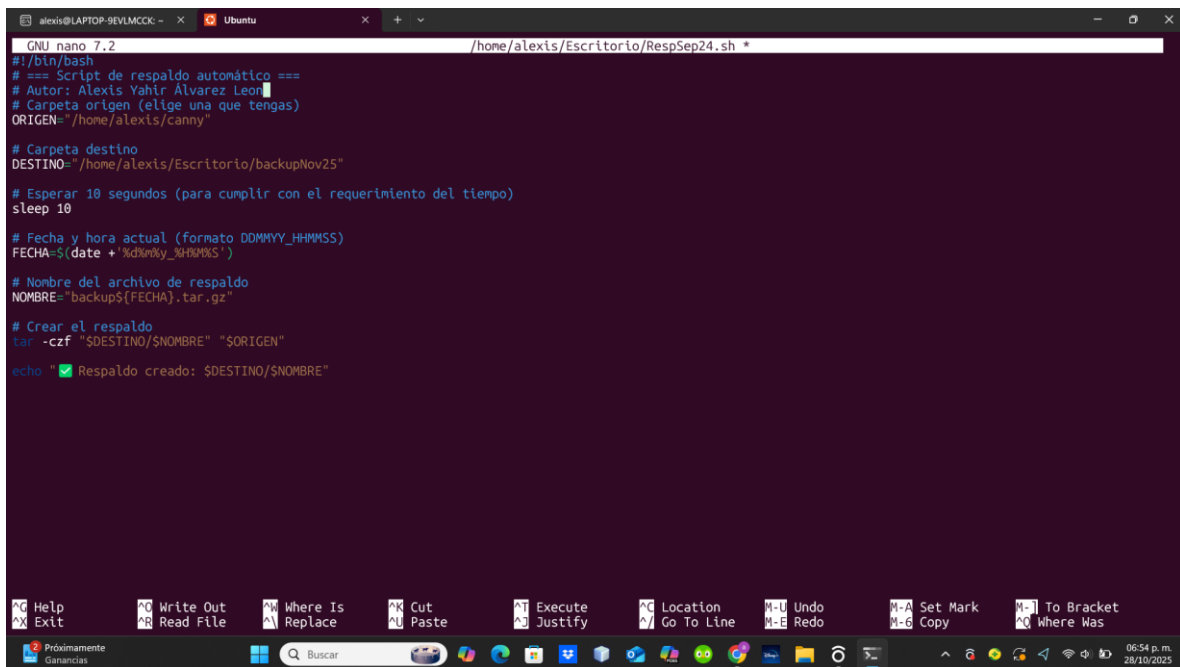


TAREA BACKUP PROGRAMADO

ALEXIS YAHIR ALVAREZ LEON

SISTEMAS OPERATIVOS

1-Se creo el archivo RespSep24.sh y se puso este codigo donde se van aguardar los respaldos en formato de fecha y se cree el respaldo



```
GNU nano 7.2 /home/alexis/Escritorio/RespSep24.sh *
#!/bin/bash
# === Script de respaldo automatico ===
# Autor: Alexis Yahir Alvarez Leon
# Carpeta origen (elige una que tengas)
ORIGEN="/home/alexis/canny"

# Carpeta destino
DESTINO="/home/alexis/Escritorio/backupNov25"

# Esperar 10 segundos (para cumplir con el requerimiento del tiempo)
sleep 10

# Fecha y hora actual (formato DDMMYY_HHMMSS)
FECHA=$(date + '%d%m%y_%H%M%S')

# Nombre del archivo de respaldo
NOMBRE="backup${FECHA}.tar.gz"

# Crear el respaldo
tar -czf "$DESTINO/$NOMBRE" "$ORIGEN"

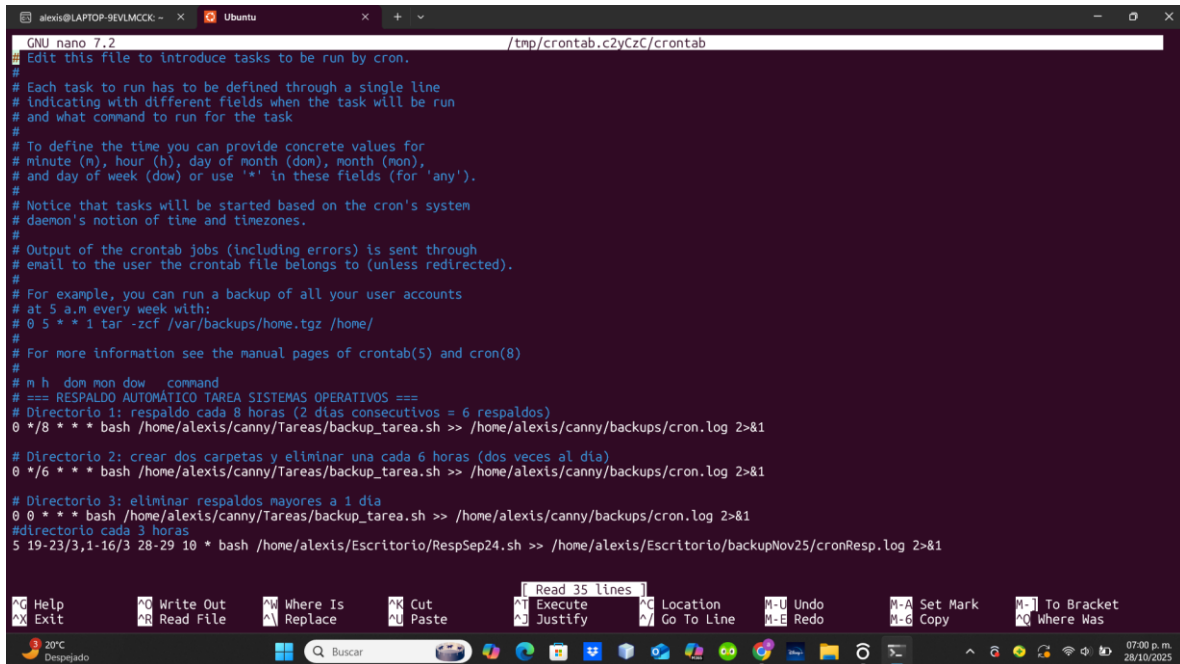
echo "✅ Respaldo creado: $DESTINO/$NOMBRE"
```

2-le dimos permisos y lo ejecutamos para ver si el archivo tar.gz se creo

```
alexis@LAPTOP-9EVLNCK: $ mkdir -p ~/Escritorio/backupNov25
alexis@LAPTOP-9EVLNCK: $ ls ~
Escritorio  archivo.txt  cesar_full.py  hola_mundo.txt  name.sh  salida.txt  to-do.txt
IDUNICO.txt  canny        crear_usuario.sh  llanto.sh  nombre.txt  saldauno.txt
Sistemas-Operativos-  canny.py  fonseca.txt  niDirectorio  output.txt  saludos1.txt
UltCon2025B.txt  cesar.py  funciones.sh  monosaurio.txt  packages.txt  saludos1.txt.gpg
alexis@LAPTOP-9EVLNCK: $ nano ~/Escritorio/RespSep24.sh
alexis@LAPTOP-9EVLNCK: $ chmod +x ~/Escritorio/RespSep24.sh
alexis@LAPTOP-9EVLNCK: $ ls -l
total 136
drwxr-xr-x 3 alexis alexis 4096 Oct 28 18:54 Escritorio
-rw-r--r-- 1 alexis alexis 1371 Oct 16 16:41 IDUNICO.txt
