Week1 Review Notes

ELEC0099: Introduce to Internet Protocol Networks 21/22

RUFENG DING

Contents

1 Introduce to the Internet **2**

1.1 History . . . 2

1.2 Internet design principles . . . 3

1.3 Future applications . . . 4

1.4 Examples of Networks . . . 4

2 Computer Networking **5**

2.1 Types of Networks . . . 5

2.1.1 LAN . . . 5

2.1.2 WAN . . . 5

2.1.3 MAN . . . 6

2.1.4 Home Networking . . . 6

2.1.5 Low Power Networking: LORA and Narrowband IoT . . . 7

2.1.6 Personal Area Networks . . . 7

2.1.7 Space Networking: Interplanetary Internet . . . 7

2.2 Types of Addressing . . . 7

2.3 Quality of Services metrics . . . 8

3 The OSI stack **9**

3.1 Connection-Oriented vs connectionless . . . 9

3.2 Understand the need for Layering . . . 9

3.3 OSI model and layers functions . . . 10

3.3.1 Physical Layer 11
3.3.2 Data Link Layer:Ethernet Frame 11
3.3.3 Network Layer 12
3.3.4 Transport Layer 12
3.3.5 Session Layer 13
3.3.6 Presentation Layer 13
3.3.7 Application Layer 13
3.4 How to connect networks 14

1 Introduce to the Internet

The Internet contains three main parts:

1. Connected computing devices:hosts/end-systems
   PCs,workstations,servers,smartphones,toasters, running network apps.
2. Comunication links
   fiber,copper,radio,satellite(**transmission rate = bandwidth**)
3. routers:forward packets
   chunk of data

1.1 History

1. 1961 *Leonard Kleinrock* create the Queuing Theory
2. 1965 *Bob Taylor and Larry Roberts* start tge ARPANET project
3. 1965 *Donald Davies (UK)* invents the packets switching
4. 1972 ARPAnet demonstrated publicly
   - NCP (Network Control Protocol) first host-host Protocol
   - first e-mail program
   - ARPAnet has 15 nodes
5. 1970 ALOHAnet: a wireless network to connect islands in Hawaii
6. 1973 Ethernet *Robert Metcalfe's PhD Thesis*

7. 1974 *Cerf and Kahn* define the principles for interconnecting networks (First network outside USA is in UCL)

    - minimalism,autonomy - no internal changes required to interconnect networks
    - best effort service model
    - stateless routers
    - decentralized control

8. 1983 TCP/IP and DNS
9. 1988 TCP Congestion Control
10. 1989 100,000 hosts a lot of new national networks
11. 1991 NSF allows for commmercial use
12. 1992 The World Wide Web developed by *Tim Berners Lee* in CERN
13. Late 90's New applications come out : p2p file sharing / Network security / Voice over IP
14. 2000 Web 2.0 Interactive app over-the-top applications (OTT)
15. 2007 iPhone use Cellular access (Cellular already dominant but not used in mobile devices)
16. Today's trends

    - video streaming
    - distribution network
    - clouds
    - software defined networks
    - privacy

Internet Access Statistics World Internet usage and population statistics

1.2 Internet design principles

End-to-end argument: Important functions(error control, encryption, delivery, acknowledgements, etc) should be implemented by the end systems.

Fate Sharing: if one put functionality in the end system, then that functionality only breaks if the end system breaks which would make the communication useless anyway

Russian Dolls: a lot of packets inside packests

Packet Switching: Statistical Multiplexing: Sequence of packets do not have fixed pattern

Note 1. Statistical multiplexing is a type of communication link sharing, very similar to dynamic bandwidth allocation (DBA). In statistical multiplexing, a communication channel is divided into an arbitrary number of variable bitrate digital channels or data streams. The link sharing is adapted to the instantaneous traffic demands of the data streams that are transferred over each channel. This is an alternative to creating a fixed sharing of a link, such as in general time division multiplexing (TDM) and frequency division multiplexing (FDM). When performed correctly, statistical multiplexing can provide a link utilization improvement, called the *statistical multiplexing gain*.

1.3 Future applications

Video is King: Video will be 82% of the traffic in 2021. 4K TV is being deployded: 25Mbits/s (5 times more than HD). 8K is next: 4 times more pixels and there will be more end-users.

Vitual Reality.

Augmented reality.

Holograms.

The Smart Home.

Transport.

Smart cities.

Internet of things (IoT) and Machine2machine (M2M).

5G.

