

ASSIGNMENT 1

School of Computing, National University of Singapore

Learning Objectives

This assignment/project provides you a chance to learn about popular network, transport and application layer protocols on the Internet (User data Gram Protocol-UDP, Transmission Control Protocol-TCP, Hyper-Text Transmission Protocol-HTTP, Internet Control Message Protocol-ICMP) and to develop services using socket programming.

A) Network Tools [20 marks]

1. (5 marks) DNS queries
 - i. (2 points) Run “**dig** www.anuflora.com” from any linux based system to find the **IP** address of the server www.anuflora.com. [Give the **screen-shot** showing the output]. State the IP address of the server (2 marks). Is the server hosted inside NUS? (1 mark)
 - ii. (2 points) Use dig to find the **IP** addresses of **mail servers** of the domain comp.nus.edu.sg.
2. (5 marks) Use **telnet** or **openssl** to connect to our SMTP server (smtp.gmail.com or smtp.comp.nus.edu.sg or smtp.nus.edu.sg) other SMTP servers where you have an account. Follow the SMTP commands as given in the IVLE->SMTP-demo to send an email to your yahoo/gmail/hotmail account. Give the **screen-shot** showing the list of commands (including telnet/openssl connection line) you have used to send the mail successfully.

Tip1: NUS SMTP and POP servers require TLS/SSL. Hence, instead of Telnet, you can use “openssl”. Alternatively, you can connect to other SMTP and/or POP servers.

```
$openssl s_client -connect smtp.gmail.com:465 -crlf -ign_eof
```

Tip2: Command for base64 encoding:

```
$echo -n "your text to encode" | base64  
or use online tools like: https://www.base64encode.org/
```

3. (5 marks) Use **telnet** or **openssl** to connect to our POP server (pop.nus.edu.sg) or other POP servers and access your mailbox by sending raw POP commands/messages. Retrieve the latest mail from your mailbox [Give **screen shot** showing your communication with POP server]. --- (Refer Appendix A for raw POP commands).

4. (5 marks) Use **telnet ftp.ietf.org 21** (or any other FTP server; you can use **openssl** instead of telnet for secured connection) to connect to our ftp server and retrieve the list of files and folder inside root or any other folder. How many TCP connections are required to retrieve a file (directory listing text file or any other file)? [Give **screen shot** showing your communication with FTP server. Include both control connection and data connection].

B) Socket programming (TCP)– Level 1 [10 marks]

1. (10 marks) Develop a TCP client program using the basic c/c++ socket library to construct a http request and retrieve the file “yourip.php” from the url:
<http://www.varlabs.org/yourip.php> [need to handle redirect/frameset] or
<http://varlabs.comp.nus.edu.sg/tools/yourip.php>

Parse the contents and display your current public IP address from the file returned by the server “varlabs”.

The output should be in the following format:

My public IP address is a.b.c.d

[Note: You should do the assignment by creating the proper HTTP messages as per the RFCs and using basic Socket library.]

C) Socket programming (TCP) – Level 2 [40 marks]

Develop a "Mini FTP client" using basic c/c++ socket library. "Mini FTP client" is a FTP client which implements a subset of the ftp command set. Your program should be able to communicate with ftp servers throughout the Internet to retrieve or upload files as per the defined requirements.

Mini FTP client Requirements

1. The Mini-FTP client shall allow a user to perform the following functions:
 - o Connect to an FTP server
 - o Print the remote working directory
 - o Change the remote working directory
 - o List all files in the remote working directory
 - o Upload/download text files to/from the server
2. Your C/C++ code is required to post FTP commands. As such, in order to implement the above functions, at least the following FTP commands must be used:
 - o PWD
 - o CWD
 - o LIST
 - o RETR <filename> (note:- simple text files only, from/to current folder)

- STOR <filename> (note:- simple text files only, from/to current folder)
 - QUIT
3. The user shall interact with the Mini-FTP client using a text-based menu system that may look as the following:

```

*****
*****Mini-FTP client*****
*****
1. Connect to FTP server
2. Print working directory
3. Change working directory
4. List all files
5. Upload file
6. Download file
7. Quit
Enter option (1-7) : _

```

The client program should continuously accept user commands and process them in a loop, as shown with the pseudocode below, until the user selects the 'Quit' option.

```

do{
    display MENU;
    accept option;
    process option;
    display results;
    wait for user to press any key;
}while(option != Quit)

```

4. Use TCP connections for all client/server communication.
5. If the server requires login, then send the username and password using the commands USER and PASS after connecting.
6. Use "extended passive mode" to open data connection using EPSV command to put the server in passive mode and to get the port number from the server for data connection. Create the data connection automatically from the FTP client program. (You can also use PASV)
7. All command replies (for example, "200 Command okay") from the server should be printed as part of the results. No need to parse the response and take actions.
8. Well written (good abstraction, error checking, readability) and well-commented code.
9. **You should not use any available libraries that provide the "ftp client functionality or implementation of ftp commands".**
10. Additional notes:

