Broc Nickodemus

P4. Consider the following string of ASCII characters that were captured by

Wireshark when the browser sent an HTTP GET message (i.e., this is the actual

content of an HTTP GET message). The characters <cr><lf> are carriage

return and line-feed characters (that is, the italized character string <cr> in

the text below represents the single carriage-return character that was contained

at that point in the HTTP header). Answer the following questions,

indicating where in the HTTP GET message below you find the answer.

GET /cs453/index.html HTTP/1.1<cr><lf> Host: gai

a.cs.umass.edu<cr><lf> User-Agent: Mozilla/5.0 (

Windows;U; Windows NT 5.1; en-US; rv:1.7.2) Gec

ko/20040804 Netscape/7.2 (ax) <cr><lf> Accept:ex

t/xml, application/xml, application/xhtml+xml, text

/html;q=0.9, text/plain;q=0.8,image/png,\*/\*;q=0.5

<cr><lf> Accept-Language: en-us,en;q=0.5<cr><lf> Accept-

Encoding: zip,deflate<cr><lf> Accept-Charset: ISO

-8859-1,utf-8;q=0.7,\*;q=0.7<cr><lf> Keep-Alive: 300<cr>

<lf> Connection:keep-alive<cr><lf><cr><lf>

a. What is the URL of the document requested by the browser?

The url is gaia.cs.umass.edu/cs453/index.html

b. What version of HTTP is the browser running?

The version is HTTP 1.1

c. Does the browser request a non-persistent or a persistent connection?

The browser requests a persistent connection indicated by Keep-Alive

d. What is the IP address of the host on which the browser is running?

You cannot tell the IP address of the host from the HTTP GET

e. What type of browser initiates this message? Why is the browser type needed in an HTTP request message?

Mozilla 5.0 The browser type is needed because the server needs to know what browser is interpreting the data so it can send send it accordingly.

P5. The text below shows the reply sent from the server in response to the HTTP

GET message in the question above. Answer the following questions, indicating

where in the message below you find the answer.

HTTP/1.1 200 OK<cr><lf> Date: Tue, 07 Mar 2008

12:39:45GMT<cr><lf> Server: Apache/2.0.52 (Fedora)

<cr><lf>Last-Modified: Sat, 10 Dec2005 18:27:46

GMT<cr><lf> ETag: “526c3-f22-a88a4c80”<cr><lf> Accept-

Ranges: bytes<cr><lf> Content-Length: 3874<cr><lf>

Keep-Alive: timeout=max=100<cr><lf> Connection:

Keep-Alive<cr><lf> Content-Type: text/html; charset=

ISO-8859-1<cr><lf><cr><lf> <!doctype html public “-

//w3c//dtd html 4.0 transitional//en”><lf> <html><lf>

<head><lf> <meta http-equiv=”Content-Type”

content=”text/html; charset=iso-8859-1”><lf> <meta

name=”GENERATOR” content=”Mozilla/4.79 [en] (Windows NT

5.0; U) Netscape]”><lf> <title>CMPSCI 453 / 591 /

NTU-ST550A Spring 2005 homepage</title><lf> </head><lf>

<much more document text following here (not shown) >

a. Was the server able to successfully find the document or not? What time

was the document reply provided?

Yes, OK indicates it connected. Tue, 07 Mar 2008 12:39:45GMT

b. When was the document last modified?

Sat, 10 Dec2005 18:27:46GMT

c. How many bytes are there in the document being returned?

Content-Length: 3874 bytes

d. What are the first 5 bytes of the document being returned? Did the server

agree to a persistent connection?

The first 5 bytes returned are <!doc Yes, the server agreed to keep the connection alive.

P9. Consider Figure 2.12, for which there is an institutional network connected to

the Internet. Suppose that the average object size is 850,000 bits and that the

average request rate from the institution’s browsers to the origin servers is 16

requests per second. Also suppose that the amount of time it takes from when

the router on the Internet side of the access link forwards an HTTP request

until it receives the response is three seconds on average (see Section 2.2.5).

Model the total average response time as the sum of the average access delay

(that is, the delay from Internet router to institution router) and the average

Internet delay. For the average access delay, use Δ /(1 – Δß ), where Δ is the

average time required to send an object over the access link and ß is the

arrival rate of objects to the access link.

a. Find the total average response time.

request rate = 16 requests/ second

average access delay = Δ /(1 – Δß)

Δ = L/R = 850,000 bits/ 100mb/sec\*16sec (1,600,000,000bits/sec) = 0.000531

ß = 3 seconds

Average Access Delay = (0.000531 seconds)/1 – (3sec)(0.000531sec) = 0.000531 seconds

Total Average Response Time = Average Access Delay + Internet Delay = 3sec + 0.000531sec = 3.000531

b. Now suppose a cache is installed in the institutional LAN. Suppose the miss rate is 0.4. Find the total response time.

miss rate = 0.4

ß = 16 requests/sec \* (1-0.04) = 9.6 requests/sec

Average Access Delay = (0.000531sec)/(1 – 9.6 \* 0.000531) = 0.00533 sec

Total Average Response Time = 0.000533 + 3 sec = 3.000533 sec

P22. Consider distributing a file of F = 15 Gbits to N peers. The server has an upload

rate of us = 30 Mbps, and each peer has a download rate of di = 2 Mbps and an

upload rate of u .

For N = 10, 100, and 1,000 and u = 300 Kbps, 700 Kbps, and

2 Mbps, prepare a chart giving the minimum distribution time for each of

the combinations of N and u for both client-server distribution and P2P

distribution.

Dcs = maxb {NF/us, F/dmin}

client server

|  |  |  |  |
| --- | --- | --- | --- |
| u | 10 (term n) | 100 (term n) | 1000 (term n) |
| 300 Kbps | 7680 | 51200 | 512000 |
| 700 Kbps | 7680 | 51200 | 512000 |
| 2Mbps | 7680 | 51200 | 512000 |

Dp2p = max{F/us, F/dmin, NF/(us ∑ ui)}

peer to peer

|  |  |  |  |
| --- | --- | --- | --- |
| u | 10 (term n) | 100 (term n) | 1000 (term n) |
| 300 Kbps | 7680 | 25904 | 47559 |
| 700 Kbps | 7680 | 15616 | 21525 |
| 2Mbps | 7680 | 7680 | 7680 |