

JamNNTPd 1.2.2

Modified by Scott Street (sstreet@ssalpha.dyndns.org)

Originally by Johan Billing (billing@df.lth.se)

2012-06-27

1. Introduction

=====

JamNNTPd is an attempt to merge dying fidonet technology with modern Usenet newsreaders. Basically, JamNNTPd is NNTP server that allows newsreaders to access a JAM, .MSG, and Squish message bases. (If you didn't understand a word of this, you probably don't want to use JamNNTPd anyway).

JamNNTPd can be used with both Linux and Windows. An executable binary is only supplied for Windows, Linux users need to compile JamNNTPd themselves.

2. Copyright

=====

The copyright of JamNNTPd belongs to Johan Billing. Permission to use, copy and distribute JamNNTPd is granted provided that this copyright notice is included. Permission to modify JamNNTPd is granted. Distributing modified versions of JamNNTPd is allowed provided that the documentation clearly states that it is a modified version. Parts of JamNNTPd may be freely used in other projects as long as credit is given.

3. Security

=====

I cannot guarantee that there are no security leaks in JamNNTPd. If you decide to use it, you do it on your own risk. If you use JamNNTPd under Linux, you should avoid running it with root privileges.

4. Using JamNNTPd

=====

4.1 Configuration options

The behaviour of JamNNTPd can be configured using the configuration options specified below. These can be given to JamNNTPd in two ways:

1) As commandline arguments

2) In a configuration file (see -config command). The preceding dash (-) is optional when an option is given in a file, it will be added if missing.

A configuration file with the default settings can be created with the -create option.

If JamNNTPd is run without any command line arguments at all, it will attempt

to read options from a file called "jamnnTPd.config" if present. Under Linux, JamNNTPd will look for this file in the "/etc" directory and under Windows in the current directory.

It is not necessary to specify any configuration options at all unless when fine-tuning JamNNTPd, the defaults have been designed to be sensible.

General options:

`-port <port> or -p <port>`

Set the port where JamNNTPd listens for connections. The default is 5000.

`-max <maxconn> or -m <maxconn>`

The maximum allowed number of connections at one time. The default is 5.

`-groups <groupsfile> or -g <groupsfile>`

`-allow <allowfile> or -a <allowfile>`

`-users <usersfile> or -u <usersfile>`

`-xlat <xlatfile> or -x <xlatfile>`

Use these to override the default locations of the config files.

`-logfile <logfile> or -l <logfile>`

Use this to override the default location of the log file.

`-noecholog`

Disables echoing of log messages to the console window.

`-debug`

If this option is used, JamNNTPd will print all sent and received text to the console window. Useful for testing.

Options for displaying messages:

`-readorigin`

Get address from the Origin line instead of the OADDRESS field of the JAM message header. This option makes JamNNTPd slower, but may be useful if your tosser does not set the OADDRESS field.

`-noencode`

By default, JamNNTPd will MIME-encodes headers with non-ascii characters. Using this option will send the headers as plain 8-bit text.

-strictnetmail

This option makes JamNNTPd use strict article counters for netmail messages. Normally JamNNTPd uses article counters that include all messages, not only those that the user is allowed to read. That behaviour is much faster, but may cause your newsreader to indicate the presence of new messages in the netmail area even when there are only messages for other users. Using this option will make JamNNTPd slower, but speed seems to be acceptable for small netmail areas with up to 1000 messages. Users will never be allowed to actually read netmail messages of other users even when this option is not used.

-def_flowed on/off

-def_showto on/off

Sets the default of the flowed and showto settings (if no default is specified on the command line, both will be on by default)

flowed: If flowed is on, JamNNTPd will use format=flowed (section 6.5), otherwise it will wrap long lines to a fixed width.

showto: Since there is no receiver for news messages, JamNNTPd can display the receiver name as a part of the sender name. With this option, this behaviour can be turned on or off.

These can be set globally and per user with parameters, see Section 4.4.

Options for posting messages:

-nostripre

JamNNTPd normally strips "Re:" from subject lines of followups. Use this option if you want to retain the "Re:".

-notearline

JamNNTPd normally puts the information from the X-Newsreader or User-Agent header field in the tearline of posted messages. This option disables this behaviour and leaves the tearline blank.

