# OfficeIRC v2.1 Technical Manual

## SUMMARY

This manual describes how to configure your chat server, connect to your server and secure your chat server. It also details trouble shooting, Internet Relay Chat (IRC) commands, Extensions to the Internet Relay Chat Protocol (IRCX) commands and frequently asked questions.

## TABLE OF CONTENTS

1. Introducing OfficeIRC Server ........................................................... 2
2. Configuring your chat server ............................................................ 3
   2.1. Status (Connection Monitor) ............................................ 3
   2.2. General Settings ............................................................ 4
   2.3. Client Rules ....................................................................... 5
   2.4. Operators .......................................................................... 7
   2.5. Operator Groups ............................................................... 8
   2.6. Security .............................................................................. 9
   2.7. Advanced Security ............................................................ 10
   2.8. DNS Blacklist ..................................................................... 11
   2.9. Port Scanner ...................................................................... 12
   2.10. Version Check .................................................................. 13
   2.11. Filtering ............................................................................. 14
   2.12. Channels ........................................................................... 16
   2.13. Channels Advanced ......................................................... 18
   2.14. Nicknames ......................................................................... 18
   2.15. Nicknames Advanced ...................................................... 19
   2.16. News Flashes .................................................................... 19
   2.17. Memorandums .................................................................. 20
   2.18. Transcripts ......................................................................... 20
   2.19. Chat Back ........................................................................... 20
   2.20. Messages ........................................................................... 21
   2.21. Server Linking ................................................................... 21
3. IRC Command Reference ................................................................ 22
4. Channel Modes ..................................................................................... 32
5. User Modes ........................................................................................... 34
6. Chat Services ........................................................................................ 36
   6.1. Channel Properties ........................................................... 36
   6.2. Nickname Properties ........................................................ 37
7. Remote Administration ....................................................................... 37
8. Frequently Asked Questions ............................................................ 38

## 1. Introducing OfficeIRC Server

OfficeIRC is an Internet Relay Chat Server (IRC) that supports the Extensions to the Internet Relay Chat Protocol (IRCX). Features and services provided make administration easy and more effective. The security and reliability make your users chat experience more enjoyable.

the chat experience for your users a more enjoyable one.

Administering your chat server is done through the Remote Control utility. This utility can be used to connect locally or remotely allowing you to run your chat server at one place and administer it anywhere.

Securing your chat server has never been so easy. It is no fun for your clients or for yourself having to deal with people trying to flood the server with text messages or trying to disrupt other users.

### *Main Security Features*

- Multiple Connection Limit
- Connection Throttling
- Server / Channel Attack Protection
- Client Rules
- Local and Network Wide bans
- In/Out Flood Detection
- IP/DNS Masking
- IP Spoof Protection
- Insecure Proxy Scanner
- Version Check
- Message Filtering

### *Supported Chat Protocols*

- Commands defined by RFC 1459 standard (IRC)
- Commands defined by the extension to RFC 1459 referred to as Extensions to the Internet Relay Chat Protocol (IRCX)

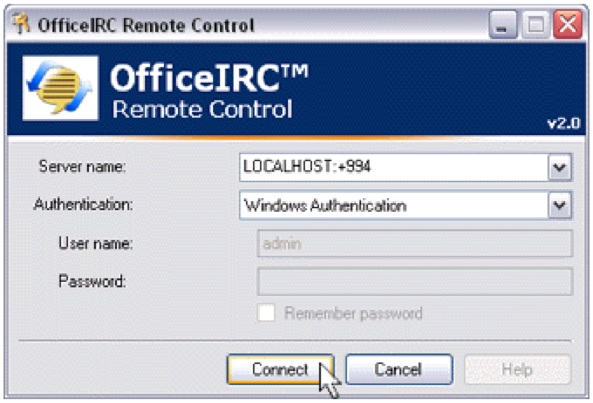
### *Client software tested for compatibility*

- mIRC
- PIRCH
- Klient
- TurboIRC
- Microsoft Chat
- Microsoft V-Chat

## 2. Configuring your chat server

Use the Remote Control utility provided to configure your chat server.  You can access it by using the 'Remote Control' shortcut found in the Start Menu.

When you run the Remote Control for the first time, a Wizard will help you name your server and create an administrator account.



Server name: LOCALHOST:+994
Authentication: Windows Authentication
User name: admin
Password:
☐ Remember password

Specify the hostname or IP address of the server you wish to administrate in the 'Server name' field.  To specify a specific chat port, use a colon followed by the port number.

*e.g. chat.officeirc.com:6667*

If the chat port is SSL encrypted, specify a plus sign after the colon.

*e.g. chat.officeirc.com:+994*

There are two types of Authentication available when connecting:

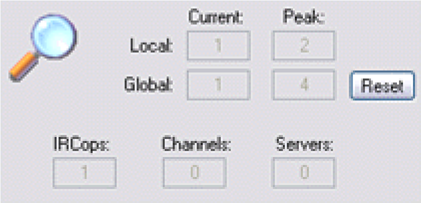
- Windows Authentication (NTLM)
- OfficeIRC Authentication

You can login locally using 'Windows Authentication' if you are currently logged into your workstation under an Administrator account.  Alternatively select 'OfficeIRC Authentication' and use the credentials of the account you created using the Wizard to login remotely.

OfficeIRC Authentication exchanges passwords in plain-text.  For improved security you should connect using an SSL encrypted connection or use 'Windows Authentication' whenever possible.

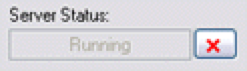
### *2.1 Status (Connection Monitor)*

Status area allows you to monitor the number of open channels, users and servers connected, server processor usage (total usage on the machine), and server uptime.



| | Current: | Peak: |
|---|---|---|
| Local: | 1 | 2 |
| Global: | 1 | 4 |

| IRCops: | Channels: | Servers: |
|---|---|---|
| 1 | 0 | 0 |



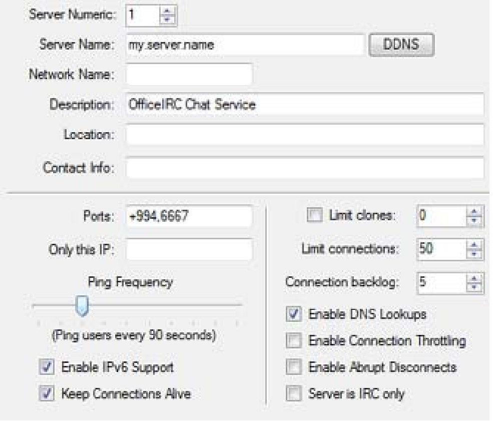
Server Status: Running

You can remotely shutdown the chat server using the button next to Server Status.

## 2.2 General Settings

General Settings is used to configure the chat server's identify, TCP/IP chat ports to which the server will accept connections and various other options explained in detail below.

Changes to the Server Name, Ports, 'Only this IP', and 'Abrupt Disconnects' require a restart for the changes to take effect.


Server Numeric: 1
Server Name: my.server.name
Network Name:
Description: OfficeIRC Chat Service
Location:
Contact Info:
Ports: +994,6667
Only this IP:
Ping Frequency
(Ping users every 90 seconds)
☑ Enable IPv6 Support
☑ Keep Connections Alive
☐ Limit clones: 0
Limit connections: 50
Connection backlog: 5
☑ Enable DNS Lookups
☐ Enable Connection Throttling
☐ Enable Abrupt Disconnects
☐ Server is IRC only

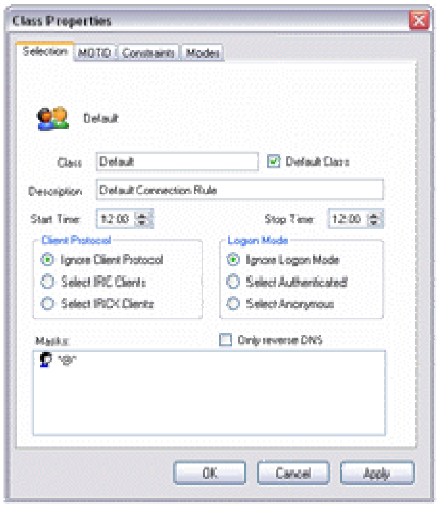
| | |
|---|---|
| Server Numeric | A unique identifier for the chat server used during Server Linking communication. |
| Server Name | A valid DNS hostname that points to the chat server's IP address. |
| DDNS | Automatically updates a dynamic DNS hostname to point to the chat server's IP address. You will need to register with a dynamic DNS provider to use this functionality. |
| Network Name | If part of a network, the network acronym or network short name |
| Description | Short description of server e.g. Joe Blog's Chat Service |
| Location | Optional, physical location of the chat server e.g. Town, Country |
| Contact Info | Administrative contact information e.g. Email, phone or URL. |
| Ports | Designates TCP ports used for accepting chat connections. When specifying multiple ports, use commas as separators. To specify an SSL port, prefix the port number with a plus symbol e.g. +994,6667 |
| Only this IP | If server has multiple network cards or is assigned multiple IP addresses, you can bind the chat server to a single IP address. |
| Ping Frequency | How often the server will send pings to make sure the connection still exists. |
| Enable IPv6 Support | Allows the server to accept connections over an IPv6 network. |
| Keep Connection Alive | Prevents long connections from timing out in case of inactivity. |
| Limit clones | Limits multiple users from the same IP address. |
| Limit connections | Number of users that may be connected on this chat server before it is considered full. |
| Connection backlog | This is the maximum number of connections allowed to wait to establish a connection. |
| Enable DNS Lookups | Clients reverse DNS address will appear instead of their IP address |
| Enable Connection Throttling | Connections will be throttled to prevent users from reconnecting too quickly to the server. |
| Enable Abrupt Disconnects | Forces an abrupt (hard) disconnection instead of connections being terminated gracefully. |
| Server is IRC only | Disables Extensions to the Internet Relay Chat Protocol (IRCX). Not recommend due to loss of functionality. |

## 2.3 Client Rules

Client Rules (Classes) allow you to treat users differently based on the selection criteria. This can be useful in a number of ways. For example, you can greet clients in a different language based on their domain, or give limited access to everyone except a specific ISP.

Users are assigned to the first Client Rule they match based on the selection criteria, beginning from the top of the list. If a user fits no Client Rule, then the server will refuse the connection.

Client Rules can be added, modified and deleted from the Client Rules area.



