1. Internet 6p

1a) Ponera att en användare laddar upp en stor fil till en fjärr webbserver genom HTTP-POST metoden. Användaren har en höghastighets Internet-koppling för sin värddator.

Se avsnitt 1.4.4 i kursboken och Wireshark-labb (TCP)

FÖRKLARING:

Vid datakommunikation tar det inte bara tid för att föra användardata (innehållet i filen) över den länk som används för egen nätverksanslutning. Dessutom skickas det inte enbart användardata utan även kontrolldatabitar, dvs. överhead som läggs till som headerfält av kommunikationsprotokollen. Med dessa anledningar utgör transmissions-tiden (L/R = datalängd/bithastighet) över länken bara en bråkdel av den verkliga, totala tiden som spenderas för en lyckad överföring av avändardata.

Till skillnad från den ”fysiska” bithastigheten på en länk, **R** = bit **R**ate (bps) är throughput **Th** ett dynamiskt mått på hur effektivt och snabbt är det att användardata (ex. från en hel fil) kan hämtas på en hel session över Internet.

Länkkapaciteten, likaså bithastighet på länken **R** anges alltid i **bps** eftersom data skickas i form av en bitström vid överföring över nätverk (seriell kommunikation). En lagrad byte av data omvandlas vid kommunikation till s.k. oktett (8 bitar i följd, seriellt). Därför skall datalängden anges som antalet bitar (och inte byte) vid beräkningar t.ex. när man skall jämföra **Th** med **R**.

Överförd användardata i antalet bitar = 8 * filstorlek i antalet i byte

1000 bitar = kilobit = kb, 1000 kb = Megabit = Mb, osv.

Filhämtningen (file **transfer**) kräver kontrollerad **transport**, och applikation-protokollet (ex. http eller ftp) som används för överföringen litar sig på TCP för tillförlitligt transport. Datamängden i filen delas upp i ett antal datablock (segment) och varje block skickas i ett IP-paket (datagram).

Den genomsnittliga **throughput** (medel antal överförda databitar per tidsenhet) räknas genom att dividera datamängden med den totala tiden för hela TCP-anslutningen.

i. Ange en matematisk ’vetenskaplig’ definition för den s.k. ”throughput” dvs. genomströmningen, för överföringen av en stor fil över Internet. *(1p)*

Throughput är den **effektiva** (netto-) överföringshastigheten räknats i bps (*bit per second*) som kommunikationsprotokollen kan åstakomma vid kontrollerad överföring av hela datamängden från filen, på en hel **session** över Internet. Den räknas som det totala antalet av de överförda användardatabitarna **F** dividerat med den totala tiden **T** som sessionen tar från initiering tills överföringen är fullbordat.

Th = F/T bps.

ii. Beskriv metoden och verktyg för att **du** skall hjälpa användaren med att räkna ut ”throughput” för den ovannämnda överförföringen. *(1p)*

Man behöver köra Wireshark för att få paketinfångst av TCP-segmenten (innehållande data eller ACK) som skickas och tas emot av servern och datorn. Naturligtvis kommer Wireshark att startas i förväg innan man startar själva hämtningen med applikationen.

Den totala datamängden räknas utan all överhead, dvs 8 * det totala antalet bytes som transporteras i TCP-segmenten. Den totala tiden för hela TCP-anslutningen fås med hjälp av Wireshark genom att mäta tiden mellan att skicka det första **SYN-segmentet** och att få det sista **ACK-segmentet**. (Stevens-graf är ett enklare sätt att få båda dessa värden, se svaret på frågan **2b**.)

iii. Vad är de **huvudsakliga** orsakerna till att det genomsnittliga ”throughput” är mycket mindre än länk-kapaciteten på själva Internet-kopplingen? **Beskriv minst två orsaker.** *(2p)*

- **Kontrollmekanismerna** som TCP-protokollet tillämpar (handskakning, stockning, bekräftelse, omsändning, ..) kräver olika mycket tider att genomföra vid olika situationer. Dessa tider är mycket större än transmissionstiden över den lokala länken.

- Olika slag av **fördröjningar** mellan värddatorn och servern (end-to-end) påverkar **RTT** (Round Trip Time) och den totala tiden för sessionen.

- Länkarna och routrarna på vägen mellan värddatorn och servern har olika bithastigheter och belastningsgrader. En låghastighets länk eller en överbelastad router kan bli **flaskhals** för hela överföringen.

