NAME

cron - daemon to execute scheduled commands (Vixie Cron)

SYNOPSIS

cron [**-j** *jitter*] [**-J** *rootjitter*] [**-m** *mailto*] [**-n**] [**-s**] [**-o**] [**-x** *debugflag*[,...]]

DESCRIPTION

The **cron** utility should be started from */etc/rc* or */etc/rc.local*. It will return immediately, so you do not need to start it with '&'.

The **cron** utility searches */var/cron/tabs* for crontab files which are named after accounts in */etc/passwd*; crontabs found are loaded into memory. The **cron** utility also searches for */etc/crontab* and files in */etc/cron.d* and */usr/local/etc/cron.d* which are in a different format (see crontab(5)).

The **cron** utility then wakes up every minute, examining all stored crontabs, checking each command to see if it should be run in the current minute. Before running a command from a per-account crontab file, **cron** checks the status of the account with pam(3) and skips the command if the account is unavailable, e.g., locked out or expired. Commands from */etc/crontab* bypass this check. When executing commands, any output is mailed to the owner of the crontab (or to the user named in the MAILTO environment variable in the crontab, if such exists). The from address of this mail may be set with the MAILFROM environment variable.

Additionally, **cron** checks each minute to see if its spool directory's modification time (or the modification time on */etc/crontab*) has changed, and if it has, **cron** will then examine the modification time on all crontabs and reload those which have changed. Thus **cron** need not be restarted whenever a crontab file is modified. Note that the crontab(1) command updates the modification time of the spool directory whenever it changes a crontab.

Available options:

-j *jitter*

Enable time jitter. Prior to executing commands, **cron** will sleep a random number of seconds in the range from 0 to *jitter*. This will not affect superuser jobs (see **-J**). A value for *jitter* must be between 0 and 60 inclusive. Default is 0, which effectively disables time jitter.

This option can help to smooth down system load spikes during moments when a lot of jobs are likely to start at once, e.g., at the beginning of the first minute of each hour.

-J *rootjitter*

Enable time jitter for superuser jobs. The same as **-j** except that it will affect jobs run by the

superuser only.

-m *mailto*
Overrides the default recipient for **cron** mail. Each crontab(5) without MAILTO explicitly set will send mail to the *mailto* mailbox. Sending mail will be disabled by default if *mailto* set to a null string, usually specified in a shell as '' or "".

-n Do not daemonize; run in foreground instead.

-s Enable special handling of situations when the GMT offset of the local timezone changes, such as the switches between the standard time and daylight saving time.

The jobs run during the GMT offset changes time as intuitively expected. If a job falls into a time interval that disappears (for example, during the switch from standard time) to daylight saving time or is duplicated (for example, during the reverse switch), then it is handled in one of two ways:

The first case is for the jobs that run every at hour of a time interval overlapping with the disappearing or duplicated interval. In other words, if the job had run within one hour before the GMT offset change (and cron was not restarted nor the crontab(5) changed after that) or would run after the change at the next hour. They work as always, skip the skipped time or run in the added time as usual.

The second case is for the jobs that run less frequently. They are executed exactly once, they are not skipped nor executed twice (unless cron is restarted or the user's crontab(5) is changed during such a time interval). If an interval disappears due to the GMT offset change, such jobs are executed at the same absolute point of time as they would be in the old time zone. For example, if exactly one hour disappears, this point would be during the next hour at the first minute that is specified for them in crontab(5).

-o Disable the special handling of situations when the GMT offset of the local timezone changes, to be compatible with the old (default) behavior. If both options **-o** and **-s** are specified, the option specified last wins.

-x *debugflag*[,...]
Enable writing of debugging information to standard output. One or more of the following comma separated *debugflag* identifiers must be specified:

bit currently not used
ext make the other debug flags more verbose

load be verbose when loading crontab files
misc
be verbose about miscellaneous one-off events
pars be verbose about parsing individual crontab lines
proc be verbose about the state of the process, including all of its offspring
sch be verbose when iterating through the scheduling algorithms
test trace through the execution, but do not perform any actions

FILES

/etc/crontab	System crontab file
/etc/cron.d	Directory for optional/modularized system crontab files.
/etc/pam.d/cron	pam.conf(5) configuration file for **cron**
/usr/local/etc/cron.d	Directory for third-party package provided crontab files.
/var/cron/tabs	Directory for personal crontab files

SEE ALSO

crontab(1), pam(3), crontab(5), pam.conf(5), periodic(8)

AUTHORS

Paul Vixie <*paul@vix.com*>