Mumble Server (Murmur) Installation Guide

ATAK

V1.0.0

Purpose

This document provides information on the installation of the Mumble server, Murmur. Murmur is an open source server that implements the Mumble protocol.

The terms “Mumble Server” and “Murmur” are used interchangeably within this document.

Table of Contents

Prerequisites

To allow the installation and some maintenance tasks, the user account must administrative rights.

For a Linux installation, this means the account having *sudo* rights.

For a Windows installation, this means that the user has, or can elevate commands to, a local administrator account.

Installation on Linux

Debian Based Distribution (e.g. Ubuntu)

Debian based Linux distributions can install Mumble directly from the application repository.

These installation instructions are taken from the official online Mumble installation guide, which should be checked for any changes before installation commences.

https://wiki.mumble.info/wiki/Installing_Mumble#Linux

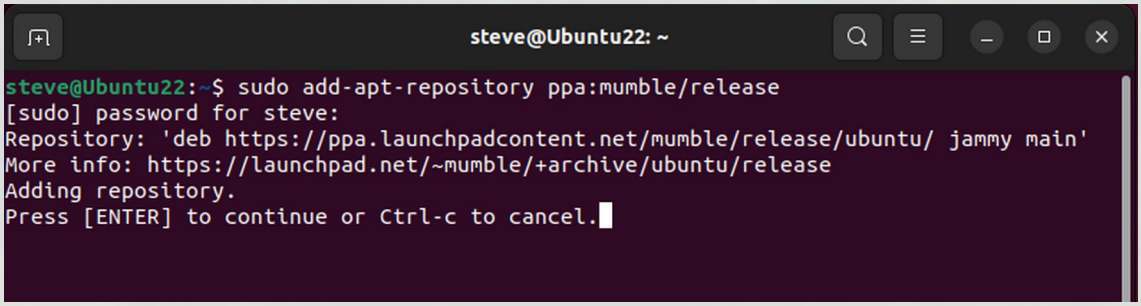
Although it is possible to install Mumble directly from the Debian Personal Package Archives (PPA), it is possible that the PPA contains an outdated version.

To ensure the latest version is installed, add the Mumble PPA before installation by opening a terminal and entering the following commands.

```
sudo add-apt-repository ppa:mumble/release
```

Add Mumble PPA Repository

When prompted, press ENTER to continue.



Update Local Packages

Once the PPA has been successfully added, update the local package information by running the following command.

