



Tentamen  
Datakommunikation I (DVGB02)  
10 april 2021, kl. 14:15 – 20:00

Karl-Johan Grinnemo

- 
1. Förklara skillnaden mellan OSI:s och TCP/IP:s referensmodeller för datakommunikationssystem? (4p)

**Lösningsförslag:**

Av svaret bör framgå att OSI:s referensmodell har 7 lager medans TCP/IP:s referensmodell har endast fyra (fem enl. lärobok), och att TCP/IP:s referensmodell ej har sessions- och presentationsskikt. I övrigt kan man nämna att OSI:s länklager egentligen är ett nätverksaccesslager som snarare definierar gränssnittet mot länklager/fysiskt lager, samt att TCP/IP:s referensmodell, till skillnad mot OSI:s, är en efterkonstruktion och inte standardiserad.

2. Förklara sambandet mellan begreppen *protokoll* (protocol), *tjänster* (services) och *primitiver* (primitives) (3p)

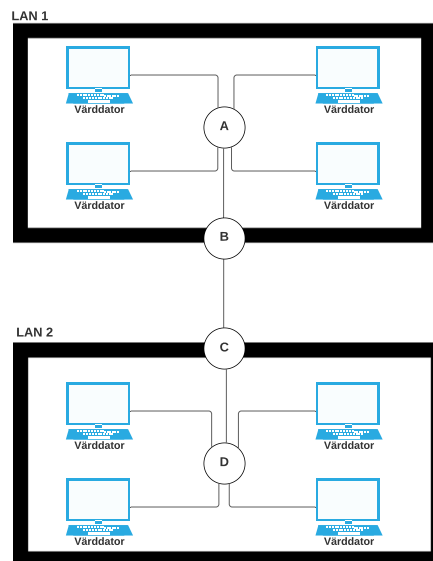
**Lösningsförslag:**

En tjänst är i princip ett antal primitiver (operationer) som ett underliggande lager erbjuder ett överliggande lager, men som inte säger något om hur dessa primitiver är implementerade. En tjänst kan implementeras av ett eller flera protokoll som beskriver ett eller flera regelverk (samling regler) som måste följas mellan motsvarande lager på sänd- respektive mottagarsidan, samt formaten på de paket som utväxlas.

3. Förklara skillnaden mellan ett *kretskopplat* (circuit switched) och ett *paketförmedlande* (packet switched) nät. Exemplifiera skillnaden med taltrafik (3p)

**Lösningsförslag:**

I ett kretskopplat nätverk skapas en fysisk förbindelse från mottagaren till sändaren. Det innebär att det finns en transmissionskanal från mottagaren till sändaren. Fast telefoni (POTS) var kretskopplat; man upprättade en förbindelse, överförde taltrafiken – utan fördröjning i nätet, och slutligen kopplade ned förbindelsen. I paketförmedlande nät så skapas ingen fysisk förbindelse mellan mottagare till sändare; nätverkets bandbredd blir endast upptagen när datapaket sänds. Om taltrafik sänds över ett paketförmedlande nät så kommer den uppleva fördröjning eftersom paketen tillfälligt lagras i alla routrar (store and forward). Vidare kan ordningsföljden mellan paket ej garanteras i ett paketförmedlande nät, så troligen kommer vissa paket i taltrafiken behöva kastas p.g.a. att de anländer mottagaren i oordning.



