Chapter 1

1.1

When one end system has data to send to another end system, the sending end system segments the data and adds header bytes to each segment.

The resulting **packages** of information, known as **packets** in the jargon of computer networks, are then sent through the network to the destination end system, where they are reassembled into the original data.

A packet switch takes a packet arriving on one of its incoming communication links and forwards that packet on one of its outgoing communication links.

Receiving End system

|  |  |  |
| --- | --- | --- |
| header byte | header byte | header byte |

Sending End system

Packet switches

One End system

Sending through network

Segments of data

= Packages of information

= Or packets in nextwork

Packet switches take incoming pakets and forward them to ultimate destinations. Most of them us **store and forwar**d transmission at input links.

Link-layer switches are typically used in access networks

**Routers** are typically used in the network core. It is a part of internet protocol. Every router keep track of multiple paths for sending packets and it choooses the “cheapest” available path for each piece of data based on destination IP address for the packet.

The sequence of communication links and packet switches traversed by a packet from the sending end system to the receiving end system is known as **a route or path** through the network.

Store-and-forward transmission means that the packet switch must receive the entire packet before it can begin to transmit the first bit of the packet onto the outbound link

Packets = trucks

Communication links = highways and roads

Packet switches = intersections

End systems = buildings

Just as a truck takes a path through the transportation network = a packet takes a path through a computer network.

End systems, packet switches, and other pieces of the Internet run protocols that control the sending and receiving of information within the Internet. The Transmission Control Protocol (TCP) and the Internet Protocol (IP) are two of the most important protocols in the Internet. The IP protocol specifies the format of the packets that are sent and received among routers and end systems. The Internet’s principal protocols are collectively known as TCP/IP

**Bandwidth** : the maximum transmission capacity of a device. Bandwidth is measured **by bit rate**: number of bits that we can send over a given period of time, usually measures in second.

Different measure of speed is the **latency** or the time it takes for **a bit** to travel form one place to another.

**The Transmission Control Protocol (TCP):** managing the sending and recieving of all your data as packets. It is liked a guaranteed mail service. When packets arrive, TCP does a full inventory check and sends back acknowledgements of each packets received. If all packets are there, TCP signs for your delivery and you are done. If TCP found some missing or incomplete TCP will send them again. The more routers the more routers we add the more reliable becomes.

SSL : secure sockets Layer

Protect from snooping or tampering.

TLS: transport layer security

* When these are active we see this lock:



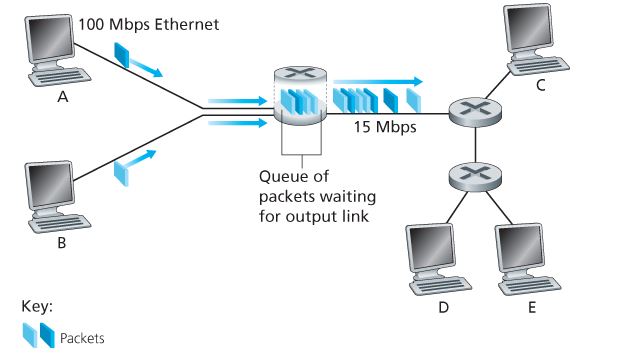
**HTTP:** hyper text transfer protocol

**HTTPS:** hyper text transfer protocol secure

**host = end system**.

Hosts are sometimes further divided into two categories: clients and servers. Informally, clients tend to be desktop and mobile PCs, smartphones, and so on, whereas servers tend to be more powerful machines that store and distribute Web pages, stream video, relay e-mail, and so on

**Queuing delays and packets loss**



If an arriving packet needs to be transmitted onto a link but finds the link busy with the transmission of another packet, the arriving packet must wait in the **output buffer**. Thus, in addition to the store-and-forward delays, packets suffer output buffer **queuing delays**.

An arriving packet may find that the buffer is completely full with other packets waiting for transmission. In this case, **packet loss will occur**—either the arriving packet or one of the already-queued packets will be dropped

A router uses a packet’s destination address to index **a forwarding table** and determine the appropriate outbound link.

The Internet has a number of special **routing protocols** that are used to automatically set **the forwarding tables. A routing protocol** may, for example, **determine the shortest path** from **each router to each destination** and use the shortest path results to configure the forwarding tables in the routers.

There are two fundamental approaches to moving data through a network of links and switches: **circuit switching and packet switching**

|  |  |
| --- | --- |
| **circuit switching** | **packet switching** |
| -the resources needed along a path (buffers, link transmission rate) to provide for communication between the end systems are reserved for the duration of the communication session between the end systems.  EX: the restaurant that requires reservations, we have to go through the hassle of calling before we leave home. But when we arrive at the restaurant we can, in principle, immediately be seated and order our meal. | - these resources are not reserved; a session’s messages use the resources on demand and, as a consequence, may have to wait (that is, queue) for access to a communication link.  EX: the restaurant that does not require reservations, we don’t need to bother to reserve a table. But when we arrive at the restaurant, we may have to wait for a table before we can be seated. |

**How the packet-switching knows which outbound link forward the packet?**

* Each packet from source that has been sent contains the destination IP address in its header and Each routers has a forwarding table that maps destination address to that outbound link. So when the packet comes to the router, the router will check on the forwarding table to pick an approriate outbound link.

**How the forwarding table was set?**

* The internet has a number of special routing protocols that are used to automatically set the forwarding tables. A routing protocol may determine the shortest path from each router to each destination and use the shortest path results to configure the forwarding tables in the routers.

**A circut in a link is implementd with either FDM(frequency-division multiplexing) or TDM(time-division multiplexing)**

Class Notes:

If we have 10 users busy so FDM is better over TDM

* FDM each circuit gets a fraction of the bandwidth
* TDM each circuit gets all of the bandwidth periodically during brief intervals of time (that is, during slots)
* Multiplexing is a technique through which several signals are concurrently transmitted over a single data link

Chapter 2

1. Sockets:

Sockets and process is analogous to the door of the house. It is an interface between application layer and transport layer.

Application developer has more control on the application layer side rather than the transport layer.

A computer can run multiple processes at the same time, so to send and receive message processes must have unique identifier which consists of: IP address and port number. Since each process or application has been assign a specific port number.

1. Application layer protocols:

Application layer defines how an application’s processes run on a different end system, pass message to each other.

In particular application layer defines:

* Type of messages
* Syntax of messages
* Messages semantics: the meaning of information in the field
* Rule for when and how the process send and respond to messages

Non-persistent vs persistent HTTP

* Non-persistent HTTP:
  + Response time: 2RTT + time for file transmission time
  + 1 RTT to initiate TCP connection
  + 1 RTT for HTTP request and first few bytes of HTTP response to return
* Persistent HTTP:
  + Response Time: RTT + time for file transmission time.
  + As little as one RTT for all the referenced objects.
* Midterm exam there is question for midterm exam.

Chapter 3: Transport Layer

**How transport layer can provide reliable data transfer and handle congestion control.**

Problems to solve of this chapter:

* Reliability:
  + Performance
  + Loss
  + No Errors

**Solution:**

Version 1.0

Version 2.0

Version 2.1

Version 2.2

Version 3.0

RDT

Version 3.0

Pipelined

Network layer: logical communication between hosts

Transport layer: logical communication between processes on different hosts.

Transport layer Multiplexing/demultiplexing

Each host has an IP address

🡪Checksum question on midterm or final

Steps:

1. Add them together = result
2. Add result with 1 wraparound =result’
3. Flip all the bit of result’