# EECS563 Exam 2 Study Guide

Stephen Longofono

Fall 2017

*The information herein was assembled from lecture notes and materials provided by Dr. James Sterbenz, along with book examples from Computer Networking: A Top-Down Approach by Kurose and Ross. This document is my interpretation of the materials presented in and outside the classroom for EECS 563 at the University of Kansas. Be sure to review the references for more authoritative information.* [1][2][3][4]

## 1 "This will be on the exam..."

This section is comprised of things which were explicitly mentioned as exam fodder.

- Explain the difference between blocking and contention.

  Blocking refers to one input-output pair interfering with w different input-output pair within switch fabric. For example, a flow from interface A to B interferes with a flow from C to D.

  Contention refers to two packets destined for the same output interface. One must wait, regardless of switch design. Contention is inevitable and is why all routers include input buffers.

- Is DNS an application or network layer protocol?

  DNS uses UDP, and thus is by Kurose's definition an application layer protocol. However, DNS is so crucial to how the Internet works at large, and is a part of the Internet hourglass, so it can be considered a network layer protocol.

- How does NAT help the IP address exhaustion problem?

  NAT allows any an arbitrary number of internal IP addresses to be addressable by a single exterior address.

- How do we "solve" the problem of running out of IP addresses?

  CIDR lets service providers use an arbitrary number of subnet bits, allowing more efficient delegation of IP addresses.

- What functions of the network layer correspond to the Internet hourglass?

  Addressing, forwarding, and routing.

- What aspects of the network layer are in the data plane?

  Forwarding

- What aspects of the network layer are in the control plane?

  Signalling, routing, and traffic management.

- Explain the difference between routing and forwarding.

  Routing describes the process of assigning a path through the network as a collection of hops.

  Forwarding describes the process within a router of mapping one interface to another, selecting the next hop to be made.

- What are the three goals of routing algorithms?

  Stability, Simplicity, and Optimal

- Which routing algorithm is better?

  Link state is better - it can be scaled with a hierarchy, it is better at isolating problems/misinformation, and it converges fast (after initial computation of routes).

  See the Table 2.

- Why is BGP part of the Internet hourglass?

  BGP must be common to all ASs, it is the glue that holds them all together as the Internet.

- Don't memorize what an ethernet frame looks like, but know all of its part and what order they are in.

  Ethernet frames consist of, in order:

  - Preamble
  - Destination Address
  - Source Address
  - Length
  - Type
  - LLC and SNAP subheader (for physical layer)
  - Payload
  - FCS error correction sequence

- Why implement switching at the link layer when it already exists at the network layer?

  Layer 2 switches respond and recover much faster than layer 3 switches. In this case, it gives an overall performance benefit to include addressing and switching at the link layer.

- What is the difference between the link layer and the transport layer?

  The link layer is concerned with moving network layer PDUs over a single hop in the network. The transport layer is concerned with moving an application layer PDU from one end system of a network to another.

- Why do we need hop by hop error control at the link layer?

  The link layer can detect and respond to error much faster than the end to end system at the transport layer can. Duplicating the error checking at the link layer provides an overall performance improvement. Also, link layer checks are more robust (can correct errors to eliminate re-transmit) and better at detecting error in the case of CRC.

- Why are layer 2 switches not a replacement for layer 3 switches?

  Layer 2 switches cannot aggregate IP addresses, so they would be stuck with enormous tables that would be impossible to lookup at speed.

# 2 Major Topics

## Network layer

- 5 Key Network layer functions:

  - Addressing - This is The hourglass of the Internet. The IP addressing protocol is a crucial part of making the Internet broadly useful and compatible
  - Forwarding - The network layer is tasked with determining how upper-layer PDUs will be passed on from any given node; forwarding decides the next hop.
  - Routing - The network layer is tasked with determining how upper-layer PDUs will traverse the network to their destination. The overall path is the route taken through the network.
  - Signalling - To facilitate communication among routers about status, cost, and routing algorithms, the network layer needs a signalling plan (ICMP)
  - Traffic management - If a node is experiencing high load or if a section of a network needs to control what traffic it will pass, the network layer is responsible.

- Network layer service models and best effort service vs. applications The network layer in the Internet is best-effort, in that it tries to get most of the packets to their destination, most of the time, eventually. I.e. no guarantees, but it will make an effor to avoid problems. This is in contrast with a best effort application, which is concerned with providing its best effort at a timely delivery. A best-effort application which failed to deliver its payload is bad; the best-effort service of the Internet failing a few packets is not a huge deal.

  There are other service models used by earlier networks and specialty networks. Things like quality of service, reliability, and order-of arrival are possible with a different network design. However, the Internet offers none of these things.

- Signalling and transfer paradigms: circuits, virtual connections, datagrams General signalling paradigms in historical networks can be grouped into the three categories of circuits, virtual connections, and datagrams.

