# **F-Chat Client Commands**

From F-List Wiki

# **Commands sent by client**

# **ACB**

This command requires chat op or higher. Request a character's account be banned from the server.

**Syntax** 

<< ACB { "character": string }

# **AOP**

**This command is admin only.** Promotes a user to be a chatop (global moderator).

**Syntax** 

<< AOP { "character": string }

#### **AWC**

**This command requires chat op or higher.** Requests a list of currently connected alts for a characters account. **Syntax** 

\_\_\_\_\_

<< AWC { "character": string }

# **Contents**

- 1 Commands sent by client
  - 1.1 ACB
  - 1.2 AOP
  - 1.3 AWC
  - 1.4 BRO
  - 1.5 CBL
  - 1.6 CBU
  - 1.7 CCR
  - 1.8 CDS
  - 1.9 CHA
  - 1.10 CIU
  - 1.11 CKU
  - 1.12 COA
  - 1.13 COL
  - 1.14 COR
  - 1.15 CRC
  - 1.16 CSO
  - 1.17 CTU
  - 1.18 CUB
  - 1.19 DOP
  - 1.20 FKS
  - 1.21 IDN
  - 1.22 IGN
  - 1.23 JCH
  - 1.24 KIC1.25 KIK
  - 1.26 KIN
  - 1.20 KIIN
  - 1.27 LCH1.28 LRP
  - 1.29 MSG
  - 1.30 ORS
  - 1.31 PIN
  - 1.32 PRI
  - 1.33 PRO
  - 1.34 RLL
  - 1.35 RLD1.36 RMO
  - 1.37 RST
  - 1.38 RWD
  - 1.39 SFC
  - 1.40 STA
  - 1.41 TMO
  - 1.42 TPN

#### **BRO**

■ 1.43 UNB

■ 1.44 UPT

This command is admin only. Broadcasts a message to all connections.

# **Syntax**

```
<< BRO { "message": string }
```

#### **CBL**

This command requires channel op or higher. Request the channel banlist. Syntax

```
<< CBL { "channel": string }

Raw sample

CBL {"channel":"ADH-c7fc4c15c858dd76d860"}
```

# **Notes/Warnings**

This command does not have a unique response command from the server; A response is sent as SYS. As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

#### **CBU**

This command requires channel op or higher. Bans a character from a channel. Syntax

```
< CBU {"character": string, "channel": string}

Raw sample

CBU {"character": "Pas un Caractere", "channel": "ADH-c7fc4c15c858dd76d860"}
```

#### **Parameters**

channel: The channel from which the character is being banned.

character: The character being banned.

#### **Notes/Warnings**

As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

#### **CCR**

Create a private, invite-only channel.

# **Syntax**

```
<< CCR { "channel": string }

Raw sample

CCR {"channel":"test"}</pre>
```

#### **Notes/Warnings**

The channel param is used as the title for the newly created channel, not the actual channel name used to join the channel. The server will send the channel's actual ID in response.

# **CDS**

This command requires channel op or higher. Changes a channel's description. Syntax

```
<< CDS { "channel": string, "description": string }
```

#### Raw sample

CDS {"description": "[color=red]No actual roleplay in here. For discussion of anything at all, please go to the correct channels.[/color] This is the channel for RP offers and announcements.", "channel": "Looking for RP"}

#### **Notes/Warnings**

As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

# **CHA**

Request a list of all public channels.

#### **Syntax**

<< CHA

# **Notes/Warnings**

This is an argumentless command.

# CIU

**This command requires channel op or higher.** Sends an invitation for a channel to a user. **Syntax** 

```
<< CIU { "channel": string, "character": string }

Raw sample

CIU {"character": "Testytest", "channel": "ADH-011aeb5bb591b1f4721a"}
```

#### **Notes/Warnings**

This command does not have a unique response command from the server; a response is sent as SYS. As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

#### **CKU**

This command requires channel op or higher. Kicks a user from a channel. Syntax

```
<< CKU { "channel": string, "character": string }
```

# **Notes/Warnings**

As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

# **COA**

**This command requires channel op or higher.** Request a character be promoted to channel operator (channel moderator).

#### **Syntax**

```
<< COA { "channel": string, "character": string }
```

#### **Notes/Warnings**

As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

# COL

Requests the list of channel ops (channel moderators).