-noreplyaddr

JamNNTPd normally adds a REPLYADDR kludge with the e-mail address of the sender in posted messages. Use this option if you don't want REPLYADDR kludges. See also see section 6.4 below.

-notzutc

JamNNTPd normally writes the timezone into a TZUTC kludge when a message is posted. You can use this option if you don't want to create TZUTC kludges.

`-nocancel`

This option disallows the cancelling (deleting) of messages by the users. If allowed, users can only cancel messages from one of their "realnames" and only if the message has not yet been sent.

`-smartquote`

The quoting style of most newsreaders is different from traditional fidonet software. If you enable this option, JamNNTPd will try to change any quoted lines to fidonet style. This means that it will try to insert the initials of the person you reply to before the '>' character and also that it will try to compound multiple generations of quotes, i.e. "AA> BB>" will be changed into "BB>>".

Reformatting quotes in this way means that the user who posts a message will no longer have final say over the final content of the message since it will be changed after he or she sends it to JamNNTPd. Since this in principle is a bad thing even if quoted text will look a lot better after reformatting, this option is turned off by default.

`-origin <origin>`

Normally JamNNTPd uses the text found in the Organization header line as the Origin line text in posted messages. You can use this switch to override the Organization line and set your own origin for all posted messages. This option is recommended to avoid users placing nasty things in your origin line.

`-guestsuffix <suffix>`

If desired, JamNNTPd can add a suffix to posts from unauthenticated users. To activate that feature, specify the suffix here.

Example: `-guestsuffix " [GUEST]"`.

`-echomailjam <echomail.jam>`

If you specify a filename here, JamNNTPd will write a line to this file with the messagebase and message number for each message that is posted. The file follows the ECHOMAIL.JAM format supported by some tossers.

`-certpemfile <filename>`

This option both enables secure socket communications (TLS) and defines the location of the server certificate in PEM format. You can use a self-sign certificate or a commercially signed one.

Options for configuration files:

`-config <configfile>`

Read options from the specified configuration file.

-create <configfile>

Create a configuration file with the default settings.

4.2 Access rights

Access rights in JamNNTPd is based on access groups. Every newsgroup in JamNNTPd belongs to an access group. Access groups are named using one letter, typically A to Z (access groups are case-insensitive). In the configuration files, you can use "*" for "all groups" and "-" for "no groups".

When a user connects to the server, he/she gets access to two set of access groups. The first set of groups are for read access and the second set of groups are for post access. Users are only allowed to read newsgroups that belong to an access group they have read access to and are only allowed to post to newsgroups belong to an access group that they have post access to. The default access groups for users are configured in the "allow" file.

A user might get access to additional access groups if he/she logs in to the server using the AUTHINFO command. The groups associated with a user are defined in the "users" file.

4.3 Configuration files

JamNNTPd uses four configuration files:

- 1) In the "groups" file, the JAM areas that JamNNTPd should provide as newsgroups are configured.
- 2) In the "allow" file, the IP numbers that are allowed to use the server are listed. Here you also set the default access rights for users before they log in.
- 3) In the "users" file, you can list users that should be given access to additional groups if they log in.
- 4) In the "xlat" file, you can configure translation between the different character sets used in your message base and your newsreader.

The format of these files can be seen in the example configuration files. You do not need to restart JamNNTPd if you change them since they are read every time a new connection to the server is made.

4.4 Logging in with parameters

Since it is likely that all users will not prefer the same settings for

the flowed and showto options (see section 4.1), these can be modified by the user by logging in with parameters.

If you want to set these options, login using a login name of this format:

```
username/option1=on,option2=off
```

If you want to set options without logging in, just omit the user name and enter anything as your password.

Examples:

```
billing/showto=off
billing/flowed=on,showto=on
/flowed=off
```

5. Compilation

=====

JamNNTPd should compile with most compilers. I use gcc under Linux and MinGW under Windows. To compile JamNNTPd, go to the src directory and type either "make linux" or "make win32" depending on what platform you are compiling JamNNTPd on. After a successful compilation, you will find a file called "jamnntpd" or "jamnntpd.exe" in the src directory.

6. Compatibility

=====

6.1 The NNTP protocol

JamNNTPd supports most of the basic NNTP protocol as specified in RFC-977. The commands IHAVE, NEWGROUPS and NEWNEWS are not implemented, but at least give valid response codes if a newsreader tries to use them. JamNNTPd also supports the XOVER and AUTHINFO commands as specified in RFC-2980. XOVER never sends information about the line counts and byte counts of messages.

