**INSTALLING
an IRC Server
(Internet Relay Chat)
on your WRT54G,GS,GL**
Version 1.02
April 2nd, 2014
Rusty Haddock/AE5AE

NEW AND IMPROVED!

- WHAT THIS DOCUMENT IS.

This document will attempt to describe how to install an IRC server on a WRT54G, GS, or GL Broadband-HamnetTM node. This server program is called   *ngircd* -- <u>N</u>ext <u>Ge</u>neration
<u>IRC</u>
<u>D</u>
aemon (or service). This document is written to get you up and running. Unlike the previous versions, no knowledge of shell access or the 'vi' editor is needed, because a minimal, yet unique, configuration has been created for you.

- WHAT THIS DOCUMENT IS NOT.

This document will NOT even START to tell you how to administer or control said server program because there are plenty of help files and tutorials on the Internet.
    Hint: Google is your friend.
    Second hint: http://www.irchelp.org
    Third hint: http://ngircd.barton.de

- WHAT IS AN IRC SERVER?

An Internet Relay Chat (IRC) Server is a program that facilitates real-time, keyboard-to-keyboard chat sessions, not only between two people, but between many people simultaneously. Conversations are held in channels (which are synonymous with 'rooms') and new

channels can (usually) be created as needed by the users with the /join command. The users make use of an IRC client program to connect to an IRC server and access the channels available.

- WHY WOULD I WANT TO USE AN IRC SERVER?

An IRC server will permit simple text messaging between any users of a mesh with an IRC client program. A multitude of client software packages are available for just about any computer system whether it be running a Windows, Unix/Linux, or Mac operating system. These servers can be configured to connect to IRC servers off mesh as well. Private messages can also be sent from station to station or messages can be exchanged in a chat room or *channel* with many people connected simultaneously.

Having a chat server running on a mesh node precludes the use of an external server (say "off mesh", on the Internet) as well as needing another PC to hang off a mesh node to act as a server. That's one less piece (or pieces) of hardware to lug around to an event, a gathering, or a club meeting.

Without IRC software the usual communications between hams on an Broadband-Hamnet mesh network entailed using a cellphone, an HT, or yelling across a field. Voice over IP is a nice solution too but usually requires more hardware like a digital telephone and some sort of PBX or SIP server on the mesh.

Another use for IRC I have found is to make a server available during a club meeting. Many attendees can connect to a channel and post to a channel relevant text that's been mentioned during the meeting. Now everyone on the channel has a unambiguous copy of that URL or other information that was spoken or displayed (e.g. street address, telephone number, directions to the local bakery for donuts, etc). Those attending without a computer or smartphone can be emailed the information with a simple cut and paste. A log could be kept of the channel conversations and posted on a website or emailed at a later time as well.

  -   WHAT CLIENT SOFTWARE DO I NEED TO USE AN IRC SERVER?

Client software should be installed on your PC or laptop, not your mesh node. There is a plethora of client programs that will talk to an IRC server. For example, on Linux there is *XChat*, *mIRCii* while on Windows there are programs like
mIRC
,
XiRCON
, and a Windows port of
XChat
. Also keep in mind that most of the Instant Messenger (IM) type of programs that support multiple IM protocols will generally support IRC communications as well. Some of these programs would include the likes of the multiplatform program called
Pidgin
and Windows-specific
Trillian
. A good list of client software is available here:

http://www.ircreviews.org/clients

When the IRC service is advertised around a mesh, the "irc:" protocol can be used to provide a URL for the IRC server. With this, most browsers can start up an appropriate plugin or external program. The Opera browser (at least version 10+) has an IRC client built in! This is quite handy. Firefox and Mozilla permit plugins/addons to be used, such as ChatZilla. Even Internet Explorer can be told to start an external program to use with a URL starting with "irc:".

PLEASE NOTE: Due to the problems ChatZilla has with interpreting <u>proper</u>
irc: URLs, ones without a fully qualified domain name (FQDN), I am no longer suggesting it as an IRC client. I strong suggest that most any of the other IRC clients or plugins for your particular browser by used.

If you've never used IRC before it would be best to get together with a friend that has. Your friend can show you some of the subtleties of using an IRC server as well as any particular IRC client program because this document will not.

    -   WHAT DO I NEED TO INSTALL AN IRC SERVER ON A MESH NODE?

Nothing more than a connection from your mesh node to the Internet. The package that has been built is now available via the

Broadband-Hamnet™ repository. If you have version 18 on your system, it would be best to remove that package prior to installing the new one, version 20.2.

The new package contains the executable program for your WRT54G, an initial configuration   file, and a startup script so the server program can be run at boot time. This is a newer version of the ngIRCd was compiled from version 20.2 source code.

Unlike the previous version, this new installation package no longer requires the user to know how to use 'vi' or access the mesh node's shell or command line interpreter. This has been circumvented by a script added to the package that automatically edits the included config file so that

1. a unique IRC net hostname is setup and
2. the IRC operator (user ID: 'TheOper') the same password as the node's 'root'   account. This is the same password set when the Broadband-Hamnet™/HSMM- MESH™ firmware was first loaded.

Doing this allows the ngIRCd program to come up and run without a pre-defined IRC operator password (that everybody would know) and a non-conflicting IRC network name, the latter being necessary when networking multiple IRC servers together.

The ngIRCd configuration file in /etc/ngircd.conf can still be hand-edited to further suit the user's needs or wants. At least now the package can be used without the need for learning 'vi' or the shell.

NOTE: ngircd does NOT need one of the "bigger memory" WRT54G's to run. A unit with 16-Meg of RAM/4-Meg of FLASH will do just fine. I currently have it running on an old version 1.0 WRT54G in my garage.