Default

Class: Default ☑ Default Class

Description: Default Connection Rule

Start Time: 12:00 Stop Time: 12:00

Client Protocol
- ● Ignore Client Protocol
- ○ Select IRC Clients
- ○ Select IRCX Clients

Logon Mode
- ● Ignore Logon Mode
- ○ Select Authenticated
- ○ Select Anonymous

Masks: ☐ Only reverse DNS

"*@*"

*Client Rule Properties*

| | |
|---|---|
| Class | Name of the Class (Client Rule) |
| Default Class | Use class for Remote Users and master copy of MOTD. |
| Description | Brief description. |
| Start Time / Stop Time | Time window in which the Class will operate in. |
| Client Protocol | Select the users by their Client Protocol. |
| Logon Mode | Select if the user has been authenticated or is anonymous. Anonymous clients are marked with a '~' prefix in their ident. |
| Only reverse DNS connections | Select only users who have a reverse DNS on their IP address. |
| Masks | Lists of masks for selecting users that apply to this rule (If no masks are supplied, will select anyone). Double-click to add a new mask and single-click to modify or delete an existing mask.<br>Supported Masks<br>• <userid>@<hostname><br>• <userid>@<hostname>:<chat_port><br>• <userid>@<hostname>$<server_address><br>• <userid>@<hostname>$<server_address>:<chat_port><br>Wildcards can be used in Masks |
| MOTD | Message of the Day, typically used to display information such as server news and rules. |
| Use Default Class | If you do not want to duplicate your Message of the Day, you can use this option and the clients will see the MOTD stored on the default class instead. |
| Attack Protection | Creates an artificial delay (in seconds) before the server will process the next message. |

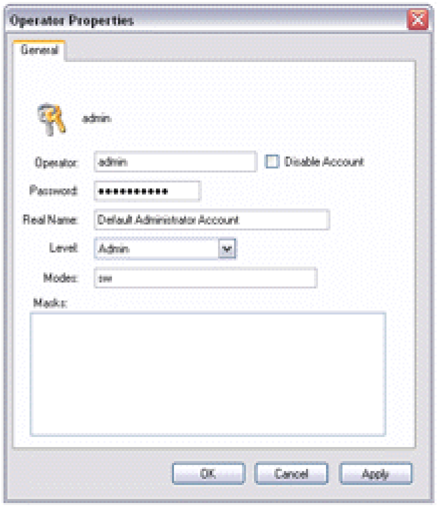
| | Messages | Invitation | Join | Wrong Password | Messages (with host) |
|---|---|---|---|---|---|
| Low | 1 | 2 | 2 | 2 | 1 |
| Medium | 2 | 4 | 3 | 4 | 1 |
| High | 3 | 5 | 4 | 5 | 2 |

| | |
|---|---|
| Cannot Log on | Cannot connect to the server. |
| Cannot Create Dynamic Channels | Cannot create an unregistered channel. |
| Cannot Join Dynamic Channels | Clients can only join registered channels. |
| Cannot be Owner/Host in Channel | Clients cannot be given control of a channel. |
| Cannot Change Nickname | Clients cannot change nickname after logging on. |
| Cannot Send Private Messages | Clients can only send messages to channels. |
| Cannot Send Receive Messages | Clients cannot receive private messages. |
| Maximum Channels | The maximum number of channels a client can join at once. |
| Class Connection Limit | Maximum users who can connect at one time using this class. |
| Nickname Delay | Time delay a client must wait before they can change their nickname. |
| Message Delay | Creates an artificial lag for each command issued by the client. |
| Password Protected | Allows you to protect your server from unauthorized access. You can choose to use a shared password or restrict access to only registered nicknames. |
| Use IP Address for identify | IP Address will be used for clients identify even if a valid hostname was returned from DNS Lookups. |
| Authenticate Class Members | Authenticates anonymous clients selected by the rule. |
| Static Domain | Gives clients selected by this rule an identify cloak. |
| Initial Language | Automatically selects the preferred spoken language. |
| Does not wish to send or receive DCC | Disable secure chat and file sending. |
| Does not wish to send or receive CTCP | Disable client software queries. |
| Invisible to users outside of channels | Hides the clients from appearing in listings. |
| Filter private messages being received | Enables content filtering for the clients. |
| Other Modes | Used to apply miscellaneous mode chars. (See User Modes) |
| Lock Modes | Prevents the removal of specified mode chars e.g. +L will prevent the Content Filtering being turned off. |
| Alternate Welcome | Alternate Welcome Message to include the Server's Description. |
| Send LUSERS | Automatically display local statistics upon connecting. |
| Send MOTD | Automatically displays the MOTD upon connecting. |
| Auto Join Channels | Force clients to join the specified channel upon connecting. Multiple channels can be specified by using commas. |
| Cannot Register Nicknames | Prevents clients from registering new nicknames. |
| Cannot Register Channels | Prevents clients from registering new channels. |

## *2.4 Operators*

Operators are clients you appoint to help control and police your chat server. Operators have access to special commands to aid them in their duties. Clients who violate your server policies or disrupt the service can be forcibly removed and banned from reconnecting by Operators. There are different types of Operator Levels allowing you to assign only the needed access to perform required tasks.

To gain Operator Status, an account must first be created then, the user simply needs to either login using these credentials or use the OPER command.



Operator: admin
□ Disable Account
Password: •••••••••••
Real Name: Default Administrator Account
Level: Admin
Modes: sw
Masks:

*Operator Properties*

| | |
|---|---|
| Operator | Name of Operator Account. |
| Disable Account | Prevents account from being used. |
| Password | Password used for authentication. |
| Real Name | Account holder's real name (optional). |
| Level | Operator Group the account has been assigned to. This determines the privileges the account has. By default, the server has 6 groups to choose from. |

| | Wallops | FNick/Move | Gag/Shun | Kill/Ban | Channels | Registrations | Network Controls | Remote Access |
|---|---|---|---|---|---|---|---|---|
| Administrator | X | GLOBAL | GLOBAL | GLOBAL | OWNER | X | X | X |
| Global Operator | X | GLOBAL | GLOBAL | GLOBAL | OWNER | X | X | - |
| System Manager | X | LOCAL | LOCAL | LOCAL | OWNER | X | - | - |
| System Operator | X | LOCAL | LOCAL | LOCAL | HOST | - | - | - |
| IRC Operator | X | LOCAL | LOCAL | LOCAL | - | - | - | - |
| Help Operator | X | - | - | - | - | - | - | - |

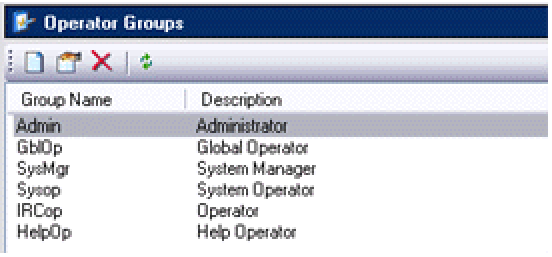
| | |
|---|---|
| Modes | Additional User Modes to set upon receiving Operator Status e.g. sw (To receive wallop and server messages). |
| Masks | For security, the account can be restricted to only work from specific IP addresses by defining a list of user masks. If no masks are supplied, anyone who supplies the correct password will authenticate. Double-click to add a new mask and single-click to modify or delete an existing mask.<br>Supported Mask<br><nick>!<userid>@<hostname> |

Most IRC Client software have an area for you to enter your IRC Operator credentials. If this is not available, the first part of your email address usually forms your UserID (<account_name>@domain) and you specify your password in the Server Password field. Authenticating in this manner will allow you to bypass server bans and give you operator status upon connecting.

If you have already connected, you can use the OPER command.

## 2.5 Operator Groups

Operator Groups allows you to create and modify groups which are used to define the privileges each Operator Level has to perform actions on users and channels locally on the server or across the network.



| Group Name | Description |
|---|---|
| Admin | Administrator |
| GblOp | Global Operator |
| SysMgr | System Manager |
| Sysop | System Operator |
| IRCop | Operator |
| HelpOp | Help Operator |

*Operator Group Properties*

| | |
|---|---|
| Group Name | Name of Operator Group. |
| Access Level | Prevents Server Operators from removing server and network access entries if they have a lower access level. |
| Dynamic / Registered Channels | Status granted to the operator upon entering channels. |

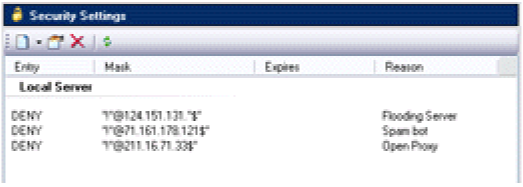
| | |
|---|---|
| None | Treated like a regular user upon entering. |
| Permit | Forced entry.  Bypass limits, bans etc. |
| Host | Receive host status upon entering channel. |
| Owner | Receive owner status upon entering channel. |

| | |
|---|---|
| Administrator | Ability to manage operator accounts, set special channel modes, full access to channel access, remote shutdown of server, and other miscellaneous rights. |
| Remote Access | Ability to change server environment settings and login using the Remote Control utility. |
| Global Operator | Allows enforcement commands to be performed on non-local users. |
| Network Controls | Can manage server links using CONNECT and SQUIT commands. |
| Help Assistant | Can view messages sent using the HELPOPS command. |
| Event Usage | Allows use of the EVENT command. |
| Wallops Usage | Allows the send and receive of WALLOPS messages. |
| Bypass Restrictions | Allows the bypassing of restrictions such as Client Rule restrictions, maximum access, filtering, watch list, max channels, registration limits, etc. |
| Show All Channels | Allows hidden and secret channels to be visible in the channel list. |
| Show All Users | Allows invisible users to be seen in user lists. |
| View IP Addresses | Displays clients IP Addresses when masking is enabled. |
| Kill User | Allows the forced disconnect of clients from the server. |
| Kill Channel | Allows channels to be shutdown using the KILL command. |
| Ban User | Allows the banning of clients from connecting to the server. |
| Zap User | Allows the adding and removal of local server ZLINE entries. |
| Force Nick | Allows the force nickname change of clients. |
| Force Join | Allows the force joining of a client to a specific channel. |
| Gag User | Allows the gagging of clients to prevent sending of messages. |
| Mode Hop | Allows the use of the SAMODE command. |
| Move User | Allows the forced move of a client to a specific channel. |
| Shun | Allows the adding and removal of local server shuns. |
| Nickname Services | Allows registering of nicknames and ability to update details. |
| Channel Creation | Allows the creation of dynamic channels. |
| Nickname Services | Allows registering of channels. |
| Transcript Access | Allows the viewing and deleting of audit and transcript logs. |

## 2.6 Security

Security area allows you to create, modify and delete local or network wide bans.

