Storm

game network protocol

General

The game uses a custom binary TCP/IP based communication scheme. The talk between client and server is affected by the client's **state**. Upon connecting, the client is assigned the *queued* state. In that state, the client can query the game information, such as the current map name, server mod and so on. The client can [down]load the required data and request to change its state to *playing*. Then the player is spawned, a *hello* message is sent and the communication continues through the messages described below.

Session port: 0x4210

Note: it is planned to use an UDP-based protocol to query server configuration without connecting (for the server list display). ***This is not considered at the moment.***

General message format

Each message begins with a 16-bit unsigned message ID.

0x0***, 0x1*** are for **queued** (client->server, server->client)

0x2***, 0x3*** are for **playing**

Messages >= 0x8000 are **mod-specific**

Queued state

10 seconds message timeout, 30 seconds total timeout! (DoS prevention)

TODO: QUE_CLR_login and responses

0x0001 - queue_client_hello

client sends upon opening connection

0x0002 - queue_client_exit

response to queue_server_welcome

client hasn't got the required resources. disconnects.

after that, it will probably attempt to retrieve the resources via HTTP

0x0003 - queue_client_loading

client confirms availability of all the required resources and begins to load them

sets timeout to 1 minute?

0x0004 - queue_client_ready

string playerNickname;

requests state change to playing

0x1000 - queue_server_welcome

string serverName;

string modName;

string mapName;

string modHttpDownload; // empty if not available

string mapHttpDownload; // empty if not available

response to queue_client_hello

0x1001 - queue_server_accepted

client's request to enter game accepted. changing state.

0x1800 - queue_server_full

server sends if server is full. disconnects then.

TODO: queue for waiting players ??

0x1801 - queue_server_denied

string reason;

server denied player's request. disconnects then.

0x1802 - queue_server_timeout

client timed out. server disconnects.

Playing state

5 seconds message timeout

0x2000 - client_heartbeat

sent every second to prevent connection from breaking

0x2001 - client_ping_response

0x3000 - server_players

uint16 num_players;
uint16 your_pid;
for each in num_players:
    uint16 pid;
    string nickname;
    float x, y, z, direction;
    uint8 status; - 0 for not spawned or dead, 1 for alive
    string weapon;

the first message sent after player enters game.

0x3001 - server_player_dead

uint16 player_id;
float x, y, z, direction;

client should spawn a corpse for 10 secs

0x3002 - server_ping

client should respond with client_ping_response ASAP

0x3003 - server_disconnected

string reason;

Mod-specific messages: DM

dm = team death match

0x8000 - dm_client_choose_team

uint16 team_id;

0x9000 - dm_server_teams

uint16 num_teams;
for each in num_teams:
uint16 team_id; (nonzero!!)
string name;
// more to come (player model etc.)

0x9001 - dm_server_player_teams

uint16 count;
for each in count:
uint16 pid;
uint16 team_id; (0 = spectating)
player's default team is spectating