1. Internet 6p

I detta dokument hänvisas till kursbokens 7:e upplaga.

1a) Se avsnitt 1.4.4 i kursboken och Wireshark-labb (TCP) (3p)

Förklaring: ”Throughput” är den **effektiva** (netto-) överföringshastigheten räknats i bps (*bit per second*) som kommunikationsprotokollen kan åstadkomma vid en kontrollerad överföring av hela datamängden från filen, på en hel **session** över Internet. Den räknas som det totala antalet överförda filens databitar **F** dividerat med den totala tiden **T** som sessionen tar från initiering (handskakning) tills överföringen är fullbordat (sista Ack). **Th** = **F/T** bps.

- Den totala tiden **T** för överföringen är inte enbart överföringen av data på länken (transmission time) utan det spenderas tid för protokollkontrollen (ex. vänta på Ack), fördröjningar i routrarna, overhead och att vägen består av delsträckor mellan klienten och servern som kan ha andra hastigheter och kö-förhållandena.

- Orsaker med negativ påverkan:

- **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 klienten och servern (end-to-end) påverkar (ökar) **RTT** (Round Trip Time) och den totala tiden för sessionen.
- Länkarna och routrarna på vägen mellan klienten och servern har olika bithastigheter och belastningsgrader. En låg-hastighets 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 meddelande, 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) Se avsnitten 1.4.3 och 5.6 i kursboken (3p)

Ett hopp är en delsträcka på en väg genom Internet och det består av en länk och router-interface. Syftet med att köra programmet är att kartlägga vägen (nätverk och routrar) från användarens värddator till måldatorn.

Numret är antalet hopp som motsvarar TTL-värdet på IP-paket som skickas av programmet.

Programmet visar också tre tider från att användarens värddator skickar paket tills den får ICMP-felmeddelande från varje router-interface på vägen dvs. RTT mellan användarens värddator och en router på vägen mäts tre gånger av programmet och visas i millisekund.

Namnet är CNAME (canonical) på routern vid varje hopp och som DNS-klienten tagit reda på med PTR-typ förfrågan på den IP-adressen som kommer med IP-paketet som sändarens adress. Det är IP-adressen för routerns interface som slänger ut paketet pga. att TTL förblir 0 (TTL exceeded).

Det är mycket troligt att det är pga. tillfälliga köbildningar och mycket trafik på SUNET-routrarna dvs. tillfällig stockning och långa köar i routrarna.

Mer Förklaring:
Tracert skickar IP-paket som innehåller ICMP-echo request meddelande upprepade gånger och samtliga är adresserade till måldatorn.

I första omgång sätts TTL-värdet i IP-paket till 1 och sedan ökas det med 1 vid nästa omgång osv. Varje omgång upprepas tre gånger med samma TTL-värde. När dessa paket skall routas på Internet, passeras ett antal routrar på vägen till måldatorn. Varje router
minskar TTLvärdet i paketet med 1 innan den vidarebefordrar det till nästa hopp. Ett paket med TTL = 0 kastas bort av routern och sändaren informeras av denna router genom ett skicka ICMP-meddelandet ”TTL exceeded”. Värddator (som kör tracert) använder informationen i dessa ICMP-meddelanden för att sammanställa en lista på de routrarna på vägen samt ett mätvärde för RTT till varje router tre gånger. Sista omgång når IP-paketen måldatorn som svarar med ICMP- echo reply meddelande.

Vid varje hopp visas:
- hopp-nummer som motsvarar värdet på TTL i de skickade IP-paketen innehållande ICMP-echo,

- tre uppmätta RTT-tider för varje omgång (hopp) från att skicka ICMP-echo tills värddaorn tar emot ICMP-meddelande ”TTL exceeded”.

- måldatorns eller routerns DNS-hostnamnet ”CNAME” vilket DNS-klienten på värddatorn (som kör tracert) tar reda på med DNS-fråga om typ-PTR,

- måldatorns eller routerns IP-adress vilken hittas som sändaradress i paket innehållande retur ICMP-meddelande.

