10.3.6 cron Facts

The **cron** daemon (crond) schedules tasks to run in the future on a regular basis.

This lesson covers the following topics:

- **cron** configuration files
- Syntax for **cron** jobs
- **cron** commands

cron Configuration Files

The following configuration files are used with **cron**:

File	Description
/etc/crontab	The /etc/crontab (cron table) file holds entries that direct commands to execute at a specific time. The /etc/crontab file: - Is used to schedule custom tasks that run system wide. - Can only be edited by the root user. **crond** runs tasks scheduled in the /etc/crontab file as the root user.
/etc/cron.*directory*	The **cron** daemon executes the scripts found in each of the following directories at the specified interval for the whole system: - /etc/cron.hourly - /etc/cron.daily - /etc/cron.weekly - /etc/cron.monthly
/var/spool/cron/*username*	If permitted, each user can create a personal crontab file located at /var/spool/cron/*username*.
/etc/cron.allow	The /etc/cron.allow file identifies users who are allowed to create their own cron jobs. If /etc/cron.allow file exists, then only users listed within it are allowed to create a crontab file in /var/spool/cron/*username.* All other users are denied, and the /etc/cron.deny file is ignored.
/etc/cron.deny	The /etc/cron.deny file identifies users who are not allowed to create cron jobs. If the /etc/cron.deny file exists, only the users listed within it are not allowed to edit /var/spool/cron/*username.* Everyone else is allowed. This file is only processed if the /etc/cron.allow file does not exist.

Syntax for cron Jobs

Each entry in the **/etc/crontab** or **/var/spool/cron/*username*** file uses a specific format. The table below illustrates the syntax for a typical cron job and provides additional examples. The asterisk (*) is a wildcard that is equal to any value.

Example	Minute	Hour	Day of Month	Month	Day of Week	Command	Description
00 5 * * 6 /bin/tar -cf /home /mnt/usb/homebak.tar	00	5	*	*	6	**/bin/tar -cf /home /mnt/usb/homebak.tar**	This schedule runs the **tar** utility on the sixth day of the week (Saturday) at the fifth hour (5:00 a.m.) and zero minutes. (Note that the days of the week are

							numbered 0 through 7, 0 and 7 being equal to Sunday.)
15 23 25 * * /bin/updatedb	15	23	25	*	*	**/bin/updatedb**	This schedule runs the **updatedb** command at 11:15 p.m. on the 25th of every month.
00 24 1 1,6 * /bin/who > /root/who.txt	00	24	1	1 and 6	*	**/bin/who > /root/who.txt**	This schedule runs the **who** command at midnight on the first days of January and June.

cron Commands

Use the following commands to manage **cron** task scheduling.

Command	Function	Examples
crontab	Manages the /var/spool/cron/*username* crontab file. Be aware of the following options: - **-e** edits the crontab file for the current user in vi. - **-l** displays the contents of the current user's crontab file. - **-r** removes the current user's crontab file. - **-u *username*** specifies a different user for the **-e**, **-l**, and **-r** options.	**crontab -e** edits the crontab of the current user. **crontab -eu username** edits the crontab file of the specified user. **crontab -l** lists the cron jobs for the current user. **crontab -lu *username*** lists the cron jobs for the specified user. **crontab -r -u username** removes the crontab file of the specified user. **crontab -r** removes the crontab file of the current user. **crontab /home/user/cronjobs** creates a crontab file using the cronjobs file for the current user.
crontab*file*	Loads a crontab job from a file. Write the file using the crontab syntax. This command overwrites the current crontab.	**crontab /home/user/cronjobs** creates a crontab file using the cronjobs file for the current user.

Be aware of the following details:

- Some distributions use separate files in the /etc/cron.d directory in addition to lines in the /etc/crontab file.
- The cron daemon (crond) is managed using its init script in the /etc/rc.d/init.d/ or /etc/init.d/ script directory on init-based distributions. For systemd-based distributions, it is managed using the crond.service file and the **systemctl** command.