Solutions Networking 2

R3

3. The process which initiates the communication is the client; the process that waits to be contacted is the server.

R12

12. When the user first visits the site, the server creates a unique identification number, creates an entry in its back-end database, and returns this identification number as a cookie number. This cookie number is stored on the user's host and is managed by the browser. During each subsequent visit (and purchase), the browser sends the cookie number back to the site. Thus the site knows when this user (more precisely, this browser) is visiting the site.

R18

18. The HOL blocking problem is a large file in the webpage (i.e. video clip) will take a long time to pass through the bottleneck link such that it blocks other small objects behind it. HTTP/2 solves this problem by breaking each message into small frames, and interleave the request and response messages on the same TCO connection.

P5

Problem 5

a) The status code of 200 and the phrase OK indicate that the server was able to locate the document successfully. The reply was provided on Tuesday, 07 Mar 2008 12:39:45 Greenwich Mean Time.

b) The document index.html was last modified on Saturday 10 Dec 2005 18:27:46 GMT.

c) There are 3874 bytes in the document being returned.

d) The first five bytes of the returned document are : <!doc. The server agreed to a persistent connection, as indicated by the Connection: Keep-Alive field

P10 (Bog løsninger forkerte, se korrigerede svar nedenfor)

Problem 10

The total download time is:

a. $2 \cdot 100$ ms $+ 8 \cdot 10^3$ bits / 10^6 bits/s $+ 5 \cdot (2 \cdot 100$ ms $+ 4 \cdot 10^5$ bits / 10^6 bits/s) = 3.208 s

b. $2 \cdot 100$ ms $+ 3 \cdot (4 \cdot 10^5$ bits / 10^6 b[illegible]s) = 1.4 s

c. $2 \cdot 100$ ms $+ 4 \cdot 10^5$ bits / 1[illegible]

d. $2 \cdot 100$ ms $+ 8 \cdot 10^3$ bits / 10^6 bits/[illegible]5 $\cdot (4 \cdot 10^5$ bits / 10^6 bits/s) = 2.208 s

Given: RTT=0.1s, transferrate=10^6 bits/s, main document=$8*10^3$ bits, image = $8*50*10^3$ bits=$4*10^5$ bits

- Transfertime(1k byte)= $8*10^3$ bits / 10^6 b/s= 0,008sek
- Transfertime(50k byte)=0,4 sek

a) We need to download the main-page first: : (2*RTT +transfertime(1kbytes))=0.208
+5*(2*RTT+transfertime(50k bytes)) = 5*(0.2+0.4)=3 sek

Ialt 3.208

b) 2 parallel connections

We need to download the main-page first: : (2*RTT +transfertime(1kbytes)) = 0.208
+
5 images on 2 parallel connections require 3 series of https (img1+2, img3+4, img5):

2*(2*RTT+transfertime(2*50kbytes)) = 2*(0.2+0.8)=2 sek
+
1*(2*RTT+transfertime(1*50kbytes)) = 0.2+0.4= 0.6 sek

Ialt 2.808

C) 6 parallel connections
We need to download the main-page first: (2*RTT +transfertime(1kbytes)) = 0.208
+
2*RTT + 5*transfertime(50 kbyte) = 0.2+2 sek

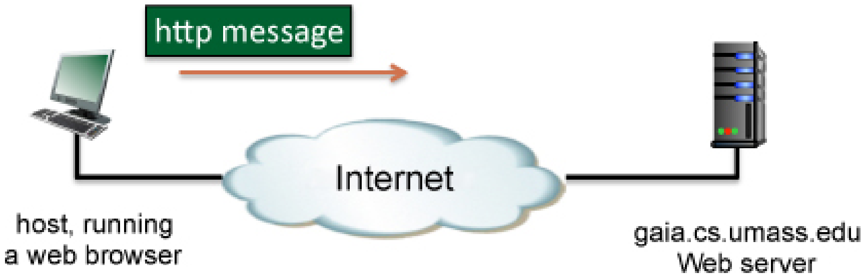
I alt: 2.408

D) One persistent connection with pipelining
We need to download the main-page first: (2*RTT +transfertime(1kbytes)) = 0.208
+RTT (5 requests+responses handled as 1)+5*transfertime(50 kbyte) = 0.1+2 sek

I alt 2.308

The HTTP GET message

Consider the figure below, where a client is sending an HTTP GET message to a web server, gaia.cs.umass.edu.



