**[Host a Deep Web IRC Server for More Anonymous Chatting](https://null-byte.wonderhowto.com/how-to/host-deep-web-irc-server-for-more-anonymous-chatting-0180682/)**

* By [**TAKHION**](https://creator.wonderhowto.com/takhion/)
* 11/11/2017 6:53 am

Internet Relay Chat, or IRC, is one of the most popular chat protocols on the internet. This technology can be connected to [the Tor network](https://null-byte.wonderhowto.com/how-to/access-dark-web-while-staying-anonymous-with-tor-0179341/) in order to create an anonymous and secure chatroom, without the use of public IP addresses.

IRC servers allow one to create and manage rooms, users, and automated functions, among other tools, in order to administer an instant messaging environment. IRC's roots began in 1988 when Jarkko Oikarinen decided to attempt to implement a new chat protocol for users at the University of Oulu, Finland. Since then, it's been widely adopted and used as a lightweight means of communication.

This chat protocol has proven especially popular in tech communities, with channels for software user groups, support and help, and discussion areas for developers and programmers. This user base eventually expanded to include hackers and security activists, many of whom enjoyed the protocol but found it lacked certain elements of privacy.

For hackers, and anyone else who desires anonymity, combining IRC with a proxying system such as Tor proved to be a valuable solution. Learning to join and create Tor-connected IRC channels allows one to form private communication platforms, for those interested in protecting their identity.

Combining Tor and IRC is more complicated than simply downloading the [Tor Browser Bundle](https://www.torproject.org/projects/torbrowser.html), so we will explain how to configure a local IRC server, connect this server to the Tor network, and connect to the server as a client.

**Step 1Gather the Prerequisites**

An IRC server will be easiest to combine with Tor on a Linux system, and will be more secure on a distribution intended for server usage. Some popular choices for server operating systems are [Debian](https://www.debian.org/), [Ubuntu Server](https://www.ubuntu.com/download/server), or [CentOS](https://www.centos.org/). Any Linux distribution is technically suitable, and while [Kali](https://www.kali.org/) is used in this example, it is not necessarily the best option for a secure server.

The Linux installation used to host the server may be on a remote Virtual Private Server (VPS), a [Raspberry Pi](https://www.amazon.com.au/dp/B01CD5VC92/?tag=wnbau-22), or even just a personal server on your local network.

**Don't Miss:** [**How to Set Up a Headless Raspberry Pi Hacking Platform Running Kali Linux**](https://null-byte.wonderhowto.com/how-to/set-up-headless-raspberry-pi-hacking-platform-running-kali-linux-0176182/)

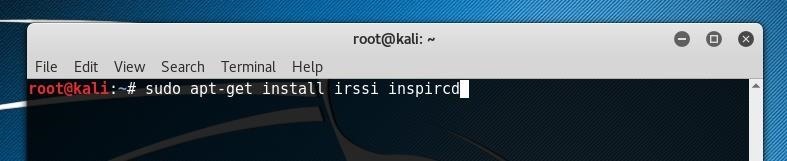
With our server itself selected, the first piece of software we'll need in order to host an IRC server is [InspiRCd](https://www.inspircd.org/). This will manage the entirety of the IRC component of our server. We'll also need to install [Tor](https://www.torproject.org/). For the server, it must be able to be called as a service, so it's best to install the Tor version available on Linux repositories.

On Debian-based distros, we can install can update our repositories using apt. On other systems, the modules are available from their respective websites. We can do this by opening a terminal window and typing the following.

sudo apt-get update

After our repositories are updated, we can install the necessary software by typing the command below.

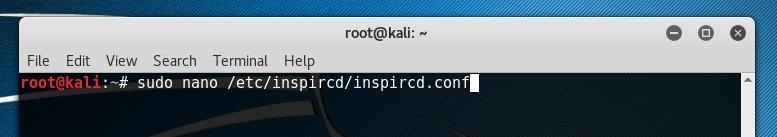
sudo apt-get install inspircd tor

[](https://img.wonderhowto.com/img/original/76/51/63644294457534/0/636442944575347651.jpg)