  - Circuits, used by the early PSTN, used a physical connection that was set up in advance. This generally involved a relatively large set-up time, and was exclusive. This was slightly improved by TDM, FDM, and bus-style circuit switching, but it was inherently inefficient in that if any given line was not actively transmitting, its bandwidth was wasted.
  - Virtual circuits fakes an actual circuit by establishing connections between each hop, and thereafter transmitting as if it were a dedicated circuit. The setup process is still long, but simple labels allow packets to be switched at hardware speeds once it is established. This is what the modern PSTN is using. The connection-oriented service keeps state about connections, and enjoys high throughput. It can exploit some efficiency of multiplexing, but the costly setup is wasteful for small messages and signalling.
  - Connectionless/Datagram service needs no real state, but in practice the state is captured in the forwarding table. This is the most flexible, and can be the most efficient in terms of data. However, a major problem with this paradigm is that the sheer amount of addresses required made it difficult to forward at line speed. For a long time, the lookup at each hop was the bottleneck

Table 1: Comparison: Connectionless vs. Connection-Oriented Paradigm

| **Characteristic** | **Connectionless** | **Connection-Oriented** |
|---|---|---|
| Setup Latency | None | High |
| Address Lookup | Slow and hard | Fast, label switching |
| Forwarding Information | Long address | Short ID |
| Resilience to Failure | Only a few packets | Dead stop on failure |
| QoS | Hard | Easy with reservation |

- Generic switch (and router) architecture

  In general, switches can be divided into the three areas of responsibility depicted below in Figure 1. The important parts common to both layer 2 and 3 switches are the manipulation and transfer control. Every inbound PDU will need to be unwrapped, processed, put through switch fabric, and processed for outbound traffic. The control plan dictates how the switch fabric will operate, and may also handle the task of filtering or special processing. High level controls for more sophisticate switches allow them to respond to traffic and network management signals, and partipate in route discovery.

- Store-and-forward router architecture and bottlenecks

  In the 80s, switches had a hard time dealing with the addressing of connectionless architecture. The lookup time for even a small network was huge, and the amount of state required to hold all addresses in the router table was a problem. This was essentially a bottleneck, the input queue of a switch filled up quickly as it struggled to walk a routing table and forward packets.

- Fast packet switch motivation and architecture

  The fast packet switch of the virtual connection architecture set special IDs in each of the switches along the route as a part of the setup process. This overhead paid off once the connection was established: any given packet had only to use its magic label swap and all delays were removed!. See slides for more hand-waviness.

- Switch fabric blocking, contention, and architecture

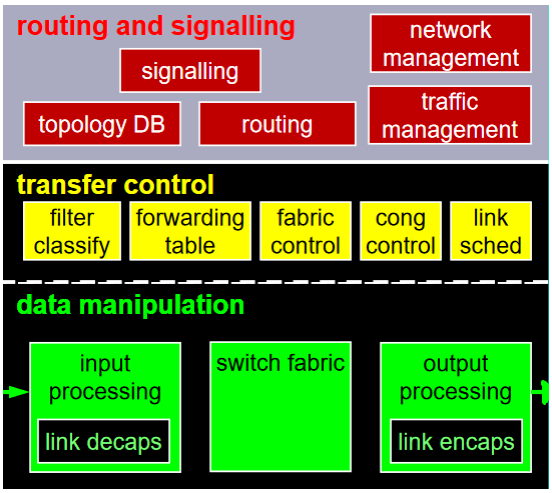


Figure 1: Generalized Switch

Switch blocking occurs where due to the design of the switching fabric, traffic on one flow can interfere with traffic to another, that is, a packet forwarded from A to B interferes with a packet forwarded from C to D. One of the desirable characteristics of a switch design is non-blocking, or low probability of blocking.

Switch contention is an inevitable phenomena where two inbound packets are forwarded to the same outbound interface. All switches must have buffers to handle contention.

- PSTN addressing structure and format

  PSTN addressing is done via telephone numbers. The last four numbers are the local subscriber line number, confined by design to 10000 unique lines in the four digits. The next 4-6 numbers are the area/region code and the country code. This has a long and exceptionally boring history, wherein a lack of foresight led to bad decisions, which led to many band-aid solutions, and now they have NANP.

- DNS functions, name structure, and operation

  DNS is a service which translates IP addresses to and from familiar host names. Technically, this is an application layer service, which operates over UDP. However, it is so fundamentally important to making the Internet easy to use that Sterbenz decided it belongs in the Network layer. One day, Kurose and Sterbenz will have a fistfight to determine once and for all where it belongs, and thereafter stop obscuring the issue behind a petty and ultimately inconsequential classification. ANYWAYS, DNS servers use a complicated hierarchical overlay network to resolve DNS requests containing hostnames to their respective IP addresses.

  The hierarchy of DNS servers is setup as follows: 13 root servers around the world maintain lists of Top Level Domain servers (.com, .edu, .mil, .org, and all country codes .uk, .ca, etc). Each of these TLD servers maintains a list of authoritative servers for specific Second Level Domains (amazon.com, ietf.org, etc.). Each of these SLDs will have a list of valid DNS records which it can return on request. Local DNS caches greatly improve performance, as a single DNS lookup could involve 3 RTT if a request is made to root, TLD, and SLD. This happens even with recursive requesting, it still needs to traverse the hierarchy.