To protect your users, you can use Flood Detection and IP/DNS Masking.



| Entry | Mask | Expires | Reason |
|---|---|---|---|
| **Local Server** | | | |
| DENY | "^*@124.151.131.*$" | | Flooding Server |
| DENY | "^*@71.161.178.121$" | | Spam bot |
| DENY | "^*@211.16.71.33$" | | Open Proxy |

<table>
<tr><td>Local Bans</td><td>Bans clients from connecting locally to your chat server.<br><table><tr><td>Deny (KLINE)</td><td>Bans a client from connecting to your chat server.</td></tr><tr><td>ZLINE</td><td>Blocks an IP connection from being established to any TCP/IP chat port.</td></tr><tr><td>Grant</td><td>Allows an exception to a ban that targets multiple clients.</td></tr></table></td></tr>
<tr><td>Network Wide Bans</td><td>Bans clients from connecting to your chat network.<br><table><tr><td>Deny (AKILL)</td><td>Bans a client from connecting to any chat server on the network.</td></tr><tr><td>Grant</td><td>Allows an exception to a ban that targets multiple clients.</td></tr></table></td></tr>
<tr><td>Flood Detection</td><td>Drops a connection when an inbound flood is detected.  Each time a client sends a message within the ‘minimum seconds’ period, the client receives a penalty.  If the client receives more penalties then the ‘maximum penalties’, the client will be disconnected.  Penalties are reset once every 60 seconds.</td></tr>
<tr><td>Automatically K-Line offenders</td><td>Bans clients detected as flooding from returning for a specified duration.<br><br><i>There are built-in safety checks to prevent the flood protection being abused by organized groups of clients targeting a victim using CTCP requests.</i></td></tr>
<tr><td>IP/DNS Masking</td><td>Cloaks IP Addresses to prevent hackers from targeting clients to initiating Denial of Service (DoS) attacks, etc.<br><table><tr><td>None</td><td>Disables IP/DNS masking of client addresses.</td></tr><tr><td>Low</td><td>Masks IP Addresses of everyone except from host / owners of channels.</td></tr><tr><td>High</td><td>Masks IP addresses of everyone.</td></tr><tr><td>Encrypted</td><td>Uses irreversible encryption to mask the full IP addresses of clients.</td></tr></table></td></tr>
</table>

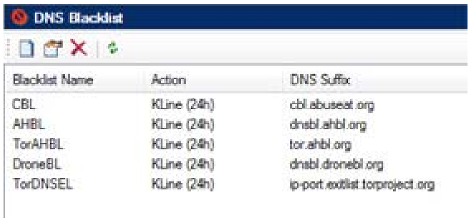
## 2.7 Advanced Security

| | |
|---|---|
| Enable Visual Confirmation | Prevents bots (non-human clients) from connecting to your server. Clients connecting using an unregistered nickname will need to respond by entering a random code matching a text-pattern. The response can be given by visiting a provided URL or using the ANSWER command. |
| Enable IP Spoof Protection | Performs a ping check upon connecting clients to prevent IP spoofing. |
| Client Overflow Protection | Prevents hackers from exploiting vulnerabilities in old IRC Clients software such as PIRCH and Microsoft Chat. |
| No aliases within channels | Disallows the changing of nickname whilst on channels. |
| No Dynamic Channels | Disallow clients from creating a non-registered channel. |
| Restrict STAT command | Blocks clients from viewing bans, operator accounts, etc. |
| Restrict use of wildcards | Disallow clients querying user lists without an exact match. Recommended if experiencing problems with spam or mass invites. |
| Hide Network Structure | Blocks clients from using the LINKS command and masks server IP addresses from 'STATS C' and server notices. |
| Idle Connections | Disconnect clients who remain idle (away) for excessive amounts of time. |
| Output Flood Detection | Prevents an outbound flood from the server. Once the saturation limit is reached within a 60 second time period, the connection is terminated. Saturation limit is the number of outbound messages sent to a client connection. A client can cause an outbound flood by issuing multiple queries that may have large results returned. |
| Maximum Access Entries | Maximum entries allowed to be added to an access list. An access list is used by both nicknames and channels to store bans, ban exceptions and invitations. |
| Maximum Nickname Length | Maximum allowed length of a nickname. By default the allowed length is 25 characters. |
| Maximum Message Length | Maximum allowed size of messages. It is not recommended to increase the size above 512 characters because some software clients are vulnerable to crashes. |
| Maximum Watch List | Maximum allowed clients or channels you are allowed to add to your watch list. The watch list is used to alert you if a client signs on or a channel is started. See the WATCH command for more information. |
| Nickname Tracking | Server temporarily keeps tracking information for clients that have recently changed nicknames. This allows for example a channel host to kick the correct person if they just changed nicknames and the old nickname was targeted. |
| Allow Null Real Name | Allows clients to logon without specifying a Real Name. |
| Enable Ident Lookups | Enable the server to attempt to obtain the clients Ident to identify the user of a particular TCP connection. This is particularly useful for abuse control and general reporting because individuals can be identified when connecting from shared server or bouncer. See the Identification Protocol (RFC 1413) for more information |
| Enable NTLM Authentication | Also known as Windows Authentication allows single sign-on for users within an office environment. The server will use accounts stored in Active Directory (Windows Domain) for authentication purposes. A user can automatically login to IRC without being prompted for credentials if using supported software and has logged into their workstation using a domain account. |

| | |
|---|---|
| Enable Safe List | Enables the ability to query the LIST command to narrow down returned results. |
| Heavy Load Restrictions | If your server is experiencing high processor usage, the server will automatically switch to heavy traffic mode and will restrict processor intensive commands. |
| Display client QUIT as PART | When clients on channels disconnect from server, they will be seen as leaving the channels instead. |
| Suppress QUIT and PART reasons | Disables the ability for a client to leave a message when departing from a channel or server.  Such messages are not subject to content filtering. |
| High Invisibility | Invisible users (User Mode +i) will not receive private messages from other users unless they are both on the same channel or the sender’s nickname is on the recipients watch list. |
| Identify After Logon | Allows a client to login using a registered nickname without properly authenticating.  Those clients will then be prompted for the password and will have 60 second to comply. |

## 2.8 DNS Blacklist

DNS Blacklist (DNSBL) is used to check connecting clients against known Internet blacklists.  Clients found on such black lists are known to be compromised and can pose a security risk.



| Blacklist Name | Action | DNS Suffix |
|---|---|---|
| CBL | KLine (24h) | cbl.abuseat.org |
| AHBL | KLine (24h) | dnsbl.ahbl.org |
| TorAHBL | KLine (24h) | tor.ahbl.org |
| DroneBL | KLine (24h) | dnsbl.dronebl.org |
| TorDNSEL | KLine (24h) | ip-port.exitlist.torproject.org |

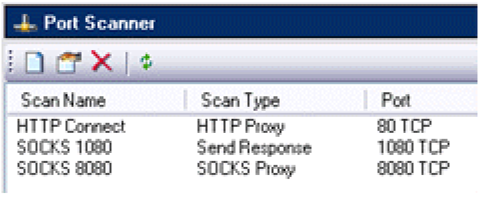
| | |
|---|---|
| Enable DNS Blacklist | Enable the DNS Blacklist feature to check incoming client connections. |

*DNSBL Provider Properties*

<table>
<tr><td>Provider</td><td>Name of the DNSBL Provider.</td></tr>
<tr><td>Disable Provider</td><td>Disables the DNSBL Provider.</td></tr>
<tr><td>Warning</td><td>Explanation given to the client when warned or punished.</td></tr>
<tr><td>DNS Suffix</td><td>Domain address of the DNSBL Provider.</td></tr>
<tr><td>Custom</td><td>Allows you to customize the query.   Default Query: %1.%2<br>%1 Client IP Address      %2 Server Port      %3 Server IP Address</td></tr>
<tr><td>Return Codes</td><td>A code is returned if the client’s IP is on a blacklist.  The codes have different meanings between providers.  Some providers combine blacklists and return different codes depending on the source.<br>You can specify which codes to accept, using commas as separators.</td></tr>
<tr><td>Action</td><td>Action to be taken if detection is made.
<table>
<tr><td>Warn</td><td>Send a warning to the client regarding the detection.</td></tr>
<tr><td>Kill</td><td>Disconnect the client from the server.</td></tr>
<tr><td>Kline (24h)</td><td>Disconnect and temporally ban client locally.</td></tr>
<tr><td>Kline</td><td>Disconnect and permanently ban client locally.</td></tr>
<tr><td>AKill (24h)</td><td>Disconnect and temporally ban client globally.</td></tr>
<tr><td>AKill</td><td>Disconnect and permanently ban client globally.</td></tr>
</table>
</td></tr>
<tr><td>Alert Wallops</td><td>Sends a WALLOPS alert across network regarding detection.</td></tr>
</table>

## 2.9 Port Scanner

Port Scanner is used to detect clients connecting using insecure proxies (bouncers). It is important to prevent these clients from connecting because they can be used to evade bans.



| Scan Name | Scan Type | Port |
|---|---|---|
| HTTP Connect | HTTP Proxy | 80 TCP |
| SOCKS 1080 | Send Response | 1080 TCP |
| SOCKS 8080 | SOCKS Proxy | 8080 TCP |

| | |
|---|---|
| Enable Port Scanner | Enable the port scanner to scan incoming client connections. |

*Port Scan Properties*

| | |
|---|---|
| Scan Name | Name of the scan. |
| Disable Scan | Disables the port scan. |
| Description | Explanation given to the client when warned or punished. |
| Scan Type | Type of detection technique to be used on the scan. |

| | |
|---|---|
| Open Port | Detects if accepting connections. |
| Connection Event | Detects if receive welcome message. |
| Send Response | Detects if port sends responses. |
| HTTP Proxy | Detect if running HTTP proxy service. |
| SOCKS Proxy | Detect if running SOCKS proxy service. |
| Custom Detection | Allow defined detection in Advance tab. |
| Masquerade | Allows detection of an alien service. |

| | |
|---|---|
| Port | Port number to scan. Common insecure ports are: |

| | |
|---|---|
| TCP 80 | HTTP Proxy |
| TCP 1080 | SOCKS Proxy |
| TCP 3128 | HTTP Proxy |
| TCP 8080 | HTTP Proxy |

| | |
|---|---|
| Protocol | Internet Protocol Type (Default TCP) |

| | |
|---|---|
| Action | Action to be taken if detection is made. |

| | |
|---|---|
| Warn | Send a warning to the client regarding the detection. |
| Kill | Disconnect the client from the server. |
| Kline (24h) | Disconnect and temporally ban client locally. |
| Kline | Disconnect and permanently ban client locally. |
| AKill (24h) | Disconnect and temporally ban client globally. |
| AKill | Disconnect and permanently ban client globally. |

| | |
|---|---|
| Alert Wallops | Sends a WALLOPS alert across network regarding detection. |

Custom Detection allows you to define detection of specific services running on computers such as Trojan Horses and other malicious software.  To define such a scan you will need to be familiar with the communication protocol being used.  For example, if the service you are trying to detect has a unique greeting, you can use the hex editor in the Advanced Tab (Receive 1~5) to make that detection by matching an expected incoming message.  You can use the Send option (in dropdown list) if it is necessary to send a message upon connection to provoke a specific response.

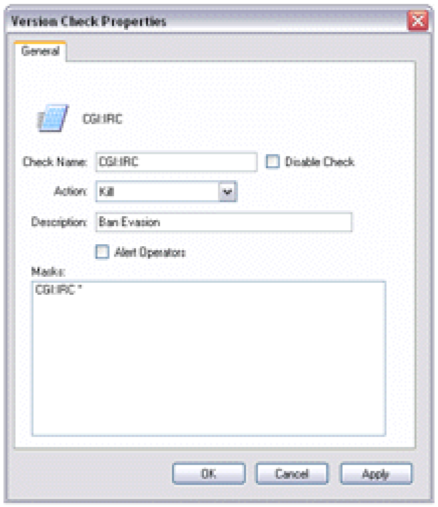
The Masquerade option allows you to detect an unknown service running on a reserved port.  This works in reserve to Custom Detection.  For example, if you wish to detect if a client is running something other then an FTP Service on port 21, your scan should try to detect an FTP Service by using hex such as 323230202A (“220 *”) in Receive #1.  This will detect the FTP service by listening for the FTP greeting message.