- Kommunikationsprotokollen lägger till en mängd kontrolldata som header vid olika lager dvs. header i segment, paket och ram (läggs även trailer) för att kontrollera överföringen av varje datablock. All detta **överhead** är nödvändigt och använder också en del av kapaciteten på länkarna.

1b) Beskriv de **två viktigaste** fördröjningarna ”delay” som **varierar från hopp till hopp** och som paket är utsatta för på vägen, genom Internet, mellan slutanvändarna. Förklara tydligt orsakerna till dessa fördröjningar. *(2p)*

Vägen mellan slutanvändarna genom Internet består av antal routrar sammankopplade med fysiska länkar som utgör delsträckor. De två viktigaste fördröjningarna som varierar från hopp till hopp är relaterade till de följande tiderna:

Transmissionstid: det tar tid att överföra databitarna för ett paket över en länk (antalet bitar/bithastigheten) och denna tid påverkas av begränsad bithastighet på den fysiska länken.

Kötid: inkommande paket skall behandlas av routern för att tas emot på ett interface och routas till utgående interface. Det är väntetid när paket finns i routerns kö som orsakas av hög trafikintensitet så att det är för många paket som skall skickas vidare på ett utgående interface.

2. Transportprotokollen 8p

2a) Vilket eller vilka av de följande meddelandena använder User Datagram Protocol (**UDP**)? Varför eller varför inte? *(3p)*

1. DNS-meddelande mellan klient och server
   Se avsnitt 3.3 i kursboken
   - DNS-meddelanden mellan klient och server består av en enda DNS-förfråga från klienten följt av ett enda DNS-svar från servern. Dessutom är DNS-meddelandet kompakt och kort dvs innehåller data på ett antal fördefinerade fält som vanligen inte överstiger 512 byte.

   DNS använder UDP för att transportera förfrågningar och svaren. Att använda UDP är snabbare och att det kommer att vara minimal overhead för att få svar. På så sätt kan servern hantera enkelt fler förfrågningar utan att behöva hantera onödvändiga TCP-anslutningar.

2. ICMP-meddelande om fel-rapportering
   Se avsnitt 4.4.3 i kursboken

   ICMP är **inte** applikation-protokoll (**inget** klient/server förhållande) utan det är ett stödprotokoll för, och är integrerat med Internet Protokollet IP på nätverkslagret i TCP/IP-protokollstack. **ICMP-meddelandet inkapslas direkt i IP-paket** utan att behöva **varken UDP eller annat** transportprotokoll. ICMP fel-meddelande får inte generera ett nytt fel-meddelande eller respons.

3. DHCP-meddelande om IP-konfiguration
   Se avsnitt 3.3 i kursboken
   DHCP är ett applikation-protokoll (klient/server förhållande) för bootstrap och **använder UDP** för att transportera meddelanden i IP-paket till **broadcast**-adressen som mottagare-adress. Dessutom kan TCP-anslutningen inte skapas från en klient som **saknar IP-konfiguration**.

2b) Följande är ett exempel på hur TCP-protokollet hanterar **stockningskontrollen** ”congestion control” på Internet. Vid en uppladdning av en medelstor textfil användes en webbklient med POST-metoden för att överföra filen till en fjärr webbserver, på ett mycket liknande sätt som gjordes vid genomförandet av Wireshark-labbarna. Efter infångsten av HTTP-trafiken med Wireshark och med TCP-filter fick man en serie av IP-paket innehållande alla sända och mottagna TCP-segmenten för denna filöverföringen. Man kunde få fram Stevens-graf; **visas på nästa sidan.**

Vid noggrann analys av data-segmenten och de motsvarande ACK-segmenten kan man konstatera följande:

1. Alla sända data-segmenten var på storlek: Maximum Segment Size (MSS).
2. POST-segmentet hade 487 bytes HTTP-header.
3. Överföringen av filen var **felfritt** och fullbordat med 9 sändningsomgångar efter det första SYN-segmentet.
4. Mottagaren ökade ”Receive Window Size” kontinuerligt så att sändaren aldrig behövde begränsa sändningen pga av storleken på den lediga bufferten.

Din uppgift är att avända grafen för att: *(5p)*

- beskriva TCP-sändarens beteende under den aktuella överföringen.
- redogöra för de algoritmer som tillämpades **vid varje sändningsomgång**.

- redovisa antalet sända segment **vid varje omgång** och varför.
- beräkna noggrant **medelvärde** på RTT och 'throughput'.