   3. UPLOADING THE CONFIGURATION AND THE EXECUTABLE FILES.

   1. From the main status page, click [Setup], provide the necessary login and password, and then click on the Administration link.

   2. Under "Package Management" click on the [Refresh] button on the "Download Package" line.   The list of available packages will be downloaded to your node.
   3. Select ngircd 20.2 from the list of packages.
   4. Click the [Download] button.
   5. When the downloading and installation has completed expect the following:

      Installing ngircd (20.2) to root...
      Configuring ngircd

      ngircd is now enabled and running. It will also start on reboots.
      Don't forget to open port 6667 in the firewall!
      Done.

6. POPPING HOLES IN THE FIREWALL.

NOTE: I originally released this document prior to v1.0.0 of the BBHN firmware and unaware of the change in default mode to 5-host Direct (formerly DMZ). Should your node be currently using 1, 5, or 13-host Direct mode (or similar DMZ mode of previous releases) then you are done! There's no need to "pop" holes in the firewall. If your node is using NAT mode, regardless of the firmware version installed, then you should follow the instructions in this section.

On the Administration webpage click on the link for *Port Forwarding, DHCP, and Services*
. Under
Port Forwarding
select the following from the pulldowns:

```
            Interface:          WiFi
   Type:              Both
   IP Address:        localnode
      For the following entry boxes enter:
            Outside Port:       6667
   Lan Port:          6667
```

Now click on the ADD button and wait for an additional input form appears underneath the one you just filled in.

If you'd like to advertise this IRC server on the mesh enter the following data into the corresponding boxes:

```
            Name:            My IRC Server     [Whatever is informative]
      Link:           irc                [and click the checkbox]
      URL:            6667               [in the box after :]
```

mesh   [in the box after /]

Now click on the ADD button and wait for an additional input form appears underneath the one you just filled in.

Once this happens, and everything looks correct, click on the Save Changes
button near the top center of the page and wait for the following message to appear:

Configuration saved and is now active.

Now you should have popped the correct holes in the router's firewall for the IRC server.

The link created, irc://yournode:6667/mesh, points to the persistent IRC channel we created in the configuration file. This is link is intended for browsers that understand, or can be made to understand, the irc: protocol or scheme name. Some web browsers have an IRC chat client built-in (e.g. Opera), others may need a plugin, and others (e.g. Internet Explorer) may need to be instructed to execute an external program with the given URL to start an IRC client.

7. CONFIGURING YOUR IRC SERVER.

As stated before, nothing further *needs* to edited for an operational setup. If the config file,   /etc/ngir cd.conf   , is edited please remember that, after saving the file, the command

ngircd -t can tell you if there's anything wrong with your config file. If so, edit the file again and make the necessary corrections.

Once your config file is correct, you will need to restart the ngIRCd program with the command:

/etc/init.d/ngircd restart HINT: It wouldn't be a bad idea to save the   ngircd.conf file onto your PC in the event that you update or reflash the firmware in your mesh node and want to restore your IRC server settings.

8. A SINGLE SERVER OR MULTIPLE SERVERS ON A MESH?

This is a question you will eventually need to answer if you plan to use IRC on your mesh. Initially, a single server should do just fine until you can get more familiar with the configuration file and be able to configure the necessary information to connect with other IRC servers on your mesh.

OK, for those of you that can not wait (i.e. most of you) here are the steps needed to have one ngIRCd connect to another. These steps presume that the node you are currently working on (the local node) will establish a connection to another mesh node's ngIRCd server (the remote node). Keep in mind that the two servers do NOT need to connect to each other; only one connection is necessary to share IRC channels between the two nodes. Also note that the network of IRC servers needs to be acyclic -- no loops with the connections.

    1.   Edit the ngIRCd configuration file on the local node with the following shell command:

        vi /etc/ngircd.conf

    2. Find the [Server] section. For each server that your node will connect to   OR be connected to from another node you will create a new section.   In this section edit the following settings by removing any leading semicolon (;) and replacing the default values accordingly: NameThis should EXACTLY match the Name setting in the remote   node's   [Gl obal]

section.HostThis is the fully qualified host name or IP address of the remote node.

NOTE:

I've had problems with names and have stuck with IP addresses.PortSimply uncomment this setting and leave the value at 6667   unless, of course, the remote node is configured to connect to the local node on a different port number. MyPasswordThis should EXACTLY match the PeerPassword setting in remote   node's [Server]

section for your node.PassiveUncomment and set the value to 'no'. Setting to 'yes' restricts the local node to only accept connections from remote nodes, not initiating them. Write the configuration file and test with:

        ngircd -t

    3. On the remote node the ngIRCd configuration file will need to be editted in a similar manner as above   EXCEPT leave the Port setting commented out and change the Passive setting to 'yes'. This lets the remote node only accept the connection, not create it. The local host will do just fine creating the connection.

Write the configuration file on the remote node and test with:

      ngircd -t

   4. Here's the tricky part: restart the remote node's IRC server FIRST! This is done with

      /etc/init.d/ngircd restart

   5. On the local node, restart the IRC server but with a twist:

      /etc/init.d/ngircd restart; logread -f   The logread command will let you watch the system log as the local node's IRC server starts up. The output will tell you if the local IRC server begins talking to the remote IRC server. Simply press Ctrl-C to stop the logread program and return to the shell.

If everything goes well, and all the configurations line up, your two IRC servers are now connected and sharing channels.

   6. FINALLY
      Start up your IRC client and have fun!