1.4 Examples of Networks

JANET(The UK education network)

Topologies: Topology-zoo

2 Computer Networking

2.1 Types of Networks

2.1.1 LAN

Local Area Network. Connected computers that are physically close together.(<1 mile) eg. Data Center Networking
Characters:

- high speed
- multi-access

Technologies:

- Ethernet (10Mbps,100Mbps,1Gbps)
- Token Ring 16Mbps
- FDDI 100Mbps

2.1.2 WAN

Wide Area Network.Connects computers that are physically far apart."long-haul network".
Characters:

- Higher delay that LAN / Fast speeds
- Traditionally less reliable than a LAN
- point-to-point

Technologies:

- Telephone lines
- Satellite communications
- SONET/SDH - ATM
- Increasingly Ethernet

2.1.3 MAN

Metropolitan Area Network **Characters**:

- larger than LAN and smaller than WAN
- Campus-wide network (FDDI,DQDB,ATM)
  - FDDI: Fiber Dsitributed Data Interface
  - DQDB: Distributed-Queue Dual-Bus
  - ATM : Asynchronous Transfer Mode
- Interconnects LANs

Technologies:

- Coaxial cable
- Microware
- Optical

eg. A big MAN : LinX : The london internet eXchange point

2.1.4 Home Networking

New challenges and opportunities
Access:

- ADSL
- Optical fiber
- Wireless, Satellite

Access inside the house:

- Wifi, Bluetooth, power cables
- Wifi signal, wifi throughout, wifi leaking
- Internet of things(wifi zigbee)

2.1.5 Low Power Networking: LORA and Narrowband IoT

Characters:

- Long distance
- 9v battery
- up to 10 years
- Bytes per minutes

2.1.6 Personal Area Networks

A niche market but an important one.
Characters:

- Healthcare and Internet of things
- Bluetooth dominates (Wifi low energy may become relevant)
- energy is critical!(You don't want to charge these devices all the time)

2.1.7 Space Networking: Interplanetary Internet

To sustain life and travel to other planets and it has a very high delay (eg. Earth-Mars: 4 to 24 minutes)

2.2 Types of Addressing

There are 4 addressing modes:

- **unicast**: message sent to one destination
- **broadcast**: message sent to all hosts in a network
- **multicast**: message sent to all members of a group
- **anycast**: message sent to closest member of a group

2.3 Quality of Services metrics

Throughput
Measured in bits per second and it depends on:

- Transmission rate of the links in the path
- packets loss
- packets delay
- application requirements

Packet loss
caused by:

- congestion in the network
- active queuing policies
- link failures

Delay
some applications require low end-to-end delay:

- vocie: 150ms
- interactive services: 10ms
- financial services: each ms costs money

Five components:

- Propagation
  dominated by the speed of light on fiber (aprox 210,000 km/s) and increased by the fact that fibers do not follow straight lines.
- Queuing
  Each link has a queue. For lightly loaded links never more than 10 packets. (overloaded links can have 100ms delay)
- Processing(eg, middleboxes)
  Routers, wireless points, end hosts takes times to analyse the packets and determining what to do with them. Routers need to check for error, determine next link, etc. In normal conditions the processing delay is very small but if data processing is involved, it can be a major source.

- Packetzations(Increasing by lower layers)
  Time to require the data to be sent. (eg. Voice to be encoded. If it's coded at 32 Kbps you have to wait 20ms to get 80 bytes) Lower layers: ADSL interleaving, used to "shuffle" fragments to increase reliability. And it add about 5ms of delay.
- Transmission(data rate)
  Determined by data rate links. (eg. On 1 Mbps link takes 10ms to send a 10000 bit packet/ On 10Gbps takes 1μs on 50Kbits/s, 200ms)

3 The OSI stack

3.1 Connection-Oriented vs connectionless

connection-oriented services uses *circuits* a single path is first established for each new connection(call setup/call release) and the *network* guarantees that the data are delievered in order, no loss or duplication.

If any thing goes wrong the connection is broken. It is possible to limit the number of connections. The network can guarantee bandwidth at connect time.(waste of bandwidth, if resources not used) And the network can refuse new connections.

eg. Telephony(PSTN), cellular network, ISDN, ATM

Connectionless Service uses *datagrams*. Each datagram is independently routed and includes the destination address.

No guarantee that the datagrams are delievered in order and are not lost or duplicated. It is direct transmission of data(no call set-up). And best-effort earnest attempt to deliver.

Resources are shared and little waste of bandwidth. If the network overly utilized, further traffic is still allowed.

eg. Postal Service, Ethernet, Internet Protocol(IP)

CO vs CL Services

3.2 Understand the need for Layering

Divide a task into separate functions and then define each piece independently.

Eastablishing a well defined interface between layers makes porting easier (Decoupling).

Major advantages:

Table 1: CO vs CL Services

	Type	Pros/Cons
PROS	CO	reliabilitym file transfer and terminal traffic main applications Faster forwarding, after call setup the path is constant Better to lock-out further cells, than to degrade service "Simple" Terninal Equipment, offload complexity to the core network
	CL	Fault tolerant, if link fails other paths available Applications like voice and video can tolerate datagram loss Better suited to bursty traffic, link reservation is a waste Fair, better allow user access, not only some lucky users Efficient for client-server applications with hundreds of clients "Simple" core network

- Code Reuse
- Extensibility
- Division of tasks. Each engineering implementing a layer does not need to know how the lower one is implemented.

3.3 OSI model and layers functions

OSI Protocol Reference Model

Parallel work to hte Internet development started in 1970s. It published its first version in 1984 by ISO(International Standard Organization). It consist two parts:

- reference model organizing networking functions in 7 layers
- a set of specific protocols for each layer(never really deployed)

It was thought to be the "serious" standard.