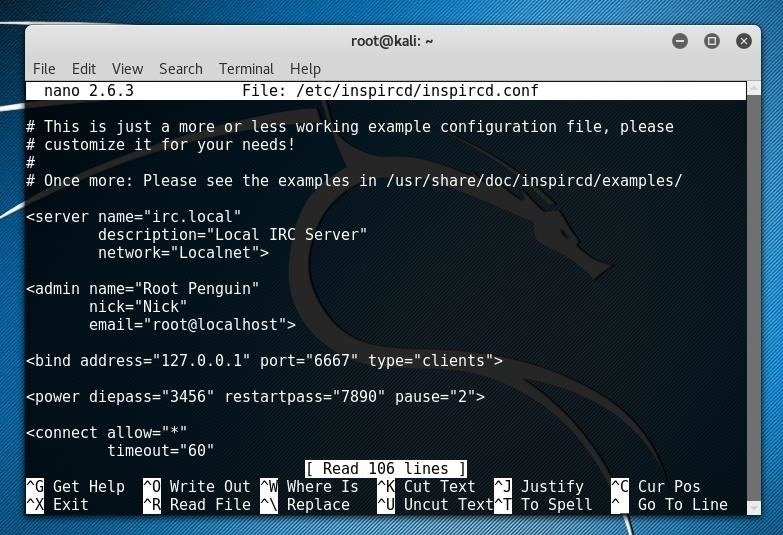
**Step 2Configure InspIRCd**

Once our IRC server daemon is installed, we can begin to configure it to meet our needs. The configuration file is available on Linux systems at /etc/inspircd/inspircd.conf. We can edit this file using Nano by running the following command.

sudo nano /etc/inspircd/inspircd.conf

[](https://img.wonderhowto.com/img/original/99/30/63644295961143/0/636442959611439930.jpg)

If IndspIRCd was installed correctly, the file which opens should appear similar to the one below.

[](https://img.wonderhowto.com/img/original/79/70/63644296017820/0/636442960178207970.jpg)

There are a number of configuration parameters which we should review and edit to suit our purposes. First, we'll want to look at the "server" parameter section seen below.

<server name="irc.local"  
description="Local IRC Server"  
network="Localnet">

We can change this to whatever we would like to name our IRC channel, but the "name" option must be formed in a similar format to "irc.website.com." While for most servers this would be important as it could be relevant to the URL which resolves to the IRC server's IP, but as we are going to instead make our server available over Tor, we can leave it whatever we choose. For this tutorial, I left these at the default "Local" settings.

**Don't Miss:** [**Access the Dark Web While Staying Anonymous with Tor**](https://null-byte.wonderhowto.com/how-to/access-dark-web-while-staying-anonymous-with-tor-0179341/)

Next, we should update the "admin" settings.

<admin name="Root Penguin"  
nick="Nick"  
email="root@localhost">

The "name" and "nick" should be changed to the preferred identification for your server admin. In order to make identifying these slightly more simple, I changed them both to "admin." I left the email setting as it was, as it does not need to be a valid email address.

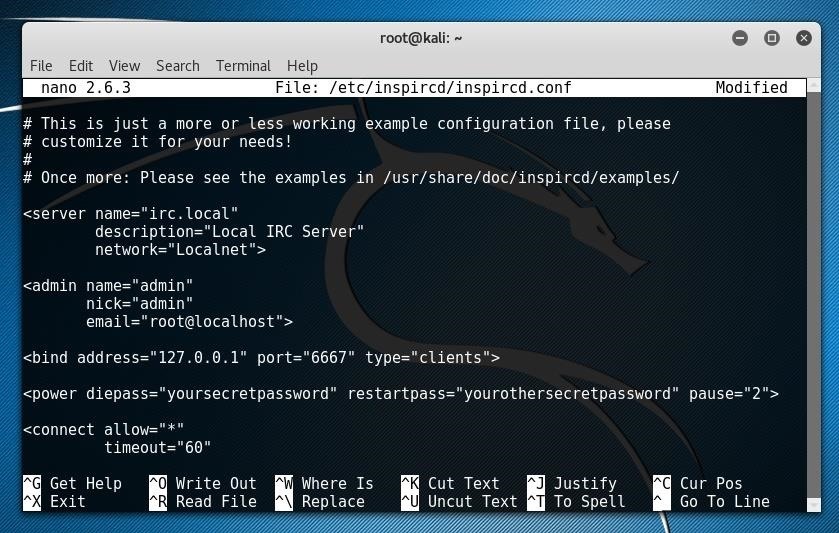
<admin name="admin"  
nick="admin"  
email="root@localhost">

In most cases, we'd want to change the "bind" value as well, but as all of our connections are going to be made through Tor, we can leave the address as 127.0.0.1 and the port at 6667 or the port of our choice.