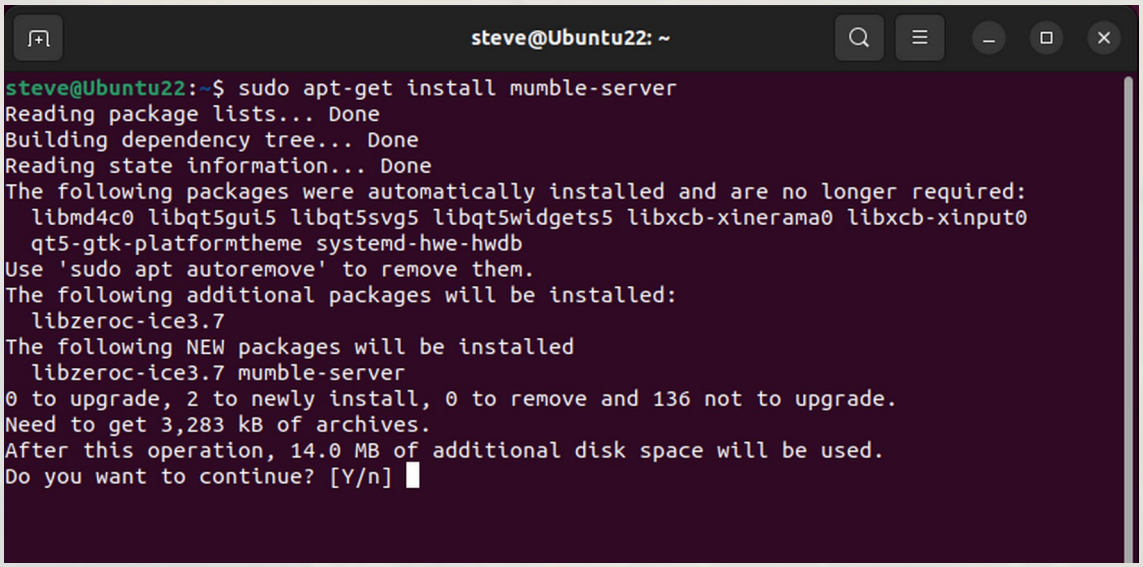
```
sudo apt-get update
```

Install Mumble Server

Run the following command to install the Mumble server.

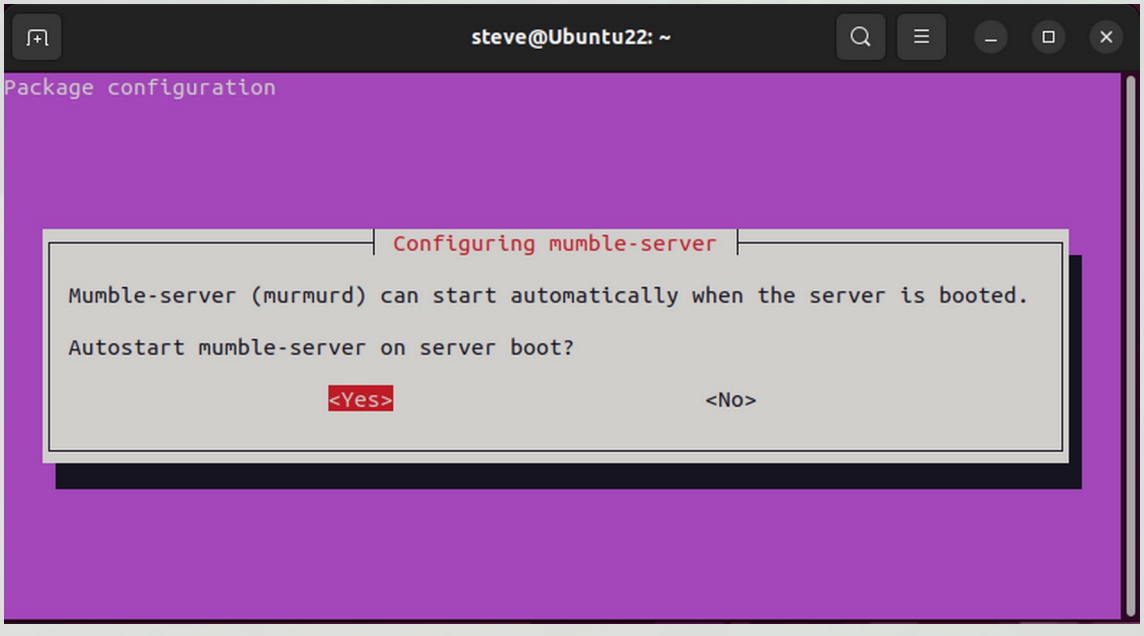
To install the Mumble Server (Murmur) enter the following.

```
sudo apt-get install mumble-server
```



Once installed, perform initial server configuration by entering the following command and answering the questions when prompted.

```
sudo dpkg-reconfigure mumble-server
```



Default configuration:

Configuration	Default Value
Auto start server on boot?	Yes
Allow Mumble server to use higher priority?	No
Enter SuperUser account password	*Enter password*

OPTIONAL: Install Mumble Client

The Mumble client can also be installed by opening a terminal and running the install command.

```
sudo apt-get install mumble
```

Red Hat Based Distribution (e.g. CentOS)

These installation instructions are taken from the official online Mumble installation guide, which should be checked for any changes before installation commences.

https://wiki.mumble.info/wiki/Install_CentOS7

Download Installation File

Download the tar.gz installation file from Mumble releases page on GitHub.

