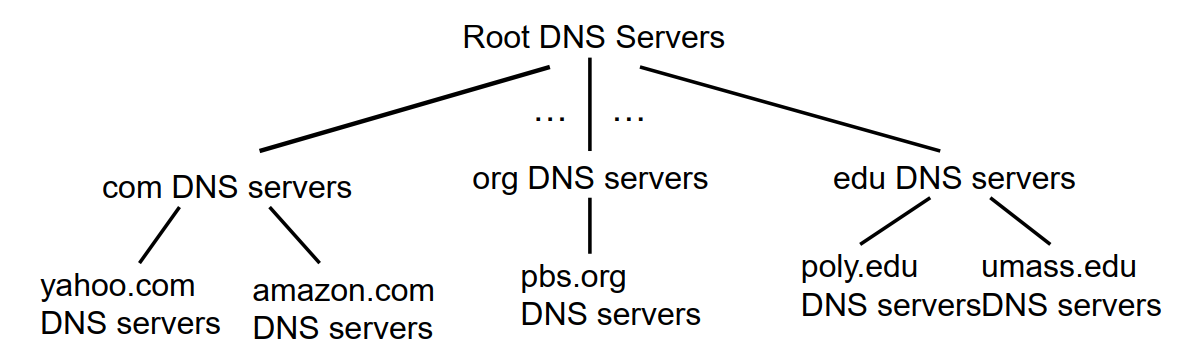


Domain Name System (DNS) is a distributed database implemented in hierarchy of many name servers.

DNS works at application layer protocol. Hosts and name servers communicate to resolve names. DNS protocol runs over UDP and uses port 53 .

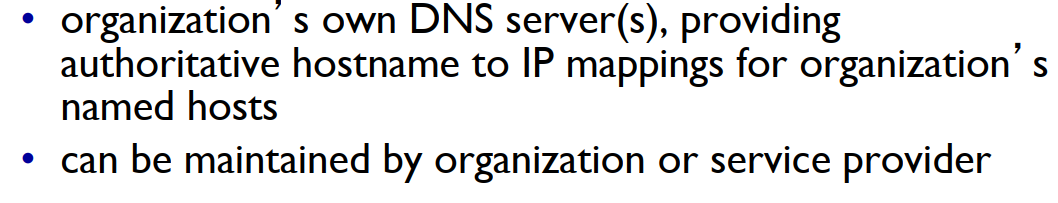
DNS categories



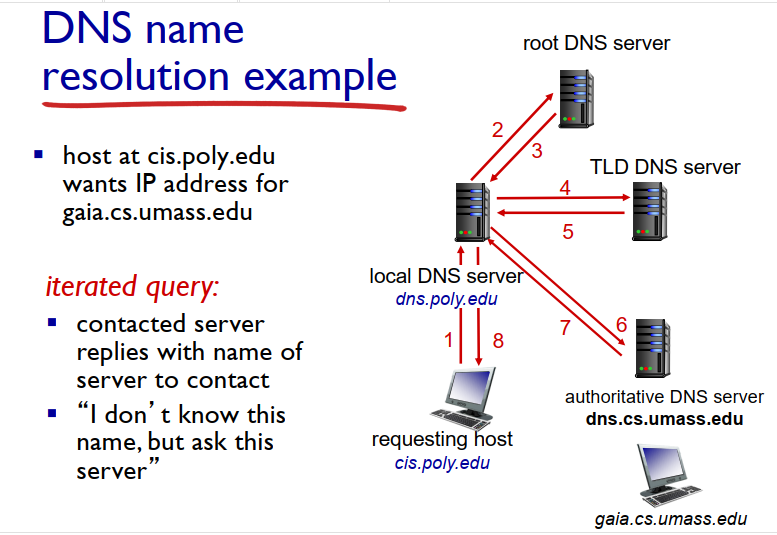
Root DNS servers which has updated record of TLD (Top Level Domain) DNS servers.

TLD : Here .com, .edu, .org are Top Level Domain.

Authoritative DNS Server: (Amazon.com, yahoo.com are Authoritative DNS server.

