PACKET:

|  |  |  |
| --- | --- | --- |
| 1 byte | 4 bytes | X bytes |
| Command code | (integer) Number of following bytes | Data |

PLAYBACK COMMAND DATA (SENDING):

1 byte: PLAY = 0, PAUSE = 1, SKIP = 2

NAME CHANGE COMMAND DATA (SENDING):

X Bytes: Name to change to

NAME CHANGE COMMAND DATA (RECEIVING):

X Bytes: The name being changed

1 Byte: Null separator (‘\0’)

X Bytes: The name it was changed to

FILE SELECT COMMAND DATA (SENDING):

1 byte (type): FILE = 0, URL = 1

X bytes: Location of media (e.g. “www.google.ca/dingdong.mp3”)

USER LIST COMMAND DATA (SENDING):

0 Bytes: Send nothing extra (this is a request, not data)

USER LIST COMMAND DATA (RECEIVING):

1 Byte: Status (0 = EXISTS (was already connected when you joined), 1 = CONNECT, 2 = DISCONNECT)

MUTE COMMAND DATA (SENDING):

1 Byte: To Set (0 = UNMUTE, 1 = MUTE)

X Bytes: Name to mute

MUTE COMMAND DATA (RECEIVING):

1 Byte: To Set on the client side (0 = UNMUTE, 1 = MUTE)

X Bytes: Name to mute

MESSAGE COMMAND DATA (SENDING):

X Bytes: The message you are sending

MESSAGE COMMAND DATA (RECEIVING):

X Bytes: The message to display on screen (don’t worry about who sent it, the sender is included in the message, just display it)

CURRENT COMMAND DATA (RECEIVING) (don’t send this):

X Bytes: The new now playing string

FILE LIST COMMAND DATA (SENDING)

0 Bytes: (This is a request only, so any additional data will be ignored)

FILE LIST COMMAND DATA (RECEIVING) (one per file):

X Bytes: The path to the file

FILE REQUEST COMMAND DATA (SENDING):

X Bytes: The path to the file

FILE REQUEST COMMAND DATA (RECEIVING) (you’ll get a bunch of these, just keep appending until a packet has FILE\_EOF as the type:

X Bytes: File data

FILE\_EOF (RECEIVING):

0 Bytes (command only)This means that the last of the file has been send, you can clean up and close the file now.