OS CS219 Notes

August 1, 2024

Lecture 1

1 Introduction

What is a computer network? A computer network is a group of interconnected devices that can exchange data and resources with each other. ~~Almost~~ all of today's devices are in one way or another connect the biggest computer network alias the Internet. There are several abstractions and nuances that enable the existence of such a huge structure.

A network consists of several end hosts which are systems that request/receive data using the network. These end hosts are connected using links which can directly connect the hosts together or more commonly connect multiple of them to switches/routers which can simplify the network while still providing connectivity among hosts.

1.1 Topologies

A group of hosts can be connected in multiple ways. The type of graph that is obtained from considering the hosts as nodes and links as edges is called the 'topology' of that network. Some examples include a bus where all hosts are connected to a common wire. Others include star topology where hosts are connected to a central host.

Links can also be classified on the basis of how many users can communicate across them.

- **Simplex:** Only one user can talk across a link
- **Duplex:** Both users can communication *simultaneously* across a network.
- **Half Duplex:** Both users use the same link to communication but not simultaneously.

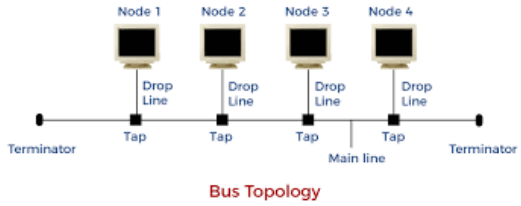

Figure 1: Bus Topology

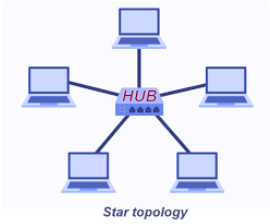

Figure 2: Star Topology

1.2 Switch

What exactly is a switch? The Switch is a network device that is used to segment the networks into different subnetworks called subnets. It can help simplify a network by grouping together lots of hosts into a sub-network. A switch has multiple incoming and outgoing links. It is capable of routing data from an incoming link to an appropriate outgoing link.

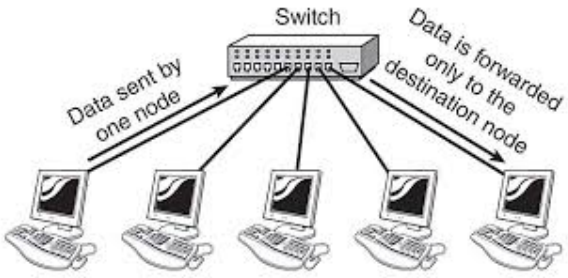

Figure 3: Example of a switch

2 Abstractions, Layering in a network

There is always the possibility to deal with the network as a single structure at once. That is to deal with the entire flow of data from the second a request is made by a 'user' and all the way till the request is serviced in one go.

However, this is very inconvenient and complicates the network in the sense that any change made to some part of the network can make or break the entire system. To combat this the network is clearly split into layers where each layer operates relatively independently and only exposes parts of it that are necessary for the higher and lower layers in the network.

To better understand this let us take an example.

The most common request on the Internet is an HTTP request (ie) a request by a

computer for a webpage. Let us see the flow of information when such a request is made.

1. **Application layer:** The URL is entered into a browser and then a user request is made. Then this URL is converted into the **IP**[1] address of the server that holds the page needed by the user.
2. **Transmission layer:** Now that we know the IP address from the application layer, the request for a page is sent to that address using the transmission layer. This layer sends the request message in manageable pieces to the network to be sent to the web server.
3. **Network layer:** Now these 'manageable pieces' need to be sent to the destination (ie) web server. The next router/link to which the message is to be sent is decided in this layer. These links 'talk' to each other in some sense and know where to send messages to reach the web server.
4. **Data Link layer:** The data link layer deals with splitting the message bit by bit and choosing the appropriate media to transfer them using (ie) Optic fiber, Wireless links etc..
5. **Physical layer:** Finally the physical layer deals with transmitting the actual bit signals over whatever media is chosen.

This by no means completely covers the functionality of each layer but rather gives a flavour of each layer's functions. It is easy to see how the abstraction is helpful as now the application layer has no need to worry about which media is used to transfer the bits and the Physical layer is oblivious to what message it is transferring.

The abstraction helps to simplify the structure of the network by helping us deal with one subproblem at a time.

Lecture 2

[1] will be dealt with later, assume it is some id for a computer