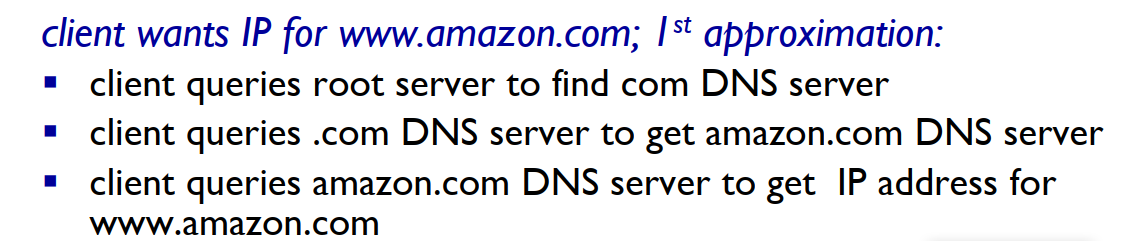


Steps of DNS name resolution:



Here, cis.poly.edu want to IP address of gaia.cs.umass.edu server.

1. Cis.poly.edu wants contact dns.poly.edu (local DNS server).
2. Then dns.poly.edu contacts root DNS server for .edu TLD server if local DNS sever does not have caching info of .edu TLD server.
3. Root DNS server send address of .edu TLD sever to Local DNS server
4. Local DNS server request IP address of dns.cs.umass.edu to TLD sever
5. TLD server sends the IP address of dns.cs.umass.edu to Local DNS server.
6. Local DNS server ask the IP address to gaia.cs.umass.edu to Authoritative DNS sever
7. Authoritative DNS sever sends the IP address of gaia.cs.umass.edu to Local DNS server.
8. Local DNS server sends the IP address of gaia.cs.umass.edu to requesting host cis.poly.edu



What are tracker torrents in P2P file distribution:

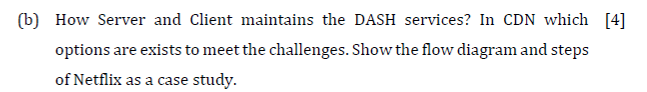
Tracker: Tracker tracks peers participating in torrent.

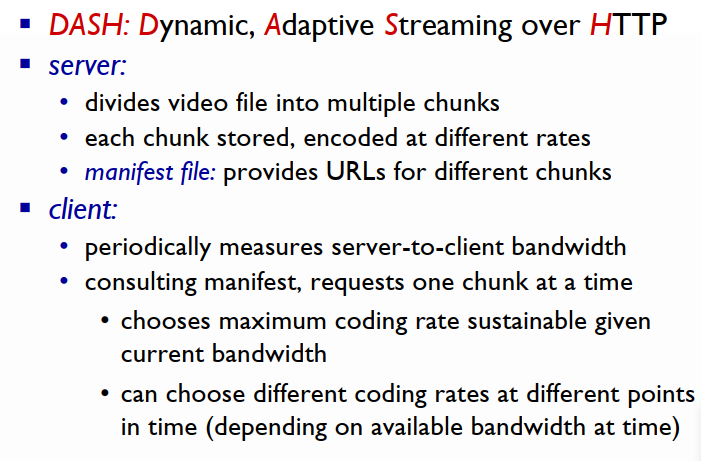
Torrent: group of peers exchanging chunks of file.

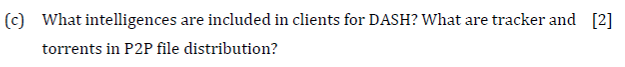
First, user download a torrent file which has all info of all a video file and what pieces are needed to complete. And tracker tracks the peer computers which has the pieces of targeted video.

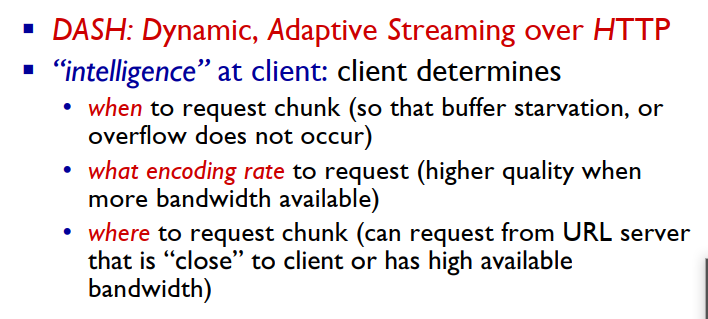
When numerous request come to a sever downloading a large file then the server will become unavailable due to network congestion, shortage of server resource etc. To resolve the problem the P2P (Peer to Peer) file distribution has been introduced.

