

Overview

This documentation will cover the network protocol that is used for communication between the video game's server and client. The document is broken down as follows:

1. Introduction
2. Simple Type Definitions
3. Complex Type Definitions
4. Login and Account Management – 0x00
5. Movement and Attack Modes – 0x10
6. Client Actions – 0x20
7. Party – 0x30
8. Trade – 0x40
9. Communication – 0x50

Introduction

An organized protocol is vital for the server and client model in a multiplayer video game. The protocol consists of a one byte header that will be represented in this document in hexadecimal form (0xFF). The first nibble (4 bits) of the header is used to group packets together based on their functions. The second nibble will be for the separate packet types in each group. After the header byte the content for each type of packet will be specified in the packets' documentation. Some objects like "creatures" are used by multiple packets and are defined as a complex type in the "Complex Type Definitions" section.

Simple Type Definitions

Void

The packet does not have any data to be transferred so only a header is sent.

Byte

8 bits of data

Integer

Signed and 32 bits long

Long

Signed and 64 bits long

Complex Type Definitions

String

Byte - length of the string = "n"

n-Bytes - the characters in the string

MD5

16-Bytes - the bytes of the 128 bit md5 hash

Position

Integer - x coordinate

Integer - y coordinate

Creature

Integer - creature's id to be used by the server and client to keep track of creatures,

An id of 0xFFFFFFFF refers to the client's character (4 bytes)

String - creature's name (1 byte + string length)

Byte - creature's type represented by a bit (1 byte)

0000 0001 – character

0000 0010 – npc

0000 0100 – monster

Integer - creature's current hp (4 byte)

Integer - creature's maximum hp (4 byte)

Position - creature's location on the map (4 + 4 bytes)

String - creature's outfit's sprite(1 byte + string length)

Item

Integer - item's id from [items.xml]

Position - item's position in inventory

Integer - item's count

File Id

Short - each file is represented by a bit, multiple files can be selected at the same time

0000 0000 0000 0001 – [attacks.xml]

0000 0000 0000 0010 – [creatures.xml]

0000 0000 0000 0100 – [items.xml]

0000 0000 0000 1000 – [npc.xml]

0000 0000 0001 0000 – [quests.xml]

0000 0000 0010 0000 – [shops.xml]

0000 0000 0100 0000 – [spells.xml]

0000 0000 1000 0000 – sprite file

0000 0001 0000 0000 – map file

Login and Account Management – 0x00 – Server to Client

Log In Success – 0x01

This packet is sent when a client authenticates successfully on log in.

Void

Client File Outdated – 0x02

This packet is sent by the server if the client's data files are out of date and need to be updated.

File Id - files which need to be updated

Log In or Account Management Failure – 0x0F

This packet is sent by the server when the client fails to authenticate in some process of log in or account

management.

String - describes the error that occurred

Send File – 0x03

This packet is used to send an updated file to a client.

File Id - file that is being sent

Long - size of the file that is being transfer in bytes = “n”

n-Bytes - content of the file

Password Change Success – 0x04

This packet is sent when a client successfully changes their password.

Void

Login and Account Management – 0x00 – Client to Server

Check File Versions – 0x05

This packet checks that all the client’s data files are up to date with the server’s.

MD5 - hash of [attacks.xml]

MD5 - hash of [creatures.xml]

MD5 - hash of [items.xml]

MD5 - hash of [npc.xml]

MD5 - hash of [quests.xml]

MD5 - hash of [shops.xml]

MD5 - hash of [spells.xml]

MD5 - hash of sprite file

MD5 - hash of map file

Request File – 0x02

This packet requests an up to date file from the server.

File Id - id of the file being requested

Character Creation – 0x03

This packet is sent to create a new character.

String - name of the new character

MD5 - password for the character

Log In – 0x01

This packet is sent to log in on a character.

String - character’s name to log in on

MD5 - md5 hash of the character’s password

Change Password – 0x04

This packet is sent to change a character’s password.

String - character’s name

MD5 - hash of the character’s current password

MD5 - hash of the new password

Log Out – 0x06

This packet is sent when the client wants to log out of the server.

Void

Movement and Attack Modes – 0x10 – Server to Client

Creature Move – 0x12

This packet is sent when a creature in the view of the client moves to another position still in view of the client.

Integer - id of the creature

Position - new position of the creature

Creature Appear – 0x13

This packet is sent when a creature comes in view of the client.

Creature - creature that appeared

Creature Disappear – 0x14

This packet is sent when a creature leaves the view of the client.

Integer - creature id of the creature that disappeared

Creature Statistics Update – 0x15

This packet is sent when the statistics of a creature in view of the client have changed.

Integer - creature id of the creature

Integer - the creature's new current hp

Integer - the creature's new max hp

Attack Failed – 0x16

This packet is sent when the attack failed.

Void

Movement and Attack Modes – 0x10 – Client to Server

Move Up – 0x11

This packet is sent when the client wants to move up on the map.

