1dv701 exam 160423 with example solutions

Problem 1 (4 + 2 + 4 = 10p)

1. What are the layers of the TCP/IP (Internet) model? What are the main responsibilities of each layer?
2. What is the difference between TCP and UDP? Give examples of when would you use each of these.
3. A TCP connection is initiated with a three-way handshake. Explain how it works and why it is necessary. Make sure your answer includes the relevant TCP header fields/flags and their values.

Answers

1. The four layers of the TCP/IP model are *Application*, *Transport*, *Internet*, and *Link*. The Application layer defines how applications communicate and exchange data. The Transport layer performs host-to-host communication. The Internet layer makes sure that datagrams can cross network boundaries. The (Data) Link layer defines how data is transferred on a local network. Each layer hides the details of the lower layers.
2. TCP and UDP are two transport protocols. UDP is a connection-less, unreliable datagram service. TCP is a connection-oriented, reliable stream service. TCP should be the default option, unless you know that you do not need some of its features and do not want to pay for the overhead. For example, if your protocol is a simple message exchange over generally reliable networks (e.g., a LAN), the overhead of TCP might be too large (consider DNS).
3. 1. The Initiator of the connection first sends a SYN to the Receiver. The SYN segment has a random sequence number, X. 2. The Receiver replies with a SYN-ACK. The sequence number is set to a random number, Y, and the ACK

number is set to $X + 1$. 3. The Initiator sends an ACK to the Receiver with the ACK number set to $Y + 1$. The handshake is used to synchronize sequence numbers, which are needed for reliable communication. The reason we use a three-way handshake is that both ends need to send both a SYN and an ACK, i.e., one end need to signal that they want to initiate and get an acknowledgment that the other end received the message.

Problem 2 (4 + 4 + 2 = 10p)

1. What are the four basic LAN topologies? What are the benefits of each topology and when would you use it?
2. Explain Token Ring and CSMA/CD, and CSMA/CA. What are these, why are they used, and where are they used?
3. Thin Ethernet (coax) uses T-connectors and terminators. Explain.

Answers

1. Bus, Mesh, Ring, Star. In a bus network, stations are attached to a single bus (cable) that is used for communication. Both ends are terminated (see Problem 2.3). A Bus network is inexpensive and easy to install. A Ring network is basically a Bus with the end points connected to form a ring. They share the benefits for Bus networks, but with the added benefit that there is always a left and right neighbor, so protocols that require that are easier to implement. A Mesh network is generally a fully connected network, i.e., every station is connected to every other station. Mesh networks are highly reliable, since there are multiple connection between each pair of stations. In a Star network, each station is connected to a central hub. Each station has its own connection, so if a connection fails, only that station is affected. Star networks are also rather inexpensive and easy to set up. Use Ring or Mesh when you need their benefits. The main difference between Bus and Star is the required cable and installation, which also impacts when you would use one over the other.
2. Imagine a Bus network. Any station can access it at any point, and if two stations send at the same time, they will interfere with each other. So, the network is a shared resource, and a station that wants to send needs exclusive access. The mentioned protocols are used for media access control, i.e., to determine which station is allowed to send. Token Ring is a controlled access protocol, where a station is only allowed to send if it holds a token. CSMA/CD and CSMA/CA are random access protocols, where stations either try to avoid or detect collisions. The access protocols are Data Link layer protocols.

3. Thin Ethernet is generally used for Bus networks, so the T-connector is used to connect to the bus at each Station, and the terminators are used at each end point to "end" the network.

Problem 3 (2 + 2 + 3 + 3 = 10p)

1. When you measure the performance in a network you can check different things. What are the two main factors you measure?
2. What does goodput mean?
3. When you implement QoS you can either choose a fine grained or a course grained approach. Explain what that is and how it affects the possibility to get QoS.
4. Explain Figure 1.

