

Naam	Initialen	Studentnummer	Studierichting

Tentamen Telematica Systemen en Toepassingen (261000)

3 november 2010, 8.45 – 12.15

Opmerkingen:

- Alleen 1 dubbelzijdig blad met aantekeningen / samenvatting (ongeacht lettergrootte / dichtheid) en een woordenboek zijn toegestaan als hulpmateriaal. Het gebruik van het boek van Kurose en Ross of enig ander gedrukt materiaal is niet toegestaan.
- Het gebruik van een rekenmachine is toegestaan. Gebruik van PDA, laptop computer, mobiele telefoon, enz., is niet toegestaan. Schakel je mobiele telefoon uit en berg hem op.
- De vragen worden gesteld in het Nederlands. Je mag in het Nederlands of Engels antwoorden, afhankelijk van waar je de voorkeur aan geeft.
- Geef je antwoorden op deze bladen. Wanneer je meer ruimte nodig hebt mag je extra papier gebruiken.
- Aanduidingen zoals “[10]” bij vragen betekenen dat je 10 punten voor die vraag kunt verdienen.
- Vul je naam, studentnummer, enz., bovenaan deze bladzijde in, en op mogelijke extra vellen papier die je gebruikt.

Alleen voor de docent:

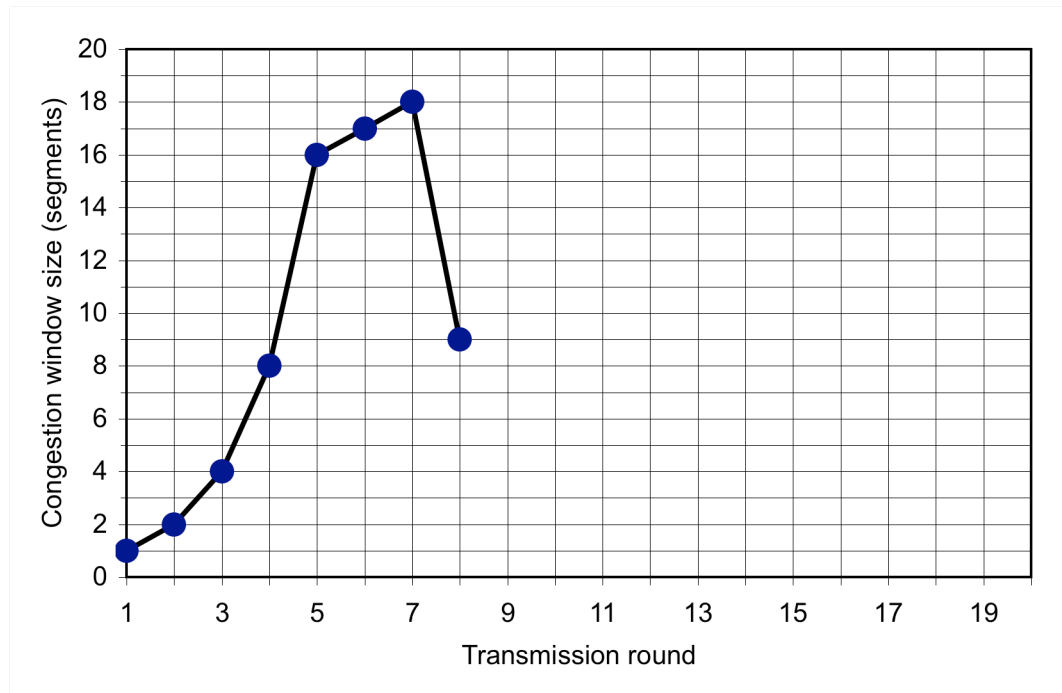
Vraag	1	2	3	4	5	6	7	Totaal
Punten								
Maximum	7	6	5	15	16	15	13	77

Lijst van afkortingen:

ACK	ACKnowledgement
AJAX	Asynchronous Javascript and XML
ARP	Address Resolution Protocol
DNS	Domain Name System
DV	Distance Vector
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IMAP	Internet Mail Access Protocol
IP	Internet Protocol
LS	Link State
MAC	Medium Access Control
MSS	Maximum Segment Size
P2P	Peer-to-Peer
POP3	Post Office Protocol – version 3
PDF	Portable Document Format
SMTP	Simple Mail Transfer Protocol
TCP	Transmission Control Protocol
URL	Uniform Resource Locator

4) TCP [15]

Beschouw het volgende verloop van de grootte van het TCP *congestion window* als functie van de tijd. De gebruikte versie van TCP is TCP Reno (TCP met *fast retransmit* en *fast recovery*)



- Wordt na de 7e transmissieronde pakketverlies gedetecteerd door het ontvangen van 3 *duplicate ACKs* of door een *time-out*? [1]
- Wat is de initiële waarde van de variabele `Threshold` (tijdens de 1ste transmissieronde)? [1]
- Wat is de waarde van de variabele `Threshold` tijdens de 8e transmissieronde? [1]

In de onderstaande vragen moet je aangeven hoe het verloop is van de grootte van het *congestion window* na transmissieronde 8. Hierbij is het gegeven dat direct na transmissieronde 15 pakketverlies wordt gedetecteerd door het optreden van een *time-out*. Voor de rest vindt er geen pakketverlies plaats.

- Wat is de grootte van het *congestion window* tijdens transmissieronde 15? [1]
- Wat is de grootte van het *congestion window* tijdens transmissieronde 16? [1]
- Wat is de waarde van de variabele `Threshold` tijdens de 16e transmissieronde? [1]
- Wat is de grootte van het *congestion window* tijdens transmissieronde 20? [2]

In de rest van deze opgave beschouwen we de delay die TCP introduceert voor een webclient die een pagina van grootte F van een server haalt. Neem hierbij de ontwikkeling van het congestion window aan zoals die in bovenstaande grafiek beschreven staat. Hierbij gaan we er van uit dat TCP segmenten verstuurt met een grootte (MSS) van 1250 bytes (10^4 bits). De grootte van headers en een http request bericht veronderstellen we verwaarloosbaar klein. We gaan er van uit dat de webclient en –server verbonden zijn door een enkele link met data rate $R = 10$ Mbps (10^7 bits/s). De roundtrip time van de link is $RTT = 2$ ms. Stel dat $F = 3750$ bytes.