2. HTTP och Webben 6p

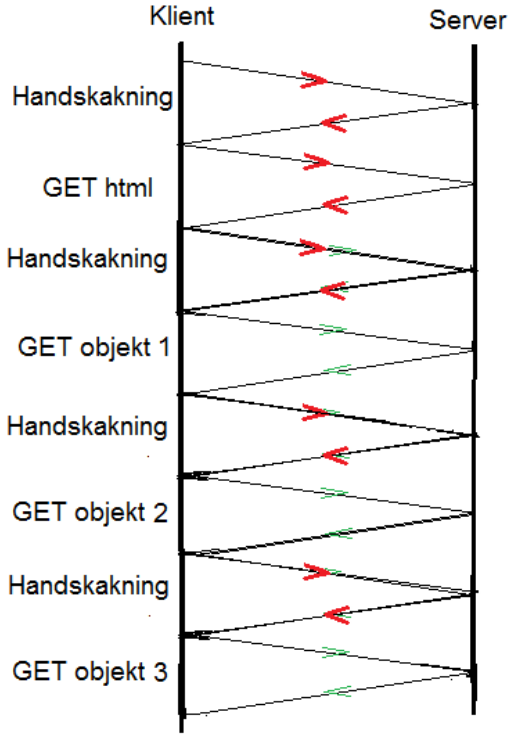
Se avsnitten 2.2.2 och 2.1.2 i kursboken
*För mer förklaring se **slides 18-22** i Kapitel_2 föreläsningsbilder.*
Se också kursbokens frågor (Ch.2 Problems: P7-8 och P10-11)

2a) **(1p)**

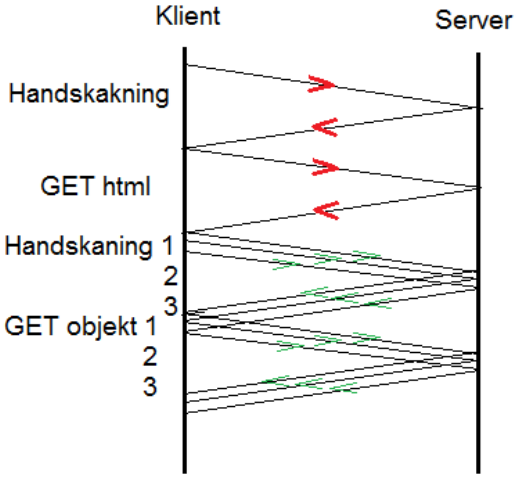
HTTP med beständiga ”persistent” anslutningar innebär att webbklienten först skapar en TCP-anslutning för att hämta HTML-filen och servern behåller denna anslutning öppen (Connection: Keep-Alive) tills klienten har hämtat de tillhörande objekten. Klienten kan hämta objekten parallellt eller seriellt med samma TCP-anslutning.

2b) **(3p)**

Det skapas en TCP-anslutning för att först hämta html-basfilen, vilket tar tid motsvarande 2*RTT. Med icke-beständiga anslutningar och efter hämtningen av html-filen stängs av anslutningen och klienten skapar **tre** olika nya anslutningar och genom varje anslutning begär klienten ett av de tre objekten med ett oberoende GET-meddelande. Servern kommer att svara med HTTP-respons OK på varje GET och förhoppningsvis direkt efter varandra. Förloppet därmed tar minst (2 + 2)*RTT för att webbläsaren kan visa hela webbsidan om hämtningen sker parallellt och det är dubbelt 8*RTT om det sker seriellt.



Seriell hämtning med
icke-beständiga (non-persistent) anslutningar



Parallell hämtning med
icke-beständiga (non-persistent) anslutningar

2c) **(1p)**

11 sockets, en gemensam socket som identifieras med portnummer 80 och serverns IP-adress, kallas välkommen-socket som alla klienter börjar handskakning med. För varje klient skapas sedan s.k. connection-socket som identifieras med port 80, serverns IP-adress, klientens portnummer och IP-adress.

2d) **(1p)**

För att tillmötesgå den stora efterfrågan från stort antal webbklienter skapas ett kluster av maskiner med olika IP-adresser som har samma DNS-namn och med samma innehåll på webbplatsen (replika). DNS-auktoritativa namnservern skall rotera adresserna (round robin) i sina svar om typ-A RRs för hostnamnet på servern för att sprida belastningen mellan flera maskiner.

3. DNS 6p

Se avsnitt 2.4 i kursboken och Wireshark-labb (DNS)

3a) **(3p)**

NS-servrarna utgör ett logiskt träd: Root, Top-Level Domain, Auktoritativa

Root-servern har databas med RRs för namnen och IP-adresser för TLD-servrarna.

TLD-servern har databas med RRs för namnen och IP-adresser för de auktoritativa servrarna.

