

Jabber Perl Bot

Usermanual

Kahless

Jabber Perl Bot: Usermanual

by Kahless

Copyright © 2001 by Kahless

Get latest version of documentation for the Jabber Perl Bot at <http://jpb.sourceforge.net> (easier:
<http://jpb.sf.net>)

Table of Contents

1. Installing	1
1.1. Requirements	1
1.2. Configuring	1
1.2.1. \$server.....	1
1.2.2. \$username (have to be changed).....	1
1.2.3. \$pass (have to be changed)	1
1.2.4. \$owner (have to be changed)	2
1.2.5. \$debug.....	2
1.2.6. \$save.....	2
1.2.7. \$filename.....	2
1.2.8. \$savefile	2
1.2.9. \$ipctimeout	2
1.2.10. \$Data::Dumper::Indent	2
1.2.11. \$IPCcom	3
1.2.12. \$IRCbot.....	3
1.2.13. \$stdIRCnick (should be edited)	3
1.2.14. \$localaddr.....	3
1.2.15. Modules.....	3
2. Using.....	4
2.1. pls (Owner functions)	4
2.1.1. rosteradd <jid>	4
2.1.2. subscribe <jid>.....	4
2.1.3. add <jid>	4
2.1.4. rosterremove <jid>.....	4
2.1.5. unsubscribe <jid>.....	4
2.1.6. remove <jid>.....	4
2.1.7. save.....	5
2.1.8. load.....	5
2.1.9. exit.....	5
2.1.10. setflags <jid> <flags>.....	5
2.1.11. IRCconnect <server> [<Nick>]	5
2.1.12. IRCquit.....	5
2.1.13. IRCjoin <channel>.....	5
2.1.14. IRCpart <channel>.....	6

2.2. sendmessage.pl.....6

Chapter 1. Installing

This is just a small guide on how to install Jabber Perl Bot.

This documentation does NOT describe how to download and extract the archive, I think you can do it yourself ;)

1.1. Requirements

- Perl (Tested with v5.6.1)
- Net::Jabber (Tested with 1.0023)
- IPC::ShareLite (Tested with 0.08)

Well.. I hope thats all ;)

1.2. Configuring

First of all you should rename the baseconfig.pl to config.pl

Then edit the config.pl and change the variables. If the description in the config.pl is not enough for you, here is something longer:

1.2.1. \$server

The server to which the bot should try to connect.

1.2.2. \$username (have to be changed)

The Username which the bot should use to login (If the login is not successfull, the bot SHOULD try to register the nick)

1.2.3. \$pass (have to be changed)

The Password which the bot should use to login.

1.2.4. \$owner (have to be changed)

The JID (Jabber Identification) of the owner (of you).

1.2.5. \$debug

Activates Debug mode (0 = deactivated / 1 = activated) - You may want to activate the debug mode because the bot has no other output yet.

1.2.6. \$save

This is for the status indicator function of the bot. If you activate saving (1 = activated) the bot will save the status of every JID in the roster into a file in the format:

```
jid<#>status<#>message<#>last change<#>
```

1.2.7. \$filename

If you have activated saving the status, the bot will write the status of all users into this file.

1.2.8. \$savefile

This file is to save and load data for the modules you loaded.

1.2.9. \$ipctimeout

This is for IPC (Inter Process Communication). E.g.: If \$ipctimeout is 30 the bot will wait 30 seconds until he writes 'ready' into the shared memory. (means that the programm, which wants to communicate with the bot has 30 seconds to read the answer of the bot.)

1.2.10. **\$Data::Dumper::Indent**

This variable just controls the style of the output of the Debug messages (for hashes). If you have no idea about perl, you may not change this flag and if you want to know more look at the Data::Dumper documentation: http://search.cpan.org/doc/GSAR/Data-Dumper-2.101/Dumper.pm#Configuration_Variables_or_Methods.

1.2.11. **\$IPCcom**

With this variable you can activate (1) or deactivate (0) the Inter Process Communication.

1.2.12. **\$IRCbot**

With this variable you can activate (1) or deactivate (0) the IRC bot.

1.2.13. **\$stdIRCnick (should be edited)**

What nick the bot should use on the IRC

1.2.14. **\$localaddr**

The Hostname of your PC. It is used by Net::IRC and I do not really know how it works, but if you have no hostname (like me) you can set this to 1 and the Net::IRC ignore it. If you set this to 0 the Net::IRC will read the system configuration

1.2.15. **Modules**

You may not edit this as user. Just if you want to add a module you have to add a sbind or bindm but this command should be in the documentation of every module.

Chapter 2. Using

After configuring the bot you can just run it with: `./bot.pl`

2.1. pls (Owner functions)

Owner functions are sent as a chat message via the jabber server. They always start with `pls` and then the command. (Your JID have to be `$owner`)

2.1.1. rosteradd <jid>

Adds <jid> to the roster.

2.1.2. subscribe <jid>

This sends a subscribe request to `jid` (Just use it if someone has not allowed the bot to see his online status.)

2.1.3. add <jid>

This basically calls `rosteradd` and `subscribe`. So you should always call `add` not `rosteradd`

2.1.4. rosterremove <jid>

Like `rosteradd` you shouldn't use it.

2.1.5. unsubscribe <jid>

Sends a `unsubscribe` to <jid>. (You shouldn't use it without `rosterremove`)

2.1.6. remove <jid>

Basically just calls rosterremove and unsubscribe. Normally You should allways use this command to remove a jid.

2.1.7. save

Opens \$savefile and calls every save() method of every module.

2.1.8. load

Opens \$savefile and calls every load() method of every module.

2.1.9. exit

Calls the save method and exits then.

2.1.10. setflags <jid> <flags>

Sets the flags of <jid> to <flags>

2.1.11. IRCconnect <server> [<Nick>]

Connects to <server>. If there is no nick the bot will connect as \$stdIRCnick. (The bot can connect to multiple servers.)

2.1.12. IRCquit

Disconnects from the active server. (May there is more documentation at the IRC module)

2.1.13. IRCjoin <channel>

Joins <channel> on the active IRC server

2.1.14. IRCpart <channel>

Parts from <channel> on the active IRC server

2.2. sendmessage.pl

Usage: ./sendmessage.pl <jid> <message>

This sends the message <message> to <jid>. Not really usefull, it is just a test for the Inter Process Communication.