- h) Hoe lang duurt het totdat de webclient de gehele pagina heeft binnengehaald? Licht je antwoord toe met behulp van een time-sequence diagram en een formule. [3]

Stel nu dat de webpagina naast de HTML file (van grootte $F = 3750$ bytes) ook nog een plaatje van $G = 8750$ bytes bevat.

- i) Hoe lang duurt het totdat de webclient de gehele pagina inclusief plaatje heeft binnengehaald in het geval HTTP met *non-persistent connections* wordt gebruikt? Licht je antwoord toe met behulp van een time-sequence diagram en een formule. [2]

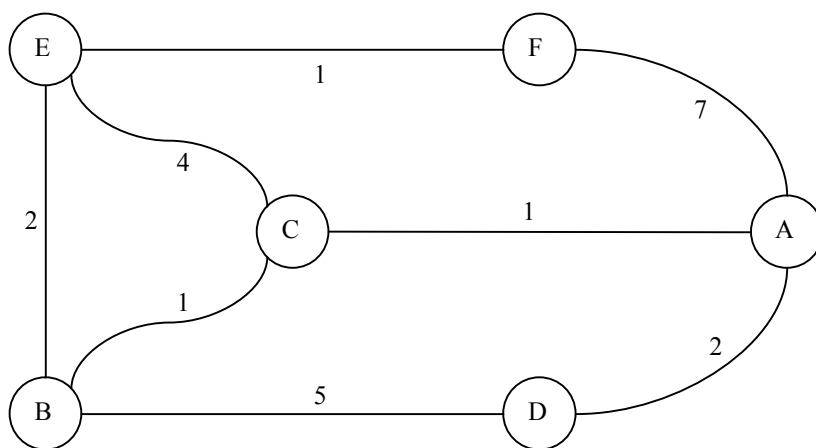
- j) Hoe lang duurt het totdat de webclient de gehele pagina inclusief plaatje heeft binnengehaald in het geval HTTP met *persistent connections* wordt gebruikt? Licht je antwoord toe met behulp van een time-sequence diagram en een formule. [2]

5) Routing [16]

- a) Er bestaan twee belangrijke routeringsalgoritmen: *distance vector* (DV) en *link state* (LS). Geef van de volgende beweringen aan, of deze waar zijn voor DV en voor LS (geen van beide of allebei is ook mogelijk), door in de lege vakjes ja of nee in te vullen: [7]

<i>Bewering</i>	<i>waar voor DV ?</i>	<i>waar voor LS ?</i>
het is een gedecentraliseerd routeringsalgoritme		
het is een globaal routeringsalgoritme		
het is een hiërarchisch routeringsalgoritme		
het kan last hebben van "counting to infinity"		
het vindt de goedkoopste paden		
elke node weet de kosten van het gevonden beste pad naar elke andere node		
elke node weet welke nodes er liggen op het gevonden beste pad naar elke andere node		

Beschouw nu het volgende netwerk met de daarin aangegeven link-kosten:



- b) Gebruik Dijkstra's algoritme voor *link-state* routing om de kortste paden van node F naar alle andere nodes te berekenen. Gebruik het algoritme om de volgende tabel in te vullen, waarin N' de verzameling van al afgehandelde nodes is, D(x) de afstand tot node x, en p(x) de voorganger van x op het kortste pad van F naar x. [5]

step	N'	D(A), p(A)	D(B), p(B)	D(C), p(C)	D(D), p(D)	D(E), p(E)
0	{F}	7,F	∞	∞	∞	1,F
1						
2						
3						
4						
5						

Neem nu aan dat er in ditzelfde netwerk een *distance-vector* algoritme wordt gebruikt **zonder** “*poisoned reverse*”. Ga er voor de volgende twee subvragen vanuit dat het algoritme geconvergeerd is.

c) Geef de *distance-vector* die node F aan node E stuurt. [2]

Neem nu aan dat er in ditzelfde netwerk een *distance-vector* algoritme wordt gebruikt **met** “*poisoned reverse*”.

d) Geef de *distance-vector* die node F aan node E stuurt. [2]

Stel dat slechts **één** node data te versturen heeft, terwijl de andere $N-1$ nodes geen data te versturen hebben.

d) Wat is de maximum *throughput* van het *broadcast* kanaal wanneer *channel partitioning* wordt gebruikt? Leg je antwoord uit. [2]

e) Wat is de maximum *throughput* van het *broadcast* kanaal wanneer slotted Aloha wordt gebruikt? Leg je antwoord uit. [2]

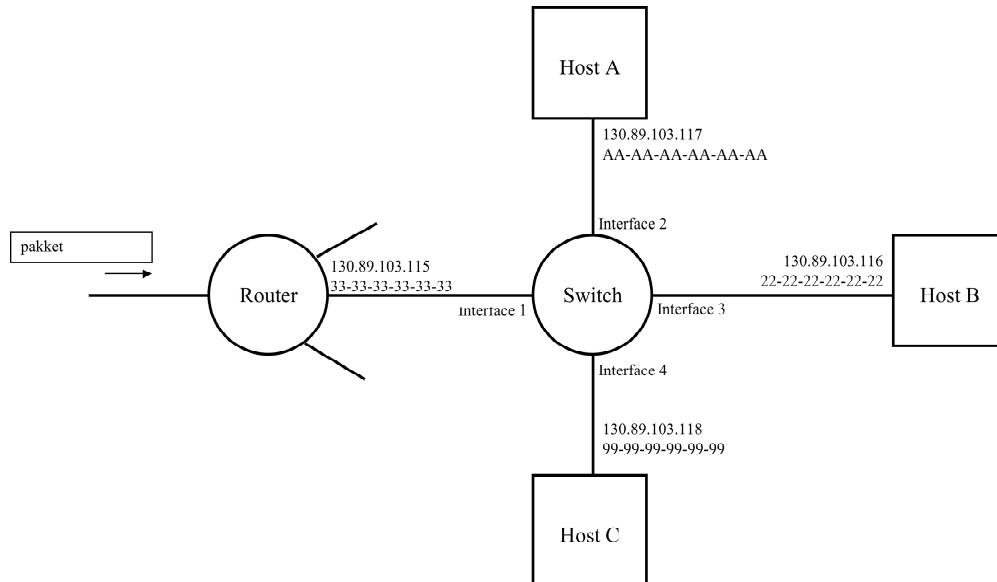
f) Wat is de maximum *throughput* van het *broadcast* kanaal wanneer *polling* wordt gebruikt? Leg je antwoord uit. [2]