Suppose the client-to-server HTTP GET message is the following:

```
GET /kurose_ross/interactive/quotation6.htm HTTP/1.0
Host: gaia.cs.umass.edu
If-Modified-Since: Mon, 23 Mar 2020 00:23:34 -0700
```

Answer the following questions:

- What is the name of the file that is being retrieved in this GET message?
- What version of HTTP is the client running?
- Does the client already have a (possibly out-of-date) copy of the requested file? Explain. If so, approximately how long ago did the client receive the file, assuming the GET request has just been issued?

Solution:

The file being fetched is /kurose_ross/interactive/quotation6.htm

- The client is running HTTP version 1.0.
- The time indicated in the browser's If-Modified-Since header field is approximately 30 minutes ago, indicating that is has a cached copy. Therefore the server will only send a copy of the requested URL in response to this HTTP GET message if the server-side copy has been changed in the last 30 minutes.

http://gaia.cs.umass.edu/kurose_ross/interactive/DNS_HTTP_delay.php
Løsning online

P22

For calculating the minimum distribution time for client-server distribution, we use the following formula:

$$D_{cs} = max\ \{NF/u_s,\ F/d_{min}\}$$

Similarly, for calculating the minimum distribution time for P2P distribution, we use the following formula:

$$D_{P2P} = max\{F/u_s,\ F/d_{min},\ NF/(u_s + \sum_{i=1}^{N} u_i)\}$$

Where, F = 20 Gbits = 20 * 1024 Mbits
u_s = 30 Mbps
$d_{min} = d_i$ = 2 Mbps

Note, 300Kbps = 300/1024 Mbps.

Client Server

		N		
		10	**100**	**1000**
	300 Kbps	10240	68266.67	682666.67
u	**700 Kbps**	10240	68266.67	682666.67
	2 Mbps	10240	68266.67	682666.67

Peer to Peer

		N		
		10	**100**	**1000**
	300 Kbps	10240	34538.67	63412
u	**700 Kbps**	10240	20821.33	28700
	2 Mbps	10240	10240	10240

Problem 22

For calculating the minimum distribution time for client-server distribution, we use the following formula:

$$D_{cs} = max\ \{NF/u_s,\ F/d_{min}\}$$

Similarly, for calculating the minimum distribution time for P2P distribution, we use the following formula:

$$D_{P2P} = max\{F/u_s,\ F/d_{min},\ NF/(u_s + \sum_{i=1}^{N} u_i)\}$$

Where, F = 15 Gbits = 15 * 1024 Mbits
u_s = 30 Mbps
$d_{min} = d_i$ = 2 Mbps

Note, 300Kbps = 300/1024 Mbps.