  DNS requests typically include one of the following, depending on the type of request desired:

  - Type A is a resolution of a name to an address or vice versa
  - Type NS is a request for the name server which handles the given name or IP
  - Type MX is a request for the mail exchange server
  - Type CNAME gets the canonical name of the given name

  The responses can be stacked, and fields in the header indicate how many of each response there are (Type A, type NS, and Others respectively).

  There is also a TTL used to determine life in a cache, and header length flags to make parsing easier.

- IP important packet fields: version, IHL, length, TTL, protocol, header checksum, addresses

  - Version - determines IPV4 vs IPV6
  - IHL - Specifies the length of the header only, minimum 20 bytes
  - length - Specifies the length of the entire datagram (minimum 20 bytes with no payload and no options)
  - TTL - This is decremented as the datagram is processed at each hop, and when it reaches 0, the datagram is dropped. This kills a datagram which may be stuck in a cycle.
  - protocol - This indicates the layer 4 protocol for appropriate demuxing when the datagram reaches its destination.
  - header checksum - This simple/weak checksum is used to verify header contents only. Uses the 1's complement of the sum of the checked data.
  - addresses - IP addresses for source and destination are supplied here.

- IP address formats: class-based, subnetting, CIDR

  IP addresses are decimal-encoded triplets of binary numbers. IPV4 uses 32-bit addressing, and IPV6 uses 128 bit addressing.

  When first handing out IP ranges to businesses and the like, a class-based system was used with 8, 16, and 24 bits pre-assigned. A subnet referred to any addresses within, for example, the class subnet 192.168.1.0/24 uses the first 24 bits (192.168.001) for the subnet address, and within that subnet, there are 255 possible addresses determined by the last three digits. This created waste: three sizes for an infinite number of use cases was quickly problematic, and addresses began to run out.

  CIDR addressing solved the problem by letting the subnet use any number of bits. This allowed businesses to choose exactly the right amount of addresses, and with some help from the governing body, makes more efficient use of the address space. However, it is still not enough. The Internet hungers.

- NAT motivation and operation

  NAT was a stopgap solution to dwindling numbers of IP addresses and billing issues ('mo money) with multiple IP addresses. NAT operates by mapping an IP address and port on a local subnet to an outward facing port using the NAT device IP. This allows a single point of entry into the subnet, and the NAT can selectively forward traffic. This allows internal IP addresses to be anything, and adds a layer of security in that no internal node is directly addressable. However, it violates the principle of transparency of addressing on the Internet.

- ICMP purpose and examples: ping and traceroute

- IPv6 motivation and 128b addresses

  IPV6 was drafted to address some things learned in practice with IPV4, and to make the addresses long enough so that we will likely never have to deal with running out again. The key improvements of IPV6 are

- Fast datagram router motivation and architecture

- IP lookup: longest prefix match

  Forwarding tables use the longest prefix match since it is easily realized in programmable hardware. The longest prefix allows certain ranges to be directed to certain interfaces, which aligns well with the way ISPs and IXPs distribute and handle IP addresses. It also has the benefit of allowing IPs to be moved; if a major company wants to keep its IP addresses, but use a different ISP, the new ISP can include more specific IP ranges in its report, ensuring that the traffic makes its way to the new network.

## Network Routing

- Difference between routing and forwarding

  Forwarding is how the network layer determines the next hop in its path, and entails mapping an ip address range to an outpu interface of a router. Routing is how the network layer decides what path a datagram will take through the network, the aggregate path of all the hops it should take.

- Graph network model and link costs

  The graph network model shows hosts and routers as nodes, and links as edges with an associated cost. These costs are used by routing algorithms to determine the best path through a network, and are assigned by network administrators according to their specific policy.

- Routing algorithm alternatives

  In general, there are a few different classifications we can apply to routing algorithms: static or dynamic, load-sensitive, centralized or decentralized.

  Static algorithms change very little if at all, and always follow the assigned paths. This typically means that costs or routing decisions were all manually set by administrators, and will not change without human intervention. Dynamic algorithms adjust periodically or in response to an event (say, a node going down or a cost change).

  State is represented differently by centralized or decentralized algorithms. Centralized algorithms use the state of the entire network at each node to determine the optimal path. Decentralized algorithms use small local state information to make decisions on a per-node basis.

  Load-sensitive algorithms respond to network load by adjusting costs along certain edges. This seems like a good idea in theory, but in practice it leads to oscillation and unstable behavior.

- Link state routing concepts and operation

  Link-State is a centralized, load-insensitive, and dynamic algorithm. When it is time to compute new paths, each node sends its connections and their costs to every other node in the network, this is used to construct a table on which to run Dijkstra's algorithm.