<bind address="127.0.0.1" port="6667" type="clients">

One important field to update is the "power" section. This provides two passwords which could be used by yourself or channel operators to restart or turn off the server. Make sure to set them to something more secure than the default password or "password."

<power diepass="password" restartpass="password">

[](https://img.wonderhowto.com/img/original/43/06/63644298766478/0/636442987664784306.jpg)

The last sections of this configuration file we'll want to modify are the "oper" or operators section, as well as the "files" section. In the operators section, we'll want to update the operator credentials to provide server authority to the users we wish. The most important part to update is the "password" field. Be sure to change the password to something more secure.

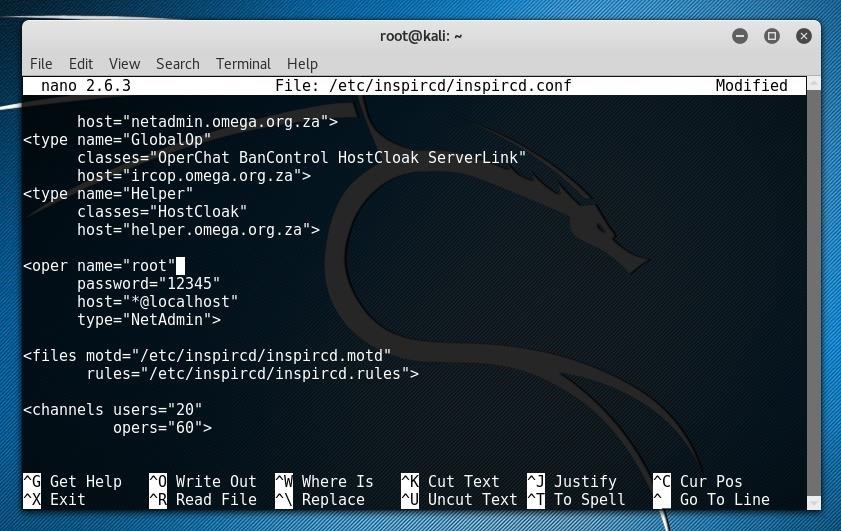
<oper name="root"  
password="12345"  
host="\*@localhost"  
type"NetAdmin">

Lastly, we'll want to define the location of the *Message of the Day* and *Rules* files.

<files motd="/etc/inspircd/inspircd.motd"  
rules="/etc/inspircd/inspircd.rules">

If these files are already set to an appropriate location, you can simply directly edit them to update these files in nano. These files will be displayed in the channel when a user requests either of them by running /mord or /rules.

nano /etc/inspircd/inspircd.motd  
nano /etc/inspircd/inspircd.rules

[](https://img.wonderhowto.com/img/original/18/93/63644299861591/0/636442998615911893.jpg)

We can further update this configuration file by referencing the [InspIRCd Wiki](https://wiki.inspircd.org/Configuration#.3Cserver.3E). Save the file by pressing *Ctrl+O* in Nano, and then press *Ctrl+X* to exit. Once the configuration file is complete, we can test our server locally.

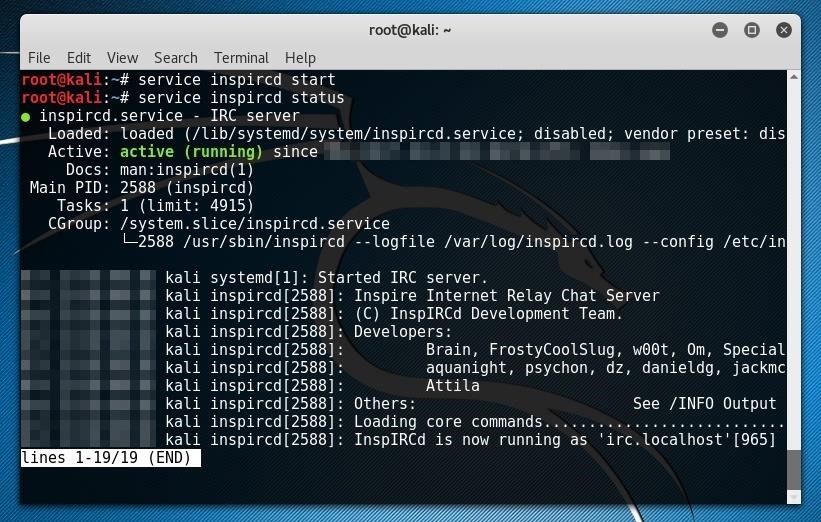
**Step 3Test the Local IRC Server**

First, we can start the server daemon using systemd by typing the following in terminal.

