# TCP/IP Summary

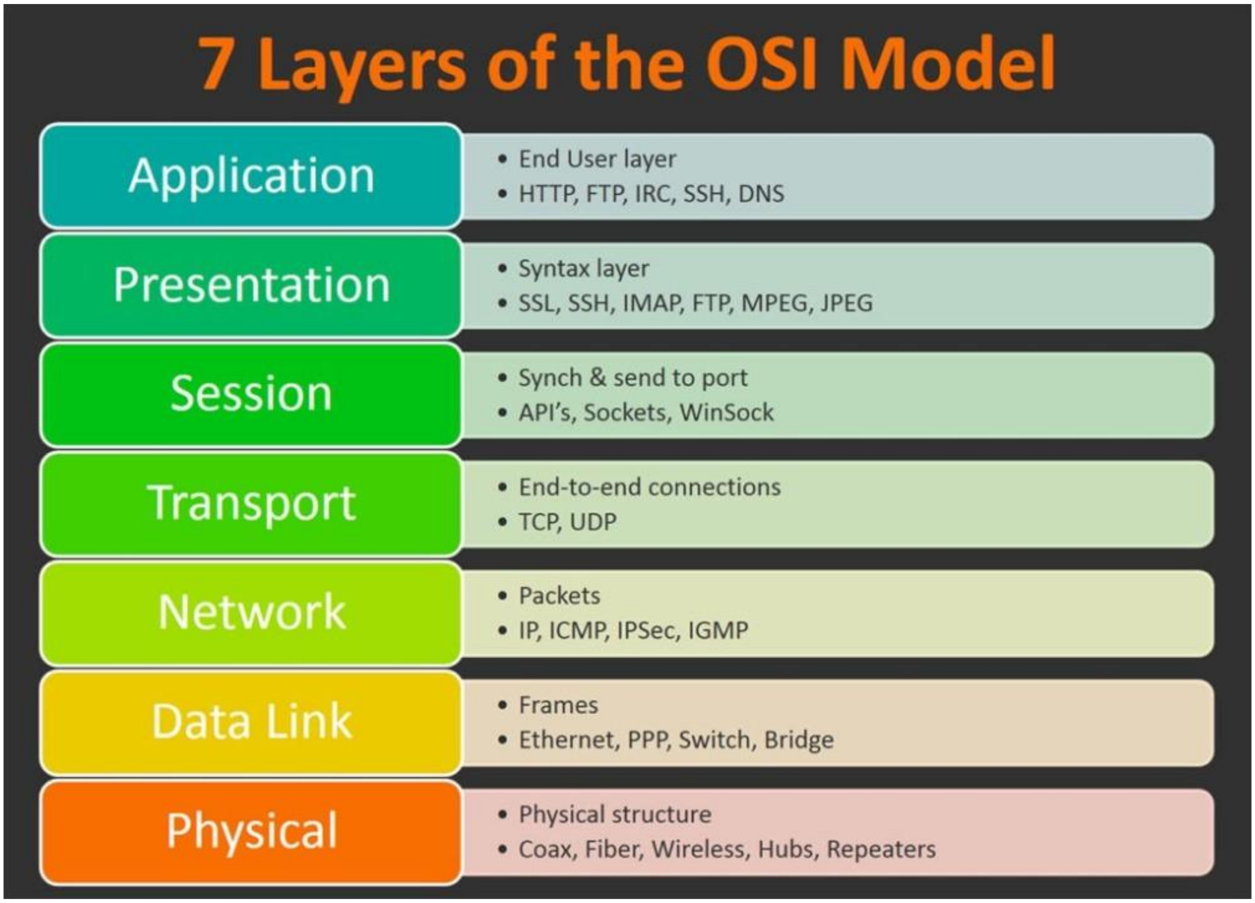
## Chapter 1

### *Osi Model*

The OSI model is the layering model of the internet. There are seven logical layers:

1. Physical
2. Link
3. Network
4. Transport
5. Session
6. Presentation
7. Application

A concept closely tied to this is multiplexing, which is the idea that different protocols can co-exist under the same infrastructure.



