CS 603 Using Sun RPC Spring 1996

1 Introduction

The goal of this project is to do client-server programming using Remote Procedure Calls. You will learn how to use the SUN-RPC implementation.

2 Part 1: Implementing Your Own Rusers Command

In this part you will implement your own *rusers* command. There already exists an rusers command in Solaris that gives you information about the users in remote hosts. This command is a client program that communicates to a special rusers server running always in the background on Solaris machines. Rusers' client and server use RPC's and follow the RPC description available in /usr/include/rpcsvc/rusers.x.

In this part of the project you should write a simplified version of the client rusers program that will be called *myrusers*. Your myrusers program has the following syntax:

```
myrusers host
```

It prints a similar output to the one you obtain by typing "rusers -l host". We suggest you not to use rpcgen to code your program. You have to build the xdr filters by hand. In this way you will see the advantages of using rpcgen in the second part of the project. You still can use rpcgen to compare your code and the automatically generated program.

In /u/u3/603/96spring/SUNWabsdk-ps/ps/NETP/ you can find postsript documentation on the SUN-RPC interface and rpcgen. I strongly recommend you to read chapters 01 to 04. They give you a very good introduction to Sun-RPC. Use *ghostview* to read the documentation and don't use the printers unnecessarily.

Part1 is due on Tuesday March 12 before 5:00pm in my office (CS-274). You just have to submit a printout of the sources of your program and the output of running "myrusers ector".

3 Part 2: Build your own RPC client and Server

In this part you should build the RPC client and server programs of your choice. You can use rpcgen to generate the basic templates for these programs. The creativity you use for your application also will be graded as well as the technical value.

Due date: to be announced.