

CSIS0234B Computer and Data Communication (Class B)

Tutorial 1

POP3 client

In this tutorial, you will write a simple POP3 client to communicate with an E-mail server, obtaining the total number and size of messages in your mail box, using the socket interface.

1. POP3 Protocol

The POP3 protocol allows one to login to a mail-server, check whether there are E-mails in your mailbox, and retrieve and delete E-mails there. POP3 runs on the well-known TCP port number 110. A dialog is started by the server, giving a message that it is an intended server. From then on, the client will make commands into the server, which executes the action and provide responses. All commands are specified in a line terminated by CR-LF (i.e., ASCII 13 followed by ASCII 10). To make it possible to finish the tutorial in time, we will only use a few selected commands in the POP3. In particular, we will use the following 4 commands:

USER *login*

Specify a login id, which mailbox is manipulated in the connection. Expect a boolean response (see below). This should be the first command used on a POP3 connection.

PASS *password*

Specify the password. Expect a boolean response. This should be the second command used. Once USER and PASS command succeeds, the connection is in a “transaction” state, where other commands can be used.

STAT

Ask for the status of the mailbox. The response looks like +OK 2 1000, which tells that 2 messages of 1000 bytes in total are in the mailbox.

QUIT

Terminate the connection. Expect a boolean response.

A boolean response is either +OK (positive) or -ERR (negative), both may be followed by some descriptive message to tell what is happening. A POP3 dialog looks like this:

```
+OK POP3 study v2001.78 server ready
USER tmchan
+OK User name accepted, password please
PASS abcd1234
+OK Mailbox open, 6 messages
STAT
+OK 6 9848
QUIT
+OK Sayonara
```

You can test all these by hand, using `telnet study.csis.hku.hk 110`. (Here `study.csis.hku.hk` is our POP3 server. If you do that, you are advised to first login virtue and change the password on virtue temporarily, since it will be echoed on the screen. The `telnet` program don't know part of the dialog is a password, so it won't hide it.) We will try to write a program that does exactly that.

2. Your task

Write a C or C++ program that performs the interaction. It should ask for a login and password. Then it should connect to the POP3 server and use the USER, PASS, STAT and QUIT commands, in that sequence. If any of them fails, print a message, terminate the connection, and exit. After the STAT command returns, print the number of messages in the mailbox and the total size that is responded by the server. The program thus runs like this:

```
> ./pop3client
Login: tmchan
Password: (hidden)
Mailbox: 6 9848
```

Here are some hints:

- Use `getpass()` in `<unistd.h>` to get the password without echoing it. See the man page.
- You will need to read message from the server, which is terminated by a CR-LF sequence. One way to do it is to read the socket character by character, and stop when a LF character is found:

```
void read_response(int fd, char buf[1024]) {
    char ch;
    int len = 0;
    for (;;) {
        if (read(fd, &ch, 1) == -1)    /* read chars one by one */
            die("Read");
        if (ch == '\n')
            break;
        if (len < 1023)                /* don't overflow the buffer */
            buf[len++] = ch;
    }
    if (buf[len-1] == '\r')
        --len;
    buf[len] = 0;
}
```

- To know whether a response is positive or negative, you only need to check the first character of it.
- When an error occur, you can print an error message from the 5th character of the response (like `printf("%s\n", response+5)`). Similarly, to print the two numbers needed, you can print from the 4th character of the response.