

YSM ICQ Client Usage Guide

Updated for version: 2.8.1

by..rad2k

Contents

- About YSM
 - ◆ What is YSM?
 - ◆ Supported Operating Systems
 - ◆ YSM Features
 - ◆ Where to get YSM
 - ◆ YSM Vocabulary
- Installing YSM
 - ◆ System Requirements
 - ◆ Installing from sources
 - ◆ Installing from binaries
- Configuring YSM
 - ◆ Pre-Compiling Configuration
 - ◇ Languages
 - ◇ Status Change Flood Protection
 - ◇ YSM War Mode
 - ◇ Slash Messaging Mode
 - ◆ Post-Compiling Configuration
 - ◇ The Configuration file
 - Initial Status ["STATUS>"]
 - UIN Number ["UIN>"]
 - The Password ["PASSWORD>"]
 - ICQ Server Information ["SERVER>" and "SERVERPORT> "]
 - AFK Timeout and AFK Disable ["IDLETIME>"]
 - Default AFK Message ["AFKMESSAGE>"]
 - ReadAFK Amount of Messages Displayed ["AFKMAXSHOWN> "]
 - HTTP/HTTPS Proxy Configuration ["PROXY>" "PROXY_PORT> " and "PROXY_HTTPS>"]
 - Init Contacts Download ["ASK_DOWNLOAD>"]
 - Beeping ["BEEP>"]
 - Amount of times to Beep ["BEEPAMOUNT>"]
 - Global Logging Activation ["LOGALL>"]
 - Incoming Message Displaying Format ["MSG_KIND> "]
 - Verbosity / Debug Levels ["VERBOSE>"]
 - YSM Version Checking ["VERSION_CHECK>"]
 - Language Charset Conversions ["CHARSET_TRANS>" and "CHARSET_LOCAL>"]
 - Browser launching path ["BROWSER> "]

- Incoming Messages Alert Modes for Ms. Windows ["WINALERT> "]

- Using YSM

- ◆ The YSM Commands

- ◇ Configuration Commands

- Setting your Nickname["nick"]
 - Setting your E-mail address ["email"]
 - Changing your ICQ password ["password"]
 - Logging configuration["log"]
 - Auto AFK configuration["afkset"]
 - Beeping ["beep"]

- ◇ Messaging Commands

- Send a message to a slave or ICQ number ["msg or m"]
 - Send a message to the last slave you sent a message to ["a"]
 - Send a message to the last slave who messaged you ["r"]

- ◇ Slave/ICQ# Commands

- Add a new slave to your slaves list ["add or addslave"]
 - Remove a slave from your slaves list ["del or delslave"]
 - Store your ICQ slaves in the ICQ servers ["save"]
 - Authorize a slave to add you to his/her contact list ["auth"]
 - Rename a slave ["rename"]
 - Print your entire slaves list ["w or slaves"]
 - Print your online slaves (not with ignore) ["wo or slavesonline"]
 - Print your online slaves (with ignore) ["wa"]
 - Request/View information about a slave or ICQ number ["whois"]
 - Search on the ICQ whitepages using an e-mail address ["search"]
 - Add or remove a slave to/from your ICQ ignore list ["ignore"]
 - Add or remove a slave to/from your ICQ invisible list ["invisible"]
 - Add or remove a slave to/from your ICQ visible list ["visible"]
 - Set an YSM encryption key with another YSM encryption user ["key"]
 - Enable or disable the Alert mode for a slave ["alert"]
 - Get Time information on a slave ["seen"]

- ◇ Client Commands

- Change or view your current status ["status"]
 - Print information about your YSM client and account ["info"]
 - Enable or disable the AFK mode ["afk"]
 - read messages that were logged while you were in the AFK mode ["readafk"]
 - Clear the screen ["clear or c"]
 - Check the amount of days/hours/minutes/seconds since you started YSM ["uptime"]
 - Print the last received message ["last"]
 - Read the results of the 'log' command for a particular slave ["hist"]
 - Execute shell commands ["!"]
 - Launch a browser for a URL ["burl"]
 - Forward incoming messages to another UIN ["forward"]
 - A stupid joke :) ["backdoor"]

YSM Manual

- Exit YSM ["quit or q"]

◇ Hot Keyboard Keys

- The default "1 2 3 4" keymap
- The TAB key
- The ESC key
- UP/DOWN arrows

◆ Hints and Tricks!

- Credits

About YSM

What is YSM?

YSM stands for "You Sick Me" an ironic name I came up with while surfing the "I Seek You" (ICQ) protocol. YSM is an Open Source ICQ client under the GPL License. Originally coded for the Linux operating system but, as time goes by, it's being ported to different platforms.

The next Manual section should list known supported platforms.

Since Mirabilis joined the Dark Side of the Force, and next to AOL decided to develop an AIM Compatible Protocol so that they wouldn't loose their current amount of ICQ users and fame+fortune, ..a new version of the ICQ protocol was born. The version Number is known as V7. The last known and used ICQ Protocol Version was v5 since v6 was just a transition to v7.