sudo service inspircd start

If the command succeeds, we can confirm it's status using the systemd **status** parameter.

sudo service inspircd status

[](https://img.wonderhowto.com/img/original/76/99/63644300258270/0/636443002582707699.jpg)

We can now test connect to our server using the IRC client of our choice. I used [Irssi](https://irssi.org/). Irssi is in most Linux repositories, and on Debian-based systems it can also be installed using [**apt**](https://null-byte.wonderhowto.com/how-to/hack-like-pro-linux-basics-for-aspiring-hacker-part-5-installing-new-software-0147591/), as seen below.

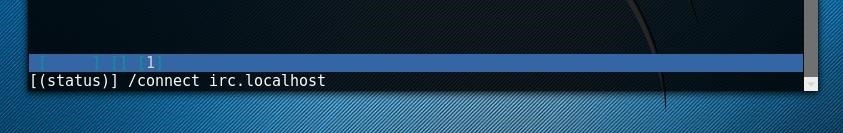
sudo apt-get install irssi

Irssi can be launched directly from the command line just by using its name.

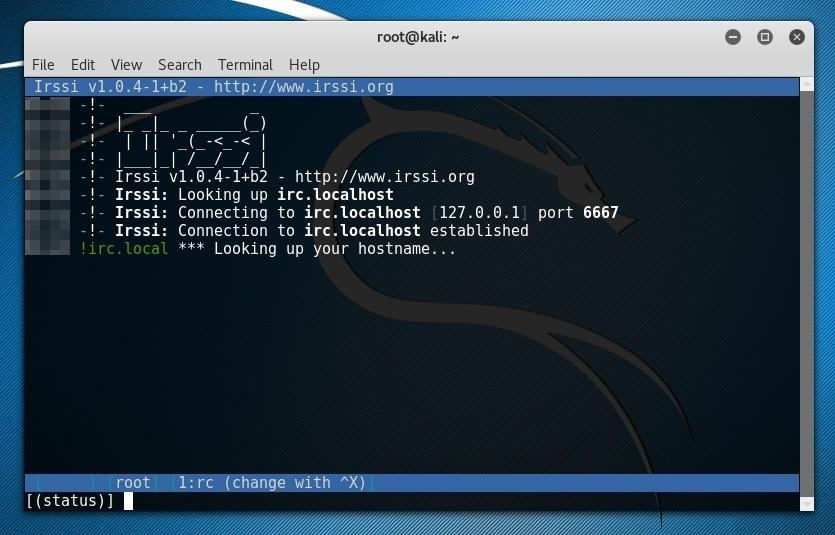
irssi

Within Irssi, we can connect to our server with the **connect** command.

/connect irc.localhost

[](https://img.wonderhowto.com/img/original/45/26/63644300721236/0/636443007212364526.jpg)

This command will connect us to our server. You should see something like the image below.

[](https://img.wonderhowto.com/img/original/81/38/63644300730627/0/636443007306278138.jpg)

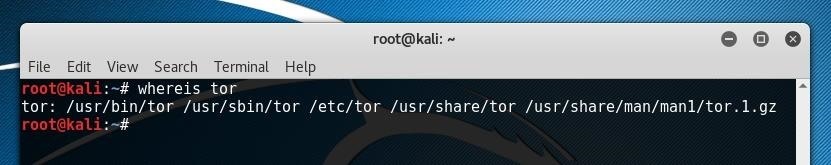
Once the IRC client has successfully connected to the local IRC server, it can be administered in the same way as a server which is connected to the internet. The test server is connected only to the machine which it is running, so the only users which could connect to the server must be on the local machine as well.

After the local IRC server connection is verified, and any IRC configurations are changed, we can disconnect from the local server and connect it to the Tor network. If you're using the Irssi IRC client, you can disconnect with **/quit**.

**Step 4Connect the IRC Server to Tor**

The Tor-routing of our IRC server will be configured in the torrc file. To find this file and confirm our Tor installation, we can use [**whereis**](https://null-byte.wonderhowto.com/how-to/hack-like-pro-linux-basics-for-aspiring-hacker-part-4-finding-files-0147327/) as in the example below.