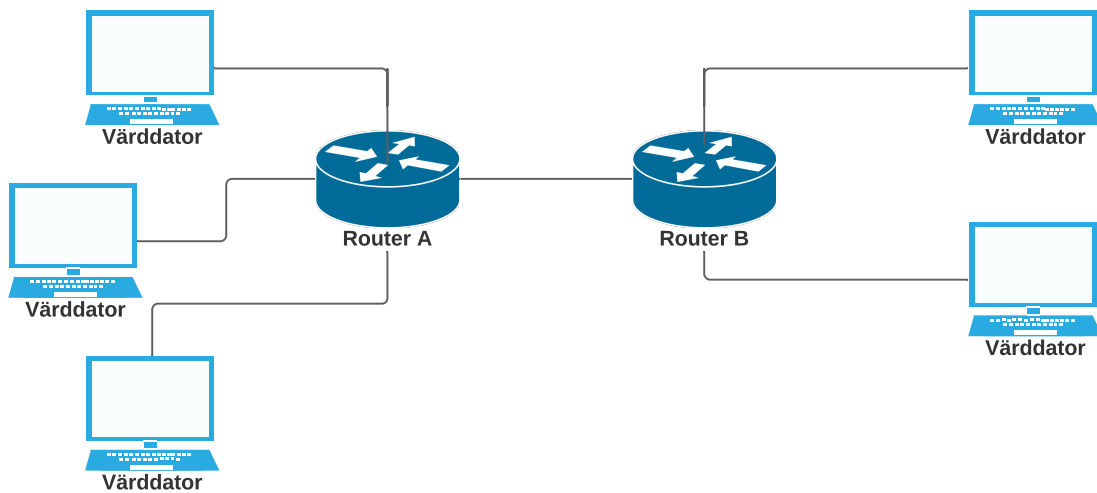
4. I figuren ovan visas ett antal värddatorer (hosts) som förbinds av ett antal länkar och noder. Ange för respektive nod vilken typ av nätelement – *router*, *switch* eller *hub* – som bör kopplas in på respektive nods plats i det visade nätet. Motivera dina val. (4p)

**Lösningsförslag:**

Noderna A och D bör vara switchar eftersom de sammankopplar värddatorer i samma LAN, med samma nätidentitet (network ID). Däremot, noderna B och C bör vara routrar eftersom de kopplar samman två olika LAN, med olika nätidentitet. Hubar bör om möjligt undvikas eftersom de inte begränsar kollisionssdomänerna till enskilda länkar utan låter hela LAN:et vara en och samma kollisionssdomän.

Subnät A: 129.16.0.0/26

Subnät B: 129.16.0.64/26



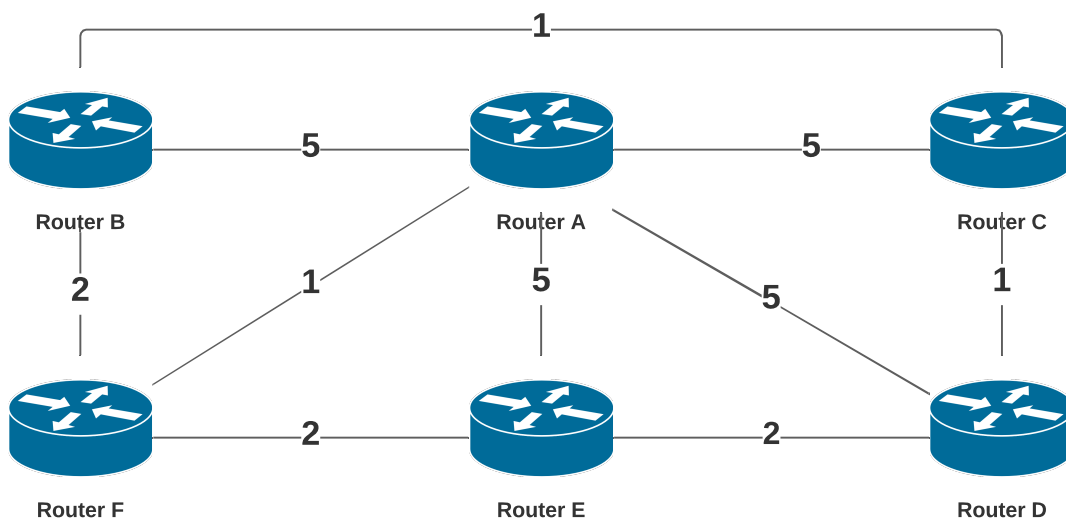
5. Betrakta nätverket i figuren ovan som består av två subnät sammankopplade via routrarna A och B.
- Som visas i figuren har enheter i subnät A tilldelats CIDR-adresserna 129.16.0.0/26 och subnät B CIDR-adresserna 129.16.0.64/26. Hur många enheter kan respektive subnät innehålla? (2p)
  - Antag att man vid ett senare tillfälle vill dela upp subnät B i två subnät. Det ena subnätet ska kunna innehålla 30 enheter och det andra subnätet 10 enheter. Hur skulle detta kunna genomföras utan att slösa med IP-adresser. (2p)
  - Ange innehållet i Router A:s och Router B:s forwarding-tabeller så att IP-paket kan skickas mellan godtyckliga värddatorer i nätverket. (6p)

