

EN 2021

① Os headers do envelope asseguram as funcionalidades de transporte desde o servidor do sender até ao servidor do receiver, por outro lado, os headers da mensagem contém informação que vai ser vista pelo utilizador, como quem enviou a mensagem, os CCs e BCCs associados, etc., ou seja, asseguram organização e funcionalidades de exibição

② O servidor deve ter, em 1º lugar, um socket, num porto conhecido pelos clientes, que fica à espera de novas conexões, quando esse socket deteta uma conexão de um novo cliente, será criada outro socket, exclusivamente para o cliente. Para gerir as múltiplas conexões e I/O de múltiplos sockets, podem ser usados métodos de multithreading, multiprocessing ou multiplexação de I/O.

⊛ (num porto diferente)

③ Caso o tamanho da janela seja maior que o espaço de números de sequência, alguns dados repetidos, enviados pelo sender, podem ser interpretados como dados novos, por exemplo, se tivermos o espaço de seq# de tamanho 4 e a janela for de tamanho 3, pode, por exemplo, dar-se o seguinte caso:

1- Sender manda packets #1, #2 e #3, com números de sequência 0, 1 e 2, respectivamente.

2- Receiver manda ACK para cases 3, movendo a sua janela para os packets #4, #5 e #6, com seq# 3, 0 e 1, respectivamente.

3- ACKs são perdidos e o sender re-transmite os mesmos pacotes iniciais (que tinham seq# 0, 1 e 2)

4- Apesar de serem dados repetidos, o rec. vai interpretá-los como novos, uma vez que a sua janela avançar mais do que era suposto.

Tal não aconteceria se o tamanho da janela fosse \leq espaço seq#.

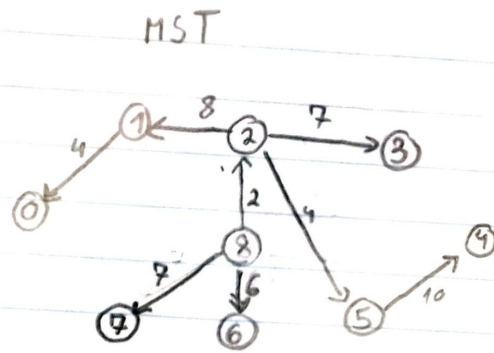
④

a) Pode ser usado um trunk link, ligado a um port especial de cada switch que recebe todas as mensagens de todos os host da rede, sendo que no header de cada packet, está incluída uma tag que identifica a VLAN a que cada packet pertence.

b) Sim, é possível, uma solução simples seria adicionar um router, conectando uma interface a um dos ports do trunk-link. Essa interface não é, também, configurada em subinterfaces, cada uma com IP diferente e responsável pela sua VLAN, este método é conhecido como ROAS (Router On A Stick)

⑤ $T = \{8, 2, 5, 6, 7, 3, 1, 0, 4\}$
 $d = \begin{bmatrix} 0 & 2 & 0 & 2 & 2 & 5 & 0 & 2 & 8 \\ 15 & 10 & 2 & 9 & 16 & 6 & 6 & 7 & 0 \\ 14 & & & & & & & & \end{bmatrix}$

$\langle 8, 2 \rangle$
 $\langle 2, 5 \rangle$
 $\langle 8, 6 \rangle$
 $\langle 8, 7 \rangle$
 $\langle 2, 3 \rangle$
 $\langle 2, 1 \rangle$
 $\langle 1, 0 \rangle$
 $\langle 5, 4 \rangle$



⑥ a) A: Gama - 10.5.0.201 → 10.5.0.206
 Rede - 10.5.0.200
 Broadcast - 10.5.0.207

11010000
 1111 - 223

B: Gama - 192.168.10.209 → 192.168.10.222
 Rede - 192.168.10.208
 Broadcast - 192.168.10.223

b) PC-1: IP - 10.5.0.201
 Máscara - 129 = 255.255.255.248
 Default Gateway - 10.5.0.206

R2-e0/0: IP - 10.5.0.206

c) 193.136.200.128/26
 $128 \xrightarrow{26, 28} 100000000 \xrightarrow{128} \begin{cases} 128 \\ 144 \\ 160 \\ 176 \end{cases}$

R1-R2: Rede - 193.136.200.128
 Broadcast - 193.136.200.143
 Gama - 193.136.200.129 - 193.136.200.142

R3-R4: Rede - 193.136.200.160
 Broadcast - 193.136.200.175
 Gama - 193.136.200.161 - 193.136.200.174

⑦ a) R1-R2: $D = 20 \text{ Km}$, $v = 5 \text{ Mbps}$
 R1-R3: $D = 10 \text{ Km}$, $v = 30 \text{ Mbps}$

$L = 5000 \text{ bits}$

$V_{prop} = 2 \cdot 10^8 \text{ m/s}$

R2-R1:
 $d_{prop} = \frac{D}{V_{prop}} = \frac{2 \cdot 10^4}{2 \cdot 10^8} = 10^{-4} \text{ s}$

R1-R3
 $d_{prop} = \frac{10^4}{2 \cdot 10^8} = \frac{1}{2} \cdot 10^{-4} \text{ s}$

$d_{trans} = \frac{L}{v} = \frac{5000}{5 \cdot 10^6} = 10^{-3} \text{ s}$

$d_{trans} = \frac{5000}{3 \cdot 10^7} = \frac{5}{3} \cdot 10^{-4} \text{ s}$

Queueing delay
 no R1

R: $d_{total} = 10^{-4} + 10^{-3} + \frac{5}{3} \cdot 10^{-4} + \frac{1}{2} \cdot 10^{-4} + 10^{-4}$

6) $d_{trans} (R2-R1) + d_{trans} (R1-R3)$ $\frac{10^6}{3} \cdot \frac{1}{3} = 10^5$

$\frac{5 \cdot 10^6}{5 \cdot 10^6} = 1s$ $\frac{5 \cdot 10^6}{3 \cdot 10^7} = \frac{5 \cdot 10^{-1}}{3}$

8) access-list 1 permit 192.168.10.208 0.0.0.15
 ip nat inside source list 1 interface Ethernet0/0 overload
 ip nat inside source list 1 interface Ethernet1/0 overload
 interface Ethernet1/1
 ip nat inside
 exit
 interface Ethernet0/0
 ip nat outside
 exit
 interface Ethernet1/0
 ip nat outside