



## 计算机学院 2021-2022(2)《计算机网络》第 3 次作业参考答案 (第 6 章)

### 一、选择题

- 1.C      2.D      3.A      4.A      5.C      6.B      7.C      8.B      9.A      10.D  
11.D      12.B      13.D      14.A      15.B      16.B      17.B      18.A      19.D

### 二、填空题

1. (1)SMTP(2)SNMP    2. (3)POP3(4)SMTP      3. (5)UDP      4.(6)代理      5.(7)解释      6.(8)事件通知  
7.(9)递归解析      8.(10)客户机/服务器(C/S)    9.(11)访问类型      10.(12)递归解析

### 三、问答题

- 文件传送协议 FTP 是因特网上使用最广泛的文件传送协议, 它提供了交换式的访问, 允许客户端指明文件的类型与格式, 允许文件具有访问权限, 并使用 TCP 协议进行信息传输。其工作过程是: 首先服务器端运行的守护进程(主进程)打开了熟知端口 21, 等待并接收来自客户端的连接请求。当有客户端连接请求到达时, 服务器端主进程启动第一个从属进程(控制进程)并建立与客户端的第一个连接(控制连接)来处理该客户端的请求, 主进程则继续侦听其它客户端的连接请求。从属进程可根据需要创建其它子进程。其次, 客户端向服务器端发出连接请求并得到服务器确认连接后, 服务器端为该客户端创建的第一个从属进程(控制进程)就与客户端建立了第一个连接(控制连接), 控制连接在会话期间一直保持打开, 用于客户端和服务端之间发送有关命令并得到命令的响应, 但它不负责实际数据文件的传送。当客户端选择并要求服务器端传送的文件时, 服务器就为该客户端创建第二个从属进程(数据进程)并建立与客户端的第二个连接(数据连接), 数据连接实际负责传送数据文件。数据文件传送结束后数据连接就被关闭。  
由于 FTP 使用了一个分离的控制连接, 因此, FTP 的控制信息是带外传送的。
- (1) 解析 IP 地址需要发出 DNS 请求命令, 题中没有指出采用的是何种查找方式, 因此认为既适用于递归查找也适用于迭代查找, 无非发出 DNS 请求命令的主机不同而已。因此, 解析 IP 地址所需要的时间为  $RTT_1+RTT_2+\dots+RTT_n$ , 得到 IP 地址后需要建立本地主机与服务器的 TCP 连接, 建立 TCP 连接的过程包含了请求连接和对连接的确认(第 2 次握手, 共一个 RTT)和再确认(第 3 次握手), 而再确认的过程中可以捎带请求万维网文档报文(单程即半个 RTT), 万维网文档响应报文(包括小图片)还需要半个 RTT(单程), 因此, 需要的总时间是:  $2RTT_w+RTT_1+RTT_2+\dots+RTT_n$ 。
  - (2) 不使用并行 TCP 的非持续 HTTP 时, 对于每一个对象都需要建立连接并读取对象, 首先建立传输万维网文档的连接并下载万维网文档(包括小图片)的 TCP, 需要时间  $2RTT_w$ , 然后点击读取三个小对象, 对每一个小对象都需要先后建立 TCP 及其请求与响应的过程(不并行), 因此, 每一个小对象需要  $2RTT_w$ , 共需要  $6RTT_w$ , 所以, 所需时间  $=RTT_1+RTT_2+\dots+RTT_n+8RTT_w$
  - (3) 使用并行 TCP 连接的非持续 HTTP 时, 需要分为两个阶段, 首先建立 TCP 并传输万维网文档, 需要  $2RTT_w$ , 然后同时建立三个并行的 TCP 连接分别用于请求和响应三个小对象, 需要  $1RTT_w$ , 接着在这三个并行的 TCP 连接上请求(再确认和捎带)并响应三个小对象的传输, 需要  $1RTT_w$ , 所以, 所需总时间  $=RTT_1+RTT_2+\dots+RTT_n+4RTT_w$ 。
  - (4) 使用流水线方式的持续 HTTP 时, 只需要一个 TCP 连接就可以传输所有内容, 但也需要分为两个阶段, 首先建立 TCP 连接并传输万维网文档, 需要  $2RTT_w$ , 然后在此 TCP 连接期间, 连续地发出对三个小对象的请求及接收响应, 共需要  $1RTT_w$ , 这样, 所需总时间  $=RTT_1+RTT_2+\dots+RTT_n+3RTT_w$ 。
- (1) Base64 编码是一种基于 64 个可打印字符来表示二进制数据的方法。其算法是:
  - ① 先把二进制代码划分为 1 个 24 位长的单元。原二进制代码按 24 位划分如下:  
11001100 10000001 00111000
  - ② 把每一个 24 位单元划分为 4 个 6 位组, 则原二进制代码按 6 位组划分后如下:  
110011 001000 000100 111000
  - ③ 把每一个 6 位组转换成 ASCII 码。6 位的二进制代码共有 64 种值, 分别用 A~Z 表示 0~25, a~z 表示 26~51, '0'~'9' 字符表示 52~61, '+' 表示 62, '/' 表示 63。如下表所示, 经查表该 24 位组对应的 ASCII 码组合为: zIE4, 此即原二进制的 Base64 编码。