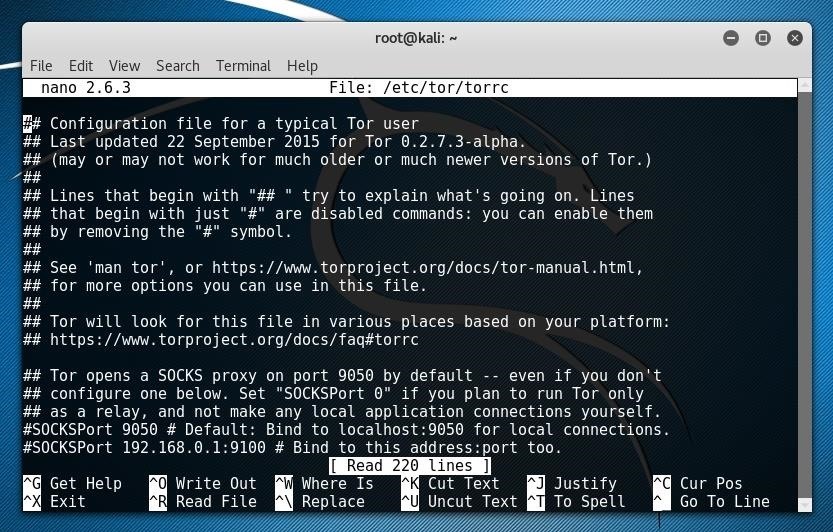
whereis tor

[](https://img.wonderhowto.com/img/original/85/48/63644301488862/0/636443014888628548.jpg)

This returns several locations which Tor uses for configuration. The "torrc" file we're looking for is most likely in /etc/tor. We can open it with the following command.

nano /etc/tor/torrc

Once opened, you should see something similar to below.

[](https://img.wonderhowto.com/img/original/36/87/63644301599550/0/636443015995503687.jpg)

The section we need to edit is after the following banner.

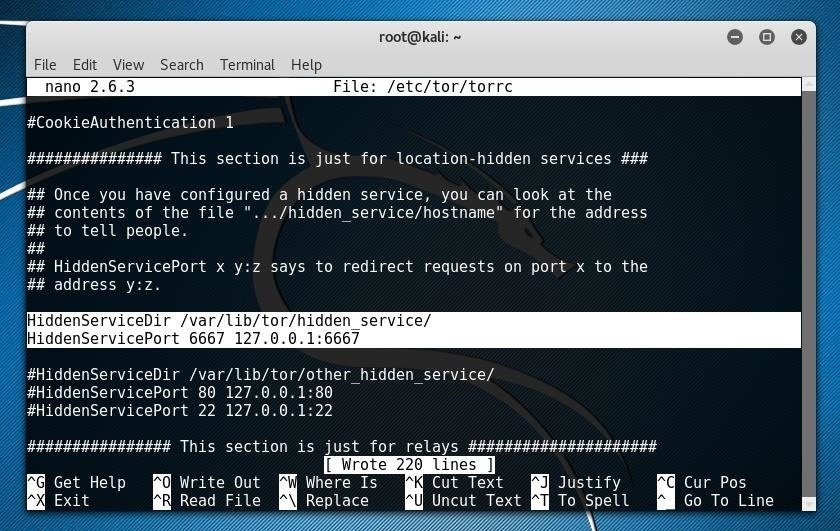
############### This section is just for location-hidden services ###

To skip directly to this section, press *Ctrl+W*, type "location-hidden" and press enter. In order specify our hidden service location to Tor, we'll want to un-comment the following two lines.

HiddenServicePort 80 127.0.0.1:80  
HiddenServiceDir /var/lib/tor/hidden\_service/

To do this, remove the "#" symbols at the beginning of these lines. We'll also want to edit one of the lines to specify our IRC port instead of port 80. When you're done, the line should look like this.

HiddenServicePort 6667 127.0.0.1:6667

[](https://img.wonderhowto.com/img/original/26/55/63644303280472/0/636443032804722655.jpg)

Write these changes to the file with *Ctrl+O*, and exit Nano with *Ctrl+X*.

Now that our Tor service is configured, we can launch it with systemctl by running the following command.