Void

Move Left – 0x12

This packet is sent when the client wants to move left on the map.

Void

Move Right – 0x13

This packet is sent when the client wants to move right on the map.

Void

Move Down – 0x14

This packet is sent when the client wants to move down on the map.

Void

Start Attack – 0x15

This packet is sent when the client wants to attack a creature.

Integer - creature id to attack

Stop Attacking – 0x16

This packet is sent when the client wants to stop attacking their target.

Void

Client Actions – 0x20 – Server to Client

Update Client Statistics – 0x21

This packet is sent when the client's statistics have changed and need to be updated.

Integer - new current hp

Integer - new maximum hp

Integer - new current mana

Integer - new maximum mana

Integer - new level

Long - new exp

Long - new amount of gold

Add Item – 0x22 (Unimplemented)

This packet is sent when an item has been added to a client's inventory.

Item - item that was added

Remove Item – 0x23 (Unimplemented)

This packet is sent when an item is removed from a client's inventory.

Position - position of the item in inventory that was removed

Change Item Count – 0x24 (Unimplemented)

This packet is sent when an item's count has changed

Position - position of the item in inventory that had its count changed

Short - new count of the item

Spell Casted – 0x28 (Unimplemented)

This packet is sent when a spell has been casted.

Integer - spell id of the spell that was casted from [abilities.xml]

Position - the position where the spell was targeted

Client Actions – 0x20 – Client to Server (Unimplemented - Client/Server)

Use Item – 0x21

This packet is sent when the client wants to use an item in inventory.

Position - position of the item in inventory

Move Item – 0x22

This packet is sent when the client wants to move an item in inventory.

Position - position where the item is now

Position - position where the item should be moved

Drop Item – 0x23

This packet is sent when the client wants to remove an item in inventory.

Position - position of the item in inventory

Cast Spell – 0x28

This packet is sent when the client wants to cast a spell.

Integer - spell id of the spell to cast from [abilities.xml]

Position - the position to target the spell

Party– 0x30 – Server to Client (Unimplemented - Client/Server)

List Party Members – 0x31

This packet is sent when a client joins a party. It lists the current members of a party.

Integer - number of party members = “n”

n-Creature - members in the party

Add Party Member – 0x32

This packet is sent when a member has been added to a party the client is a member of.

Creature - new party member

Remove Party Member – 0x33

This packet is sent when a member has been removed from a party the client is a member of.

Integer - creature id of the removed member

Party – 0x30 – Client to Server (...)

Trade – 0x40 – Server to Client (Unimplemented - Client/Server)

Send Trade Request – 0x41

This packet is sent when a trade has been requested by another client to this client.

Integer - creature id of the initiator of the trade

Send Add Trade Item – 0x42

This packet is sent when the other participant in a trade adds an item.

Item - item that was added

Send Remove Trade Item – 0x43

This packet is sent when the other participant in a trade removes an item.

Position - the position of the item to remove

Send Change Trade Gold Amount – 0x44

This packet is sent when the other participant in a trade changes the amount of gold being traded.

Long - new amount of gold

Send Trade Accepted – 0x45

This packet is sent when the other participant accepts the trade.

Void

Trade Change Failed – 0x4E

This packet is sent when a change to the trade like an item removed or added fails.

String - reason for the failure

Trade Completed – 0x46

This packet is sent when the trade is completed

Void

Trade Failed – 0x4F

This packet is sent when a trade is not accepted by both participants or when a trade request has failed.

String - reason for the failure

Trade – 0x40 – Client to Server (Unimplemented - Client/Server)

Request Trade – 0x41

This packet is sent when the client wants to request a trade with a character in their view.

Integer - creature id of the creature to trade with

Accept Trade – 0x42

This packet is sent to accept a trade.

Void

Reject Trade – 0x43

This packet is sent to reject a trade.

Void

Add Trade Item – 0x44

This packet is sent to add an item to the active trade.

Item - item to add

Remove Trade Item – 0x45

This packet is sent to remove an item from the active trade.

Position - the position of the item to remove

Change Trade Gold – 0x47

This packet is sent to change the amount of gold in the active trade.

Long - the new amount of gold to trade

Communication – 0x50 – Server to Client

Send Message – 0x51

This packet sends a message to a client.

String - name of creature that sent the message, if 0-length the message is anonymous

String - message contents

Communication – 0x50 – Client to Server

Send Message – 0x51

This packet is sent when the client is sending a text message.

String - message contents

Buy From NPC Shop – 0x56 (Unimplemented - Client/Server)

This packet is sent when a client wants to buy an item from an NPC's shop.

Integer - NPC's id

Integer - item Id to buy

Sell To NPC Shop – 0x57 (Unimplemented - Client/Server)

This packet is sent when the client wants to sell an item to an NPC's shop.

Integer - NPC's id

Position - inventory position of item to sell