### Lösningsförslag:

- Ett /26-nät kan innehålla  $2^{32-26} - 2 = 62$  enheter.
- Det ena subnätet kan vara 129.16.0.64/27 (30 enheter) och det andra subnätet 129.16.0.96/28 (14 enheter).
- Forwarding-tabeller i Router A och Router B.

| Destination | Mask            | Nästa Hopp |
|-------------|-----------------|------------|
| 129.16.0.0  | 255.255.255.192 | Direkt     |
| 129.16.0.64 | 255.255.255.192 | Router B   |

| Destination | Mask            | Nästa Hopp |
|-------------|-----------------|------------|
| 129.16.0.0  | 255.255.255.192 | Router A   |
| 129.16.0.64 | 255.255.255.192 | Direkt     |

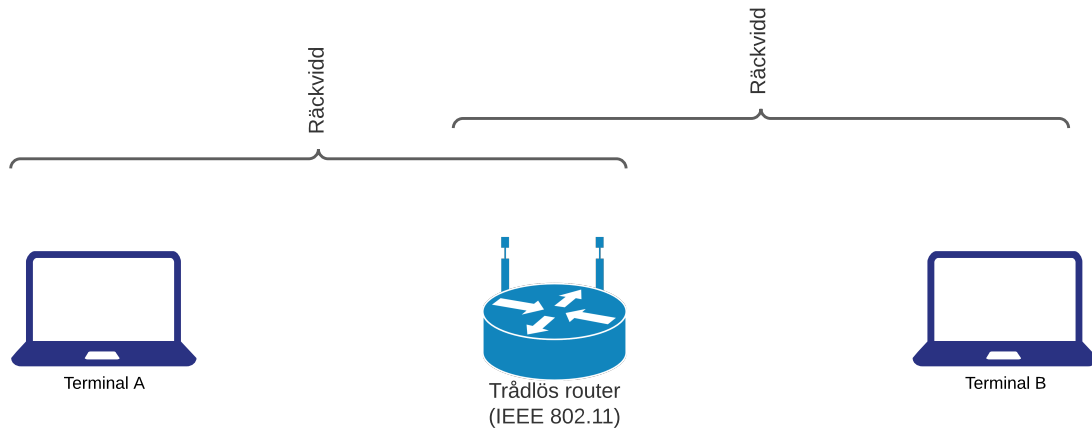


6. Visa genom att fylla i nedanstående tabell hur Dijkstras algorithm beräknar den kortaste vägen mellan router A och övriga routrar i ovan nät. Länkkostnaderna anges på respektive länk. (3p)

| Iteration | List | $D_{AB}$ | Path | $D_{AC}$ | Path | $D_{AD}$ | Path | $D_{AE}$ | Path | $D_{AF}$ | Path |
|-----------|------|----------|------|----------|------|----------|------|----------|------|----------|------|
| 1         | {A}  | 5        | A-B  | 5        | A-C  | 5        | A-D  | 5        | A-E  | 1        | A-F  |
| 2         |      |          |      |          |      |          |      |          |      |          |      |
| 3         |      |          |      |          |      |          |      |          |      |          |      |
| 4         |      |          |      |          |      |          |      |          |      |          |      |
| 5         |      |          |      |          |      |          |      |          |      |          |      |
| 6         |      |          |      |          |      |          |      |          |      |          |      |

