

**Computer Science and Engineering Department, SVNIT, Surat**  
**B.tech.-III, Semester-V**  
**Computer Networks (CS303)**  
**Assignment - 4**  
**APPLICATION LAYER**

## **SOCKET PROGRAMMING**

1. Create a basic Web server that can only handle one request. In more detail, when a client (browser) contacts your Web server, it will i) create a connection socket; (ii) receive the HTTP request from this connection; (iii) parse the request to identify the particular file being requested; (iv) get the requested file from the server's file system; (v) create an HTTP response message consisting of the requested file preceded by header lines; and (vi) send the response over the TCP connection to the requesting browser. A "404 Not Found" error message ought to be returned by your server whenever a browser requests a file that isn't on your server.
2. Create a client ping application. The process involves having your client send a straightforward ping message to a server, get a pong message in return, and then calculate the time difference between the two. The Round Trip Time is the name of this lag (RTT). The functionality offered by the client and server is comparable to that of the common ping software found in contemporary OS systems.
3. Develop a straightforward mail client that can send email to any recipient. Your client must first open a TCP connection with a mail server (like a Google mail server), use the SMTP protocol to communicate with it, send an email to a recipient (like your buddy) using the mail server, and then shut the TCP connection.
4. Create a Web proxy. Your proxy creates a fresh HTTP request for the same item and sends it to the origin server when it gets an HTTP request for an object from a browser. The proxy prepares a new HTTP response and delivers it to the client when it gets the matching HTTP response from the origin server that contains the object.

## **HTTP**

1. Give brief details about HTTP. What is the difference between HTTP and HTTPS?
2. Write down the steps to capture HTTP request packets for the following URL.  
URL: <http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html>
3. Answer the following questions for the above URL request.
  - a) Which version your browser and server are running on?

- b) What is the IP address of your host machine and server?
- c) List out the languages accepted by your browsers.
- d) What is the status code returned from the server to your browser?
- e) What is the size of the content received from the server?
- f) Check the last modification date of the retrieved HTML file.
- g) Did you receive the content of the file as a response?
- h) Did you receive the content of the file if you requested the same HTML file?

## DNS

Apply *nslookup* on the following URL and answer the following questions related to the DNS.

URL: [www.mit.edu](http://www.mit.edu)

1. Are DNS queries sent and received using TCP or UDP?
2. What is the destination port of the DNS query and source port of the DNS response?
3. What is the IP address of the DNS query message? Verify the IP address of the local DNS server using *ipconfig*.
4. What is the “Type” of the DNS query sent?