1. 注册：

recvbuf[0] == REGISTER (0x01) 长度1

[1]: 用户名tag 0x11 长度1

[2]~[51]: 用户名 长度50

52： 密码tag 0x12 长度1

53~102： 密码 长度50

response:

[0] REGISTER (0x01) 长度1

[1] 注册结果tag 0x21 长度1

[2] 注册结果 0x01—成功 0x02—失败 长度1

[3] 注册失败原因tag 0x22 长度1

[4] 注册失败原因0x01-用户名过长，0x02-用户密码过长 0x03-用户名已注册 长度1

1. 登录：

recvbuf[0] == LOGIN (0x02) 长度1

[1]: 用户名tag 0x11 长度1

[2]~[51]: 用户名 长度50

52： 密码tag 0x12 长度1

53~102： 密码 长度50

response:

[0] LOGIN (0x02) 长度1

[1] 登录结果tag 0x21 长度1

[2] 登录结果 0x01—成功 0x02—失败 长度1

[3] 登录失败原因tag 0x22 长度1

[4] 登录失败原因0x01-用户已登录，0x02-密码错误 0x03-无此用户 长度1

现原因2,3合并为2，原因为用户名或密码错误

2.1登录通知：

recvbuf[0] == LOGINOTHER (0x03) 长度1

[1]: 用户名tag 0x11 长度1

[2]~[51]: 用户名 长度50

2.2获取列表：循环发送，一次一个

recvbuf[0] == LOGINOTHERS (0x04) 长度1

[1]:用户名tag 0x11 长度1

[2]~[51]: 用户名 长度50

3，刷新玩家列表：

recvbuf[0] == REFRESHPLAYERLIST (0x06) 长度1---与2.2相应

[1]:用户名tag 0x11 长度1

[2]~[51]: 用户名 长度50

response：

同2.2

4、登出

recvbuf[0] == LOGINOUT (0x05) 长度1

[1]:用户名tag 0x11 长度1

[2]~[51]: 用户名 长度50

5、对局请求/回应：

[0] == PLAYGAMEREQ (0x69) / PLAYGAMERSP(0x70)

[1]:请求方tag 0x61

[2]~[51]:请求方 长度50

[52]:被请求方tag 0x62

[53]~[102]: 被请求方 长度50

[103]:让子数tag 0x63

[104]让子数index: 16进制char转int 长度1

[105]:贴子tag 0x64

[106]:贴子index 16进制char转int 长度1

[107]:黑白tag 0x65

[108]:黑白 长度1

[109]:申请结果tag 0x66 (PLAYGAMERSP)

[110]:申请结果 0x01—同意，0x00—不同意 (PLAYGAMERSP)

5.1 服务器方面的响应：

[0] == 0x69,则转发给被请求方

[0] == 0x70,则转发给请求方，且：

解析双方用户名，并找到客户端节点，对手指针互指

6、落子

recvbuf[0] == PLACEONEPIECE (0x71) 长度1

[1]:落子方tag 0x61

[2]~[51]: 落子方 长度50

[52]:接受方tag 0x62

[53]~[102]: 接受方 长度50

[103]坐标tag 0x19

[104]坐标x 16进制char

[105]坐标y 16进制char

7、认输

recvbuf[0] == PLACEONEPIECE (0x72) 长度1

[1]:认输方tag 0x61

[2]~[51]: 认输方 长度50

[52]:接受方tag 0x62

[53]~[102]: 接受方 长度50

8、刷新

[1]:请求端用户名tag 0x61

[2]~[51]: 请求方 长度50