Client Server

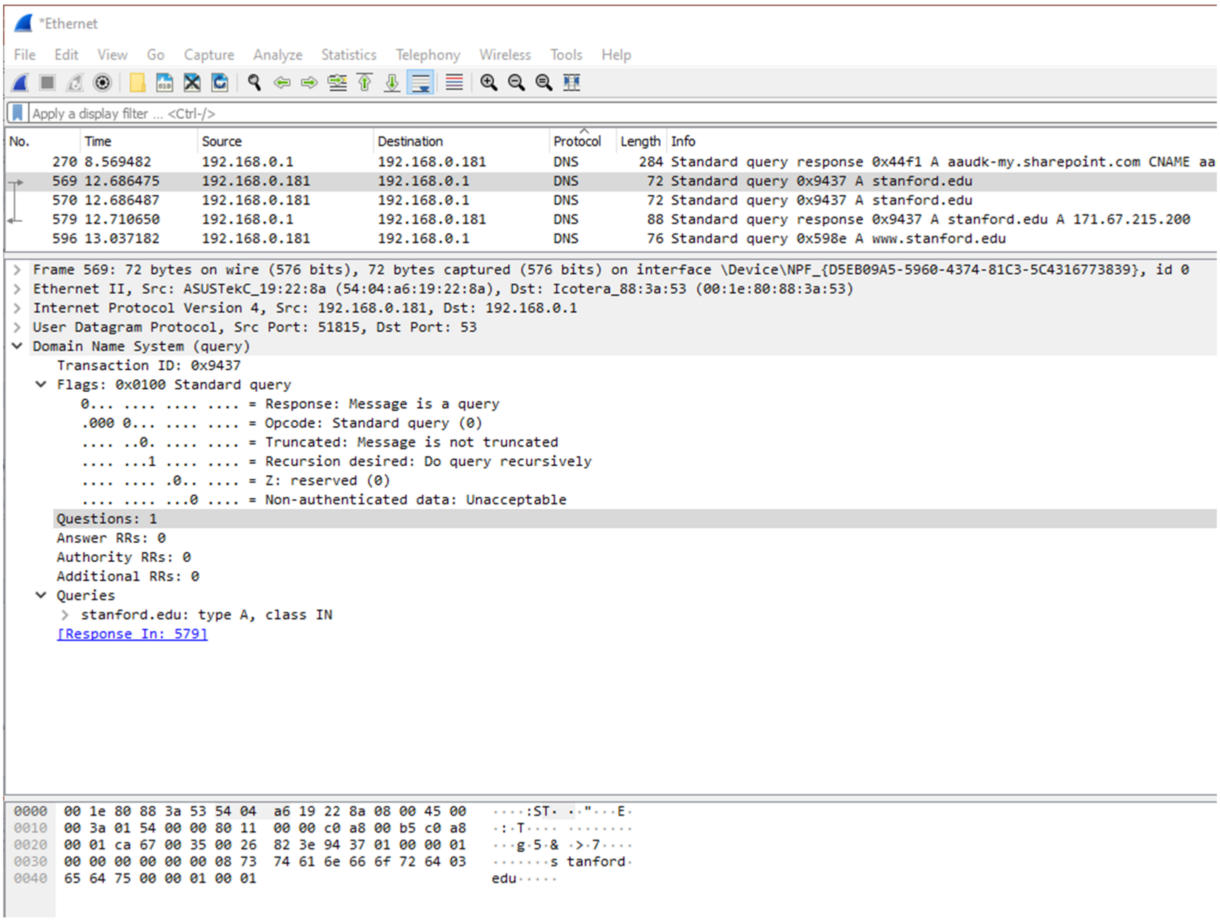
		N		
		10	**100**	**1000**
	300 Kbps	7680	51200	512000
u	**700 Kbps**	7680	51200	512000
	2 Mbps	7680	51200	512000