Hex can be manually typed into the Hex Editor on the left side; the right side can be used to type ASCII characters.  For text matching, there are 3 options available: Like, Exact, and Anywhere.  If using ‘Like’ matching, wildcards can be used.

### *2.10 Version Check*

Version Check is used for checking the product name and version of client software used to connect to the server.

Using the Version Check can be useful if you wish to only allow web-based clients, prohibit the use of a specific client (e.g. CGI:IRC can be used as a bouncer), or simply notify clients that they are using outdated or vulnerable software.



Check Name: CGI:IRC ☐ Disable Check

Action: Kill

Description: Ban Evasion

☐ Alert Operators

Masks:

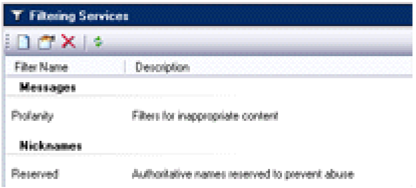
CGI:IRC *

### *Version Check Properties*

<table>
<tr><td>Enable Version Check</td><td>Enables the Version Check to request the client version upon connecting to the server.</td></tr>
<tr><td>Kill non answering users</td><td>Disconnects a client if no version response is received.</td></tr>
<tr><td>Check Name</td><td>Name of the version checking group.</td></tr>
<tr><td>Disable Check</td><td>Disable version check.</td></tr>
<tr><td>Action</td><td>Action to be taken if version response matches any entries in the group.
<table>
<tr><td>Allow</td><td>Allow entry and not to be subjected to any further actions.</td></tr>
<tr><td>Warn</td><td>Send a warning to the client regarding detection.</td></tr>
<tr><td>Kill</td><td>Disconnect the client from the server</td></tr>
<tr><td>Kline (24h)</td><td>Disconnect and temporally ban client locally.</td></tr>
<tr><td>Kline</td><td>Disconnect and permanently ban client locally.</td></tr>
<tr><td>AKill (24h)</td><td>Disconnect and temporally ban client globally.</td></tr>
<tr><td>AKill</td><td>Disconnect and permanently ban client globally.</td></tr>
</table></td></tr>
<tr><td>Description</td><td>Explanation given to the client when warned or punished.</td></tr>
<tr><td>Alert Wallops</td><td>Sends a WALLOPS alert across network regarding detection.</td></tr>
<tr><td>Masks</td><td>Masks matching the version response e.g. “mIRC *”.  Double-click to add a new mask and single-click to modify or delete an existing mask.</td></tr>
</table>

## *2.11 Filtering*

Filtering allows you to control the content on your server, from blocking unauthorized advertising to censoring inappropriate and offensive language.



| Filter Name | Description |
|---|---|
| **Messages** | |
| Profanity | Filters for inappropriate content |
| **Nicknames** | |
| Reserved | Authoritative names reserved to prevent abuse |

There are 4 filter types you can perform:

- Messages – *Allows the sender of the message to be punished and the message itself to be censored or blocked.*
  - Channel messages and whispers if filtering is enabled on the channel.
  - Private messages if filtering is enabled in the assigned Client Rule.
  - Real Names given during logon if enabled in Extended Filtering.
  - Channel topics if enabled in Extended Filtering.
  - Messages received across network if enabled in Extended Filtering.
- Channel Names – *Prevents channel impersonations and channel names containing profanity.  Can also be used as a redirect for renamed channels.*
- Nicknames – *Reserves nicknames / prefixes used by operators and network services and prevents inappropriate names from being used.*
- File Names – *Prevent the sending of files with specific extensions (e.g. EXE, BAT, COM, etc) and file names of known viruses and prohibited content.*

When creating a filter, it is recommended that you filter for a content type which has a group of words with a similar punishment.  For example, keep unsolicited advertising separate from a profanity filter and then create additional filters such as ‘Unwelcome Remarks’, ‘Bad Language’, ‘Racist Comments’ etc.  This will allow you to be more lenient on minor indiscretions, and harshly punish the worst offences.

## *Filter Properties*

| | |
|---|---|
| Filter Name | Name of the filter. |
| Disable Filter | Disables the content filter |
| Filter Type | Type of content for which this filter applies. |
| Description | Brief description explaining the purpose of the filter. |
| Words | A list of words for which the filter will scan content. When adding a Word entry, it is not necessary to use wildcards unless filtering for File Names. |

| | |
|---|---|
| Character art recognition | *Brackets, slashes and other symbols which look similar to alphabetical letters are converted e.g. “\/\/()R\|)” will be converted to “WORD”.* |
| Check anywhere in a word | *Checks if the phase appears within words e.g. Detects the presence of “WORD” in “Words”, “Wording”, “Small-words” etc.* |
| Check for word slurring | *Checks if a word is slurred or jumbled e.g. Detects the presence of “WORD” in “Worrrrrrd”, “zWzOzRzDz” etc.* |
| Convert language symbols | Converts language symbols into their equivalent alphabetical letters e.g. “ŵºŗĐ” will be converted to “WORD”. |
| Check for leading characters | Checks the message for words which have not been properly joined e.g. If trying to find a match on “WORD”, it will be detected when looking at “Wor dings”, “W o r d”, “zzzWO RDzz” etc. |

| | |
|---|---|
| Penalties | If the client has been caught by the filter, he will face punishment by the selection in Penalties. If this is the first offence, you can be more lenient (in the ‘First offence’ dropdown) and repeated offences could result in being banned (in ‘Second offence’ and ‘Subsequent offences’ dropdowns). |

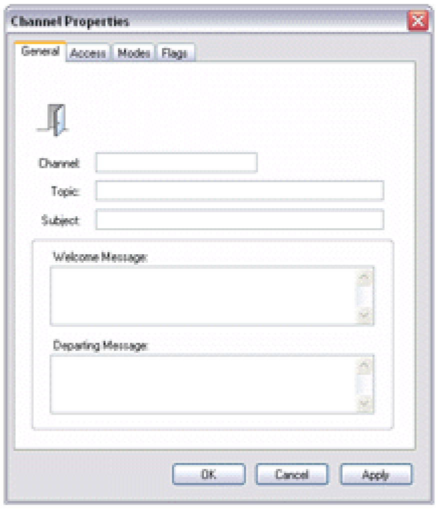
| | |
|---|---|
| Take No Action | No action will be taken. |
| Previous Action | Uses the previous action. |
| Cannot Register | Prevents channel/nickname from being registered. |
| Redirect | Moves the client to the channel name specified. |
| Warn Recipient | Sends warning to the recipient. |
| Cannot Join | Cannot join or create the channel. |
| Warn Sender | Sends a warning to the client. |
| Censor | Censors words instead of blocking whole message. |
| Block | Blocks the entire message. |
| Take Voice | Block and revoke speaking privileges in a channel. |
| Kick | Kicks the client from the channel. |
| Ban | Kick and ban the client from the channel. |
| Gag | Gags the client without their knowledge. |
| Kill | Disconnect the client from the server. |
| Kline (24h) | Disconnect and temporally ban client locally. |
| Kline | Disconnect and permanently ban client locally. |
| AKill (24h) | Disconnect and temporally ban client globally. |
| AKill | Disconnect and permanently ban client globally. |

| | |
|---|---|
| Redirect | Used to specify the channel name if using a redirect as a penalty. |
| Censor Mask | Used to specify mask if censoring messages e.g. *CENSORED* |
| Warning Messages | Used for specifying warnings depending on reoccurring offences. |

## 2.12 Channels

Channel services allows you to create and manage registered channels (Chat Rooms). Only locally registered channels appear in the list.

All registered channels are classified as sticky. This means that channels store changed settings such as properties, access, and modes when the channel is not being used.



| | |
|---|---|
| Allow users to register channels | Allows clients to register their own channels via the REGISTER command. |
| Limit registrations | Allows you to limit the maximum allowed registered channels. |
| Account required to register channels | Must be using a registered nickname. |
| Default channel limit | Maximum members allowed to join when no channel limit set. |
| Automatically remove channels | Drops dormant registrations after a period of time. |

### *Channel Properties*

<table>
<tr><td>Name</td><td>Name of the channel (Chat Room).<br><br>Channel prefix determines the channel type. To create a global channel (RFC1459), prefix the name with the # (Hash/Pound) symbol otherwise the channel will be an extended global channel (IRCX) which supports Unicode. Local channels (RFC1459) which are inaccessible across a network can be created by prefixing the name with an & (Ampersign) symbol.</td></tr>
<tr><td>Topic</td><td>Topic of the current discussion.</td></tr>
<tr><td>Subject</td><td>Keywords used for search engines.</td></tr>
<tr><td>Welcome Message</td><td>Message automatically sent to clients after they join the channel.</td></tr>
<tr><td>Departing Message</td><td>Message automatically sent to clients after they leave the channel.</td></tr>
<tr><td>Channel Passwords</td><td>When joining a channel, the password can be supplied using the JOIN command.<br><br><table>
<tr><td>Owner</td><td>Channel owner password for full access.</td></tr>
<tr><td>Host</td><td>Channel host password for moderator access.</td></tr>
<tr><td>Voice</td><td>Voice password for gaining speaker privileges.</td></tr>
<tr><td>Member</td><td>If set, password is required to join the channel.</td></tr>
</table></td></tr>
<tr><td>Member Account</td><td>Nickname the channel registration is associated with.</td></tr>
<tr><td>Access List</td><td>Used to store access entries such as channel bans, invitations, ban exceptions etc. See the ACCESS command for more information.</td></tr>
</table>