### Lösningsförslag:

| Iteration | List     | $D_{AB}$ | Path  | $D_{AC}$ | Path    | $D_{AD}$ | Path | $D_{AE}$ | Path  | $D_{AF}$ | Path |
|-----------|----------|----------|-------|----------|---------|----------|------|----------|-------|----------|------|
| 1         | {A}      | 5        | A-B   | 5        | A-C     | 5        | A-D  | 5        | A-E   | 1        | A-F  |
| 2         | {AF}     | 3        | A-F-B | 5        | A-C     | 5        | A-D  | 3        | A-F-E |          |      |
| 3         | {AFB}    |          |       | 4        | A-F-B-C | 5        | A-D  | 3        | A-F-E |          |      |
| 4         | {AFBE}   |          |       | 4        | A-F-B-C | 5        | A-D  |          |       |          |      |
| 5         | {AFBEC}  |          |       |          |         | 5        | A-D  |          |       |          |      |
| 6         | {AFBECD} |          |       |          |         |          |      |          |       |          |      |



7. Betrakta ovan WLAN-nät (IEEE 802.11). Två trådlösa terminaler A och B är uppkopplade mot det trådlösa nätet i *infrastructure mode*. De befinner sig inte inom varandras räckvidd.
- När någon av terminalerna A och B vill initiera en uppkoppling så skickar den aktuella terminalen ett RTS-meddelande till den trådlösa routern som besvarar detta meddelande med ett CTS-meddelande. Vad är syftet med dessa två meddelanden? (2p)
  - Förhindrar utbytet av RTS/CTS att meddelanden som skickas mellan terminal A och B aldrig kolliderar? Motivera. (2p)
  - Antag att de trådlösa länkarna mellan terminalerna och den trådlösa routern är brusiga. Vad gör det i ett sådant läge problematiskt att sätta upp en TCP-session (connection) mellan terminal A och B? (2p)
  - Antag att den trådlösa routern har en kapacitet på 100 Mbps. Gör ett överslag på den maximalt möjliga genomströmningen mellan terminalerna A och B. Var noga med att redovisa uträkning och eventuella antaganden. (2p)

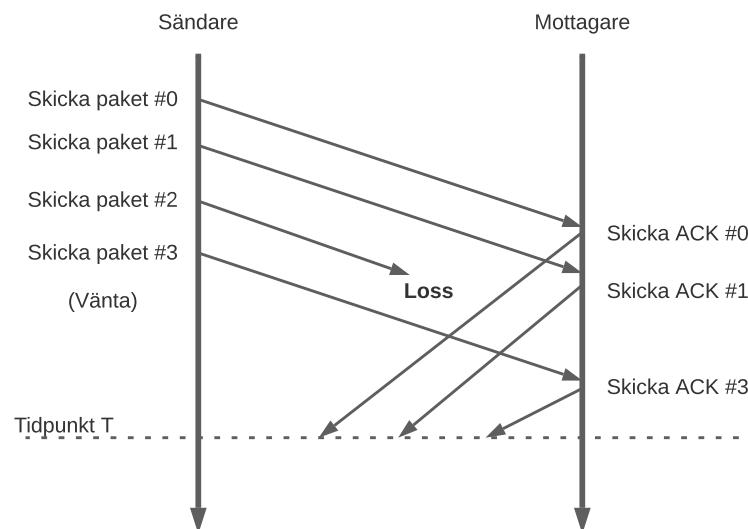
### Lösningsförslag:

- Med meddelandet RTS anger terminal A eller B för den trådlösa routern, d.v.s. accesspunkten, att den behöver en kanal att sända på. Den trådlösa routern besvarar ett RTS-meddelande från en av terminalerna med ett CTS-meddelande som anger att den reserverar kanalen till den aktuella terminalen, och som därmed förhindrar andra terminaler som befinner sig inom den trådlösa routerns räckvidd från att sända.
- Nej, om CTS-meddelandet från den trådlösa routern, på grund av t.ex. brus, inte når någon eller några av de terminaler som vill sända samtidigt så kan det resultera i kollisioner.