7 Layers of the OSI Model

| | |
|---|---|
| Application | • End User layer<br>• HTTP, FTP, IRC, SSH, DNS |
| Presentation | • Syntax layer<br>• SSL, SSH, IMAP, FTP, MPEG, JPEG |
| Session | • Synch & send to port<br>• API's, Sockets, WinSock |
| Transport | • End-to-end connections<br>• TCP, UDP |
| Network | • Packets<br>• IP, ICMP, IPSec, IGMP |
| Data Link | • Frames<br>• Ethernet, PPP, Switch, Bridge |
| Physical | • Physical structure<br>• Coax, Fiber, Wireless, Hubs, Repeaters |

## *Tcp/Ip*

Tcp is a transport layer protocol. It is 'transport control protocol'. It has the following characteristics:

1. Connection-oriented, stream-based protocol focused on in-order message delivery.
2. So it does not preserve message boundaries.
3. Both hosts must acknowledge each other's messages.
4. Deals with packet loss, duplication, rate limiting, and reordering (flow control and congestion control).

## *Udp*

Udp is a transport layer protocol. It is 'user datagram protocol'. Is has the following characteristics:

1. Enforces message boundary.
2. Supports multicast delivery.
3. No flow or digestion control

## *Ports*

Port numbers are unsigned 16 bit integers. They are abstractions and represent nothing physical. Ports are like highway toll booths and allow data from various protocols to transfer information through (e.g. just as telephone extensions numbers allow a single person to rendezvous with several different people).

# Chapter 2

## *Ip Address Classes*

### Unicast

There are five classes of Ipv4 addresses, A, B, C, D, and E. D is multicast, E is reserved, A, B, and C are unicast and have their address space split between network addresses and host addresses. This allows network administrators to partition their address space between subnets and hosts.

### Subnet Mask

The subnet mask is an assignment of bits used by a host to determine how the subnet information is partitioned from the host information in an IP address. Subnet masks are purely a local matter at the site (internet routing traffic does not require knowledge of the subnet mask.

At the site you can also use different length subnet masks in the same network. This allows for the flexibility of different subnetworks to be set up with different numbers of hosts.

### Broadcast

Each Ipv4 subnetwork has a special address is reserved to the subnet broadcast address. The subnet broadcast address is formed by setting the network portion of an IPv4 address to the appropriate value and all the bits in the Host field to 1. For example 128.32.1.14 becomes 128.32.1.255

There are some special-use addresses. These are address ranges used for special purposes. For example, 192.168.0.0 are addresses for private networks and never appear on the public internet.

### Multicast

- Multicast addresses are sometimes called group addresses and identify a group of host interfaces rather than a single host. Under software control, the protocol stack of each host can join and leave a multicast group.
- When a host sends a multicast datagram, it creates one using its own unicast address as the source, and a multicast address as the destination.
    - All hosts in the group will receive the datagram.
- The sender is not generally aware of how many hosts will receive the datagram.
    - This is known as ASM (any-source multicast). Here any sender can send a datagram to the group, and anyone who joins the group can receive it.
    - There is also source-specified multicast, where there is only a single sender per group.
        - In this case a host specifies the address of a channel, which comprises both a group address and a source address.

# Chapter 3

## *Link Layer*

- The PDU (protocol data unit) on the link layer is usually called a Frame (to distinguish it from a segment (transport) or packet (network) layer PDU).

- The basics of the frame include a source and destination mac address (each 6 bytes) and a type field (specifies the packet type contained within (e.g. IPv4)) that doubles as a length field (total header is 14 bytes). There is also a checksum at the end (4 bytes). The standard size of the frame is 1518 bytes (which includes the header and checksum), but was extended to 2000 bytes.