Neem nu aan dat **meerdere** nodes data te versturen hebben.

g) Leg voor ieder van de 3 genoemde *multiple access* protocollen uit hoe welke invloed dit heeft op de *throughput* van het broadcast kanaal (zoals gevonden in respectievelijk d, e en f). Blijft deze gelijk, stijgt deze, of daalt deze? Leg je antwoord uit. (Opm.: je hoeft geen getal of formule te geven). [3]

7) Switches and routers [13]

Beschouw het netwerk in de onderstaande figuur. Een *router* is met één van zijn vier interfaces met een *switch* verbonden, die verder met Host A, Host B en Host C verbonden is. De *interfaces* van de *switch* zijn genummerd, en voor ieder van de met de *switch* verbonden nodes is het MAC adres en het IP adres gegeven.



Gedurende deze gehele vraag zullen we een pakket met *destination* adres 130.89.103.116 bekijken (met bestemming Host B), dat in de *router* aankomt.

a) Schrijf het *destination* adres in binaire notatie. [1]

Stel dat de *forwarding* tabel van de router de volgende *entries* bevat:

Destination Address	Link Interface
130.89.103.0/24	1
130.89.103.64/28	2
130.89.103.114/29	3
0.0.0.0/0	4

b) Met welke *entry/entries* heeft het *destination* adres een *match*? [2]

c) Welke *link interface* zal worden gekozen om het pakket naar door te sturen? Waarom? [2]

Laten we nu aannemen dat het pakket is doorgestuurd naar de *link interface* die verbonden is met de *switch*. Stel dat de ARP tabel voor deze interface (van de *router*) een *entry* bevat die relevant is voor het genoemde pakket. Neem verder aan dat de *switch* tabel (van de *switch*) initieel leeg is.

- d) Welke kolommen kun je vinden in een ARP tabel? [2]
- e) Op welke *interface(s)* zal het pakket door de *switch* verstuurd worden (nadat het is aangekomen van de *router*)? [1]
- f) Hoe zal de *switch* tabel van de *switch* er uit zien nadat het pakket doorgestuurd is naar Host B? Gebruik onderstaande tabel om de *entries* in te vullen. (de kolom tijd is weggelaten en hoeft niet ingevuld te worden). [2]

Address	Interface

We beschouwen nogmaals dezelfde situatie als in deelvraag e) en f), maar nu nemen we aan dat ook de ARP tabel initieel leeg is.

- g) Op welke *interface(s)* zal het pakket door de *switch* verstuurd worden (nadat het is aangekomen van de *router*)? [1]
- h) Hoe zal de *switch* tabel van de *switch* er uit zien nadat het pakket doorgestuurd is naar Host B? Gebruik onderstaande tabel om de *entries* in te vullen. (de kolom tijd is weggelaten en hoeft niet ingevuld te worden). [2]

Address	Interface

Uitwerkingen

Tentamen Telematica Systemen en Toepassingen (261000) 3 november 2010, 8.45 – 12.15

Opmerkingen:

- Alleen 1 dubbelzijdig blad met aantekeningen / samenvatting (ongeacht lettergrootte / dichtheid) en een woordenboek zijn toegestaan als hulpmateriaal. Het gebruik van het boek van Kurose en Ross of enig ander gedrukt materiaal is niet toegestaan.
- Het gebruik van een rekenmachine is toegestaan. Gebruik van PDA, laptop computer, mobiele telefoon, enz., is niet toegestaan. Schakel je mobiele telefoon uit en berg hem op.
- De vragen worden gesteld in het Nederlands. Je mag in het Nederlands of Engels antwoorden, afhankelijk van waar je de voorkeur aan geeft.
- Geef je antwoorden op deze bladen. Wanneer je meer ruimte nodig hebt mag je extra papier gebruiken.
- Aanduidingen zoals “[10]” bij vragen betekenen dat je 10 punten voor die vraag kunt verdienen.
- Vul je naam, studentnummer, enz., bovenaan deze bladzijde in, en op mogelijke extra vellen papier die je gebruikt.

Alleen voor de docent:

Vraag	1	2	3	4	5	6	7	Totaal
Punten								
Maximum	7	6	5	15	16	15	13	77

Lijst van afkortingen:

ACK	ACKnowledgement
AJAX	Asynchronous Javascript and XML
ARP	Address Resolution Protocol
DNS	Domain Name System
DV	Distance Vector
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IMAP	Internet Mail Access Protocol
IP	Internet Protocol
LS	Link State
MAC	Medium Access Control
MSS	Maximum Segment Size
P2P	Peer-to-Peer
POP3	Post Office Protocol – version 3
PDF	Portable Document Format
SMTP	Simple Mail Transfer Protocol
TCP	Transmission Control Protocol
URL	Uniform Resource Locator

1) World Wide Web [7]

- a) Worden Javascript programma's uitgevoerd door de server of door de browser? Wordt er in Web 2.0, als onderdeel van het AJAX framework, nog steeds gebruik gemaakt van Javascript, of heeft men daar voor een andere technologie gekozen? [1]
- *Javascript programma's worden altijd door de browser uitgevoerd*
 - *Binnen Web 2.0 wordt zeer sterk gebruik gemaakt van Javascript technologie. Dit blijkt ook uit het feit dat AJAX een afkorting is van "Asynchronous Javascript And XML"*
- b) Waarom stuurt HTTP de hele URL, dus ook de hostname mee? De hostname is immers reeds via DNS tot een IP adres omgezet, en de IP en TCP protocollen hebben er al voor gezorgd dat een verbinding met de web server machine is opgebouwd. [2]
- *Op 1 machine, dus op 1 IP adres, kunnen meerdere web servers draaien. Als je de hostname niet zou meesturen, weet de web server machine niet welke web server het request moet afhandelen. Daarom wordt ook de hostname van de server meegestuurd.*
- c) Wat is het doel van het HTTP 'Conditional Get' commando? [2]
- *Met het 'Conditional Get' commando kan een web cache een web server vragen een specifiek object alleen te versturen als het na een bepaald tijdstip nog is aangepast. Als het niet is aangepast, stuurt de web server een response message met als status: "304 Not Modified".*
- d) Verschillende browsers kunnen web pagina's in principe op een verschillende manier representeren. Kan dit verschil verklaard worden door het feit dat sommige browsers HTTP 1.0, en andere browsers HTTP 1.1 gebruiken? Leg uit. [2]
- *Antwoord: Nee*
Het verschil tussen HTTP 1.0 en HTTP 1.1 is vooral op het gebied van TCP connectie gebruik. De opmaak (dus representatie) van een pagina wordt niet door HTTP, maar door HTML bepaald