https://github.com/mumble-voip/mumble/releases

Once downloaded, the file must be unzipped into a new folder by running the following commands in a terminal window.

```
tar -vxjf ./murmur-static_x86-1.3.4.tar.bz2
sudo mkdir /usr/local/murmur
sudo cp -r ./murmur-static_x86-1.3.4/* /usr/local/murmur/
sudo cp ./murmur-static_x86-1.3.4/murmur.ini /etc/murmur.ini
```

Create User and Group

Once the Mumble server has been unzipped into the correct location, a new user and group must be created to run the server.

```
sudo groupadd -r murmur
sudo useradd -r -g murmur -m -d /var/lib/murmur -s /sbin/nologin
murmur
sudo mkdir /var/log/murmur
sudo chown murmur:murmur /var/log/murmur
sudo chmod 0770 /var/log/murmur
```

Initial Configuration

Before starting the server for the first time, the following changes must be made to the `/etc/murmur.ini` configuration file.

```
database=/var/lib/murmur/murmur.sqlite
logfile=/var/log/murmur/murmur.log
pidfile=/var/run/murmur/murmur.pid

# Reminder: When changing the port that murmur will listen to you
will need to    also update the firewall.
# Update the firewall by editing /etc/firewalld/services/murmur.xml
# Then run "sudo firewall-cmd --reload"
port=64738

# Comment out the following setting since the service will already
be executing   as the correct user:
# uname=murmur
```

Run Mumble Server As a Background Service

To run the Mumble server as a background service, a systemd unit file must be created.

Using a text editor, open an editor and create a new file named:

`/etc/systemd/system/murmur.service`.

Paste the following into the file.

```
[Unit]
Description=Mumble Server (Murmur)
Requires=network-online.target
After=network-online.target mariadb.service time-sync.target

[Service]
User=murmur
Type=forking
ExecStart=/usr/local/murmur/murmur.x86 -ini /etc/murmur.ini
PIDFile=/var/run/murmur/murmur.pid
ExecReload=/bin/kill -s HUP $MAINPID

[Install]
WantedBy=multi-user.target
```

Regenerate the Murmur pid Folder

On some systems, the `/var/run` folder may be deleted after a reboot. To regenerate the pid folder for Murmur, create a configuration file under `/etc/tmpfiles.d/murmur.conf` and add the following as the file contents.

```
d /var/run/murmur 775 murmur murmur
```

Rotate Logs Files

To set up log rotation create the `/etc/logrotate.d/murmur` file with the following content.

```
/var/log/murmur/*log {
    su murmur murmur
    dateext
    rotate 4
    missingok
    notifempty
    sharedscripts
    delaycompress
    postrotate
        /bin/systemctl reload murmur.service > /dev/null 2>/dev/null
|| true
    endscript
}
```

Firewall Changes

The firewall must be updated to allow TCP/UDP traffic through to the Murmur server. If the default port has been changed, then this must be reflected in the firewall rules.

Create a new configuration file `/etc/firewalld/services/murmur.xml` and add the following content.

```
<?xml version="1.0" encoding="utf-8"?>
<service>
  <short>Murmur</short>
  <description>Mumble Server (Murmur)</description>
  <port protocol="tcp" port="64738" />
  <!-- Reminder: Update /etc/murmur.ini so that it uses the same
ports -->
  <port protocol="udp" port="64738" />
</service>
```

Then add the new firewall rule to the default zone and reload.

```
sudo firewall-cmd --permanent --add-service=murmur
sudo firewall-cmd --reload
```

Finishing Up

Update the system so that it is ready to start the murmur service

```
sudo systemd-tmpfiles --create /etc/tmpfiles.d/murmur.conf
sudo systemctl daemon-reload
```

To temporarily start the service (until the next reboot):

```
sudo systemctl start murmur.service
```

To tell your system to auto-start the murmur service when booting (will NOT immediately start murmur, only every time the system is booted):

```
sudo systemctl enable murmur.service
```

