

## 3.1-TAREAS PROGRAMADAS (LINUX)

## Índice:

COMANDO AT: .....	2
1) Programa una tarea con el comando at de manera que a las 12:00 horas almacene en un fichero procesos.txt todos los nombres de los procesos que se están ejecutando junto con información de estos. ....	2
2) Averigua los ficheros que contienen información de nombres de usuarios a los que les está permitido o no usar el comando at. ....	3
COMANDO CRON:.....	4
1) Programa una tarea con el comando crontab que realice una copia/backup comprimida con tar.gz de tu /home/usuario/Descargas a las 12:00 horas diariamente bajo /opt mediante crontab. ....	4
2) Señala otra manera de configurar tareas programadas en Linux, mediante cron. Programa una tarea libre mediante cron scripts en los directorios cron.hourly, cron.daily, cron.weekly y cron.monthly en /etc/, que sean diferentes en cada uno y teniendo en cuenta que el planificador de tareas trabaja en segundo plano. Muestra el contenido del script que se llevará a cabo. ....	5
INTERFAZ GRÁFICA: .....	7
1) ¿Cómo programar tareas mediante entorno gráfico en Linux? Investiga diferentes programas, elige uno y programa una tarea libre. Documentala. ....	7

## COMANDO AT:

- 1) Programa una tarea con el comando at de manera que a las 12:00 horas almacene en un fichero procesos.txt todos los nombres de los procesos que se están ejecutando junto con información de estos.

Instalamos AT:

**apt install at**

Ahora ponemos:

**at 12:00**

y luego el comando que queremos usar, en nuestro caso:

**ps -f > procesos.txt**

Una vez terminado, usamos **Ctrl+D** para salir.

```
Cristobal CON ROOT  martes 16 diciembre 2025 09:57
[root@server2asir usuario]$at 12:00
warning: commands will be executed using /bin/sh
at Tue Dec 16 12:00:00 2025
at> ps -f > procesos.txt
at> <EOT>
job 1 at Tue Dec 16 12:00:00 2025
Cristobal CON ROOT  martes 16 diciembre 2025 09:58
[root@server2asir usuario]$
```

- 2) Averigua los ficheros que contienen información de nombres de usuarios a los que les está permitido o no usar el comando at.

**/etc/at.allow:** Lista los usuarios que tienen permitido usar at.

**/etc/at.deny:** Lista los usuarios que tienen prohibido usar at.

**NOTA:** Si at.allow existe, solo los usuarios listados ahí pueden usarlo. Si no existe, se consulta at.deny. Si ninguno existe, dependiendo de la distribución, solo el root podría usarlo.

```
Cristobal CON ROOT  martes 16 diciembre 2025 10:01
[root@server2asir usuario]$cat /etc/at.deny
alias
backup
bin
daemon
ftp
games
gnats
guest
irc
lp
mail
man
nobody
operator
proxy
qmaild
qmail
qmailp
qmailq
qmailr
qmails
sync
sys
www-data
```

## COMANDO CRON:

- 1) Programa una tarea con el comando crontab que realice una copia/backup comprimida con tar.gz de tu /home/usuario/Descargas a las 12:00 horas diariamente bajo /opt mediante crontab.

Creamos la carpeta descarga y nos bajamos algo.

**mkdir Descargas**

Abrimos crontab:

**crontab -e**

elegimos **nano**

```
[root@server2asir Descargas]$crontab -e
no crontab for root - using an empty one

Select an editor. To change later, run 'select-editor'.
 1. /bin/nano      <---- easiest
 2. /usr/bin/vim.basic
 3. /usr/bin/vim.tiny
 4. /bin/ed

Choose 1-4 [1]: 1
```

Ponemos al final del archivo

**00 12 \* \* \* tar -czf /opt/backup\_descargas\_\$(date +%F).tar.gz  
/home/usuario/Descargas**

```
# For more information see the manual pages of crontab(8) and cron(8)
#
# m h dom mon dow   command
00 12 * * * tar -czf /opt/backup_descargas_$(date +%F).tar.gz /home/usuario/Descargas|
```

Guardamos y salimos. Ya está configurado.

