CSCI537
Introduction to Distributed Computing

Report on

RPCgen A3

Dept. of Computer & Information Science,
IUPUI

Name: Keyur Kirti Mehta

Contents

1. Introduction

The assignment is to implement different operations using the RPCgen library. This library will be implemented in C programming language which will make remote procedure calls to the server to perform the operation.

The assignment is implemented using multi-threaded server so as multiple client can be connected and request for file transmission.

2. Design Decisions and Implementation

1. RPCgen

RPCgen tool generates the code in C programming language which implements the RPC protocol. This involves 5 components as:

- Client – It calls the client stub. This is local call made by the client. It sends the parameters along with the call.
- Client Stub – Stub consolidates the parameters together called as marshalling and sends to the runtime.
- RPC Communication Package (aka RPCRuntime) – This will send the call to the server machine. It actually transfer the call to the server stub.
- Server Stub – It unpacks the parameters sends by the client and send the request to actual server. This is also called as unmarshalling.
- Server – The actual processing of the request take place on the server. And it follows the same process to send response back to the client.

RPCgen generates header file which will be common for bother client and server code. The client and server uses this header file for the code. The XDR file defines the data structure mentioned in the header file. It will only be generated when different data structure is used. The client stub and server stub is used to marshal and unmarshal the parameters and actual call. The client code makes the call using the client stub for the server code.

In order to generate the stubs, the .x file with all the signature of the functions and data structure used in it needs to be created. Once the .x file is created, then execute the rpcgen –C –a <.x> command in order to generate all the files. It will create header file, XDR file, client and server stubs and client and server code template.

2. Operation

The assignment implements 5 operations using RPC which are as follows:

1. Host Name:

This will return the host name of the server to the client. It will return the string with name of the server which is running.

2. **MergeSort:**

This operation will accepts 2 arrays from the user and marshal them in the structure type. This is passed to the server. Server will merge and sort the bother the integer list and return the sorted 1 list back to the client. I have implemented the insertion sort to sort the list of integer in ascending order.

3. Encrypted Echo:

This operation will takes the string as an input from the client and encrypt the string on server side using any simple technique. The server will return the encrypted version of the string back to the client. I have used simple encryption technology which will perform the operation on the ASCII value of each character of the string. It will multiply the ASCII value of the char by 3 and add 10 to it.

```
outputEcho[i] = (char) (((*argp)[i] * 3) + 10);
```

4. List the files:

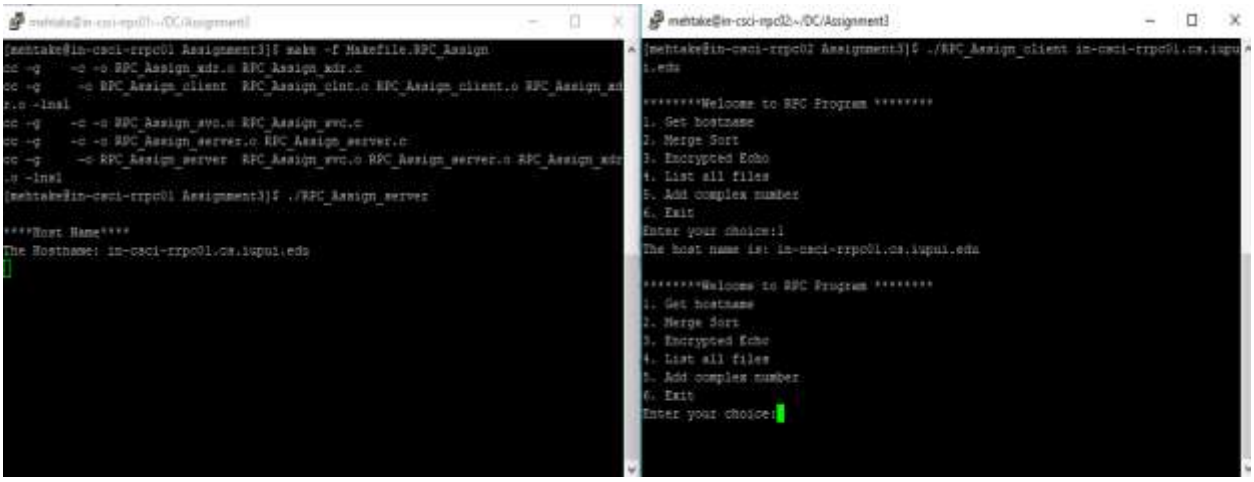
This operation will return the list of files present in the current directory of the server. It will return the string which contains the name of all the files separated by '\n' char.

