

خودکارسازی فرایندها در گنو/لینوکس

به عنوان یک مدیر سیستم، شما باید اقداماتی را دائما انجام دهید. هر بار که این اقدامات تکراری را انجام می‌دهید که ممکن است زمانبر باشند، زمان خود را برای انجام سایر امور از دست خواهید داد و ممکن است حتی زمانی برای انجام آنها نداشته باشید. برخی اوقات ممکن است نیاز داشته باشید تا برخی امور را زمانی که فایل سیستم دارای حداقل درگیری است انجام دهید. مانند صبح بسیار زود یا نیمه شب. راه حل این مشکلات خودکارسازی امور و نوشتن اسکریپت‌های پوسته برای آنهاست. شما می‌توانید به سیستم گنو/لینوکس خود بگویید هنگامی که شما در خواب هستید، کارهایتان را انجام دهد.

سه ابزار خودکارسازی که بطور معمول وجود دارند، می‌توانند هر برنامه، دستور یا اسکریپتی را اجرا نمایند. هر یک از این ابزارها دارای روش اجرای خاص خودش می‌باشد.

دستور at

دستور at می‌تواند برنامه یا دستور مورد نظر شما را در ساعت خاصی اجرا کند. این دستور تنها می‌تواند دستور داده شده را یکبار اجرا کند. زمان مشخص شده در دستور at می‌تواند هر زمانی در آینده باشد، از ساعت و دقیقه گرفته تا تعداد روز. استفاده از دستور at به دو شکل امکان پذیر است:

\$ at [-q letter][-f file][-m]v TIME

\$ at -c job [job...]

جدول ۱ گزینه‌های این دستور را شرح داده است.

گزینه	شرح
-m	پایان اجرای یک دستور را با ارسال یک نامه الکترونیکی به کاربر اعلام خواهد کرد. به طور عادی تنها هنگامی نامه الکترونیکی ارسال خواهد شد که خطایی رخ دهد یا دستور اجرا شده دارای خروجی باشد
-f file	دستورات را بجای خط فرمان از یک فایل خواهد خواند
q -letter	برنامه را در صف مشخص شده قرار خواهد داد. letter به معنی میزان اولویت دستور برای اجرا می‌باشد. بجای آن باید از حروف a تا z یا A تا Z استفاده کنید. دستوراتی که دارای letter بالاتری هستند، با اولویت پایین‌تری اجرا خواهند شد. صف a برای دستور at و صف b برای دستور batch پیش‌گزینه می‌باشند
-v	زمان اجرای دستور را چاپ خواهد کرد
-l	لیستی از تمامی دستورات در حال انتظار برای اجرا را که متعلق به کاربر جاری است، نمایش خواهد داد
-c	دستورات لیست شده در خط فرمان را به خروجی استاندارد که معمولا صفحه نمایش می‌باشد، ملحق می‌کند

جدول ۱ گزینه‌های دستور at

زمان بخشی اجباری در دستور at می‌باشد، به غیر از هنگامی که از گزینه -l استفاده می‌کنید. فرمت زمان وارد شده می‌تواند به صورت ۱۲ ساعته با فرمت hh:mm که پس از آن am یا pm قرار خواهد گرفت و یا با فرمت ۲۴ ساعته به صورت چهار رقمی مانند ۱۶۲۰ باشد. همچنین شما می‌توانید بجای استفاده از زمان‌های عددی، از کلمات کلیدی مانند noon, teatime, midnight یا now نیز استفاده کنید.

با مشخص کردن تاریخ می‌توانید محدوده اجرایی دستور at را باز هم گسترده‌تر نمایید. نام ماه را باید بصورت متنی و روز ماه را بصورت عددی مشخص کنید. این امکان وجود دارد که مشخص کنید دستور در چه روزی از هفته اجرا شود. همچنین امکان بکارگیری کلمات کلیدی today یا tomorrow نیز وجود دارد. در صورتی که تنها زمان مشخص شده باشد، دستور در اولین باری که زمان مشخص شده فرا رسد، اجرا خواهد شد.

