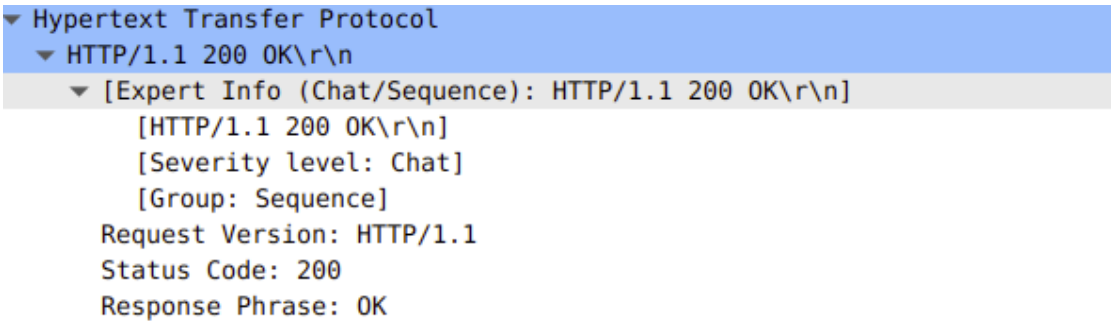
Lab 1

Exercise 3

Question 1:

What is the status code and phrase returned from the server to the client browser?

Answer:



Status code: 200

Response Phrase: OK

Question 2:

When was the HTML file that the browser is retrieving last modified at the server? Does the response also contain a DATE header? How are these two fields different?

Answer:

Last modified date: Tuesday, 23/9/2003 05:29:00 GMT

Yes, the response contains a Date header.

Difference:

Date header is the time when the response is made and sent.

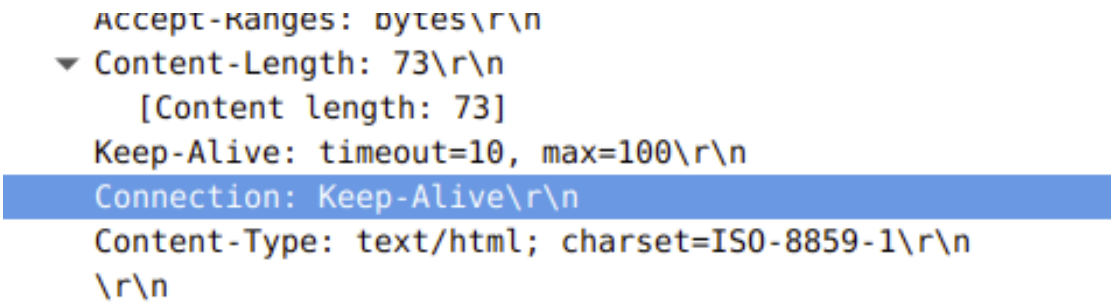
Last modified time is the time when the html file is last modified.

Question 3:

Is the connection established between the browser and the server persistent or non-persistent? How can you infer this?

Answer:

The connection between the browser and the server is persistent.



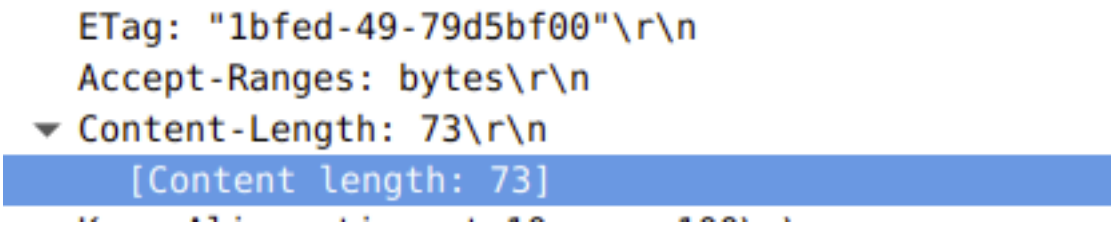
Connection status: keep-alive

Question 4:

How many bytes of content are being returned to the browser?

