Module 2: TCP/IP Layered Architecture

Compiler: Sarah Meng Li

Definition 2.0 The TCP/IP Reference Model (simple naming and addressing scheme)

TCP/IP means **Transmission Control Protocol and Internet Protocol**. It is the network model used in the current Internet architecture as well. **Protocols** are set of rules which govern every possible communication over a network. These protocols describe the movement of data between the source and destination or the internet. They also offer simple naming and addressing schemes.

Notes: This summary is made based on lecture notes of CSCI 3171 Network Computing taught by Dr. Srinivas Sampalli in fall 2019.

Network Architecture

Definition 2.1: Network architecture defines a framework or a blue print for the design of networks.

Remark: with all the functions that need to be provided in a network, its design can be a daunting task. Hence, network architecture takes a layered approach.

Definition 2.2: Layering refers to splitting up network functions into easily manageable sub-functions, delegating responsibilities, and using abstractions to hide complexity. The **layer model** is also referred to as a **protocol stack** or **suite**.

Properties 2.3

1) Each layer performs a set of well-defined functions.
2) Each function is implemented by a set of protocols.
3) Each layer has an interface with its neighboring layers.

TCP/IP Protocol Suite

Definition 2.4: TCP/IP is the industry standard network architecture for network connected to the Internet.

1) There are many other network architectures (that run on legacy systems). However, all hosts connected to the Internet must have the TCP/IP architecture.
2) TCP/IP divides all network functions into five layers.

Layer 5	Application	software
Layer 4	Transport	
Layer 3	Network	
Layer 2	Data Link	hardware
Layer 1	Physical	

TCP/IP and Message Transfer

Process 2.5

- Every message entity begins its journey at the Application Layer in the sending

host (Host A)

- It trickles through each of the five layers and exists at the Physical Layer.
- At the receiving host, the message makes its way upward from the Physical Layer to the Application Layer.
- Each interconnection device that the message goes through in the network takes the message up through multiple layers and back down.
  - Repeater: Layer 1 only
  - Bridger/Switch: Layer 1 and 2
  - Router: Layer 1, 2, and 3

Data Encapsulation/Decapsulation Principle

Process 2.6

- The three core layers (Transport, Network, and Data Link) each add a header to each data chunk on its journey down the **protocol stack**. Note that the **data link lay** adds a trailer as well.
- The headers and trailers are removed when the data chunk moves up the protocol stack.
- **The header and trailer contain all the important info to perform the network functions.**

Functions of Layers

2.7 Application Layer Functions: User interface to network services data preparation

- Allow users to access the network
- Data conversion between formats
- Data compression and encryption
- **Example services**
  - File transfer, access and management
  - Mail services
  - Directory services (distributed database sources and access)
  - Network management
  - Network virtual terminal (remote access)

2.8 Transport Layer Functions: End-to-end transport between hosts

- Segmentation and reassembly
  - Divides message stream into segments and vice versa
- Connection control
  - Establish connection in a connection-oriented service
- Flow control
  - Provide technique to control the data from the sender so that the receiver buffer does not get overloaded
- Error control
  - Provide technique to retransmit lost or corrupted segment

Remark: Internet applications can be classified into two types of services: connection-oriented service and connectionless service.
Remark: The transport layer is responsible for connection set up and connection tear-down in a connection-oriented device.

2.9 Network Layer Functions: Network to network data transfer

- Concerned with delivery across multiple networks
- Path determination and routing are the main functions
- **Using logical addressing—IP Addresses**

2.10 Data Link Layer Functions: Data Transfer on a network

- Medium access control
  - Technique to regulate access to a link
- Error detection and control
  - Frame reliability
- Flow control
  - Regulate flow of frames
- Users physical addressing
  - **MAC addresses or Ethernet addresses**

2.11 Physical Layer Functions: Interface to the link

- Bits to signal conversion and vice versa
- Defines bits representation and data rates
- Define characteristics of interface between the device and the link

Def 2.12 Connection-oriented service

- Involves a **connection setup** phase, **message exchange** phase, and **connection tear-down** phase.
- Receiver needs to be aware/ready.
- Usually a reliable service (retransmission is initiated in case of failure)
- **Example:** HTTP, FTP, Telnet, Voice over IP

Def 2.13 Connectionless service

- No connection is set up
- Each message or packet is transmitted independently.
- Receiver need not be aware/ready
- Retransmission may not be initiated in the event of failure and delivery may not be guaranteed.
- **Example:** Network management messages (SNMP), DNS messages, trivial FTP (TFTP), traceroute

Protocols in TCP/IP

TCP/IP Layers	Function in a nutshell	Example protocols
Application	User interface to network	**HTTP**, HTTPS, SSH, FTP, Telnet, SFTP, NFS,

	interface	SMTP, SNMP
Transport	End-to-end data transfer	**TCP**(connection oriented), UDP (connectionless)
Network	Network-to-network data transfer	**IP** (used to look up the routing table), Routing: RIP (used to build routing table), OSPF, IGRP, EIGRP Support: ICMP, IGMP, ARP, RARP
Data link Physical	Data transfer on a network and interface to link	LAN (short distance):**IEEE 802.3** (Ethernet), IEEE 802.5 (Token Ring), IEEE 802.11 (WiFi) WAN (long distance): SLIP, PPP, FR

Data Encapsulation at Sender

Layer 5	Application	software
Layer 4	Transport	
Layer 3	Network	
Layer 2	Data Link	hardware
Layer 1	Physical	

Application			Data		
Transport			TCP/UDP Header (TH)	Data	
TH Segment (port address)					
Network		IP Header	TCP Header (TP)	Data	
Packet/Datagram (IP address and protocol ID)					
Data Link	802.3 Header	IP Header	TCP Header	Data	802.3 Trailer
Physical	Frame (MAC address and typecode)				

Addresses

Three sets of addresses for setting up a unique connection between two processes"

(1) Port addresses: source and destination port number

- The client chooses a random port number in the range 49152 to 65535. These are called **ephemeral/dynamic port numbers**.
- The source uses a standard port number in the range 0 to 1023. There are called **standard port numbers**.
- The port numbers between 1024 and 49151 are called **registered port numbers**. They are used for services such as GoogleTalk, Skype, Network game servers, etc.
- These addresses can be found in Transmission Control Protocol part of the packet detail pane in Wireshark.

(2) IP addresses
    - Normally, the original source and the final destination IP addresses are used. One scenario of exception to this rule is when the destination is hidden behind a firewall.
    - These addresses can be found in Internet Protocol part of the packet detail pane in Wireshark.

(3) MAC addresses
    - These changes from network to network and only have local significance.
    - These addresses can be found in Ethenernet part of the packet detail pane in Wireshark.