(V 0.3 7-nov-2019) DS · Ex 1 · Supplementary · 17th-December-2018

1. [2] Describe the responsibilities and Interfaces of Internet Architecture layer 3

   The IP protocol, running at hosts and routers, allows locating Internet hosts but, the applications running at those hosts cannot be located by using only IP, it must supplemented with an *application key* that represents the applications running in the hosts. That *application key* is included in the UDP protocol data unit, the UDP datagram. Conceptually, from a network architecture standpoint, that key is known as a multiplexing key. The standard name given to that multiplexing key is port and its width is 16 bits. This is the single mission assigned to UDP: exporting port numbers so applications running in hosts can be uniquely contacted. In fact, UDP will not fulfill any of the responsibilities proper of the Internet Service Model beyond the capability of representing applications with port numbers.

   By contrast to UDP, TCP does attempt to compensate for some of the facets of the Internet Service Model. The most important responsibility of TCP consists of *reliable transmission* or providing guarantees of retransmission of lost packets and and of packets that contain errors and of suppressing duplicated packets.

2. [1] What fields of a UDP segment form their multiplexing key?

   The **destination port** (Contrast the multiplexing keys at play in Internet Architecture layer 3)

3. [1] Explain the Failure and Mobility transparencies of distributed systems. Put an example of each.

   Failure: Some components of distributed systems are designed and implemented in such a way that when some kinds failures take place, their functionality remains intact, thereby resulting transparent to their users. Example: A web server for a specific topological area of Internet undergoes an OS crash and the CDN (Content Distribution Network) monitor, reroutes web requests to this server to another server. Users may only undergo a slight increase in web server bandwidth, *i.e.,* functionally, the web serving remains intact, therefore, the failure results transparent.

   Mobility: A distributed system offers this transparency if applications don't have to know exactly where a resource is located; the distributed system has the capacity to transparently locate the resource without requiring this knowledge at the hosts. Example: Mobile IP, a mobile user migrating from one network to another nearby network. The fact that the host has moved networks results transparent to the user and applications running at the host.

4. [2] Build a 3-tier distributed system that uses real C/S technologies. List the technology you have chosen at each tier, the intervening C/S protocol at play between each tier and the responsibility of each.

   - Presentation: Java Server Pages
     - Protocol = http
     - Presentation and interaction with the user
   - Business: Java Servlets
     - MySQL
     - Compute the business model of the 3-tier application, *i.e.,* the logic governing the transformations of data applying to the specific productivity context where the application is automating some aspect of their *business*
   - Persistence: Hibernate and MySQL
     - Save data in persistent storage such that it can be retrieved every time the computer system is rebooted, etc

5. [3] Discuss whether a TCP module can be used for clock synchronization between client and server according to the synchronization procedure that we explained in the lectures. Assume that no change can be done on the structure of the TCP protocol data unit, including the options field.

   Clock synchronization can not be accomplished by using TCP since it does not provide the Receive timestamp, thereby hampering the calculation of T_{RESP} and Rtt_N. Also, TCP provides no guarantee that the software clock used for generating the T_{sval} timestamp is the system's real time of day, but simply a monotonous real-time clock, *i.e.,* a clock that can measure time differences (A time-difference clock) but that can not provide the real time of day.

6. [2] Explain the essential characteristics of the NTP protocol

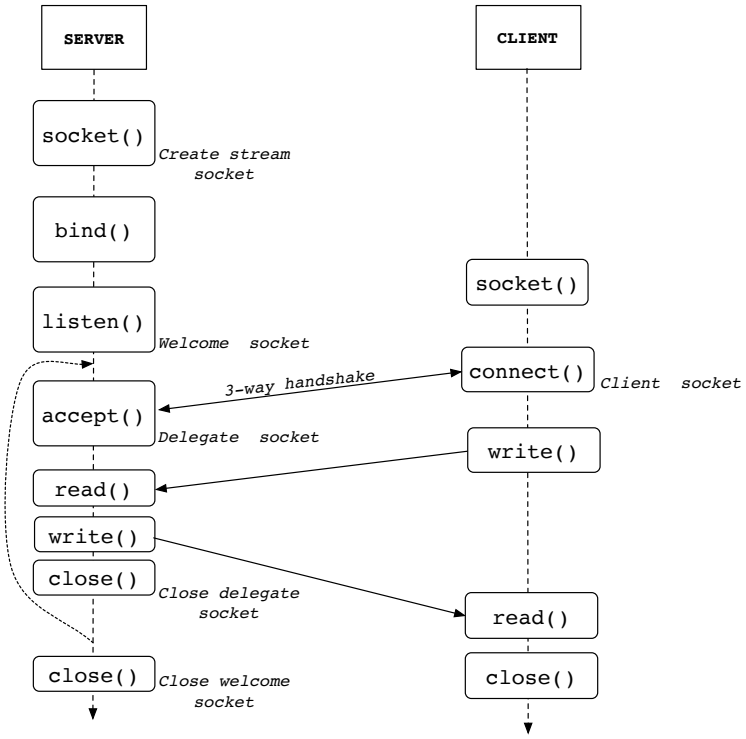
NTP stands for Network Time Protocol. The transport used by this protocol is UDP and serves for synchronizing Internet host clocks by having the clock server send its local time as the response to the time request sent by the requester.