2) Elektronische post [6]

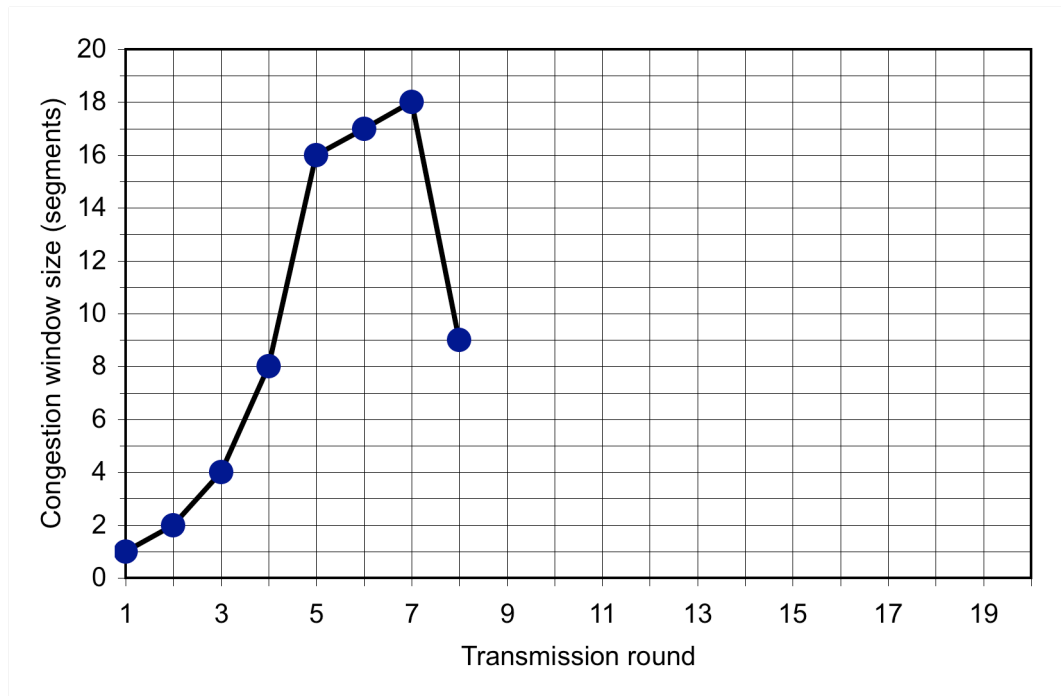
- a) Er zijn drie belangrijke email protocollen: SMTP, POP3 en IMAP. Zijn deze protocollen alternatieven van elkaar, of dienen ze verschillende doelen en moeten ze in combinatie gebruikt worden? Leg uit. [2]
- *SMTP is voor het versturen van email (tussen client en server, en tussen servers). POP en IMAP zijn "mail access protocollen", waarmee een client zijn mail kan ophalen bij de server. POP is een relatief simpel protocol, waarmee het niet mogelijk is mail op de server in folders te organiseren. Bij IMAP kan dat wel. Als je toch je mail wilt organiseren in folders, zul je bij POP alle mail lokaal moeten opslaan en organiseren. Je kunt dan niet met verschillende machines bij al je mail (want die staat immers lokaal op 1 machine).*
- b) Leg uit hoe het kan gebeuren dat, als ik een PDF file van 1 MByte als attachment verstuur, het email bericht beduidend groter dan 1 MByte kan worden. [2]
- *Email berichten worden MIME gecodeerd verstuurd; base64 is een voorbeeld van een dergelijke codering. Dit soort coderingen gebruiken niet alle 8 bits van een byte. Hierdoor zijn voor het gecodeerde bericht meer bytes nodig dan voor het oorspronkelijke attachment.*
- c) Is het mogelijk een mail server en een web server exact dezelfde DNS naam te geven, bijvoorbeeld "utwente.nl" (dus de DNS naam van de web server begint niet met "www" en de DNS naam van de mail server begint niet met "smtp")? Motiveer je antwoord. [2]
- *Dit is zeker mogelijk. DNS kent immers een apart RR (Resource Record) type voor mail servers (TYPE=MX). De web server kan dan van TYPE=A of TYPE=CNAME zijn.*

3) Peer-to-Peer [5]

- a) Wat is, in één woord, het sleutelvoordeel van peer-to-peer applicaties in vergelijking met klassieke client-server applicaties? [2]
- *schaalbaarheid.*
- b) Stel je hebt P2P software op je systeem geïnstalleerd. Hoe weet deze software, nadat je systeem is opgestart, met welke peers contact moet worden gemaakt? [1]
- *Dit is een typisch bootstrap probleem. Normaal gesproken zal het P2P programma, aan het einde van een sessie, een aantal peers in een file opslaan waarmee het contact heeft gehad. Voor het geval deze file verloren gaat, of voor het geval de software voor het eerst wordt gebruikt, zijn er een aantal peers vast gecodeerd in de software.*
- c) Wat is de relatie tussen trackers, torrents en chunks? [2]
- *Bij BitTorrent worden files in stukken van een bepaalde lengte (bv. 256 KB) opgehakt; dergelijke stukken heten chunks. Deze chunks worden in het algemeen op meerdere systemen gehost; de systemen die samen alle chunks van een file hosten, worden een torrent genoemd. De tracker houdt de administratie van chunks over de torrent bij.*