Between Versions 5 and 7, the protocol suffers Important and 'unforgiveable' changes. The old used UDP is replaced by TCP, turning any pre-existing clients in a useless state. Besides the communication protocol, data structures have been shaped into FLAPs, SNACs and the use of TLV's. I highly recommend a deep read of AIM's Protocol Description by Scott Werndorfer that you can find <in [aimdoc-website](#)>.

Anyway, thanks to TCP we are now able to implement SSL (Secure Sockets Layer) and go through some Internet "roads" we weren't really able to go through before (such as HTTPS), in the UDP era. Most Microsoft Windows users must have noticed v7 once ICQ 2000 was released. Along the interesting features, v7 brought a new friend to town, called "ads spam".

A question flows.. is the upgrade to v7 a 'MUST do' ? Many programmers have spent precious human time programming icq clones that can be found anywhere, is it really necessary to code new icq clients?. The answer is 'Yes'. Mirabilis has been leaving old protocol versions aside and, nowadays, v5 packets are being lost and some ignored, this leads to broken login sessions, lost messages (mostly offline) , etc...

By the time I'm writing this YSM Manual (1st May 2002), a series of v7/v8 ICQ clients showed up. Most of these (at least c++/perl based) depend on libicq2000 (a new project by Barnaby Gray, in c++) and mod icq2000 (a perl library by Robin Fisher.)

Since by the time I started to code YSM (Dec. 2001) there were no C/C++ known v7 projects, I decided to begin coding a small implementation of the new protocol basing myself on a reverse done by Massimo Melina that can be found next to his RQ delphi client at <http://rejetto.com/>. About one or two months later, Ickle is released (Barnaby's C++ gtk client) with it's own protocol implementation too.

I wanted a simple console client that required no extra libraries to compile out of my YSM project, but I was unsure about it's future as I knew

YSM Manual

more clients were coming out (such as the new version of centerICQ by Konst, which depends on Barnaby's library). As YSM was already in a usable state and I had done the coding split in different tidy modules, I realised YSM could turn into an ICQ C library to be used by any C language programmer. I was amazed of the possitive reply I got from mostly everywhere, and that's what kept me on road.

I continued working on the YSM Client in order to improve my protocol code and at the same time continue the development of the unique C language ICQ clone. My idea of simplicity let YSM compile on linux, BSD, Windows, BeOS and Qnx! (at least those were really tested, maybe there are more platforms). Mostly YSM users were in charge of the small modifications each port required.

After a deep research on a pretty undocumented side of the ICQ protocol, YSM was proudly the first ICQ clone for it's supported platforms to Download and Upload Server side contact lists.

So that's it. YSM is still being developed, and by now it's the first/only C language client around. I'll start working on the YSM library soon. Just trying to help!.

Supported Operating Systems

YSM is known to run in :

1. **Linux** : Using the ./configure shellsript.
2. **BSD** : Using the ./configure shellsript.
3. **Windows** : Has it's "ysm.dsp" for Visual C++, and a Makefile.win for Borland.
4. **BeOS** : Has it's Makefile.BeOS
5. **QNX** : Using the ./configure shellsript.
6. Know of another operating system in which YSM runs? [mail me](#)

These micros were 32 bit x86. I once received a mail from an MIT lab with a 64 bits patch. If you know of other than x86 platforms, please mail me the details.

For futher instructions on Installing YSM in these platforms, read the "Installing YSM" Manual Section.

YSM Features

These are the features I like most about my YSM client :

1. Crypto between YSM clients using a rijndael algorithm implementation [AES in CBC mode] [using the 'key' command].
(Note: YSM Encryption fully supports data confidentiality and it's implementation is known to be vulnerable to Message integrity attacks, which are really harmless. An attacker after having hijacked your tcp connection with the ICQ server (hard to achieve) will be able to repeat messages [replay attacks, repetition attacks, etc..] after a few successfull sniffings. This note is just for information and you shouldn't really care since the consequence of a succesfull attack is harmless, **the attacker will NOT be able to read the data in your messages.**)
2. Downloading server-side stored ICQ contacts.
3. Uploading/Saving in the ICQ servers local ICQ contacts [using the 'save' command].
4. ICQ Clients Fingerprinting, makes possible the recognition of remote ICQ clients [shown in the 'whois' command].

5. The AFK mode.
[read about the 'afk' command].
6. Quick and easy to compile code, leaving aside –must have– library requirements.
7. Full support for the Windows operating system, plus some extra YSM Windows features.
8. Console based, meaning no mouse usage is required :)
9. Multi Language support.

Where to get YSM

YSM has it's Internet website at <http://ysmv7.sourceforge.net/> .
It's also listed in freshmeat, at <http://freshmeat.net/projects/ysmicq/>

For an updated list of sites linking YSM check the Community section in YSM's website.