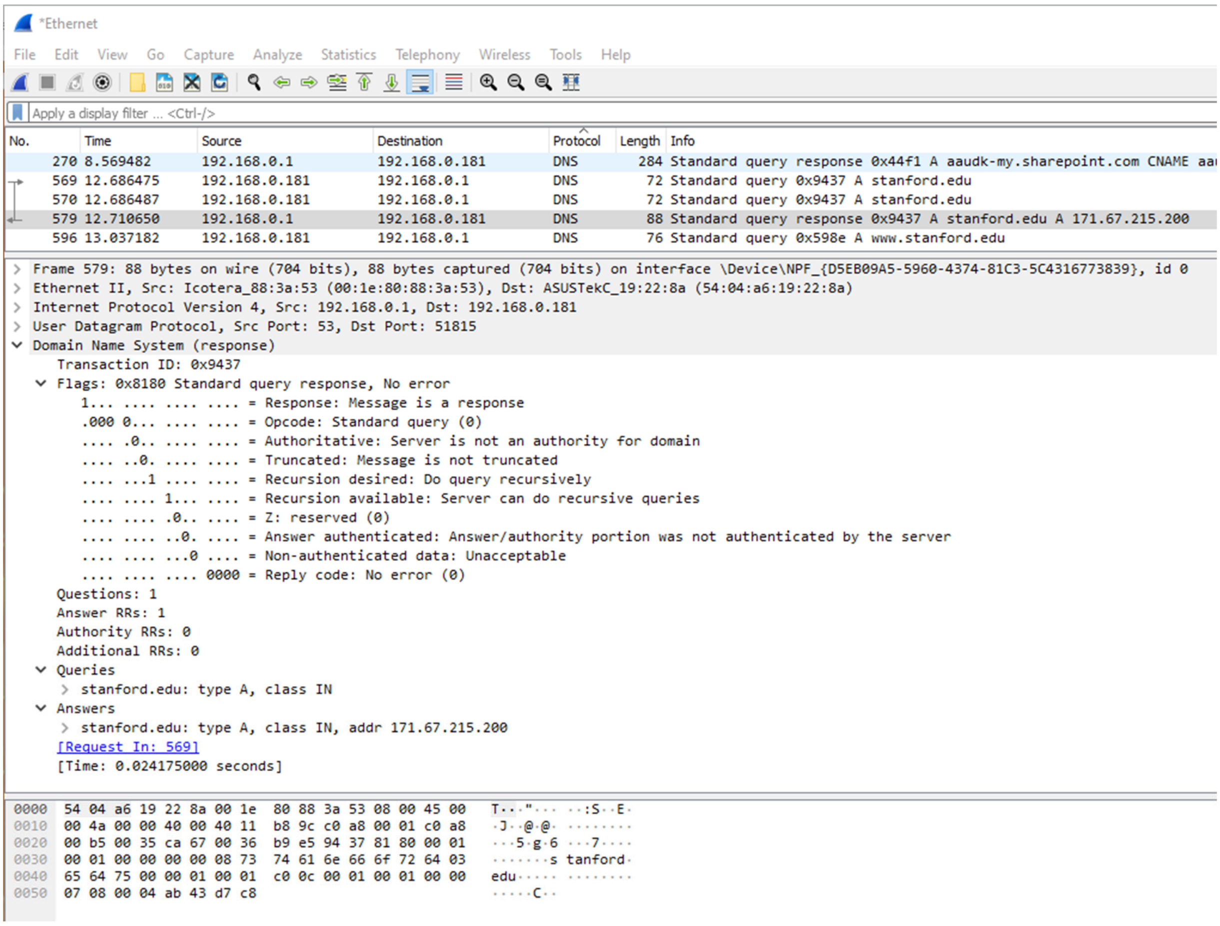
Peer to Peer

		N		
		10	**100**	**1000**
	300 Kbps	7680	25904	47559
u	**700 Kbps**	7680	15616	21525
	2 Mbps	7680	7680	7680

Practice

Eksempel (se også slides/video)