4) TCP [15]

Beschouw het volgende verloop van de grootte van het TCP *congestion window* als functie van de tijd. De gebruikte versie van TCP is TCP Reno (TCP met *fast retransmit* en *fast recovery*)



- a) Wordt na de 7e transmissieronde pakketverlies gedetecteerd door het ontvangen van 3 *duplicate ACKs* of door een *time-out*? [1]
- 3 *duplicate ACKs*
- b) Wat is de initiële waarde van de variabele *Threshold* (tijdens de 1ste transmissieronde)? [1]
- 16 *segmenten*
- c) Wat is de waarde van de variabele *Threshold* tijdens de 8e transmissieronde? [1]
- 9 *segmenten*

In de onderstaande vragen moet je aangeven hoe het verloop is van de grootte van het *congestion window* na transmissieronde 8. Hierbij is het gegeven dat direct na transmissieronde 15 pakketverlies wordt gedetecteerd door het optreden van een *time-out*. Voor de rest vindt er geen pakketverlies plaats.

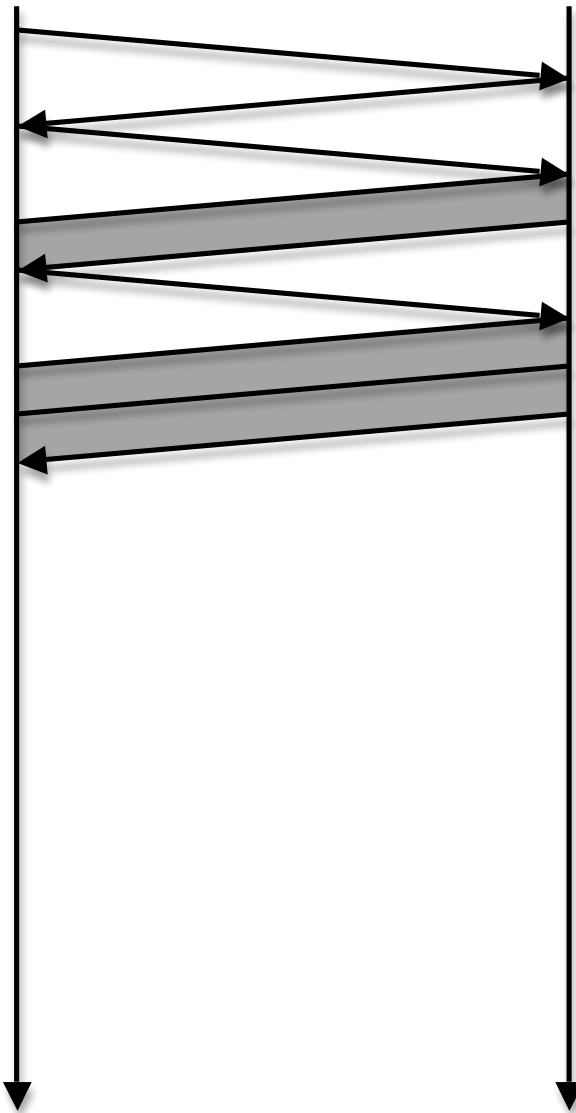
- d) Wat is de grootte van het *congestion window* tijdens transmissieronde 15? [1]
- 16 *segmenten*
- e) Wat is de grootte van het *congestion window* tijdens transmissieronde 16? [1]
- 1 *segment*
- f) Wat is de waarde van de variabele *Threshold* tijdens de 16e transmissieronde? [1]
- 8 *segmenten*
- g) Wat is de grootte van het *congestion window* tijdens transmissieronde 20? [2]
- 9 *segmenten*

In de rest van deze opgave beschouwen we de delay die TCP introduceert voor een webclient die een pagina van grootte F van een server haalt. Neem hierbij de ontwikkeling van het congestion window aan zoals die in bovenstaande grafiek beschreven staat. Hierbij gaan we er van uit dat TCP segmenten verstuurt met een grootte (MSS) van 1250 bytes (10^4 bits). De grootte van headers en een http request bericht veronderstellen we verwaarloosbaar klein. We gaan er van uit dat de webclient en server verbonden zijn door een enkele link met data rate $R = 10$ Mbps (10^7 bits/s). De roundtrip time van de link is $RTT = 2$ ms. Stel dat $F = 3750$ bytes.

h) Hoe lang duurt het totdat de webclient de gehele pagina heeft binnengehaald? Licht je antwoord toe met behulp van een time-sequence diagram en een formule. [3]

- *1 RTT voor openen TCP verbinding*
- *1 RTT + MSS/R voor 1ste segment*
- *1 RTT + 2 MSS/R voor 2e en 3e segment*

$$3 \cdot RTT + 3 \cdot \frac{MSS}{R} = 3 \cdot 0.002 + 3 \cdot \frac{10^4}{10^7} = 0.009 \text{ s}$$