If anyone knows of any other places from where to download YSM and/or would like to submit YSM somewhere, please do so and [mail me](#) about it!

YSM Vocabulary

These are the definitions of the words used in the YSM ICQ Client that may not have an instant meaning to an YSM user :

Slave: An ICQ Contact.

AFK: "away from keyboard", is one of the YSM features. AFK may either be activated manually or automatically (when a specified amount of minutes without any keyboard input is reached).

When AFK is activated, a series of steps are followed:

- An auto–response message is sent to anyone who messages you.
- The message is logged (readable with the 'readafk' command).
- In case you were in "FREE4CHAT" or "ONLINE", your status is changed to "AWAY".

Mood: Status. Mood may either be "ONLINE", "AWAY", "DND", "FREECHAT", "NA", "OCCUPIED" or "INVISIBLE".

Installing YSM

System Requirements

YSM's idea of portability is responsible for the short length of this section :).

Running requirements

A little faith, YSM should run as fast as wind in old systems, no particular version of libraries are required.

Building requirements

Unix like systems require 'make' in order to use the included Makefiles.

Windows users will need a Microsoft Visual C++ compiler.

Important Note on Threads, Must read.

The last version of YSM supports 'pthreads' (Posix Threads) and Windows native threads. For non Windows operating systems, if the "configure" shellsript detects support for 'pthreads', the "YSM_WITH_THREADS" constant will be defined. You should not worry about this constant, since **if your system lacks of pthreads support YSM will compile without any wining in a single threaded mode** disabling a few features.

Installing from sources

Pre Installation Notes

The sources for all supported OS is a unique ".tgz" file (Tar + Gzip compression).

If you still haven't downloaded this file, don't read any further, go back to the "About YSM" Manual Section for downloading instructions.

These are the steps to follow on both unix like systems and Windows:

Unix like systems:

1. tar zxvf ysmv7_x_x_x.tgz
2. cd ysmv7_x_x_x/
3. ./configure
4. make
5. Optionally, as root : make install

Steps 1 to 4 will generate an ysm executable binary file in the src/ directory.

Step 5 will install the ysm binary and its manual page in your system.

Windows systems:

1. Use Winzip or any uncompression tool for uncompressing "ysmv7_x_x_x.tgz".
2. Double click on "ysm.dsp" or open your Visual C++ program and choose File, Open Project, then go to the "ysmv7_x_x_x/" directory and double click on the .dsp file.
3. Press 'F7' to start compiling.

Post Installation Notes

The first time you run YSM, a configuration file will be created. In Windows systems, I've decided to use the \$USERPROFILE environment variable as root path for the creation of the YSM directory (Where the configuration file and extra files are stored.) In case USERPROFILE fails to exist, C:\ is used as the root Path.

On unix like systems, the root path is the user's Home directory (~/).

The default name for the configuration file is "ysm-cfg" . Windows systems will store this file under a directory called "ysm" and unix like systems under ".ysm" (Note the '.' (dot) before its name).

Attention: an **alternate** configuration path/name can be used by specifying "-c path_to_config" to the ysm binary.

An example would be: `./ysm -c ysm-cfg` for using a local configuration file.

For further instructions on the Configuration file, read the "Configuring YSM" Manual Section destined to describe the basic configuration file parameters.

Installing from binaries

If you downloaded a pre-compiled version of YSM, just read the "Post Installation Notes" in the previous "Installing from sources" Manual subSection.

Configuring YSM

YSM has a two stops road for configuration. Starting with a Pre-Compiling Configuration (usually left with defaults), and ending with a Post-Compiling Configuration (possibly the one you are looking for).

Pre-Compiling Configuration

The Pre-Compiling Configuration allows Multi-Language support, between some other interesting/required options. All of them are described in the following sub sections:

Languages

YSM has Multi-Language support. Although it is not configurable in runtime, it's what I've been able to implement so far. Thanks to the YSM users, many translations have been already done basing in an original english version (By the time I'm writing this: 2 French translations, Italian, Portuguese (Brasil), Spanish, German, Russian and Swedish).

In order to choose the language you want, edit the "YSM_Lang.h" file inside the "src" directory and make sure you leave the desired language definition (i.e.: #define SPANISH) out of any comments and alone (important for not generating conflicts).

Status Change Flood Protection

The Status Change Flood Protection is nothing more than a comfortable policy which, is indeed, activated by default. What the policy takes care of is not showing simple status changes of any slaves, hence the screen is clean of constant status changing floods.

The policy permits only "Offline to Anything" and "Anything to Offline" status changes to be shown in the screen.

In case you want to disable the Flood Protection policy, you will need to remove the line that reads "#define YSM_SILENT_SLAVES_STATUS" from the "YSM.h" file you will find inside the "src" directory.

YSM War Mode

The YSM War Mode is an undocumented project. For further assistance you will have to read YSM's source code. The War Mode is disabled by default. No, I'm not a bofh, I'm affraid of huge 4 meters tall company lawyers.