  Oscillations are possible in link-state algorithms, especially if traffic is encapsulated in the cost of any given link. Link costs are often set to encourage flow through high bandwidth, but changes or asymmetric cost can result in cycles or oscillation.

- Distance vector routing concepts and operation

  Distance-vector routing uses only the immediate neighbors and their state to determine the best route recursively. A distance vector is shared, which contains the cost from the current node to its neighbors. When distance vectors are shared among neighbors, if the cost through a neighbor to another node is cheaper than an existing known interface (or new), then the cheapest path to the node is updated to run via the new route. This is based on the Bellmann-Ford equation, which states that along the shortest path, the shortest distance to and endpoint is the shortest path to some midpoint plus the cost from that midpoint to the endpoint.

  Whenever a cost changes, a new DV is computed and sent out. Whenever a new DV is received, a new local DV is computed, and sent out IF IT HAS CHANGED.

  DV routing has the characteristic of good news traveling fast - a reduced link cost will cause multiple new DVs to be sent out and will quickly propagate through the network. However, if a link cost increases significantly, it can take a very long time to propagate. The reason for this is that any given link will change its DV and send it out, and this will likely change the neighbors' DV, which will again change the current node DV, and so forth. This is the main idea of the "count to infinity" problem, a large change in cost will count toward convergence in very small steps, and the larger the change in cost, the mre steps it will take to converge.

Table 2: Comparison, Link-State vs. Distance-Vector

| | **Link State** | **Distance Vector** |
|---|---|---|
| Message Complexity | High, flooding | Low, neighbors |
| State Maintenance | High, entire subnet | Low, neighbors |
| Convergence Time | Fast, if ignore compute time | Slow for cost increase |
| Robustness to Errors | Errors confined to a link | Errors propagate |
| Scalability | Only with hierarchy | Poor since convergence is slow |

- PSTN routing types: HIER, DNHR, RTNR

  PSTN routing developed with the growing size of the network, and thus has several different routing methods.

  HIER refers to the fixed hierarchy - when connecting from a low-level switch, if the desired endpoint is not found, there is a set order by which the hierarchy is traversed ("Hunting order"). This hunting order looks

laterally, steps up and looks low, looks laterally, steps up, looks to the new low, and so on, until it finds its destination or terminates. This had the advantages of directly mapping to the physical topology, and being simple to implement in hardware for switches. However, it lacks flexibility and it difficult to deal with load.

DNHR is dynamic non-hierarchical routing, which operates as a mesh of switches. In addition to a primary connection for a given pair of nodes, there is also a two-hop equivalent path stored. This permitted changing preferred routes (which is primary versus alternative), and dealing with load/congestion.

RTNR is a real-time non-hierarchical routing, which builds on DNHR by using a distributed algorithm to determine per-call paths based on current network load.

- Internet routing structure: ASs / routing domains

  The Internet evolved as multiple NETs in tandem, which eventually joined together at gateway nodes. Internet routing to bring them together thus needed to deal with the complexity of bringing disparate systems together. Today, the disparate systems are different, and are classified as autonomous systems or routing domains. Larger ISPs may include several interior ASs, and smaller ISPs or IXPs may only be a single AS. There are many IGPs (interior gateway protocols) that any given AS can use to communicate with other ASs within an ISP. This is in contrast to the EGP (external gateway protocol) that links togeter ISPs; there can be only one EGP, or the whole system will not work. Every agreed on BGP, and that is what we are stuck with.

  Major IGPs include OSPF and ISIS, both of which are LS routing algorithms. There are many others, only a few of which use distance-vector.

  Highlander is BGP-4. There can be only one. BGP is a path-vector protocol, neither LS nor DV. Why this is important is beyond me; if you know how it works, the classification is arbitrary.

- Intradomain routing protocol OSPF concepts and operation OSPF stands for Open Shortest Path First. OSPF is an LS routing algorithm, and sends its meta-messages over IP using protocol ID 89. A "hello" message is used to establish neighbors, the results of which are stored in an adjacency database. Flooding is done via reliable (ACKed) link-state announcements, whenever a link cost changes or every 30 mins. OSPF support secured messaging, multicast, and hierarchy. In general, costs are manually configured by wizards.

  OSPF areas facilitate hierarchy by allowing LS to bee flooded only within an area, or only on a backbone connected to border routers. These border routers connect to other ASs or BGP, and thus form a sort of intermediate layer. Generally, this is done to restrict traffic or deal with non-engineering policy.

- Interdomain routing protocol BGP concepts and operation BGP is very broken but is so entrenched in the Internet hourglass that it cannot be fixed except through existing equipment and channels.

  BGP involves advertising routes to other BGP endpoints (ASs), each of which consists of a prefix and attributes. A prefix refers an endpoint host in the AS in question, and the important attributes are the NEXTHOP (the IP sybnet address of the border router of the next AS along the path) and the ASPATH (the series of ASs through which the advertisement has passed). Policy dictates how these advertisements are made, and in general, they are such that one major ISP will not needlessly forward another's traffic. In order for any subnet to be visible, it must be advertised by its AS.

