

MTA R1.1 RemoteAdministration Protocol

(From Multi Theft Auto Wiki:

http://wiki.mtasasa.com/index.php?title=MTA_R1.1_remote_administration_protocol)

This page was last modified 23:39, 1 June 2007.

Current protocol version is **R1.1**.

Packet List

This is where the packet ID's are detailed. Descriptions of how the individual packets are made up will be given further down.

NOTE: Every packet sent and received has a byte of value 0x6D as the 1st byte. This will be denoted as ADMIN_PACKET_HEADER throughout this document.

- ADMIN_CONNECT = 0x00
- ADMIN_DISCONNECT = 0x01
- ADMIN_PING = 0x02
- ADMIN_CHAT = 0x03
- ADMIN_PLAYER_JOIN = 0x04
- ADMIN_PLAYER_PART = 0x05
- ADMIN_PLAYER_LOGIN = 0x06
- ADMIN_KICK = 0x07
- ADMIN_BAN = 0x08
- ADMIN_ADDBAN = 0x09
- ADMIN_UNBAN = 0x0A
- ADMIN_CHANGE_SERVER_PASSWORD = 0x0B
- ADMIN_SERVER_SHUTDOWN = 0x0C
- ADMIN_PLAYER_MUTE = 0x0D
- ADMIN_PLAYER_FREEZE = 0x0E
- ADMIN_PLAYER_NICK_CHANGE = 0x0F
- ADMIN_PLAYER_DEATH = 0x10
- ADMIN_PLAYER_PM = 0x11
- ADMIN_RACE_LIST = 0x12
- ADMIN_START_RACE = 0x13
- ADMIN_SET_MAX_PLAYERS = 0x14
- ADMIN_PLAYER_FINISHED_RACE = 0x15
- ADMIN_PLAYER_PING = 0x16
- ADMIN_PLAYER_RESPAWN = 0x17

Packet Description Formatting

This is where the packets are described. They will be covered in order of packet ID (Check the enumerations above). Full descriptions of the packets and what is sent and received is described.

The packet makeup key is made up of bytes sent (Hexadecimal values are given when there is not an enumeration available).

Strings sent in the bytestream will be denoted like: `[string(description)]`

Variable bytes sent will be denoted like: `[byte(description)]`

Looped information in packets will be denoted like:

`{ loop < number of times > (makeup of each iteration of loop) end-loop }`

ADMIN_CONNECT	0x00		
ADMIN_DISCONNECT	0x01	//	
ADMIN_PING	0x02	ADMIN_BAN_SUCCESS	0x00
ADMIN_CHAT	0x03	ADMIN_BAN_FAILED	0x01
ADMIN_PLAYER_JOIN	0x04	ADMIN_UNBAN_SUCCESS	0x00
ADMIN_PLAYER_PART	0x05	ADMIN_UNBAN_FAILED	0x01
ADMIN_PLAYER_LOGIN	0x06	//	
ADMIN_KICK	0x07	ADMIN_SERVER_PASSWORD_CHANGED	0x00
ADMIN_BAN	0x08		
ADMIN_ADDBAN	0x09	#define	
ADMIN_UNBAN	0x0A	ADMIN_SERVER_PASSWORD_FAILED	0x01
ADMIN_CHANGE_SERVER_PASSWORD	0x0B	ADMIN_SERVER_PASSWORD_DISABLED	0x02
ADMIN_SERVER_SHUTDOWN	0x0C	//	
ADMIN_PLAYER_MUTE	0x0D	ADMIN_SERVER_SHUTDOWN_SUCCESS	0x00
ADMIN_PLAYER_FREEZE	0x0E	ADMIN_SERVER_SHUTDOWN_WAITING	0x01
ADMIN_PLAYER_NICK_CHANGE	0x0F	ADMIN_SERVER_SHUTDOWN_DISABLED	0x02
ADMIN_PLAYER_DEATH	0x10	//	
ADMIN_PLAYER_PM	0x11	ADMIN_PLAYER_MUTED	0x00
ADMIN_RACE_LIST	0x12	ADMIN_PLAYER_UNMUTED	0x01
ADMIN_START_RACE	0x13	ADMIN_MUTE_SUCCESS	0x02
ADMIN_SET_MAX_PLAYERS	0x14	ADMIN_UNMUTE_SUCCESS	0x03
ADMIN_PLAYER_FINISHED_RACE	0x15	//	
ADMIN_PLAYER_PING	0x16	ADMIN_MUTE_FAILED	0x00
ADMIN_PLAYER_RESPAWN	0x17	ADMIN_UNMUTE_FAILED	0x01
//		//	
CONNECT_DISCONNECT	0x00	ADMIN_PLAYER_FREEZED	0x00
CONNECT_FAILED_BAD_PASS	0x01	ADMIN_PLAYER_UNFREEZED	0x01
CONNECT_CONNECTED	0x02	ADMIN_FREEZE_SUCCESS	0x02
CONNECT_TIMED_OUT	0x03	ADMIN_UNFREEZE_SUCCESS	0x03
CONNECT_TOO_MANY_USERS_FROM_IP	0x04	//	
CONNECT_BANNED	0x05	ADMIN_FREEZE_FAILED	0x00
CONNECT_NAME_IN_USE	0x06	ADMIN_UNFREEZE_FAILED	0x01
CONNECT_INVALID_VERSION	0x07	//	
//		CHAT_PRINT	0x00
ADMIN_CONSOLE_CHAT	0x00	CONSOLE_ECHO	0x01
ADMIN_ADMIN_CHAT	0x01	CHAT_PRINT_FAILED	0x02
ADMIN_PLAYER_CHAT	0x02	CONSOLE_ECHO_FAILED	0x03
//		//	
ADMIN_PLAYER_PARTED	0x00	ADMIN_RACE_LIST_NEW	0x00
ADMIN_PLAYER_KICKED	0x01	ADMIN_RACE_LIST_MAP	0x01
ADMIN_PLAYER_BANNED	0x02	ADMIN_RACE_LIST_END	0x02
ADMIN_PLAYER_TIMED_OUT	0x03	//	
//		ADMIN_RACE_STARTED	0x00
ADMIN_LOGIN_SUCCESS	0x00	ADMIN_RACE_FAILED	0x01
ADMIN_LOGIN_FAILED	0x01	ADMIN_RACE_NO_PLAYERS	0x02
ADMIN_LOGIN_LOGGED_IN	0x02		

ADMIN_CONNECT

Description:

This is sent to the server to connect to it. The server will respond depending on whether the attempt was successful or not.

Packet sent to server:

```
ADMIN_PACKET_HEADER ADMIN_CONNECT 0x02 [string(password)] 0x7C
[string(username)] 0x00
```

Response packet from server:

If the connect attempt is successful, the following packet is received:

```
ADMIN_PACKET_HEADER ADMIN_CONNECT [byte(Max players)] [byte(Current max
players)]
0xFF { loop < Number of players >([byte(Player ID)] [byte(Player rcon
level)]
[byte(Player mute status)] [byte(Player frozen status)] [byte(1st byte of
player IP)]
[byte(2nd byte of player IP)] [byte(3rd byte of player IP)] [byte(4th byte
of player IP)]
[string(Player Name)] 0xFF) end-loop } 0x00
```

If the connect attempt is unsuccessful, the following packet is received:

```
ADMIN_PACKET_HEADER ADMIN_DISCONNECT [byte(Disconnect reason)]
```

Notes:

Player rcon level ranges from 0x00 to 0x05 and denotes the level of rcon the player has in the server. Player mute status is either 0x00 (muted) or 0x01 (not muted). Player frozen status is either 0x00 (frozen) or 0x01 (not frozen).

The disconnect reason has the following enumerations:

- CONNECT_DISCONNECT = 0x00
- CONNECT_FAILED_BAD_PASS = 0x01
- CONNECT_CONNECTED = 0x02
- CONNECT_TIMED_OUT = 0x03
- CONNECT_TOO_MANY_USERS_FROM_IP = 0x04
- CONNECT_BANNED = 0x05
- CONNECT_NAME_IN_USE = 0x06
- CONNECT_INVALID_VERSION = 0x07

ADMIN_DISCONNECT

Description:

This packet is sent when to disconnect from the server. The server will send a response confirming this.