Se avsnitt 3.7 i kursboken och Wireshark-labb (TCP)

Överföringen var felfritt och fullbordat utan begränsning från mottagarens buffertstorlek (window size). Detta innebär **ideella** förhållanden för att **TCP-sändaren följer stockningsalgoritmer** när det gäller storleken på **cwnd** (Congestion Window).

Från grafen ser man att efter etableringen av TCP-anslutningen dvs. vid tiden 0,12 s (1*RTT), skickar sändaren **ett** segment som syns vid den första sändningsomgången och sedan vid 2*RTT skickas **två** segment. Denna dubblering av antalet segment i **cwnd** fortsätter vid 3*RTT; **fyra** segment, vid 4*RTT; **8** segment, och vid 5*RTT sändes **16** segment och tiden är tydligen 0,6 s. Detta beteende förklaras med att sändraen var i "slow start" som sedan fortsätter vid 6*RTT med **32** segment.

Fördubblingen av **cwnd** upphör vid omgång 7*RTT och **cwnd** ser ut att ha ökats med bara **ett segment** och sändaren skickade **33** segment och sedan **34** segment vid 8*RTT. Detta beteende förklaras med att sändaren övergick i "congestion avoidance" med linjär ökning av **cwnd** när en **tröskel** har nåtts (uppenbarligen **ssth** är satt till **32** segmnet).

Vid den sista, nionde sändningsomgången tog filen slut när de sista **24** segmenten av filens data har skickats.

Ett medelvärde på RTT kan räknas ut genom att t.ex. dela 0,6 s på 5 = 0,12s (**lätt att läsa i grafen**) och detta värde kan kännas igen med mulipla-RTT vid varje omgång (0,12, 0,24, 0,36, 0,48, 0,60, ...). Den totala tiden är därmed 10*RTT = 10 x 0,12 =1,2 s (sista RTT är för att ta emot de sista ACKs).

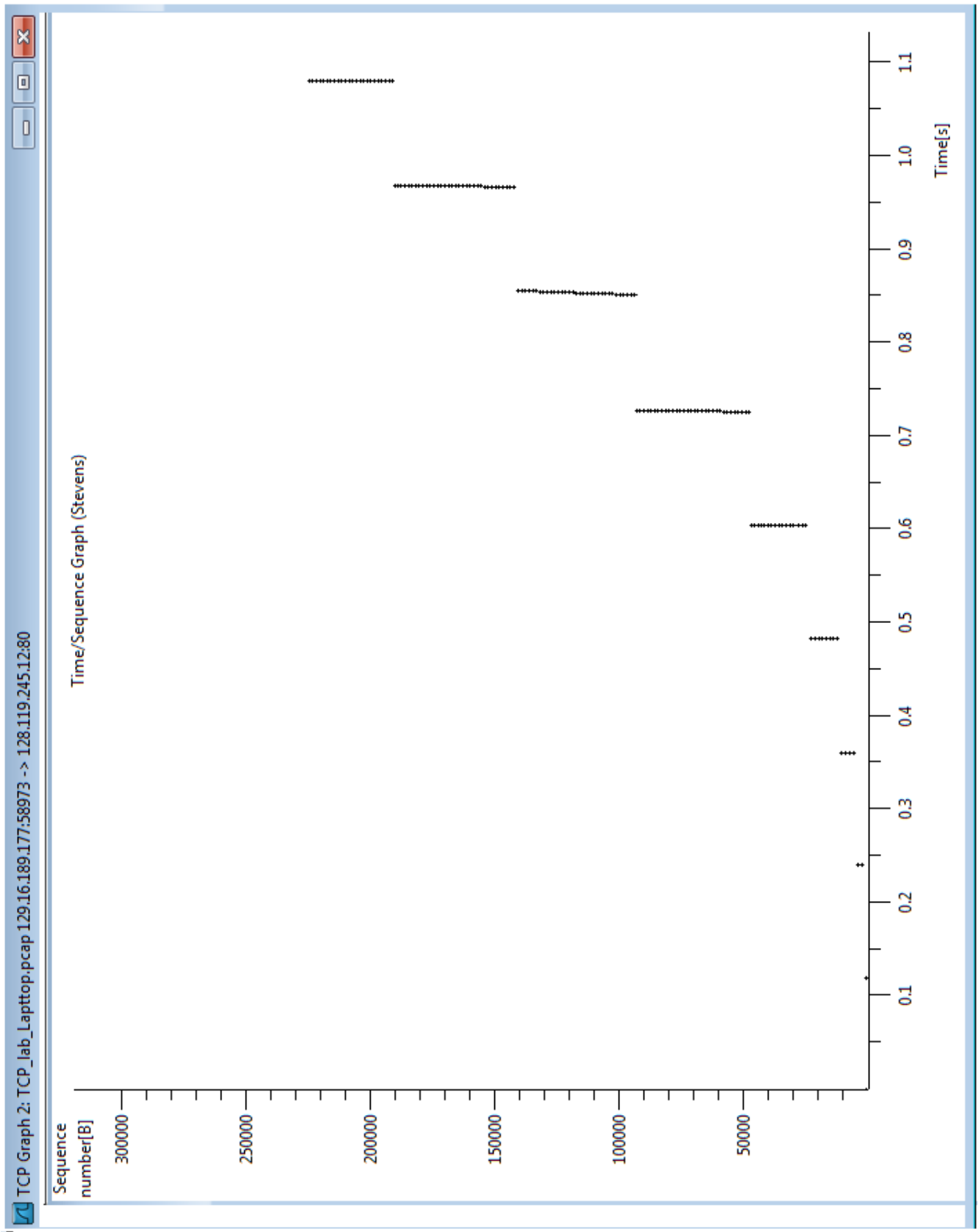
Den totala datamängden som motsvarar filstorleken kan räknas från antalet sända segment (1+2+4+8+16+32+33+34+24 = 154) och med antagandet att alla segment har normal MSS dvs 1460 bytes då blir det 154 x 1460 = 224 840 bytes. Det första segmentet innehöll POST-header samt de första bytes från filen. Throughput kan därmed räknas mycket noggrant med:

