**第一章 计算机网络和因特网**

从具体构成（硬件软件）

由一群遵循TCP/IP协议的ISP，按照松散的层次结构组织而成的网络的网络

对于通信功能的实现有指导作用：ISP内部实现，ISP之间互联

为分布式应用提供通信服务的基础设施

对于服务接口的定义有指导作用：有序、可靠的数据交付服务，不可靠的数据交付服务

上行速率< 2.5 Mbps (典型地 < 1 Mbps)

下行速率< 24 Mbps (典型地 < 10 Mbps)

下行速率最高 30Mbps，上行速率最高 2 Mbps

注：接入网链路：主机连接到互联网的链路；主干链路：路由器间的链路。

计算机网络是网络的网络。

名词中英文：

计算机网络：computer network

主机：host

端系统：end system

通信链路：communication link

分组：packet

分组交换机：packet switch

链路层交换机：link-layer switch

路由器：router

传输速率：transmission rate

比特/秒：bit per second ,bps

因特网服务提供商ISP: internet service provider

数字用户线DSL：digital subscriber line

因特网工程任务组IETF: internet engineering task force（研发互联网标准）

请求评论RFC：request for comment（IETF互联网协议的标准文档）

2. 从服务的角度

互联网：为应用程序提供服务的基础设施。

互联网络 = 能够为应用提供通信服务的通信架构 + 使用通信服务相互配合工作的分布式应用

协议（protocol）：对等层实体在通信过程中所遵循的规则的集合，定义了交换报文的格式、次序以及报文发送、发送或其他事件所采取的动作。

1.2 网络边缘

网络结构 = 网络边缘（应用、主机）+ 网络核心（路由器）+ 接入网络和通信链路

主机 = 端系统

网络边缘：运行应用的端系统

C/S模式：客户端服务器模式，客户端向服务器发出请求，服务器响应，提供服务，集中式的服务，客户端之间相互之间不直接通信，且服务器有固定的周知的IP地址。

对等模式P2P：很少有专门的服务器，任何主机都可以作为服务器或者客户端，是分布式的服务，具有自扩展性和成本效率。

面向连接的服务TCP，无连接的服务UDP

1.3 网络核心

网络核心：路由器的网状网络

主要功能：实现数据交换

数据交换方式及比较：

分组交换：（统计多路复用）

将要传输的数据分为一个个分组，将分组从一个路由器传送到另一个路由器，最终到达目标，在每段链路上都使用链路的最大传输能力，路由器上使用存储转发。

存储转发传输（store and forward transmission）：接收分组，存储分组，处理，转发。有存储转发时延，总的时延称为端到端时延。

有虚电路网络和数据报网络

电路交换（线路交换）：两台主机通信时，会在主机间创建一条名副其实的连接，路径上的交换机都将为该连接维护连接状态，资源不同享。

特点：

分组交换：性能不可靠，资源共享，无创建时间，适用于突发传输

电路交换：性能可靠，资源不共享，有创建时间，不适用于突发传输，可能有资源浪费。

1.4 接入网

接入网：将端系统连接到边缘路由器的链路或网络

边缘路由器：端系统到任何一个远程端系统路径上的第一台路由器

住宅接入：ADSL，HFC，，，，

1.5 互联网络结构与ISP

网络的网络

多层等级结构：接入ISP，区域ISP，第一层ISP

对等：同一层相互连接，降低成本

IXP：提供多个ISP的对等 internet exchange point

内容提供商网络（content provider network）：在全球部署数据中心DC，与多处的较低层ISP相连，尝试绕过叫高层，部署DC之间的专网，用户接入后通过离用户最近的DC为之服务。

因特网内容提供商ICP（internet content provider）

1.6 分组交换网络中的延迟与丢失

1.延迟：（delay）

节点处理时延：（nodal processing delay）检查分组首部，决定出链路等，高速路由器大约花费不到微秒。

排队时延：（queuing delay）毫秒到微秒

传输时延：（transmission delay）发射分组的时间

传播时延：（propagation delay）与距离有关

d n o d a l = d p r o c + d q u e u e + d t r a n s + d p r o p d\_{nodal} = d\_{proc} + d\_{queue} + d\_{trans} +d\_{prop}