5. Add Complex numbers:

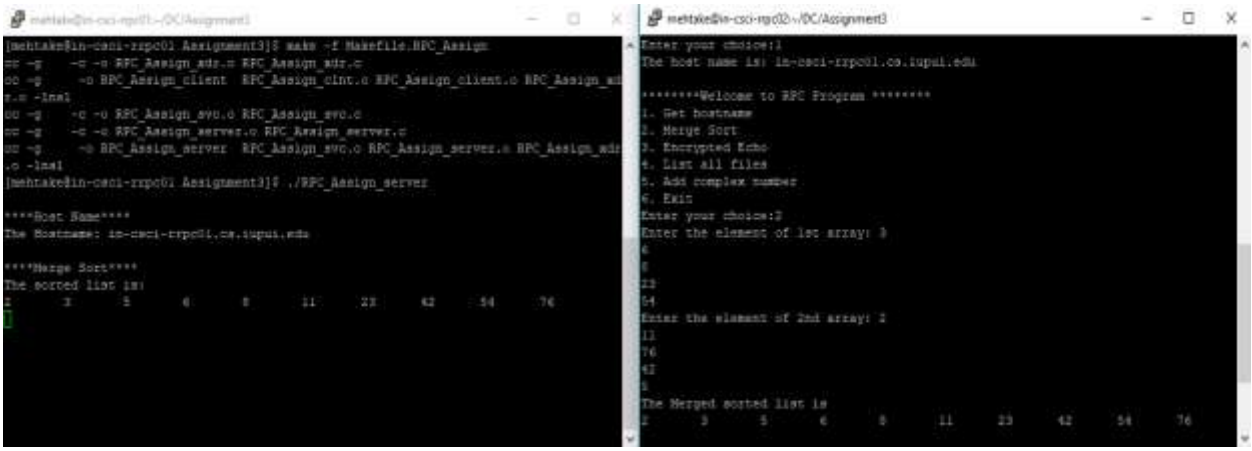
The user needs to enter the 2 complex number for this operation. Client stub will marshall these numbers in one structure object and send across the server. Sever will perform the addition of the 2 complex numbers and sends the result back to the client.