امکان دیگر اضافه کردن زمان به صورت مثلا now +2 است. در این مثال، دستور ۲ ساعت بعد از زمان کنونی اجرا خواهد شد. کلمه next نیز به معنی +1 می‌باشد. برای اینکه کاملاً نحوه استفاده از دستور at را درک کنید، در زیر مثال‌هایی ارائه شده است:

\$ ls -l | at 20:34

\$ ls -l | at 4:20 pm November 12

\$ ls -l | at 16:25 Oct 16

\$ ls -l | at midnight next day

\$ ls -l | at midnight +1 day

\$ ls -l | at 2 am Monday

\$ ls -l | at now

پس از اینکه صف بندی دستورات را به اتمام رساندید، از گزینه -l برای لیست کردن آنها استفاده کنید. برای حذف یک دستور زمان‌بندی شده، از دستور atrm استفاده کنید.

دستور batch

نحوه عملکرد دستور batch بسیار شبیه به دستور at می‌باشد با این تفاوت که دستور شما بجای زمان خاص، هنگامی اجرا خواهد شد که میزان بار سیستم (System Load) به زیر 1.5 برسد. در زیر نحوه استفاده از این دستور را مشاهده می‌کنید:

\$ batch [-q letter] [-f file] [-mv] [TIME]

نحوه استفاده از گزینه TIME همانند دستور at بوده با این تفاوت که گزینه‌ای انتخابی است.

دستور cron

در سیستم‌هایی که دائما روشن می‌باشند، مانند سیستم‌های سرورس‌دهنده، وظایف خودکار باید با استفاده از نرم‌افزار cron اجرا شوند. cron هنگام راه‌اندازی سیستم اجرا شده و هر دقیقه بررسی می‌کند که چه دستوری را باید اجرا نماید. دستوراتی را که cron اجرا خواهد کرد، در فایل `/etc/crontab` تعریف می‌شوند.

دستوراتی که در `/etc/crontab` وجود دارند، عموما برای وظایف سیستمی استفاده می‌شوند. به طور پیش‌گزیده سه وظیفه در این فایل تعریف شده‌است که هر کدام از آنها اسکریپت‌های موجود در دایرکتوری‌های خاصی را که قرار است به طور روزانه، هفتگی یا ماهانه اجرا شوند، اجرا می‌کند. شما حتما مجبور نیستید دستورات یا اسکریپت‌های خود را برای اجرا شدن درون این دایرکتوری‌ها قرار دهید. امکان تعریف آنها بصورت جداگانه در فایل `/etc/crontab` وجود دارد. در زیر محتویات موجود در یک فایل `/etc/crontab` پیش‌گزیده را مشاهده می‌کنید:

#more /etc/crontab

```
/ #etc/crontab: system-wide crontab
```

```
#Unlike any other crontab you don't have to run the `crontab'
```

```
# command to install the new version when you edit this file.
```

```
#This file also has a username field, that none of the other crontabs do.
```

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

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

```
# m h dom mon dow user command
```

```
***17root run-parts --report /etc/cron.hourly
```

```
***6 25root test -x /usr/sbin/anacron || run-parts --report  
/etc/cron.daily
```

```
7 **6 47root test -x /usr/sbin/anacron || run-parts --report  
/etc/cron.weekly
```

```
**1 6 52root test -x /usr/sbin/anacron || run-parts --report  
/etc/cron.monthly
```

علامت ستاره (*) به معنای هر روز، هفته و ماه می‌باشد. پس از پنج فیلد نخست در هر خط که به ترتیب شامل دقیقه، ساعت، روز در ماه، ماه و روز در هفته می‌باشند،

نام کاربری که دستور توسط آن اجرا خواهد شد نوشته می‌شود (در مثال بالا کاربر ریشه می‌باشد). سپس دستور به همراه تمامی آرگومان‌ها و اطلاعات مورد نیازش برای اجرا نوشته می‌شود. هنگامی که زمان فعلی با زمان هیریک از ورودی‌های این فایل یکسان شوند، آن دستور اجرا خواهد شد. جدول ۲ نشان‌دهنده نحوه نگارش زمان در فایل `/etc/crontab` می‌باشد.

