



LPIC-1. Examen 102. Objectiu 107.2





LPI 107.2. Automatitzar tasques d'administració del sistema amb treballs programats

Wiki:

http://acacha.org/mediawiki/index.php/LPI_107.2



Objectius

107.2. Automatitzar tasques d'administració del sistema amb treballs programats	
	<ul style="list-style-type: none"> ▪ Objectiu: Els candidats han de ser capaços d'utilitzar cron o anacron per tal d'executar tasques en intervals regulars i utilitzar at per tal d'executar tasques en moments específics. ▪ Pes: 4
	<p>Àrees Clau de Coneixement:</p> <ul style="list-style-type: none"> ▪ Gestionar tasques de cron i at ▪ Configurar l'accés dels usuaris als serveis cron i at
	<p>La següent és una llista parcial de fitxers, termes i utilitats utilitzades:</p> <ul style="list-style-type: none"> ▪ /etc/cron.{d,daily,hourly,monthly,weekly} ▪ /etc/at.deny ▪ /etc/at.allow ▪ /etc/crontab ▪ /etc/cron.allow ▪ /etc/cron.deny ▪ /var/spool/cron/* ▪ crontab ▪ at ▪ atq ▪ atrm
	<p>Apunts: LPI 107.2. Automatitzar tasques d'administració del sistema amb treballs programats</p>



Cron

♦ Vixie-cron

- ♦ Servei de planificació de tasques basat en temps molt utilitzat en SO Unix-like (hi ha ports per a altres SO)
- ♦ És una abreviatura de **cronògraf**
- ♦ Permet als usuaris executar comandes o guions de shell de forma automàtica i periòdica.
- ♦ Utilitzat sovint pels administradors de sistemes com a eina per automatitzar tasques d'administració.
- ♦ Hi ha diverses implementacions i té una llarga història de modificacions. Utilitzarem el cron vixie, de l'autor Paul Vixie (també autor de Bind)

Hi ha altres implementacions de cron, com la de AT&T que poden diferir lleugerament



Servei cron

♦ Cron és un dimoni (abans executable crond)

- ♦ Dimoni: aplicació no interactiva que s'executa en segon terme de forma continuada

- ♦ Control del servei:

```
$ sudo /etc/init.d/cron  
Usage: /etc/init.d/cron COMMAND
```

- Opcions: start|stop|restart|reload|force-reload|status
- **Ubuntu:** servei **upstart**

```
$ sudo service cron status  
cron start/running, process 1387
```

- ♦ Instal·lació:

```
$ sudo apt-get install cron
```

- Normalment ja ve instal·lat



Configuració

♦ Crontab

- ♦ Són els fitxers que configuren cron
- ♦ També és l'ordre que permet consultar i editar els fitxers crontab
- ♦ Consulta:

```
$ crontab -l  
# m h dom mon dow    command
```
- ♦ Cada usuari té el seu crontab (incloent el superusuari)
- ♦ Editar crontab:
 - \$ crontab -e
 - \$ sudo crontab -e
 - \$ sudo crontab

```
#Modificar crontab d'altres usuaris:  
$ sudo crontab -e -u sergi  
#Eliminar crontab  
$ crontab -r  
$ sudo crontab -r -u sergi  
#Editar crontab a partir d'un fitxer  
$ sudo joe fitxer  
$ crontab fitxer
```



Configuració

♦ Format del fitxer

- ♦ S'utilitza l'editor del sistema (variable d'entorn EDITOR)
- ♦ El fitxer té 6 camps pels usuaris normals i 7 pel superusuari (pot indicar quin usuari executa la tasca)
 - **Camp 1 (minut):** Valor entre 00 i 59 o llista de valors
 - **Camp 2 (hora):** Valor entre 00 i 23 o llista de valors
 - **Camp 3 (dia):** Valor entre 1 i 31 o llista de valors
 - **Camp 4 (mes):** Valor entre 1 i 12 o llista de valors
 - **Camp 5 (weekday):** Valor entre 0 i 7 (0 i 7=diumenge) o sun, mon, tue, wen, thu, fri i sat. També permet llista de valors