d

nodal

​

=d

proc

​

+d

queue

​

+d

trans

​

+d

prop

​

2.排队时延与丢包

排队时延：取决于入、出流量以及流量的性质（时间上的分布）

流量强度（traffic intensity）：

L a / R La/R

La/R

注：a为每秒到达分组个数 L为每个分组的比特数，R为链路带宽

流量强度与\*\*dqueue\*\*有关

丢包：分组到达链路的速率超过了链路的输出能力，超过排队缓冲区容量，到达的分组被丢弃。

节点性能：时延，丢包率

吞吐量：在源和端之间传输的速率，有瞬间吞吐量和平均吞吐量。取决于源和目的地之间路径上的最小传输速率。端到端路径尚限制端到端吞吐的链路称为瓶颈链路

1.7协议层次与服务模型

分层原因：网络比较复杂，分层实现比较容易设计。概念化、结构化，易于维护和升级。

分层：指将复杂的网络功能划分成功能明确的层次，上层利用下层提供的服务来实现本层的协议，从而为更上层提供更复杂的服务。

术语和概念：

服务：底层实体向上层实体提供它们之间通信的能力

服务访问点SAP：services access point 上层使用下层提供的服务通过层间的接口

原语（primitive）：上层使用下层服务的形式

面向连接与无连接的服务：（connection oriented service）（connectionless service）面向连接：建立连接，通信，拆除连接。在通信前双方先建立连接。无连接：通信前无需建立连接，不预留资源，不需要双方都活跃。

协议：对等层实体之间在相互通信的过程需要遵循的规则的集合

服务与协议之间的关系：

区别：服务是下层对上层提供它们之间通信的能力，协议是对等层实体通信的规则集合

联系：本层协议的实现要靠下层提供的服务；本层实体通过协议为上层提供更高级的服务。

数据单元DU：（data unit）

协议数据单元PDU：（protocol data unit）

分层模型与每层的功能：

应用层：为人类或其他应用进程提供网络应用服务 协议数据单元名称 报文message

传输层：主机之间的数据传输 报文段 segment

网络层：为数据报从源到目的选择路由 分组 packet

链路层：相邻网络节点间的数据传输 帧 frame

物理层：线路上传送bit 位 bit

封装与解封装：向下逐层增加首部信息，向上逐层减少首部信息，路由器在三层，链路层交换机在二层。

第二章 应用层

2.1网络应用原理

应用架构：C/S模式 对等模式 混合体

C/S模式：客户端服务器模式，客户端向服务器发出请求，服务器提供服务，是集中式的服务，客户端相互之间不直接通信，且服务器有固定的周知的IP地址，可扩展性差。

对等模式P2P：很少有专门的服务器，任何主机都可以作为服务器或者客户端，是分布式的服务，具有自扩展性和成本效率。

进程间通信：

同主机：操作系统定义的通信方式

不同主机：利用网络提供的架构交换报文

1. 标识进程：IP + TCP或UDP端口号

2. 传输层提供的服务：在服务访问点SAP提供服务，形式是应用程序接口API

对应的是SOCKET和SOCKET API

SOCKET的作用是减少需要通过层间的信息

TCP SOCKET包括 源IP，源端口，目标IP，目标端口。每次传输报文时，告知传输层套接字即可，下层可以利用套接字查表获取端口号和IP用于报文段或数据报的封装。

