



Periodic Processes

tsaimh (2022-2023, CC BY-SA)
wangth (2017-2021, CC BY-SA)
? (1996-2016)

國立陽明交通大學資工系資訊中心

Information Technology Center of Department of Computer Science, NYCU

Handbook and Manual pages

- Official guide and be found at
 - <https://www.freebsd.org/doc/en/books/handbook/configtuning-cron.html>
 - https://www.freebsd.org/doc/zh_TW/books/handbook/configtuning-cron.html

CRON – Schedule Commands (1)

- What we want?
 - Do things at right time automatically
- cron daemon
 - The daemon that handles periodic execution
 - cron daemon reads configuration file and executes commands on time

[cron\(8\)](#), [crontab\(1\)](#), [crontab\(5\)](#)

CRON – Schedule Commands (2)

- Configuration file

- So called: **crontab** (cron table)
- Location of user cron configuration file
 - Every user can have at most one crontab file and this file will be named the user's login ID
 - Edit using crontab(1) command

System	Cron Dir
FreeBSD	/var/cron/tabs
Red Hat	/var/spool/cron
Solaris	/var/spool/cron/crontabs
SunOS	/var/spool/cron/crontabs

- Location of System Cron Configuration file
 - /etc/crontab

CRON – Schedule Commands (3)

- Configuration File Format

- Ignored

- Blank lines or leading spaces and tabs

- Comments

- pound-sign lines whose first non-space character is a #

- environment setting

- name = value

- Default environment variables

- LOGNAME, SHELL, PATH, HOME, MAILTO

```
SHELL=/bin/sh
```

```
PATH=/etc:/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin
```

- cron command example:

```
# minute hour day month weekday command
33      7      *      *      *      /usr/local/bin/rsync -al -delete /home/ backup:/raid/home/
```



CRON – Schedule Commands (4)

- cron command format – *minute hour day month weekday command*

Field	Description	Range
minute	Minute of the hour	0 ~ 59
hour	Hour of the day	0 ~ 23
day	Day of the month	1 ~ 31
month	Month of the year	1 ~ 12
weekday	Day of the week	0 ~ 6 (0 = Sunday)

- Rule Matching
 - * matches everything
 - Single character matches exactly
 - Dash(-) matches range
 - Comma(,) matches any listed value
 - Slash(/) matches skips of the number's value through the range.

CRON – Schedule Commands (5)

- crontab time format example

45 10 * * 1-5 → AM 10:45, from Mon. to Fri.

10 * * * * → On the 10th minute of each hour

*/3 * * * * → Every three minutes periodic e.g., $1-59/2 = 1, 3, 5, 7, 9, \dots, 59$

30 15 5 * * → PM 3:30 of each 5th day

0 0 14 2 * → On the Midnight of Valentine's day

5 0-6 * * * → On the 5th minute, from 0 to 6 o'clock

0,30 * 13 * 5 → Every half-hour on Fri. and the 13th day

If both the day of month and day of week aren't *, then cron will run the command when either field matches the current time. So this command will actually run every Friday and every 13th day.

- crontab example

20	1	*	*	*	find /tmp -atime +3 -exec rm -f {} ';'
55	23	*	*	0-3,6	/home/tsaimh/cputemp-check.sh

CRON – Schedule Commands (6)

- Special strings to specify the time

string	meaning	in 5 fields format
@reboot	Run once, <u>at startup</u> .	N/A
@yearly	Run once a year	0 0 1 1 *
@annually	(same as @yearly)	
@monthly	Run once a month	0 0 1 * *
@weekly	Run once a week	0 0 * * 0
@daily	Run once a day	0 0 * * *
@midnight	(same as @daily)	
@hourly	Run once an hour	0 * * * *
@every_minute	Run once a minute	*/1 * * * *
@every_second	Run once a second	

crontab command

- [crontab\(1\)](#)

\$ crontab -e [-u user]

- Edit the [user's] crontab using editor

\$ crontab -l

- List the content of the crontab

\$ crontab -r

- Remove the current crontab

\$ crontab *filename*

- Install *filename* as your crontab

crontab management (1)

- To Allow or deny user from using cron daemon
 - By default, all users can have their own crontab
 - allow file
 - A list of users that may use crontab, any other not in the list can not use it
 - deny file
 - Reverse meaning

- log

System	Allow or deny file	Log file
FreeBSD	/var/cron/ {allow,deny}	By syslogd
Red Hat	/etc/cron. {allow,deny}	/var/log/cron
Solaris	/etc/cron.d/cron. {allow,deny}	/var/cron/log
SunOS	/var/spool/cron/cron. {allow,deny}	By syslogd