<table>
<tr><td>View Mode</td><td>Allows you to set the channels visibility.<table>
<tr><td>Public</td><td>Channel appears in list and members are visible.</td></tr>
<tr><td>Private</td><td>Channel appears in list but topic and members are hidden.</td></tr>
<tr><td>Hidden</td><td>Channel hidden from list, can only be queried using exact name and members are hidden.</td></tr>
<tr><td>Secret</td><td>Channel hidden from list, cannot be queried from outside and members are hidden.</td></tr>
</table></td></tr>
<tr><td>Member Limit</td><td>Maximum number of members allowed inside at one time.</td></tr>
<tr><td>Auditorium</td><td>Restricts visibility and messaging within a channel. Members can only see themselves and the hosts/owners within the channel. Any message sent from a member can only be viewed by hosts/owners. Hosts/owners can see all members within the channel and their messages are visible to all members of the channel.</td></tr>
<tr><td>Invitation Only</td><td>An invitation is required to join the channel. Invitations can be issued using the INVITE command or adding an INVITE entry to the channel access list.</td></tr>
<tr><td>Only Authenticated Users</td><td>Only authenticated clients can join the channel.</td></tr>
<tr><td>Moderated</td><td>Only members given voice permissions can speak within the channel.</td></tr>
<tr><td>Cannot Change Passwords</td><td>Prevent channel owner from changing channel passwords.</td></tr>
<tr><td>Cannot Change Modes</td><td>Prevents channel hosts from changing the channel modes.</td></tr>
<tr><td>Channel Host Guard</td><td>Prevents channel hosts from giving host status to members.</td></tr>
<tr><td>Controlled Locally</td><td>Only local administrators can control the channel.</td></tr>
<tr><td>No Client Message Formatting</td><td>Instructs client software to turn off message formatting.</td></tr>
<tr><td>No External Messages</td><td>Only members within the channel can send messages.</td></tr>
<tr><td>No Whispering</td><td>Disables the ability for members to send whispers within the channel.</td></tr>
<tr><td>Only Host Change Topic</td><td>Prevents members changing the channel topic.</td></tr>
<tr><td>Restrict Channel Access</td><td>Deny hosts from viewing or modifying channel access list.</td></tr>
<tr><td>Restricted to Server Operators</td><td>Only server operators can join the channel.</td></tr>
<tr><td>Join Flow</td><td>Interval in seconds before channel allows another member to join.</td></tr>
<tr><td>Lag</td><td>Artificial delay before client can send another message to channel.</td></tr>
<tr><td>Rate</td><td>Reduces the speed rate of messages, protecting a channel from being flooded.</td></tr>
<tr><td>Caps Lock</td><td>Converts messages to lowercase if contains a lot of capital letters.</td></tr>
<tr><td>Client GUID</td><td>Only permits clients of a specific chat protocol to join channel.</td></tr>
<tr><td>Only this Class</td><td>Only clients from the selected Client Rule (Class) can join the channel.</td></tr>
<tr><td>Client-specific data</td><td>Parameters developers can provide to customized client software.</td></tr>
<tr><td>Content rating (PICS)</td><td>Allows you to rate the channel based on its content.</td></tr>
<tr><td>Language code</td><td>Preferred language using ISO 639 Language Codes.</td></tr>
<tr><td>Echo To Source</td><td>Messages sent into channel are echoed back to sender.</td></tr>
<tr><td>Knock Notifications</td><td>Hosts/owners receive knock notifications when a client fails to join.</td></tr>
<tr><td>Filter Channel Content</td><td>Enables content of messages sent to channel to be filtered.</td></tr>
<tr><td>Save Channel Transcript</td><td>Saves channel conversations to a server log file.</td></tr>
<tr><td>Automatically Reset Access</td><td>Clears channel access list when last member leaves.</td></tr>
<tr><td>Cloneable</td><td>Clones channel when member limit is reached and redirects new members.</td></tr>
<tr><td>Permanent Channel</td><td>Prevents channel being unregistered if dormant.</td></tr>
<tr><td>Auto Start Channel</td><td>Allow channel to remain open and appear in listing when empty.</td></tr>
</table>

### 2.13 Channels Advanced

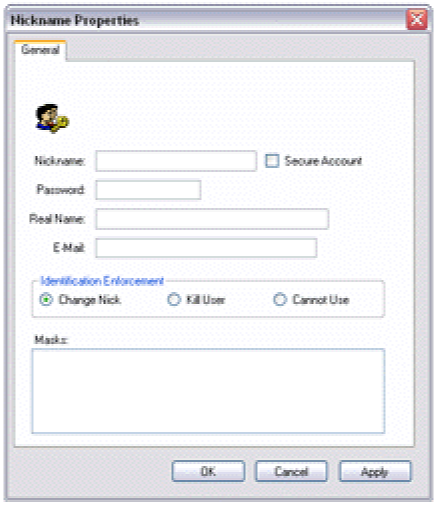
Creation Modes allows you to set the initial modes upon creation of a new channel.

| | |
|---|---|
| Auto Start all Channels | All channels remain open and appear in listing when empty. |
| Isolate clients by classes | Can be used by shared servers to separate client groups. Clients cannot view others outside of their Client Rule (Class) or join the same channels. |
| Display Channel Creation Date | Display channel creation date when querying modes. |
| Display Topic Set By Information | Display Set By information with the channel topic. |
| Hide Channel Access Masks | Cloaks the user masks used in the channel access list. |
| Single line mode change | Displays mode's with parameters on separate lines. |

### 2.14 Nicknames

Nickname services allows you to create and manage registered nicknames. Only locally registered nicknames appear in the list.

The reason for registering is to reserve the nicknames of client's to prevent unauthorized use.



Nickname:

□ Secure Account

Password:

Real Name:

E-Mail:

Identification Enforcement

● Change Nick ○ Kill User ○ Cannot Use

Masks:

| | |
|---|---|
| Allow users to register nicknames | Allows clients to register their own nicknames using the REGISTER command. |
| Limit registrations | Allows you to limit the maximum allowed registered nicknames. |
| Authenticate nickname owners | Anonymous clients logging on using a registered nickname will be authenticated. |
| Channels allowed per registration | Limits maximum channels an individual can register. |
| Automatically remove nicknames | Drops dormant registrations after a period of time. |
| Identification Enforcement | Specify action to take for all nickname registrations if users do not identify within the allowed time period.<br>Options include having nick forcibly changed, connection terminated or unable to use nickname unless identified during logon. If 'Users Preference' is selected, enforcement is decided from individual nickname registrations. |

*General*

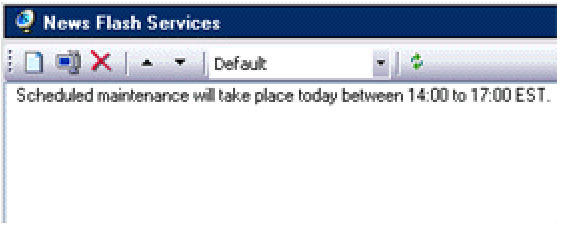
| | |
|---|---|
| Nickname | Name of registered nickname |
| Secure Account | Prevents non-administrators from accessing nickname registration. |
| Password | Password used for authentication. |
| Real Name | Real Name of the owner. |
| E-Mail | Email address of the owner. |
| Identification Enforcement | Action to take if user who fails to identify within allowed time period. |
| Masks | Matching client can use nickname. Double-click to add a new mask and single-click to modify or delete an existing mask.<br>Supported Mask: <userid>@<hostname> |

## 2.15 Nicknames Advanced

| | |
|---|---|
| Assign Nickname Dynamically | If connecting using NTLM or ODBC Authentication, automatically assigns client a nickname based on username. |
| Force Nicknames to Lowercase | Clients cannot use capital letters within nicknames. |
| Verify Email Addresses | New nickname registrations must have email address verified. Clients are sent an email containing a URL that activates their nickname registration. |
| Automatically Assign Client Rule | If using ODBC Authentication, clients are auto assigned to specified Client Rule. |
| Automatically Register Nickname | If using NTLM or ODBC Authentication, clients automatically have a registered nickname created. Reserves the name and allows client to receive messages sent while offline. |
| Mail Server | Allows you to specify SMTP settings for registration verification. |
| Enable ODBC Authentication | Authenticate connecting clients against a database of user accounts. Using ODBC drivers, you can connect to databases such as Access (MDB), MS-SQL, MySQL etc. |
| Data Source Name | Name of DSN created in the ODBC Data Source Administrator. |
| Password Encoding | Specify if passwords stored are plain-text or encoded. |
| Table Name | Specify which table stores the user accounts. |
| User Field | Specify the field name for the Username. |
| Password Field | Specify the field name used for storing passwords. |
| Class Field | Specify the field name if using the 'Automatically Assign Client Rule' option, otherwise keep blank. |

## 2.16 News Flashes

News Flash services allow you to send news, tips and advertising notices to your clients. A list of notices can be defined for each Client Rule. The server will endlessly loop though defined notices and broadcast to clients at the set frequency.



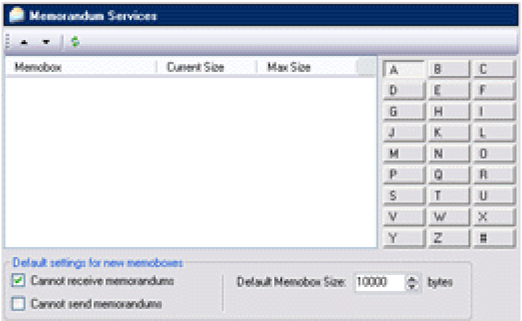
Scheduled maintenance will take place today between 14:00 to 17:00 EST.

| | |
|---|---|
| Enable Newsflash | Enable the news flash service. |
| Do not broadcast to Server Operators | Excludes operators from receiving broadcast. |
| Broadcast newsflashes to network | Sends notices of Default Class across network. |
| Broadcast Frequency | Sets broadcast interval between news flashes. |

### *2.17 Memorandums*

Memorandums services are used to send and receive messages from clients who weren't online at the same time.  Server automatically alerts clients upon logging in of any unread messages.  Clients can use the MEMO command to send memos and access inbox.

You can enable memo services for a specific nickname registration using the checkbox. You can also view the current memobox size and increase capacity using the toolbar.


Memorandum Services

Default settings for new memoboxes

☑ Cannot receive memorandums

□ Cannot send memorandums

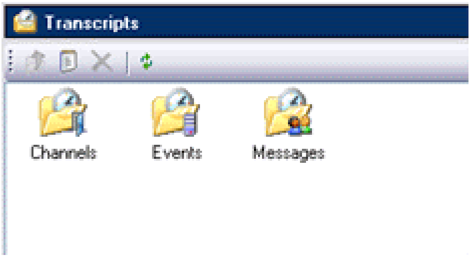
Default Memobox Size: 10000 bytes

| | |
|---|---|
| Cannot receive memorandums | By default new registrations cannot receive memos. |
| Cannot send memorandums | By default new registrations cannot send memos. |
| Default Memobox Size | Sets default capacity of memobox for new registrations. |

### *2.18 Transcripts*

Transcript Services allows you to remotely view and delete server log files.  By default, your server will not perform any logging.

There are 3 types of log files: Channel Transcripts, Server Audit logs, and Private Message logs.


Transcripts

| | |
|---|---|
| Enable channel transcripts | Saves transcripts of channels with the 'Save Channel Transcript' flag checked. |
| Enable private message logging | Saves private conversations to a log file.  Messages sent via DCC Chat are excluded from the log.  If you need to enforce complete logging, we recommend disabling the ability to send and receive DCC requests and Lock Modes "+e" from the Client Rules. |
| Enable server auditing | Saves all activity to a daily server log file. |
| Automatically delete transcripts | Auto deletes old log files after a specified duration. |

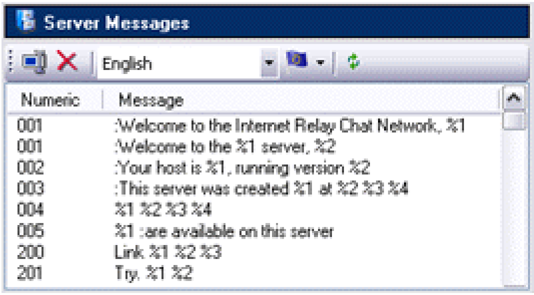
### *2.19 Chat Back*

Chat Back allows clients to view recent messages from channel transcript logs using the CHATBACK command.

| | |
|---|---|
| Enable Chat Back command | Allows clients to query a channels transcript log. |
| Chat Back on Channel Entry | Display recent transcript entries upon joining channels. |
| Default Replay | If no parameters given, returns the specified criteria. |
| Maximum Replay | How far back the transcript log can be queried and limits returned output. |

## 2.20 Messages

Messages area allows you to customize your server messages. You can also translate your server messages for clients which have a different language preference.