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
No. Time Source Destination Protocol Length Info
270 8.569482 192.168.0.1 192.168.0.181 DNS 284 Standard query response 0x44f1 A aaudk-my.sharepoint.com CNAME aa
569 12.686475 192.168.0.181 192.168.0.1 DNS 72 Standard query 0x9437 A stanford.edu
570 12.686487 192.168.0.181 192.168.0.1 DNS 72 Standard query 0x9437 A stanford.edu
579 12.710650 192.168.0.1 192.168.0.181 DNS 88 Standard query response 0x9437 A stanford.edu A 171.67.215.200
596 13.037182 192.168.0.181 192.168.0.1 DNS 76 Standard query 0x598e A www.stanford.edu
Frame 569: 72 bytes on wire (576 bits), 72 bytes captured (576 bits) on interface \Device\NPF_{D5EB09A5-5960-4374-81C3-5C4316773839}, id 0
Ethernet II, Src: ASUSTekC_19:22:8a (54:04:a6:19:22:8a), Dst: Icotera_88:3a:53 (00:1e:80:88:3a:53)
Internet Protocol Version 4, Src: 192.168.0.181, Dst: 192.168.0.1
User Datagram Protocol, Src Port: 51815, Dst Port: 53
Domain Name System (query)
Transaction ID: 0x9437
Flags: 0x0100 Standard query
0... .... = Response: Message is a query
.000 0... .... = Opcode: Standard query (0)
.... ..0. .... = Truncated: Message is not truncated
....1 .... = Recursion desired: Do query recursively
.... .0.. .... = Z: reserved (0)
....0 .... = Non-authenticated data: Unacceptable
Questions: 1
Answer RRs: 0
Authority RRs: 0
Additional RRs: 0
Queries
stanford.edu: type A, class IN
[Response In: 579]
0000 00 1e 80 88 3a 53 54 04 a6 19 22 8a 08 00 45 00
0010 00 3a 01 54 00 00 80 11 00 00 c0 a8 00 b5 c0 a8
0020 00 01 ca 67 00 35 00 26 82 3e 94 37 01 00 00 01
0030 00 00 00 00 00 00 08 73 74 61 6e 66 6f 72 64 03
0040 65 64 75 00 00 01 00 01
s tanford
edu


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
No. Time Source Destination Protocol Length Info
270 8.569482 192.168.0.1 192.168.0.181 DNS 284 Standard query response 0x44f1 A aaudk-my.sharepoint.com CNAME aa
569 12.686475 192.168.0.181 192.168.0.1 DNS 72 Standard query 0x9437 A stanford.edu
570 12.686487 192.168.0.181 192.168.0.1 DNS 72 Standard query 0x9437 A stanford.edu
579 12.710650 192.168.0.1 192.168.0.181 DNS 88 Standard query response 0x9437 A stanford.edu A 171.67.215.200
596 13.037182 192.168.0.181 192.168.0.1 DNS 76 Standard query 0x598e A www.stanford.edu
Frame 579: 88 bytes on wire (704 bits), 88 bytes captured (704 bits) on interface \Device\NPF_{D5EB09A5-5960-4374-81C3-5C4316773839}, id 0
Ethernet II, Src: Icotera_88:3a:53 (00:1e:80:88:3a:53), Dst: ASUSTekC_19:22:8a (54:04:a6:19:22:8a)
Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.181
User Datagram Protocol, Src Port: 53, Dst Port: 51815
Domain Name System (response)
Transaction ID: 0x9437
Flags: 0x8180 Standard query response, No error
1... .... = Response: Message is a response
.000 0... .... = Opcode: Standard query (0)
.... .0.. .... = Authoritative: Server is not an authority for domain
.... ..0. .... = Truncated: Message is not truncated
....1 .... = Recursion desired: Do query recursively
.... 1... .... = Recursion available: Server can do recursive queries
.... .0.. .... = Z: reserved (0)
....0. .... = Answer authenticated: Answer/authority portion was not authenticated by the server
....0 .... = Non-authenticated data: Unacceptable
.... 0000 = Reply code: No error (0)
Questions: 1
Answer RRs: 1
Authority RRs: 0
Additional RRs: 0
Queries
stanford.edu: type A, class IN
Answers
stanford.edu: type A, class IN, addr 171.67.215.200
[Request In: 569]
[Time: 0.024175000 seconds]
0000 54 04 a6 19 22 8a 00 1e 80 88 3a 53 08 00 45 00
0010 00 4a 00 00 40 00 40 11 b8 9c c0 a8 00 01 c0 a8
0020 00 b5 00 35 ca 67 00 36 b9 e5 94 37 81 80 00 01
0030 00 01 00 00 00 00 08 73 74 61 6e 66 6f 72 64 03
0040 65 64 75 00 00 01 00 01 c0 0c 00 01 00 01 00 00
0050 07 08 00 04 ab 43 d7 c8
s tanford
edu