Sample Question: Headers, Headers and Headers (please discuss)

Consider a 4-layer protocol stack with the following layers (numbered 1 to 4 from bottom to top), physical, network, transport and application. The physical layer uses one-byte flags to mark both the start and end of a packet (i.e. a one-byte header and a one-byte trailer). The other three layers all use four-byte headers. The transport layer imposes that the maximum packet payload (i.e. data portion) for this layer must not exceed 100 bytes. Other layers have no such restrictions. Calculate the percentage overhead assuming an application message of 450 bytes is to be transmitted.

Resource created about a month ago (Monday 16 July 2018, 02:50:37 PM), last modified 26 days ago (Friday 27 July 2018, 11:18:57 AM).

Comments

(/COMP3331/18s2/forums/search?forum_choice=resource/17224) (/COMP3331/18s2/forums/resource/17224)

Add a comment

Michael Yoo (/users/z5165635) 16 days ago (Mon Aug 06 2018 13:33:49 GMT+1000 (Australian Eastern Standard Time)), last modified 16 days ago (Mon Aug 06 2018 13:34:02 GMT+1000 (Australian Eastern Standard Time))

4 x (1 + (4 + (4 + (4 + (100)))) + 1) + 1 x (1 + (4 + (4 + (4 + (50)))) + 1) = 520

Therefore, (520 - 450)/450 = 0.1555 = approx. 15.6% overhead.

Reply

Nadeem Ahmed (/users/z1058484) 16 days ago (Mon Aug 06 2018 13:52:36 GMT+1000 (Australian Eastern Standard Time))

Try again.

Reply

Michael Yoo (/users/z5165635) 16 days ago (Mon Aug 06 2018 14:44:27 GMT+1000 (Australian Eastern Standard Time))

Maximum application layer packet size = 100 (imposed by transport layer)

Maximum application payload size = 100 - 4 (application layer header) = 96

Overhead per payload = 2 (physical) + 4 (network) + 4 (transport) + 4 (application) = 14 bytes

Number of packets needed = ceil(450 / 96) = 5 packets.

Therefore (14*5) / 450 = 0.1555 = approx. 15.6% overhead.

Reply

Nadeem Ahmed (/users/z1058484) 16 days ago (Mon Aug 06 2018 14:51:39 GMT+1000 (Australian Eastern Standard Time))

Hint: Segments are formed at the transport layer.

Reply

Michael Yoo (/users/z5165635) 16 days ago (Mon Aug 06 2018 16:47:12 GMT+1000 (Australian Eastern Standard Time))

Application Layer:

- 4 (header) + 450 (payload); 454 (total)

Transport Layer: 454 bytes segmented into 5 segments.

- 4 (header) + 100 (payload); 104 (total)
- 4 (header) + 100 (payload); 104 (total)
- 4 (header) + 100 (payload); 104 (total)
- 4 (header) + 100 (payload); 104 (total)
- 4(header) + 54 (payload); 58 (total)

Network Layer:

- 4 (header) + 104 (payload); 108 (total)
- 4 (header) + 104 (payload); 108 (total)
- 4 (header) + 104 (payload); 108 (total)
- 4 (header) + 104 (payload); 108 (total)
- 4 (header) + 58 (payload); 62 (total)

Physical Layer:

- 2 (header) + 108 (payload); 110 (total)
- 2 (header) + 108 (payload); 110 (total)

- 2 (header) + 108 (payload); 110 (total)
- 2 (header) + 108 (payload); 110 (total)
- 2 (header) + 62 (payload); 64 (total)

Total amount of bytes sent: 504

Original payload: 450

Protocol overhead: (504 - 450) / 450 = 54 / 450 = 0.12 = 12%.

Reply

Nadeem Ahmed (/users/z1058484) 16 days ago (Mon Aug 06 2018 17:17:42 GMT+1000 (Australian Eastern Standard Time))

Bingo!!!. Well done

Reply

Michael Yoo (/users/z5165635) 16 days ago (Mon Aug 06 2018 16:37:56 GMT+1000 (Australian Eastern Standard Time)), last modified 16 days ago (Mon Aug 06 2018 16:38:27 GMT+1000 (Australian Eastern Standard Time))

> The transport layer imposes that the maximum packet payload (i.e. data portion) for this layer must not exceed 100 bytes.

Doesn't this mean that the transport layer won't accept packets that would exceed the maximum segmentation size, so no segments are formed?

Reply

Md. Imam Mannaf (/users/z5217295) 22 days ago (Tue Jul 31 2018 21:49:17 GMT+1000 (Australian Eastern Standard Time))

For each payload packet of 100 bytes there is 3x4+1=13 overhead bytes.

So the percentage is 13x100/113=11.5

Reply

Nadeem Ahmed (/users/z1058484) 16 days ago (Mon Aug 06 2018 11:37:25 GMT+1000 (Australian Eastern Standard Time))

Read the statement of the question carefully and give it another try.

Reply