The clocks of the hosts running NTP are organized into a hierarchy where atomic clocks are located on the topmost level. Levels are technically known as strata (The Latin plural of stratum) with the topmost level corresponding to stratum 1. Typically a common Internet host having an off-the-shelf quartz oscillator is located at stratum 3 or greater. This protocol is based on a probabilistic algorithm that usually accesses several, differing quality clock servers.

7. [1] Explain the essential aspects about the adjtime() Posix library.

The adjtime() library function monotonically updates a system's software clock by accelerating it or by decelerating it, according to whether the clock is behind or ahead with another, reference clock. The first argument in its formal parameter list is the offset in μs that is to be added to the system software clock so it will be in sync with the reference clock.

8. [1] Develop an interaction diagram between a TCP Client socket and a TCP Server socket that expresses the full life-cycle of a TCP connection. Remark the passive open and active open interactions and clearly explain which protocol messages are exchanged (SYN, ACK-SYN, ACK, FIN, etc).



9. [6] Host A wants to synchronize its clock with host B's clock, to that end, it performs 10 time requests by using ICMP. The timestamps sent by B, in response to each timestamp request, are included in the table below.
   a. Compute the time that must be set at host A right after receiving the last timestamp so that this clock is synchronized with host B's clock. Explain the calculation process.
   b. What argument would pass to adjtime() in the preceding question?

Check out http://paloalto.unileon.es/ds/SG/SG-DS-Clocks-NTP.pdf at pg. 32 for an exercise like this one solved.

10. [2] Explain the Nagle's algorithm. Remark its essential purpose.

Nagle's algorithm was devised for avoiding sending a small packet (One which size is less than MSS) when some ACK is pending, thus, it avoids flooding the network with small packets since these result too costly for the networks. It was of

particular importance when networks didn't offer the bandwidth and low latencies available today, mainly due to the availability of optical fibers.

11. [1] [M] Assume two Internet hosts C and S form a TCP connection. S sends a TCP segment to C containing the following relevant field values: ACK 3500, AWS 0. Mark the statements that are undoubtedly true about the foregoing TCP connection:

    - ☐ a. The flow control window size announced by C is 3500
    - ☐ b. The congestion control window size announced by C is 3500
    - ☐ c. The free space at the Receive Buffer of S may not be 0
    - ☐ d. The free space at the Receive Buffer of S may be 0
    - ☐ e. The Receive Buffer size of S should be about the same size as MSS
    - ☐ f. The free space at the Receive Buffer of C is 0

12. [2] [M] Consider TCP flow control and the channel comprised of the transmitter (C) and the receiver (S) in a TCP connection involving hosts C and S. Assume that the last segment sent by receiver S has the following field values: ACK 1000, WS 1000 and that this TCP uses no option. Which of the following options represent TCP valid messages that could be sent by C in the foregoing scenario?

    - ☐ a. 0-Byte segment having SN = 2000, ACK =3200, WS = 700
    - ☐ b. 2-Byte segment having SN =1999, ACK =3200, WS = 700
    - ☐ c. 1000-Byte segment having SN = 1001, ACK =1249, WS = 0
    - ☐ d. 18-Byte segment having: SN = 1983, ACK =1850, WS = 700
    - ☐ e. 998-Byte segment having SN = 1001, ACK =1850, WS = 0
    - ☐ f. 498-Byte segment having SN = 2500, ACK =1851, WS = 699

