

**NAME**

**newsyslog** - maintain system log files to manageable sizes

**SYNOPSIS**

**newsyslog** [-CFNPnrsv] [-a *directory*] [-d *directory*] [-f *config\_file*] [-S *pidfile*] [-t *timefmt*]  
[[-R *tagname*] *file* ...]

**DESCRIPTION**

The **newsyslog** utility should be scheduled to run periodically by cron(8). When it is executed it archives log files if necessary. If a log file is determined to require archiving, **newsyslog** rearranges the files so that "*logfile*" is empty, "*logfile.0*" has the last period's logs in it, "*logfile.1*" has the next to last period's logs in it, and so on, up to a user-specified number of archived logs. It is also possible to let archived log filenames be created using the time the log file was archived instead of the sequential number using the -t option. Optionally the archived logs can be compressed to save space.

A log can be archived for three reasons:

1. It is larger than the configured size (in kilobytes).
2. A configured number of hours have elapsed since the log was last archived.
3. This is the specific configured hour for rotation of the log.

The granularity of **newsyslog** is dependent on how often it is scheduled to run by cron(8). Since the program is quite fast, it may be scheduled to run every hour without any ill effects, and mode three (above) assumes that this is so.

**OPTIONS**

The following options can be used with **newsyslog**:

**-f** *config\_file*

Instruct **newsyslog** to use *config\_file* instead of */etc/newsyslog.conf* for its configuration file.

**-a** *directory*

Specify a *directory* into which archived log files will be written. If a relative path is given, it is appended to the path of each log file and the resulting path is used as the directory into which the archived log for that log file will be written. If an absolute path is given, all archived logs are written into the given *directory*. If any component of the path *directory* does not exist, it will be created when **newsyslog** is run.

**-d** *directory*

Specify a *directory* which all log files will be relative to. To allow archiving of logs outside the root, the *directory* passed to the **-a** option is unaffected.

**-v** Place **newsyslog** in verbose mode. In this mode it will print out each log and its reasons for either trimming that log or skipping it.

**-n** Cause **newsyslog** not to trim the logs, but to print out what it would do if this option were not specified. This option implies the **-r** option.

**-r** Remove the restriction that **newsyslog** must be running as root. Of course, **newsyslog** will not be able to send a HUP signal to syslogd(8) so this option should only be used in debugging.

**-s** Specify that **newsyslog** should not send any signals to any daemon processes that it would normally signal when rotating a log file. For any log file which is rotated, this option will usually also mean the rotated log file will not be compressed if there is a daemon which would have been signalled without this option. However, this option is most likely to be useful when specified with the **-R** option, and in that case the compression will be done.

**-t** *timefmt*

If specified **newsyslog** will create the "rotated" logfiles using the specified time format instead of the default sequential filenames. The filename used will be kept until it is deleted. The time format is described in the strftime(3) manual page. If the *timefmt* argument is set to an empty string or the string "DEFAULT", the default built in time format is used. If the *timefmt* string is changed the old files created using the previous time format will not be automatically removed (unless the new format is very similar to the old format). This is also the case when changing from sequential filenames to time based file names, and the other way around. The time format should contain at least year, month, day, and hour to make sure rotating of old logfiles can select the correct logfiles.

**-C** If specified once, then **newsyslog** will create any log files which do not exist, and which have the **C** flag specified in their config file entry. If specified multiple times, then **newsyslog** will create all log files which do not already exist. If log files are given on the command-line, then the **-C** or **-CC** will only apply to those specific log files.

**-F** Force **newsyslog** to trim the logs, even if the trim conditions have not been met. This option is useful for diagnosing system problems by providing you with fresh logs that contain only the problems.

**-N** Do not perform any rotations. This option is intended to be used with the **-C** or **-CC** options

when creating log files is the only objective.

**-P** Prevent further action if we should send signal but the "pidfile" is empty or does not exist.

**-R** *tagname*

Specify that **newsyslog** should rotate a given list of files, even if trim conditions are not met for those files. The *tagname* is only used in the messages written to the log files which are rotated. This differs from the **-F** option in that one or more log files must also be specified, so that **newsyslog** will only operate on those specific files. This option is mainly intended for the daemons or programs which write some log files, and want to trigger a rotate based on their own criteria. With this option they can execute **newsyslog** to trigger the rotate when they want it to happen, and still give the system administrator a way to specify the rules of rotation (such as how many backup copies are kept, and what kind of compression is done). When a daemon does execute **newsyslog** with the **-R** option, it should make sure all of the log files are closed before calling **newsyslog**, and then it should re-open the files after **newsyslog** returns. Usually the calling process will also want to specify the **-s** option, so **newsyslog** will not send a signal to the very process which called it to force the rotate. Skipping the signal step will also mean that **newsyslog** will return faster, since **newsyslog** normally waits a few seconds after any signal that is sent.

**-S** *pidfile*

Use *pidfile* as syslogd(8)'s pidfile.

If additional command line arguments are given, **newsyslog** will only examine log files that match those arguments; otherwise, it will examine all files listed in the configuration file.

## FILES

*/etc/newsyslog.conf*

**newsyslog** configuration file

*/etc/newsyslog.conf.d*

By default each file in this directory ending in '.conf' and not beginning with a '.' will be included by the default *newsyslog.conf*.

*/usr/local/etc/newsyslog.conf.d*

By default each file in this directory ending in '.conf' and not beginning with a '.' will be included by the default *newsyslog.conf*.

## COMPATIBILITY

Previous versions of the **newsyslog** utility used the dot (".") character to distinguish the group name. Beginning with FreeBSD 3.3, this has been changed to a colon (":") character so that user and group names may contain the dot character. The dot (".") character is still accepted for backwards compatibility.

## SEE ALSO

bzip2(1), gzip(1), xz(1), zstd(1), syslog(3), newsyslog.conf(5), chown(8), syslogd(8)

## **HISTORY**

The **newsyslog** utility originated from NetBSD and first appeared in FreeBSD 2.2.

## **AUTHORS**

Theodore Ts'o, MIT Project Athena

Copyright 1987, Massachusetts Institute of Technology