Chap9 网络应用层

### 应用层概述

(应用层概述没有出现在考纲里，感觉也没有什么考点，看看概念理一下思路即可)

#### 【其他】

网络应用程序：互相通信的分布式进程

应用层协议：应用程序的一部分，定义应用程序之间交换的信息以及相应的动作，利用底层协议提供的服务

进程通信：

- 同一主机 —— 利用操作系统提供IPC (interprocess communication 进程间通信)
- 不同主机 —— 利用应用层协议
  - 用户代理 (user agent)：用户 & 网络应用程序 之间的接口 (e.g. web浏览器、流媒体播放器)
  - 应用程序编程接口 (API)：应用程序和传输层之间的接口
  - “应用层寻址”：进程通过 `<IP address, local port>` 定位对方进程，其中IP指明对方主机，port指明对方进程

应用程序需要的传输服务：数据丢失、带宽、延迟上的容忍度

互联网传输协议提供的服务：见P8

### 客户/服务器模型

#### 【背诵】

地位：**C/S模型**是网络应用的基础 (Client / Server)

典型的C/S模型系统：**DNS、HTTP、FTP**

相关方：客户/服务器分别指参与一次通信的两个应用实体

- **客户方主动请求**
- **服务器方被动响应** (等待通信的建立，并提供服务)
  - 并发性 —— 服务器软件
    - 一部分用于接受请求并创建新的进程或线程
    - 另一部分用于处理实际的通信用过程

使用的传输层协议：

- 基于连接的**TCP** (适用于可靠的交互过程)
- 无连接的**UDP** (适用于可靠性要求不高 / 实时交互的过程)
- 同时使用TCP、UDP

# DNS (域名系统)

## 【背诵】

概述：

- 典型的客户 / 服务器交互系统
- 多层次、基于域的命名系统，使用分布式数据库实现
- DNS的request / response报文封装在 UDP 分组中

操作过程：

1. request - 应用程序需要进行域名解析时，它成为client，并试图与本地域名服务器发出请求，建立连接
2. reply - 本地域名服务器找到对应IP地址，给出响应
3. (多层次的DNS：本地域名服务器找不到时，自身成为上级域名服务器的client，继续解析)

摘自MashPlant的网原笔记 ——

域名解析过程：应用->区域域名服务器-(若找不到)->根域名服务器-(若找不到)->authoritative域名服务器

- 根域名服务器
  - 可能不知道 authoritative域名服务器，但
  - 知道中间的域名服务器，而中间域名服务器知道如何与 authoritative域名服务器联系
- 因此出现了两种请求模式 ——
  - Recursive query (递归式)：问一个人就是把任务完全交给它
    - 【优点】向用户提供较好的服务
    - 【缺点】根域名服务器的负担重
  - Iterated query (反复式)：可以从别人那里得到额外信息。别人：“我不知道，但可以问它”
    - 【优点】负担较小
    - 【缺点】提供的服务不太好（需要自己查，而不是甩了锅就完事）