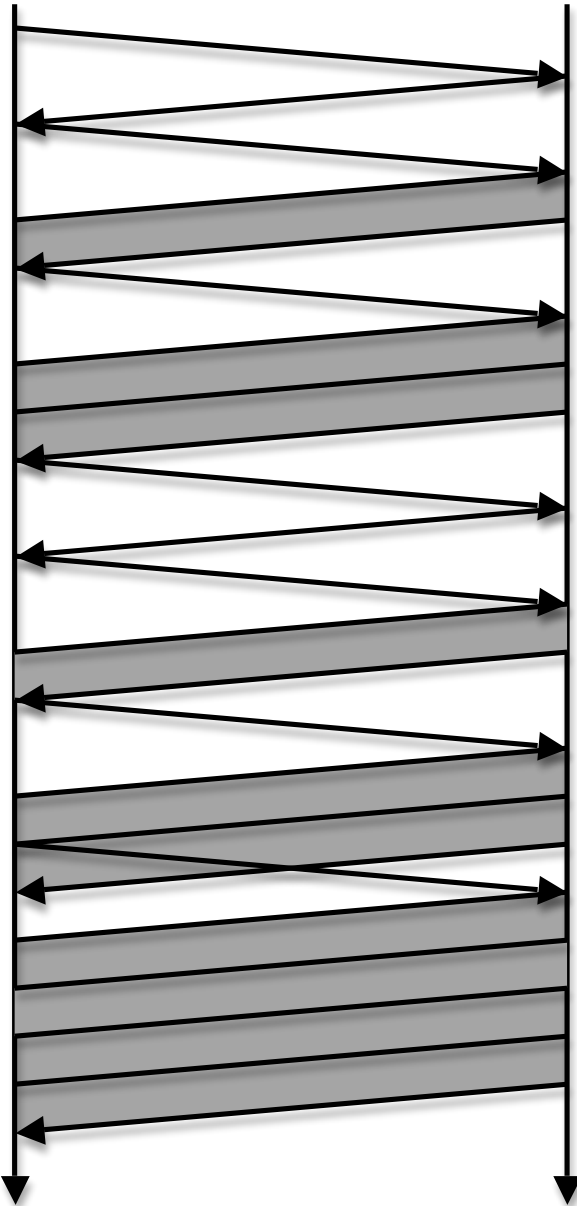


Stel nu dat de webpagina naast de HTML file (van grootte $F = 3750$ bytes) ook nog een plaatje van $G = 8750$ bytes bevat.

i) Hoe lang duurt het totdat de webclient de gehele pagina inclusief plaatje heeft binnengehaald in het geval HTTP met *non-persistent connections* wordt gebruikt? Licht je antwoord toe met behulp van een time-sequence diagram en een formule. [2]

- $3 \text{ RTT} + 3 \text{ MSS}/R$ voor HTML file
- 1 RTT voor openen TCP verbinding plaatje
- $1 \text{ RTT} + \text{MSS}/R$ voor 1e segment plaatje
- $1 \text{ RTT} + 2 \text{ MSS}/R$ voor 2e en 3e segment plaatje
- $1 \text{ RTT} - \text{MSS}/R + 4 \text{ MSS}/R$ voor 4e, 5e, 6e en 7e segment plaatje

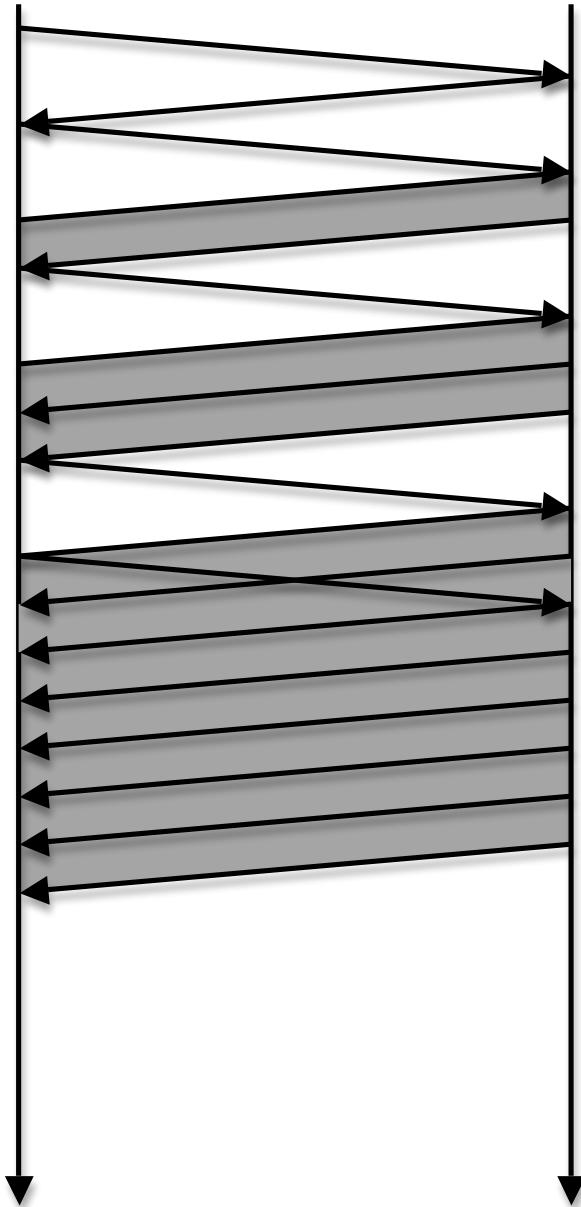
$$3 \cdot \text{RTT} + 3 \cdot \frac{\text{MSS}}{R} + 4 \cdot \text{RTT} + (7-1) \cdot \frac{\text{MSS}}{R} = 7 \cdot \text{RTT} + 9 \frac{\text{MSS}}{R} = 7 \cdot 0.002 + 9 \cdot \frac{10^4}{10^7} = 0.023 \text{ s}$$



j) Hoe lang duurt het totdat de webclient de gehele pagina inclusief plaatje heeft binnengehaald in het geval HTTP met *persistent connections* wordt gebruikt? Licht je antwoord toe met behulp van een time-sequence diagram en een formule. [2]

- *3 RTT + 3 MSS/R voor HTML file*
TCP verbinding is nog open:
1 RTT + 4MSS/R voor 1^e 4 segmenten plaatje
3 MSS/R voor 5e t/m 7e segment plaatje (RTT valt tijdens 1^e 4 segmenten)

$$3 \cdot RTT + 3 \cdot \frac{MSS}{R} + RTT + 7 \cdot \frac{MSS}{R} = 4 \cdot RTT + 10 \frac{MSS}{R} = 4 \cdot 0.002 + 10 \cdot \frac{10^4}{10^7} = 0.018 \text{ s}$$

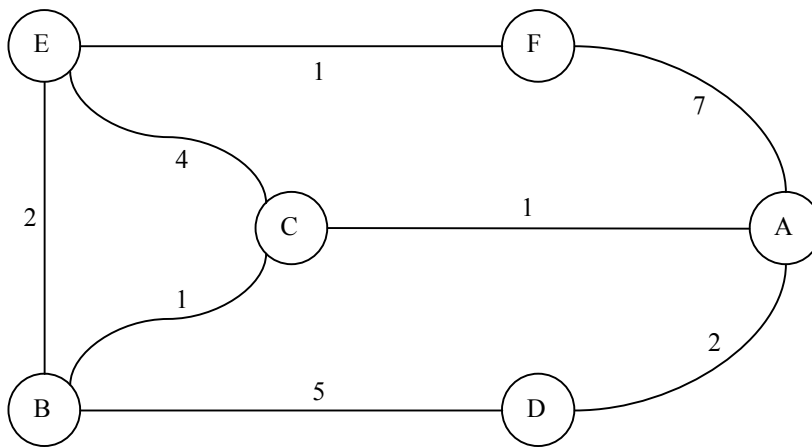


5) Routing [16]