# **Syntax**

```
<< COL { "channel": string }
```

# **Notes/Warnings**

This command does not have a unique response command from the server; A response is sent as SYS. As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

#### **COR**

This command requires channel op or higher. Demotes a channel operator (channel moderator) to a normal user.

# **Syntax**

```
<< COR { "channel": string, "character": string }
```

# **Notes/Warnings**

As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

#### **CRC**

This command is admin only. Creates an official channel. Syntax

```
<< CRC { "channel": string }
```

# **CSO**

This command requires channel op or higher. Set a new channel owner. Syntax

```
<< CSO {"character":"string","channel":"string"}
```

# Raw sample

```
CSO {"character":"Jinni Wicked","channel":"ADH-3875a3c8c11325b49992"}
```

#### **Parameters**

- \* character: the character to set as owner
  - channel: which channel to set the owner in

# **Notes/Warnings**

This command is only implemented in F-Chat 1.0, at the time of this writing. As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

# **CTU**

**This command requires channel op or higher.** Temporarily bans a user from the channel for 1-90 minutes. A channel timeout.

#### **Syntax**

```
<< CTU { "channel":string, "character":string, "length":num }

Raw sample

CTU { "channel":"Frontpage", "character":"Treebob", "length":"30"}</pre>
```

#### **Parameters**

- \* channel: the channel from which to remove the character
  - character: the character to timeout.
  - length: the time, in minutes, to keep the character out of the room

#### **Notes/Warnings**

I have no idea how the server replies. Someone should probably grab me some logs.

#### **CUB**

**This command requires channel op or higher.** Unbans a user from a channel. **Syntax** 

```
<< CUB { channel: "channel", character: "character" }
```

# **Notes/Warnings**

This command does not have a unique response command from the server; A response is sent as SYS. As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

# **DOP**

**This command is admin only.** Demotes a chatop (global moderator). **Syntax** 

```
<< DOP { "character": string }
```

# **FKS**

Search for characters fitting the user's selections. Kinks is *required*, all other parameters are optional. **Syntax** 

```
<< FKS { "kinks": [int], "genders": [enum], "orientations": [enum], "languages": [enum], "furryprefs": [enum], "roles": [enum] }
```

#### Raw sample

FKS {"kinks":["523","66"],"genders":["Male","Maleherm"], "orientations":["Gay","Bi - male preference","Bisexual"], "languages":["Dutch"], "furryprefs":["Furs and / or humans","Humans ok, Furries Preferred","No humans, just furry characters"], roles:["Always dominant", "Usually dominant"] }

#### **Parameters**

kinks: identified by kinkids, available here (http://www.f-list.net/json/chat-search-getfields.json?ids=tr ue), along with the full list of other parameters.

genders: can be any of "Male", "Female", "Transgender", "Herm", "Shemale", "Male-Herm", "Cuntboy", "None"

orientations: can be any of "Straight", "Gay", "Bisexual", "Asexual", "Unsure", "Bi - male preference", "Bi - female preference", "Pansexual", "Bi-curious"

languages: can be any of "Dutch", "English", "French", "Spanish", "German", "Russian", "Chinese", "Japanese", "Portuguese", "Korean", "Arabic", "Italian", "Swedish", "Other"

furryprefs: can be any of "No furry characters, just humans", "No humans, just furry characters", "Furries ok, Humans Preferred", "Humans ok, Furries Preferred", "Furs and / or humans" roles: can be any of "Always dominant", "Usually dominant", "Switch", "Usually submissive", "Always submissive", "None"

#### IDN

This command is used to identify with the server.

#### **Syntax**

```
<< IDN { "method": "ticket", "account": string, "ticket": string, "character": string, "cname": string, "cversion": string }
```

#### **Parameters**

Ticket: A ticket can be aquired using the JSONP endpoint, see the JSON endpoints section of the documentation for details.

cname: The client's identifying name.

cversion: The client's identifying version

# **Notes/Warnings**

If you send any commands before identifying, you will be disconnected.

# **IGN**

A multi-faceted command to handle actions related to the ignore list. The server does not actually handle much of the ignore process, as it is the client's responsibility to block out messages it recieves from the server if that character is on the user's ignore list.

.....