Slash Messaging Mode

Continuing with 'comfortable features', here is the "Slash Messaging Mode". Since the default way of messaging a slave is "m <name|uin> data" and a big percentage of YSM users were old mICQ users, the Slash Messaging Mode will turn messaging into "m <name|uin>/data" .

Since the Slash Messaging Mode is –disabled– by default, in order to enable it you will need to edit the "YSM.h" file inside the "src" directory, and move the "#define YSM_SLAVE_SLASH_MESSAGE" line out of the "#if 0 #endif" lines.

Post-Compiling Configuration

Post-Compiling Configuration is an easy task in YSM. Most Settings are stored in a configuration file and the rest can be done in runtime inside YSM (Check the next YSM Commands section).

The Configuration File

Although each version of YSM usually has new configuration fields, most of these configurable settings have stored default values. This means old configuration files fully work with new YSM releases. If you need to set a parameter that is missing in your current version of the configuration file, just add the required parameter between the beginning of the file and before the [SLAVES] tag. Comments are preceeded with a "#" sign.

These are the up-to-date available configuration settings:
(Both ' ' ' should be removed when used in the configuration file.)

Initial Status | "STATUS>"

The "STATUS>" setting let's the user choose his/her status to be set right after logging in the ICQ

service.

Its value may be any of the available Status states. (Check the vocabulary section the word "mood").

Example: *STATUS>NA*

UIN Number | "UIN>"

The UIN Number is your ICQ number. This setting shouldn't be of your worry since YSM takes care of filling the parameter after asking you the first time you run YSM.

Example: *UIN>63873081*

The Password | "PASSWORD>"

This is the password for the previously specified ICQ number (or UIN).

Instead of specifying your plaintext password, you may specify the word "SECURE" in uppercase.

By specifying "SECURE" you will be prompted for your password everytime you start YSM.

Example: *PASSWORD>SECURE*

ICQ Server Address | "SERVER>"

This is the address of the remote ICQ server. In case you need to use another server, just replace the

default address (login.icq.com) with the one you want.

Example: *SERVER>login.icq.com*

ICQ Server Port | "SERVERPORT>"

This is the port of the remote ICQ server. You shouldn't worry about setting this parameter.

The default used is 5190.

Example: *SERVERPORT>5190*

AFK Timeout and AFK Disable | "IDLETIME>"

This sets the amount of minutes to wait with no keyboard input before turning into the Automatic AFK mode. You may also disable Auto-AFK by specifying 0 minutes.

Example: *IDLETIME>10* or *IDLETIME>0*

Default AFK Message | "AFKMESSAGE>"

This will be the default message sent in the auto-reply message. The message will be used either when AFK is turned on manually specifying no message or when Auto-AFK turns on.

Example: *AFKMESSAGE>I'm not here right now, sorry. This message is sent automatically. Your message has been logged.*

ReadAFK Amount of Displayed Messages | "AFKMAXSHOWN>"

This is the amount of messages shown on every page displayed by the "readafk" command.

The default value is '3'.

Example: *AFKMAXSHOWN>3*

HTTP/HTTPS Proxy Configuration | "PROXY>"

This is the IP address of the HTTP/HTTPS proxy to use. If you don't wish to use a Proxy, just specify a Zero as value.

Example: *PROXY>192.168.10.2* or *PROXY>0*

HTTP/HTTPS Proxy Configuration | "PROXY_PORT>"

This is the Port number of the Proxy in case you specified a Proxy address.

Example: *PROXY_PORT>666*

HTTP/HTTPS Proxy Configuration | "PROXY_HTTPS>"

This is the flag that tells YSM whether your proxy is an HTTPS proxy or not.

If your proxy is HTTPS, set the value to "1", else set it to "0".

Example: *PROXY_HTTPS>1* or *PROXY_HTTPS>0*

Init Contacts Download | "ASK_DOWNLOAD>"

If its value is "1" you will be prompted for downloading your server-side stored contacts everytime you start YSM.

If set to 0, contacts will be downloaded without asking.

Its default value is 0.

Example: *ASK_DOWNLOAD>0*

Beeping | "BEEP>"

If its value is "1", a certain amount of BEEPS will be present when receiving messages, alerts on users, etc.

The amount of times to beep is configured using the "BEEPAMOUNT>" setting in the configuration file.

If its value is "0", Beeping will be disabled. BEEP can be enabled/disabled in runtime using the "BEEP" command.

Its default value is 1.

Example: *BEEP>1*

Amount of times to Beep | "BEEPAMOUNT>"

This variable sets the amount of times to beep when the BEEP variable is on.

Beeping can also be activated/deactivated using the "BEEP" command in runtime.

Its default value is 1.

Example: *BEEPAMOUNT>1*

Global Logging Activation | "LOGALL>"

This activates logging for every slave in your list. It will generate a file in your YSM directory named by the UIN of the slave where it will store the incoming/outgoing data. The data will then be readable by the "hist" command.