drwxr-xr-x 3 alexis alexis 4096 Aug 23 17:34 Sistemas-Operativos-
-rw-r--r-- 1 antonio yaned 768 Sep 26 16:08 UltCon2025B.txt
-rw-r--r-- 1 alexis alexis 47 Sep 1 17:15 archivo.txt
drwxr-xr-x 13 alexis alexis 4096 Oct 21 21:36 canny
-rw-r--r-- 1 alexis alexis 1 Aug 28 15:33 canny.py
-rw-r--r-- 1 alexis alexis 1236 Oct 23 16:49 cesar.py
-rw-r--r-- 1 alexis alexis 1237 Oct 23 16:50 cesar_full.py
-rwxr-xr-x 1 root root 1224 Oct 16 16:40 crear_usuario.sh
-rw-r--r-- 1 alexis alexis 91 Sep 25 15:35 fonseca.txt
-rwxr-xr-x 1 alexis alexis 75 Oct 16 20:18 funciones.sh
-rw-r--r-- 1 alexis alexis 15 Oct 16 16:12 hola_mundo.txt
-rwxr-xr-x 1 root root 225 Oct 16 15:52 llanto.sh
drwxr-xr-x 2 alexis alexis 4096 Oct 16 20:14 niDirectorio
-rw-r--r-- 1 alexis alexis 128 Sep 25 15:36 monosaurio.txt
-rwxr-xr-x 1 root root 76 Oct 16 16:14 name.sh
-rw-r--r-- 1 alexis alexis 13 Oct 16 16:15 nombre.txt
-rw-r--r-- 1 alexis alexis 438 Sep 25 15:44 output.txt
-rw-r--r-- 1 alexis alexis 38297 Aug 28 20:28 packages.txt
-rw-r--r-- 1 alexis alexis 219 Sep 25 15:43 salida.txt
-rw-r--r-- 1 alexis alexis 219 Sep 25 15:39 saldauno.txt
-rw-r--r-- 1 root root 13 Oct 24 16:59 saludos1.txt
-rw-r--r-- 1 root root 94 Oct 24 17:00 saludos1.txt.gpg
-rw-r--r-- 1 root root 65 Oct 16 15:46 to-do.txt
alexis@LAPTOP-9EVLNCK: $
alexis@LAPTOP-9EVLNCK: $ bash ~/Escritorio/RespSep24.sh
tar: Removing leading '/' from member names
Respaldo creado: /home/alexis/Escritorio/backupNov25/backup281025_185653.tar.gz
alexis@LAPTOP-9EVLNCK: $ ls -lh ~/Escritorio/backupNov25
total 50M
-rw-r--r-- 1 alexis alexis 50M Oct 28 18:56 backup281025_185653.tar.gz
alexis@LAPTOP-9EVLNCK: $
```