- 2) Señala otra manera de configurar tareas programadas en Linux, mediante cron. Programa una tarea libre mediante cron scripts en los directorios cron.hourly, cron.daily, cron.weekly y cron.monthly en /etc/, que sean diferentes en cada uno y teniendo en cuenta que el planificador de tareas trabaja en segundo plano. Muestra el contenido del script que se llevará a cabo.

Las carpetas de cron son:

```
[root@server2asir cron.monthly]$ nano /etc/cron
cron.d/      cron.daily/  cron.hourly/ cron.monthly/ crontab      cron.weekly/
```

ls -l /etc | grep cron

```
[root@server2asir usuario]$ ls -l /etc | grep cron
drwxr-xr-x  2 root root      4096 dic 16 11:34 cron.d
drwxr-xr-x  2 root root      4096 dic 16 10:16 cron.daily
drwxr-xr-x  2 root root      4096 dic 16 10:15 cron.hourly
drwxr-xr-x  2 root root      4096 dic 16 10:24 cron.monthly
-rw-r--r--  1 root root      1136 mar 23  2022 crontab
drwxr-xr-x  2 root root      4096 dic 16 10:22 cron.weekly
```

- hourly/

registro\_recursos.sh

```
#!/bin/bash
# Sincroniza la hora y registra el uso de RAM
date >> /var/log/recursos_sistema.log
free -m >> /var/log/recursos_sistema.log
echo "---" >> /var/log/recursos_sistema.log
```

- cron.daily/

limpieza\_sistema.sh

```
#!/bin/bash
# Limpiar caché de paquetes (ejemplo para APT)
apt-get clean
# Borrar archivos en /tmp con más de 7 días de antigüedad
find /tmp -type f -atime +7 -delete
```

- cron.weekly/

reporte\_usuarios.sh

**#!/bin/bash**

**# Generar un informe de accesos de la última semana**

**echo "Reporte semanal de accesos - \$(date)" >> /var/log/reporte\_accesos.txt**

**last --since -7days >> /var/log/reporte\_accesos.txt**

**echo "-----" >> /var/log/reporte\_accesos.txt**

- cron.monthly/

backup\_config.sh

**#!/bin/bash**

**# Hacer copia de seguridad de la carpeta de configuración de red**

**FECHA=\$(date +%Y-%m)**

**tar -czf /root/backup\_network\_\$(FECHA).tar.gz /etc/network/ 2>/dev/null**

## INTERFAZ GRÁFICA:

- 1) ¿Cómo programar tareas mediante entorno gráfico en Linux? Investiga diferentes programas, elige uno y programa una tarea libre. Documentala.

<https://webmin.com/download/>

- `curl -o webmin-setup-repo.sh`  
<https://raw.githubusercontent.com/webmin/webmin/master/webmin-setup-repo.sh>
- `sudo sh webmin-setup-repo.sh`
- `sudo apt-get install webmin --install-recommends`

Como tenemos instalado webmin, vamos a usar eso: <https://10.2.7.56:10000>

Y buscamos **Scheduled Cron Jobs**.

The screenshot shows the Webmin interface for managing scheduled cron jobs. The page title is 'Scheduled Cron Jobs'. At the top, there is a search bar and several action buttons: 'Select all', 'Invert selection', 'Create a new scheduled cron job', 'Create a new environment variable', 'Control user access to cron jobs', and 'Manually edit cron jobs'. Below these is a table with columns 'User', 'Active?', and 'Command'. The table lists several cron jobs for the 'root' user, including system maintenance tasks like 'dpkg', 'apt-get', and 'cron.daily' jobs. At the bottom, there are buttons for 'Delete Selected Jobs', 'Disable Selected Jobs', and 'Enable Selected Jobs'. A red circle highlights the 'Create a new scheduled cron job' button. Below the table, there is a section for 'Stop Cron Daemon' and a checkbox for 'Start Cron Daemon Boot?'.

