BITS, Pilani - Hyderabad Campus

CS F303 (Computer Networks)

Second Semester 2022-23

Lab Sheet 3

1. **Revisiting nslookup**
   a) Recall nslookup from the first lab sheet. Read the man page for nslookup.
      i. What options are to be use to query a local DNS server for an NS record?
         [hint: -type=NS]
      ii. If your computer is using the localhost as a default DNS server then you need to disable it for this lab. Follow these steps for Ubuntu: https://askubuntu.com/a/907249
2. **Analysing DNS request and response**
   a) On wireshark start a capture on your current interface filter for your ip and for dns traffic
      [hint: ip.addr==172.16.82.22 and dns]
   b) Simultaneously run ‘nslookup bits-pilani.ac.in’ on a terminal
      i. Locate the DNS query message on wireshark
         1. What is the IP address of the DNS server being queried? What port is the DNS service running on the DNS server?
         2. What is the id of the query?
         3. How many questions are there in the query?
         4. What record type is being requested in the query? What is the class of the name in the query?
         5. Has recursion been desired by the client?
      ii. Locate the DNS response message to the above query on wireshark
         1. What is the id of the response? Does the id match the query?
         2. Is the response from an authoritative server?
         3. Is recursive quering supported by the server?
         4. How many answers have been provided by the server?
         5. How many records of other authoritative servers are included in the answer?
      iii. Repeat parts (i) and (ii) for the command ‘nslookup -type=mx bits-pilani.ac.in’

3. **Introducing socket programming in C**

   a) What is a socket in Computer Networks?

   b) Read the man pages of the socket, bind, listen and accept functions

   c) Download the following sample programmes, understand what they are doing, compile and run them. Satisfy yourself that you understand how the programmes work

      i. Client: https://drive.google.com/file/d/1t0Z1xzISisjcB8BPWmXlNw3e7S4WctA7/view?usp=share_link

      ii. Server: https://drive.google.com/file/d/17TZ22bxYmOYbqZZ_3ypc5aqq30uJcJ2b/view?usp=share_link

   d) The above programmes are designed in a way that the server accepts one request and terminates. Modify the programme so that the server continues to accept requests without terminating

   e) The modification in part (d) above is for a single-threaded server, so if multiple clients request simultaneously then the server will not be able to serve them. Modify the programme to create a multi-threaded server (recall OS) where every client is handled in a separate thread and the main thread is ready to receive connections

4. Write a http client programme using sockets. The programme should take the name of the server, the port and the object to fetch as command line arguments. Fetch the object over the socket and save it on disk. Open the file to verify the content. [hint: wireshark is your friend]

   a) Example: `./http_client info.cern.ch 80 /`