# Chapter 4

## *ARP: Address Resolution Protocol*

- Responsible for the dynamic mapping between the IPv4 address (network layer addaress) and the hardware address (link layer) used by various network technologies.
- ARP is most relevant for systems sharing the same IPv4 prefix where its normal for the link layer to be able to deliver a single message to all attached network devices (or to reach the router when the destination of a datagram is not on the subnet).
    - ARP sends an ethernet frame called an ARP reques to every host on the shared link-layer segment (link-layer broadcast).
        - The request contains the IPv4 address of the destination host and seeks an answer to the question ‘if you are configured with IPv4 address XYZ as one of your own, please respond to me with your MAC address’.
            - In this message, the Ethernet destination address is the broadcast address, the ethernet frame type field is 0x0806.
                - For the ARP request, all fields in the ARP message are set except for Target Hardware Address, which is set ot 0.
            - All systems within the same broadcast domain receive the ARP request. The attached subsystem that contains the IPv4 address replies to the request with its corresponding Mac address.
                - The receiving host (the one that broadcasted the arp request) later stores the mapping in memory.
            - Once the mapping is established, the original sender that forced the ARP request can now send its original datagram.
            - Direct delivery can now be achieved since the sender can fashion an ethernet frame with the ethernet address leaned by the ARP exchange.

## *ARP Cache*

The ARP cache that stores the dynamic mapping IPv4 and hardware address have a 20 minute timeout for each entry every time it is used. (the timeout is restarted each time the entry is referenced).

# Chapter 5

## *IP: Internet Protocol*

Ip is a best-effort connectionless datagram delivery system. There is no guarantee of the fate of a packet, and all error handling and reliability guarantees must be implemented by the upper layers.

- The basic IP header is 20 bytes and contains:
    - Total packet length
    - Header checksum
    - TTL – the number of jumps (routers) in which a datagram can pass.
        - The number starts at ~64 and, when it reaches 0, is thrown away. The sender is notified with an ICMP message.
    - Protocol - Protocol contained in the payload of the packet (e.g. UDP, TCP).
    - Source and Destination IP
- IP demultiplexes incoming IP datagrams to a particular transport protocol based on the value of the protocol field in the header. This means that the port numbers can be made independent among the transport protocols.
    - This means two **different** servers can use the same port number and IP address as long as they use different transport protocols.
- IP packets can fragment (be split up) because link layer framing imposes an upper limit on the maximum size of a frame that can be transmitted. IP employs fragmentation and reassembly.
    - Fragmentation involves splitting up a message into multiple parts. Reassembly involves putting them back together
        - Reassembly only happens at the destination location. The reason being is that multiple IP fragments belong to the same initial unfragmented packet can take different routs to their source. If their sources are different, and intermediate routers are trying to reassemble them, no single router will have all the information required to reassembly the fragments (neither will it have all the fragments).

# Chapter 7

## *Network Address Translation*

- Network address translation was primarily conceived to deal with the depletion of available IPv4 addresses.
- With NAT, internet addresses need not be globally unique, and can consequently be reused in different parts of the internet, called address realms (large numbers of hosts can share one or more globally routable IP addresses).
- When incoming and outgoing traffic passes through a single NAT device that partitions the inside (private) traffic from the outside (public) traffic, all internal systems can be provided internet connectivity as clients using locally assigned private IP addresses.
    - This is achieved because NAT actively rewrites the addressing information in each packet in order for communication between a privately addressed system and a conventionally addressed internet host to work.
        - Modifying an address at the IP layer also requires recalculating the header checksum and potentially other checksums at the transport layer.
    - One implementation of NAT is NAPT (or IP masquerading), which involves rewriting the address of outgoing packets to a single address, and assigning a unique port to that traffic to distinguish it from another host in the private network (so that incoming traffic can be routed to the correct host).