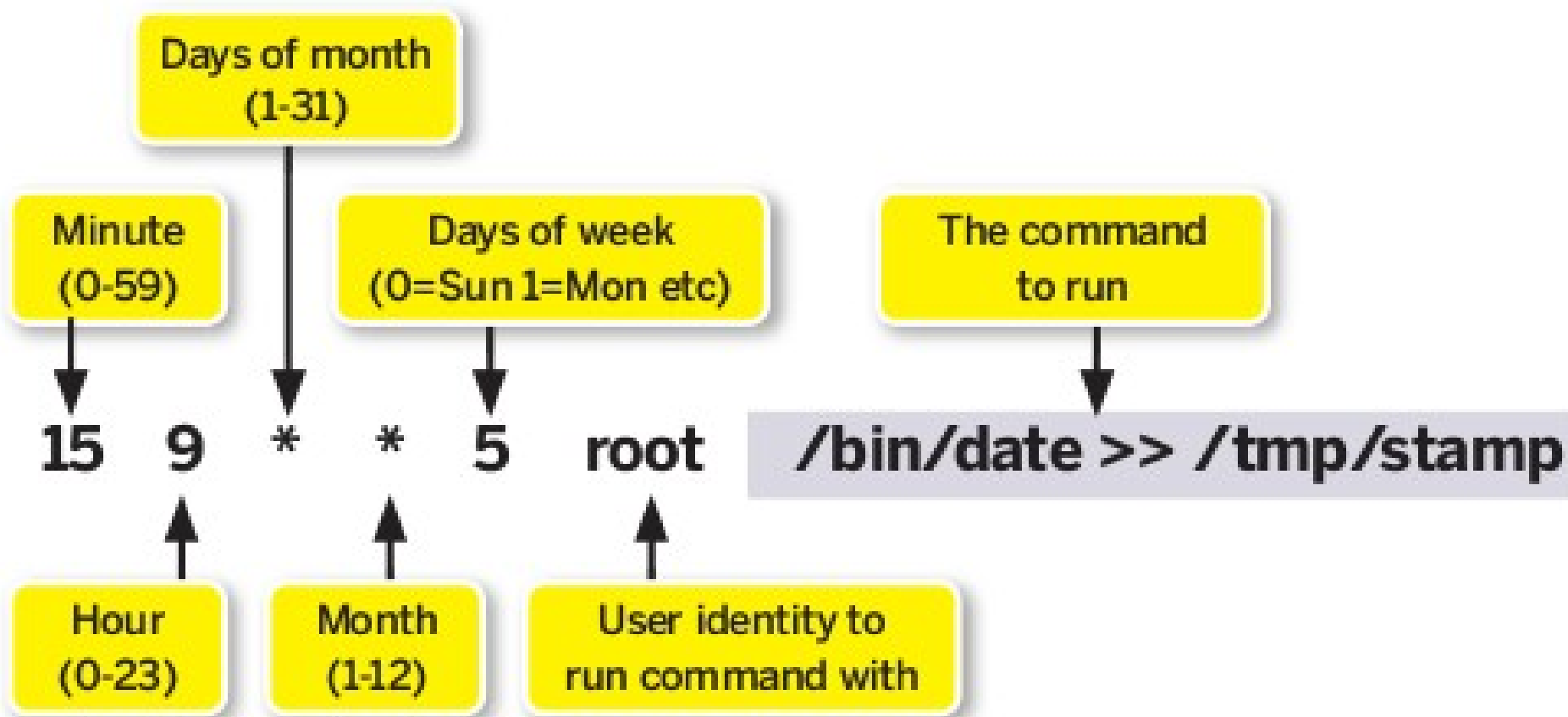
Configuració

- ◆ **Camp 6 (usuari):** nom de login de l'usuari. Només crontabs del sistema o de superusuari.
- ◆ **Camp 7 (ordre):** Ordre o guió d'ordres a executar.
- ◆ Els asterisc signifiquen tots els valors (tots els dies, totes les hores, tots els mesos, etc...)

```
$ cat /etc/crontab
...
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:...
# m h dom mon dow user  command
17 * * * * root    ... /etc/cron.hourly
25 6 * * * root    ... /etc/cron.daily )
47 6 * * 7 root    ... /etc/cron.weekly )
52 6 1 * * root    ... /etc/cron.monthly )
```