13. [1] [M] Consider a TCP connection between client C and server S. Assume S sends a segment containing no data which ACK bit is activated and having ACK SN = 2000 (TCP field ACK Sequence Number). This segment is lost. Tick the true statements among the following:

    a. ❑ A TCP RTO timer event could indeed retransmit the lost ACK segment
    b. ❑ The bytes contained in the last segment received by S comprise SN through 1999
    c. ❑ Receiver S has received a total of 2000 Bytes in order (In sequence) and without errors
    d. ❑ An ACK segment later sent by S and containing an ACK SN greater than 2000 would render the retransmission of the lost segment unnecessary altogether
    e. ❑ A TCP RTO timer at C could retransmit the lost ACK segment

14. [2] [M] The client, C, of a TCP connection receives a segment from the server, S, which contains the following field values: ACK bit set, ACK SN = 3500 and AWS = 1000. Tick all the true statements about what could happen right after this time point in the TCP connection (Assume no TCP option has been enabled).

    a. ❑ C could transmit a window of 3500+1000 Bytes
    b. ❑ C will transmit a window of 1001 Bytes
    c. ❑ C will transmit a window of 1000 Bytes
    d. ❑ The most advanced ACK that S could send to C as a response to a correct segment sent from C to S, would contain an ACK SN = 4500
    e. ❑ C could transmit a segment carrying no data and having SN = 3499
    f. ❑ C could transmit a segment carrying no data and having SN = 3500
    g. ❑ C could transmit a segment carrying no data and having SN = 3800

15. [3] [M] Tick the true statements about a TCP connection when the TCP at a host A sends a segment with the flag ACK set and ACK SN = 2000

    - ☐ a. The last segment received by A carried data through SN 1999 and the SN of the next expected segment is 2000
    - ☐ b. The last segment received by A carried data through SN 1999 and the SN of the next expected segment is 1999
    - ☐ c. The TCP module of host A has successfully received all the segments containing data bytes through SN 2000 and the next expected SN is 2001
    - ☐ d. The last received segment carried data bytes through SN 2000
    - ☐ e. The TCP module of host A has received a contiguous block of data that comprises SN through and including SN 1999
    - ☐ f. The TCP module of host A has received a contiguous block of data that comprises bytes from the Initial Sequence Number through and including SN 1999
    - ☐ g. The TCP module of host A has received a contiguous block of 0 actual data bytes (Note what happens right after 3-way handshake)
    - ☐ h. The TCP module of host A has received all the data bytes from SN 0 through SN 1999
    - ☐ i. None of the former options is true

16. [3] Figure 1 represents a 3-way handshake initiated by host A with host B. Tick all the statements that are true:

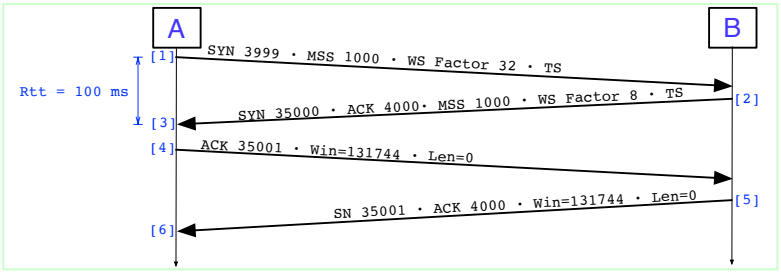


Fig. 1. Note: **Win** is the effective Advertised Window Size, like in Wireshark; it is not the TCP Window field contents

- ☐ a. The value contained in the Window field of segment [4] is 131744
- ☐ b. The value contained in the Window field of segment [4] is 16468
- ☐ c. The value contained in the Window field of segment [2] is 8
- ☐ d. Both hosts, A and B can measure the Rtt of each transmitted segment
- ☐ e. The SN that host A will use for sending the first actual data segment from its stream is 3999
- ☐ f. The SN that host A will use for sending the first segment containing actual data from its stream is 4000
- ☐ g. The Initial Sequence Number used by host B is 35001
- ☐ h. Per the TCP options set in the handshake, we can claim that host A has the Nagle algorithm activated
- ☐ i. It's host B that does the active open

17. Assume that, right after the handshake depicted in Figure 1, the transmissions depicted in Figure 2 take place. The values contained in fields TSVal and TSecr are the fixed comma integers X0 through X7 (Their concrete values are irrelevant, that is the reason why we are representing them symbolically). Respond to the related questions below:

a. [1] What's the value of X5 as a function of the values contained in TSval and Tsecr?