Server Messages

English

| Numeric | Message |
|---|---|
| 001 | :Welcome to the Internet Relay Chat Network, %1 |
| 001 | :Welcome to the %1 server, %2 |
| 002 | :Your host is %1, running version %2 |
| 003 | :This server was created %1 at %2 %3 %4 |
| 004 | %1 %2 %3 %4 |
| 005 | %1 :are available on this server |
| 200 | Link %1 %2 %3 |
| 201 | Try. %1 %2 |

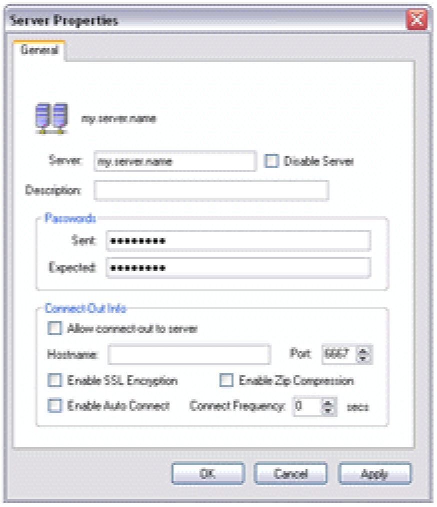
## 2.21 Server Linking

Server Linking allows you to link your server with another server (or group of servers) running either OfficeIRC, DreamForge or UnrealIRCD.

When linked to one or more servers, your server becomes part of a Chat Network. On such a network, all users and global channels are shared.

You may wish to link your server for many reasons, such as merging servers to increase users, load balancing if in the thousands of users, or to have a backup server on standby.

If you are trying to deploy an internal communications network for a company which has multiple large sites, we recommend running a server locally at each site and use server linking to connect all the sites together. This would reduce outbound traffic, quicken response times and create a fault-tolerant system which would continue to function locally if the Internet was down.



Server Properties

General

my.server.name

Server: my.server.name ☐ Disable Server

Description:

Passwords

Sent: ********

Expected: ********

Connect-Out Info

☐ Allow connect-out to server

Hostname: Port: 6667

☐ Enable SSL Encryption ☐ Enable Zip Compression

☐ Enable Auto Connect Connect Frequency: 0 secs

### Considerations when linking

1. Your server name and server numeric must be unique on the network
2. It is recommended for your server clocks to be synchronized.
3. Make sure there are no zlines in place that might block the connection.

### Channel collisions

When two channels with the same name exists on both servers prior to linking, after the link is established, the channels become merged together. The oldest channel will be preserved, overwriting the newer version and demoting any hosts/owners.

### Nickname collisions

When two clients with the same nickname are logged in from both servers prior to linking, the most recently connected client will be terminated after the link is established.

*Server Link Properties*

| | |
|---|---|
| Server | Server name of the destination server. |
| Disable Server | Disables server from establishing link. |
| Description | Short description of destination server. |
| Sent Password | Password sent to destination server. Must match password specified in Expected field on the destination server. |
| Expected Password | Password expected from destination server. Must match password specified in Sent field on the destination server. |
| Allow connect-out to server | Allows server to connect-out using the CONNECT command. *Only one server needs to connect-out.* |
| Hostname | Host Address (or IP Address) of destination server. |
| Port | TCP chat port of destination server. |
| Enable SSL Encryption | Use if connecting to an SSL enabled chat port. |
| Enable Zip Compression | Reduces bandwidth but increases processor usage. |
| Enable Auto Connect | Automatically attempts to connect-out to destination server. |
| Connection Frequency | Interval between connect-out attempts if using Auto Connect. |

## 3. IRC Command Reference

NOTE: To use an IRC command in a text box, type a forward slash and then type the command in the message box.

In the list below, certain characters indicate the type of information you must enter:

- A pipe character (|) indicates OR.
- Angle brackets (<>) indicate the type of the information you must enter.
- Square brackets ([]) indicate an optional part of the syntax.
- Curly brackets ({}) indicate that the entry can be multiple.

### ACCESS (IRCX)

```
Usage:   ACCESS <object> LIST
         ACCESS <object> ADD <level> <mask> [<timeout> [<reason>]]
         ACCESS <object> DELETE <level> <mask>
         ACCESS <object> CLEAR [<level>]
```

Access is used to create, delete and list access entries for an object. An access entry is used by an object to grant or deny access.

Access Entry Types:

| | |
|---|---|
| DENY | Disallow access to an object that is accessible. |
| GRANT | Allow access to an object that is inaccessible. |
| HOST | Host access to specified channel. |
| OWNER | Owner access to specified channel. |
| VOICE | Voice access to specified channel. |
| INVITE | Invitation to specified channel. |
| SHUN | Read-only access to server, same as SHUN command. |
| ZLINE | Disallow access to server, same as ZLINE command. |

The object can be a channel name, nickname registration, user, $ (server), * (network), client rule (class) or operator account. The timeout is the minutes until the access entry expires. A value of 0 indicates unlimited duration.

### ADMIN

Usage:  ADMIN [<server>]

Returns administrator contact information.

### AKILL

Usage:  AKILL <user>|<mask> [<timeout> :][<comment>]

Places a network-wide ban to prevent any matching users from connecting.  The timeout is the seconds until the ban expires.  A value of 0 indicates unlimited duration.

### ANSWER (OIRC)

Usage:  ANSWER <code>

If Visual Confirmation is enabled, allows a user to submit their answer.

### AUTH (IRCX)

Usage:  AUTH <SASL mechanism> <sequence> [:<parameter>]

Authenticates a client using an SASL authentication mechanism during login.  NTLM is supported which allows single sign-on using Windows Authentication.

### AWAY

Usage:  AWAY [<message>]

Sets an away message which is sent as an automatic reply to any received messages.

### CHATBACK (OIRC)

Usage:  CHATBACK <channel> [<duration> [<query limit>]]

If enabled, allows users to query recent entries from a channel's transcript log.  The duration is the minutes of how far back to view.

### CHGHOST

Usage:  CHGHOST [<user>] <cloak>

Cloaks the hostname of a specified user.

### CLEARDEAD

Usage:  CLEARDEAD

Removes empty registered channels belonging to servers that are no longer connected.

### CONNECT

Usage:  CONNECT <server> [<port> [<remote server>]]

Initiates an outbound connection to establish a server link.  The specified server must be configured to connect-out.

### CPUREPORT (OIRC)

Usage: CPUREPORT +|-

View the CPU processor usage of the machine the server is running on.

### CREATE (IRCX)

Usage: CREATE <channel> [<modes> [<modeargs>]]
CREATE <object> <[arguments]>
Creates a new object/channel and/or join an existing channel. Use mode ‘e’ to force a clone of a clonable channel or ‘c’ to create and join a channel only if it does not exist.

### DATA | REPLY | REQUEST (IRCX)

Usage: DATA <user>|<channel> <tag> :<message>

Sends tagged data, requests or replies to a client or channel.

### DESTROY (OIRC)

Usage: DESTROY <object>

Deletes the specified object. Cannot be used to destroy a channel. Non-registered channels are automatically destroyed when last user leaves.

### DIE

Usage: DIE <password>

Remotely shutdowns the chat server. The command is password protected to prevent accidental use. Use the account password of the currently logged in server operator or the hard-coded password “SCRAM” (if using Windows Authentication).

### EVENT (IRCX)

Usage: EVENT ADD|DELETE <event> <mask>
EVENT LIST
Allows the logging of server activity such as incoming client connections. Use the EVENT command to add, delete and view the event list.

Event types: CHANNEL, MEMBER, SERVER, CONNECT, SOCKET or USER.

### FJOIN

Usage: FJOIN <user> <channel>

Forces user to join the specified channel.

### FNICK

Usage: FNICK <user> <new nickname>

Forces a nickname change on the specified user.

### HELPOPS

Usage: HELPOPS <message>

Sends a message calling for help to all help operators currently online.

### INFO

Usage:	INFO [<server>]

Returns information which describes the server version, when it was started and other miscellaneous information.

### INVITE

Usage:	INVITE <user> <channel>

Sends the user an invitation to join the specified channel.

### IRCX (IRCX)

Usage:	IRCX

Enables IRCX mode and displays IRCX status.

### ISON

Usage:	ISON <nickname>{ <nickname>}

Queries if the following nicknames are currently logged in.

### JOIN

Usage:	JOIN <channel>{,<channel>} [<password>{,<password>}]
	JOIN 0

Lets you enter the specified channels (Chat Rooms). To gain elevated access (using an ownerkey or hostkey) or to join a password protected channel, the password must be supplied. Specifying 0 will exit you from all channels.

### KICK

Usage:	KICK <channel> <user> [<comment>]

Forcibly removes a member from the specified channel.

### KILL

Usage:	KILL <user>|<channel> [<comment >]

Terminates a user’s connection with the chat server.

### KLINE

Usage:	KLINE <user>|<mask> [<timeout> :][<comment>]

Places a local ban to prevent any matching users from connecting. The timeout is the seconds until the ban expires. A value of 0 indicates unlimited duration.

### LASTSEEN (OIRC)

Usage:	LASTSEEN <nickname>

Checks nickname registration and WHOWAS for the date/time user was last seen.

### LINKS

Usage: LINKS [[<remote server>] <server mask>]

Lists and maps the locations of online servers on the chat network.

### LIST

Usage: LIST [<query>,{<query>}]

Returns a list of opened channels with their member count and topics.

If SafeList enabled, the following queries are supported:

| | |
|---|---|
| <channel> | Select only the specified channels. |
| <# | Select channels with less than # members. |
| ># | Select channels with more than # members. |

### LISTX (IRCX)

Usage: LISTX [<query>,{<query>}] [<query limit>]

An extended version of the LIST command used to return a list of opened channels.

Supported queries:

| | |
|---|---|
| <# | Select channels with less than # members. |
| ># | Select channels with more than # members. |
| C<# | Select channels created less than # minutes ago. |
| C># | Select channels created greater than # minutes ago. |
| L=<mask> | Select channels with language property matching the mask string. |
| N=<mask> | Select channels with name matching the mask string. |
| R=0 | Select unregistered channels. |
| R=1 | Select registered channels. |
| S=<mask> | Select channels with subject matching the mask string. |
| T<# | Select channels with a topic changed less than # minutes ago. |
| T># | Select channels with a topic changed greater than # minutes ago. |
| T=<mask> | Select channels that topic matches the mask string. |
| <query limit> | Maximum number of channels to be returned. |
| <mask> | Select channels with name or topic matching the mask string. |

### LUSERS

Usage: LUSERS [<mask> [<server>]]

Displays information about the number of users logged on the server and network.

