Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'cron.8' command

$ man cron.8

CRON(8) System Administration CRON(8)

NAME

crond - daemon to execute scheduled commands

SYNOPSIS

crond [-c | -h | -i | -n | -p | -P | -s | -m<mailcommand>]

crond -x [ext,sch,proc,pars,load,misc,test,bit]

crond -V

DESCRIPTION

Cron is started from /etc/rc.d/init.d or /etc/init.d when classical sysvinit scripts are used. In case systemd is enabled, then unit file is installed into /lib/systemd/system/crond.service and daemon is started by systemctl start crond.service command. It returns immedi? ately, thus, there is no need to need to start it with the '&' parame? ter.

Cron searches /var/spool/cron for crontab files which are named after accounts in /etc/passwd; The found crontabs are loaded into the memory.

Cron also searches for /etc/anacrontab and any files in the /etc/cron.d directory, which have a different format (see crontab(5)). Cron exam? ines all stored crontabs and checks each job to see if it needs to be run in the current minute. When executing commands, any output is mailed to the owner of the crontab (or to the user specified in the MAILTO environment variable in the crontab, if such exists). Any job output can also be sent to syslog by using the -s option.

There are two ways how changes in crontables are checked. The first method is checking the modtime of a file. The second method is using the inotify support. Using of inotify is logged in the /var/log/cron log after the daemon is started. The inotify support checks for changes in all crontables and accesses the hard disk only when a change is detected.

When using the modtime option, Cron checks its crontables' modtimes ev? ery minute to check for any changes and reloads the crontables which have changed. There is no need to restart Cron after some of the crontables were modified. The modtime option is also used when inotify can not be initialized.

Cron checks these files and directories:

/etc/crontab

system crontab. Nowadays the file is empty by default. Origi? nally it was usually used to run daily, weekly, monthly jobs. By default these jobs are now run through anacron which reads /etc/anacrontab configuration file. See anacrontab(5) for more details.

/etc/cron.d/

directory that contains system cronjobs stored for different users.

/var/spool/cron

directory that contains user crontables created by the crontab command.

Note that the crontab(1) command updates the modtime of the spool di? rectory whenever it changes a crontab.

Daylight Saving Time and other time changes

Local time changes of less than three hours, such as those caused by the Daylight Saving Time changes, are handled in a special way. This only applies to jobs that run at a specific time and jobs that run with a granularity greater than one hour. Jobs that run more frequently are scheduled normally.

If time was adjusted one hour forward, those jobs that would have run

in the interval that has been skipped will be run immediately. Con? versely, if time was adjusted backward, running the same job twice is avoided.

Time changes of more than 3 hours are considered to be corrections to the clock or the timezone, and the new time is used immediately.

It is possible to use different time zones for crontables. See crontab(5) for more information.

PAM Access Control

Cron supports access control with PAM if the system has PAM installed. For more information, see pam(8). A PAM configuration file for crond is installed in /etc/pam.d/crond. The daemon loads the PAM environment from the pam_env module. This can be overridden by defining specific settings in the appropriate crontab file.

OPTIONS

-h Prints a help message and exits.

-i Disables inotify support.

-m This option allows you to specify a shell command to use for sending Cron mail output instead of using sendmail(8) This com? mand must accept a fully formatted mail message (with headers) on standard input and send it as a mail message to the recipi? ents specified in the mail headers. Specifying the string off (i.e., crond -m off) will disable the sending of mail.

-n Tells the daemon to run in the foreground. This can be useful when starting it out of init. With this option is needed to change pam setting. /etc/pam.d/crond must not enable pam_loginuid.so module.

-p Allows Cron to accept any user set crontables.

-P Don't set PATH. PATH is instead inherited from the environment.

-c This option enables clustering support, as described below.

-s This option will direct Cron to send the job output to the sys? tem log using syslog(3). This is useful if your system does not have sendmail(8), installed or if mail is disabled.

-x This option allows you to set debug flags.

-V Print version and exit.

SIGNALS

When the SIGHUP is received, the Cron daemon will close and reopen its log file. This proves to be useful in scripts which rotate and age log files. Naturally, this is not relevant if Cron was built to use sys? log(3).

CLUSTERING SUPPORT

In this version of Cron it is possible to use a network-mounted shared /var/spool/cron across a cluster of hosts and specify that only one of the hosts should run the crontab jobs in this directory at any one time. This is done by starting Cron with the -c option, and have the /var/spool/cron/.cron.hostname file contain just one line, which repre? sents the hostname of whichever host in the cluster should run the jobs. If this file does not exist, or the hostname in it does not match that returned by gethostname(2), then all crontab files in this directory are ignored. This has no effect on cron jobs specified in the /etc/crontab file or on files in the /etc/cron.d directory. These files are always run and considered host-specific.

Rather than editing /var/spool/cron/.cron.hostname directly, use the -n option of crontab(1) to specify the host.

You should ensure that all hosts in a cluster, and the file server from which they mount the shared crontab directory, have closely synchro? nised clocks, e.g., using ntpd(8), otherwise the results will be very unpredictable.

Using cluster sharing automatically disables inotify support, because inotify cannot be relied on with network-mounted shared file systems.

CAVEATS

All crontab files have to be regular files or symlinks to regular files, they must not be executable or writable for anyone else but the owner. This requirement can be overridden by using the -p option on the crond command line. If inotify support is in use, changes in the symlinked crontabs are not automatically noticed by the cron daemon. The cron daemon must receive a SIGHUP signal to reload the crontabs.

This is a limitation of the inotify API.

The syslog output will be used instead of mail, when sendmail is not installed.

SEE ALSO

crontab(1), crontab(5), inotify(7), pam(8)

AUTHOR

Paul Vixie ?vixie@isc.org?

Marcela Ma?l??ov? ?mmaslano@redhat.com?

Colin Dean ?colin@colin-dean.org?

Tom?? Mr?z ?tmraz@fedoraproject.org?