

Final Project Report
Xuehai Zhang
915292

Topic: Rpc client/server to become a music server.

Description:

This project is the extension for Homework 7. It simulates a server to response the request for music.

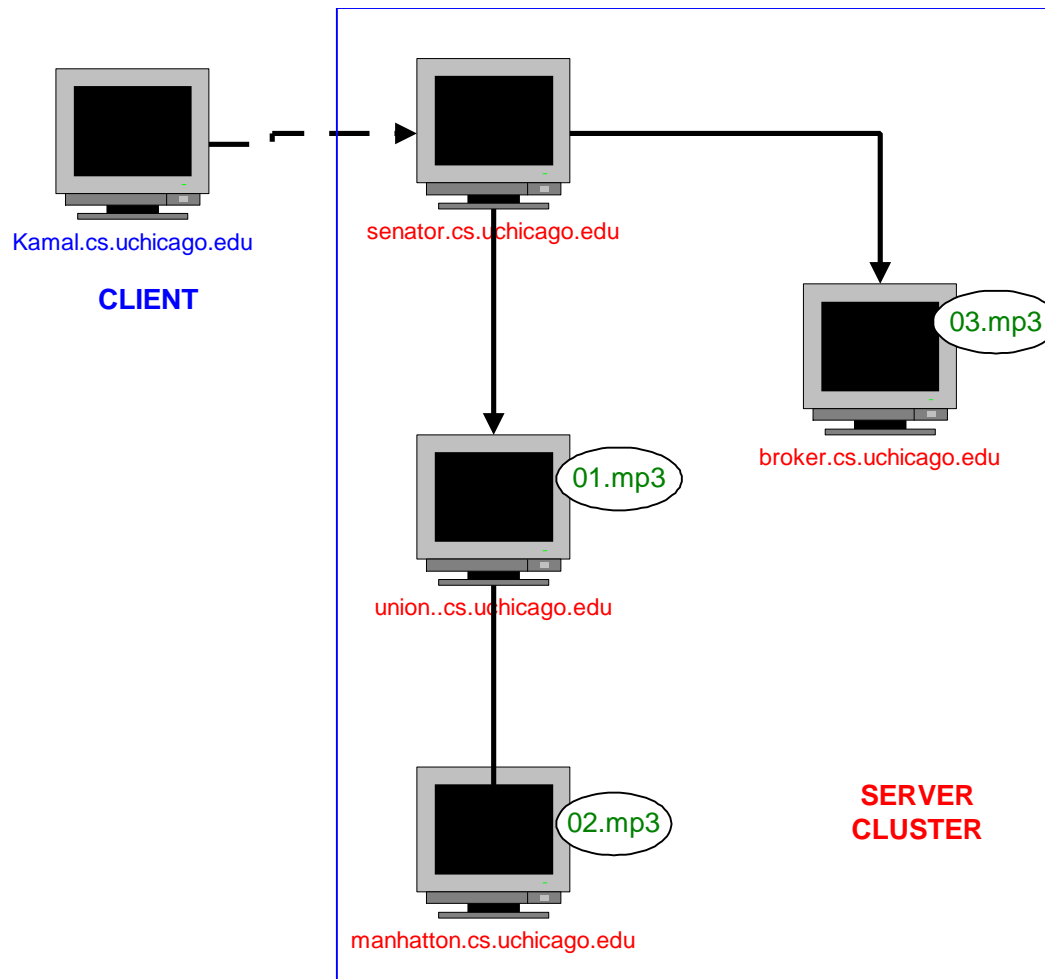
The project requirements said, add support for multiple servers, so that a single client request automatically searches multiple servers. Since the legal troubles of Napster, there has been increased interest in anonymous distributed information storage (see [Clark2000](#)). So rather than implementing your multiple server search from the viewpoint of the client searching multiple servers, have the server which the client contacts, pass the search on to other servers. The server with the data should return it back down the line. Don't bother with the encryption keys until you have the basic search passing down. For extra credit, implement the encryption described by Clark.