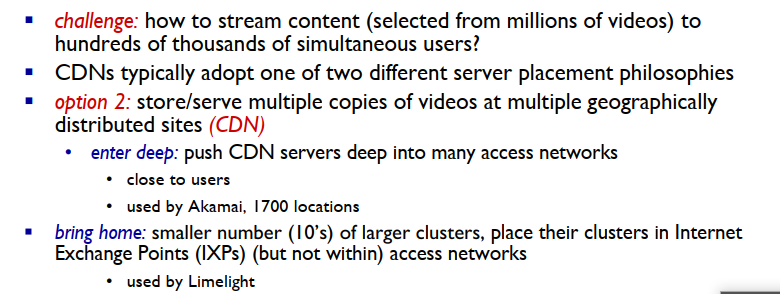
Here each client which use this peer system, then this client act as server to upload chunk of files to other clients. Here more clients means more sources to download which makes faster download. Clients (Group of peer or torrent)

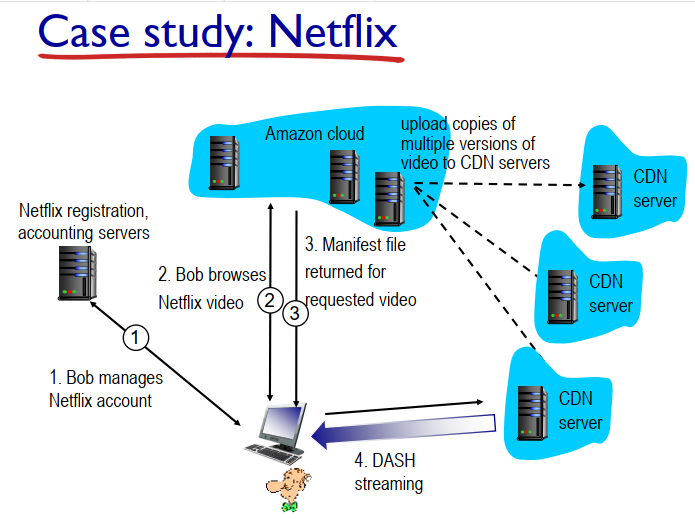




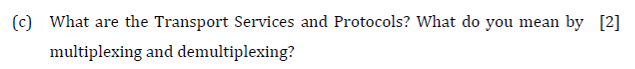


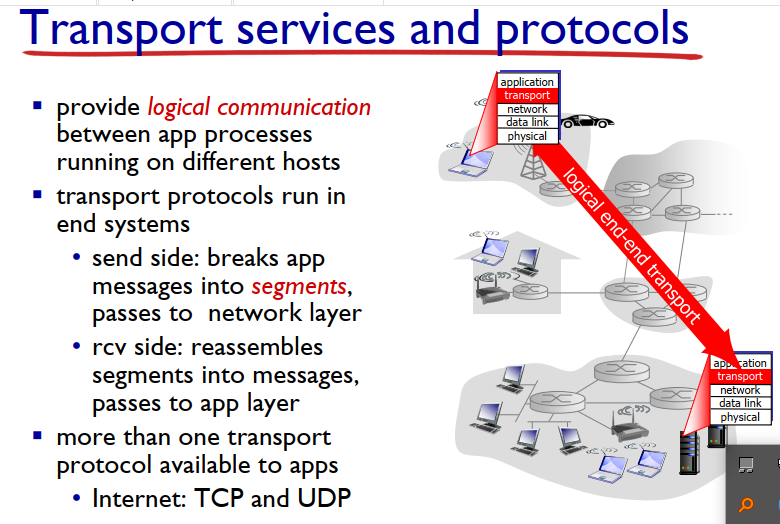




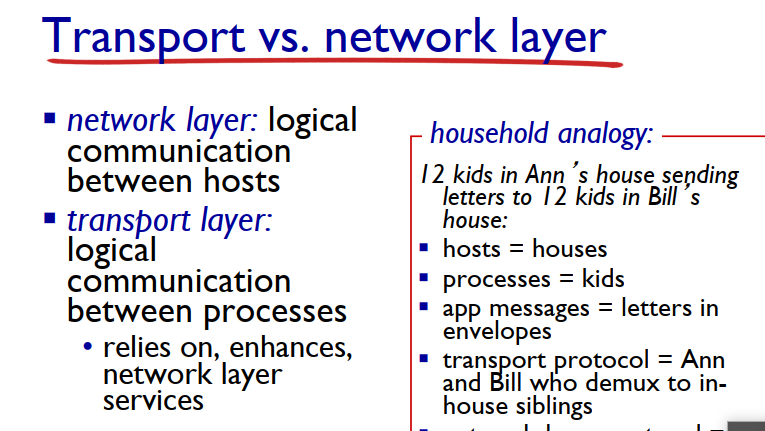


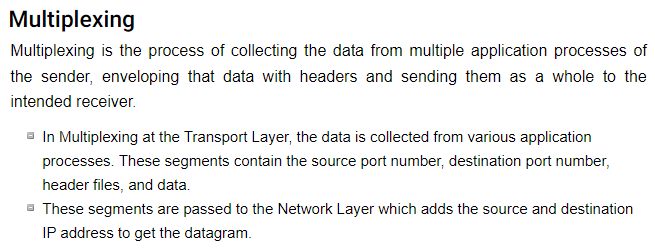