UDP SOCKET包括 源IP，源端口。每次传输报文时，告知传输层套接字和目标的IP和端口号，接收报文时，传输层上传对方的IP和端口号。

3. 定义应用层协议的报文格式，解释，时序等。

应用层协议：定义了运行在不同端系统上的应用进程如何交换报文，规定了报文的类型语法语义规则等。

传输层提供的服务性能：数据丢失率，吞吐量，延迟，安全性

UDP存在的必要性：

虽然没有加强IP提供的通信的性能，但是能够区分不同的进程；

相比TCP，省去了连接建立时间；

不做可靠性的工作，省时，适用于实时性要求高于正确性要求的应用；

没有拥塞控制与流量控制，能按照设定的速度发送数据。

安全的TCP

SSL：（secure sockets layer 安全套接字层）运行在应用层，提供加密的TCP连接

需要该应用程序的客户端和服务端都包含SSL代码。

2.2WEB和HTTP

1. WEB页面：

由对象组成，对象可以是HTML文件，JPEG图像，java小程序等

WEB页包含一个基本的HTML文件，该基本的HTML文件又包含若干对象的引用

通过URL对每个对象进行引用

HTML （hypertext markup language）超文本标记语言

URL（uniform resource locater）统一资源定位地址 ：主机名+对象的路径名

HTTP（hypertext transfer protocol）超文本传输协议

采用C/S模式，HTTP定义了客户端和服务端之间通信的报文的格式，解释和时序

获取页面过程：建立TCP连接，服务器端口号80；浏览器与服务器交换HTTP报文；TCP连接关闭。

无状态协议（stateless protocol）：服务器响应客户的请求而不存储任何有关客户的状态信息。

2. HTTP连接类型：

非持续连接（non-persistent connection）：每个响应/请求对都是经一个单独的TCP连接发送。

1个HTML基本文件和10个其他对象，则每次都建立TCP连接、请求响应，共11次。

并行非持续：同上的WEB页面，首先建立连接、请求响应，之后并行10个建立连接、请求响应。

缺点：多个TCP连接，占用资源

持续连接（persistent connection）：所有请求及响应都是经相同的TCP连接发送

同上的WEB页面，首先建立连接、请求响应，之后进行10次请求响应。

（默认）带流水线的持续：同上的WEB页面，首先建立连接、请求响应，之后按流水线的方式发送10次请求，得到10个响应，几乎在1个RTT内得到10个对象的响应报文。

RTT（round-trip time）往返时间：一个短分组从源到目标再返回源的总延时。（无传输时延）

一次请求响应总时间 ≈ RTT + 传输延时

3. HTTP报文：

格式：ASCII

请求报文：

请求行：方法 URL HTTP版本 cr lf

多个首部行：首部名：值 cr lf

cr lf

实体体：

方法：GET POST HEAD PUT DELETE

注：GET方法实体体位空，可以通过URL字段提交表单。POST方法实体体可用于提交表单，同时仍可以请求一个页面。首部行中的Host对Web缓存很重要。

响应报文：

状态行：版本 状态码 短语 cr lf

多个首部行：首部名 值 cr lf

cr lf

实体体：

状态码 短语：200 OK 301 Moved Permenently 404 Not Found

注：首部行Date表示发送报文的时间，与Last-Modified不同。首部行中的Last-Modified对web缓存很重要。条件GET的响应报文的实体体可为空。

4. COOKIE：

cookie的作用是弥补HTTP的无状态特点，让服务器能够限制用户的访问或者把内容与用户身份联系起来。

4个组成部分：第一次响应的Set-cookie首部行，请求中的cookie首部行，用户端由浏览器管理的cookie文件（包含服务器主机名和识别码），web服务器后端数据库中存储的表项。

5. Web缓存（web cache）（代理服务器 proxy server）：

在存储空间保存最近请求过的对象的副本，无需访问原始服务器，就可以满足客户的请求。

作用：大大减少对客户请求的响应时间；大大减少一个机构的接入链路到因特网的通信量，无需增加带宽。

使用方法：配置浏览器，对某对象的访问首先定向到web缓存器。

内容分发网络CDN（content distribution network）：CDN公司在因特网上安装了许多地理上分散的缓存器，使大量流量实现了本地化。

条件GET方法：GET方法 + 首部行If-Modified-Since。解决了缓存中副本的陈旧问题。如果请求内容有修改过，则原始服务器发送最新的版本。

2.3 EMAIL

异步通信媒介

因特网电子邮件系统：用户代理、邮件服务器、SMTP

简单邮件传输协议SMTP（simple mail transfer protocol）

基于TCP（Port：25），邮件报文的体部分只能采用7比特ASCII，使用持续连接。

过程：TCP连接建立->(SMTP)->HELO->指示收发双方的邮件地址->发送报文->QUIT（或者指示新的收发双方，发送第二个报文）->TCP连接关闭

邮件内容结束标志：cr lf . cr lf

SMTP对比HTTP：

相同：用于传送文件，可以采用持续连接

不同：HTTP为拉协议，SMTP为推协议；SMTP报文首部和主体都必须为7位ASCII，HTTP的数据部分无编码限制；HTTP一个对象一个报文，SMTP多个对象一个报文。