### MEMO (OIRC)

Usage: MEMO LIST|PURGE
MEMO SEND <nickname> <message>
MEMO READ|DELETE|UNDELETE <memo id>

Allows the sending of messages to users who are currently offline. Multiple lines can be generated by embedding '\n' in the message. Sent messages can be viewed using the LIST and READ subcommands.

### MODE

Usage: MODE <channel> [+|-]<modes> [<modeargs>{ < modeargs>}]
MODE <user> [+|-]<modes>

Allows both users and channels to have their mode changed. See channel/user modes section for a list of supported modes.

### MOTD

Usage: MOTD [<server>]

Displays the server’s message of the day. Usually administrative information and rules can be found within the message.

### MOVE

Usage: MOVE <user> <channel>

Forces user to leave all channels and join the specified channel.

### NAMES

Usage: NAMES <channel>{,<channel>}

Displays a list of members who are inside the specified channel. If performing the command on a channel you are not inside, invisible members are excluded from the list.

### NICK

Usage: NICK <new nick>

Allows a user to select a new nickname.

### NOTICE

Usage: NOTICE <receiver>{,<receiver>} <message>

Sends a message to a channel or user.

### OBJECTS (OIRC)

Usage: OBJECTS [<type>]

Displays a list of server objects such as client rules and operator accounts.

Object Types:

| | | | |
|---|---|---|---|
| # | Channel Registrations | M | Server Messages |
| ‘ | Nickname Registrations | O | Server Operator Accounts |
| @ | Memo boxes | P | Port Scans |
| C | Client Rules (Classes) | S | Server Link Profiles |
| F | Content Filters | V | Version Checks |
| G | Server Operator Groups | | |

### OPER
Usage: OPER <account> <password>

Grants you server operator status upon successful authentication. Having operator status allows you to use the many operator commands defined for your operator group.

### PART
Usage: PART <channel>{,<channel>} [<comment>]

Lets you leave the specified channels (Chat Rooms).

### PASS
Usage: PASS <password>
PASS <channel> <password>

Allows you to use a registered nickname or receive channel owner/host privileges.

### PING | PONG
Usage: PING <server1> [<server2>]

Test whether the connection is still alive and calculate the lag time.

### PRIVMSG
Usage: PRIVMSG <receiver>{,<receiver>} <message>

Sends a private message to a channel or user.

### PROP (IRCX)
Usage: PROP <object> *
PROP <object> <property>[,<property>]
PROP <object> <property> [<data>]
PROP <object> <property> :

Allows you view, add, change or delete data properties for channels, nickname registrations and other server objects.

### QUIT
Usage: QUIT <comment>

Closes the current session with the server.

### RAKILL
Usage: RAKILL <mask>

Removes a network-wide ban that was set using the AKILL command.

### REGISTER (OIRC)
Usage: REGISTER <nickname> <ownerkey> [<email address>]
REGISTER <channel> [<ownerkey>]
Allows a user to register a new nickname or channel.

### SAMODE

Usage: SAMODE <channel> [+|-]<modes> [<modeargs>{ < modeargs>}]

Allows server operators to set modes inside channels without having owner/host status.

### SHUN

Usage: SHUN [+|-<mask>]

Allows you to view, add or remove masks from the server's shun list. Prevents matching users from being able to send messages without their knowledge.

### SILENCE

Usage: SILENCE [+|-<mask>]

Allows you to view, add or remove masks from your ignore list. Prevents matching users from being able to send you messages.

### SLOG (OIRC)

```
Usage:  SLOG LIST <type>|<channel>
        SLOG GET <type>|<channel> <log name> <position> <return>
        SLOG SEARCH <type>|<channel> <log name> <position> :<search query>
        SLOG DELETE <type>|<channel> <log name>
```

Allows full access to the server's transcript logs. Log files can be listed, downloaded, searched and deleted using subcommands. Log types include EVENTS, MESSAGES and CHANNELS.

Supported Types:

| | |
|---|---|
| EVENTS | Server events. |
| MESSAGES | Private messages sent to users. |
| CHANNELS | Lists channels with transcript files. |
| <channel> | Events and messages sent to channel. |

### SQUIT

Usage: SQUIT <server> [<comment>]

Disconnects the specified server from the chat network.

### STATS

Usage: STATS <query> [<server>]

Queries the server for certain statistics.

Supported Queries:

| | |
|---|---|
| Y | Returns a list of client rules (classes). |
| Q | Returns a list of prohibited nicknames. |
| P | Returns a list of prohibited channel names. |
| T | Returns a list of prohibited words. |
| C | Returns a list of servers allowed to establish a server link. |
| Z | Returns a list of zap lines set using the ZLINE command. |
| K | Returns a list of local and network-wide user bans. |
| O | Returns a list of operator accounts. |

### TIME

Usage: TIME [<server>]

Displays the server's local time and date.

### TOPIC

Usage: TOPIC <channel> [<new topic>]

Allows you to change or view the topic of a channel.

### TRACE

Usage: TRACE <server>|<user>

Allows you to find the route to the specified user or server from across the chat network.

### UNKLINE

Usage: UNKLINE <mask>

Removes a local ban that was set using the KLINE command.

### UNREGISTER (OIRC)

Usage: UNREGISTER <channel>|<nickname>

Allows a user to drop the registration of a nickname or channel.

### UNZLINE

Usage: UNZLINE <ip address>

Removes a local zap line set using the ZLINE command.

### USERHOST

Usage: USERHOST <user>{ <user>}

Displays the identify and away status of the specified users.

### USERS

Usage: USERS [<server>]

Returns a list of users logged into the server.

### VERSION

Usage: VERSION [<server>]

Displays the server software version number.

### WALLOPS

Usage: WALLOPS <message>

Sends a network-wide message to users receiving wallop notices.

### WATCH

Usage: WATCH +|-<nickname>{,+|-<nickname>}
WATCH +|-<channel>{,+|-< channel>}
WATCH <subcommand>

Allows you to add or remove entries on your watch list. When a user is online/offline or a channel is present/empty, the server will automatically notify you.

Subcommands:

| | |
|---|---|
| c | Clears all entries from the watch list. |
| s | Shows the watch list and how many users are watching you. |
| l | Displays only online/present entries from the watch list. |
| L | Displays entire watch list indicating the status of each users/channels. |

### WHISPER (IRCX)

Usage: WHISPER <channel> <user>{,<user>} <message>

Sends a private message to multiple users specified within a channel, allowing a group of users to talk privately without needing to join an additional channel.

### WHO

Usage: WHO [<channel>|<user> [<g|h|o|w>]]

Displays a list of users which includes their full name, away status and other information. If performing the command on a channel, it will display the users on the channel (except invisible users if not inside the same channel).

| | |
|---|---|
| g | Returns only members who are gone. |
| h | Returns only members who are here. |
| o | Returns only server operators. |
| w | Returns only members receiving wallops. |

### WHOIS

Usage: WHOIS [<server>] <user>{,<user>]

Displays detailed information regarding a user. This includes the real name, hostname, server they are logged into, channels etc.

### WHOWAS

Usage: WHOWAS <nickname>

Displays information about a previous client for a limited period of time.

### ZLINE

Usage: ZLINE <ip address> [<timeout> :][<comment>]

Allows you to add a firewall style ban to prevent a user from establishing a connection. No notices or lookup attempts are performed when a user attempts a connection.

## 4. Channel Modes

**a** Authenticated clients only (IRCX)

*Allows only authenticated clients to join the channel.*

**b** Ban flag

*Usage: +b <nick!userid@hostname>*

*Bans users matching the specified mask from entering the channel.*

**d** Cloneable channel (IRCX)

*Causes the channel to clone itself upon reaching the channel limit. Users will automatically be redirected to the next channel in sequence e.g. <channel>1, <channel2>, etc.*

**e** Except flag / cloned channel

*Usage: +e <nick!userid@hostname>*

*Adds an exception to a channel ban.*

**f** No client message formatting (IRCX)

*Instructs the client software to not perform message formatting.*

**g** Operator guard

*Prevents channel hosts from giving members host status.*

**h** Hidden channel (IRCX)

*Hides the channel from channel listings and can only be queried if exact name is used.*

**i** Invite only

*Only clients with an invitation can join the channel. Invitations can be granted using the INVITE command or adding an INVITE entry to the channel access list.*

**j** Restrict access list

*Restricts use of the channel access list to only the channel owner.*

**k** Password protected

*Usage: +k <password>*

*Requires an entry password to join the channel. Password must be supplied using the JOIN command.*

**l** Member limit flag

*Usage: +l <limit>*

*Set the maximum number of members allowed to enter the channel at one time.*

**m** Moderated channel

*Prevents regular members from speaking inside the channel unless they have been given voice privileges by a channel host/owner.*

**n** No external messages

*Blocks the sending of messages into the channel from users who are not inside.*

**o** Channel operator flag

*Usage: +o <user>*

*Allows you to give channel host status to the specified user.*

**p** Private channel

*Hides the topic in channel listings and users cannot query the member list from outside.*

**q** Channel owner flag (IRCX)

*Usage: +q <nickname>*

*Allows you to give channel owner status to a member. Can only be set by the channel owner.*

**r** Registered channel (IRCX)

*Prevents the channel from being automatically destroyed when empty.*

**s** Secret channel

*Hides the channel from channel listings and cannot be queried by non-members.*

**t** Only operators can change topic

*Prevents regular members from changing the topic of the channel.*

**u** Show knock notifications (IRCX)

*Displays notifications to the channel hosts/owners when users fail to join the channel.*

**v** Voiced flag

*Usage: +v <user>*

*Allows you to give voice privileges to a member inside a moderated channel.*

**w** No whispering flag (IRCX)

*Disables the ability for members to use the WHISPER command inside the channel.*

**x** Auditorium (IRCX)

*Restricts the visibility and messaging within a channel. Members can only see themselves and the hosts/owners in the channel.*

**y** Lock channel modes

*Prevents the channel hosts from changing the modes.*

**z** Service channel (IRCX)

*Indicates that a service is monitoring / running on the channel.*

**A** Controlled Locally

*Allows only the local server administrator to gain control of the channel.*

**C** Clear channel access on empty

*Automatically clears the channel access list when the last member leaves.*

**E** Echo messages back to source (IRCX)

*Echoes messages back to the sender.*

**F** Filter message content

*Enables the content filter to censor messages sent to the channel.*

**I** Invite flag

*Usage: +I <nick!userid@hostname>*

*Allows users matching the specified mask to enter an ‘Invite only’ channel.*

**K** Cannot change channel passwords

*Prevents the channel owner from changing the channel passwords.*

**L** Auto Start Channel

*Allows channel to remain open and appear in channel listing when empty.*

**M** Server operators only

*Allows only server operators to join the channel.*

**P** Permanent channel

*Protects the channel registration from being removed by the channel owner or automatically if the channel is unused.*