  In general, policy dictates how advertisements propagate through the Internet. If an AS decides to advertise, i passes its endpoint prefix followed by itself as the NEXTHOP to another AS from a border router, in what is loosely called an external (eBGP) message. The border router on the next AS will send out internal messages (iBGP) to all its network's nodes using TCP, and so on until another border router is reached. This border router, if its policy says to advertise, will adjust the NEXTHOP to its own IP, append its AS label to the ASPATH, and proceed with an eBGP message to neighboring ASs.

  Hot potato routing is used to select among multiple routes in BGP. The idea is that the forwarding tables will be adjusted such that the least cost path to the nearest border router is used. This may incure a longer overall route, but the "hot potato" part is just concerned with getting the packet out of the current AS as fast as possible.

## Link Layer and LANs

- Link layer functions and services

  The link layer is the HBH analogue of the transport layer - it is responsible form moving frames over a single link. The link layer performs error checking since it offers a substantial overall performance gain, and includes support for flow control, multiplexing, and switching.

Key services are framing (encapsulation), link media access, reliable delivery, and error detection/correction.

- Link framing and delineation: synchronous, problems with counting, stuffing, preamble pattern, physical code

  At this level, the incoming data is bits. An important problem is how to determine where one layer 2 frame ends and the next begins. Several methods are in use, the most common being a preamble pattern.

  Stuffing involves delineating the start, end of header, and end of frame with a special byte or bit sequence. The name comes from the fact that in order to prevent false positives, bits or bytes are stuffed into the data to prevent erroneous flags, and then removed on de-encapsulation.

  Synchronization is a difficult problem to to dispersion and other long-distance time effects. It is very difficult to run a link with synchronized frame delivery.

  Counting relies on fixed-size frames, and would work fine, if we could be sure of the start of a frame, which we cannot.

  Physical codes rely on level changes to get the attention of the receiver, and then transmit immediately.

  Preambles use a long, repetitive sequence that is very improbable in data (say, alternating for 24 bits) that determines the beginning of a message. This is often combined with bit stuffing to reduce errors.

- Link types and topologies: point-to-point mesh vs. shared medium

  Point to point links have one dedicated transmitter per medium. They may be multiplexed (TDM), but there is still only a single transmitter connected at any given time.

  Shared medium has many transmitters physically connected to the same medium. This means there is contention for the line, and there needs to be a Medium Access Control (MAC) protocol to handle use.

  Meshes are formed by connecting switches point to point. This space-division allows hosts to attach to a local switch and still reach other hosts far away. It is relatively easy to scale by adding more switches and stacking interconnections.

  Buses and rings connect to a single shared medium - this is simple, but difficult to scale.

- Bus and ring shared medium topology and comparison

  Bus topologies were some of the first, but they present a number of problems. Physical constraints are present - due to delay, there is a fixed maximum length for a segment. Also, every attached host shares the same bandwidth, so large networks are not feasible. Early ethernet used this.

  Ring topologies also suffer from the bandwidth constraint, and experience issues with spacing - hosts that are too close together experience timing issues. There are no collisions, since traffic only flows in one direction. Contention remains an issue. Token ring is a common example - you have the token, you get to use the ring.

  Both are vulnerable to a host or a line being severed. Ring topologies developed dual-rings to get around this, but it still is inherently fragile.

- IEEE 802 LAN protocol stack: 802.2, 802.3, 802.5
  - 802.2 - Logical Layer Control - media independent layer. Since retired.
  - 802.3 Ethernet
  - 802.5 Token Ring

- Ethernet evolution by orders of magnitude in rate

  The ethernet bandwidth specifications have grown by orders of magnitude 10/100/1000 Mbps, 1/10/40/100 Gbps, up to terabit, but terabit was deemed too far ahead of its time and is shelved for now.

- Transport networks: PDH, SONET, and OTN, TDM, WDM

  SONET (synchronous optical network) uses rings that can detect severs, and wrap the ring around at switches - this is a substantial improvement in resilience and is in use in many fiber networks. SONET rings are also used by the PSTN to move data.

- Link layer multiplexing and switching

  Link layer multiplexing entails muxing several lower bandwidth links into a higher one (at n times the clock rate for the mux control), and vice versa.

  Link layer switching entails switching in time or space at the link layer, unbeknownst to the upper layers. Link layer switches/hubs/bridges are designed to not interfere with upper layers in any way.

  Link layer switching is cheaper and generally faster, but often duplicates some of the functionality of upper layers. That said, there are advantages to being able to ignore physical topology in network layer devices.

  Multiplexing can be done in time or in frequencies at the link layer. The PTSN and modern cable companies do this to provide Internet access, voice, cable and services on one connection. There is also wavelength division multiplexing for optical networks, but really, that is just a special case of FDM.