در صورتی که دقیقه یا ساعت به صورت یک ستاره (*) تعریف شده باشد، `cron` آن دستور را در هر دقیقه یا هر ساعت اجرا خواهد کرد. این ممکن است باعث `Over Load` شدن سیستم در اثر وجود حجم زیادی از پروسه‌ها شود. بنابراین توصیه می‌کنم که علامت ستاره را تنها در فیلدهای روز در ماه، ماه یا روز در هفته استفاده نمایید.

نام فیلد	مقادیر مجاز
m یا دقیقه	۰ تا ۵۹
h یا ساعت	۰ تا ۲۳
dom یا روز در ماه	۱ تا ۳۱
mon یا ماه	۱ تا ۱۲
dow یا روز در هفته	۰ تا ۷ که عدد صفر نشان‌دهنده یکشنبه می‌باشد

جدول ۲ نحوه نگارش زمان در فایل `crontab`

فایل‌هایی که در شاخه `/etc/cron.daily` قرار دارند، بطور روزانه اجرا خواهند شد. در زیر نمونه‌ای از محتویات این دایرکتوری را مشاهده می‌کنید :

```
#ls -l /etc/cron.daily/
```

```
total 52
```

```
-rwxr-xr-x 1 root root 311 Jul 30 16:02 0anacron
```

```
-rwxr-xr-x 1 root root 429 Feb 2 2004 acct
```

```
-rwxr-xr-x 1 root root 502 May 14 2004 bsdmainutils
```

```
-rwxr-xr-x 1 root root 782 Apr 26 2004 exim4-base
```

```
-rwxr-xr-x 1 root root 419 Oct 27 2003 find
```

```
-rwxr-xr-x 1 root root 89 Oct 9 2002 logrotate
```

```
-rwxr-xr-x 1 root root 946 May 13 2004 man-db
```

```
-rwxr-xr-x 1 root root 86 Aug 18 2002 modutils
-rwxr-xr-x 1 root root 1288 Sep 30 13:00 mysql-server
-rwxr-xr-x 1 root root 495 Nov 19 2001 netkit-inetd
-rwxr-xr-x 1 root root 345 Oct 1 17:45 quota
-rwxr-xr-x 1 root root 2571 Jan 20 2004 standard
-rwxr-xr-x 1 root root 1307 Jul 27 20:00 syslogd
```

cron تنها برای مدیران سیستم قابل استفاده نیست و کاربران عادی نیز می‌توانند از آن سود ببرند. هر کاربر می‌تواند فایل crontab خاص خود را با استفاده از دستور crontab filename ایجاد نماید. گزینه‌هایی که به همراه این دستور قابل اجرا هستند عبارتند از گزینه -ا- برای لیست کردن فایل‌های crontab مخصوص کاربر، گزینه -e- جهت ویرایش فایل‌های crontab مخصوص کاربر و گزینه -r- برای حذف فایل‌های crontab متعلق به کاربر. فرمت فایل‌های crontab مربوط به کاربران کاملاً مشابه با فایل /etc/crontab می‌باشد. برای اینکه کاربران قادر به استفاده از crontab باشند، باید در فایل /etc/cron.allow اضافه و یا از فایل /etc/cron.deny حذف شوند. به اینصورت می‌توانید تعیین کنید که کدامیک از کاربران حق استفاده از cron را خواهند داشت. در صورتی که هر یک از این فایل‌ها بر روی سیستم شما وجود ندارند، می‌توانید بطور دستی آنها را ایجاد کنید. ایجاد یکی از فایل‌های cron.allow یا cron.deny کافی خواهد بود. درون این فایل‌ها، هر یک از نام‌های کاربری را در یک خط جداگانه تایپ نمایید.