Example: *LOGALL>1*

Incoming Message Displaying Format | "MSG_KIND>"

This sets the format displayed in the screen of incoming messages.

There are three (3) message formats available:

0 – Displays a two lines message. The first line is a complete timestamp. The second line shows the Slave name if the remote user is a slave, the UIN, remote Status and data length next to the incoming message data.

1 – Displays a three lines message. Timestamp, remote info and data.

2 – Displays an IRC style message.

The default used is "2".

Verbosity / Debug Levels | "VERBOSE>"

The default level is "2".

These verbosity/debug Levels allow a quick YSM debugging possible by increasing the verbosity level.

They also let the user remove certain strings which may be considered 'extra' in normal usage by lowering the verbosity level. These are the verbosity levels :

0 – Base Verbosity. Required for basic output.

1 – Functional. Some Important messages are shown in this level.

2 – Extra. Is the default Level. Has some extra information, some users might want to lower to 1. (I wouldn't :P)

6 – Debug 1. Shows incoming SNAC and FLAP information plus some server side notices.

7 – Debug 2. Shows incoming packets dump, downloaded Slave additions and sent keep alive packets.

8 – Debug 3. Not yet implemented.

9 – Debug 4. Not yet implemented.

Example: `VERBOSE>2`

YSM Version Checking | "VERSION_CHECK>"

Version Checking allows an up-to-date version check on every YSM startup.

It's activated by default. If a Proxy is configured, Version checking gets disabled.

Example: `VERSION_CHECK>1`

Language Charset Conversions | "CHARSET_TRANS>"

If you require a special charset for sending messages, it's name should be specified here.

Russian Generic for Trans is "CP1251".

Example: `CHARSET_TRANS>CP1251`

Language Charset Conversions | "CHARSET_LOCAL>"

If you require a special charset for receiving messages, it's name should be specified here.

Russian Generic for Local is "KOI8R".

Example: `CHARSET_LOCAL>KOI8R`

Browser launching path | "BROWSER>"

This variable has the path to your browser that will be launched using the "BURL" command later on when running YSM. If you want to enable the "BURL" command, put the full path to your browser.

Examples of "BROWSER" values are:

BROWSER>/usr/local/bin/mozilla

BROWSER>/usr/bin/lynx

BROWSER>c:\progra~1\ie\explore~1\ie\explore.exe

If set to 0, the "burl" command will be disabled.

It's default value is 0.

Example: BROWSER>0

Incoming Messages Alert Modes for Ms. Windows | "WINALERT>"

YSM when compiled for Win32 has support for different Alerting types of new incoming messages. The default used is "3".

Type "1" pops up the minimized console Window (if minimized)

Type "2" only blinks in the task bar.

Type "3" pops up the minimized console Window (if minimized) and blinks the window in the task bar.

Example: WINALERT>3

Using YSM

Even though using YSM is a very simple task (hehe), I've decided to open the mind of the YSM users and explain them in detail how to get this program working. I've split this section in three different subsections, one describing the commands, another for the keyboard hot keys and the last one with tips and hints on how to fully become an YSM lover (leaving subliminal messages aside).

The YSM Commands

Configuration Commands

The following is a description of commands I've identified as 'configuration related' commands. [The following rule doesn't apply to the '**nick**' and '**email**' commands since their values are stored in the ICQ servers] : The values modified using these commands are NOT stored in the configuration file. If you want to make permanent changes to one of these settings, set its value in the configuration file using its corresponding option previously described in the last subsection called "**The Configuration File**".

Setting your Nickname | "nick"

The "nick" command is used to show or change your current ICQ Nickname.

If no arguments are supplied, the action taken is 'show your current nickname'.

Usage: nick [new_nick_name]

Setting your E-mail address | "email"

The "email" command is used to show or change your current E-Mail Address. If no arguments are supplied, the action taken is 'show your current e-mail addy'.

Usage: email [new_email_address]

Changing your ICQ password | "password"

The "password" command is used to change your real ICQ password.

Note: This will change your REAL account password. You will have to update your password in the configuration file manually (in case you dont have SECURE on).

Usage: password new_password

Logging configuration | "log"

The "log" command allows the activation of either Global logging or logging on a specific slave. The data logged is outgoing and incoming messages to/from a certain slave. A different log file is created for each slave being logged, and the results are readable using the "**hist**" command or just by editing any of the generated files in your YSM directory. As stated before in the previous subsection called "**The Configuration File**", Global logging is activated by default.

Global logging can be enabled/disabled by using either "**ON**" or "**OFF**" as arguments.

For enabling/disabling logging on a specific slave, you must supply a slave nickname to the log command.

If no arguments are supplied, the current settings are shown.