- Link layer error detection and control

  Error detection is done in several ways. CRCs implement a much stronger version of checksumming that uses sequences of bits a modulo arithmetic to detect errors. Multidimensional parity checking allows single bits to be detected AND corrected, as does hamming codes (parity with nested groups of bits).

  Forward error detection is a redundant checking field placed in the header to check against the FCM at the tail.

  In general, LL error detection is open-loop error control - it attempts to correct errors on its own to keep the network as a whole running smoothly.

  Error control mechanisms are the same as they were at layer 4. Typically, stop and wait is used since latency is low and it is very simple.

- Link layer components: evolution through bridges, hubs, and switches

  Out of necessity, a number of link layer devices were developed beyond NICs.

  - Problem: limited physical distance for ethernet buses. Solution - bridges which repeat everything received on one interface at the other interface, like a repeater.
  - Problem: bridges are pretty dumb, and invites unnecessary contention among segments. Solution - Use a learning bridge, which notes what MAC addresses came from which interface, and only repeat on other interface.
  - Problem: office topology did not match network topology, resulting in easily broken networks that were hard to administer. Solution - star-connection Hubs. Hubs were essentially a ring network in a box, and allowed traffic and debugging to be localized.
  - Problem: shared medium still sucks. I still have to share time and bandwidth with other hosts on my hub. Solution - L2 Switches. Replaces the hub by a switch on the MAC address, spatially distributing networks and allowing hosts full bandwidth when it is their turn. Note: routers are still needed to move traffic outside the network.
  - Problem: How can we achieve more sophisticated networks? Solution: VLAN, trunking, and tunneling.

- VLANs, 802.1Q trunking, and L2 tunneling

  VLANS are necessary to allow more sophisticated networks where multiple companies share a LAN, working groups are distributed across the internet, and many other edge cases. VLAN-enabled switches allow interfaces to be segregated from one another in terms of broadcast domain and visibility of frames.

  How does this work? IDs are manually configured by a sysadmin such that any each VLAN only knows about certain IDs. That is, each table will include only certain IDs for mapping, and otherwise it acts as a normal switch would. Observe that in order to reach another VLAN, you would need to call out to an external router, which would then redirect the frame back to the VLAN switch and forward it properly. There can be no direct mapping among VLANs. Some VLAN switches include a simple router for this purpose.

  This presents a problem: we need a dedicated output interface for each VLAN. This is clearly not scaleable. To get around this, we use trunking. Trunk links are special interfaces that belong to all VLANs. By inserting a special ID tag into the frame header, it is possible to determine which trunk line to send to, as if there were an external router. This requires still more configuration, but it makes the system more scaleable. See 802.1Q.

  L2 Tunneling allows the appearance of a local network to upper layers. Special components are used which transparently use IP to connect to remote equipment, making a virtual link to the remote LAN.

- ARP concepts and operation

  ARP solves the problem of layer 2 addressing. This Address Resolution Protocol handles directing an IP frame to the appropriate next hop on the link layer.

  Each host on a subnet, including edge routers, has a unique MAC address. Each host maintains an ARP table, containing a mapping from IP to MAC appropriate for the subnet at hand. ARP relies on the behavior of switches to send out broadcast messages with the source IP and MAC address. Initially, the ARP tables are empty, so the special MAC of all 1s is used to broadcast on the subnet. Routers will recognize the destination IP in their forwarding table, and prepare an ARP response to the sending party. A similar discovery process is used for intra-subnet communication. Thereafter, the MAC addresses are known and included in the frame header.

# 3 Example Questions

## Chapter 4

1. What do ping, traceroute, nslookup, and dig do? What protocols are they associated with?

   Ping uses ICMP to measure RTT and packet loss from one host to another.

   Traceroute uses ICMP messages with incrementing TTL to trace a path from one host to another. Triplets of ICMP messages are sent out, and make it one router further each time. When they reach the ultimate destination, the ICMP payload contains a phony upper layer port to induce a special error condition that is easily recognized.

   nslookup allows you to interact with DNS servers from the command line. It uses DNS protocol on port 54 to lookup a hostname given an IP address, an IP address given a hostname, a canonical host name, a mail exchange server, and many others.

   dig is used to query DNS servers for just about all possible record information.

2. What is the name of a network layer packet? Both layer 3 and layer 2 have packet switches. What is the difference?

   Network layer packets are called datagrams, but really everyone just says packet and can tell what you mean. Network switches include routing mechanisms, buffers, control mechanisms, and signalling logic, along with a forwarding table, switch fabric, and sometimes a NAT. Link layer switches contain simple HW to handle reading a frame, mapping its source MAC to the interface it came in on, and either directing it to a previously mapped output interface or broadcasting it to all interfaces. Typically, Layer 2 switches move frames within a single network, and layer 3 switches move packets among and within networks.

3. What are the main functions of the network layer in the control plane and the data plane? In the control plane, the network layer is responsible for managing traffic, routing, and signalling. In the data plane, the network layer is responsible for addressing and forwarding. Note that forwarding tables are computed based on signalling among routers, and so in some sense, forwarding is a joint effort of both planes.