Th = 8*(224 840 – 487) bit/ 1,2 s = 1 495 686,67 bit/s ≈ 1,5 Mbps

Liknande ungefärligt värde på datamängden kan mätas i grafen som höjden på y-axeln vid slutet av sändningen och det är ca 225 000 bytes. Även med approximation och graf-värden fås samma resultat:

Th = 8*(225 000)/1,2 = **1,5** Mbps

(**Th** är mycket mindre än länkens bithastighet **R** som var **100** Mbps!).


TCP Graph 2: TCP_lab_Lapttop.pcap 129.16.189.177:58973 -> 128.119.245.12:80
Time/Sequence Graph (Stevens)
Sequence number[B]
300000
250000
200000
150000
100000
50000
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1.1
Time[s]

3. Ethernet & Trådlöst LAN 4p

Se avsnitten 5.4.3 och 6.3 i kursboken

Förklaring:

Brygga (bridge):
En lager-2 enhet innebär att datatrafiken inom enheten hanteras med ett länk-protokoll, vanligen MAC. Detta medför att enheten tar emot ramar på inkommande interface, bearbetar de olika fälten i header (och eventuellt trailer) för att skicka (eller vidarebefordra) innehållet i datafältet (ofta IP-paket) i en ram på ett utgående interface.

En lager-2 enhet brukar kallas för brygga (bridge) och arbetar inom ett lokalt nätverk, för att jämföra med lager-3 enhet som är IP-router på Internet.

MAC-adress-tabellen i Ethernet-switch:
Switchen är en flerport brygga som kontrollerar Ethernet-ramarnas MAC-adresser innan de skickas vidare. Alla switchportar är **Ethernet**-interface som använder **802.3 -MAC**. Switchen skapar och uppdaterar sin MAC-adress-tabell (dynamiskt) med självlärning. Vid varje inkommande ram på en switchport läser switchen av **sändarens MAC-adress** i ramens header för att spara **den** i MAC-adress-tabellen för denna port. Switchen använder denna tabell för att avgöra till vilken port skall en ram skickas vidare om mottagares MAC-adress finns (är lärt finnas) i portens MAC-adress-tabell. Switchen skickar vidare en ram till den port där **mottagarens MAC-adress** finns med i MAC-adress-tabellen.

Om en Ethernet-ram kommer in till switchen via en port och den skall till en mottagare med MAC-adress som **inte** finns i adress-tabellen **vidarebefordras kopia av ramen** till alla andra portar utom den port som ramen kommit ifrån.

WLAN-AP:
Vid infrastruktur-installationer, agerar accesspunkten (AP) som en MAC-brygga mellan den trådlösa (WLAN 802.11) och den trådbundna (Ethernet 802.3) delarna av LAN:et. AP har förutom **trådlöst interface,** ett **Ethernet-interface** anslutet till en switch. AP:ens uppgift vid en sådan installation, är att förmedla all trafik mellan stationerna oavsett MAC-typen (802.3 eller 802.11) och omvandlar ramarna från ena sidan till den andra.