Usage: log ON|OFF or log [SLAVE_NICK]

Auto AFK configuration | "afkset"

The "afkset" command can be used for enabling/disabling the Automatic AFK mode and also for setting the amount of minutes to wait without any keyboard input before entering into the Automatic AFK mode.

The Auto-AFK mode can be enabled/disabled using either "**ON**" or "**OFF**" as arguments.

For setting a new amount of minutes for Auto-AFK, use a number as argument.

If no arguments are supplied, the current settings are shown.

Usage: afkset ON|OFF or afkset [amount_of_minutes]

Beeping | "beep"

The "beep" command is used to enable or disable 'beeping' on incoming messages.

Beeping can be enabled/disabled by using either "**ON**" or "**OFF**" as arguments.

If no arguments are supplied, the current beep setting is shown.

Usage: beep [ON|OFF]

Messaging Commands

The following commands are used for sending messages either to a slave or a non slave ICQ number.

Messaging in the YSM Client has two (2) modes. If no message data is specified, YSM enters in a comfortable message mode where you end a message with a "." (dot) in a clean line, or cancel the message with a "#" in a clean line too. Messaging gets to be more easy by using the TAB key described later in the "Hot Keyboard Keys" sub-section.

Send a message to a slave or ICQ number ["msg or m"]

The "msg" or "m" command is used to send an ICQ message to a slave or a UIN.

Usage: msg slave_name [message] or msg ICQ_NUM [message]

Send a message to the last slave you sent a message to ["a"]

The "a" command is used to message the last slave you sent a message to.

Usage: a [message]

Send a message to the last slave who messaged you ["r"]

The "r" command is used to reply to the last message you received from a slave.

Usage: r [message]

Slave/ICQ# Commands

The following commands are used to administrate your slaves list (or contacts list) and they should cover all your contact list needs. From adding new slaves to requesting information or adding to your invisible/visible/ignore lists, etc.

Add a new slave to your slaves list ["add or addslave"]

The "add" or "addslave" command is used to add a new slave to your slaves list.

Usage: add slave_name slave_uin or addslave slave_name slave_uin

Remove a slave from your slaves list ["del or delslave"]

The "del" or "delslave" command is used to remove a slave from your slaves list.

Usage: del slave_name or delslave slave_name

Store your ICQ slaves in the ICQ servers ["save"]

The "save" command is an important YSM feature called "YSM Political Asylum for Slaves". It stores local slaves in the ICQ servers in a remote group called "YSM" that is created if it doesn't yet exist. The importance of this feature can be more appreciated from a 'backup' point of view, since in case there is a problem and you loose your configuration file with your slaves list, all server side stored contacts will be automatically downloaded anytime you start YSM and saved back in your configuration file.

If no parameters are given to the save command, the entire list will be stored in the server in groups of 5, so you will have to call save again after the first 5 finished storing to store the next 5 slaves. The save command also accepts a slave name as parameter, in this case, it will store the selected slave.

Usage: save [slave_name]

Authorize a slave to add you to his/her contact list ["auth"]

The "auth" command is used to send an OK authorization to a slave or a UIN.

Usage: auth slave_name or auth ICQ_NUM

Rename a slave ["rename"]

The "rename" command is used to rename an existing slave in your configuration file.

Usage: rename current_slave_name new_slave_name

Print your entire slaves list ["w or slaves"]

The "w" or "slaves" command is used to print all your existing slaves list.

The format of the output is :

[slave_name slave_status flags] sorted by their current status.

The flags are 3:

IG (meaning its in your ignore list)

IN (meaning its in your invisible list)

and VI (meaning its in your visible list)

The flags are modified by the 'ignore' 'invisible' and 'visible' commands respectively.

In case a slave has a Birthday, a BIRTHDAY identifier will show up next to his nick.

Both "w" and "wo" commands if followed by a string, will display only matches taking the string as a pattern.

Usage: w [pattern] or slaves [pattern]

Print your online slaves (not with ignore) ["wo or slavesonline"]

The "wo" or "slavesonline" command is used to print your online slaves.

The format of the output is the same as the "w" command described before.

The same goes for the arguments.

Note: Slaves in your IGNORE list, are NOT shown using "wo".

The "wa" and "w" commands instead, show them.

Usage: wo [pattern] or slavesonline [pattern]

Print your online slaves (with ignore) ["wa"]

The "wa" command acts the same way as the "wo" command but also displays the slaves in your IGNORE list.

The format of the output is the same as the "w" command described before.

The same goes for the arguments.

Usage: wa [pattern]Request/View information about a slave or ICQ number ["whois"]

The "whois" command shows information about an existing slave or, if specified a UIN number instead, about a nonslave (yet? :-D) ICQ user. The output of an existing slave shows the same information a non-slave output shows, adding some extra fields. The fields shown in both are: *Nick, First, Last, e-Mail, Age, Sex, City and State*. The output for an existing slave also has: *IP Address[internal], IP Address[external], listening port and Client Fingerprint*. In case the slave has a Birthday, a greeting shows up.