- c Om de trådlösa länkarna är brusiga ökar det sannolikheten för paketförluster på länklagret och fördröjningar på transportlagret, något som felaktigt kan tolkas som stockning av TCP.
- d Vi noterar att varje TCP-segment måste genomföra två hopp: ett första hopp från terminal A till den trådlösa routern och ett andra hopp från den trådlösa routern till terminal B. Även varje ACK-segment måste göra två hopp, fast i motsatt riktning. Vidare noterar vi att när något sänds över det första hoppet, kan inget sändas över det andra hoppet. Detta gör att den maximala genomströmningen mellan terminalerna är 100 Mbps dividerat med 4, d.v.s. 25 Mbps.
8. Ange två anledningar till varför *link-state routing* inte används över hela Internet? (2p)

### Lösningsförslag:

En anledning är att link-state routing använder sig av *flooding* vilket leder till att ett stort antal meddelande genereras och därmed ej är skalbart. En annan anledning är att Dijkstra:s algorithm som utgör grunden för link-state routing inte är skalbar (har en tidskomplexitet på  $O(N^2)$ , där  $N$  är antalet nätnoder, t.ex. routrar). En tredje anledning är att link-state routing inte gör det möjligt för olika ISP:er (autonoma system) att implementera sina egna policies.



9. Två ARQ-mekanismer är *go-back-N* och *selective repeat*. I figuren ovan visas tid-sekvensdiagrammet för en av dessa ARQ-mekanismer. Som antyds i figuren är sändfönstret 4 paket. Paketens sekvensnummer löper mellan 0 och 15.
- a Vilken ARQ-mekanism används i tid-sekvensdiagrammet i ovan figur? Motivera. (2p)
- b Vilka paket ingår i sänd- respektive mottagarfönstret vid tidpunkten  $T$  (4p)
- c Är det möjligt att öka storleken på sändfönstret till 10 paket? Motivera. (2p)

**Lösningsförslag:**

- a Det är selective repeat eftersom paket #3 kvitteras trots att paket #2 förlorats.
  - b I sändfönstret ingår paketen med sekvensnumren  $\{0,1,2,3\}$  och i mottagarfönstret ingår paketen med sekvensnumren  $\{2,3,4,5\}$ .
  - c Nej, med ett sändfönster större än 8 paket så kan sekvensnumren i sänd- och mottagarfönstren överlappa. T.ex. antag att sänd- och mottagarfönstren är på 9 paket, och följande scenario utspelar sig:
    1. Sändaren har ett sändfönster,  $W_{sänd} = 0, 1, 2, 3, 4, 5, 6, 7, 8$ , och skickar iväg samtliga paket i sändfönstret;
    2. Alla skickade paket tas emot av mottagaren, som därefter har ett mottagarfönster:  $W_{mott} = 9, 10, 11, 12, 13, 14, 15, 0$ ;
    3. Samtliga kvittenser från mottagaren förloras;
    4. Sändaren får en *timeout* på paket #0, som sänds om.
    5. Mottagaren tar emot paket #0 som ett nytt paket i stället för en omsändning. Ett fel uppstår.
10. Vad är syftet med HTTP:s "kakor" (cookies)? (2p)

