lab2.md

CS3331 Lab 2: HTTP and UDP Sockets

Exercise 3 - Using Wireshark to understand basic HTTP request/response messages

http-ethereal-trace-1

Question 1

The status code is 200 and the response phrase is 'OK'.

Question 2

The HTML page the browser is retrieving was last modified on Tue, 23 Sep 2003 05:29:00 GMT. The response also contains a `DATE` header. This stores the date the request was made rather than the last modified date.

Question 3

The connection between the browser and the server appears to be persistent as the `Connection` field is set to `Keep-Alive` which indicates the connection will not be dropped between the browser and the server after the HTTP request is made and responded to but instead will stay open until either the browser or the server drops the connection.

Question 4

The data contained inside the HTTP response packet is of `text/html` type.

www.bbc.co.uk

Question 1

The status code is 200 and the response phrase is 'OK'.

Question 2

It is not clear when the webpage was last modified on as the `Last-Modified` field is not set in the HTTP respoonse.

Question 3

The connection is persistent as indicated by the `Connection` field being set to `Keep-Alive` .

Question 4

The data contained inside the HTTP response packet is of `text/html` type.

Exercise 4

Question 1

No, there is not an `If-Modified-Since` field in the HTTP GET request.

Question 2

Yes, the response indicates the last time the requested page was modified was Tue, 23 Sep 2003 05:35:00 GMT.

Question 3

Yes, there appears to be an `If-Modified-Since` field in the HTTP GET request. The field stores the following date value: Tue, 23 Sep 2003 05:35:00 GMT.

Question 4

The second HTTP GET request returned a 304 status code and the response phrase `Not-Modified`. The server did not explicitly return the contents of the file since the `If-Modified-Since` field is used to determine whether the version of the file already stored in the web cache has been updated since it was cached, and since it hasn't, the GET reponse returns with no explicit content body and just lets the client know it hasn't been modified.

Exercise 5: Implementing a ping server

```
# Retrieve command line args library.
import sys
# Socket programming library.
import socket
# Retrieve system time library.
from datetime import datetime

# Store input server parameters - host and port to ping to.
host = sys.argv[1]
port = sys.argv[2]
serverAddr = (host, int(port))
bufferSize = 2048

# Create a UDP socket instance using IPv4 addresses (AF_INET) and
# TCP protocol (SOCK_DGRAM).
s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

# Ping the server.
for i in range(0, 9):
    # Save current time stamp.
    timestamp = datetime.now()
    # Store ping message to send to server.
    message = "ping to %s, seq = %d, timestamp = %s" % (host, i, str(timestamp))
    try:
        # Wait for up to one second for a reply.
        s.settimeout(1)
        # Send message to the server.
        sent = s.sendto(message, serverAddr)
```

```
            # Store reply from server.
            reply = s.recv(bufferSize)
            # Calculate rtt.
            delta = datetime.now() - timestamp
            rtt = delta.microseconds
            print "ping to %s, seq = %d, rtt = %d ms" % (host, i, rtt)
        except:
            continue

    # Close the socket.
    s.close()
```