Packet sent to server:

```
ADMIN_PACKET_HEADER ADMIN_DISCONNECT
```

Response packet from server:

```
ADMIN_PACKET_HEADER ADMIN_DISCONNECT 0x00
```

ADMIN_PING

Description:

This packet is sent from the server to the Remote Admin clients every second. The Remote Admin client should respond.

Packet received from server:

```
ADMIN_PACKET_HEADER ADMIN_PING
```

Packet sent to server in response:

```
ADMIN_PACKET_HEADER ADMIN_PING
```

ADMIN_CHAT

Description:

This is the packet that the server sends for all chat. Remote Admin clients should send this packet to chat.

Packet sent:

```
ADMIN_PACKET_HEADER ADMIN_CHAT [string(Chat text)] 0x00
```

Packet sent from server on any chat:

```
ADMIN_PACKET_HEADER ADMIN_CHAT [byte(Player ID)] [byte(Chat type)]  
[string(Chat text)]
```

Notes:

Player ID will be 255 if an Admin or Console are chatting. The chat type has the following enumeration:

- ADMIN_CONSOLE_CHAT = 0x00
- ADMIN_ADMIN_CHAT = 0x01
- ADMIN_PLAYER_CHAT = 0x02

ADMIN_PLAYER_JOIN

Description:

This packet is received from the server when a new player joins the game.

Packet received from server:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_JOIN [byte(Player ID)] [string(Player  
name)] 0x00 [byte(1st byte of player IP)]  
[byte(2nd byte of player IP)] [byte(3rd byte of player IP)] [byte(4th byte  
of player IP)]
```

ADMIN_PLAYER_PART

Description:

This packet is received from the server when a player parts.

Packet received from server:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_PART [byte(Player ID)] [byte(Part reason)]
```

Notes:

The part reasons have the following enumerations:

- ADMIN_PLAYER_PARTED = 0x00
- ADMIN_PLAYER_KICKED = 0x01
- ADMIN_PLAYER_BANNED = 0x02
- ADMIN_PLAYER_TIMED_OUT = 0x03

ADMIN_PLAYER_LOGIN

Description:

This packet is received from the server when a player logs in to rcon admin.

Packet received from server:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_LOGIN [byte(Player ID)] [byte(Login type)]  
[byte(Rcon level)]
```

Notes:

The rcon level will be between 0x00 and 0x05.

The login type has the following enumerations:

- ADMIN_LOGIN_SUCCESS = 0x00
- ADMIN_LOGIN_FAILED = 0x01
- ADMIN_LOGIN_LOGGED_IN = 0x02

ADMIN_KICK

Description:

This packet is sent to the server to kick a player. The response will be a ADMIN_PLAYER_PART packet if successful.

Packet sent:

```
ADMIN_PACKET_HEADER ADMIN_KICK [byte(Player ID)]
```

ADMIN_BAN

Description:

This packet is sent to the server to ban a player. The response will be a ADMIN_PLAYER_PART packet if successful.

Packet sent:

```
ADMIN_PACKET_HEADER ADMIN_BAN [byte(Player ID)]
```

ADMIN_ADDBAN

Description:

This packet is sent to the server to ban an IP. The server will respond with a packet outlining the outcome.

Packet sent:

ADMIN_PACKET_HEADER ADMIN_ADDBAN [string(IP)]

Response from server:

ADMIN_PACKET_HEADER ADMIN_ADDBAN [byte(Response)]

Notes:

The response has the following enumerations:

- ADMIN_BAN_SUCCESS = 0x00
- ADMIN_BAN_FAILED = 0x01

ADMIN_UNBAN

Description:

This packet is sent to the server to unban an IP. The server will respond with a packet outlining the outcome.

Packet sent:

ADMIN_PACKET_HEADER ADMIN_UNBAN [string(IP)]

Response from server:

ADMIN_PACKET_HEADER ADMIN_UNBAN [byte(Response)]

Notes:

The response has the following enumerations:

- ADMIN_UNBAN_SUCCESS = 0x00
- ADMIN_UNBAN_FAILED = 0x01

ADMIN_CHANGE_SERVER_PASSWORD

Description:

This packet is sent to the server to change the server password, if it is enabled on the server. The server will respond with the outcome.

Packet sent:

ADMIN_PACKET_HEADER ADMIN_CHANGE_SERVER_PASSWORD [string(New password)]

Response from server:

ADMIN_PACKET_HEADER ADMIN_CHANGE_SERVER_PASSWORD [byte(Response)]

Notes:

The response has the following enumerations:

- ADMIN_SERVER_PASSWORD_CHANGED = 0x00
- ADMIN_SERVER_PASSWORD_FAILED = 0x01
- ADMIN_SERVER_PASSWORD_DISABLED = 0x02

ADMIN_SERVER_SHUTDOWN

Description:

This packet is sent to the server to shut it down, if enabled on the server. The packet must be sent twice within 5 seconds for this to be successful.

Packet sent:

ADMIN_PACKET_HEADER ADMIN_SERVER_SHUTDOWN

Response from server:

ADMIN_PACKET_HEADER ADMIN_SERVER_SHUTDOWN [byte(Response)]

Notes:

The response has the following enumerations:

- ADMIN_SERVER_SHUTDOWN_SUCCESS = 0x00
- ADMIN_SERVER_SHUTDOWN_WAITING = 0x01
- ADMIN_SERVER_SHUTDOWN_DISABLED = 0x02

ADMIN_PLAYER_MUTE

Description:

This packet is sent to the server to mute/unmute a player. It is also sent from the server when a player is muted/unmuted and responds to a packet with a success or failure.

Packet sent:

ADMIN_PACKET_HEADER ADMIN_PLAYER_MUTE [byte(Player ID)] [byte(Mute/unmute)]

Notes:

The mute/unmute byte has the following enumerations:

- ADMIN_PLAYER_MUTED = 0x00
- ADMIN_PLAYER_UNMUTED = 0x01

Packet sent from server if successful:

ADMIN_PACKET_HEADER ADMIN_PLAYER_MUTE [byte(Player ID)] [byte(Muter ID)]
[byte(Mute/unmute)]

Notes:

The muted ID is either a player ID (the player who muted the person) or it is 0xFF for a mute from the Console or 0xFE for a mute from Remote Admin. The Mute/unmute byte has the following enumerations:

- ADMIN_MUTE_SUCCESS = 0x02
- ADMIN_UNMUTE_SUCCESS = 0x03

Packet sent from the server if unsuccessful:

ADMIN_PACKET_HEADER ADMIN_PLAYER_MUTE [byte(Player ID)] [byte(Already muted/unmuted)]

Notes:

The already muted/unmuted byte has the following enumerations:

- ADMIN_MUTE_FAILED = 0x00
- ADMIN_UNMUTE_FAILED = 0x01

ADMIN_PLAYER_FREEZE

Description:

This packet is sent to the server to freeze/unfreeze a player. It is also sent from the server when a player is frozen/unfrozen and responds to a packet with a success or failure.

Packet sent:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_FREEZE [byte(Player ID)]  
[byte(Freeze/unfreeze)]
```

Notes:

The freeze/unfreeze byte has the following enumerations:

- ADMIN_PLAYER_FREEZED = 0x00
- ADMIN_PLAYER_UNFREEZED = 0x01

Packet sent from server if successful:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_FREEZE [byte(Player ID)] [byte(Freezer  
ID)] [byte(Freeze/unfreeze)]
```

Notes:

The Freezer ID is either a player ID (the player who froze the person) or it is 0xFF for a freeze from the Console or 0xFE for a freeze from Remote Admin.

The Freeze/unfreeze byte has the following enumerations:

- ADMIN_FREEZE_SUCCESS = 0x02
- ADMIN_UNFREEZE_SUCCESS = 0x03

Packet sent from the server if unsuccessful:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_FREEZE [byte(Player ID)] [byte(Already  
frozen/unfrozen)]
```

Notes:

The already frozen/unfrozen byte has the following enumerations:

- ADMIN_FREEZE_FAILED = 0x00
- ADMIN_UNFREEZE_FAILED = 0x01

ADMIN_PLAYER_NICK_CHANGE

Description:

This packet is sent from the server when a player changes their nickname.