邮件报文格式：

指SMTP命令中DATA后面发送的内容的格式。

MIME多媒体邮件扩展：解决7位ASCII的限制问题，将不符合要求的数据编码成7位。

邮件访问协议：

用户代理- SMTP -发送方邮件服务器- SMTP -接收方邮件服务器- POP3/IMAP/HTTP -用户代理

POP3：post office protocol -version 3 第三版的邮局协议

IMAP：internet mail access protocol 因特网邮件访问协议

2.4 DNS

DNS：Domain Name System

必要性：IP地址不好记忆，且数字没有实际意义

分层命名，分布式数据库完成名字到地址的转换，运行在UDP 53号端口的应用层协议。

层次化命名：

TLD：top lever domains 顶级域

命名：…子域.子域.顶级域

DNS服务器：

根DNS服务器：13个不同组织管理。提供TLD服务器的IP地址。

顶级域DNS服务器：（TLD服务器）对应顶级域，提供其下的权威服务器的IP地址。

权威DNS服务器：对应子域，提供主机名对应的地址。

本地DNS服务器：代替主机向其他DNS服务器进行查询。由ISP管理，通过ISP接入时，ISP会提供一个DNS服务器。

资源记录与DNS报文：

RR：resource records资源记录

TTL：time to live 生存时间（单位：s）

RR格式：name Value Type ttl class （对于internet Class为IN，下面不考虑）

主机名 IP地址 A

别名 规范名 CNAME

别名 规范名 MX

域 某DNS服务器主机名 NS

注：邮件服务器别名与主机别名可以相同，查询时在查询报文中的type不同。

DNS报文：首部（标识符、标志位、4个数量）、问题、回答、权威、附加

DNS解析：

过程：应用调用解析器，解析器通过UDP段向本地DNS服务器查询，再将IP地址返回给该应用。

例如访问Web页面的协议使用顺序： （调用解析器）DNS查询 UDP发送 UDP接收 得到DNS响应 （浏览器）TCP建立 HTTP请求

查询分类：递归查询（根服务器负担过重），迭代查询。

DNS缓存：

作用：改善时延性能，减少因特网上传输的DNS报文数量。由于本地服务器可以缓存TLD服务器的地址，所以多数时候，根服务器被绕过了。

原理：当一个DNS服务器接收到一个回答报文时，它会将其缓存在本地存储器，并设置一个TTL（默认2天）

2.5 P2P（了解概念和优势）

P2P：几乎没有一直运行的服务器，任何端系统都可以作为服务器或者客户端，提供分布式的服务，扩展性好，传输速率快。P2P网络是peer构成的应用层面的逻辑网络。

DHT：distributed hash table 分布式散列表

2.6 视频流与内容分发网络CDN（了解）

DASH：Dynamic Adaptive Streaming over HTTP 基于HTTP的动态自适应流

客户获取视频的过程：客户获取告示文件（包含不同质量视频的URL）；客户按照情况请求不同质量的内容块。

CDN：content distribution network内容分发网络

ICP（互联网内容提供商）在各个地区部署缓存节点，用户通过域名重定向向最近缓存节点请求内容。

2.7 TCP的SOCKET编程

2.8 UDP的SOCKET编程

第三章 传输层

3.1 传输层服务

传输层服务：为运行在不同主机上的应用进程提供逻辑通信。发送方将应用层报文分成报文段，再传递给网络层，接收方反之。

传输层与网络层的区别：传输层实现的是进程之间的逻辑通信，网络层实现的是主机之间；传输层对网络层的服务进行了加强。

UDP：user datagram protocol 用户数据报协议

TCP：transmission control protocol 传输控制协议

网络层：尽力而为的交付服务（best-effort delivery service）

TCP：可靠、保序、多路复用解复用、拥塞控制、流量控制、面向连接

UDP：多路复用解复用

都不提供的服务：延时保证、带宽保证

3.2 多路复用与解复用

指的是TCP、UDP在网络层之上实现的进程到进程之间的通信

复用：传输层协议根据套接字中包含的端口信息封装传输层分组，将分组交给网络层。

解复用：传输层协议根据网络层交付的分组中的传输层首部，得到端口号，并查到套接字，将应用层数据交给套接字。

UDP套接字：2元组

TCP套接字：4元组

具体过程…

3.3 UDP

UDP的必要性：

UDP相比于IP，提供了区分主机上不同进程的服务；

相比于TCP，UDP没有连接建立时间；

不做可靠性的工作，适用于对实时性要求高于正确性的应用；

没有流量控制和拥塞控制，可以按照设定的速度发送分组。

UDP报文首部（4\*2Bytes）：源端口号、目的端口号、包括首部的长度、校验和

UDP检验和：按每16bits将所有内容求和取反。求和时，溢出要回卷。