- a) Er bestaan twee belangrijke routeringsalgoritmen: *distance vector* (DV) en *link state* (LS). Geef van de volgende beweringen aan, of deze waar zijn voor DV en voor LS (geen van beide of allebei is ook mogelijk), door in de lege vakjes ja of nee in te vullen: [7]

<i>Bewering</i>	<i>waar voor DV ?</i>	<i>waar voor LS ?</i>
het is een gedecentraliseerd routeringsalgoritme	<i>ja</i>	<i>nee</i>
het is een globaal routeringsalgoritme	<i>nee</i>	<i>ja</i>
het is een hiërarchisch routeringsalgoritme	<i>nee</i>	<i>nee</i>
het kan last hebben van "counting to infinity"	<i>ja</i>	<i>nee</i>
het vindt de goedkoopste paden	<i>ja</i>	<i>ja</i>
elke node weet de kosten van het gevonden beste pad naar elke andere node	<i>ja</i>	<i>ja</i>
elke node weet welke nodes er liggen op het gevonden beste pad naar elke andere node	<i>nee</i>	<i>ja</i>

Beschouw nu het volgende netwerk met de daarin aangegeven link-kosten:



- b) Gebruik Dijkstra's algoritme voor *link-state* routing om de kortste paden van node F naar alle andere nodes te berekenen. Gebruik het algoritme om de volgende tabel in te vullen, waarin N' de verzameling van al afgehandelde nodes is, $D(x)$ de afstand tot node x , en $p(x)$ de voorganger van x op het kortste pad van F naar x . [5]

step	N'	$D(A), p(A)$	$D(B), p(B)$	$D(C), p(C)$	$D(D), p(D)$	$D(E), p(E)$
0	{F}	7,F	∞	∞	∞	1,F
1	{FE}	7,F	3,E	5,E	∞	
2	{FEB}	7,F		4,B	8,B	
3	{FEB C}	5,C			8,B	
4	{FEB C A}				7,A	
5	{FEB C A D}					

Neem nu aan dat er in ditzelfde netwerk een *distance-vector* algoritme wordt gebruikt **zonder** “*poisoned reverse*”. Ga er voor de volgende twee subvragen vanuit dat het algoritme geconvergeerd is.

c) Geef de *distance-vector* die node F aan node E stuurt. [2]

- $\mathcal{D}_F(A) = 5, \mathcal{D}_F(B) = 3, \mathcal{D}_F(C) = 4, \mathcal{D}_F(D) = 7, \mathcal{D}_F(E) = 1, \mathcal{D}_F(F) = 0$
(or $\mathcal{D}_F = [5, 3, 4, 7, 1, 0]$)

Neem nu aan dat er in ditzelfde netwerk een *distance-vector* algoritme wordt gebruikt **met** “*poisoned reverse*”.

d) Geef de *distance-vector* die node F aan node E stuurt. [2]

- $\mathcal{D}_F(A) = \infty, \mathcal{D}_F(B) = \infty, \mathcal{D}_F(C) = \infty, \mathcal{D}_F(D) = \infty, \mathcal{D}_F(E) = \infty, \mathcal{D}_F(F) = 0$
(or $\mathcal{D}_F = [\infty, \infty, \infty, \infty, \infty, 0]$)

6) Multiple Access [15]

In deze vraag beschouwen we verschillende *multiple access* protocollen: *channel partitioning*, Slotted Aloha en *polling*.

- a) Op welke twee manieren kan het kanaal worden onderverdeeld, in het geval van *channel partitioning*? [2]
- *in frequentie en in tijd*
- b) Beschrijf de basiswerking van slotted Aloha. [2]
- *tijd is verdeeld in sloten, en een transmissie van een frame start altijd vanaf het begin van een slot, en duurt een slot.*
 - *wanneer een node een nieuw frame wil versturen doet deze dat in het eerstvolgende slot. Wanneer er een collision optreedt wordt, zolang het frame niet succesvol verzonden is, het frame in ieder volgende slot opnieuw verstuurd met kans p*
- c) Wat is het essentiële verschil tussen een *polling* protocol en een *token passing* protocol? [2]
- *bij polling geeft een master node andere nodes de beurt om te zenden*
 - *bij token passing wordt een token rondgestuurd om te bepalen welke node mag zenden: de node die het token heeft (ontvangen) mag een pakket versturen, en stuurt daarna het token door naar een volgende node*

Beschouw een *broadcast* kanaal met $N = 10$ nodes (voor *polling* is er ook nog een extra *polling* node) en een *data rate* van $R = 10^7$ bits/s. De grootte van een slot (in het geval van *channel partitioning* en slotted Aloha) of het maximum aantal bits wat verstuurd mag worden als een node *gepold* wordt (in het geval van *polling*), is $Q = 10^6$ bits. Voor het *polling* protocol is de *polling delay* (de hoeveelheid tijd die verstrijkt tussen het afronden van een transmissie (van maximaal Q bits) en de start of transmissie van de volgende node), $d_{poll} = 10$ ms.

