

Cron

عملي مشترك

محتوى مجاني غير مخصص للبيع التجاري



نظم تشغيل (1)

RB Informatics;

CRON

- هو جدول زمني لوظائف Linux يستخدم لإعداد المهام التي تعمل بشكل دوري في تاريخ أو فترة زمنية محددة .
- و يتم استخدام مهمة Cron لتحديد الأوامر المحددة أو ملفات shell script التي يحددها المستخدمون في ملفات Crontab .
- يتم مراقبة هذه الملفات بواسطة Cron daemon و يتم تنفيذ المهام وفقاً لجدول زمني محدد مسبقاً .
- يتم الإشارة إلى مهام Cron باستخدام ملفات Crontab
- يتم تحميل هذه الملفات في الذاكرة و يتم مراقبتها للإجراءات المحددة مسبقاً
- يستيقظ Cron كل دقيقة لفحص جميع Crontabs المخزنة و معرفة ما إذا كان هناك أمر يجب تنفيذه في الدقيقة الحالية .
- مثلاً :
 - أحتاج إلى إرسال Email معين كل يوم في ساعة محددة
 - تنفيذ backup كل يوم جمعة (على سبيل المثال)
 - Scheduler للمهام
- و غيرها من الأمور التي يحددها المستخدم و تتطلب تنفيذاً خلال فترة زمنية محددة

ملاحظة:

يستعمل Cron ال Cron daemon حيث تعمل هذه الخدمة منذ بدأ تشغيل النظام و هي التي تقوم بالفحص كل دقيقة للتأكد فيما إذا كان هناك مهمة مجدولة يجب تنفيذها و تتم كتابة المهام التي يريد المستخدم من خلال Crontab

إذن لنبدأ بكتابة المهام ضمن Crontab :

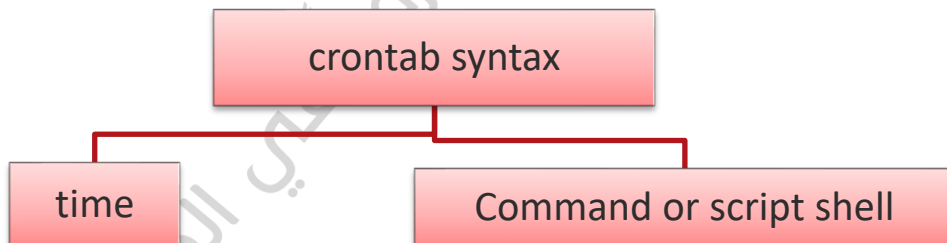
- يتم استخدام الأمر `crontab -e` للبدء بإدخال المهام المطلوبة و التعديل عليها .
- يظهر ملف Crontab الموضح أدناه

```
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
```

Read 22 lines

Get Help WriteOut Read File Prev Page Cut Text Cur Pos
Exit Justify Where Is Next Page UnCut Text To Spell

- سطر المهمة المراد تنفيذها يكون بالشكل `*/path/to/command`
- النجوم الخمسة تمثل الوقت و عوضا عن `path/to/command` يتم كتابة الأمر المطلوب



*	*	*	*	*
[minute]	[hour]	[day of the month]	[month]	[Day of the week]
(0-59)	(0-23)	(1-31)	(1-12)	(0-6)
			(or JAN – DEC)	(or name Sunday represented by SUN)

✍ لكتابة مهمة إنشاء مجلد على سطح المكتب في الساعة ال 3:00 ظهراً من كل يوم (على سبيل المثال) نقوم بالخطوات التالية :

1. **Crontab -e** سيظهر لدينا :

```
rawan@rawan-virtual-machine: ~
Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
```

- هنا محرر النصوص المستخدم هو vim ، و لكتابة تعليمة إنشاء المجلد نضغط على "S" للتحويل إلى وضع الإدخال ، ثم نكتب :

