Configuring event-triggered commands with Incron

TRIGGER HAPPY

While cron doggedly keeps to a fixed schedule, Incron monitors directories and runs commands when changes occur.

BY CHARLY KÜHNAST


limeyrunner, Fotolia

Cron is a constant companion for admins like myself. My crontab and I have both grown over the years – Crontab has gotten longer, and I've gotten wider. Maybe I'm sentimental, but I decided it was time to let my old friend venture out into unexplored territory with a little help from Incron [1]. This cron extension uses an event triggered approach, rather than traditional time-based scheduling, monitoring directories, and running commands when specific changes occur.

Before you can start working with Incron, you need kernel 2.6.13 with built-in Inotify support and the matching header file *inotify.h*. The file is typically located in */usr/include/sys/*; some distributions add a file called *inotify-syscalls.h*.

If you like, you can change the installation paths in the Makefile and then go on to *make && make install*. After completing the build and installing, you should have *incrond*, *incrontab*, and the manpages.

Path, Event, Command

Incrond is a daemon, and it disappears into the background after launching. Of course, Incrontab is the central element. Calls to Incrontab follow the Crontab syntax for the main: *icrontab -e* opens the table for editing, *icrontab -l* displays the content, and *icrontab -r* deletes the table. The *incrontab* format is very simple. Each line contains three entries:

```
path event command+parameter
```

Whenever an *event* occurs in a monitored *path*, Incron runs the matching *command*. Table 1 shows you the events Incron can monitor. When an event occurs, Incron sets the parameters listed in Table 2, and the *command* can then use them for its own purposes. If you modify the Incrontab, there is no need to let Incrond know, as the daemon parses the control file periodically.

Let's look at an example. Whenever a user deletes a file in the */var/run/daemon/* directory, I want Incrond to delete the */var/log/daemon.log* file, too. The *incrontab* line looks like this:

```
/var/run/daemon IN_DELETE rm ➥
/var/log/daemon.log
```

This simple entry illustrates the power of Incron. At last – cron has a friend it can play with. ■

Table 1: Monitorable Events

Event	Description
IN_ACCESS	Read access
IN_MODIFY	Write access
IN_ATTRIB	Metadaten (inode or attribut) changed
IN_CLOSE_WRITE	File opened for writing and then closed
IN_CLOSE_NOWRITE	File opened without writing and then closed
IN_CLOSE	*IN_CLOSE_WRITE* or *IN_CLOSE_NOWRITE*
IN_OPEN	File opened
IN_MOVED_FROM	File moved out of this directory
IN_MOVED_TO	File moved into this directory
IN_MOVED	*IN_MOVED_FROM* or *IN_MOVED_TO*
IN_DELETE	File deleted
IN_DELETE_SELF	The monitored directory has been deleted
IN_ALL_EVENTS	One of the above-mentioned events

Table 2: Command Parameters

Variable	Description
$@	provides the *path*
$#	Name of the file that triggered the *event*
$%	Shows the triggered *event*
$$	The dollar sign itself

INFO

[1] Incron: *http://incron.aiken.cz*

THE AUTHOR

Charly Kühnast is a Unix System Manager at the data center in Moers, near Germany's famous River Rhine. His tasks include ensuring firewall security and availability and taking care of the DMZ (demilitarized zone).