**Chapter 3**

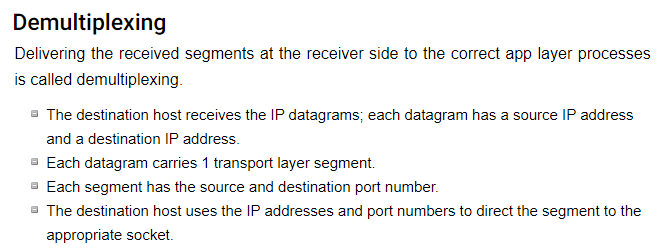


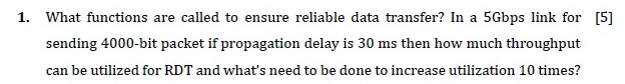


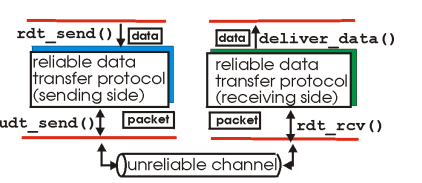
**Important**

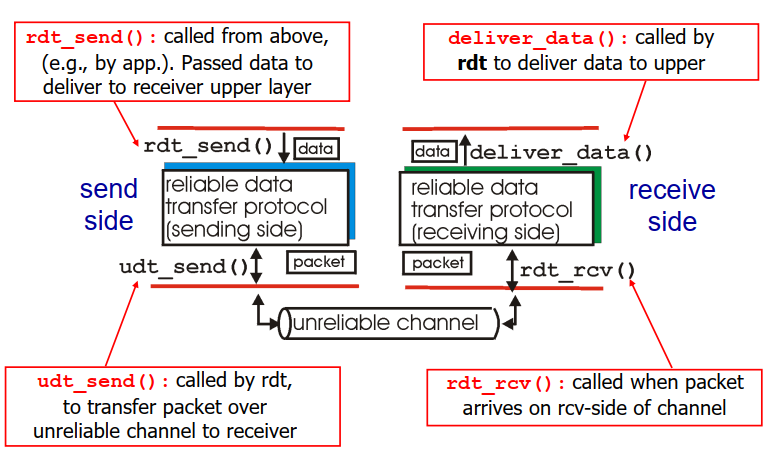


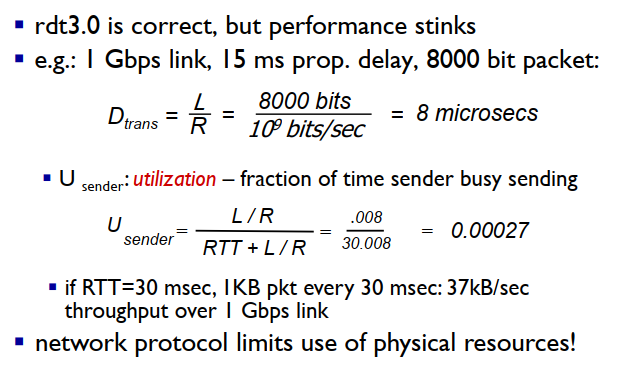




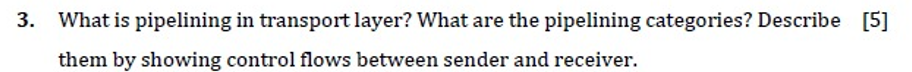


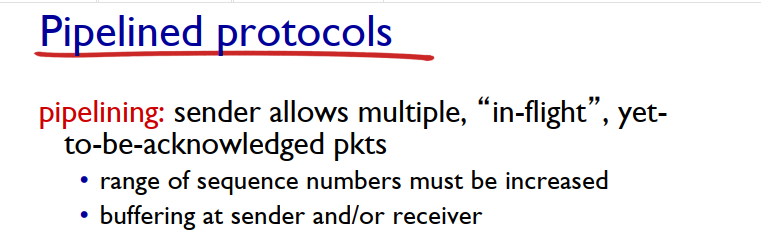




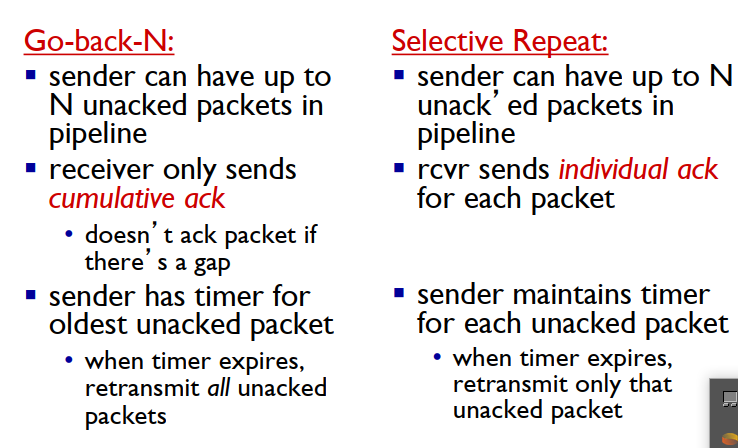


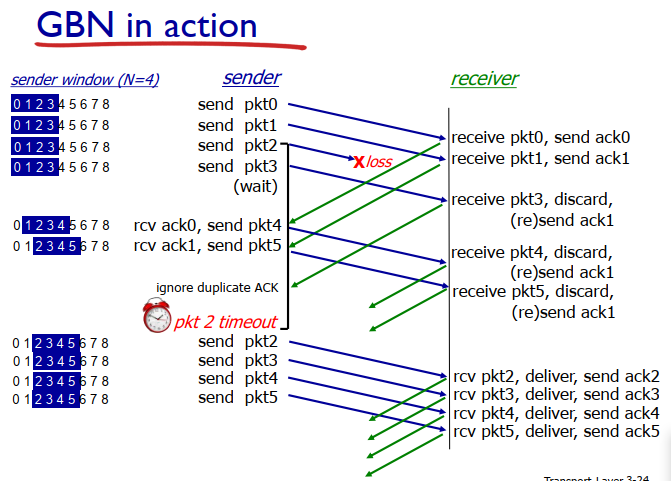
**Use Pipelining to increase utilization upto 3 times**