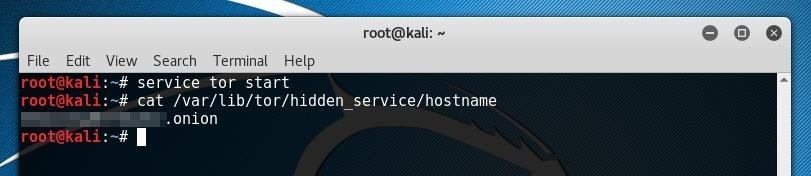
service tor start

Once the Tor service is running, an onion address should be generated and placed in /var/lib/tor/hidden\_service/hostname. We can retrieve this address with the [**cat**](https://null-byte.wonderhowto.com/how-to/hack-like-pro-linux-basics-for-aspiring-hacker-part-3-managing-directories-files-0147293/) command, seen below.

cat /var/lib/tor/hidden\_service/hostname

If needed, it's also possible to customize this onion address using tools like [Eschalot](https://github.com/ReclaimYourPrivacy/eschalot).

**Don't Miss:** [**How to Host Your Own Tor Hidden Service with a Custom Onion Address**](https://null-byte.wonderhowto.com/how-to/host-your-own-tor-hidden-service-with-custom-onion-address-0180159/)

[](https://img.wonderhowto.com/img/original/98/81/63644303741190/0/636443037411909881.jpg)

If **cat** returns an onion address, your IRC server is now available over Tor!

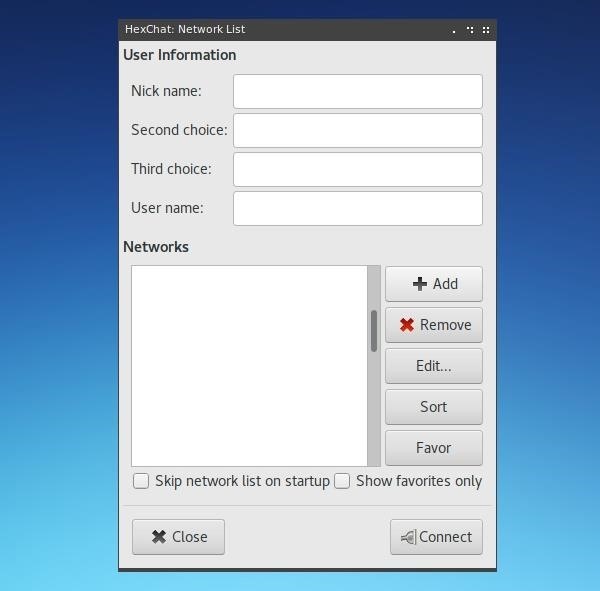
**Step 5Connect to an IRC Server Over Tor**

Now that your server is connected to Tor, it's available for anyone with the address to connect to it. As such, in order to test our server connectivity, it's best to do it from a different device than the one on which the server is running. If you've installed your server on a VPS or virtual machine, just switch back to using the device you configured it with.