**T** Transcript flag

*Saves the channel activity and conversations to a transcript file.*

## 5. User Modes

**a** Administrator flag

*Indicates user is a Server Administrator.*

**d** Invisible on channels

*Allows a service agent to monitor a channel without appearing in the member list.*

**e** Does not wish to send or receive DCC

*Prevents the sending or receiving DCC request used for secure chat and file transfers.*

**g** Global operator flag

*Indicates user is a Global Operator.*

**h** Help operator flag

*Allows server operators to receive help messages send using the HELPOPS command.*

**i** Invisible flag

*Hides user from appearing in user lists except from members on the same channel.*

**o** IRC operator flag

*Indicates user is a Server Operator.*

**p** Does not wish to send or receive CTCP

*Prevents the sending or receiving of CTCP messages used for queries clients directly.*

**r** Restricted connection

*Indicates user is being restricted by Client Rule (class) constrains.*

**s** View server messages

*Allows user to receive server notices.*

**w** View wallops messages

*Allows user to receive WALLOPS messages.*

**x** IRCX Mode (IRCX)

*Indicates user is in IRCX mode.*

**y** View whois and kick attempts

*Sends notification if a user attempts to kick you or performs a WHOIS on you.*

**z** Gag flag

*Usage: MODE <user> z*

*Allows server operators to gag users without their knowledge.  Gagged users cannot send messages to channels or users.*

**A** Anonymous flag

*Allows Server Administrators to hide their status and remain anonymous.*

**C** Prefers to speak in Chinese

*Indicates that the user prefers to speak in Chinese.*

**E** Prefers to speak in English

*Indicates that the user prefers to speak in English.*

**F** Prefers to speak in French

*Indicates that the user prefers to speak in French.*

**G** Prefers to speak in German

*Indicates that the user prefers to speak in German.*

**I** Prefers to speak in Italian

*Indicates that the user prefers to speak in Italian.*

**J** Prefers to speak in Japanese

*Indicates that the user prefers to speak in Japanese.*

**L** Filter incoming messages

*Filters private messages for inappropriate content using server-side filtering.*

**P** Prefers to speak in Portuguese

*Indicates that the user prefers to speak in Portuguese.*

**R** Readable text

*Strips out text formatting and colours codes from all incoming communication.*

**S** Prefers to speak in Spanish

*Indicates that the user prefers to speak in Spanish.*

## 6. Chat Services

The interface to access Chat Services are completely built into the server. Traditionally service agents (such as ChanServ, NickServ and MemoServ) have been used because it has been an area not covered in any IRC protocols until IRCX was introduced that handled the channel registrations. We extended the protocol to handle nickname registrations using existing commands and added a new command for memo services. The benefits of having the services interface being built-in allows client software to properly interface using server commands and numeric replies.

*Nickname Services*

| | |
|---|---|
| To register a nickname use:<br>/register <password> | To unregister a nickname use:<br>/unregister |
| To setup an allow list, use:<br>/access <nick> add owner <mask> | To view all registration settings use:<br>/prop <nick> * |
| To change a setting use:<br>/prop <nick> <property> :<new value> | To change password use:<br>/prop <nick> ownerkey <new password> |

*Channel Services*

| | |
|---|---|
| To register a channel use:<br>/register <channel> [<ownerkey>] | To unregister a channel use:<br>/unregister <channel> |
| To add an AOP use:<br>/access <channel> add host <mask> | To view current AOPs use:<br>/access <channel> |
| To remove an AOP use:<br>/access <channel> delete host <mask> | To change password use:<br>/prop <channel> ownerkey <new password> |

Use standard IRC/IRCX commands to manage your channel registration. Supply your channel password using the join command to gain channel owner status upon entering.

*Memo Services*

| | |
|---|---|
| To list your memo box use:<br>/memo list | To read a memo use:<br>/memo read <memo id> |
| To send a memo use:<br>/memo send <nick> <message> | To delete a memo use:<br>/memo delete <memo id> |
| To undelete a memo use:<br>/memo undelete <memo id> | To purge your deleted memos use:<br>/memo purge |

### 6.1 Channel Properties

| | |
|---|---|
| OID (R/O) | Internal object identifier for the channel. |
| NAME (R/O) | Name of the channel. |
| PEAK (R/O) | Highest recorded member count. |
| ACCOUNT | Nickname registration channel is associated with. |
| CREATION (R/O) | Time that the channel was created. |
| CREATOR (R/O) | Identity of the channel creator. |
| MANAGED | Location of channel registration. |
| CLASS | Allows only users of the specified client rule (class) to enter. |
| CLIENTGUID | Only allows clients using the specified protocol to enter. |
| CAPS | Converts messages with capitals to lowercase. |

| | |
|---|---|
| JOINS | Prevents a rapid influx of users entering the channel. |
| LAG | Adds an artificial delay between messages from the same user. |
| RATE | Adds an artificial delay between all messages sent to channel. |
| LANGUAGE | Preferred language to be spoken inside channel. |
| OWNERKEY | Password used to gain owner status when entering channel. |
| HOSTKEY | Password used to gain host status when entering channel. |
| VOICEKEY | Password used to gain voice status when entering channel. |
| MEMBERKEY | Keyword required to enter the channel. |
| PICS | Current PICS rating of the channel. |
| TOPIC | Current topic of the channel. |
| SUBJECT | Subject keywords for search engines |
| CLIENT | Client-specified information. |
| ONJOIN | Message sent to users after entering the channel. |
| ONPART | Message sent to users after leaving the channel. |

## 6.2 Nickname Properties

| | |
|---|---|
| OID | Internal object identifier for the nickname registration. |
| NAME | Name of the nickname registration |
| CREATION | Time that the nickname registration was created. |
| MANAGED | Location of nickname registration. |
| CANNOTRECEIVEMEMO | Nickname registration cannot be used to receive memos. |
| CANNOTSENDMEMO | Nickname registration cannot be used to send memos. |
| EMAIL | Email address of the owner. |
| LASTSEEN | Time when the nickname registration was last used. |
| MEMOSIZE | Limits the size of the memobox. |
| OWNERKEY | Password used to authenticate as the registration owner. |
| PROTECTION | Identification enforcement option. |
| REALNAME | Real name of the owner. |
| REGLIMIT | Maximum channel registrations allowed. |
| SECURE | Only allows local administrators to manage the registration. |
| VERIFIED | If the value is 0, nickname awaiting email verification. |

## 7. Remote Administration

All the functionality provided using the Remote Control utility can be performed via IRC commands.

The CREATE command can be used to create the following objects:

| | |
|---|---|
| <channel> | Channel (Chat Room) |
| <server> | Server Linking Profile |
| <nickname> | Nickname Registration |
| $C_<class> | Client Rule (Class) |
| $F_<filter> | Content Filter |
| $F_<filter>_<word> | Word entry for a Content Filter |
| $G_<group> | Server Operator Group |
| $O_<account> | Server Operator Account |
| $P_<scan> | Port Scan |
| $V_<check> | Version Check |

To view and change objects properties, use the PROP command.  Masks can be added using the ACCESS command.  To delete an object use the DESTROY command.

To view the current properties of the default Client Rule (class) you can use the following commands:

```
/PROP $C_Default *
$C_Default Name Default
$C_Default Order 2
$C_Default Description Default Connection Rule
$C_Default IsDefault 1
$C_Default StartTime 12:00
$C_Default StopTime 12:00
$C_Default ClientProtocol 0
$C_Default LogonMode 0
$C_Default ReverseDNSClients 0
-snip-
$C_Default End of properties
```

```
/ACCESS $C_Default
$C_Default Start of access entries
$C_Default GRANT *!*@*$*:* 0 SYSTEM
$C_Default End of access entries
```

*Updating Message of the day*

| | |
|---|---|
| To view the MOTD use:<br>/prop $C_Default motd | To clear the MOTD use:<br>/prop $C_Default motd : |
| To add a line of text to MOTD use:<br>/prop $C_Default motd <text> | To add a blank line to MOTD use:<br>/prop $C_Default motd \n |

All server log files for channels, server events, and private messages can be accessed using the SLOG command.

The server settings can be viewed and changed using the PROP $ command.
For example you can turn off news flashes using "/PROP $ NewsFlash 0"

## 8. Frequently Asked Questions

How do I connect to my own server?
To connect to your server, you need to use a piece of software called an IRC client. For the server address, you should use "127.0.0.1". If you are using mIRC, you can directly type "/server 127.0.0.1:6667" in the status window.

How do I configure my server to accept Internet connections?
Make sure you configure your firewall to accept inbound connections to port TCP 6667. Try defining additional chat ports because Internet Service Providers (ISPs) sometimes block inbound traffic to certain ports such as 6667. If you are running the server within a private home/office network, you are most likely using NATs and will need to setup Port Forwarding (Special Port) on your router. To setup Port Forwarding, specify TCP 6667 as both the external and internal port and use the private IP address of your chat server. If having difficulties configuring your firewall/router, you should contact the manufacturer.

Why do I get "No Authorization for this server"?
Check your client rules (classes) to make sure they are accepting clients.

How do I become an IRC Operator?
You can become an IRC Operator by using the command “/OPER <account> <password>” in the status window.

How do I ban a user from my server?
You can ban a user from your server using the command “/KLINE <user> <comment>”.

How do I get Ops in a registered channel?
Leave the channel and re-join using the command “/JOIN <channel> <password>.

I have a friend who is running a server, can we link?
You can server link with OfficeIRC, DreamForge and UnrealIRCD servers.  See the Server Linking section for more information.

Can I bind an IP Address to the chat ports?
You need to specify the IP Address in the “Only this IP” field located in the “General Settings” section of the Remote Control utility.

How do I obtain a domain name for a server with a dynamic IP?
You need to create an account with a provider such as www.no-ip.com or www.dyndns.org and obtain a Dynamic DNS then, enter this account information in “DDNS Service” located in the “General Settings” section of the Remote Control utility. Whenever your server is started or a dial-up connection is detected, your new IP address will be automatically synchronized.

How can my clients connect using a web browser?
You need to download and install a java IRC client such as JPilot on your website.  Most java IRC clients require you to host the applet on the same server IP address.

I am hosting a special event, how can I control things?
You can use features such as “Moderated Channel” to only allow selected members to speak within the channel.  If you are accepting questions or running a support channel, you can create an “Auditorium Channel” which restricts the visibility of members and allows only the channel owner and hosts to view messages.

How do I log the conversations inside a chat room?
You must enable “Channel Transcripts” located in the Transcripts section of the Remote Control utility and each channel then needs to have ‘Save Channel Transcript’ checked.

What are the system requirements?
Requires the .Net Framework 2.0 which is available on Microsoft Windows 2000, XP, 2003 and Vista though the Windows Update.  You will require a network connection with a fast enough upload speed to handle the anticipated load.  We recommend allocating 2Kbps for each concurrent user connection.  For example to host a chat community of 500 concurrent users, you will need an Internet connection with a 1Mbps upload rate.

Check for updates at http://www.officeirc.com