- If a service is using a local Ip address and wants to be made public to the internet, it must be found through the external address of the NAT. The limitation here is that there is only one set of port numbers for each of its own IP-transport protocol combinations, thus if the NAT has only a single external IP address (like in the case of Ip masquerading), only on internal machine can have its traffic forwarded for each transport protocol.

## *Firewalls*

- The two major types of firewalls commonly used include *proxy firewalls* and *packet-filtering firewalls*.
    - The main difference is the layer in the protocol stack at which they operate.
        - Packet filtering firewall is an internet router that drops datagrams that fail to meet specific criteria.
            - The simplest firewalls are stateless, they treat each datagram individually
        - Proxy firewall operates as an endpoint of TCP and UDP transport.
            - These are hosts running one or more application layer gateways

# Chapter 8

## *ICMP*

- Protocol that provides a way to gather diagnostic information (e.g. how long a packet roundtrip takes and if its been dropped)
- It provides delivery of error and control messages that may require attention.
- There are two groupings of ICMP messages relating to problems with IP datagram delivery:
    - Error messages – (e.g. destination unreachable, malformed packet)
        - When one is sent, it contains a copy of the full IP header from the offending (original) datagram,plus what caused the error to be genrated
    - Query information – (e.g. echo request and echo reply)
- The ICMP payload is the first portion of the original datagram..
- ICMP messages are usually rate limited with a token bucket.

# Chapter 9

## *Multicast*

- There are many applications that deliver information to multiple recipients. For example interactive conferencing. These types of services use multicast (giving a separate copy to each destination would be inefficient).
- Multicasting involves broadcasting a single message only to receivers that are interested in it. This is achieved by having the receivers independently indicating interest. This involves some complicated multicast state that must be maintained by hosts and routers as to what traffic is of interest to what receivers.
    - In the TCP/IP model of multicasting, receivers indicate their interest in what traffic they wish to receive by specigying a multicast address and optional list of sources. This

        information is maintained as soft state which hots and routers will update regularly or will time out and be deleted. When deleted, delivery of multicast traffic to the receiver will cease.

- Generally only user applications that make use of UDP take advantage of broadcasting and multicasting (since TCP is a connection oriented protocol).
- A multicast MAC address has the low-order bit of the high order byte set to 1.
- Broadcast addresses are formed by placing all 1 bits in the host portion of the address. For example (10.0.0.0/25) -> 10.0.0.127 (turning the last 32 – 25 = 7 bits on)
- ‘There are two types of multicast:
    - Any source multicast (ASM) – you specify the group address you want to join, irrespective of the snder.
    - Source specific multicast – Allows end stations to explicitly include or exclude traffic sent to a multicast group from a particular set of senders.

# Chapter 10

## *Udp*

- Udp is a datagram-oriented, transport-laytter protocol that preserves message boundaries. It does not provide error correction, sequencing, duplicate elimination, control flow or congenstion control.
    - It can provide error detection via the checksum in its header
        - A checksum of 0x0000 indicates that the user has not compute a checksum. If the checksum just happens to be 0x0000, the twos complement is taken (0xffff)
    - Source ports in the UDP header are optional (and usually set to 0 if the sender of the datagram never requires a reply)

# Chapter 11

## *Domain Name System*

- DNS is a distributed client/server networked database that is used by TCP/IP applications to map between host names and IP addaresses.
    - TCP and Ip protocol implementations know nothing about DNS, they opeate with their own addresses.
- Applications use resolvers to contact one or more DNS servers to perform lookup tasks against a zone database, such as converting a ost name to an IP address and vice cersa. Resolvers then contact a local name server, and this server may act recursively to contact one of the root servers or otherservers to fulfill the request.
    - Most DNS servers, and some resolvers, cache information learned in order to provide it to subsequent clients for some period of time called the time to live.
- DNS is most commontly used to determine the IP address that corresponds to a particular name, but can be used for many other things. These other thing’s are mapped by objects called resource records and include but are not limited to the following:
    - Data type resource records

- Query type resource records
    - Meta type resource records
- The A and AAAA records are used to provide an IP address given a particular name.
- CNAME records provide a way to alias another domain.
- PTR (RR) records take an IP and return an domain.