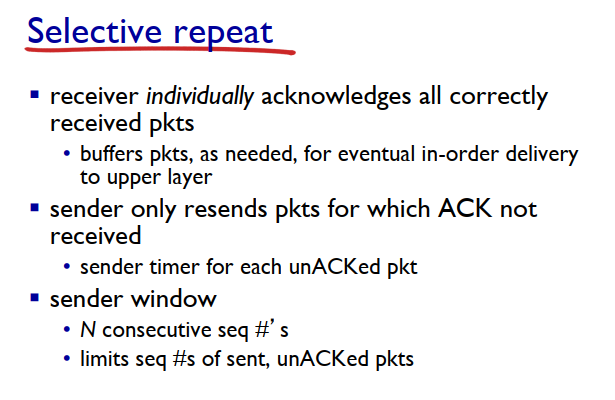


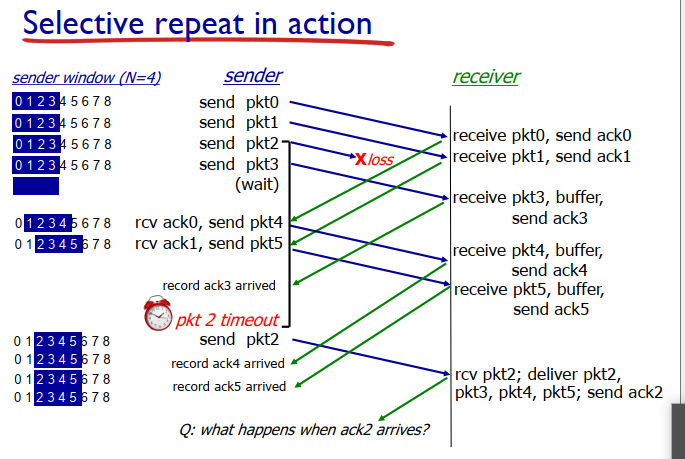


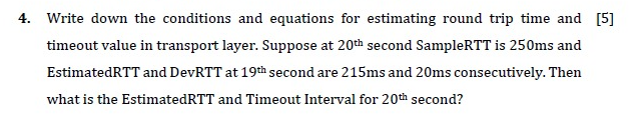


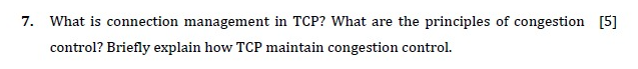


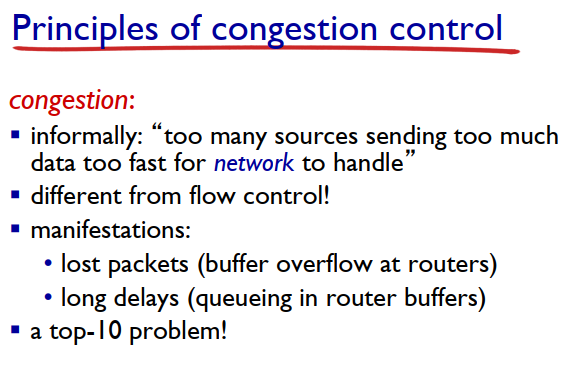


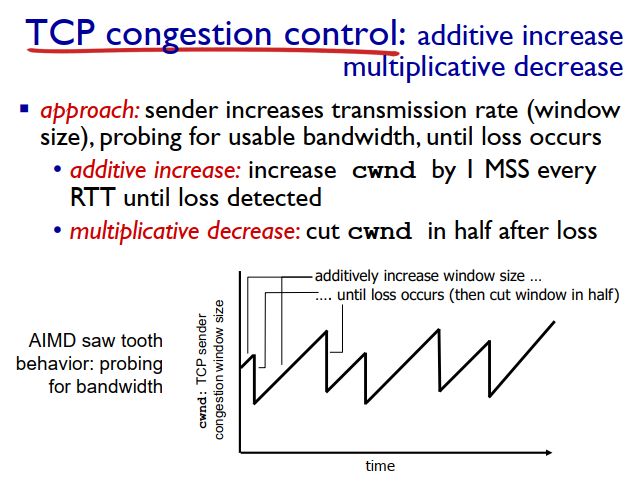