3. Screenshots

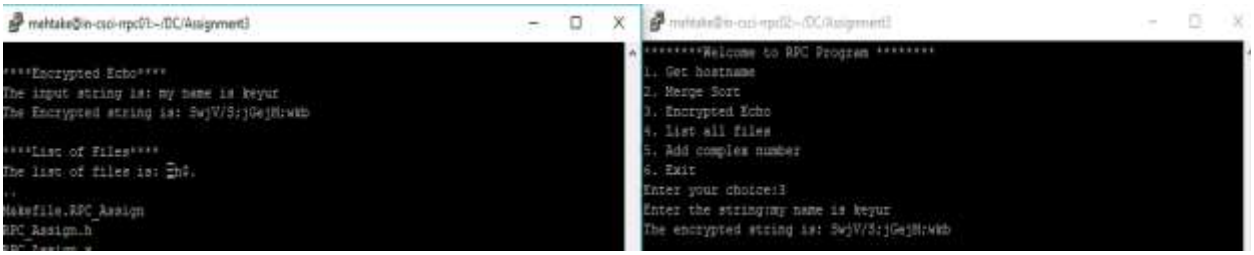
1. Host Name



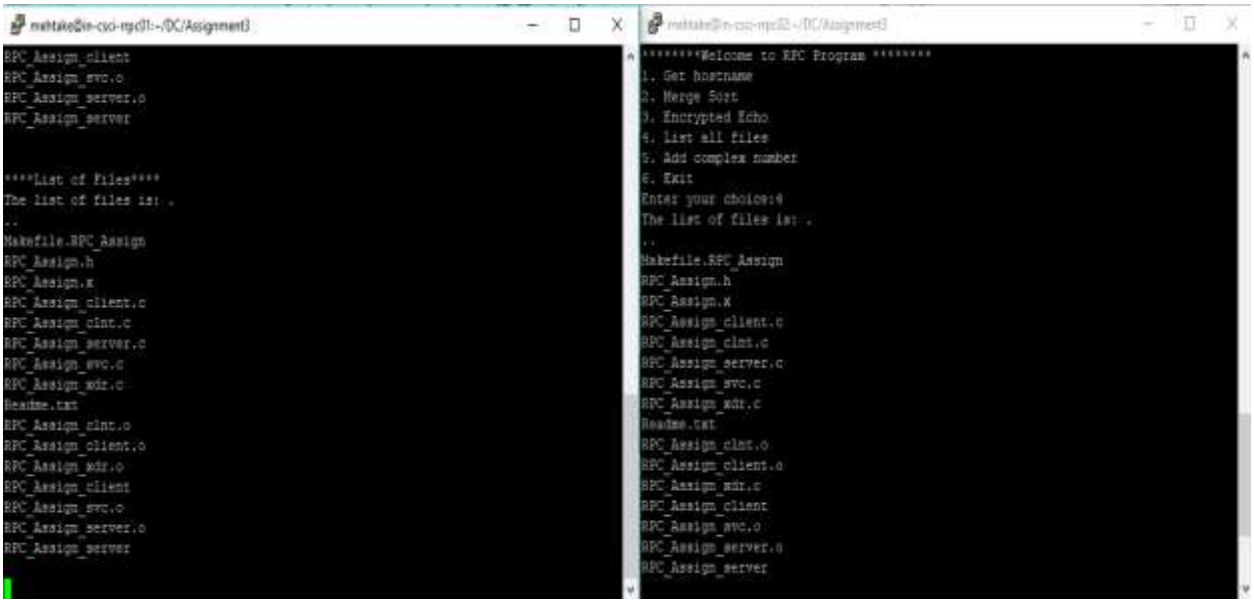
2. Merge and Sort 2 list



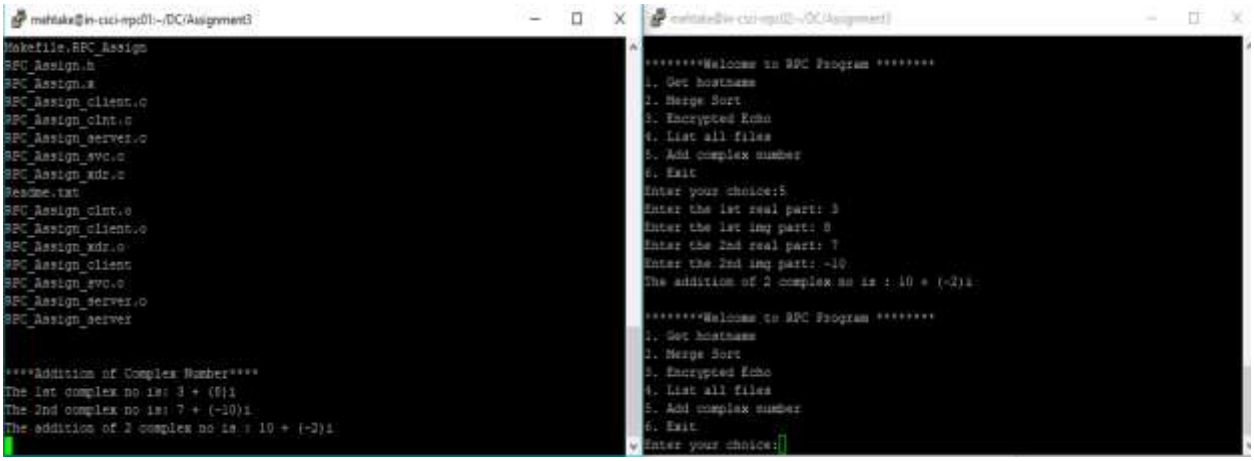
3. Encrypt the String



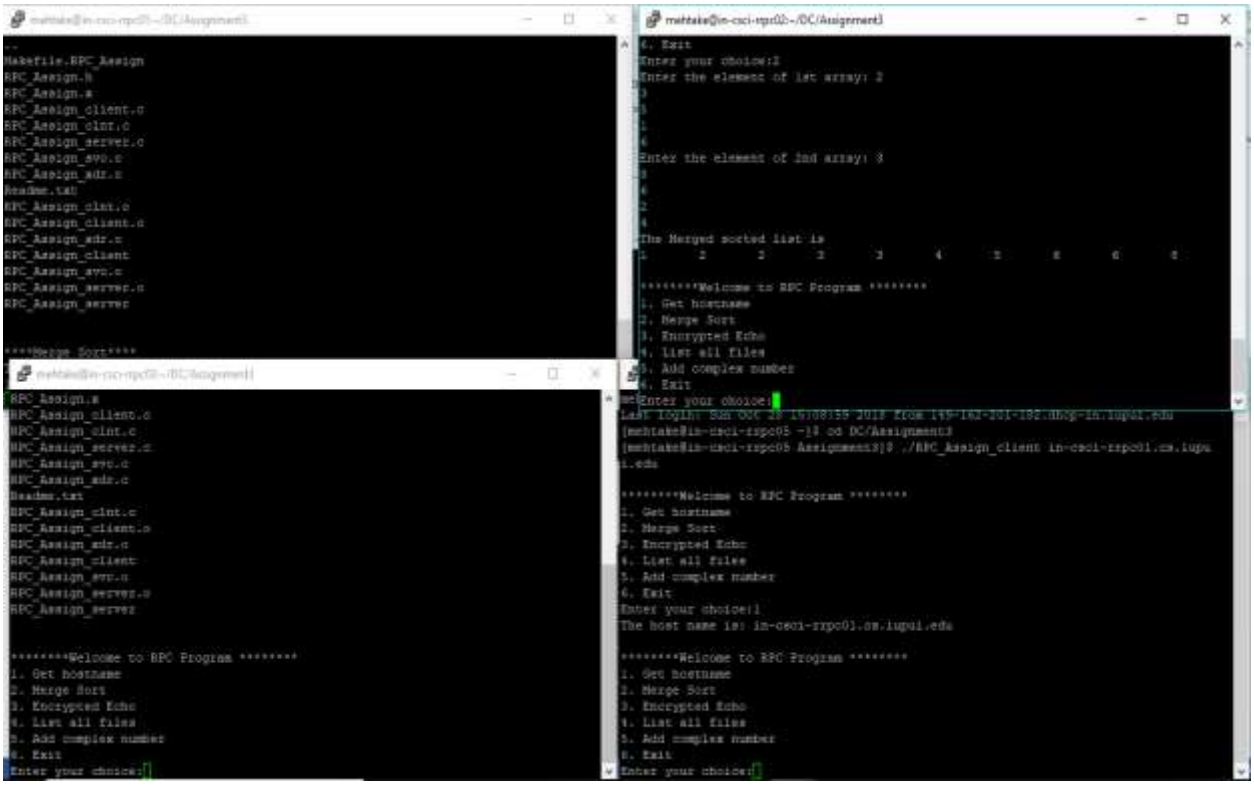
4. List all the File



5. Add Complex Number



6. Multi-Threading



4. Conclusion

The assignment implements the RPCgen library successfully in C. It provide the knowledge of the Remote Procedure Call with help of auto generated stubs.

Advantage:

1. RPCgen generates all the stubs for client and server which actually performs all the communication between client and server. So, user does not need to worry about remote calls. It is more efficient and makes simpler for programmer to code.
2. Programmer just need to implement the server logic of operation and client control flow.
3. It also supports the multithreading on server side. So, multiple client can send concurrent requests.

Disadvantage:

1. It can pass only 1 arguments during the call. So, all the parameters must be enclosed in structure and pass to the server call.

References

1. https://docs.oracle.com/cd/E19253-01/816-1435/rpcgenpguide-24243/index.html
2. https://en.wikipedia.org/wiki/Remote_procedure_call