域名的结构：

- 分为两类：组织结构 (.org) / 地理结构 (.cn表示中国)
- 特点：大小写无关；最长255个字符，每部分最长63个字符

**资源记录：** DNS数据库中用来表示主机和子域的信息，域名解析成功的返回结果就是它，形式上是五元式<Domain\_name, Time\_to\_live, Type, Class, Value> (具体含义见P20-22)

**区域划分：** DNS将域名空间划分为许多无重叠区域，每个区域覆盖了域名空间的一部分，区域的边界划分是人工设置的。每个区域有一个主域名服务器和若干个备份域名服务器。

## SNMP (简单网络管理模式, Simple Network Management Protocol)

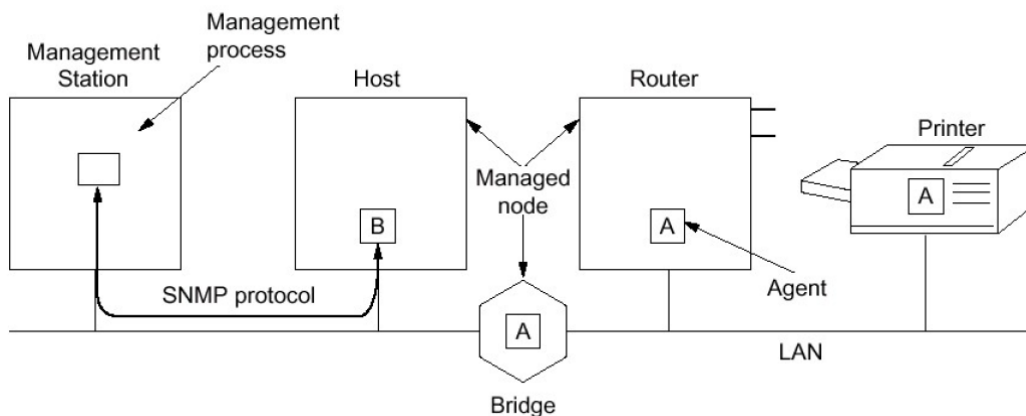
封装在 UDP 分组中。

## 【背诵】

五大管理功能：性能管理、故障管理、配置管理、记账管理、安全管理

SNMP网管模型的四个组成部分：agent、manager、information (MIB)、protocol (SNMP)

- 被管理节点：运行SNMP代理程序 (agent)，维护本地数据库 (记录状态历史)
- 管理工作站：运行专门的网络管理软件 (manager)，使用管理协议与被管理节点上的SNMP代理通信，维护管理数据库
- 管理信息：每个站点使用多个对象描述自己的状态，所有的对象组成MIB (管理信息库)
- 管理协议 (SNMP)：定义了网络管理工作站和SNMP代理之间的通信过程 & 协议数据单元 (PDU=SDU+PCI)
  - 管理工作站用来查询、修改被管理节点的状态
  - 被管理节点用来向管理工作站产生“陷阱” (trap) 报告



管理信息结构SMI：变量被定义为对象 (object)，相关对象们被集合成组 (group)，组被汇集成模块 (module)。

管理信息库MIB：包含10个组，manager向agent查询的就是保存在MIB中的对象的值。

【注意】被管理节点是server，管理工作站是client

## 【概念】

抽象语法表示法1 (ASN.1)：一种标准的对象定义语言

要理解抽象语法表示法，ASN.1给了一个参考，要学会给定场景设计类似编码

- 数据描述定义：
  - 基本数据类型：Integer=2、Bit String (bits) =3、Octet String (bytes) =4、Null=5、Object Identifier (type) =6
  - 对象命名树：使用编码 (如上1~6)，唯一确定每个标准中的对象
  - 构造新类型的方法：Sequence (多种类型的list)、Sequence of (一种类型的list)、Set、Set of、Choice (union)，或者重新标记一个老的类型 (类似 #define)
- 传输语法定义：
  - 基本编码规则BER：ASN.1类型的值 → 无二义的字节序列
  - 传输内容：
    - 标志符：tag | type | number (具体见P36)
      - tag在0~30之间时，用低5位表示
      - tag > 30时，低5位为11111，用后面字节表示

## ■ 数据长度域

### ■ 第一个字节:

1. 数据长度 $<128$ 则填入该长度, 此时由于 $\log_2 128 = 7 < 8$ 可知, 最高位一定为0
2. 数据长度 $>128$ 则最高位=1, 低7位=x, x为用于表示长度的字节个数 (只需要向后查找x个字节, 就能获得真正的长度)

### ■ 例子:

- **数据域长度**

第一个字节表示:  
①长度 $<128$ 则就是长度, ② $>128$ 则MSB=1, 低7位= $x$  (dec), x为用于表示长度的字节个数 (向后查找x个字节, 获得真正的长度)

  - 当长度  $< 128$ 字节时, 用一个字节表示长度, 高位为“0”
  - 当长度  $\geq 128$ 字节时, 第一个字节高位为“1”, 低7位表示后面表示长度的字节个数, 后面的若干个 ( $\leq 127$ ) 字节表示长度
  - 例, 数据长度1000字节, 则长度域包括3个字节, 第一个字节为“10000010”, 后两个字节为“00000011”和“11101000”拼起来表示1000(dec)