X0. The reason is that, according to RFC 7323, the Rtt measured by host A should include any *delack* that might have been inserted by the receiver, B. The delack-inflated Rtt is preferred to the tiny Rtt that would result if the Rtt sample were taken by using the latest unacknowledged segment instead of the earliest one. The answer is X0.

b. [1] Explain what procedure is used by host A to measure the current RttSample of 101.2 ms at time point [4]

$TimeOfDay_A - T_{secr} = TimeOfDay_A - X0$

c. [1] Just before time point [4] the current EstimatedRtt is 105.3 ms. What RTO time length is host A scheduling for protecting the retransmission of the segment transmitted at time point [5]? Assume TCP is executing the Karn-Partridge algorithm with $\alpha = 0{,}9$.

$EstimatedRtt[n+1] = 0.9 \cdot EstimatedRtt[n] + 0.2 \cdot RttSample[n]$
$EstimatedRtt[n+1] = 0.9 \cdot 105.3\ ms + 0.1 \cdot 101.2\ ms$
$EstimatedRtt[n+1] = 104.89\ ms;$

$RTO[n+1] = 2 \cdot 104.68\ ms = 209.78\ ms$

		alpha	0,9
Initilal SRTT	Sample	SRTT formula	RTO
105,3	101,2	104,89	**209,78**

d. [1] In which of the time points [1] or [2] will host A *start* the Retransmission Timer?

The RTO will be started at time point [1], the time at which the first data packet is sent

e. [1] In which of the time points [4] or [5] will host A *restart* the Retransmission Timer?

The RTO will be started at time point [4] since the received ACK does advance **snd.una**

f. [1] Assume that the segment transmitted at time point [5] is lost and that host A transmits another three segments containing data, afterwards. Demonstrate whether the retransmission of the segment sent at [5] happens as the result of a 3-DUP or of an RTO.

The number of segments transmitted after segment [5], 3 segments, cause a 3-DUP for ACK SN = 22000

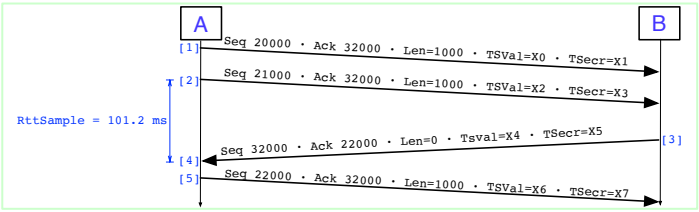


Fig. 2

18. The segment transmissions depicted in Figure no. 3 are taking place after a normal 3-way handshake that happened earlier and which MSS option is set to 1000 (This 3-way handshake is not the same as that in Figure 1). Respond to the following questions related to the foregoing scenario:

a. [1] Can the Nagle's algorithm be activated at host A's connection side (Socket)? Explain your answer.

Nagle's algorithm cannot have been activated at A's side since host A is transmitting a partial segment (The last one which size is 10 whilst the MSS is 1000) at a time when no ACK to the transmitted segments has been received

b. [1] Explain how many of the segments in Figure 3 don't have the flag ACK activated.

Since all of the segments contain data and are transmitted ever after the 3-way handshake, all must have the flag ACK set.

c. [1] What ACK Sequence number would host B have to include upon sending one segment that ACKed the block of data contained in segments [6] and [7].

ACK SN = 11010

d. [1] Is the situation explained in the preceding question acceptable by TCP, namely that the TCP is acking a two-segment of in-order data block by sending only one cumulative ACK? Justify your answer.

Yes, this situation corresponds to the Delayed Ack TCP concept.

e. [1] How many of the segments in Figure 3 have the SYN flag set?

Since the 3-way handshake took place before the data transfers depicted in Figure 3, no further segment can have the SYN flag set.

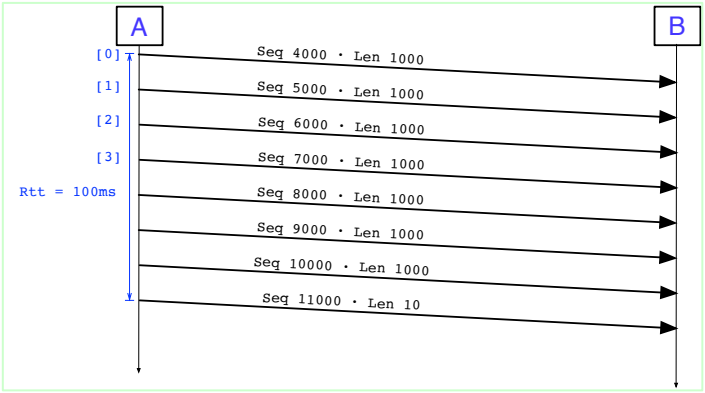


Fig. 3