Installation on Windows

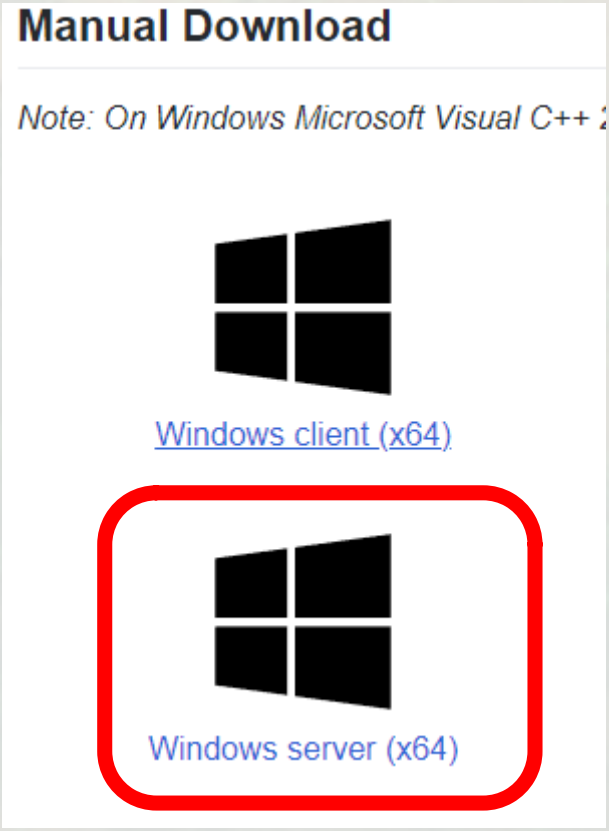
Before installing and configuring the mumble server, ensure:

1. You have a static IP in order to port forward your router/modem for the voice chat server
2. You have port forwarding enabled to the port you want the server to use (default port is 64738)
3. Configure your network/firewall to allow the port for murmur (default is 64738 in TCP and UDP)

Download the installation file

Download the mumble server installer from:

https://www.mumble.info/downloads/

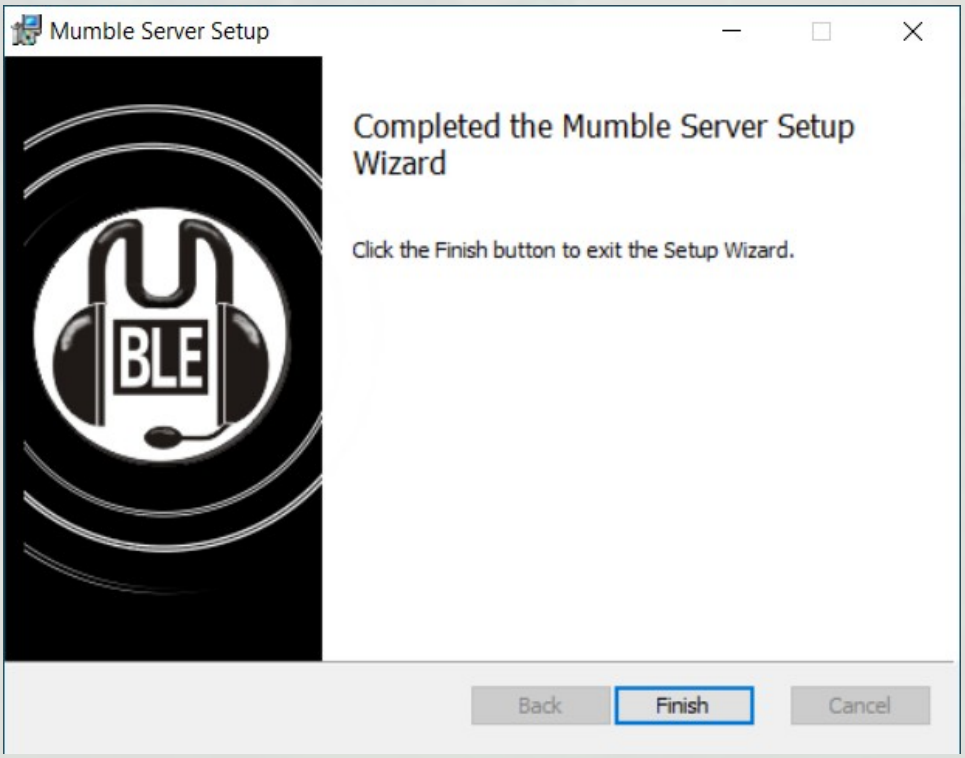


Then open the installed file

Mumble server set-up wizard

Accept the License Agreement and select “Install”:

Select “Finish” when installed:



Mumble server configuration file

You have to manually create the murmur.ini configuration file.