# **Syntax**

```
<< IGN { "action": enum, "character": string }

Raw sample

IGN {"action": "add", "character": "Teal Deer"}
   IGN{"action": "delete", "character": "Teal Deer"}
   IGN{"action": "notify", "character": "Teal Deer"}
   IGN{"action": "list"}</pre>
```

#### **Parameters**

action

- add: adds the character to the ignore list
- delete: removes the character from the ignore list
- notify: notifies the server that character sending a PRI has been ignored
- list: returns full ignore list. Does not take 'character' parameter.

# **JCH**

Send a channel join request.

#### **Syntax**

------

```
<< JCH { "channel": string }

Raw sample

JCH {"channel": "Frontpage"}</pre>
```

# **Notes/Warnings**

As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

# **KIC**

**This command requires chat op or higher.** Deletes a channel from the server. **Syntax** 

```
<< KIC { "channel": string }
```

# **Notes/Warnings**

Private channel owners can destroy their own channels, but it isn't officially supported to do so.

# **KIK**

**This command requires chat op or higher.** Request a character be kicked from the server. **Syntax** 

```
<< KIK { "character": string }
```

# **KIN**

Request a list of a user's kinks.

### **Syntax**

```
<< KIN { "character": string }
```

# **Notes/Warnings**

This information can also be acquired from a JSON endpoint.

#### LCH

Request to leave a channel.

# **Syntax**

```
<< LCH { "channel": string }
```

# **Notes/Warnings**

As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

# LRP

Sends a chat ad to all other users in a channel.

#### **Syntax**

```
<< LRP { "channel": string, "message": string }

Raw sample

LRP {"message": "Right, evenin'", "channel": "Frontpage"}</pre>
```

#### **Notes/Warnings**

If you send more than one ad per ten minutes the message will not be sent, and an ERR will be returned. At this time of writing, there is a maximum length for LRP messages of 50000 bytes (effectively 50000 characters). You should however rely on VAR messages from the server to get the correct limit on your client. As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

### **MSG**

Sends a message to all other users in a channel.

#### **Syntax**

```
<< MSG { "channel": string, "message": string }
Raw sample</pre>
```

```
MSG {"message": "Right, evenin'", "channel": "Frontpage"}
```

# **Notes/Warnings**

If you send more than one message a second the message will not be sent, and an ERR will be returned. At this time of writing, there is a maximum length for MSG messages of 4096 bytes (effectively 4096 characters). You should however rely on VAR messages from the server to get the correct limit on your client. As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

# **ORS**

Request a list of open private rooms.

**Syntax** 

<< ORS

# **Notes/Warnings**

This is an argumentless command.

#### PIN

Sends a ping response to the server. Timeout detection, and activity to keep the connection alive. **Syntax** 

<< PIN

#### **Notes/Warnings**

You have to respond to pings or you will be disconnected, as that will be detected as a timeout. The server will try to get a ping response three times, each time waiting 30 seconds. So if you don't respond, you will be disconnected after 90 seconds. Sending multiple pings within ten seconds will get you disconnected also. This is an argumentless command.

\_\_\_\_\_\_

# **PRI**

Sends a private message to another user.

**Syntax** 

<< PRI { "recipient": string, "message": string }

# **Notes/Warnings**

There is flood control; the same as the MSG command. At this time of writing, the maximum length of a private message is 50000 bytes (effectively characters).

# **PRO**

Requests some of the profile tags on a character, such as Top/Bottom position and Language Preference.

# **Syntax**

```
<< PRO { "character": string }
```

# **Notes/Warnings**

This information can also be acquired from a JSON endpoint.

# RLL

Roll dice or spin the bottle.

# **Syntax**

```
<< RLL { "channel": string, "dice": string }

Raw sample

RLL {"channel":"ADH-dce8eb7af86213ac4c15","dice":"bottle"}

RLL {"channel":"ADH-dce8eb7af86213ac4c15","dice":"9d100"}

RLL {"channel":"ADH-c7fc4c15c858dd76d860","dice":"1d6+1d20"}

RLL {"channel":"ADH-c7fc4c15c858dd76d860","dice":"1d6+10000"}</pre>
```

#### **Parameters**

channel: Which channel the command is being used in

dice:

- bottle: selects one person in the room, other than the person sending the command.
- #d##: rolls # dice with ## sides, each.
- #d##+#d##: rolls more than one size of dice.
- #d##+##: adds a number (###) to the roll.

