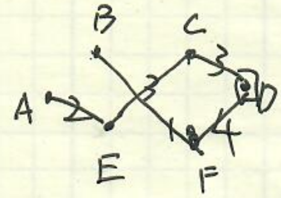
CS535 hw5 | Jiankun Dong

4 Using Dijkstra's algorithm, we get:



D: { DC 3, DF 4, DFB 5, DCE 6, DCEA 8 }

∴ D's new table:

To	Cost	Line
A	8	C
B	5	F
C	3	C
D	0	-
E	6	C
F	4	F

8. Routing vs forwarding vs. Switching

Switching uses the MAC address to determine what devices to send data to, it collects data from one port to send to other ports. (Datalink layer)

Forwarding simply send data ~~from~~ base on the forwarding table. (Networking layer)

Routing is similar to forwarding, but use the IP address instead of a forwarding table (Network layer as well).

10. ∵ 3 layer, use number around 16 to minimize size of routing table.

① Try 16 clusters ~~regions~~: $\frac{4800}{16} = 300$ ∵ we ~~also~~ want to use small routing table.

Try 15 regions: $\frac{300}{15} = 20$ ∴ going to use 15 as # routers each region. 20 as # clusters.

∴ $\boxed{\text{20 clusters, 16 regions, 15 routers each}}$

② $4800 = 2^6 \times 5^2 \times 3 = 25 \times 2^4 \times (3 \times 2^2) = 25 \times 16 \times 12$

∴ $\boxed{\text{25 clusters 16 regions 12 routers also works}}$

28. IP Ltotal 1024B (Heder+data)
$T_{live} = 10 sec$

$\because$ IP use 16 bits to store ID

$\therefore$ # of max packet $= 2^{16} - 1 = 65535$

$\therefore$ Max speed $= \dfrac{65535 \times 1024 \times 2^3 \text{ bits}}{10 \text{ sec}} = 53.686272 \times 10^6$ bps

$\approx \boxed{53.69 \text{ Mbps}}$

35

A: First address: 198.16.0.0
Last address: 198.16.15.255 (4096 total) log4096=12, 32-12=20
Prefix: 198.16.0.0/20 (end 198.16.(0b) 0000 1111.255
Mask: 11111111.11111111.11110000.00000000 (start 198.16.(0b) 0001 0111.~~255~~

B: First address: 198.16.16.0
Last address: 198.16.23.255 (2048, log 2048=11, 32-11=21)
Prefix: 198.16.16.0/21
Mask: 11111111.11111111.11111000.00000000

C: First address: 198.16.~~24~~ 32.0 Because B 198.16.(0b) 0001 0111.255
Last address: 198.16.~~39~~ 47.255 we need 12 bits for 4000 host
Prefix: 198.16.~~24~~ 32.0/20 $\therefore$ C start 198.16.(0b) 0010 0000.0
Mask: 11111111.11111111.11110000.00000000

C end 198.16.(0b) 0010 1111.0

D: First address: 198.16.64.0 D need 8000 hosts
$\therefore$ D start 198.16.(0b) 010 0 0000.0

Last address: 198.16.95.255

Prefix: 198.16.64.0/19
Mask: 11111111.11111111.11100000.00000000.

37. Do not need to split up:

New Seg: 29.18.(0b) 0001 0010.0
Prefix: 29.18.60.0/22

just add it to the routing table w/ the previous 29.18.0.0/16

46 We only apply checksum to the header because:

① checksum of header takes way less time to compute

② data's checksum can be done @ different layer.

48. 16 byte address = 128 bits

$\therefore$ total addresses $= 2^{128} \approx 3.4 \times 10^{38}$

$\therefore$ Time $= \frac{3.4 \times 10^{38}}{10^6} \times 10^{-12}$ sec $= \boxed{3.4 \times 10^{20} \text{ sec}}$

(1 year $\approx \pi \times 10^7$ sec, so about 1.1×10^{13} years!)

Day 1: For 64 bits used in IPv6,

~~Time = Time in Problem 48 / 64~~

Total address $= 2^{64}$

$\therefore \frac{\text{Time } 64}{\text{Time } 128} = \frac{1}{2^{64}} \approx \frac{1}{1.8 \times 10^{19}} \approx 5.4 \times 10^{-20}$

$\therefore$ Time 64 $\approx$ Time128 $\times 5.4 \times 10^{-20} = 3.4 \times 5.4$ sec

$= \boxed{18.36 \text{ sec}}$ (!) way faster than I thought...

Day 2: 1) It feels like ~~stws~~ switching & bridges from the datalink layer...

2) I think we can view the problem as a case of building a ethernet network and finding out which 'host' goes to which 'port'.