The contents of the file can be found here:

https://github.com/mumble-voip/mumble/blob/master/auxiliary_files/mumble-server.ini

1. Create a new folder called ‘Mumble’ in
    i. *C:/Users/*Name of user*/AppData/Local/*
    ii. Within that folder create another folder called ‘Murmur’
        1. Alternatively, right click on the mumble-server application and ‘Run as Administrator’
        2. A text file called ‘murmur’ will be created where the mumble server has been installed (usually C:\Program Files\Mumble\server)
        3. Open the text file and find where the programme is “Initializing settings from” and create the folders to match that path
2. Within that folder create another folder called ‘Murmur’
3. Create a text file
4. Copy and paste the configuration from the URL above
5. Save it as ‘murmur.ini’ inside the Murmur folder you have just made in the C drive

In this configuration file it recommended to change:

```
welcometext= “Change your welcome text here”
port=
serverpassword=
bandwidth=
users=
registerName=
serverpassword=
```

“welcometext” is the text that will welcome people to your server

“port” is to be used if you want to use other than the default

“bandwith” to set the maximum bandwidth clients are allowed

“users” is the maximum users allowed on the server

“registerName” will need to be un-commented and enter what you want your server to be called

“serverpassword” create a password for the server

Save the text file and now restart the mumble server by:

1. Right-click on the murmur icon in your task bar area
2. Select “Quit Murmur”
3. Re-open/run (as administrator) the murmur server application

You can now install the client and connect to the server.

General Mumble Server Configuration

Mumble Client Download

Some server configuration can only be carried out using a Mumble client client. If not already installed, this can be accomplished using the instructions found on the Mumble download page.

https://www.mumble.info/downloads/

Server Configuration File

Editing the configuration file allows full control over the server configuration.

For Debian systems, the configuration file can is located at:

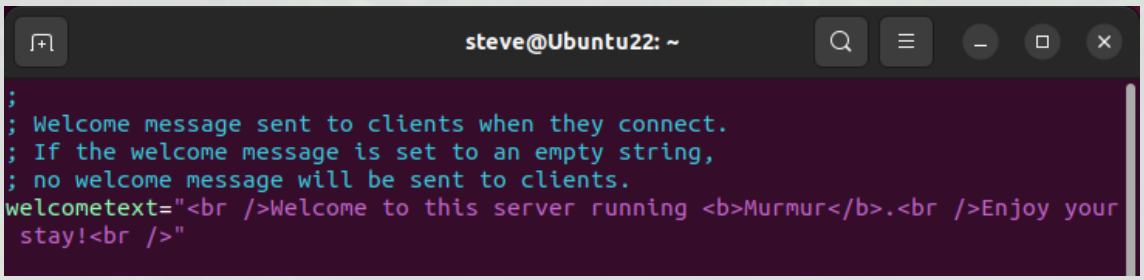
```
/etc/mumble-server.ini
```

while on Red Hat systems it can be found at:

```
/etc/murmur.ini
```

Welcome Message

When a user connects to Mumble, a welcome message is displayed. This can be edited in the configuration file by setting the `welcometext` entry. If no text is specified, then no message is sent to the user when they connect.



Configuring Certificates

To create a self-signed cert that is fulfils all requirements for the Plugin, you can run the following command:

```
openssl req -x509 -sha256 -nodes -days 1080 -newkey rsa:2048 -keyout
(servername).key -out (servername).cer -subj '/CN=(IP Address)' -addext
extendedKeyUsage=1.3.6.1.5.5.7.3.1
```

Where (servername) is the name of your server and (IP Address) is the IP address mumble is using.

