Project 1: Transparent Remote File Operations

AndrewID:xiangyus
Name: Xiangyu Sun

RPC Concurrent Strategy:

1. Client:
    a. Initiate the TCP connection in the _init() function.
    b. Close the TCP connection in the _fini() function.
    c. During this process of the client RPC operations, the connection keeps alive.
2. Server:
    a. Multi Process server.
    b. When receiving a TCP connection request, the server forks a process to handle this connection.

RPC General Serialization Protocol:

- The first 8 bytes store the length of the incoming message so the server and the client can dynamically assign memory to store the message.
- To be able to send and receive the information, the software use a while loop to record how many bytes of this message have been sent or received.
1. Client:
    a. The name of the function is stored right after the first 8 bytes so the server can recognize which RPC function the client is calling.
    b. Parameters of this function is stored after the function name and separated by a '\0'.
    c. Unfixed length of parameters will have a length information stored before the this parameter.
2. Server:
    a. Similar to client, the first 8 bytes store the length of this message. Then store the errno and all the other returned values. All the returned values are separated by '\0'.
3. File Descriptors:
    a. Based on the suggestions found in Piazza, on server side, all the received file descriptors will be added by an offset; on the client side, all the received file decriptors will be minused by the same offset.

RPC DirTreeNode Serialization and Deserialization Protocol:

1. Serializaiton on Server side
    a. Use PreOrder DFS traverse to traverse the tree. When traversing to a new node, malloc a new chunk of memory to dynamically change the size of memory to store the serialization information and move the information in the old chunk into this new one.
    b. The information of the node to be stored will be the length of the name, and the name, the number of the subdirs. If the node has subdirs, the following bytes will store the offsets of these subdir nodes. If no subdirs, the following bytes will store the infomation of another node.
2. Deserialization on Client side
    a. Use the same DFS traverse method to traverse the chunk of memory based on the stored offset.
    b. While traversing, generate the node struct and return the node to linked them to their parent.