Packet sent from server:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_NICK_CHANGE [byte(Player ID)] [string(New  
nickname)]
```

ADMIN_PLAYER_DEATH

Description:

This packet is sent from the server when a player dies.

Packet sent from server:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_DEATH [byte(Player ID)]
```

ADMIN_PLAYER_PM

Description:

This packet is sent from the server when a player pm's another player. It is also sent from the server when an admin pm's a player.

Packet sent:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_PM [byte(PM type)] [byte(From player ID)]  
[string(Message)]
```

Packet sent from server:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_PM [byte(From player ID)] [byte(To player  
ID)] [string(Message)]
```

Packet sent from server if successful:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_PM [byte(From Admin)] [byte(PM Type)]  
[byte(To player ID)] [string(Message)]
```

Packet sent from the server if unsuccessful:

```
ADMIN_PACKET_HEADER ADMIN_PLAYER_PM [byte(From Admin)] [byte(PM Type)]
```

Notes:

The PM type byte has the following enumerations:

- CHAT_PRINT = 0x00
- CONSOLE_ECHO = 0x01
- CHAT_PRINT_FAILED = 0x02
- CONSOLE_ECHO_FAILED = 0x03

ADMIN_RACE_LIST

Description:

The racelist is send over multiple packets. They are sent on join and when the list changes.

Packet sent from server:

```
ADMIN_PACKET_HEADER ADMIN_RACE_LIST [byte(List state)]  
ADMIN_PACKET_HEADER ADMIN_RACE_LIST [byte(List state)] [2 bytes(Race ID)]  
[string(Race name)]
```

Notes:

The List state byte has the following enumerations:

- ADMIN_RACE_LIST_NEW = 0x00
- ADMIN_RACE_LIST_MAP = 0x01
- ADMIN_RACE_LIST_END = 0x02

ADMIN_START_RACE

Description: This packet is sent to the server to start a race. It is also sent from the server when a new race starts.

Packet sent:

ADMIN_PACKET_HEADER ADMIN_START_RACE [2 bytes (Race ID)]

Packet sent from server:

ADMIN_PACKET_HEADER ADMIN_START_RACE [byte (Start type)] [2 bytes (Race ID)]

Notes:

Start type has the following enumerations:

ADMIN_RACE_STARTED = 0x00

ADMIN_RACE_FAILED = 0x01

ADMIN_RACE_NO_PLAYERS = 0x02

ADMIN_SET_MAX_PLAYERS

Description:

This packet is sent to change the max players of the server. The server will also send a packet when the max players is changed. Note that this number cannot go over the max players in the server config, nor can it ever go above 32.

Packet sent:

ADMIN_PACKET_HEADER ADMIN_SET_MAX_PLAYERS [byte (Max players)]

Response from server:

ADMIN_PACKET_HEADER ADMIN_SET_MAX_PLAYERS [byte (Max players)]

Notes:

Max players will be 0x00 if the attempt to change them failed, otherwise it will be the hexadecimal value of the new number of max players.

ADMIN_PLAYER_FINISHED_RACE

Description:

This packet is received from the server when a player finishes.

Packet received from server:

ADMIN_PACKET_HEADER ADMIN_PLAYER_FINISHED_RACE [byte (Player ID)]

ADMIN_PLAYER_PING

Description:

This packet is received every 2 seconds from the server with all player pings.

Packet received from server:

ADMIN_PACKET_HEADER ADMIN_PLAYER_PING { loop < Number of players
>([byte (Player ID)] [byte (Ping)]) end-loop } 0xFF

Notes:

The Ping byte must be multiplied with 4 to get the actual ping.

ADMIN_PLAYER_RESPAWN

Description:

This packet is received from the server when a player respawns.

Packet received from server:

ADMIN_PACKET_HEADER ADMIN_PLAYER_RESPAWN [byte(Player ID)]