

Internet Engineering

241-461

Robert Elz

kre@munnari.OZ.AU

kre@coe.psu.ac.th

<http://fivedots.coe.psu.ac.th/~kre>

Contents

- ◇ Why Fragmentation?
- ◇ Why Reassembly?
- ◇ When to Fragment
- ◇ When to Reassemble
- ◇ How to Fragment
- ◇ How to Reassemble (cont)
- ◇ Why not fragment
- ◇ How to avoid fragmentation (PMTUD)

Reassembly Issues

- ◇ What happens if a fragment is lost ??
 - That is, never arrives at destination
- ◇ Does reassembly queue wait forever ?
 - No - we hope anyway!
- ◇ What prevents waiting forever ?

TTL and Reassembly

- ◇ Time To Live
 - Exists in every IP packet header
 - Including in headers of fragments
 - ▷ which are IP packets anyway!
 - Including fragments on reassembly queue
- ◇ TTL is upper limit (in seconds)
 - for how long packet can exist
- ◇ Every second
 - ▷ Or more often sometimes
 - TTL must be decremented
 - ▷ decreased by one
- ◇ When TTL reaches 0
 - packet must be discarded
 - Even if at destination in reassembly queue

TTL and Reassembly (2)

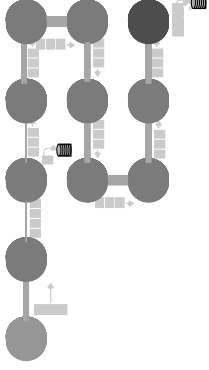
- ◇ TTL cleans up reassembly queues
 - eventually the TTLs reach 0
 - fragments on queue discarded
- ◇ Also note TTL procedure
 - ▷ when packets merged
 - Take minimum of TTL values from fragments
 - TTL must NEVER increase
 - ▷ not in any packet or fragment
- ◇ Be aware of the effect of lost fragments
 - Entire packet ends up lost
 - ▷ Fragments not lost discarded at destination
 - ▷ Because the reassembly queue never merges
 - Back into the original packet
 - ▷ The missing piece leaves a hole
- ◇ So...

Contents

- ◇ Why Fragmentation?
- ◇ Why Reassembly?
- ◇ When to Fragment
- ◇ When to Reassemble
- ◇ How to Fragment
- ◇ How to Reassemble
- ◇ Why not fragment
- ◇ How to avoid fragmentation (PMTUD)

Why not fragment?

- ◇ One lost fragment
 - means whole packet lost
- ◇ But fragments not lost in network
 - still delivered to destination
 - even though useless
 - waste bandwidth



Why not fragment (2)

- ◇ Usual cause of packet loss
 - Congestion
 - ▷ Very busy link
 - ▷ Insufficient capacity for demand
- ◇ Congestion occurs
 - data must be discarded
 - happens to be a fragment
 - ▷ other fragments still transmitted
 - ▷ over congested link
 - more data must be discarded
 - ▷ probably some other packet
- ◇ Discard good data
 - Transmit useless data !!

Fragmentation should be avoided

whenever possible

Contents

- ◇ Why Fragmentation?
- ◇ Why Reassembly?
- ◇ When to Fragment
- ◇ When to Reassemble
- ◇ How to Fragment
- ◇ How to Reassemble
- ◇ Why not fragment
- ◇ How to avoid fragmentation (PMTUD)
 - Send packets small enough
 - What is Small enough?
 - Path MTU Discovery

Avoiding Fragmentation

- ◇ If host sends small packets
 - fragmentation will not be needed
- ◇ If packet is lost
 - host can retransmit just that packet
- ◇ Other packets that reach destination
 - can be retained and used
 - ▷ Transport Protocol issues (TCP)
- ◇ How small is small enough?
 - 68 bytes certainly
 - ▷ Packets 68 bytes or less cannot be fragmented
 - ▷ Network must be able to send 68 byte packets
 - on every link used for IP
 - Using 68 byte packets
 - ▷ 20 byte IP header (minimum)
 - ▷ 20 byte TCP header (minimum)
 - 40 bytes of headers
 - ▷ 28 bytes remain for data
 - Almost 60% overhead - best case

Avoiding Fragmentation (2)

- ◇ 68 byte packets avoid fragmentation
 - 100% certainly
 - But too much overhead
- ◇ Can we send bigger packets
 - And still avoid fragmentation?
- ◇ Perhaps
 - It depends upon the network
- ◇ Usually assume 576 byte packets are OK
 - Certainly not guaranteed
 - Some slow links fragment around 200 bytes
 - ▷ so packets transmit quickly
- ◇ Almost always OK
 - Because of common mistake
 - Often believed that 576
 - ▷ is minimum required MTU
 - 576 is minimum packet reassembly size