6 位二进制	表示字符	6 位二进制	表示字符	6 位二进制	表示字符	6 位二进制	表示字符	6 位二进制	表示字符
000000	A	001101	N	011010	a	100111	n	110100	0
000001	B	001110	O	011011	b	101000	o	110101	1
000010	C	001111	P	011100	c	101001	p	110110	2
000011	D	010000	Q	011101	d	101010	q	110111	3
000100	E	010001	R	011110	e	101011	r	111000	4
000101	F	010010	S	011111	f	101100	s	111001	5
000110	G	010011	T	100000	g	101101	t	111010	6
000111	H	010100	U	100001	h	101110	u	111011	7
001000	I	010101	V	100010	i	101111	v	111100	8



001001	J	010110	W	100011	j	110000	w	111101	9
001010	K	010111	X	100100	k	110001	x	111110	+
001011	L	011000	Y	100101	l	110010	y	111111	/
001100	M	011001	Z	100110	m	110011	z		

- ④ 如果不足 24 位,则在 Base64 编码后用“=”表示最后一组的代码只有 8 位,用“=”表示最后一组的代码只有 16 位。另外,回车和换行都被忽略,它们可以在任何地方插入。

(2) quoted-printable 编码适用于当所传送的数据中只有少量的非 ASCII 码,例如汉字。其算法是:①对所有可打印的 ASCII 码,除特殊字符等号“=”外,都不改变;②等号“=”和不可打印的 ASCII 字符以及非 ASCII 码的数据的编码方法是:先将每个字节的二进制代码用两个十六进制数字表示,然后在面前再加上一个等号“=”。

被传送的二进制数据 11001100 10000001 00111000 中前两个字节是汉字,最后一个字节是 ASCII 字符 8,因此需要将前两个字节进行转换。第一个字节转换成十六进制数为 CC,第二个字节转换成十六进制为 81,因此原 3 个字节(24 位)编码后的传送代码为“=CC=818”,共 7 个字节(56 位),编码开销= $\frac{56-24}{24} = 133.3\%$

4. 每一台主机都需要配置 IP 地址和子网掩码才能进行网络通信, DHCP 协议又称为动态主机配置协议,它为主机提供了即插即用连网的机制,允许每台主机加入新网络时能通过该协议自动地获取 IP 地址而不用手工配置。除了普通主机可使用该协议外,还可用于无盘工作站以获取 IP 地址。

当一台计算机第一次运行引导程序时,其 ROM 中并没有该主机的 IP 地址、子网掩码或某个域名服务器的 IP 地址,因为这些网络上使用的信息都是通过软件配置的,也称为逻辑地址或软件地址。只要主机所连接的网络上运行 DHCP 协议的服务器,那么计算机就可以通过广播方式发送发现报文,而由 DHCP 服务器单播提供报文,从而使计算机得到 IP 地址等信息。

5. (1) TCP 和 UDP 是传输层的两个重要协议, TCP 是面向连接的、可靠的协议,而 UDP 是尽最大努力传送的、不可靠的、无连接的协议。应用层存在者众多的应用协议,这些协议根据使用的环境和目的,有些需要可靠性,这种可靠性由传输层的 TCP 来支持,而部分应用层协议则要求简单、实时从而不需要可靠性,这部分应用层协议由传输层的 UDP 来支持。使用 UDP 协议的应用层协议,它们的可靠性也可以由应用程序本身的编码来实现。另外,使用 UDP 的应用层协议通常传输的数据量较小,发生错误的几率也小。

(2) SMTP 是邮件协议中的基础的传输协议,用于邮件代理向邮件服务器以及邮件发送方邮件服务器向邮件接收方邮件服务器发送邮件,但这个协议只能支持 ASCII 的邮件传输,不支持 ASCII 之外的其它语言文字、图像、音频等的邮件传输, MIME 是 SMTP 的扩展协议,它首先将非 ASCII 码邮件转换成 ASCII 邮件然后用 SMTP 进行传输,所以, MIME 使用了 SMTP,是 SMTP 的更高层,而不是代替 SMTP 协议。协议栈的上层的实现需要调用下层提供的功能,而下层则是为上层提供服务。因此, MIME 应画在 SMTP 之上。

(3) 路由器选择协议 RIP 是用于中小型网络中构造路由器路由表的协议,该协议需要定时地与相邻路由器交换整个路由表信息。从协议的层次结构看,这个被交换的路由表信息被封装在 UDP 数据报中;从原理上看,路由器构造路由表需要在路由器中运行有关进程, RIP 报文就是运行在不同路由器上的进程间交换信息的单位。因此, RIP 被放在了应用层。