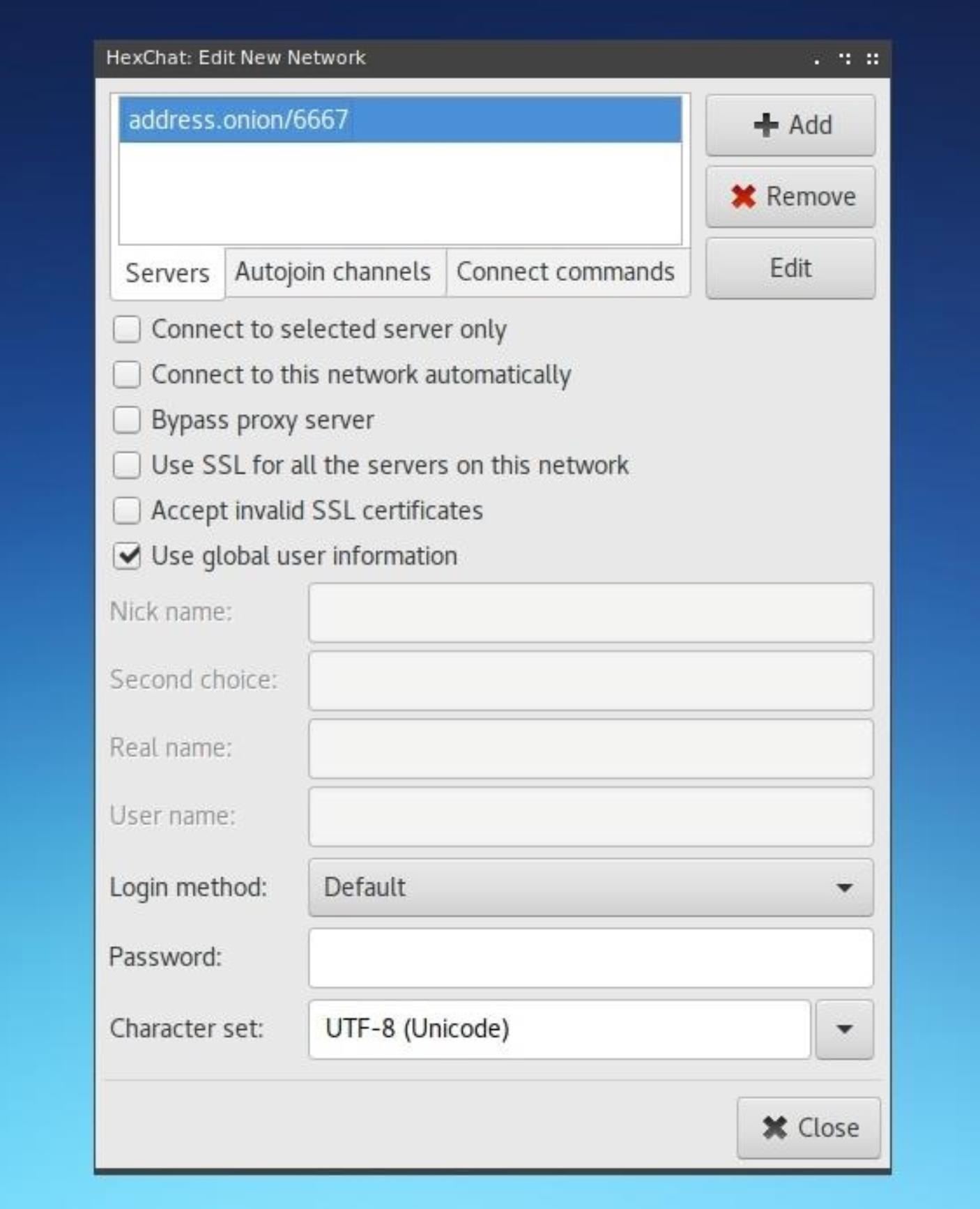
In order to test connecting to my server, I used an IRC client I already had installed, [HexChat](https://hexchat.github.io/). It's also available in most Linux repositories, and on Debian-based systems, can be installed with **apt** as seen below.

sudo apt-get install hexchat

After launching HexChat, set your preferred nickname choices, and then select "Add" in the networks section.

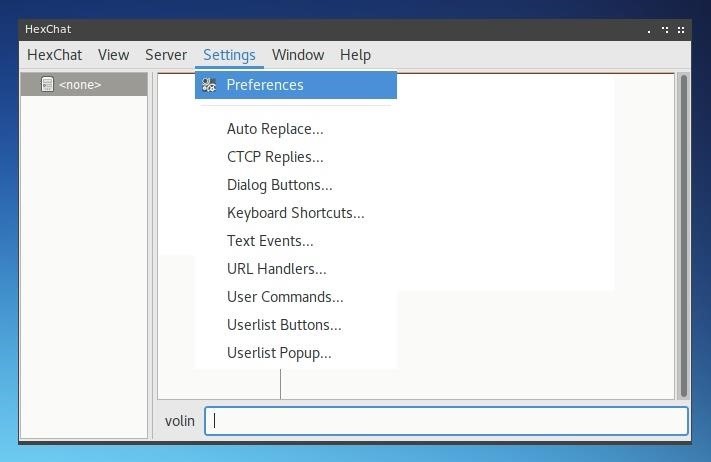
[](https://img.wonderhowto.com/img/original/15/84/63644305484144/0/636443054841441584.jpg)

Assign a name to your new network, and then right-click on it and hit "Edit."

[](https://img.wonderhowto.com/img/original/48/32/63645925842661/0/636459258426614832.jpg)

Within this edit window, update the address shown in the *Servers* box to be the onion address which was assigned to it by Tor in the previous step.