Usage: whois slave_name or whois ICQ_NUMSearch on the ICQ whitepages using an e-mail address ["search"]

The "search" command is used to search the ICQ whitepages using an e-mail address as reference.

If a match is found, the results will show the UIN and nick of the remote user.

Usage: search email@addressAdd or remove a slave to/from your ICQ ignore list ["ignore"]

The "ignore" command is used to add or remove a slave to/from your ICQ Ignore list. By adding a slave to this list, the server will filter all incoming messages from the slave. YSM takes one step forward (again? :-D) and automatically adds the same slave to your ICQ Invisible list so the slave isn't aware of you being online. Before being added to your ignore list, the slave must be stored in the ICQ servers using the 'save' command which was described in detail before. *The "ignore" command called once* will add a slave to the ignore list, if called again it will remove the slave from such list. You may identify a slave in ignore by using the "w" or "wo" commands and looking at the flags next to the status of a slave [more information about the flags in the "w" command].

Usage: ignore slave_nameAdd or remove a slave to/from your ICQ invisible list ["invisible"]

The "invisible" command is used to add or remove a slave to/from your ICQ Invisible list. By adding a slave to this list, the server won't notify the slave of your ICQ presence unless you also add the slave to your Visible list by using the "visible" command described next. Before being added to your invisible list, the slave must be stored in the ICQ servers using the 'save' command which was described in detail before. *The "invisible" command called once* will add a slave to the invisible list, if called again it will remove the slave from such list. You may identify a slave in invisible by using the "w" or "wo" commands and looking at the flags next to the status of a slave [more information about the flags in the "w" command].

Usage: invisible slave_nameAdd or remove a slave to/from your ICQ visible list ["visible"]

The "visible" command is used to add or remove a slave to/from your ICQ Visible list. Usually the visible list is used when you are in invisible status, invisible to everyone, and you wish to be seen by a certain slave, that way you add him/her to your ICQ Visible list. Before being added to your visible list, the slave must be stored in the ICQ servers using the 'save' command which was described in detail before. *The "visible" command called once* will add a slave to the visible list, if called again it

will remove the slave from such list. You may identify a slave in invisible by using the "w" or "wo" commands and looking at the flags next to the status of a slave [more information about the flags in the "w" command].

Usage: visible slave_name

Set an YSM encryption key with another YSM encryption user ["key"]

The "key" command is used to set an encryption key to be used between 2 (two) YSM clients. One of the YSM Clients (Slave A) has to specify or generate a key and send it to the other Client (Slave B) (usually not over icq, but in person, over the phone, encrypted over mail, etc). The other YSM Client (Slave B) then sets the same key with slave A and the conversation is encrypted. The key is stored in the configuration file so no further processing is required.

As recently stated, the key can be specified or generated randomly. If after specifying the slave name to the key command you specify a question mark '?', a key will be generated randomly and set for the specified slave name. For specifying a key, instead of using the question mark '?', specify a key. **The key can be maximum 32 bytes long and each byte must be in hexadecimal.** For example the key *AABBCCDDEEFF112233* is valid, while the key *AAJJKKLLMM* is not, since the digits J, K, L and M aren't valid in hexadecimal. **In order to identify incoming encrypted messages**, you will notice two "*" (asterisk) chars next to the remote slave name in IRC kind incoming messages. By using '?' as a parameter not only a key is generated but it is also set. The "key" command always requires a slave name as the first parameter. If the second parameter is not specified (which is either the '?' or the key), then the key for the specified slave is cleared.

Usage: key slave_name [key] or key slave_name [?]

Enable or disable the Alert mode for a slave ["alert"]

The "alert" command is used to set a slave into an "alert mode" that will generate an alert 'beep' sound and display an 'alert message' on any status change the specified slave goes through. The same command enables/disables the alert mode a slave.

Usage: alert slave_name

Get Time information on a slave ["seen"]

The "seen" command displays the "Signon time", "Last status Change time", and "Last Received Message time" of a specified slave.

Usage: seen slave_name

Client Commands

The following commands affect your YSM icq session.

Change or view your current status ["status"]

The "slave" command is used to change your current ICQ status.

You may change to any of the following: ONLINE – FREECHAT – AWAY – NA – DND – INVISIBLE.

It also allows a pattern to be specified as an argument instead of a full status name. For example: specifying "on" instead of "online" will also change your status to online.

If no arguments are supplied, your current status is shown.

Usage: status [new_status] or status [new_status_pattern]

Print information about your YSM client and account ["info"]

The "info" command is used to show valuable information about your YSM icq session.