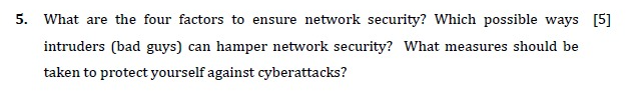


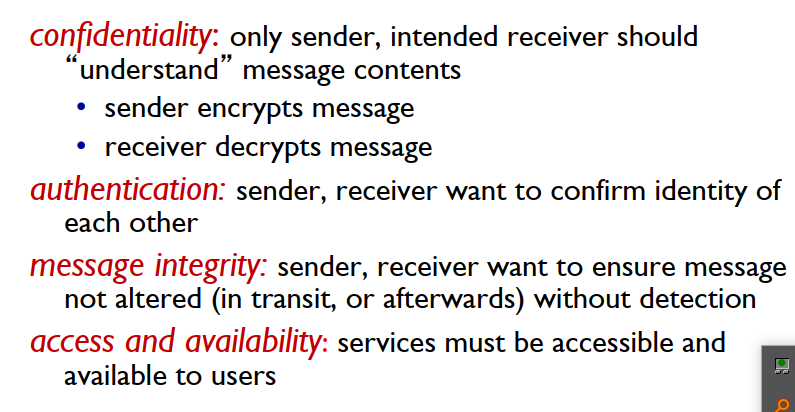


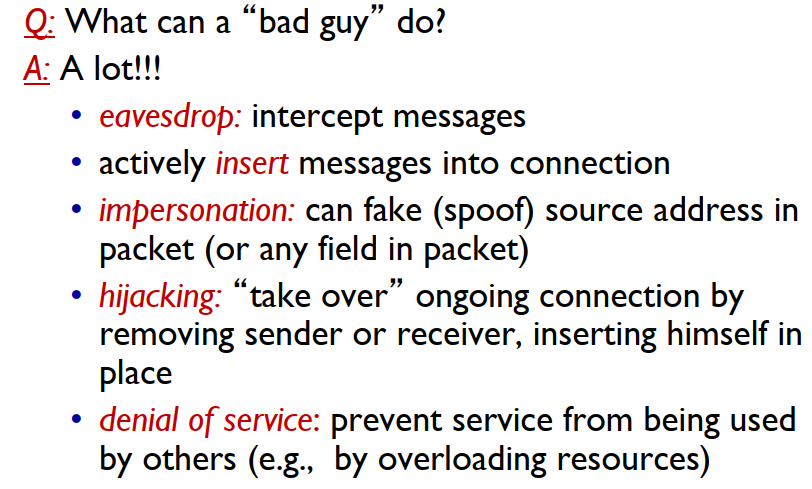


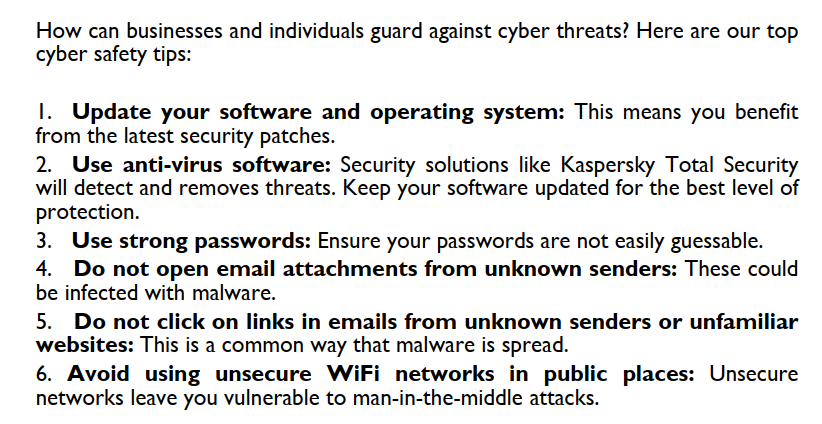


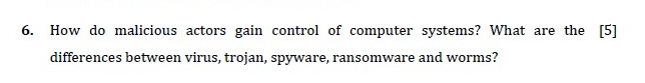


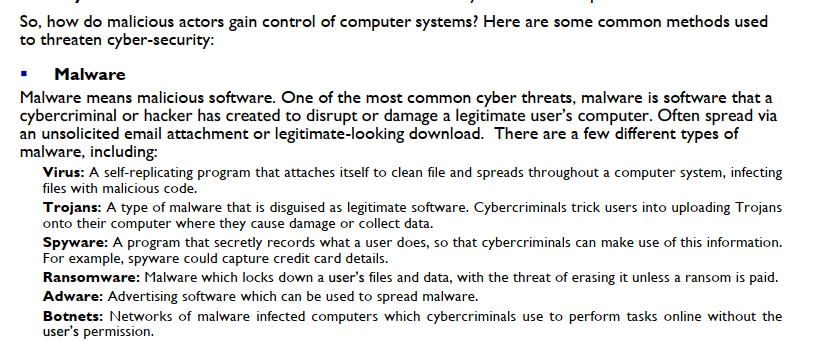