For example, if your server has the name "**test**" and IP address "**192.168.1.1**" the command will look as follows:

```
openssl req -x509 -sha256 -nodes -days 1080 -newkey rsa:2048 -keyout
test.key -out test.cer -subj '/CN=192.168.1.1' -addext
extendedKeyUsage=1.3.6.1.5.5.7.3.1
```

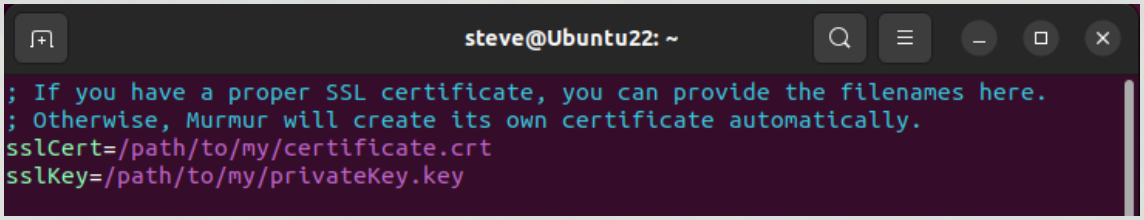
If your host has multiple IP Addresses on it, for example the host IP is "IP Address 1" but mumble runs on "IP Address 2", you will need to modify the command as such:

```
openssl req -x509 -sha256 -nodes -days 1080 -newkey rsa:2048 -keyout
(servername).key -out (servername).cer -subj '/CN=(IP Address)/CN=(IP
Address 2)' -addext extendedKeyUsage=1.3.6.1.5.5.7.3.1
```

Using the earlier example and a second IP address of "**10.1.0.1**", the command will look as such:

```
openssl req -x509 -sha256 -nodes -days 1080 -newkey rsa:2048 -keyout
test.key -out test.cer -subj '/CN=192.168.1.1/CN=10.1.0.1' -addext
extendedKeyUsage=1.3.6.1.5.5.7.3.1
```

The certificate used should be valid and have a complete certificate chain ideally which goes back to a root Certificate Authority (CA). Once stored on the server, the path to the certificate its private key file are specified by editing `sslCert` and `sslKey` entries in the file.

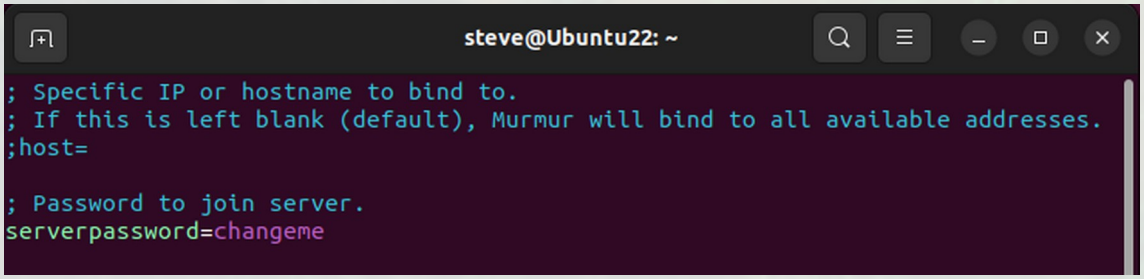


If mumble is installed through Docker Compose, the certificates are added in a slightly different way, this can be achieved by mounting a directory containing the files into your docker container and then setting environment variables. This can be seen in this example Docker compose file:

```
services:
     mumble-server:
          image: mumblevoip/mumble-server:[tag]
          container_name: mumble-server
          hostname: mumble-server
          restart: on-failure
          ports:
               -64738:64738
               -64738:64738/udp
#         expose:
#              - 6502
          volumes:
               [directory your cert is in]:/data/
          environment:
               -MUMBLE_SUPERUSER_PASSWORD=[password]
               -MUMBLE_CONFIG_SSLKEY=/data/
[servername].key
               -MUMBLE_CONFIG_SSLCERT=/data/
[servername].cer
```

Server Password

Open the file in a suitable editor and edit the `serverpassword` entry.