# Chapter 12

## *TCP: Transmission Control Protocol*

- TCP, called the sliding window protocol, is a connection-oriented and streaming-based transport layer protocol focused on congestion and flow control as well as in-order data delivery. It does not preserve message boundaries.
    - The absence of record boundaries means that if sender A writes 30 bytes, then 50 bytes, the receiver may receive an 80 byte chunk, 1 byte then 79 bytes, these are all valid combinations.
    - The guarantee of in-order delivery means that TCP may be forced to hold on to data with larger sequence numbers before giving it to an application until a missing lower sequence number segment is filled in.
- The TCP header contains:
    - the source and destination port which, combined with the source and destination Ip addresses, is able to uniquely identify a connection. This is often called an endpoint or socket.
    - A TCP checksum that covers the TCP header and payload
- Unlike in UDP, packets go in both directions. A receive may send it own data but at the mionimum it needs to acknowledge incoming data. When it does so (by sending an Acknowledgement with the sequence number contained in the received packet), the sender releases a copy it stores (for redundancy, in the case it needs to retransmit) and sends the next packet in its queue.
    - When a packet is sent, a timer is initialized. If an ack for the corresponding packet isn't heard within the timer, the packet is resent (assumed to be lost in transit).
- There may be cases where the receiver is overwhelmed by the sender or vice versa. In this case, the receiver can signal to the sender which window size it should use (placed in the TCP header). This is called flow control (in particular, a *window advertisement* or a *wwindow update*).
    - The receiver and sender both negotiate an optimal window size over time, but what if the hardware in between the sender and receiver can't keep up (for example, the network in between is congested and can't keep up, leading to packet loss). This is addaressed by a special form of flow control called congestion control.
        - Congestion control involves the sender slowing down so as to not overwhelm the network between itself and the receiver

# Chapter 13

## *TCP: Connection Management*

- Since TCP is connection-based, it needs to manage connection state.
    - A connection goes through three phases: setup, datatransfer, and teardown.
        - Setup – a typical setup is a three way handshake.
            - Syn, Syn + Ack, Ack
            - The first Syn is sent by the client initializing an *active* open, the next SYN performs a *passive* open.
                - There is also a *simultaneous open* (each end transfers a syn before sending a syn to the other side)which occurs when both hosts send each other a syn at the same time. In this case the handshake is four-way
            - If a SYN is sent and no SYN + ACK is heard back, the active opener employs exponential backoff (where the time it takes before the next retry doubles).
        - Teardown – modified three way handshake
            - Fin + Ack, Ack, Fin + Ack, Ack
                - The sender (active closer) sends a FIN + ack specifying the current sequence number the receiver expects to see (as well as acking the last data)
                - The passive closer sends an Ack, it then sends a Fin + Ack,
                - The active closer then sends an ack back
            - The reason the teardown is 4-way is that it's possible to be in a half-close state. This is uncommon but it operates as "I am done sending data, so send a FIN to the other end, but I still want to receive data from the other end until it sends me a FIN"
            - PSH is usually sent with FIN as it indicates that the server has no additional data to send.
- **Sequence numbers** for the first instance of a single connection are chosen randomly for security reasons. Subsequent connections between the same hosts (new instantiations) are always larger as to make sure there is no overlap in sequence numbers between the current and previous connection (at least during the end of the first relative to the start of the second)
- **Selective acknowledgements** occur when a receiver receives out of order sequence of datablocks that cause a hole in the stream. The receives selectively acknowledges the later bytes, so that the sender can retransmit only a portion of what is missing.