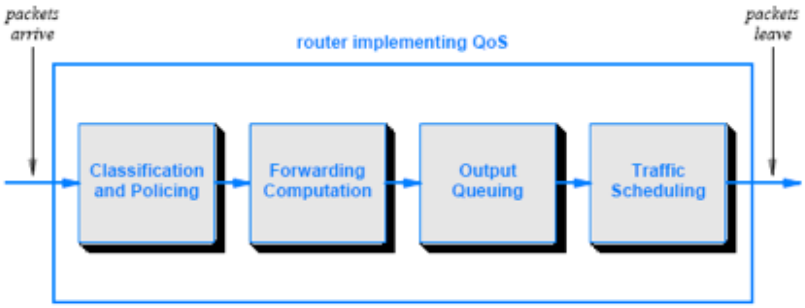

Figure 1: Router QoS implementation.

1. The two main characteristics you measure are Delay (latency) that you measure in milliseconds and throughput that is measured in bits per second.
2. Goodput is the effective data rate achieved by an application in a client. It includes not only the network aspect but also how much time it will take for the server to create the answers it is sending back to the client.
3. For fine grained Qos you you need to handle each transport session separately, making sure the network can meet the expected QoS for session. For course grained QoS you refer each session to one of a few types of services and give different types different priority. This mean you can never give any guarantees for a single session in course grained QoS, but the implementation is much easier.

4. The figure show how a router implementing QoS typically work. Each incoming data packet passes each of the four steps before it is finally sent out on one network interface:
    1. the router classifies the packet by assigning to it a flow identifier and discards packets violating the service agreement (e.g. to many packets per second)
    2. looking up the correct output interface in the routing table
    3. put packets in one of several output queues connected to the output interface depending on the flow identifier
    4. selecting from which queue the next packet to be sent will be taken from.

Problem 4 (2 + 2 + 4 + 2 = 10p)

1. What are A, B and C-class addresses? How do these compare to CIDR?
2. How does a computer get (is assigned) an IP address? Explain at least two different ways.
3. Assume you want to access *google.com*. How is the name translated to an IP address? Explain all the steps, and include various possibilities in each step.
4. *google.com* is an (DNS) A record. What does that mean? Give examples of two other types of records and explain what they mean.

Answers

1. An IP address is divided into two parts, the network and the host parts. The three classes determine how large (in bits) each part is. In a class A address, the network address is the first eight bits, and the host address is the remaining 24. For B, it is 16 for each, and in C, it is 24 bits for the network, and 8 bits for hosts. The number of bits for each determines how many such networks there can be, and how many hosts a network can have. The class system used to be the basis for routing, but it was replaced by CIDR, Classless Inter-Domain Routing. The difference is that in CIDR, the network can be an arbitrary number of bits. The size is controlled via the netmask (often written using the form /X). So, given the address 192.168.2.0/24, the network address would be 24 bits and the host 8 bits (i.e., same as a class C).
2. It can be assigned manually (by editing a file or setting in a GUI), via DHCP, or via RARP.
3. The name is either translated via a static mapping on the machine (i.e., the hosts file), or DNS. If you use DNS, your machine will first check its cache, and if it

is not cached it will contact the local DNS server (configured by, e.g., DHCP). The local DNS server will also check its cache, and if it is not cached their either, the local DNS will ”walk the tree”. It starts with the root server and asks for the server for ”.com”. It then contacts the ”.com” server and asks for the server for ”google”, and then contacts the DNS server at Google and asks for the host you are looking for, and its IP is returned, cached by the local server, and then returned to your computer. Queries can be recursive, i.e., just like the query from your machine to the local DNS, or iterative, i.e., just like how you local DNS server finds the address of the host at Google.

4. An A record maps a hostname to one or more IPv4 addresses. An AAAA record maps to IPv6 addresses. Other common types are MX for mail (SMTP) servers and CNAME (canonical name) for aliases

Problem 5 (4 + 3 + 3 = 10p)

1. Routing protocols on Internet typically use two different metrics. What are these and what is the purpose of using a metric?
2. Explain the concept *Autonomous System*.
3. Internet multicast routing is considered to much harder then unicast routing. Describe why this is the case. Also give an example of when multicast routing could be beneficial.