The information displayed is: **Your UIN # – Your current Status – The Amount of existing Slaves – The last Sequence Number – Beeping setting if active – Logall setting if active – Your local IP Address – Your ICQ NickName – Your ICQ FirstName – Your ICQ LastName – Your ICQ E-mail – YSM Client version – Threads support status.**

Usage: info

Enable or disable the AFK mode ["afk"]

The "afk" command is used to manually enable/disable the AFK Mode. For a definition of the AFK mode, check the Vocabulary section, at the beginning of this manual. If no argument is specified to the AFK command (when enabling AFK) a default AFK message is used (defined as AFKMESSAGE in the configuration file). A message can be specified as an argument.

Usage: afk [message]

read messages that were logged while you were in the AFK mode ["readafk"]

The "readafk" command is used to administrate the messages that were stored/received while your YSM client was in AFK mode. The amount of messages shown each time is configured in your configuration file, under the option named AFKMAXSHOWN. It's recommended to clear the messages after having read them all (using the keys shown in the menu at the bottom).

Usage: readafk

Clear the screen ["clear or c"]

The "clear" or "c" command is used to clear the screen.

Usage: clear or c

Check the amount of days/hours/minutes/seconds since you started YSM ["uptime"]

The "uptime" command is used to show the amount of days, hours, minutes and seconds that passed since you started your YSM session.

Usage: uptime

Print the last received message ["last"]

The "last" command is used to show the last received message by any of yours slaves. This means messages by non-slave users (users out of your list) arent stored.

Usage: last

Read the results of the 'log' command for a particular slave ["hist"]

The "hist" command is the implementation of the so known "ICQ Messages History". This command reads the message logs generated by the 'log' setting. If you want to have history on every user, you should set the LOGALL setting in your configuration file to "1" (activated).

Usage: hist slave_name

Execute shell commands ["!"]

The "!" command is used to execute commands in your current shell. Unlike other applications, the argument must be 'one space' separated from the command. The output of the specified command is shown in the same console.

Usage: ! command

Launch a browser for a URL ["burl"]

The "burl" command launches a browser for a specified URL. The browser must be configured in your configuration file using the "BROWSER>" variable. Right after executing the browser, control goes back to your client.

Usage: burl URL

Forward incoming messages to another UIN ["forward"]

The "forward" command is used to redirect incoming messages to an alternate UIN. The uin of the sender is stored right after the data so the final receiver knows who the sender was. You may either specify a UIN or a slave name. If no arguments are supplied, the forward command clears its settings.

Usage: forward [UIN] or forward [Slave_name]

A stupid joke :) ["backdoor"]

The "backdoor" command prints just a joke :). Dont believe me? read the source!.

Usage: backdoor

Exit YSM ["quit or q"]

The "q" or "quit" command is used to logoff from the ICQ network and leave the YSM session. In summary, used to exit the YSM client.

Usage: quit or q

Hot Keyboard Keys

These are the keys you should learn in order to make your YSM ICQ session more comfortable.

The default "1 2 3 4" keymap

There is a default keymap for four (4) YSM commands which are 'help, wo, w and afk'. The default mapping sets the following:

key: **1** action: **help**
key: **2** action: **wo**
key: **3** action: **w**
key: **4** action: **afk**

The TAB key

The TAB key is one of the most useful hot-keys in YSM. It will react differently depending on any of the following situations:

When typing a slave name in a command: It will complete the rest of the name.
After receiving a message from a slave: It will generate a reply body to the slave.
After sending a message to a slave: It will generate a new message body to the same slave.
Without typing anything: It will cycle with a message body through all online slaves.

The ESC key

The ESC key is used to clear the current active prompt line. It is also used to cancel messages which are being written in a non-comfortable mode (in oneline mode?).

The UP/DOWN arrows

The UP and DOWN arrow keys are used to cycle through the commands history. The history stores up to 15 commands before it starts re-using the buffers.

Hints and Tricks!

Hello and welcome to the YSM Hints and Tricks section! ..now, let the show begin. These are the tips the author itself gives you in order to enjoy your YSM in a way you never did before!

Read about the AFK mode in the vocabulary section and the AFK related commands in the YSM Commands manual section in order to understand what the AFK mode is, and decide whether you want to use it or not. Why am I saying this..well, the automatic AFK is by default activated in 10 minutes (as set in the IDLETIME> configuration file parameter) of keyboard inactivity and configuring it or even disabling it might be a good idea.

Get in touch with the keyboard Hot-Keys which were described in the "Hot-Keys" manual section. These keys will speed up your usage incredibly and probably many problems you had will get solved.

Save your favorite slaves in the ICQ servers using the "save" command described in the YSM Commands section. If anything happens and you lose your configuration file, the slaves that were stored in the ICQ servers will be automatically downloaded the next time you start YSM.

Use YSM Encryption with your YSM friends right now! Why should you send important information in plain-text over the internet?! Tell all your friends who are using YSM to set an encryption key with you and hide your conversations from the Big brother family!.

Learn about the YSM for Win32 features that will turn your Win32 session into heaven! (Check the WINALERT> Configuration file setting, the "z" command, etc.).

Credits

rad2k [rad2k at mail dot ru] [ICQ 63873081]

EOF
the end.