6.2 Format of news messages

JamNNTPd probably breaks the RFC-1036 specification on some minor points, but seems to work well enough with most newsreaders.

MIME is supported. Headers with non-ascii characters are encoded using quoted-printable or base64 unless disabled with -noencode. Message bodies are always sent as 7bit or 8bit. The charset is always set to "us-ascii" if a message does not contain non-ascii characters.

Posted messages can either be in plain text (8bit or 7bit) or encoded with quoted-printable or base64. Posted messages are only accepted if they are in the format text/plain (i. e. HTML and multipart messages will be rejected). Crossposted messages will be rejected. Messages longer than 32k bytes will also be rejected.

6.3 MSGID / Message-ID

JamNNTPd does not present the MSGIDs found in the JAM messagebase as Message-IDs to the newsreader, but rather uses its own dummy Message-IDs instead. The references line in followups will be replaced by the proper REPLY line.

6.4 REPLYADDR

When a message is posted to a JAM messagebase, JamNNTPd converts the original from address to a REPLYADDR kludge. According to FSC-0035, REPLYADDR kludges should be accompanied by a REPLYTO line. JamNNTPd does not create a REPLYTO line, but this should not be a major problem. If you want to disable the REPLYADDR kludge altogether, use the -noreplyaddr option.

6.5 format=flowed

A recent extension to MIME (RFC-2646) specifies a format for "flowed" text, i. e. text that only has line breaks between paragraphs and not after every line. This format is more well-suited for fidonet messages than the traditional format since fidonet has always used "flowed" text.

JamNNTPd uses format=flowed unless disabled with the -def_flowed switch or with login parameters, and it is preferable to use a newsreader that also supports this format. Unfortunately, only few newsreaders support this format today. JamNNTPd will also work with other newsreaders, but messages will look slightly worse both on the NNTP and fidonet side.

6.6 Character set translation

JamNNTPd has good support for character sets. The character set translation is configured in the "xlat" file and uses CHS files in the GoldED+ format for the actual translation. Extended CHS files with 256 character translations are supported and a character may be translated to up to four characters.

6.7 Netmail

JamNNTPd now also supports netmail. In netmail areas, users can only read messages to or from one of the names configured in the jamnntpd.users file. Replies to netmails are handled transparently and the name and address of the recipient are taken from the original message. When a user wants to write a new netmail, the name and address of the recipient are specified on the first line of the new message using this format:

To: name,address

Example:

To: Johan Billing, 2:15/87

A To: line can also be used to specify an alternative recipient in both netmail and echomail areas.

6.8 Tested newsreaders

JamNNTPd has been found to work with the following newsreaders:

Mozilla 1.4.1
Outlook Express 6.00.2800.1106
KNode 0.7.2
Xnews 5.04.25
40tude Dialog 2.0.7.1
Forte Free Agent 1.93/32.576
Lynx 2.8.5

Of these, only Mozilla seems to support format=flowed.

7. How to create additional *.chs files

=====

The *.chs files in the "xlat" and "unicode/xlat" directories are character set translation tables in the GoldED+ format. The files were created from mappings files found at this URL:

<http://www.unicode.org/Public/MAPPINGS/>

Fallback sequences for characters that don't exist in the target charset were taken from Markus Kuhn's transliteration tables found at the URL below:

<http://www.cl.cam.ac.uk/~mgk25/download/transtab.tar.gz>

If you want to create additional translation tables, you can easily do so with the supplied utility "makechs".

Syntax for makechs:

makechs <fromchrs> <destchrs> <frommap> [<destmap>]

<fromchrs> is the fidonet name of the charset you want to convert from.

<destchrs> is the fidonet name of the charset you want to convert to.

<frommap> is the Unicode mappings file for the source charset.

<destmap> is the Unicode mappings file for the destination charset. If you don't supply a mappings file, makechs will instead create a chs file that converts the source charset to utf-8.

The output of makechs is written to the console and needs to be redirected to the desired file.

Examples:


```
makechs IBMPC LATIN-1 map/cp437.txt map/8859-1.txt >437_iso.chs  
makechs IBMPC UTF-8 map/cp437.txt >437_utf.chs
```

Mappings files for all imaginable character sets can be found at the Unicode site above.