3a) Varför betraktas både Ethernet-switchen och den trådlösa accesspunkten som lager-2 enhet i det lokala nätverket? Förklara tydligt. *(1p)*

Båda de två enheterna utför sina arbetsuppgifter genom att hantera innehållet i header-fälten på ramarna enligt ett MAC-protokoll som utför funktioner på länklagret (lager-2).

Switchen kontrollerar MAC-adresserna på de **802.3 Ethernet**-ramarna som skickas via switchens portar så att vidarebefordra till de anslutna enheterna som via switchen är kopplade till det lokala Ethernet-baserade nätverket. Ramarna passerar genom switchen utan modifiering.

AP arbetar aktivt och deltar i kommunikationen på länk-lagret genom att vara mottagare /sändare för 802.11 MAC-ramarna på radiolänken.

3b) Vad är huvudskillnaderna mellan Ethernet-switchen och den trådlösa accesspunkten i deras arbetssätt och funktionalitet som lager-2enhet? **Beskriv minst tre**. *(3p)*

Skillnaderna omfattar:

1. Förhållandet till de anslutna enheterna och rollen eller delaktigheten i lager-2 kommunikation mellan de anslutna enheterna.
2. Hanteringen av informationen i MAC-header (särskillt MAC-addresserna) inför leveransen av ram-innehållet (datafältet).
3. Typen av MAC-protokoll på enhetens interface, om det är samma eller olika protokoll på varje interface.
4. Tabeller för de anslutna enheternas MAC-adresser som används för leverans eller vidarebefordring av ramarna mellan de olika interfacen.

Switchen är **transparent** för de anslutna enheterna. Det är ingen lager-2 förhållande eller interaktion mellan switchen och de anslutna enheterna utom det som hanteras på det fysiska lagret (autosensing och autonegogiation).

Switchen bara läser av MAC-adresserna i MAC-headern utan att ändra på något eller vara delaktig i själva kommunikationen. Switchen använder MAC-adress-tabell för att kunna vidarebefordra ramarna, (Se förklaringen ovan).

Däremot kräver AP **associering** för de trådlösa enheterna (STAs) som kommer att informeras om AP via beacon-ramarna. Kommunikationen mellan de associerade stationerna (STAs) över radiolänken går via AP som en lager-2 mellanhandsenhet och som har MAC-adress (BSSID) för sitt trådlösa interfacet.

Om AP tar emot 802.11 MAC ram från en trådlös station STA som skall till annan i samma BSS kommer AP att bearbeta om MAC-headers olika fält och beräknar om trailer innan den skickar till mottagar-STA. AP behöver inte (förutom listan på de associerade STAs) ha MAC-adress-tabell liksom den som switchen skapar dynamiskt .

Vid infrastruktur-installationer agerar accesspunkten AP som en MAC-brygga mellan den trådlösa (WLAN 802.11) och den trådbundna (Ethernet 802.3) delarna av LANet.

AP har förutom **trådlöst interface,** ett **Ethernet-interface** anslutet till en switch. Vid en sådan installation har AP för uppgift att förmedla all trafik mellan stationerna oavsett MAC-typen (802.3 eller 802.11) och omvandlar ramarna från ena sidan till den andra.

När AP får en Ethernet-ram med mottagare-MAC-adress som tillhör en av de associerade trådlösa STAs inom sitt täckningsområde, extraherar accesspunkten datafältet, skapar en ny .11-ram som adresseras med användning av adresserna i Ethernet-ramen och AP sänder ramen över den trådlösa radiolänken.

Omvänt om AP får en .11 ram från en associerad STA och med mottagare-adress som **inte** tillhör annan STA, extraherar accesspunkten datafältet, skapar en ny .3 ram som adresseras med användning av adresserna i .11-ramen och sedan skickas ramen över Ethernet till switchen den är kopplad till.

4. IP-adresser 8p

4a) Ett nätverk har tilldelats prefixet 1**30.17.44.0/23**. Nätverket skall bestå av tre subnät som sammankopplas med en enda intern router. Ett av subnäten skall ha utrymme för **minst dubbelt** så många IP-adresser som vart och ett av de andra lika stora två subnäten. Hela adress-utrymmet i prefixet skall användas optimalt (fullt ut) för dessa tre subnet. *(5p)*

i. Beräkna subnäten enligt ovan. Ange subnät-adressen och subnät-mask för varje subnät i **decimal** form.