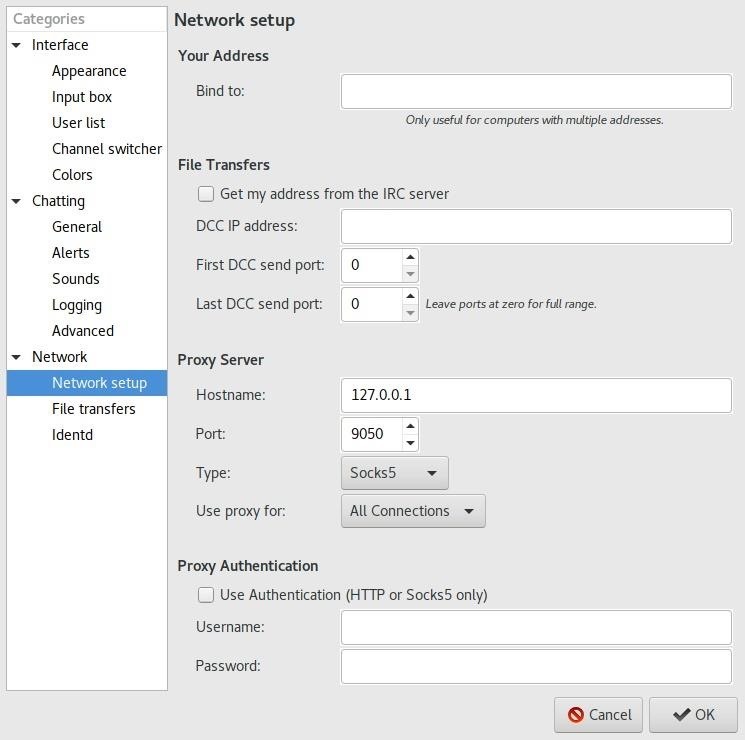
Close this window, and then select "Connect." The connection will not immediately be established, as we need to change our network settings to resolve the onion address. To do this, click on "Preferences" under the "Settings" menu on top of the new window which has opened.

[](https://img.wonderhowto.com/img/original/64/25/63644305919706/0/636443059197066425.jpg)

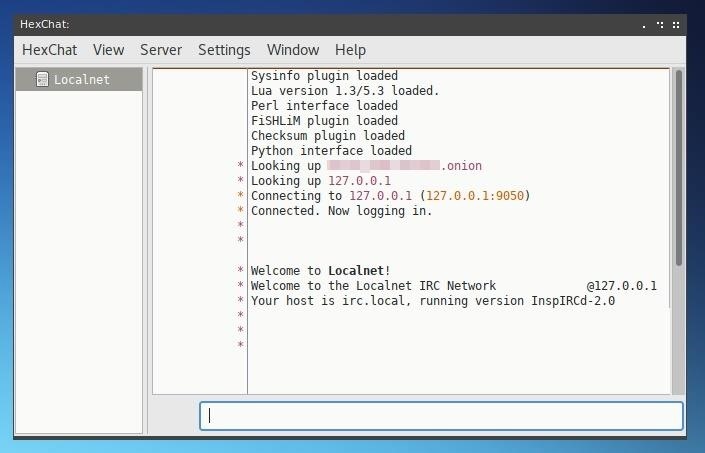
Within the Preferences window, select "Network Settings" under the Network category in the left menu. On this page, under *Proxy Server* change the *Hostname* to **127.0.0.1**, the *Port* to **9050**, and select "Socks5" for the *Type*.

This utilizes Tor as a proxy server for the client and allows onion address resolution. Keep in mind that this will only work on devices were Tor can be called as a service on port 9050. On Linux systems, the Tor service can be started in the same way as if it were a server, by using systemctl. To do so, type the command below.

service tor start

[](https://img.wonderhowto.com/img/original/98/18/63644306059112/0/636443060591129818.jpg)

Once the network configuration is updated, hit "OK." HexChat should attempt to reconnect to the onion address provided. This time, it should work.

[](https://img.wonderhowto.com/img/original/01/55/63644306460738/0/636443064607380155.jpg)

If the *Welcome* message appears, you've successfully created a Tor-connected IRC server!

You may notice that after the onion address is looked up, the IP provided is the localhost address on the server itself, 127.0.0.1. This is because Tor routes connecting clients to the server and does not at any point disclose the server's public IP, so clients connecting to it see the server almost as if it were on their local network, rather than at a secret IP hidden by an onion address.

Now that your server is running, you can customize it like any other IRC server and begin inviting more users to interact with its channels.