System crontab: /etc/crontab

- System crontab
 - /etc/crontab

```
SHELL=/bin/sh
PATH=/etc:/bin:/sbin:/usr/bin:/usr/sbin
HOME=/var/log
#minute hour mday month wday who 1 command
*/5 * * * * root /usr/libexec/atrun
*/11 * * * * operator /usr/libexec/save-entropy
0 * * * * root newsyslog
1 3 * * * root periodic daily 2
15 4 * * 6 root periodic weekly
30 5 1 * * root periodic monthly
1,31 0-5 * * * root adjkerntz -a
```

periodic utility (1)

- periodic(8)

- Run periodic system function under /etc/periodic

```
$ ls -ld /etc/periodic/*
drwxr-xr-x  2 root  wheel  1024 Sep 26 21:43 /etc/periodic/daily
drwxr-xr-x  2 root  wheel   512 Sep 27 03:49 /etc/periodic/monthly
drwxr-xr-x  2 root  wheel   512 Sep 27 03:49 /etc/periodic/security
drwxr-xr-x  2 root  wheel   512 Sep 27 03:49 /etc/periodic/weekly
```

```
$ ls /etc/periodic/daily
100.clean-disks      200.backup-passwd  405.status-ata-raid  430.status-rwho
110.clean-tmps       210.backup-aliases 406.status-gmirror   440.status-mailq
120.clean-preserve   300.calendar        407.status-graid3    450.status-security
130.clean-msgs       310.accounting      408.status-gstripe   470.status-named
140.clean-rwho       330.news            409.status-gconcat   500.queuerun
150.clean-hoststat   400.status-disks    420.status-network   999.local
```

- periodic utility

- /etc/periodic.conf
- /etc/defaults/periodic.conf

```
$ ls -al /usr/local/etc/periodic/security
total 18
drwxr-xr-x  2 root  wheel    4 Apr 12  2017 .
drwxr-xr-x  8 root  wheel    8 Aug 20  2016 ..
-r-xr-xr-x  1 root  wheel 4944 Apr  2  2017 410.pkg-audit
-r-xr-xr-x  1 root  wheel 1686 Apr  2  2017 460.pkg-checksum
```

periodic utility (2)

- For custom system programs: /usr/local/etc/periodic

```
$ ls -l /usr/local/etc/periodic/  
total 19  
drwxr-xr-x  2 root  wheel  6 Apr 12  2017 daily  
drwxr-xr-x  2 root  wheel  4 Aug 20  2016 hourly  
drwxr-xr-x  2 root  wheel  4 Aug 20  2016 monthly  
drwxr-xr-x  2 root  wheel  3 Aug 20  2016 reboot  
drwxr-xr-x  2 root  wheel  4 Apr 12  2017 security  
drwxr-xr-x  2 root  wheel  6 Apr 12  2017 weekly
```

```
$ ls -l /usr/local/etc/periodic/daily/  
total 18  
-r-xr-xr-x  1 root  wheel  1512 Jul 29  2016 402.zfSnap  
-r-xr-xr-x  1 root  wheel  1073 Jul 29  2016 403.zfSnap_delete  
-r-xr-xr-x  1 root  wheel  2746 Apr  2  2017 411.pkg-backup  
-r-xr-xr-x  1 root  wheel  2506 Apr  2  2017 490.status-pkg-changes
```

periodic utility (3)

- Execution order depends on filenames
 - Use number as prefix to control the order
- All scripts under that directory will be executed
 - Unlike /etc/rc.conf
 - Even though there is no "YES" in /etc/periodic.conf
- /etc/periodic.conf

```
$ cat /etc/periodic.conf
daily_clean_tmps_enable="YES"           # Delete stuff daily
daily_clean_tmps_dirs="/tmp /var/tmp"   # Delete under here
daily_clean_tmps_days="3"               # If not accessed for
daily_status_zfs_enable="YES"           # Check ZFS
daily_status_ntpd_enable="YES"          # Check NTP status
weekly_dehydrated_enable="YES"
...
```

- Driven by crontab(1)

at command

- [at\(1\)](#)
 - executes commands at a specified time
at [-q queue] [-f file] [-mldbv] time
or at [-q queue] [-f file] [-mldbv] -t [[CC]YY]MMDDhhmm[.SS]
- at management
 - atq: View job queue
 - atrm: Remove jobs
 - /var/at/at.{allow,deny}
 - By default, only root can execute "at" command
- Driven by crontab(1)
 - Invoked every 5 minutes