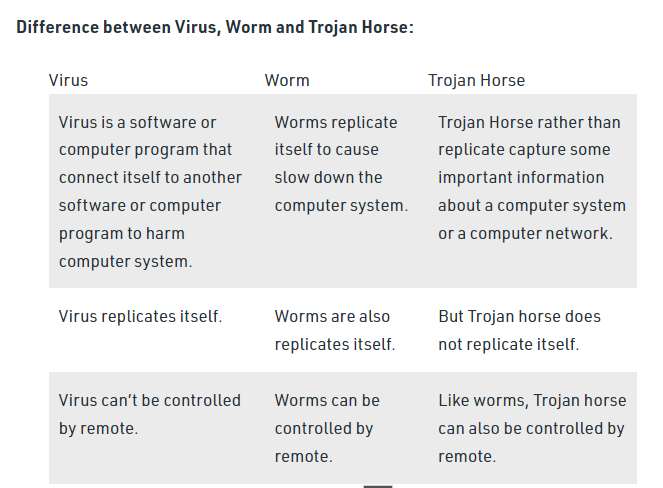


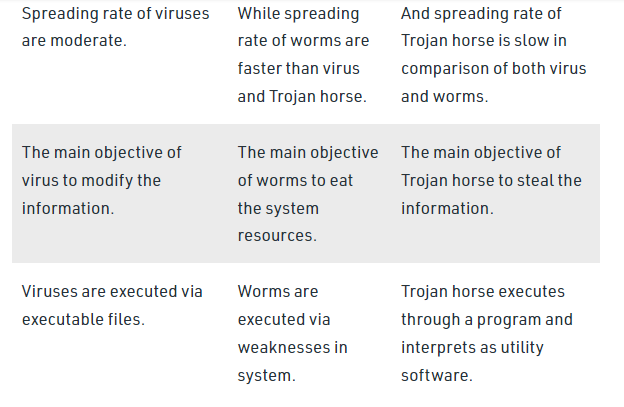




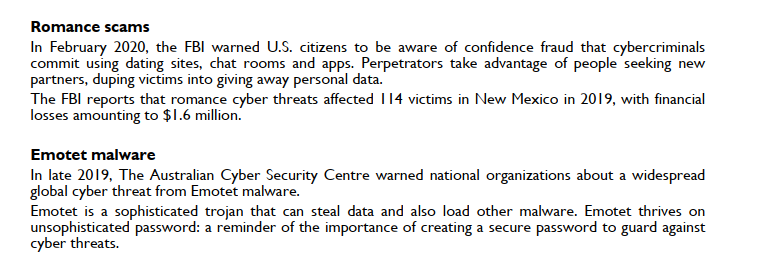












Courses



Year-End Discount: **10% OFF** 1-year and **20% OFF** 2-year subscriptions!Get Premium