Answer:

73 bytes are being returned to the browser.

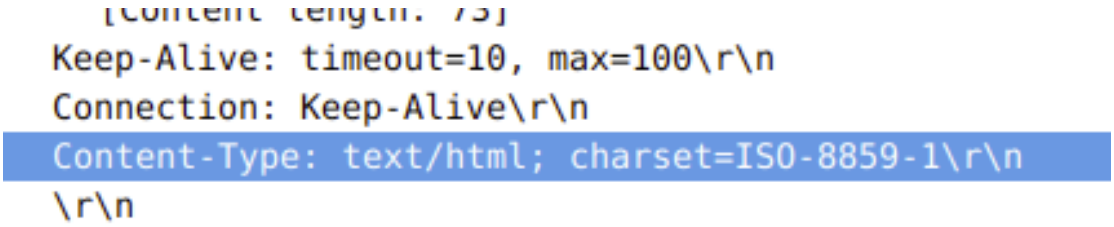


Question 5:

What is the data contained inside the HTTP response packet?

Answer:

Html text is contained inside the HTTP response packet.



Exercise 4

Question 1:

Inspect the contents of the first HTTP GET request from the browser to the server. Do you see an “IF-MODIFIED-SINCE” line in the HTTP GET?

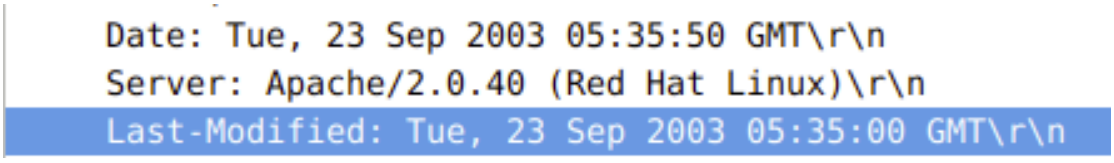
Answer:

No, there is no “IF-MODIFIED-SINCE”line in the http get.

Question 2:

Does the response indicate the last time that the requested file was modified?

Answer:



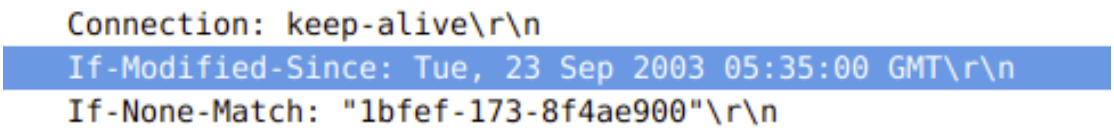
Last modified date: Tuesday, 23/09/2003 05:35:00 GMT

Question 3:

Now inspect the contents of the second HTTP GET request from the browser to the server. Do you see an “IF-MODIFIED-SINCE:” and “IF-NONE-MATCH” lines in the HTTP GET? If so, what information is contained in these header lines?

Answer:

Yes, there are “IF-MODIFIED-SINCE” and “IF-NONE-MATCH” lines in the http get.



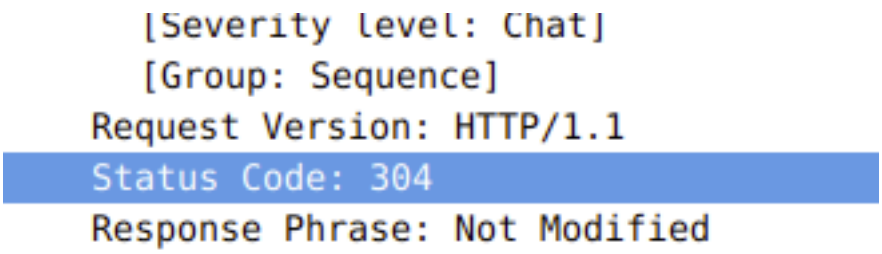
“If-Modified-Since” contains the last modified date: Tuesday, 23/09/2003 05:35:00 GMT

“If-None-Match” contains the value of Etag.