Configuració





Exemple

◆ **MessageBox cada 10 minuts**

◆ \$ crontab -e

```
# m h dom mon dow   command
10,20,30,40,50 * * * * env DISPLAY=:0 /usr/bin/zenity --warning
--title="Et recomanem..." --text="Que DESCANSIS" --width=500 >>
/home/sergi/logfile 2>&1
```

◆ Consulta del fitxers de log:

```
$ sudo tail --lines=500 /var/log/syslog | grep CRON
..
...
Apr 17 19:10:01 BSFHPCasa CRON[17449]: (sergi) CMD
(zenity --warning --title="Et recomanem..."
--text="Qu\303\250 DESCANSIS\!" --width=500)
```



/var/spool/cron

♦ Carpeta on es guarda la informació dinàmica de cron

♦ Crontabs: /var/spool/cron/crontabs/

```
$ sudo ls -l /var/spool/cron/crontabs/  
total 8  
-rw----- 1 root  crontab 274 2010-02-02 08:53 root  
-rw----- 1 sergi crontab 335 2010-04-17 15:01 sergi
```

♦ Seguretat i permisos

- No es poden modificar directament
- BIT de **GID** de l'ordre crontab:

```
$ whereis crontab  
crontab: /usr/bin/crontab /etc/crontab  
$ ls -la /usr/bin/crontab  
-rwxr-sr-x 1 root crontab 31712 2009-09-15 15:12 /usr/bin/crontab
```



Variables d'entorn

- ♦ **Es poden establir variables d'entorn a cada fitxer crontab.**
 - ♦ Les següents variables d'entorn s'estableixen segons el contingut del fitxer **/etc/passwd** de l'usuari:
 - **SHELL, USER, LOGNAME, i HOME**
 - ♦ **IMPORTANT:** La shell per defecte és sh (**/bin/sh**).
 - ♦ Totes les VE es poden modificar excepte USER.
 - ♦ Es pot establir TZ però s'ignorarà excepte si l'utilitza només en una ordre.
 - ♦ **MAILTO** s'utilitza per indicar el login name de l'usuari del sistema al qual se li envia el resultat de les ordres



Altres mètodes per automatitzar tasques

♦ Fitxer **/etc/crontab**

- NO es recomana editar-lo directament

♦ Carpeta **/etc/cron.d**

- Aquí és on les aplicacions col·loquen els seus scripts cron. No cal que siguin executables.

♦ Utilitzar les tasques diàries, setmanals, ...

- Segons el contingut de **/etc/crontab**
 - Cada hora al minut **17** s'executen el fitxers executables de la carpeta **/etc/cron.hourly**
 - Cada dia a les **6:65** s'executen el fitxers de **/etc/cron.daily**
 - Cada diumenge a les **6:47** s'executen el fitxers de **/etc/cron.weekly**
 - Cada dia 1 dels més a les **6:52** s'executen el fitxers de **/etc/cron.monthly**



Control de la sortida estàndard

- ◆ Els resultats dels crons s'envien per mail intern al propietari del cron o a el que s'hagi especificat a la variable **MAILTO**.
- ◆ Això és pot canviar amb redireccions. Per exemple:

```
cmd | mail -s "Subject of mail" email
```

- ◆ Si no es vol enviar correu es pot redireccionar a **/dev/null**:

```
$ cmd > /dev/null
```

- ◆ O es pot fer un log a un fitxer:

```
$ cmd >> log.file
```

- ◆ Dos majors que indiquen mode append. Un sol major es sobreescriu el contingut del fitxer de log.