Det är nio bitar (från höger) i prefixet som skall användas för adressering. Man delar prefixet i två subnät. Det ena 130.17.44.**0/24** (255.255.255.0) används för det stora subnätet, och det andra 130.17.45.**0/24** delas ytterligare i två lika stora subnät 130.17.45.**0/25** och 130.17.45.**128/25** (255.255.255.128).

*PS. Man kan istället välja att dela upp det första subnätet 130.17.44.**0/24** i två lika stora subnät 130.17.44.**0/25** och 130.17.44.**128/25** (255.255.255.128).*

ii. Hur många gilltiga host-adresser har varje subnät utrymme för?

Det stora subnätet har utrymme för 254 = 2^8 -2 host-adresser, och varje av de två andra subnäten har utrymme för 126 = 2^7 -2 host-adresser.

iii. Välj lämpliga adresser för routerns interface i alla tre subnät.

Man kan välja exempelvis 130.17.44.**1/24**, 130.17.45.**1/25**och 130.17.45.12**9/25**. Varje router-interface (3 st.) skall tilldelas en giltig host-adress från varje subnät.

iv. Till vilket subnät tillhör följande adress **130.17.45.128**? Kan den användas som IP-adress för en värddator? Varför eller varför inte?

Enligt det första valet av subnäten är adressen 130.17.44.**128** reserverad som subnätets adress. Kan inte användas som unicast-adress (för en värddator) i IP-paketets header, utan enligt konvention kan den bara användas som information om subnätet i form av nåbar destinationadress i routingtabellen och uppdateringar.

4b) *(3p)*

i. Hur många 30-bitars subnät kan man få utav prefixet **33.22.11.160/27**? Motivera svaret.

Det är tre bitar (**101**XXX00) som kan användas för dessa subnät vilket ger åtta 30-bitars subnät;

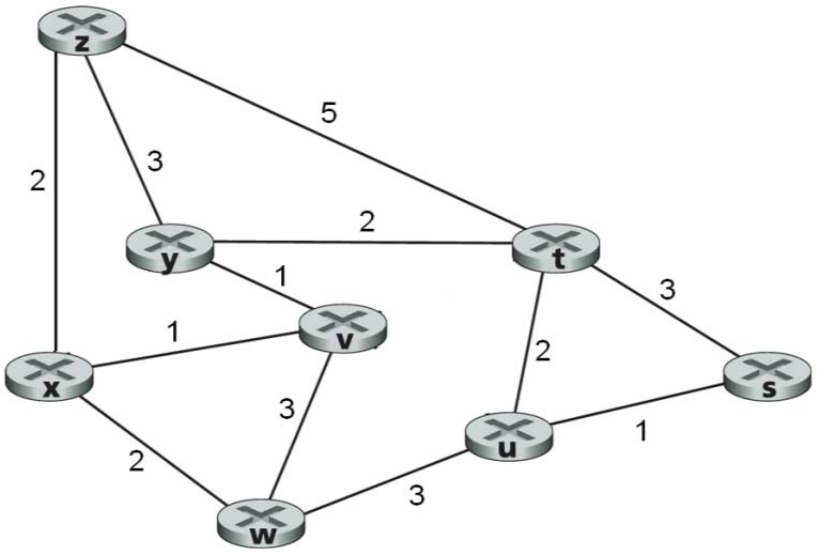
ii. Ange subnät-adresserna och subnät-masken i **decimal** form.
Det sista oktettet blir .160, .164, .168, .172, .176, .180, .184, .188/30, subnätmask 255.255.255.252.

iii. Välj det sista subnätet av de 30-bitars subnäten i din lösning och redovisa de adresserna som kan användas som host-adresser.

Subnätet 33.22.11.188/30, host- adresser (sista oktett): .189-.190

5. Routing och Routingalgoritmer 8p

5a) Figuren nedan visar topologin för ett IP-nätverk som består av åtta noder (routrar) markerade med bokstäver ”s, t, u, v, w, x, y och z”. Noderna är anslutna till varandra med de länkar som visas i figuren där siffrorna bredvid anger de aktuella länk-kostnaderna.