Den **auktoritativa** servern har databas med alla RRs som tillhör organisationens nätverk och domän (t.ex. namnen och IP-adresser för de olika servrarna som tillhör organisationen).

DNS gigantiska databas är distribuerad mellan dessa servrar och speglar själva namnstruktur.

Om DNS-klient skickar förfråga till root-server om IP-adressen för www.chalmers.se så får man inte adressen, men root-servern hänvisar till TLD-servrarna för.se genom att ange NS-namnen och adresserna för .se. Om nu klienten skickar samma fråga till en TLD server under Sveriges domän, får man inte heller adressen utan hänvisar TLD-servern till de auktoritativa servrarna för Chalmers domän genom att ange NS-namnen och adresserna för chalmers.se. Nu kan klienten skicka förfrågan till en av Chalmers DNS-servrarna t.ex. ns1.chalmers.se och slutligen får svar med adressen förutsatt att namnet är rätt värddator-namn.

3b) **(2p)**

nslookup använder datorns DNS-klient för att skicka förfrågor till DNS-servrar i hierarkin. DNS-klienten, i detta fall skickar **DNS**-förfråga till Googles auktoritativa namnservern **ns2.google.com** som har ansvar om Google-domän och inte om andra domäner, med andra ord denna server har databas om Google-domäns RRs och inte erbjuder rekursivt svar. Därför svarar denna server inte på förfrågan utan ”Query refused”.

Kommandot använder default RR-typ **A** som innebär att klienten söker IP-adressen på webbservern för domänen **ibm.com**.

.

3c) **(1p)**

DNS använder främst User Datagram Protocol (UDP) på port nummer 53 för att transportera förfrågningar och svaren. DNS-meddelanden består av antingen en enda DNS-förfråga från klienten eller av ett enda DNS-svar från servern. Dessutom är DNS-meddelandet kompakt och kort dvs. innehåller data på ett antal fördefinierade fält som vanligen inte överstiger 512 byte. UDP header är 8 byte och IP-header är normalt 20 byte och svaret innehåller IP-adresserna för klienten (destination) och servern (avsändare).

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 icke-nödvändiga TCP-anslutningar.

4. Transportprotokollen 8p

4a) Se avsnitt 3.2 och figur 3.3 i kursboken (2p)

Dessa nummer är sändar-portnummer och mottagar-portnummer och används av både TCP och UDP för att identifiera vilket socket (applikation) som segmentets data kom ifrån på en host och till vilket socket (applikation) på den andra hosten skall data levereras. På server-sida är portnummret allmänt känt (0-1023) för en specifik applikation, medan på klient-sidan ofta slumpas numret från ett intervall (49152–65535).

4b) Se avsnitt 3.7 (Fast Recovery and Reno) i kursboken 6p)

