CS594

Internet Draft

Intended Status: IRC Class Project Specification

Expires: May 2021

Internet Relay Chat Class Project

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79. This document may not be modified, and derivative works of it may not be created, except to publish it as an RFC and to translate it into languages other than English.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as InternetDrafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on May 30, 2021.

Copyright Notice

Abstract

This memo describes the communication protocol for an IRC-style client/server system for the Internetworking Protocols class at Portland State University.

Table of Contents

1. Introduction

This specification describes a simple Internet Relay Chat (IRC) protocol by which clients can communicate with each other. This system employs a server service that a client can use to request or add to a shared room message history.

Users can join rooms which are a shared message chain that can be requested by members.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying significance described in RFC 2119.

In this document, the characters ">>" preceding an indented line(s) indicates a statement using the key words listed above. This convention aids reviewers in quickly identifying or finding the portions of this RFC covered by these keywords.

3. Basic Information

All communication described in this protocol takes place over TCP/IP, with the server listening for connections on port 6667. Clients send requests to the server and the server responds with the requested data, or after making the requested change.

4. Message Infrastructure

4.1 Basic Message Structure

The message is an ASCII string encoded with C#'s Encoding Library. It consists of a four character request code, followed by a 4 digit user id, followed by an optional variable data string based on the type of request. The maximum size message received or sent by the server is 1024 bytes.

4.2 Request Codes

Message Code	Usage
NWID	request new user id, UserName in message, returns user id number to be used in further communication
CHLS	request a list of channels, no message needed, returns comma separated list of channels
CRCH	Create a new channel, channel name in message, returns if the channel was created successfully
JNCH	Join a channel, channel name in message, returns channel chat history
LVCH	Leave a channel, channel name in message, if the user is no longer in the channel or if the channel doesn't exist
CHCH	Request channel chat history, channel name in message, returns channel chat history
MEMB	Requests a list of members in a channel, channel name in message, returns comma separated list of members

PING	Checks if the server is responding, no message required, returns "Hello Client!"
MECH	Messages a channel, channel name,text in message, returns chat history for channel
GBYE	Signals ending of connection, no message required, returns "Bye!"

4.3 User ID Numbers

When a user first connects with a NWID Command, they are assigned an ID number, encoded as 4 characters via string parsing, ie: code 1 -> “0001”. This code must be used in all further communication to identify the user, with the exception of the NWID command, where the ID should be 0000, or the PING command, where the ID is not needed and any id number may be used. The user id number 0 is reserved for users who have not been assigned an ID yet.

4.4 Message Data

Message Type	Data field contents
NWID	Requested Alias to me associated with rooms and channels created by the user
CHLS	Not Required
CRCH	The name of the requested new channel
JNCH	Room name of the room that the client is trying to leave
LVCH	Room name of the room that the client is trying to leave
CHCH	Room name of the room that the client is trying to get the message history for
MEMB	Room name of the room that the client is requesting the members of
PING	Not Required
MECH	Room name to message, followed by the message, separated by ‘,’
GBYE	Not Required

4.5 Response Data

Message Type	Data Returned by the server
NWID	The User’s new ID as a string
CHLS	Comma separated list of rooms
CRCH	“Already Exists” if the channel exists or “Channel Created” if the channel was created
JNCH	The chat history of the room or “Channel Does Not Exist” if the room doesn't exist
LVCH	“User is no longer in channel” or “Channel Does Not Exist” if the channel doesn’t exist
CHCH	Returns the chat history of the channel, “User is not a member of this channel” if the user is not a member of the channel, or “Channel Does not exist” if the channel doesn't exist
MEMB	Returns a comma separated list of Aliases, “User is not a member of this channel” if the user is not a member of the channel, or “Channel Does not exist” if the channel doesn't exist
PING	“Hello Client” is always returned
MECH	The modified channel chat history, or “Channel doesn't exist” if the channel doesn't exist
GBYE	“Bye” is always returned

4.6 Room Chat History

The room chat history is stored as a string consisting of the opening message, followed by each message separated by line breaks. Each message consists of the sender’s Alias, followed by the chat message, separated by a colon and a space.

5 Conclusion and Future Work

This specification provides a barebones framework for client server interactions for the handling of a user chat service.

Future work may include using this specification to monitor the chat history of a channel in near real time.

6 Security Considerations

Messages sent using this system have no protection against inspection, tampering or outright forgery. Any client using another connected client’s user ID will appear to the server as the same user and will respond as such. Users seeking secure messaging should use or implement a derivation of this protocol, or another protocol.

7 IANA Considerations

None

8 Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

9 Acknowledgements

This document was based on the supplied RFC sample