- L'exemple anterior només redirecciona la sortida estàndard. Es pot redireccionar també els errors amb:

```
cmd >> logfile 2>&1
```



Cron i sh

♦ sh

- ♦ Hi ha petites diferències entre sh i bash. Cron té limitacions extremes com a shell
- ♦ Que un script funcioni amb bash no vol dir que funcioni amb la shell de cron. Un exemple:

```
/backups/scripts/files_backup.sh >& /logs/files_backup.log
```

- Això funciona amb bash però si ho executem amb sh o cron ens donarà l'error. En canvi en:

```
$ sh  
# /backups/scripts/files_backup.sh >& /disk/logs/files_backup.log  
sh: Syntax error: Bad fd number
```



Troubleshooting

♦ Els scripts s'executen amb sh

- Per tant comproveu que la vostra comanda funciona amb sh

♦ Salts de línia

- Vigileu que les comandes acabin amb un salt de línia, tant al fitxer cron (per exemple /etc/cron.d/elmeucron) com l'script que escriviu

♦ Eines gràfiques

- Cal indicar la variable d'entorn DISPLAY. Per exemple:

```
00 06 * * * env DISPLAY=:0.0 gui_appname
```

- Si es tenen diversos monitors no heu d'oblidar indicar quin voleu utilitzar



Troubleshooting

- A més a Ubuntu 9.10 Karmic Koala cal:

- ~\$ xhost +local:

- ♦ **Scripts**

- **Shebang:** `#!/bin/sh`

- ♦ **Diferencia de format entre crontab de superusuari i crontab d'usuari normal**

- 1 columna més a root --> Permet indicar l'usuari

- ♦ **Caràcter %**

- S'utilitza a cron per indicar una nova línia, si voleu utilitzar aquest caràcter cal escapar-lo amb `"\%"`.



Troubleshooting

- ◆ **Utilitzar fitxers de log per veure el resultat de les ordres**
 - Mostrar tant la sortida estàndard com la sortida estàndard d'error:
 - `echo "hola!">> /home/sergi/logfile 2>&1`
- ◆ **Comprovar a syslog que s'executa cron**

```
$ sudo tail --lines=500 /var/log/syslog | grep CRON
..
...
Apr 17 19:10:01 BSFHPCasa CRON[17449]: (sergi) CMD (zenity
--warning --title="Et recomanem..." --text="DESCANSA"
--width=500)
```



Control d'accés a cron

- ♦ Es pot especificar qui té permisos per executar cron i qui no. L'ús és similar al cas dels fitxers host (o molts altres dimonis). Dos fitxers controlen els permisos i les denegacions:
 - **/etc/cron.allow** i **/etc/cron.deny**
- ♦ Per impedir que un usuari utilitzi cron es coloca el nom de l'usuari al fitxer cron.deny
 - Si es vol evitar que tots els usuaris utilitzin cron es pot utilitzar **ALL**:
 - **\$ sudo bash -c "echo ALL >> /etc/cron.deny"** o
 - **\$ sudo -s**
 - **# echo ALL >> /etc/cron.deny**