## ■ 数据域

- Integer: 二进制编码
- 其他: 见P38

例子:

	Tag type	Tag Number	Length	Value
Integer 49	00	000010	00000001	00110001
Bit String '110'	00	000011	00000010	00000101 11000000
Octet String "xy"	00	000100	00000010	01111000 01111001
NULL	00	000101	00000000	
Internet object	00	000110	00000011	00101011 00000110 00000001
Gauge 32 14	01	000010	00000001	00001110

Fig. 7-34. ASN.1 encoding of some example values.

## 电子邮件

### 【背诵】

体系结构:

- 用户代理: 一般为用户进程 (user agent)
- 消息传输代理: 一般为系统的后台进程 (mail server)

- 简单邮件传输协议SMTP (Simple Mail Transfer Protocol, 注意和SNMP区别) : 只支持用户代理→邮件服务器, 或邮件服务器之间, 不支持邮件服务器→用户代理 (如果要的话得用POP3 或 IMAP, 当时SMTP在80~90年代被应用时大家是去服务区上看邮件的)

回忆:

- SMTP: Simple Mail Transfer Protocol
- SNMP: Simple Network Management Protocol

电子邮件的组成: 信封、信件内容

电子邮件的扩展: MIME (Multipurpose Internet Mail Extensions), 对图像声音等支持, 使用编码将信息转化为ASCII字符流。

### 【概念】

消息传输代理在源端主机和目的主机的25号端口建立TCP连接, 使用SMTP进行通信 (还有其他可用的协议如POP3、IMAP, SMTP是最基本的) ;

消息传输代理和用户代理两个方向的发送完成后, 释放TCP连接(QUIT)。属于持久 (Persistent) 方式。

## WWW (World Wide Web)

### 【背诵】

WWW模型的四个元素:

- Web页面: 由对象 (object) 组成, 用URL标识地址 (协议类型+server地址+object路径名, 如 [https://news.tsinghua.edu.cn/ztxw/tzgjdsj\\_zzcczyq.htm](https://news.tsinghua.edu.cn/ztxw/tzgjdsj_zzcczyq.htm))
- 浏览器 (browser) : 用户访问网页的客户端client
- Web服务器: 存储Web对象 (object) 的server
- 超文本传输协议HTTP (基于TCP, 在80端口)

### 【概念】

Web缓存: 浏览器不访问原服务器 (origin server), 而去访问代理服务器 (proxy server)

- 缓存命中, Web对象 (object) 被立刻通过HTTP response返回
- 不命中, 代理服务器向原服务器请求获得该object再返回

【优点】响应快, 减轻远端服务器的负担

【问题】可能取到错误的数据

## FTP (文件传输协议)

### 【背诵】

工作过程:

- 客户端连接到FTP服务器TCP的21号端口, 然后建立两个并行的TCP连接:
  1. port 21 => 控制: 在客户端和服务端之间交换命令、响应 (采用telnet)
  2. port 20 => 数据: 传递文件数据 (可以双向使用, 不必始终存在)
- FTP服务器维护状态: 当前目录, 身份认证

## 简单总结

- **DNS、SNMP**都基于UDP，电子邮件SMTP、HTTP、FTP、TELNET基于TCP

回忆：chap8 封装在UDP分组的协议

- RIP（路由表周期性广播，TCP做不到组播/广播）
  - DNS（避免TCP连接建立延迟）
  - SNMP（避免TCP在拥塞控制的时候把网管的包优化了，牛牛岂是你想吃就吃）
- 早期网络如ARPANET（1969）规模小，因此不需要很复杂的协议工具，但规模增长以后就出现了DNS（1983）、SNMP（1990）以管理网络。
  - 客户/服务器模型：**DNS、HTTP、FTP**
  - 带内传输：HTTP、SMTP（电子邮件）；带外传输：FTP