Stel dat slechts één node data te versturen heeft, terwijl de andere $N-1$ nodes geen data te versturen hebben.

d) Wat is de maximum *throughput* van het *broadcast* kanaal wanneer *channel partitioning* wordt gebruikt? Leg je antwoord uit. [2]

- *de actieve node krijgt steeds 1 uit N sloten*

$$\frac{R}{N} = \frac{10^7}{10} = 10^6 \text{ bits/s}$$

e) Wat is de maximum *throughput* van het *broadcast* kanaal wanneer slotted Aloha wordt gebruikt? Leg je antwoord uit. [2]

- *de actieve node krijgt alle capaciteit want er zijn geen collisions*

$$R = 10^7 \text{ bits/s}$$

f) Wat is de maximum *throughput* van het *broadcast* kanaal wanneer *polling* wordt gebruikt? Leg je antwoord uit. [2]

- *het versturen van een frame van Q bits duurt de transmissietijd (Q/R) + de tijd om alle nodes te pollen (N · d_{poll})*

$$\frac{Q/R}{Q/R + N \cdot d_{poll}} \cdot R = \frac{Q}{Q/R + N \cdot d_{poll}} = \frac{10^6}{10^6/10^7 + 10 \cdot 0.01} = \frac{10^6}{0.2} = 5 \cdot 10^6 \text{ bits/s}$$

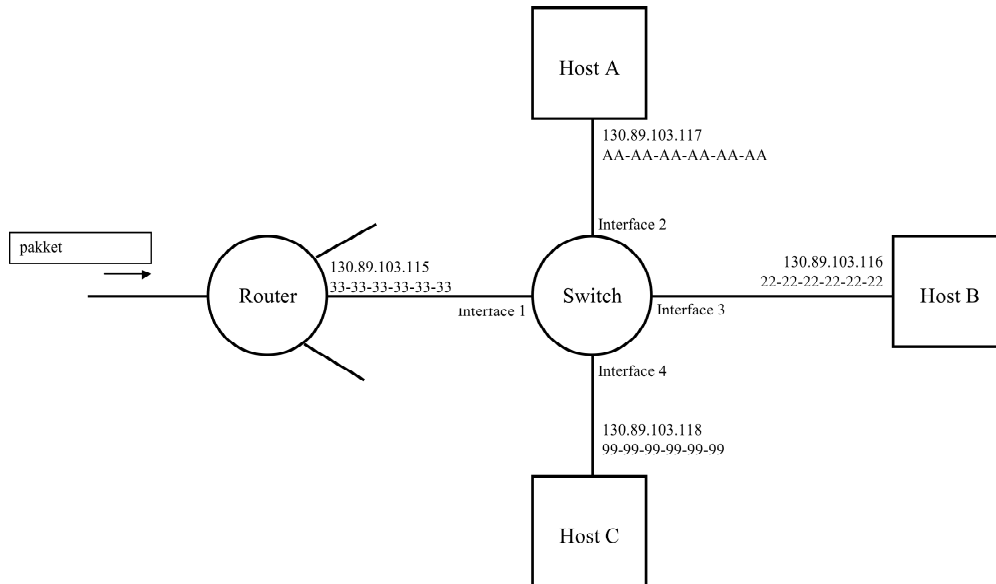
Neem nu aan dat **meerdere** nodes data te versturen hebben.

g) Leg voor ieder van de 3 genoemde *multiple access* protocollen uit hoe welke invloed dit heeft of the *throughput* van het *broadcast* kanaal (zoals gevonden in respectievelijk d, e en f). Blijft deze gelijk, stijgt deze, of daalt deze? Leg je antwoord uit. (Opm.: je hoeft geen getal of formule te geven). [3]

- *channel partitioning:*
de throughput stijgt lineair met het aantal actieve nodes omdat meer sloten per timeframe benut worden
- *slotted Aloha:*
de throughput daalt omdat collisions zullen gaan optreden en het kanaal soms leeg zal blijven omdat nodes slechts met kans p een hertransmissie doen
- *polling:*
de throughput zal stijgen omdat per transmissie minder tijd aan polling verspild wordt

7) Switches and routers [13]

Beschouw het netwerk in de onderstaande figuur. Een *router* is met één van zijn vier interfaces met een *switch* verbonden, die verder met Host A, Host B en Host C verbonden is. De *interfaces* van de *switch* zijn genummerd, en voor ieder van de met de *switch* verbonden nodes is het MAC adres en het IP adres gegeven.



Gedurende deze gehele vraag zullen we een pakket met *destination* adres 130.89.103.116 bekijken (met bestemming Host B), dat in de *router* aankomt.

a) Schrijf het *destination* adres in binaire notatie. [1]

- *10000010.01011001.01100111.01110100*

Stel dat de *forwarding* tabel van de router de volgende *entries* bevat:

Destination Address	Link Interface
130.89.103.0/24	1
130.89.103.64/28	2
130.89.103.114/29	3
0.0.0.0/0	4

b) Met welke *entry/entries* heeft het *destination* adres een *match*? [2]

- *de entries die verwijzen naar link interface 1, 3 en 4*

c) Welke *link interface* zal worden gekozen om het pakket naar door te sturen? Waarom? [2]

- *3, omdat deze entry de langste prefix heeft (/29), en longest prefix match gebruikt wordt*

Laten we nu aannemen dat het pakket is doorgestuurd naar de *link interface* die verbonden is met de *switch*. Stel dat de ARP tabel voor deze interface (van de *router*) een *entry* bevat die relevant is voor het genoemde pakket. Neem verder aan dat de *switch* tabel (van de *switch*) initieel leeg is.

d) Welke kolommen kun je vinden in een ARP tabel? [2]

- *IP address, MAC address, TTL*

e) Op welke *interface(s)* zal het pakket door de *switch* verstuurd worden (nadat het is aangekomen van de *router*)? [1]

- *2,3 en 4*

f) Hoe zal de *switch* tabel van de *switch* er uit zien nadat het pakket doorgestuurd is naar Host B? Gebruik onderstaande tabel om de *entries* in te vullen. (de kolom tijd is weggelaten en hoeft niet ingevuld te worden). [2]

Address	Interface
<i>33-33-33-33-33-33</i>	<i>1</i>

We beschouwen nogmaals dezelfde situatie als in deelvraag e) en f), maar nu nemen we aan dat ook de ARP tabel initieel leeg is.

g) Op welke *interface(s)* zal het pakket door de *switch* verstuurd worden (nadat het is aangekomen van de *router*)? [1]

- *3*

h) Hoe zal de *switch* tabel van de *switch* er uit zien nadat het pakket doorgestuurd is naar Host B? Gebruik onderstaande tabel om de *entries* in te vullen. (de kolom tijd is weggelaten en hoeft niet ingevuld te worden). [2]

Address	Interface
<i>22-22-22-22-22-22</i>	<i>3</i>
<i>33-33-33-33-33-33</i>	<i>1</i>