Anta att routing mellan noderna i den ovanstående figuren är baserad på algoritmen ”link-state”. Använd Dijkstra’s algoritm (*inte huvudräkning*) för att räkna ut den bästa vägen (med minsta kostnad) från nod ’**v**’ till **varje** annan nod på nätverket.
Redovisa dina resultat enligt algoritmen i en tabell steg för steg fram till lösningen. *(2p)*

Se avsnitt 4.5.1 i kursboken
Resultat enligt Dijkstra’s algoritm:

Steg	N’	D(s),p(s)	D(t),p(t)	D(u),p(u)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	v	∞	∞	∞	3,v	1,v	1,v	∞
1	vx	∞	∞	∞	3,v	----	1,v	3,x
2	vxy	∞	3,y	∞	3,v		----	3,x
3	vxyw	∞	3,y	6,w	----			3,x
4	vxywz	∞	3,y	6,w				----
5	vxywzt	6,t	----	5,t				
6	vxywztu	6,t		----				
7	vxywztus	----						

Observera att vid val mellan två noder som ger lika vägskostnad fordras **grannen** som SPF, annars slumpas en av dem.

5b) Baserat på beräkningen från **5a)**: *(2p)*

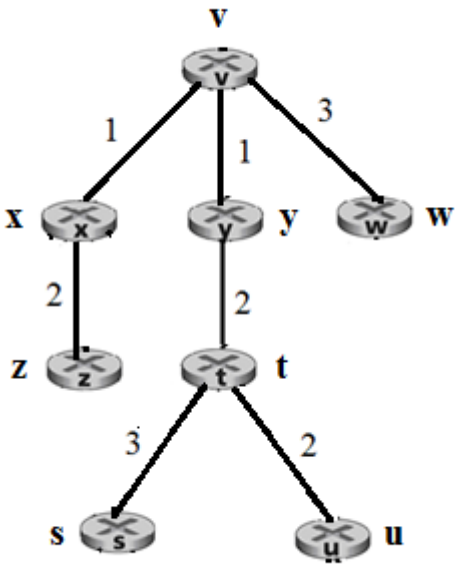
i. Sammansställ resultatet till en routing-tabell för nod ’**v**’.

Routing-tabell för nod ”v”:

Destination	Next hop	Cost
s	y	6
t	y	3
u	y	5
w	v	3
x	v	1
y	v	1
z	x	3

ii. Rita en graf (topologibild) för de bästa vägarna (med minsta kostnad) från nod ’**v**’ till alla andra noder i nätverket.

Graf: nod ”v” **är rot:**



5c) *(2p)*

- Vilken routinginformation anges med ”Distance Vector” DV? Hur förmedlas denna info i ett nätverk som använder DV-routingprotokoll?

”distance vector” är information om (via noden) **nåbar destination**, **vägkostnaden** och next hop.

Routingprotokoll som använder ”distance vector” skickar, med jämna intervaller, kopia av routingtabellen ***till grannroutrar*.**

- Vilken routinginformation anges med ”Link-State” LS? Hur förmedlas denna info i ett nätverk som använder LS-routingprotokoll?

”link state” är information om nodens identitet, grannroutrar, **status och kostnad av alla aktiva länkar** som noden är direktansluten till.

Routingprotokoll som använder ”link-state” skickar **”link state”** uppdatering vid uppstart och vid förändringar ***till alla andra routrar*** i nätverket.

5d) *(2p)*

- Vilken information innehåller en ”route” och som anges i routingtabellen? Använd egna ord för att ge generell beskrivning för en ”route”.

Varje rad i routingtabellen talar om ”för att nå en **destination** (IP-adress/subnätmask) anlitas **next-hop** (gateway eller nästa routers IP-adress) och paketen skickas **via interface**” (NIC eller annat gränssnitt för nätverkanslutning). Denna information kallas för en ”route”.

- Ange i decimal form informationen för en ”default route”.

Om ett IP-paket har mottagaradress som inte matchar någon specifik route i routingtabellen då skickas paketet över ”default route”. **0.0.0.0 0.0.0.0** är den decimala presentationen för denna route (IP-adress subnätmask).

Default route är den väg (via interface, next hop) som routern vidarebefordrar alla paket med mottagaraddresser som inte matchar specifika vägar (routes) i tabellen.

Destination	Mask	Next hop	Via interface	Metric
0.0.0.0	**0.0.0.0**	**x.y.z.w**	**interface I**	**M**

6. nslookup och DNS-information 6p

En student använder en dator som är ansluten till Internet via Chalmers nätverk.

Studenten kör kommandot *nslookup* och resultatet visas nedan.