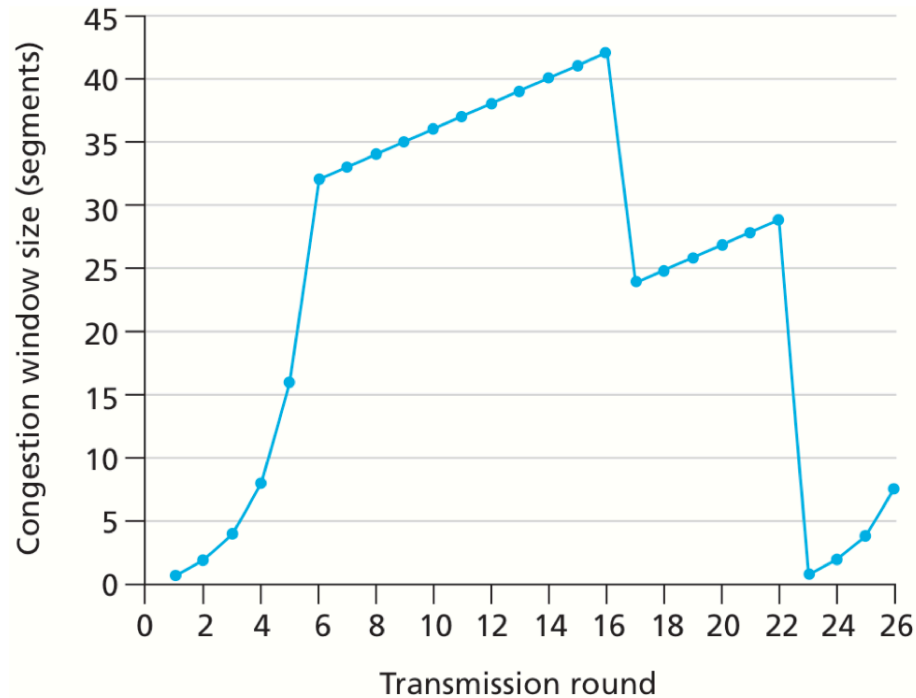
**Lösningsförslag:**

HTTP använder sig av "kakor" för att lagra tillståndsinformation och därmed kunna kundanpassa webbplatser.

11. Hur skiljer sig socketprogrammering i TCP från UDP (4p)

**Lösningsförslag:**

Av svaret bör framgå att en TCP-förbindelse identifieras av en fyrtupel,  $\langle \text{sändarens IP-adress, sändarens portnummer, mottagarens IP-adress, mottagarens portnummer} \rangle$ , medans UDP är ett förbindelseöst protokoll där flera klient-applikationer kan skicka paket till samma port på serversidan. Det bör också framgå att på serversidan av en TCP-applikation så särskiljer man mellan två typer av sockets: dels en server-socket som lyssnar efter uppkopplingsbegäran från TCP-klienter, och dels en förbindelse-socket (connection socket) som skapas för varje uppsatt förbindelse.



12. Figuren ovan visar hur stockningsfönstret (congestion window) på sändarsidan hos ett TCP-flöde (TCP Reno) utvecklas över tid. Besvara nedanstående frågor med utgångspunkt från denna figur.

- Identifiera under vilka transmissionsrundor (transmission rounds) som sändarsidan befinner sig i *slow start* respektive *congestion avoidance* (2p)
- Som framgår av figuren upplever sändaren två segmentförluster: en under den 16:e transmissionsrundan och en under den 23:a transmissionsrundan. På vilket sätt detekteras respektive segmentförlust av sändaren? Skedde det via en *timeout* eller tre duplicerade kvittenser (duplicate ACKs) (2p)
- Vilket värde har *ssthresh* under den 18:e transmissionsrundan? (1p)
- Hur många segment har skickats av sändarsidan och kvitterats av mottagarsidan vid inledningen av 10:e transmissionsrundan om vi antar att sändfönstrets storlek begränsas av storleken på stockningsfönstret? (2p)
- Hur många segment har skickats av sändarsidan och kvitterats av mottagarsidan vid inledningen av 10:e transmissionsrundan om vi inte kan anta att sändfönstrets storlek begränsas av storleken på stockningsfönstret? Motivera. (2p)



**Lösningsförslag:**

- a Slow start:  $[1,6]$ ,  $[23,26]$ ; Congestion avoidance:  $[6,16]$ ,  $[17,22]$ .
- b Segmentförlusten under 16:e transmissionsrundan detekteras med hjälp av tre duplicerade ACK:ar, medans segmentförlusten under den 23:e transmissionsrundan detekteras via en timeout.
- c *ssthresh* har värdet 21 ( $42/2$ ) under den 18:e transmissionsrundan.
- d Vi beräknar "arean" under grafen mellan  $[1,10[$ :  $1+2+4+8+16+32+33+34+35 = 165$  segment. Notera att segmenten som sänds ut under den 10:e rundan ej ska räknas med.
- e Då är frågan olöslig eftersom  $W_{sändfönster} = \min\{cwnd, rwnd\}$ .

Slut på tentamen