# **Notes/Warnings**

# can be any number 1-9. ## can be any number 1-500. ### can be any number up to 10000. It is possible to add up to 20 sets of dice and/or numbers. As with all commands that refer to a specific

channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

# **RLD**

This command requires chat op or higher. Reload certain server config files Syntax

```
<< RLD { "save": string }
```

# **Notes/Warnings**

Somebody who can actually use this command please correct this, and add the server response, if there is one?

# **RMO**

This command requires channel op or higher. Change room mode to accept chat, ads, or both. Syntax

```
<< RMO {"channel": string, "mode": enum}

Raw sample

RMO {"channel": "ADH-c7fc4c15c858dd76d860", "mode": "chat"}
```

#### **Parameters**

channel: Which channel is being changed.

mode:

- chat: Show only MSG.
- ads: Show only LRP.
- both: Show MSG and LRP.

#### **Notes/Warnings**

As with all commands that refer to a specific channel, official/public channels use the name, but unofficial/private/open private rooms use the channel ID, which can be gotten from ORS.

# **RST**

This command requires channel op or higher. Sets a private room's status to closed or open. (private Syntax

\_\_\_\_\_\_

```
<< RST { "channel": string, "status": enum }
   _____
```

#### Raw sample

```
>>RST {"channel":"ADH-06c3db8a4789498d6ae8","status":"public"}
```

#### **Parameters**

- \* channel: this should be the channel id
  - status: this can be 'public' or 'private'

# **Notes/Warnings**

This command does have a unique response command from the server.

# **RWD**

This command is admin only. Rewards a user, setting their status to 'crown' until they change it or log out.

# **Syntax**

```
<< RWD { "character": string }
```

# **SFC**

Alerts admins and chatops (global moderators) of an issue.

#### **Syntax**

```
<< SFC { "action": "report", "report": string, "character": string }
```

#### **Parameters**

action: the type of SFC. The client will always send "report".

report: The user's complaint

character: The character being reported

# **Notes/Warnings**

The webclients also upload logs and have a specific formatting to "report". It is suspected that thirdparty clients cannot upload logs.

#### STA

Request a new status be set for your character.

#### **Syntax**

```
<< STA { "status": enum, "statusmsg": string }
     Raw sample
   ______
```

```
STA {"status": "looking", "statusmsg": "I'm always available to RP:)", "character": "Hexxy"}
```

#### **Parameters**

Status: Valid values are "online", "looking", "busy", "dnd", "idle", "away", and "crown".

# **Notes/Warnings**

Crown is a special value, and should not be sent by a client, as it is set by the RWD command. Idle is an automatic status set by some clients, when the user has not interacted with the client for a certain amount of time.

#### **TMO**

This command requires chat op or higher. Times out a user for a given amount minutes. **Syntax** 

```
<< TMO { "character": string, "time": int, "reason": string }
```

Time: an integer value of the timeout duration in minutes, being a minimum of one minute, and a maximum of 90 minutes.

# **Notes/Warnings**

This is an account-wide timeout from the chat, not a channel timeout for a character.

#### **TPN**

"user x is typing/stopped typing/has entered text" for private messages.

#### **Syntax**

```
<< TPN { "character": string, "status": enum }
```

#### Raw sample

TPN {"character":"Leon Priest", "status": "clear"}

#### **Parameters**

status: can have a value of "clear", "paused", and "typing".

# **Notes/Warnings**

It is assumed a user is no longer typing after a private message has been sent, so there is no need to send a TPN of clear with it.

# **UNB**

**This command requires chat op or higher.** Unbans a character's account from the server. **Syntax** 

```
<> UNB { "character": string }
```

# **UPT**

Requests info about how long the server has been running, and some stats about usage. **Syntax** 

```
<< UPT

Raw sample

UPT
```

#### **Notes/Warnings**

This command takes no parameters.

F-Chat	
Clients	F-Chat 1.0   F-Chat 2.0   F-Chat 3.0   F-Chat Pidgin/Adium Plugin   slimCat   Official List of Third-Party Clients   Rules for Chat Bots
For Users	F-Chat Command Reference   List of BBCode Tags   Smileys   Chat Help   Chat Troubleshooting   Websockets Troubleshooting   Chat Rules   F-Chat Official Channels
For Developers	Protocol   Error Codes   Client Commands   Server Commands   JSON Endpoints

Retrieved from "https://wiki.f-list.net/index.php?title=F-Chat\_Client\_Commands&oldid=2589"

Categories: Developer Resources | F-Chat

■ This page was last edited on 3 April 2017, at 00:15.