In my implementation, I fulfill the requirements of the project. I provide the following functions:

Function	Usage	Description
I	\$ I	Initialize the music index of a server
i	\$ i <filename>	Insert/upload a music file from server
d	\$ d <filename>	Delete a music file from server
l	\$ l <filename>	Lookup /search a music file from the server
t	\$ t <filename>	Transfer/download a music file from the server
s	\$ s	Show server cache content
q	\$ q	Quit

In the implementation, I have achieved to make the server clusters transparent to client. In testing, I use the following topology:

I use kamal.cs.uchicago.edu as the client machine to execute the request functions; senator-, union-, broker-, manhattan- .cs.uchicago.edu are 4-server clusters. I let kamal connects to senator, that means, for client kamal, it can only see one music server senator, all the other servers are transparent to it. I let union has music file 01.mp3, broker has music file 03.mp3 and manhattan has the music file 02.mp3.



Script [appended below]:

Below is the script collected at client kamal. You see, it tries a lot of functions. For example, it use “t 02.mp3” to download the music file 02.mp3 from server senator, but since there is no 02.mp3 in senator, senator will direct the download request to manhattan through union, then manhattan will first transfer the 02.mp3 to senator, then senator will transfer the 02.mp3 to kamal. You will learn more details about the other function calls.

The attached file senator.txt, union.txt , broker.txt and manhattan.txt are the scripted generated at each of the cluster server. If you explore it, you will learn how the transparent rpc function calls work.

Future Work:

Till now in this implementation, the server clusters can only provide service for one client , since we do not use multithreads to provide service for multiple request. We can implement it in the future.

Another consideration is the asynchronization transfer. Which means the client needs not wait the return from the server.

Script started on Fri Jun 8 15:05:41 2001
hai@kamal:s% c broker

remote server is broker.cs.uchicago.edu

I

Key directory initialized to empty.

i 03.mp3

searchfile result: nothing

begin transfer

end transfer]

end transfer, transfer file len 4096

transfer 0 part

end transfer, transfer file len 4096

....

end transfer, transfer file len 4096

transfer 330 part

end transfer, transfer file len 4096

transfer 331 part

end transfer, transfer file len 4096

transfer 332 part

end transfer, transfer file len 2813

transfer 333 part

transfer end signal

03.mp3 inserted.

corresponding file copied!

q

program quits.

hai@kamal:s% c manhattan

remote server is manhattan.cs.uchicago.edu

I

Key directory initialized to empty.

i 02.mp3

searchfile result: nothing

begin transfer

end transfer]

end transfer, transfer file len 4096

transfer 0 part

end transfer, transfer file len 4096

transfer 1 part

...

end transfer, transfer file len 4096

transfer 337 part

end transfer, transfer file len 4096

transfer 338 part

end transfer, transfer file len 4096

transfer 339 part

end transfer, transfer file len 195

transfer 340 part

transfer end signal

02.mp3 inserted.

corresponding file copied!

q

program quits.

hai@kamal:s% c senator

remote server is senator.cs.uchicago.edu

I

Key directory initialized to empty.

l 01.mp3

searchfile result: 01.mp3

File 01.mp3 found.

l 02.mp3

searchfile result: 02.mp3

File 02.mp3 found.

. l 03.mp3

searchfile result: 03.mp3

File 03.mp3 found.

t 02.mp3

searchfile result: 02.mp3

Found file 02.mp3, ready to download ...

open 02.mp3 for download

call remote tranferdown

client: received packet len 4096

call remote tranferdown

client: received packet len 4096

call remote tranferdown

...

client: received packet len 4096

call remote tranferdown

client: received packet len 4096

call remote tranferdown

client: received packet len 4096

call remote tranferdown

client: received packet len 195

call remote tranferdown

client: received packet len 0

client: finish downloading

Successfully download the file 02.mp3

d 02.mp3

searchfile result: 02.mp3

File 02.mp3 deleted.

q

program quits.

hai@kamal:s% ^D

Script done on Fri Jun 8 15:12:19 2001