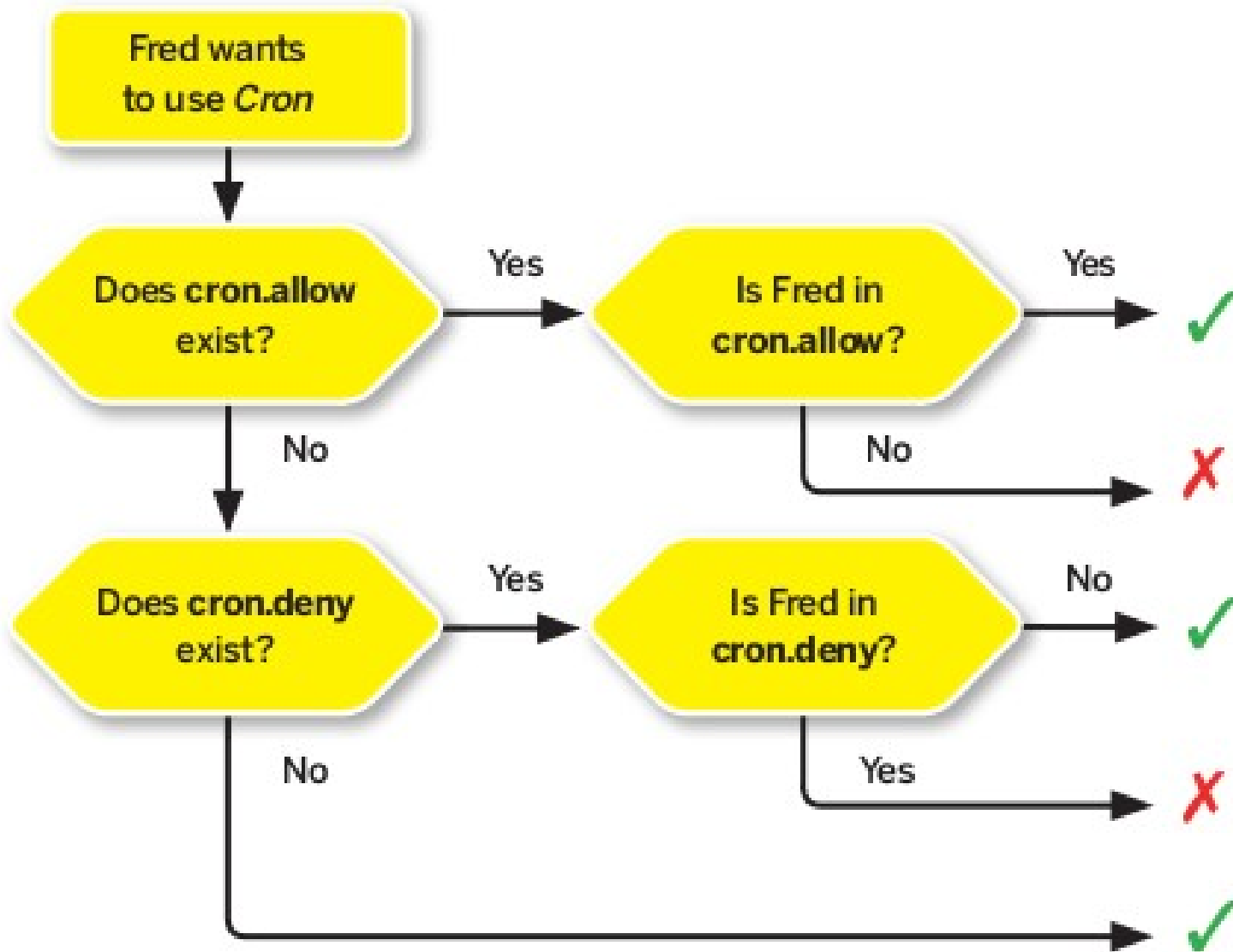
Control d'accés a cron

♦ Permetre l'usuari foo:

- `# echo cog >>/etc/cron.allow`
- Si no hi ha res als fitxers tothom pot utilitzar cron.
- ♦ Si utilitzem el fitxer `cron.allow` només els usuaris que surtin en aquest fitxer podran utilitzar cron.
- ♦ A una Ubuntu 9.10 els fitxers no estan ni indicats, els hauríeu de crear.
 - Per tant per defecte tots els usuaris poden utilitzar cron.



Control d'accés a cron





Exemples

- ♦ Mantenir despert. Un missatge cada dia a cada hora en punt

```
0 * * * * echo "WAKE UP" 2>&1 /dev/console
```

- ♦ Cada deu minuts:

```
10,20,30,40,50 * * * * /adm/checkdaemon 2>&1 | /bin/mail  
-s "CRON:Check" root
```

- ♦ Exemples backup

- Executar tots els dies un scripts de còpia de seguretat a les 7:00 del matí:
 - 0 7 * * * /home/usuario/copiadeseguridad.sh
- Executar tots els primers dies de mes un scripts de copia de seguretat a les 7:00 del matí:
 - 0 7 1 * * /home/usuario/copiadeseguridad.sh
- Executar tots els divendres a les 21:30 un script de còpia de seguretat:
 - 30 21 * * 5 /home/usuario/copiadeseguridad.sh



anacron

♦ Asynchronous cron

- ♦ Anacron a diferència de cron no pressuposa que la màquina s'està executant de forma continuada.