Answers

1. There are many possible metrics but the two that are generally used are hop count or administrative cost. Metrics are used to help routing protocols to find the ”best” path through the network when they update routing tables.
2. A flexible, soft definition. Set of networks and routers under one administrative authority. Intuition: a single corporation. Inside an AS you use one specific routing protocol, but you are free to choose whatever protocol you prefer.
3. Difficult because Internet multicast allows;
   - Arbitrary computer to join multicast group at any time,
   - Arbitrary member to leave multicast group at any time,
   - Arbitrary computer to send message to a group (even if not a member)

   It is beneficial when many clients would like to receive the same information at the same time, a typical situation is live video streaming.

Problem 6 (4 + 4 + 2 = 10p)

1. What is congestion and why/when is it a problem? If a majority of the traffic on the Internet is HTTP, is congestion still a big problem? Why/why not?
2. Figure 2 shows the advertised congestion window over time. Explain what is happening, using the named points in time (A–G).
3. What is ICMP and what is it used for?

Answers

1. Congestion happens when a network node is carrying more data than it can handle. This can result in packet delays or drops. When congest gets bad, it can result in a congestion collapse, where the congestion prevents or limits any useful communication. So, congestion can have a big impact on the quality of service. With a major of HTTP traffic, it simply depends. The old version (1.0) of HTTP used a new connection for each operation (so ten connections if you fetch an HTML page and nine images), so there would be many connections which would increase load on the network. In HTTP/1.1, a persistent connection can be used for all connections, so there would be less load.
2. (A) Slow start with an exponential increase, (B) Loss detected via fast retransmission (Lecture 4, slide 53 (2017)) so CWND is divided by 2, (C) additive slow start, (D) another loss detected via fast retransmission, (E) another additive slow start, (F) loss detected via timeout at sender, and (G) exponential slow start. Note the difference between the two difference ways of detecting loss and their effect on the CWND.
3. ICMP is the Internet Control Message Protocol. Is is used to send errors and operational messages and for debugging. Example messages are for example Echo request (used by ping) or Destination unreachable.

Problem 7 (2 + 4 + 4 = 10p)

1. What is a *Socket* (in the context of computer networking)? Explain what it is used for and the different kinds that exist.
2. You are asked to write a program that uploads photos using UDP. Each UDP packet can at most be 100 bytes, but the photos can be of arbitrary size. Provide (pseudo) code/algorithm for both client and server that uses *Stop-and-Wait ARQ*.
3. Improve your solution to use a sliding window. Use a reasonable window size (motivate your choice).

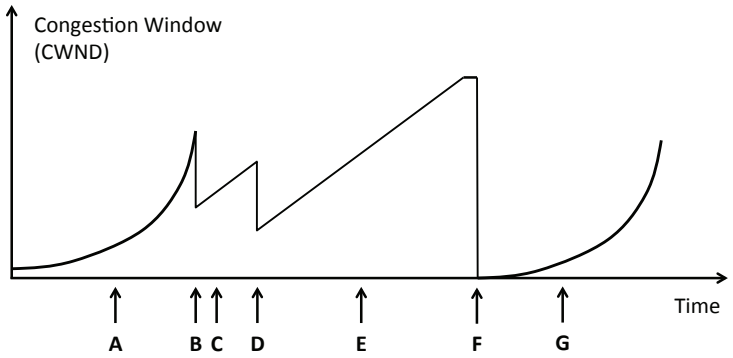

Figure 2: The advertised congestion window over time.