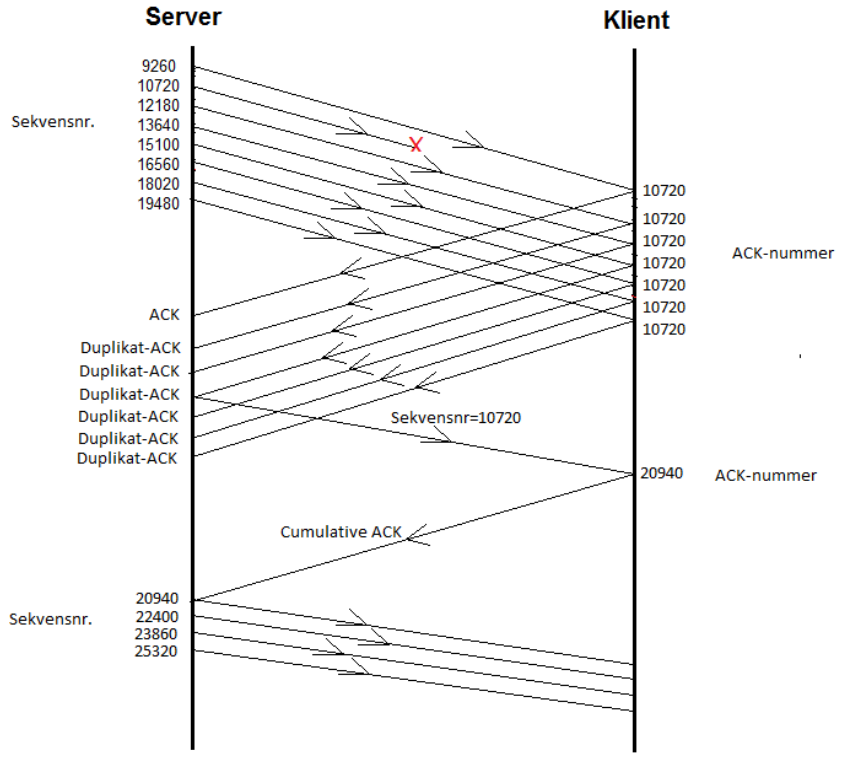
Se också slides 68 & 92 Kapitel_3

TCP-mottagaren kan placera segmenten i buffert men det blir gap. TCP-mottagaren hos klienten kommer att skicka duplikat ACK varje gång kommer segment som inte fyller gapet för segment med sekvensnummer 10720.

TCP-sändaren hos servern får trippel duplikat ACKs för ett och samma segment. Detta indikerar en kortvarig stockning vilket innebär att ett tidigare segment bland de sända segmenten är förlorat. TCP övergår till ”Congestion Avoidance” efter ”Fast Recovery” dvs. sända om segmentet med sekvensnummer 10720. Nya värdet på tröskeln anger storleken i antal segment på halva “Congestion Window” vid senaste händelse. Det är uppenbart att CogWin var 8 segment då blir nya tröskel 4 segment.

När mottagaren tar emot segmentet med sekvensnummer 10720 som fyller gapet, skickar den omedelbart kumulativ ACK som bekräftar alla mottagna segmenten i ordning och utan fel och i detta fall blir ACK-nummer lika med sekvensnumret 19480 + datamängden 1460 (MSS) = 20940.

”Congestion Avoidance” är det då sändar-TCP har nått tröskeln och börjar öka CongWin med ett segment i taget efter varje RTT dvs. linjär ökning (om det får ACK på alla tidigare sända segmenten) för att undvika stockningen.



5. Ethernet & Trådlöst LAN 8p

5a) Se avsnitten 6.4.1-6.4.3 i kursboken **(4p)**

- Värddatorn börjar med att genomföra AND-operation mellan serverns IP-adress och subnätmasken och konstaterar att den tillhör inte samma subnät (extern adress).

- Värddatorn **A** är konfigurerad med IP-adress för access-routern som default gateway för att skicka paket utanför sitt eget subnät. IP-paketen skall kapslas in i Ethernet-ramar inför överföringen inom det lokala nätverket. Då behöver värddatorn veta routerns MAC adress och med hjälp av **ARP** skickas en broadcast-förfråga så att den som har gateways IP-adress skall svara med sin MAC adress.

- När ramen innehållande ARP-förfrågan kommer till Ethernet-switchen med MAC-mottagaradressen som är broadcast (FF-FF-FF-FF-FF-FF) kommer switchen att vidarebefordra kopia av ramen till alla andra porter 2, 3, och 4 förutom port 1 som värddatorn A är ansluten till. Switchen sparar A-MAC mappat till port 1 med hjälp av sändaradressen i ramens header.

Routerns Ethernet-interface får en kopia av ramen via sin switchport 4 och läser av IP-adressen i ARP-meddelandet och konstaterar att det är sin egen. Då skickar den sin MAC-adress i ett ARP-svar inkapslat i en unicast-ram på det lokala nätverket, adresserat till värddatorns MAC-adress med destination A-MAC.

- När ramen innehållande ARP-svar kommer till switchen vidarebefordrar switchen ramen endast till port 1 som värddatorn A är ansluten. Samtidigt som switchen uppdaterar mappningen R-MAC → port 4.

- Värddatorn tar emot ARP-svaret och lagrar gateways MAC-adress/IP-adress i ARP-tabellen. Nu kan värddatorn skicka IP-paket (innehållande SYN-TCP-segment) som är adresserade till den externa serverns IP-adress genom att kapsla detta paket i Ethernet-ram adresserad till gateways MAC-adress. Access-routern R använder sedan sin routingtabell för vidareleverans av paketet över Internet. Routern uppdaterar ARP-tabellen för sitt Ethernet-interface med mappning A-MAC → A-IP.

Switchen nu har båda mappningar: A-MAC → port 1 och R-MAC → port 4.