检验通过：检验范围（要回卷）+检验和=全1 （通过检验不一定没有差错）

3.4 可靠数据传输原理

rdt：reliable data transfer 可靠数据传输

FSM：finite-state machine有限状态机

rdt1.0：下层信道可靠

rdt2.0：下层信道可能出现比特差错。增加差错控制编码ACK、NAK，出错重传。

rdt2.1：ACK、NAK可能出错。一旦出错，接收方无法区分来的分组是出错重传的还是新的分组。增加序号进行区分。

rdt2.2：省去NAK。

rdt3.0：下层信道可能出现比特差错以及丢包。增加定时器，超时重传。定时器过早超时，会使传输效率减小。

流水线：利用率计算、需要增加序号范围、需要设置缓存

流水线的差错恢复方法：

序号空间：用来区分分组的所有序号，k比特

窗口长度：能同时等待确认的序号数，N

GBN：Go-Back-N 回退N步 （又称为滑动窗口协议）

累积确认，单个时钟，重传未ACK的全部分组；乱序不接收，ACK最后一个正确分组，无接收缓存。

SR：selective repeat 选择重传

单个确认，独立时钟，选择重传；乱序缓存，对应每个分组给出ACK。

最大窗口长度计算：考虑序号从0开始的一次传输的全部ACK都丢失时，接收方窗口包含的最后一个序号应小于发送方窗口包含的最前面的序号。

GBN：N≤2^k-1 得 N≤2^k-1

SR：2N-1≤2^k-1 得 N≤2^(k-1)

3.5 TCP

TCP：transmission control protocol 传输控制协议

全双工服务，TCP将应用层传递的数据看作字节流。

MSS：maximum segment size 最大报文段长度

MTU：maximum transmission unit 最大传输单元（最大链路层帧长度）

注：MSS取决于MTU，一般以太网MTU为1500字节，除去TCP、IP头部共40字节后，MSS为1460字节。

TCP报文段结构：

一般为20字节头部，包括：源端口号、目的端口号、序号、确认号、接收窗口、首部长度、标志字段、检验和。

TCP实现的服务：可靠数据传输、流量控制、拥塞控制

可靠数据传输：

序号：报文段首字节在字节流中的编号

确认号：期望下一次收到的字节的序号

注：TCP接收方如何处理乱序的分组，与具体程序有关

往返延时估计与超时定时器设置：

延时估计：

每隔一段时间测量某个分组的RTT（如果该分组是重传的，则不计入）得到SampleRTT。按照公式更新：EstimatedRTT = (1-a) EstimatedRTT + a SampleRTT (推荐a=0.128)

上面的平均方式称为指数加权移动平均。

偏差：DevRTT = (1-b) DevRTT + b |SampleRTT - EstimatedRTT|

定时器设置：

TimeoutInterval = EstimatedRTT + 4 DevRTT

快速重传：在没有超时的情况下，如果收到对方对于某一段的3个冗余ACK，则立即重传（包括非冗余的ACK，共4个相同的ACK）

流量控制：

作用：消除发送方使接收方缓存溢出的可能性。

考虑A给B发送数据：

RcvBuffer：主机B的接收缓存区大小

rwnd：主机B的空闲接收缓存。B在给A的ACK中会同时告知A当前的rwnd。

LastByteSent - LastByteAcked：主机A已发送但未确认的字节数

流量控制要求主机A：LastByteSent – LastByteAcked ≤ rwnd

注：当A收到rwnd=0时，必须继续发送只有1个字节数据的报文段。B在rwnd=0时，收到了A的小报文段会回复A它当前的rwnd，直到rwnd非0，传输得到恢复。

TCP连接管理：

1 SYN 初始序号

2 SYNACK 初始序号 确认序号

3 ACK 确认序号

3.6 拥塞控制

作用：防止过多的数据传输导致网络拥塞

检测拥塞：超时，3个冗余ACK。

速率控制：LastByteSent – LastByteAcked ≤ min{ rwnd,cwnd }

cwnd: congestion window 拥塞窗口

策略：目标：不拥塞且充分利用带宽

超时或3个冗余ACK，则cwnd减小

ACK，则cwnd增大

不断试探上限

慢启动：（slow start）

cwnd从1个MSS开始，每次收到确认就增加1个MSS（相当于每过一个RTT，MSS翻倍）。直到出现超时事件，设置一个变量ssthresh = cwnd/2 （慢启动阈值）再次开始慢启动。cwnd等于ssthresh时进入拥塞避免。

拥塞避免：

每个RTT只将cwnd增加1个MSS。出现超时，则ssthresh = cwnd/2 cwnd=1 进入慢启动；出现3个冗余ACK，则ssthresh = cwnd/2 cwnd=cwnd/2+3MSS 进入拥塞避免。

TCP的公平性：

加性增，乘性减，最终趋于平等的共享带宽