4. What changed between IPV4 and IPV6?

   There were four major changes:

   - Checksum removed
   - Options allowed but now outside header to provide consistent size
   - ICMPv6 new version of ICMP
   - "Packet to big" Fragmentation not allowed.

   It also brought about IPV4 tunneling, since many parts of the Internet still only support IPV4. In that case, the IPV6 datagrams become the payload of the IPV4 datagrams.

5. What is the role of a forwarding table in a router? A forwarding table provides a pre-computed mapping of an IP address to an interface on a router. In general, this is accomplished at speed using programmable hardware.

6. What is the difference between routing and forwarding? Routing is concerned with the overall path a datagram will take as it moves along the edges of a network to its destination. This is akin to plotting the major highways you will take on a road trip. Forwarding refers to which of the possible interfaces a datagram will exit after entering a router in a network. Forwarding is akin to deciding which exit to take off a given highway, or which ramp to take off a roundabout.

7. Discuss why each input port in a high-speed router stores a shadow copy of the forwarding table. Shadow copies are necessary

   Shadow copies allow each interface to make forwarding decisions independently. If it were all done through the router's processor, it may become a bottleneck for performance under load.

8. Describe how packet loss can occur at input ports. Describe how this loss can be eliminated without resorting to more buffers.

   Packet loss on input ports occurs under high load, or when switching and forwarding cannot keep up with the influx of packets (say, if the switch is blocking or if the forwarding lookup is the bottleneck). Once way to mitigate this is to give each interface a shadow copy of the forwarding table, and to use a switching fabric which is non-blocking.

9. Describe how packet loss can occur at output ports. Can this loss be prevented by increasing the switch fabric speed?

   Packet loss at the output ports occurs when the switching speed is faster than the transmission rate on the output medium. This is ultimately driven by the input transmission rate. One way to prevent problems is to properly match the processing and switching speeds to work well together under maximum load.

10. What field in the IP header is used to prevent a packet from circulating the network forever?

    The TTL (time to live) field is decremented at each hop, and discarded if it reaches zero. This effectively limits how long a packet can circulate the network, preventing it from going forever.

11. Do routers have IP addresses? If so, how many?

    Yes, routers will need to have one IP address per interface

12. Suppose there are three routers between a source host and a destination host. Ignoring fragmentation, an IP datagram sent from the source host to the destination host will travel over how many interfaces? How many forwarding tables will be indexed to move the datagram from the source to the destination?

13. Suppose an application generates chunks of 40 bytes of data every 20 msec, and each chunk gets encapsulated by a TCP segment and then an IP datagram. What percentage of each datagram will be overhead, and what percentage will be application data?

14. Suppose you purchase a wireless router and connect it to your cable modem. also suppose that your ISP dynamically assigns your connected device (that is, your wireless router) one IP address. Also suppose that you have five PCs at home that us 802.11 to wirelessly connect to your wireless router. How are IP addresses assigned to the five PCs? Does the wireless router use NAT? Why or why not?

15. What is a private network address? Should a datagram with a private network address ever be present in the larger public Internet?

    Private network addresses are used either inside a NAT subnet or inside a router subnet to internally identify end systems. They are called private because they are not directly addressable from the Internet, they must filter through the router. A datagram with a private IP should never appear in the public Internet.

16. When a large datagram is fragmented, where and how are those fragments reassembled?

    Fragmented packets are reassembled at their final destination. All fragments have the same ID field, and all but the last fragment will have a flag indicating there are more packets. The fragment offset is used to sort the fragments into the original order, as they often arrive in the wrong order.

17. How does generalized forwarding differ from destination-based forwarding?

    Destination-based forwarding uses only the destination IP address to determine the next hop. Generalized forwarding can use any number of fields in the header to make the decision, allowing multiple dimensions for routing and classification and in turn more sophisticated networks.

18. Describe the differences between forwarding tables that use destination-based forwarding and those used by OpenFlow.

    Openflow uses any of the header fields and/or the ingress interface to match a rule stating how to forward.

19. Describe the "match plus action" operation of an SDN controlled router. Name three fields that might be matched against.

    The match is a prefix or field in the header that is used to map to a specific forwarding action. All incoming packets that find a match trigger the given action (which need not be forwarding, it could be anything). A packet might be matched by the interface it arrived on, the source MAC address, or its TCP port.

## Chapter 5

1. Compare and contrast the properties of centralized and distributed routing algorithms.

   Centralized routing algorithms generally require the state of the entire network to make a decision - gathering that information can be costly and computing the optimal paths can have high complexity/

2. What is the "count to infinity" problem?

   The count to infinity problem refers to the phenomena where a distance vector algorithm takes a long time to converge if a link cost increases substantially. Basically, the change propagates very slowly, and doubles back on itself, causing unpredictable behavior as it slowly settles.