2. **0 15 *** mkdir ~ /Desktop/my-folder**

- بعد الانتهاء من كتابة التعليمة نضغط Esc ثم "wq": من أجل الحفظ ، و ها هو المجلد الذي قمنا بإنشائه

```
rawan@rawan-virtual-machine: ~/Desktop$ cd
rawan@rawan-virtual-machine: $ crontab -e
no crontab for rawan - using an empty one
crontab: installing new crontab
"/tmp/crontab.B0WH5I/crontab": bad minute
errors in crontab file, can't install.
Do you want to retry the same edit? (y/n) y
crontab: installing new crontab
"/tmp/crontab.B0WH5I/crontab": bad minute
errors in crontab file, can't install.
Do you want to retry the same edit? (y/n) y
crontab: installing new crontab
"/tmp/crontab.B0WH5I/crontab": bad minute
errors in crontab file, can't install.
Do you want to retry the same edit? (y/n) y
crontab: installing new crontab
rawan@rawan-virtual-machine: $
```

بعض التوضيحات حول كتابة الأوقات ضمن التعليمات :

- في حال كانت الأوقات جميعها * * * * * هذا يعني أنه سيتم تنفيذ المهمة "كل" دقيقة من كل ساعة في كل يوم في كل شهر .
- إذن عند وضع * في أي وقت من الأوقات (دقيقة , ساعة , ...) هذا يعني كل (دقيقة , ساعة , ...)
- تستخدم الفاصلة للدمج بين وقتين

على سبيل المثال :

المهمة المراد تنفيذها في بداية و منتصف كل ساعة فتتم الكتابة على الشكل التالي :

..... * * * * * 0 , 30

بدلاً من كتابة أمرين

0 * * * * *

30 * * * * *

- تستخدم "-" لتحديد مجال الأوقات المراد تنفيذ المهمة فيها

على سبيل المثال :

نريد تنفيذ المهمة في الدقيقة الأولى و الثانية حتى الثلاثين في كل ساعة فتتم كتابة الأمر:

..... * * * * * 0-29

- تستخدم "/" لتحديد مدة زمنية

على سبيل المثال :

لتحديد مدة زمنية مثلاً كل 3 ساعات ، كل 4 دقائق نكتب :

0 */3 * * * *



يتم تنفيذ المهمة كل دقيقة

***** 12

يتم تنفيذ المهمة في الدقيقة 12 من كل ساعة

* /15 ***** 0,15,30,45...

يتم تنفيذ المهمة كل 15 دقيقة

***** 04

يتم تنفيذ المهمة في الساعة 4:00 من كل يوم

***** 2-4

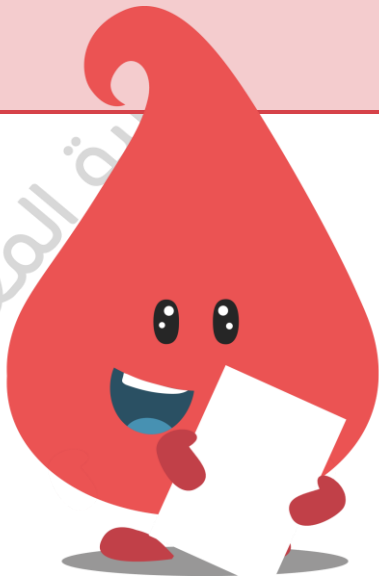
يتم تنفيذ المهمة كل يوم الثلاثاء و الأربعاء و خميس

: Think with RBCs

04**2-4

20, 40*/8*7-12

!?



إدارة ملفات تكوين Cron

- يعتبر Crontab نوعاً من ملفات التكوين تستخدم لتحديد متى ستقوم أنظمة Linux بتشغيل مهام معينة بشكل دوري .
- يحتوي Crontab على إعدادات البيئة و الأوامر التي تحدد ما يجب تنفيذه و متى.

↪ أنواع ملفات Crontab :

يوجد نوعان :

1. ملفات Crontab التي تخص المستخدمين الفرديين

2. ملفات Crontab على مستوى النظام

- لكل مستخدم ملف Crontab خاص به يحمل اسم حسابه كما و يتم تنفيذ الملف باسم المستخدم الذي يملكه
- بإمكانك تفقد ملفات Crontab التي يملكها المستخدم بتفقد الملفات في المجلد الذي يحتفظ بملفات Crontab

```
ls /var/spool/cron/crontabs
```

```
root@ubuntu-sandbox:~ > ls /var/spool/cron/crontabs/
root user1 user2
```

- كما ذكرنا مسبقاً للتعديل على ملفات Crontab باستخدام الأمر `crontab -e`
- لعرض محتوى ال Crontab الخاص بك مع عدم الإمكانية على التعديل يمكنك استخدام الأمر `Crontab -l`
- لحذف محتوى Crontab يمكنك استخدام الأمر `crontab -r -i`