- To understand how the commands work, you can use TELNET or OPENSLL to connect to a FTP server (to port 21) and try out the commands. Also, you can login the TCP messages using packet sniffers such as WireShark. (Refer RFC 959, RFC 1579 & RFC 2428)
- To test commands and application, you can use ftp.ietf.org FTP server (any other free FTP servers). This server accepts anonymous login. (USER <anonymous> & PASS <anonymous>). Alternatively, you can use any open source FTP server software (recommended -

<https://security.appspot.com/vsftpd.html> for Unix-based systems and FileZilla-
<https://filezilla-project.org/> for Windows based systems) and set up your own FTP server.

- For the demo, your client should work with any one of these ftp servers. [Your client need NOT work with all FTP server].
- TCP Connections (buffer/delay) – If you write a message to a TCP socket, it may be buffered for a while before sending to the recipient. To force it to send immediately, you can use *flush()* method of the stream object.

D) Socket programming (RAW) – Level 3 [30 marks]

1. (25 marks) Write a c/c++ program using the basic c/c++ socket library to check your NAT router's (NUSNET NAT) behavior with respect to ICMP error messages. The program should have a UDP client module to send request to any host outside your NAT router (eg. outside campus). Request for a TCP/UDP service that is most possibly not running in the destination host. This will result in "port unreachable" ICMP error message. (Sometimes it may also result in "host unreachable". It is Ok for this assignment). You can also set TTL value to a small number to generate "host unreachable" message. Your program should have a RAW socket to retrieve this ICMP error message. The payload of the ICMP error message should contain IP header and first 8 bytes of IP payload.
 - i. [18 marks] Your program should display the source IP address from the IP header that is found in the payload of the ICMP error message.
 - ii. [3 marks] Is the IP address retrieved in part (a) your PC's/laptop's IP address? Is it private or public IP address?
 - iii. [4 marks] Describe the behavior of your NAT router with respect to ICMP error messages when compared to UDP/TCP reply messages? What are the advantages and disadvantages of this behavior? [Answer for part 'c' should be short and should not have more than 5 lines]

Note:

- Firewalls at your laptop or router may block ICMP error messages. You may need to turn-off firewall to receive ICMP error messages.
- WINDOWS users can install VM ("VMWARE workstation Player" or Virtualbox) and then install Ubuntu/Linux on top of the VM to get Linux environment for the assignment.
- If you are using a VM to run your code, you have to turn on 'bridged connection' otherwise the ICMP packets won't be passed into the VM.
- To check – You can use tools like Wireshark (www.wireshark.org) to check whether your laptop/VM can receive ICMP error message.

Submission:

- All submissions should be done in respective folders of IVLE before due date.

Submission Date 1: (For Section A and B) - Friday 21-Sep-2018

Submit **one RAR file** (change filename to your *Student Number*) containing **one** word-docx (or PDF) for Part-A and **one** (.c or .cpp) file for Part-B to 'IVLE->Assignment1-PartAB-

Submission'. In addition to solutions for Part A, the word-docx may also include instructions for compiling/running your source code.

Submission Date 2: For Section C. - Monday 01-Oct-2018

Submit **one RAR file** (change filename to your *Student Number*) containing **one** word-docx (or PDF) and **one** (.c or .cpp) file to 'TVLE->Assignment1-PartC-Submission'.

Submission Date 3: For Section D. - Monday 08-Oct-2018

Submit **one RAR file** (change filename to your *Student Number*) containing **one** (.c or .cpp) file and **one** word-docx (or PDF) with instructions for compiling/running your source code to 'TVLE->Assignment1-PartD-Submission'.

If you have any question/clarification, discuss through **piazza.com** under the folder 'Assignment 1'.

Bhojan Anand /NUS

'Students who approach education from a **deep-learning perspective** make significant improvements, remember what they learn, and feel empowered to make a difference. Those who take a surface approach to learning may get good grades, but they rarely benefit much in the long term'. – *from several research findings.*

Learning in depth is the key focus of CS3103. Try more beyond the questions above.

[We have a Zero Tolerance for Plagiarism Policy. If you are here only for marks/grades, just let me the grade/mark you prefer to have!]

More on deep learning approach ...

Students Who Take a Deep Approach--

- *Attempt to understand material for themselves*
 - *Seek rigorous and critical interaction with knowledge content*
 - *Relate ideas to previous knowledge and experience*
 - *Discover and use organizing principles to integrate ideas*
 - *Relate evidence to conclusions*
 - *Examine the logic of arguments*
-