

Protocol Documentation for Echat

Echat Overview

This document explains the protocol used for communication between the client and the MultiThreaded Server. It describes how commands are structured, their purposes, and how the server processes them.

Protocol Details

Communication Basics

- **Transport Protocol:** TCP
- **Port:** Default is 42069 .
- **Message Format:** Plain text strings terminated by newline (\n).
- **Connection:** Each client establishes a separate connection to the server using a Telnet-like client.

Command Structure

Commands are composed of a command keyword followed by optional arguments. The general syntax is:

`<command> [arg1] [arg2] ...`

- Commands are case-insensitive.
- Arguments are separated by spaces.

Supported Commands

Authentication Commands

1. Register:

- **Usage:** register <username> <password>
- **Description:** Registers a new user with the server.
- **Response:** Success or failure message.

2. Login:

- **Usage:** login <username> <password>
- **Description:** Authenticates an existing user.
- **Response:** Success message, or error if authentication fails.

Room Management Commands

1. Join Room:

- **Usage:** join <room_name>
- **Description:** Joins the specified chat room if the user has sufficient permissions.
- **Response:** Confirmation message, or error if the room doesn't exist or permissions are insufficient.

2. Create Room:

- **Usage:** createroom <room_name> <PERMS>
- **Description:** Requests the creation of a new chat room with the specified permission level.
- **Response:** Confirmation of request submission, or error if the room already exists.

3. List Rooms:

- **Usage:** listrooms
- **Description:** Lists all available chat rooms and their required permission levels.
- **Response:** List of room names and their permissions.

4. Leave Room:

- **Usage:** leave
- **Description:** Leaves the current chat room.
- **Response:** Confirmation message.

Messaging Commands

1. Send Message:

- **Usage:** <message> (while in a room)
- **Description:** Sends a message to all users in the current room.
- **Response:** Message broadcasted to the room.

2. Direct Message:

- **Usage:** msg <username> <message>
- **Description:** Sends a private message to a specific user.
- **Response:** Confirmation message, or error if the user is offline.

User Management Commands

1. List Users in Room:

- **Usage:** users
- **Description:** Lists all active users in the current chat room.
- **Response:** List of usernames.

2. Logout:

- **Usage:** logout
- **Description:** Logs the user out of the server.
- **Response:** Confirmation message.

Administrative Commands

1. Approve Room:

- **Usage:** approveroom <room_name>

- **Description:** Approves a pending room creation request.
- **Response:** Confirmation of approval.

2. Deny Room:

- **Usage:** denyroom <room_name>
- **Description:** Denies a pending room creation request.
- **Response:** Confirmation of denial.

3. List Room Requests:

- **Usage:** listroomrequests
- **Description:** Lists all pending room creation requests.
- **Response:** List of room requests, including requester and required permissions.

4. Notify Users:

- **Usage:** notify <PERMS> <message>
- **Description:** Sends a notification to all users with the specified permission level or higher.
- **Response:** Confirmation message.

Periodic Reporting

- **Interval:** Every 1 hour (configurable).
- **Content:**
 - Total users.
 - Active users.
 - Active user details (username and permission level).
- **Notification:** Sent to users with HIGH permissions.

Permissions Hierarchy

- **LOW:** Basic users.
- **MEDIUM:** Managers or operators.
- **HIGH:** Administrators with full control.

Permissions are checked for:

1. Room creation and access.
2. Command execution (e.g., room approvals).

Example Workflows

1. Basic User Joining a Room

1. User connects via Telnet:

```
telnet <server-ip> 42069
```

2. Registers or logs in:

```
register john password123
```

3. Joins a room:

```
join General
```

4. Sends a message:

```
Hello, everyone!
```

2. Admin Approving a Room

1. Admin logs in:

```
login admin adminpass
```

2. Lists pending room requests:

```
listroomrequests
```

3. Approves a room:

approveroom NewRoom

Error Handling

- **Unknown Commands:** Respond with Unknown command. Type help for a list of commands.
- **Invalid Arguments:** Respond with Invalid command. Usage: <correct usage>.
- **Permission Denied:** Respond with Insufficient permissions to execute this command.