C:\>**nslookup -type=MX kth.se**
Server: res1.chalmers.se
Address: 129.16.1.53

Non-authoritative answer:
kth.se MX preference = 20, mail exchanger = mx-alt1.kth.se
kth.se MX preference = 10, mail exchanger = mx.kth.se
kth.se MX preference = 30, mail exchanger = tarbaby.junkemailfilter.com

kth.se nameserver = ns2.chalmers.se
kth.se nameserver = b.ns.kth.se
kth.se nameserver = nic2.lth.se
kth.se nameserver = a.ns.kth.se

mx-alt1.kth.se internet address = 130.237.48.48
mx-alt1.kth.se internet address = 130.237.48.70
mx-alt1.kth.se internet address = 130.237.32.10
a.ns.kth.se internet address = 130.237.72.246
b.ns.kth.se internet address = 130.237.72.250
ns2.chalmers.se internet address = 129.16.253.252

Instruktioner för svaren:
- Studera **noggrant** den information som *nslookup* framställer.
- Dina svar på de följande delfrågorna måste innehålla **förklaringar**.
- I dina svar använd **DNS-termer** såsom; RR (Resource Record), domän, rekursivt**,** iterativt, lokal eller officiell namnserver, TLD-server, .. m.m.

6a) Vilken DNS-information efterfrågade studenten med hjälp av *nslookup*? Ditt svar skall förklara utförligt **kommandots syntax** som studenten använde i detta fall. *(1p)*

Studenten frågade efter **namnen på de officiella mailservrarna** för domänen **kth.se**. Studenten specificierat **RR**-typen med **-type=MX**, som står för ”**Mail eXchanger**” dvs mail-servrarna för domänen **kth.se**. Studenten angav inte vilken namnserver som skall kontaktas av DNS-klienten utan blir det en av de lokala Chalmers-servrarna som finns i IP-konfigurationen.

6b) Varför står det ” Non-authoritative answer”? Vad innebär det att svaret är icke-auktoritativt? Varifrån kommer detta svar? *(1p)*

Det står ” Non-authoritative answer” därför att **svaret kom från** res1.chalmers.se **(**med IP-adressen **129.16.1.53)** vilken är den lokala (s.k. cache-only) DNS-servern på chalmers.se vilken är naturligtvis **inte en** officiell ”authoritative” DNS-server för domänen **kth.se**.

Denna server erdjuder **rekursion** till DNS-klienter på Chalmers genom att kontakta andra namnservrar i hierarkin för att skaffa DNS-informationen. Informationen sparas i cache och kan används för att svara på DNS-frågorna. Till skillnad från ett ”Authoritative answer” om klienten får svaret direkt från en namnserver som har RR-databas om domänen (dvs. en officiell ”authoritative” DNS-server).

6c) Beskriv med egna ord hur olika DNS-servrar blev kontaktade i DNS-hierarkin för att få detta svar. Rita gärna en figur om DNS-kommunikationen som behövdes för ändamålet. *(2p)*

När Studentens DNS-klient skickade **DNS**-förfråga till sin lokala DNS-server på Chalmers, skickade denna server **iterativt** först en fråga till **root-server** om TLD-servern för **.se** och sedan efter svar skickade den lokala DNS-servern en ny fråga till **TLD-server** om DNS-servrar för **kth.se** för att sedan ta reda på de officiella email-servrarna för domänen. Efter att ha fått svar läggs informationen i cache och skickas svar till klienten.

6d) Vilka olika delar (sektioner) av DNS-informationen innehåller **DNS-meddelandet** som svar, och som används av *nslookup* för att framställa resultatet? Beskriv dessa delar av **meddelandet** tydligt och med egna ord ***utan kopior av namnen och adresser som visas ovan***. *(2p)*

(frågan är INTE om att beskriva vad som visas ovan i resultatet från nslookup)

De olika delarna (sections) av DNS-information i meddelandet, som innehåller svaret, är:

- ”Questions”: Själva frågan om RR av typ MX för domänen **kth.se**.
- ”Answers”: Svaret, dvs. de tre RRs för namnen på domänens mailservrar.
- ”Authority”: Information om de officiella ”authoritative” namnservrarna för domänen. 4 st. RRs av typ NS om namnen på de namnservrarna.
- ”Additional information”: Extra information i form av RRs av typ A om IPv4-adresser för en av de tre mailservrarna i ”Answers” samt RRs av typ A om IPv4-adresser för tre officiella ”authoritative” namnservrar för domänen.

****

Lycka Till!