Certificate Validation

In order to work with the Voice ATAK plugin, the certificate must be valid as follows.

Certificate

- The certificate subject DN must start with “CN=<<mumble host>>”.
- The certificate must be issued by a trusted issuer.

Certificate Alternative Names

- One of the certificate subject alternative names must match the following criteria:
  - The first entry must contain the value of an IP address (7) or DNS address (2), and
  - The second entry must contain the Entry 1 must contain the Mumble server host.

Certificate Purpose

- The certificate key usage must state that it may be used as an SSL certificate.

Certificate Chain

- The certificate must contain a valid issuer chain.

NOTE:

If no certificate/private key is supplied then Mumble will generate it's own self-signed certificate which will then be presented to clients when they connect.

Restarting the Mumble Server

The Mumble server must be restarted following any change in configuration. To restart, open a terminal session and enter the following command. Any connected users will be disconnected.

Debian Linux

```
sudo service mumble-server restart
```

Red Hat Linux

```
sudo systemctl restart murmur
```

Windows

To restart the mumble server in windows you have to right click on the mumble server logo and select "Quit Murmur"

Then re-open the application.

User Permissions

When installed, the server has a single admin user with the username *SuperUser*. To give admin rights to other users they must first have connected and registered.

NOTE: The *SuperUser* password was set during the server installation.

Editing Admin Permissions

1. Start the Mumble client and connect to the server using the username ‘SuperUser’ and the password that was set during installation.
2. Right click on the Root channel and select *Edit*.
3. Go to the *Groups* tab.
4. In the drop-down, select the *admin* group.
5. In the *Members* drop-down, enter the username of the user to be made admin (Note, you must type the name).
6. Click OK.

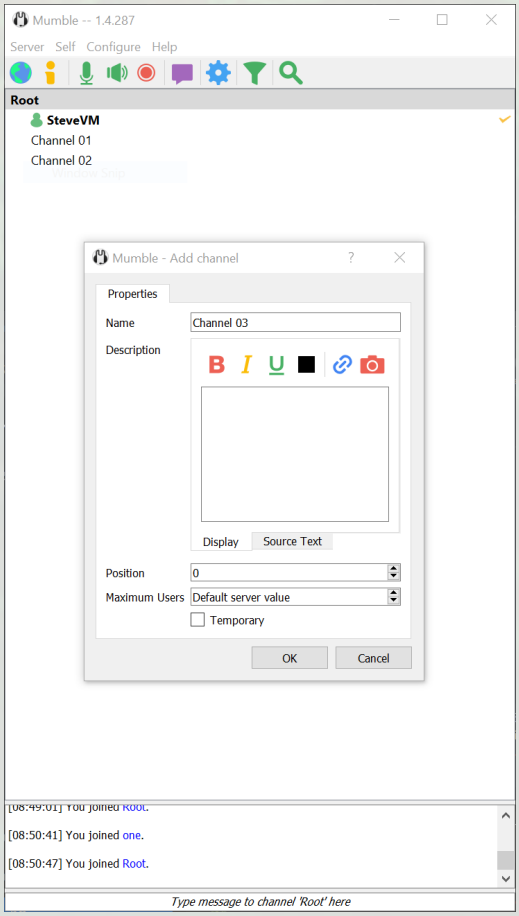
Creating Channels

1. Start the Mumble client and log into the Mumbler server with an admin account (see above for admin account configuration).
2. Right click on the *Root* channel and click *Add*.
3. Enter the new channel information as required:

   a) *Name* – the name of the channel.

   b) Description – a free text description.

   c) *Position* – Use to order channels, negative numbers are placed above the “0” position channel. Dragging channels to change order has the same effect.



d) *Maximum Users* – the max number of users allowed in the channel.

e) *Temporary* – If a channel is temporary then it will be automatically removed once all users have left it.

4. Click *OK*.