I fortsättning är alla tabeller konsistenta för denna kommunikation, så att paket från servern innehållande TCP-segment med ACK-SYN levereras till A direkt. MAC- och ARP-tabellerna uppdateras (refreshed) så länge kommunikationen pågår mellan värddatorn A och den externa servern.

5b) Se avsnitt 7.3.2 i kursboken **(4p)**

I **WLAN** tillämpas **CSMA/CA** mekanismer kollektivt (Multiple Access) så att en trådlös station STA som vill sända en ram med *normal* storlek, skall först lyssna på radiokanalen (Carrier Sense) och se om det är ledig.

Är kanalen ledig, väntar varje STA en förbestämd tid **DIFS**, sänder **hela ramen** om kanalen är fortfarande ledig och sedan väntar varje STA på **ACK** från mottagaren (som i detta fall är den associerande accesspunkten AP). Då händer det en kollision

vid AP som gör att AP:en inte kan skicka ACK till någon av de två stationerna. En positiv bekräftelse ”ACK” används av CSMA/CA för att informera sändaren om lyckad överföring över radiolänken. ACK är nödvändigt med anledningen av att radiolänken är mer utsatt för störningar, brus och interferens så att de sända ramarna kan lätt drabbas av bitfel och även **kollisioner** kan, trots dessa mekanismer, inträffas.

Efter en viss time-out (vänte-tid på ACK), är utebliven ACK en indikation för sändaren om att försöka sända om samma ram och därför **väljs nu slumpmässigt en väntetid** (**back-off time)** av varje STA-sändare **för att försöka** sända om sin ram.

Förhoppningsvis väljs back-off-tid olika av stationerna och varje STA startar nedräkning av timer när kanalen är ledig och en av STAs blir först som kan sända sin ram om den valde tiden är kortare. Den andra STA stoppar sin timer under sändningen (kanalen är inte ledig längre). Blir det en annan kollision med tredje STA ökas back-off-tiden nu så att en STA blir vinnare och sändningen lyckas.

6. IP-adresser och subnetting 6p

Se avsnitt 4.3 i kursboken, extra material och Lab-2

En lösning är att först dela upp prefixet **33.22.20.0/22** i två lika stora subnät:

De första subnätet **33.22.20.0/23** tilldelas den stora nätverksdelen och det andra **33.22.22.0/23** skall subnettas ytterligare i två mindre subnät.

Det ena subnätet **33.22.22.0/24** kan användas för adresseringen av en av de mindre nätverksdelarna medan det andra **33.22.23.0/24** skall användas för adresseringen av den andra.

(Alternative tilldelas subnätet **33.22.22.0/23** den stora avdelningen medan subnätet **33.22.20.0 /23** delas upp i två mindre subnät **33.22.20.0/24** resp. **33.22.21.0 /24** för de två mindre avdelningarna.)

6a) **(3p)**

33.22.20.0, mask: 255.255.254.0 för den stora avdelningen, vilket ger 510 IP-adresser (2^9 -2) **33.22.20.1** upp till **33.22.21.254**.

33.22.22.0, mask: 255.255.255.0 för en av de mindre avdelningar, vilket ger 254 IP-adresser (2^8 -2) **33.22.22.1** upp till **33.22.22.254.**

33.22.23.0, mask: 255.255.255.0 för den andra mindre avdelningen, vilket ger 254 IP-adresser (2^8 -2) **33.22.23.1** upp till **33.22.23.254.**

6b) **(1p)**

3 Ethernet Interface 100/1000 Mbps:

33.22.20.**1**/23 33.22.22.**1**/24 33.22.23.**1**/24

1 Seriell Interface: 200.150.100.**250**/30 (Förutsatt att ISP använder 200.150.100.**249**/30)

6c) **(1p)**

IP-adress: 33.22.21.254, Mask: 255.255.254.0, Default Gateway: 33.22.20.1

6d) **(1p)**

På PC:n konfigureras som default gateway den IP-adress som en routers interface i subnätet är konfigurerad med.

När en IP-destination ligger utanför det eller de nätverk som PC:n är ansluten till (avgörs genom jämförelse, bitvis AND, med egen subnätmask och både egen och destinations IP-adress) kommer PC att skicka ARP för den IP-adress som är konfigurerad som default gateway för att få dennes MAC-adress. Därefter skickas ramen med default gateways MAC-adress som destination på länklagret och IP-adress för slutmålet som destination på nätverkslagret.
Default gateway är next-hop för default route (0.0.0.0/0) i PC:s routingtabell.