Question 4:

What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain.

Answer:



Status code: 304

Response phrase: Not Modified

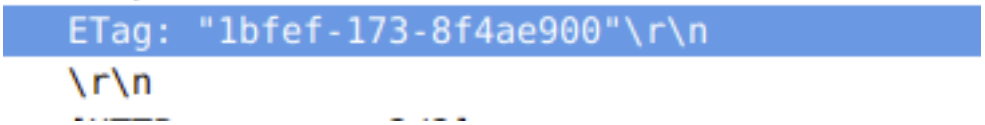
No, the server does not explicitly return the contents of the file.

Reason: If the response code is 304, it means that the file is the same as the one when it was last modified, so the response will not return any content of the file and the browser will directly use the disk cache.

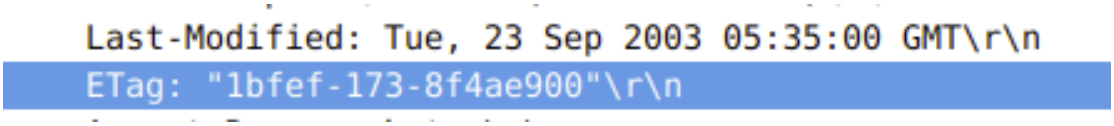
Question 5:

What is the value of the Etag field in the 2nd response message and how it is used? Has this value changed since the 1 response message was received?

Answer:



The value of the Etag field in the 2nd response message: 1bfef-173-8f4ae900

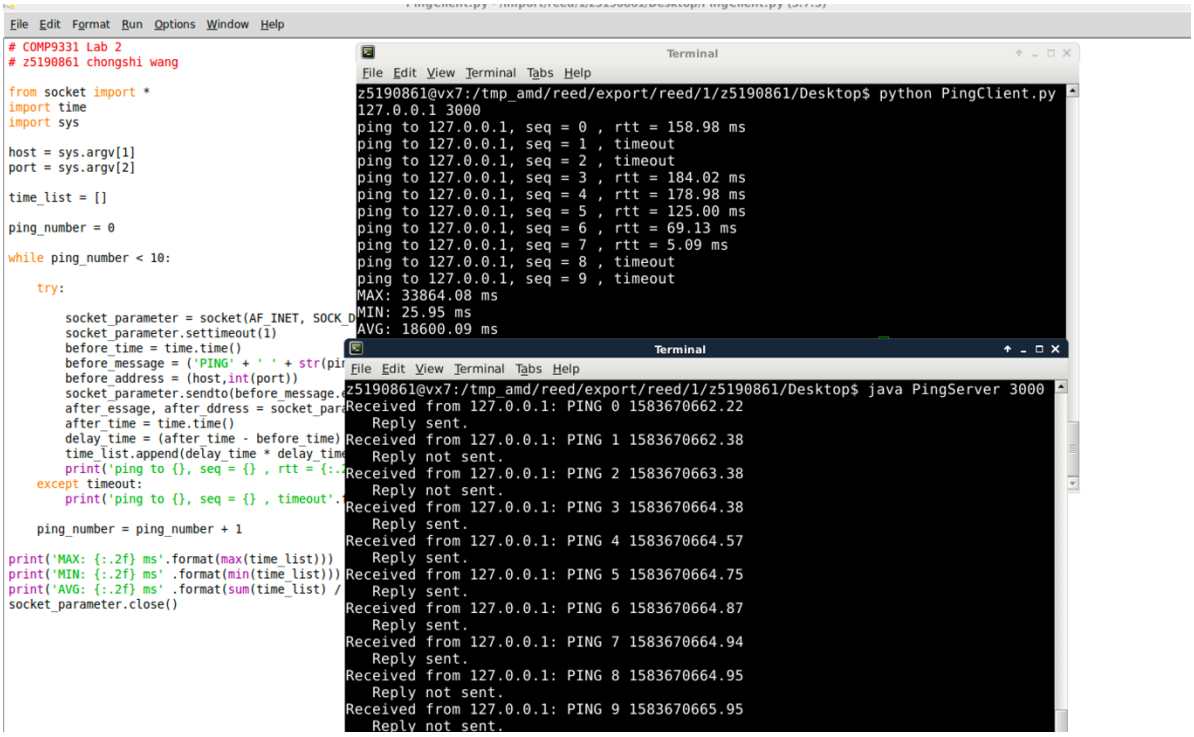


The value of the Etag field in the 1st response message: 1bfef-173-8f4ae900

Etag is used to represent the version of resources. When sending a request, the browser asks the server if this version of resources is valid. If it is valid, the browser will directly use the disk cache.

It is the same as the second value.

Exercise 5


File Edit Format Run Options Window Help
# COMP9331 Lab 2
# z5190861 chongshi wang
from socket import *
import time
import sys
host = sys.argv[1]
port = sys.argv[2]
time_list = []
ping_number = 0
while ping_number < 10:
try:
socket_parameter = socket(AF_INET, SOCK_D
socket_parameter.settimeout(1)
before_time = time.time()
before_message = ('PING' + ' ' + str(pi
before_address = (host,int(port))
socket_parameter.sendto(before_message.
after_essage, after_ddress = socket_para
after_time = time.time()
delay_time = (after_time - before_time)
time_list.append(delay_time * delay_time
print('ping to {}, seq = {} , rtt = {:.2
except timeout:
print('ping to {}, seq = {} , timeout'.
ping_number = ping_number + 1
print('MAX: {:.2f} ms'.format(max(time_list)))
print('MIN: {:.2f} ms' .format(min(time_list)))
print('AVG: {:.2f} ms' .format(sum(time_list) /
socket_parameter.close()
Terminal
File Edit View Terminal Tabs Help
z5190861@vx7:/tmp_amd/reed/export/reed/1/z5190861/Desktop$ python PingClient.py
127.0.0.1 3000
ping to 127.0.0.1, seq = 0 , rtt = 158.98 ms
ping to 127.0.0.1, seq = 1 , timeout
ping to 127.0.0.1, seq = 2 , timeout
ping to 127.0.0.1, seq = 3 , rtt = 184.02 ms
ping to 127.0.0.1, seq = 4 , rtt = 178.98 ms
ping to 127.0.0.1, seq = 5 , rtt = 125.00 ms
ping to 127.0.0.1, seq = 6 , rtt = 69.13 ms
ping to 127.0.0.1, seq = 7 , rtt = 5.09 ms
ping to 127.0.0.1, seq = 8 , timeout
ping to 127.0.0.1, seq = 9 , timeout
MAX: 33864.08 ms
MIN: 25.95 ms
AVG: 18600.09 ms
Terminal
File Edit View Terminal Tabs Help
z5190861@vx7:/tmp_amd/reed/export/reed/1/z5190861/Desktop$ java PingServer 3000
Received from 127.0.0.1: PING 0 1583670662.22
Reply sent.
Received from 127.0.0.1: PING 1 1583670662.38
Reply not sent.
Received from 127.0.0.1: PING 2 1583670663.38
Reply not sent.
Received from 127.0.0.1: PING 3 1583670664.38
Reply sent.
Received from 127.0.0.1: PING 4 1583670664.57
Reply sent.
Received from 127.0.0.1: PING 5 1583670664.75
Reply sent.
Received from 127.0.0.1: PING 6 1583670664.87
Reply sent.
Received from 127.0.0.1: PING 7 1583670664.94
Reply sent.
Received from 127.0.0.1: PING 8 1583670664.95
Reply not sent.
Received from 127.0.0.1: PING 9 1583670665.95
Reply not sent.