Answers

1. A socket is an internal endpoint for sending or receiving data. It is a representation of a system resource, managed by the networking stack. In the scope of this course, we have discussed datagram and stream sockets.
2. This is quite similar to what you did for your tftp server. This is a text explanation, but you should easily be able to translate it to pseudo code. On the client side, if there is something left to send, send it and start a timer. Then listen for an ACK. If you get the ACK, continue and send the next part (or end if there is nothing more to send). If you get a timeout, resend the same data, start the timer, and listen for an ACK... On the server side, continue to receive data and then send an ACK. Note that this solution fails if the the connection is slow, and the server receives and ACKs something, but the client times out before the ACK is received. You can easily fix this by adding sequence numbers to whatever is being sent.
3. First, change the client to send w parts of the photo at once and include a sequence number with each part sent. Start a time out for each part sent, and wait for an ACK or a timeout. If you get an ACK for the first unACKed part, move the window one step forward, send the next part and start a timeout. If you get a timeout, resend the part that timed out and all parts you send after it. On the server, receive data and check the sequence number. If it is the part you expect, send an ACK with the sequence number. If it is not a part you expect, discard it. In this version of sliding window, we discard the remainder of the window each time we get a timeout, so it can be quite inefficient if we have many errors. So, in this case, it makes sense to keep the window reasonably small to reduce overhead.

Problem 8 (2 + 2 + 4 + 2 = 10 p)

1. What is the difference between a switch and a hub? Explain the internals.
2. How does a computer attached to a shared Ethernet LAN decide whether to accept a packet?
3. Calculate the spanning tree for the network in Figure 3 on the following page. Show the final tree and explain the algorithm used to calculate it.
4. If Host E and F communicate, what hosts can see their message? What if Host E and A communicate? Assume that all hosts have their Ethernet adapters in *promiscuous mode*.

Answers

1. A hub is basically a repeater while the switch is a set of bridges. This means that a hub takes anything that comes in on any of its ports and repeats it on all outgoing ports. A switch keeps track of which port the station that should receive the messages is on, and only sends it to that port. So, a switch is smarter and more efficient (since packages generally does not get send on ports where nobody cares about them).
2. The network interface card/adapter has an address (a MAC/Ethernet address) and it only accepts a package if the destination address in the packet matches the address of the adapter (unless the adapter is in promiscuous mode).
3. (You can find the algorithm in Lecture 8, slide 17 (2017)). Start by electing a root bridge, in this case the bridge with the lowest ID will win. So, each bridge announces who it is, and who it think will be root, and its distance to that node (e.g., 1, 1, 0 for Bridge 1). Once a node receives a message with a lower numbered root, it realizes that it is not the root, so it sends out its id, the new root id, and its distance to the root. So, if Bridge 2 receives the root message sent by Bridge 1, it will send 2, 1, 1 to say that Bridge 2 thinks Bridge 1 is the root and its distance to the root is 1. In the example, the spanning tree would be Bridge 1 as the root with Bridge 2 and 4 as its immediate children. Bridge 3 would be a child of either Bridge 2 or 4, it does not matter which one (depends on which message it gets first, no point in changing paths to the root if you cannot improve it).
4. If we assume the spanning tree from above, and that Bridge 3 is a child of Bridge 2, then a message from E would pass through LAN E, B, A, D, and F. Any host connected to those LANs would be able to see the contents. If E to A, just remove everything after A.

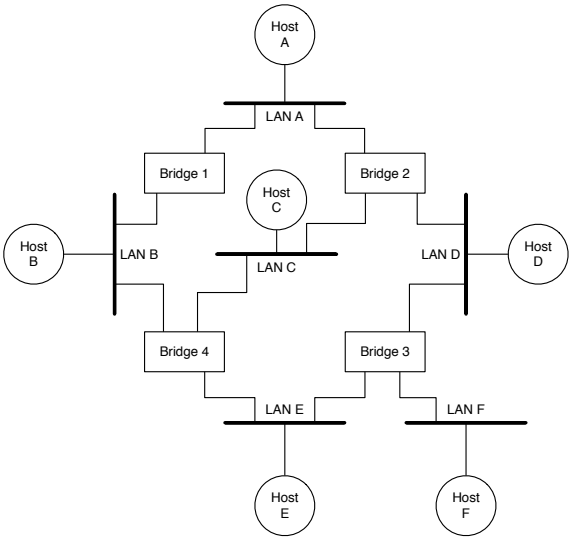

Figure 3: A network with bridges.