Les tasques de cron que “s'executen” quan la màquina està aturada es salten i mai s'executen

- ♦ Només l'administrador del sistema pot executar i pot configurar tasques d'anacron
- ♦ Anacron només pot executar tasques un cop al dia (o en períodes més grans: setmanes, mesos, etc.) en contrast amb cron que es pot arribar a executar cada minut.
- ♦ Si el sistema es torna a iniciar o s'inicia després de la mitjanit, les tasques diàries són executades després de cert retard i de forma ordenada (una tasca anacron a l'hora).



Anacron

- ♦ De fet anacron **utilitza cron** per executar-se, consulteu el fitxer **/etc/cron.d/anacron**.
 - S'assegura que les tasques diàries s'executin sempre encara que siguin a hores que l'ordinador no està encès:

```
$ cat /etc/cron.d/anacron
...
#30 7 * * * root test -x /etc/init.d/anacron &&
/usr/sbin/invoke-rc.d anacron start >/dev/null
30 7 * * * root start -q anacron || :
```

- Anacron no és un dimoni o servei. Només s'executa durant l'arrancada del sistema o es executa periòdicament per cron.



Configuració

♦ Fitxer `/etc/anacrontab`

- ♦ No hi ha ordre `anacrontab`

```
$ cat /etc/anacrontab
...
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
```

- ♦ Sintaxi: `period|@period_name delay job-identifier command`

- **period:** en dies o també es pot especificar `@monthly` (un sol cop al més amb independència del nombre de dies)
- **delay:** S'especifica en minuts
- **job-identifier:** string sense espais en blanc que identifica la tasca als fitxers de log
- **command:** Un ordre o guió d'ordres a executar.



at

- ♦ Permet especificar que es puguin executar ordres un sol cop en un moment donat:

```
$ date
ds abr 17 19:49:16 CEST 2010
$ tty
/dev/pts/15

#Ara li diem que executi un hola a les 19:52:

$ at 19:52
warning: commands will be executed using /bin/sh
at> echo "hello" | wall
at>                <-- Ctrl+D
job 1 at Sat Apr 17 19:52:00 2010
```

```
$ echo "Hola" | wall | at now + 1 minutes
```



at

- ♦ **Es pot consultar una llista de les tasques pendents amb atq (at queue)**

```
$ date
dl abr 19 21:02:11 CEST 2010
$ tty
/dev/pts/2
$ at 21:04
warning: commands will be executed using /bin/sh
at> echo hola | wall
at>                                <-- Ctrl+D (estil heredoc)
job 8 at Mon Apr 19 21:04:00 2010
$ atq
8   Mon Apr 19 21:04:00 2010 a sergi
```

- ♦ El divendres a les 4:45 de la tarda
- ♦ Es pot eliminar una tasca amb **atrm**

at 4:55pm Friday



Control d'accés a at

♦ Molt similar a cron

- ♦ És pot especificar qui te permisos per executar at i qui no.
 - /etc/at.allow
 - /etc/at.deny
- ♦ Si es vol evitar que tots els usuaris utilitzin cron es pot utilitzar ALL:
 - `$ sudo bash -c "echo ALL >> /etc/cron.deny"`
 - Si vols permetre el usuari foo:
 - `# echo cog >>/etc/at.allow`
- ♦ Si no hi ha res als fitxers tothom pot utilitzar at.



Reconeixement 3.0 Unported

Sou lliure de:



copiar, distribuir i comunicar públicament l'obra



fer-ne obres derivades

Amb les condicions següents:



Reconeixement. Heu de reconèixer els crèdits de l'obra de la manera especificada per l'autor o el llicenciador (però no d'una manera que suggereixi que us donen suport o rebeu suport per l'ús que feu l'obra).

- Quan reutilitzeu o distribuïu l'obra, heu de deixar ben clar els termes de la llicència de l'obra.
- Alguna d'aquestes condicions pot no aplicar-se si obteniu el permís del titular dels drets d'autor.
- No hi ha res en aquesta llicència que menyscabi o restringeixi els drets morals de l'autor.

Advertiment

Els drets derivats d'usos legítims o altres limitacions reconegudes per llei no queden afectats per l'anterior
Això és un resum fàcilment llegible del text legal (la llicència completa).

<http://creativecommons.org/licenses/by/3.0/deed.ca>