دستور anacron

در مواردی که کامپیوتر به صورت ۲۴ ساعته روشن و فعال نیست، cron پاسخگوی اجرای امور نخواهد بود، در حالی که ابزار anacron این وابستگی حتمی به روشن بودن ۲۴ ساعته کامپیوتر را ندارد. در صورتی که کامپیوتر در ساعتی که قرار است دستوری اجرا شود خاموش باشد، anacron حتماً آنرا به هر حال بعداً اجرا خواهد کرد.

همانند cron، ابزار anacron نیز دارای یک فایل پیکربندی است که دستوراتی را که باید اجرا کند را از درون آن می‌خواند. هر خط این فایل به یک دستور مجزا اختصاص دارد. در زیر یک فایل پیکربندی نمونه anacron که در مسیر /etc/anacrontab قرار دارد نمایش داده شده‌است:

#more /etc/anacrontab

```
/ #etc/anacrontab: configuration file for anacron
```

```
#See anacron(8) and anacrontab(5) for details.
```

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

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

```
# These replace cron's entries
```

```
1 5 cron.daily nice run-parts --report /etc/cron.daily
```

```
7 10 cron.weekly nice run-parts --report /etc/cron.weekly
```

```
@monthly 15 cron.monthly nice run-parts --report /etc/cron.monthly
```

نخستین شماره نشاندهنده وقفه یا فاصله تعداد روزهایی است که بین دو اجرای دستور قرار خواهند گرفت. شماره دوم نمایش دهنده تاخیر پیش از اجرای دستور به مدت ثانیه می باشد. پس از این دو شماره نیز دستور با پارامترها و آرگومان های مورد نیازش قرار خواهد گرفت.

هنگامی که دستوری اجرا می شود، زمان آن ثبت شده و anacron می داند که در چه زمانی آنرا اجرا کرده و چه زمانی باید آنرا مجددا اجرا کند. زمان بین دو اجرا نمی تواند کمتر از یک روز باشد، زیرا anacron مقایسه خود را بر مبنای روز و نه بر مبنای زمان انجام می دهد. هنگامی که اجرای دستوری به اتمام می رسد، anacron آنرا با یک پیغام اعلام می کند.

نحوه استفاده از این دستور به شکل زیر می باشد:

```
#anacron [-s][-f][-n][-d][-q][job]...
```

```
# anacron -u [job]...
```

جدول ۳ لیستی از گزینه های قابل استفاده به همراه این دستور را نمایش می دهد. این گزینه ها به قابلیت انعطاف این ابزار می افزایند. هرچند که anacron یک سرویس بوده و در هنگام بوت شدن کامپیوتر یا تغییر در سطوح اجرایی (runlevels)، اجرا می شود. برای تغییر anacron می توانید اسکریپت /etc/init.d/anacron را ویرایش نمایید، ولی این کار را در صورتی انجام دهید که با اسکریپت نویسی آشنایی داشته باشید.

گزینه	شرح
-f	زمان ذخیره شده را در نظر نگرفته و دستورات را اجرا خواهد کرد
-u	زمان‌های ذخیره شده را به زمان فعلی بروز می‌کند، ولی هیچ دستوری را اجرا نخواهد کرد
-s	اجرای دستورات را به صورت پی‌درپی انجام خواهد داد. دستور بعدی قبل از اتمام دستور در حال اجرای جاری اجرا نخواهد شد
-n	دستورات را اکنون اجرا می‌کند
-d	دستور را در پس‌زمینه انجام نخواهد داد. این گزینه خروجی دستور را نمایش داده و خطاها در syslog ثبت خواهند شد. خروجی دستورات نیز طبق روال پست خواهد شد
-q	جلوی هرگونه ارسال پیام به خروجی استاندارد را خواهد گرفت. این گزینه تنها به همزاه گزینه -d قابل استفاده می‌باشد

جدول ۳ گزینه‌های قابل استفاده به همراه دستور anacron