[**My Learning**](https://www.educative.io/learn)[**Explore**](https://www.educative.io/explore)[**Personalized Paths**](https://www.educative.io/learning-plans)[**Projects**](https://www.educative.io/projects)[**Skill Paths**](https://www.educative.io/paths)[**Assessments**](https://www.educative.io/assessments)

**Create**

Trusted answers to developer questions

**How to compute DevRTT, estimated RTT, & time-out interval in CCN**

AKASH BAJWA

**Tired of LeetCode? 😩**

Learn the 24 patterns to solve any coding interview question without getting lost in a maze of LeetCode-style practice problems. Practice your skills in a hands-on, setup-free coding environment. 💪

Ace Coding Interviews

**Overview**

We use different methods and mediums for sending and receiving data in our daily life, like SMS. Similarly, if we talk about computers, communications occur through networks in computers. Imagine that you send a text to your friend and are waiting for their reply, and then you receive a reply. The timethatis taken to complete the process of requesting (your text) and receiving a response (friend's answer) is known as **RTT (round trip time)**.

**Measures of RTT**

There are three ways to calculate RTT:

1. Estimated RTT
2. Deviation in RTT
3. Time-out Interval

**Estimated RTT**

In computer communication networking, the RTTs of different packets can be different. For example, the first packet takes the round trip time of 1.1ms**,**the second packet takes 1.3ms, and the third packet takes 0.98ms so, each sample RTT varies. That is why **estimated RTT** is used, as it is the average of recent measurements, not just the current sample RTT.

**Formula:**

**Estimated RTT = (1- α) \* Estimated RTT + α \* Sample RTT**

Where typically, the value of α = 0.125.

For example, the sample RTT is 100ms, we have to compute the estimated RTT using α = 0.125, and we assume the value of the estimated RTT just before the sample RTT was 110ms. So, by using the formula, we get:

    Estimated RTT = (1-0.125) \* 110ms + (0.125) \* 100ms

    Estimated RTT = (0.875) \* 110ms + 12.5ms

    Estimated RTT = 96.25ms + 12.5ms =**108.75**ms

Hence, the required estimated RTT is 108.75ms.

**Deviation in RTT**

**Deviation in RTT**, also known as **Dev-RTT** is a measure that indicates how evenly the RTT is distributed during the measurement. It depends upon the previous estimated RTT and helps to find the retransmission time-out.

**Formula:**

**DevRTT = (1- β) \* DevRTT + β \* |Sample RTT - Estimated RTT|**

Where typically, the value of β = 0.125.

For example, the sample RTT is 100ms, we have to compute the DevRTT, and we assume the value of the estimated RTT is 108.75ms and the previous DevRTT was 20ms. So, by using the formula, we get:

    Dev RTT = (1-0.125) \* 20ms + (0.125) \* |100ms - 108.75ms|

    Dev RTT = (0.875) \* 20ms + (0.125) \* 8.75ms

    Dev RTT = 17.5ms + 1.09ms = 18.59ms

Hence, the required DevRTT is 18.59ms

**Time-out**

**Time-out** is longer than RTT, but as RTT varies, we need to add some margin i.e., **safety margin**,to it. The selection of a time-out value is essential because the longer the value of estimated RTT, the slower its performance. We will be facing long delays in this case. Similarly, in the case of a minimal value, the connection can be lost before the RTT completes or before the arrival of the response or acknowledgment. So for the safety, we use:

**Formula:**

**Time-out Interval = 4 \* DevRTT + Estimated RTT**

Where the DevRTT value is used for a safety margin.

So, using the above-computed values of DevRTT = 18.59ms and Estimated RTT = 108.75ms, we compute the time-out interval as:

    Time-out Interval= 4 \* 18.59ms + 108.75ms

    Time-out Interval= 74.36ms + 108.75ms = 183.11ms

Hence, the required time-out interval is 183.11ms.

We can see that the time-out interval is dependent on both DevRTT and estimated RTT. Also, DevRTT is dependent on estimated RTT. So, if we need to find DevRTT, we have to compute the estimated RTT first. To compute the time-out interval, we have to find both DevRTT and estimated RTT first.