Table 2: OSI Protocol Reference Model

No.	Layer	function	scope
7	application	application-specific protocols	end-to-end
6	presentation	data representation and encoding	
5	session	dialog and sychronisation	
4	transport	message transfer(error/flow/congestion control,CM)	
3	network	network routing/addressing(CM)	global
2	data link	data link control(framing,data transparency,EC)	local
1	physical	mechanical/electrical/optical interface	

CM: Connection management EC: Error control

3.3.1 Physical Layer

Responsibility:transmission of raw bits over a communication channel. (Encode format /cable types etc.)

Issues:

- mechanical and electrical interfaces
- time per
- distances

> **Note 2. Manchester Code** is a line code in which the encoding of each data bit is either low then high, or high then low, for equal time. It is a self-clocking signal with no DC component. Consequently, electrical connections using a Manchester code are easily galvanically isolated.

And there are lots of digital encoding format: NRZ-L/NRZI/Bipolar-AMI/Pseudoternary etc.

3.3.2 Data Link Layer:Ethernet Frame

Ethernet uses a method called **Carrier Sense Multiple Access with collision Detection** (CSMA/CD).

In Ethernet the medium is shared, computers **sense** the medium to check if anybody is tranmitting and if not, they start sending. While sending they check if the wave is changed. If it does than a **collision is detectd**. The computer waits a random amount of time and then retransmits again.

Another method is called the **Carrier-sense multiple access with collision avoidance** (CSMA/CA). It is uesd in Wifi/802.11. The Sender need to sned RTS to Revceiver to get a CTS and then send the Data packet. When finished get ACK from Receiver.(hand shake)

eg. EPC (Electronic Product Code) RFID reader and tag.

Error Detection
Networks are unrelable so the data link can help here:

Error correcting codes:

1. Hamming codes
2. Binary convolutional codes
3. Reed-Salomon codes
4. Low-Density Parity Check codes

Error detecting codes

1. Parity
2. Checksum
3. Cyclic Reducndancy Code (CRC)

3.3.3 Network Layer

Responsibilities:

- Path selection between end-systems(routing).
- Subnet flow control.
- Fragmentation and reassembly.
- Translation between different network types.

Issues:

- *packet* headers
- virtual circuits

Non-IP network layer: (historically) ATM, CLNP. There are other design choices include: admission control/ congestion control/ reliability.

3.3.4 Transport Layer

Responsibilities:

- provides virtual end-to-end links between peer processes.
- end-to-end flow control/ congestion control.
- reliable communication.

3.3.5 Session Layer

Responsibilities:

- Dialog Control(Keep track of whose turn is it to transmit)
- Token Management(preventing both parties from attempting the same critical operation simultaneously)
- Synchronization(checkpointing long transmission to allow time to pick up from where they left the event of a crash and subsequent recovery)

In TCP/IP it does not exist formally. Many people identify some functions provided by some protocols as being part of **Session Layer**. eg:

- RTP and RTCP for video conferencing
- Cookies for http
- SIP -Session Initiation Protocol
- ...

Session layer functionality is usually implemented by the application itself.

3.3.6 Presentation Layer

Responsibilities:

- data encryption
- data compression(lossless or lossy)
- data conversion

Many protocol suits do not include a Presentaion Layer.

Encryption Two kinds: Symmetric and Asymmetric. It allows for confidentiality and authentication and the management of public keys is the hardest part.

3.3.7 Application Layer

Responsibilities: anything not provided by any of the other layers.

Issues:

- application lebel protocols
- approprate selection of ”type of service”

Examples of application layer protocols:

- **HTTP** for the world wide web. This is the protocol used for web brosers to get web pages form web servers.
- **SMTP,POP,IMAP** to read and receive email.
- signaling protocols like **SIP** for voice over **IP**.
- **Routing protocols** are application level protocols that affect the Network Layer.

They are very easy to implement in any programming language.

3.4 How to connect networks

- Repeater: physical layer
- Bridge: datalink layer
- Router: network layer
- Gateway: network layer and above

Repeater copies bits form one part of the sme network to another and it does not look at any bits, just regenerates them. It allow the extension of a network beyond physical length limitations.

Hub is a star topology - multi-access device. It regenerate bits and provide muti-port device linking network segments. It copy all vilid data to all ports and allow the extension of a network beyond physical length limitations.

Bridge copies frames from one part of a network to another. And it can operate selectively - does not copy all frames(must look at data-link headers). Provides isolation and so improves performance. It can extends the network beyond physical length limitations.

Router transfer packets from one network to another. Makes decisions about what *route* a packet should take (looks at network headers).

Table 3: Criticisms about OSI and TCP/IP

OSI	TCP/IP
bad timing	not distinguish clearly between service/interface/protocols
bad technology/complex standard	not easily exatendatble/lacks generality
bad implementations	link layer not well defined
bad politics	many protocols have bad implemantations

Two elephants lead the world using TCP/IP (OSI using too many times define the standard).