↪ تطبيق عملي :

- من حساب ال root أو ال sudoer نقوم بتنفيذ ما يلي :

1. نذهب إلى المسار `/etc/cron.d` و ننشأ ملف جديد ليكن اسمه `cron1` ، ثم ننفذ التعليمة `vim` لفتح الملف السابق و ندخل السطر التالي :

```
***** username echo "hello from cron.d" >> /home/$USER/desktop/scheduling
```

2. ثم نقوم بتنفيذ التعليمة `crontab -e` و نقوم بتعديل الملف الظاهر و ندخل إلى السطر :

```
***** echo "hello from crontab" >> /home/$USER/desktop/scheduling
```

3. ثم ننشأ ملف جديد ضمن `/etc/cron.hourly` و لنسميه مثلاً `cron2` ، ثم ننفذ التعليمة `vim` لفتح الملف السابق و ندخل السطر التالي :

```
echo "hello from cron.hourly" >> /home/$USER/desktop/scheduling
```

4. نتأكد من تعديل صلاحيات الملف `cron2` لنضيف الصلاحية `x` ، نفتح الملف `/etc/anacrontab` من خلال `vim` و نضيف السطر التالي :

```
@daily 2 cron3 echo "hello from anacron" >> /home/$USER/desktop/scheduling
```

5. نهايةً نقوم بإضافة مهمة واحدة باستخدام : التعليمة `at TIME` و نحدد الوقت الذي نريده ، ثم نضيف السطر التالي :

```
at> echo "hello from at" >> /home/$USER/desktop/scheduling
```

6. و الآن نراقب الملف `/home/$USER/desktop/scheduling` و يجب أن تظهر لنا الرسائل في الوقت الذي حددناه من جميع المهام المجدولة .

7. لإجبار ال `anacron` على تنفيذ مهامها الآن ننفذ التعليمة : `anacron -d -n -f` .

UBUNTU NETWORK CONFIGURATION

- لعرض عنوان ال IP الحالي يمكنك استخدام الأمر `ip a` أو `ip addr`

```
kbuzdar@ubuntu:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:0e:2f:13 brd ff:ff:ff:ff:ff:ff
    inet 192.168.72.140/24 brd 192.168.72.255 scope global dynamic ens33
        valid_lft 1757sec preferred_lft 1757sec
    inet6 fe80::20c:29ff:fe0e:2f13/64 scope link
        valid_lft forever preferred_lft forever
```

UBUNTU Set Static IP Add في

- يستخدم أوبونتو netplan لإدارة الشبكة الافتراضية ، حيث يتم تخزين ملف إعدادات netplan في المسار
etc/netplan/
- للوصول إلى ملف الإعدادات نستخدم الأمر : ls /etc/netplan ، سيعيد هذا الأمر اسم ملف الإعدادات باللاحقة
yaml ، فيكون اسم الملف 01-network-manager-all.yaml
- يتم التعديل باستخدام أي محرر نصي و هنا تم استخدام محرر nano
- نضيف الأسطر التالية على ملف الإعدادات

```
network:
version: 2
renderer: NetworkManager
ethernets:
  ens33:
    dhcp4: no
    addresses:
      - 192.168.72.140/24
    gateway4: 192.168.72.2
    nameservers:
      addresses: [8.8.8.8, 8.8.4.4]
```

- نحفظ التغييرات باستخدام Ctrl + x
- و لتطبيق التغييرات نستخدم الأمر sudo netplan apply
- في حال كانت الإعدادات صحيحة سيرجع الأمر sudo netplan apply رسالة "configuration accepted" و
إلا سيتراجع الإعداد السابق
- للتحقق من أن عنوان ال IP قد تغيّر نستخدم الأمر ip a ، و هذا سيعرض معلومات حول واجهة
الشبكة (سيعرض التغييرات التي قمت بها)



SET THE DAYNAMIC IP ADDRESS

- قم بتعديل ملف إعدادات netplan باستخدام أي محرر نصوص

```
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    ens33:
      dhcp4: yes
      addresses: []
```

```
$ sudo netplan try
```

```
$ sudo netplan apply
```

وإلى هنا نكون قد وصلنا لنهاية المحاضرة..

دمتم بخير ^-^