3. Is it necessary that every autonomous system use the same intra-AS routing algorithm? Why or why not?

   No, each AS is given autonomy over what it uses internally. This makes it easy to deal with the large variety of ISPs and ASs in the wild, and does not impose any unnecessary regulation.

4. Why are different inter-AS and intra-AS protocols use in the internet?

   Same as above. There is only one inter-AS protocol, BGP.

5. True or False: When an OSPF router sends its link state out, it is only sent to its immediate neighbors. False. OSPF uses a link-state algorithm, and thus floods information to all nodes at once.

6. Why does OSPF have areas? What are the areas for?

   Areas are necessary to divide an ISPs domain into sections, for any number of reasons. This creates an artificial "backbone" layer, which uses OSPF messages to direct traffic among interior ASs.

7. Describe BGP

   Provides a means to obtain subnet reachability info from neighboring ASs (eBGP). Propagates reachability info to all routers internal to the AS (iBGP).

   BGP peers use TCP to exchange BGP messages. Initially, BGP peers exchange their entire BGP routing table. Incremental updates are sent thereafter which reduces bandwidth usage and processing overhead. Keep alive messages sent periodically to verify connection.

8. Define and compare subnets, prefixes, and BGP routes

   A subnet is the range of IP addresses which are reachable from a base IP address provided by a prefix. The prefix is shared by all hosts on a subnet. A BGP route defines the AS-path to a subnet prefix through the internet. It involves high-level aggregations of many IP prefixes in each of the ASs, and there may be many BGP paths to a subnet. Conflicts are resolved by the longest prefix - it is always given precedence over shorter ones.

9. How does BGP use the NEXT-HOP and AS-PATH attributes? BGP uses the NEXTHOP attribute to determine the IP address of the edge router for the nearest AS along the AS path. This allows the sending party to get its PDU out of the AS. The ASPATH describes a path through the Internet as a series of AS IDs followed by a subnet destination. There may be many such paths advertised, which are propagated through the internet by BGP messages.

### Chapter 6

1. If all the links in the Internet provided reliable delivery service, would it be necessary to implement reliability for TCP or IP?

   Yes, because of the way routing and forwarding are done, packets could still be lost due to TTL expiring or being dropped at input/output queues.

2. Suppose two nodes start to transmit at the same time a packet of length L over a broadcast channel of rate R. Will there be a collision if the propagation delay is less than $\frac{L}{R}$?

   Since propagation delay is less than $\frac{L}{R}$, it is less than the transmission delay, so there will be no collision.

3. Suppose nodes A, B, and C each are joined to the same broadcast LAN. if A sends thousands of IP datagrams to B labeled with B's MAC address, will C's NIC process these frames? What if A sends frames with the broadcast MAC address?

   No. NICs are designed to only process frames with their MAC address or a broadcast MAC address. If the frame was a broadcast MAC address, then C would indeed process the frames, and see if the IP address matched its own or an IP in its forwarding table (if C is a router).

4. How big is the MAC address space? What about IPV4 and IPV6?

   MAC address space is huge ($2^48$). IPV6 is even larger ($2^128$). IPV4 is not large enough for current needs ($2^32$).

5. Consider the count-to-infinity problem in the distance vector routing. Will the count-to-infinity problem occur if we decrease the cost of a link? Why? How about if we connect two nodes which do not have a link?

   The count to infinity problem only applies to links which have increased in cost (good news travels fast, bad news sucks). If we decrease the cost of a link, the good news will quickly converge. If we add a new link, it could either provide a faster path or a slower path between the two nodes. In the case of the former, it would be a cost decrease and thus would converge quickly. Otherwise, it would not trigger sending out a new distance vector since it would not be the shortest path.

6. Why would the token-ring protocol be inefficient if a LAN had a very large perimeter?

   It would take a long time for the token to make its way around the ring, even if each host only got to transmit one frame.

7. Consider two subnets which are interconnected by a router. The router has two ARP modules, each with its own ARP table. Is it possible that the same MAC address appears in both tables? Explain.

   It is not possible that the router has the same MAC address in both tables. Each NIC gets its own MAC address, so from a router, they would all be unique. The only possibility is if both subnets shared another router joining them, but this is not what is described (it also wouldn't work without VLANs or other fancy equipment).

8.

# References


[1] J. Kurose and K. Ross. *Computer Networking: A Top-Down Approach, 7th Ed.* Pearson, London, 2017.

[2] J. Sterbenz. *Introduction to Communication Networks: Link Layer and LANs.* URL: `https://www.ittc.ku.edu/~jpgs/courses/intronets/lecture-link-lan-intronets-display.pdf`.

[3] J. Sterbenz. *Introduction to Communication Networks: Network Layer.* URL: `https://www.ittc.ku.edu/~jpgs/courses/intronets/lecture-network-intronets-display.pdf`.

[4] J. Sterbenz. *Introduction to Communication Networks: Network Routing.* URL: `https://www.ittc.ku.edu/~jpgs/courses/intronets/lecture-routing-intronets-display.pdf`.