# Chapter 14

## *TCP: Timeout and Retransmission*

- sTCP provides a reliability guarantee for data transfer. This means TCP resends data it believes is lost. To decide which data to resend, TCP depends on a continuous flow of acknowledgements from receiver to sender. When data sengements or acknowledgements are lost, tcp initialized a retransmission of the lost data.
- There are two ways  to signal that data has been lost

    - Timer approach – TCP sets a timer when it sends data, and if the data is not acknowledged when the timer expires, it retransmits (sender-based).
        - TCP connection is abandoned after three failed retransmissions and 100s
        - TCP is allowed to repacketize a segment (on a retransmit), which can increase performance. This means it doesn't need to resend the same segment (it can send a bigger segment)
    - Ack approach – When the receiver sends back acks for data that was just sent (or SACKs) (for example, sender sends 10, then 12. Receiver sends ack for 10 twice (or dup threshold is surpassed)), (receiver based).
        - SACKs are enabled during the establishment of a connection when the SACK-permitted option is available.
        - Each SACK block contains two 32-bit sequence numbers representing the first and last sequence numbers of the missing data
        - TCP sometimes delays sending an ack for some amount of time in hope that it can be combined with the ack that it needs to send to some other application. This is a form of piggybacking that is often used with bulk data transfer.
- While packet loss is relatively common, there are other packet anomalies. Out of order delivery and duplication of packets may also happen.

# Chapter 15

## *TCP: Data Flow and Window Management*

- TCP uses the window size to ensure that the sender and receiver dynamically adjust the amount of data they send between them (such that one doesn't overrun the other)
    - The window size field communicates how much space the sender of a segment has reserved for storing incoming data the peer sends (how large the receive buffer is). When the window size is 0, this communicates there is no space left.
        - Once this happens, the sending TCP begins to probe the peer's window to look for an increase in the offered window.
            - It will also do this on a timer incase the receivers window offset was lost
            - Normal TCP never gives up on window probes, while it does eventually give up trying to perform retransmissions.

# Chapter 16

## *TCP: Congestion Control*

- TCP congestion control is most important during bulk data transfers. The congestion control algorithm(s) attempt to prevent the network from being overwhelmed by large amounts of traffic.
- The most basic idea can be seen with a sender and a receiver. If a sender's rate is > than the rate the receiver can process, the receiver must buffer data. Once its buffer grows to the bounds of its capacity, it must start dropping updates.
- **I NEVER ACTUALLY FINISHED READING THIS CHAPTEr.**

# Chapter 17

## *TCP: Keepalive*

- When no data is flowing, TCP sends nothing (no data flows across an idle tcp connection).
- In some cases it may be useful for a server to become aware of the termination or loss of a connection with a peer (which may not be evident if a router connecting the two goes down and no data is being sent). In other circumstances it is desirable to keep a minimal amount of data flowing over a connection, even if the applications do not have any to exchange.
    - **TCP keepalive** provides this feature. It is driver b a keepalive timer. The timer fires, a keepalive probe is send and the peer receiving the probe responds with an ACK.
        - A downside is that, if an intermediate router is rebooting and we send a keepalive, tcp will incorrectly think its peer host has crashed.
        - TCP Keepalive is a feature most useful for a server that wants to know if its client host crashed (and is only expecting a relatively short-duration dialogue). A half-open connection is valid, and a server will wait forever after it sends a segment before it hears back. **The keepalive feature is intended to detect half open connections.**
- Either side (or both) can request TCP keepalive, which is off by default. If there is no activity after keepalive time, the other host is probed over a keepalive interval until some threshold is reached (or until it has heard back). The connection is terminated if nothing is heard back.
    - A keepalive probe is an empty (or 1 byte) segment with a sequence number that **is equal to one less the largest ACK number seen from the peer so far**.
        - Because this sequence number has already been acked by the receiving TCP, the arriving egment does no harm. It simply elicits an ACK that is used to determine whether the connection is still operating.
- When a TCP keepalive arrives at a host that has no more knowledge about the connection (e.g. the host has been rebooted), the host will send a reset segment back to the sender. This signals that the connection is no longer active and should be terminated.