Le damos a Create New Scheduled Cron Job.



## ☆ Create Cron Job

## Job Details

## Execute cron job as

usuario



## Active

☒ Yes ☐ No

## Command

echo "\$(date) ¿Ha salido Half-Life 3?: NO"

## Input to command

|

## Description

Registro de si ha salido Half-Life 3

## When to execute

☒ Simple schedule Daily (at midnight) ▾ ☐ Times and dates selected below ..

## Minutes

☒ All☐ Selected ..

0	12	24	36	48
1	13	25	37	49
2	14	26	38	50
3	15	27	39	51
4	16	28	40	52
5	17	29	41	53
6	18	30	42	54
7	19	31	43	55
8	20	32	44	56
9	21	33	45	57
10	22	34	46	58
11	23	35	47	59

## Hours

☒ All☐ Selected ..

0	12
1	13
2	14
3	15
4	16
5	17
6	18
7	19
8	20
9	21
10	22
11	23

## Days

☒ All☐ Selected ..

1	13	25
2	14	26
3	15	27
4	16	28
5	17	29
6	18	30
7	19	31
8	20	
9	21	
10	22	
11	23	
12	24	

## Months

☒ All☐ Selected ..

January
February
March
April
May
June
July
August
September
October
November
December

## Weekdays

☒ All☐ Selected ..

Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday

Tip: Use Ctrl-click (or ⌘-click on Mac) to select or deselect multiple items, or Shift-click to select a range

## Date range to execute

☒ Run on any date☐ Only run from

Jan ▾



to

Jan ▾



Create

## Scheduled Cron Jobs


Find Cron jobs matching  
☒ Select all ☐ Invert selection    ☒ Manually edit cron jobs

User	Active?	Command	Move
<input type="checkbox"/> root	Yes	<a href="#">/etc/cron.daily/dpkg</a> <a href="#">/etc/cron.daily/appport</a> <a href="#">/etc/cron.daily/logrotate</a> <a href="#">/etc/cron.daily/apt-compat</a> <a href="#">/etc/cron.daily/cracklib-runtime</a> <a href="#">/etc/cron.daily/man-db</a>	
<input type="checkbox"/> root	Yes	<a href="#">/etc/cron.weekly/man-db</a>	
<input type="checkbox"/> root	Yes	<code>test -e /run/systemd/system    SERVICE_MODE=1 /usr/lib/x86_64-linux-gnu/e2fsprog ...</code>	
<input type="checkbox"/> root	Yes	<code>test -e /run/systemd/system    SERVICE_MODE=1 /sbin/e2scrub_all -A -r</code>	
<input type="checkbox"/> root	Yes	<code>if [ \$(date +%w) -eq 0 ] &amp;&amp; [ -x /usr/lib/zfs-linux/trim ]; then /usr/lib/zfs-li ...</code>	
<input type="checkbox"/> root	Yes	<code>if [ \$(date +%w) -eq 0 ] &amp;&amp; [ -x /usr/lib/zfs-linux/scrub ]; then /usr/lib/zfs-l ...</code>	
<input type="checkbox"/> root	No	<code>tar -czf /var/backups/home.tgz /home/</code>	
<input type="checkbox"/> root	Yes	<code>tar -czf /opt/backup_descargas_\$(date +%F).tar.gz /home/usuario/Descargas</code>	
<input type="checkbox"/> usuario	Yes	<code>echo "\$(date) ¿Ha salido Half-Life 3?: NO"</code>	

☒ Select all ☐ Invert selection    ☒ Manually edit cron jobs

  
 Shut down the crond background server process that runs scheduled cron jobs. This will prevent commands from executing at their specified times.

 ☒ Yes ☐ No Change this setting to enable or disable starting the scheduled cron jobs daemon at system boot time.