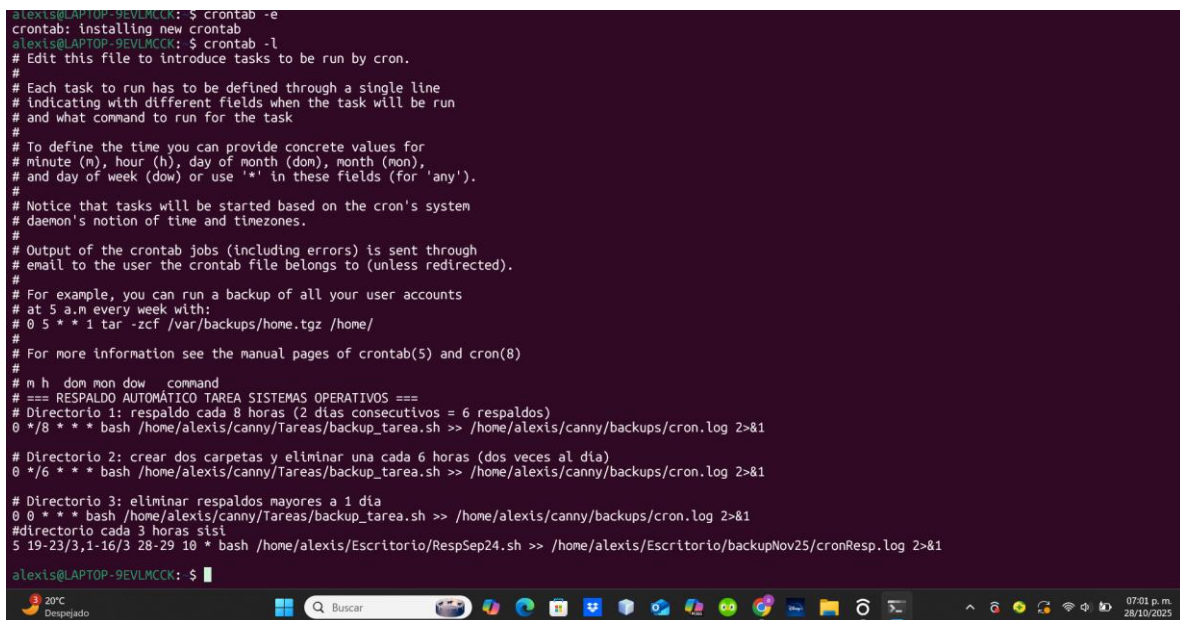
(si se creo ) este es el del manual

3- ejecutamos el comando `crontab -e` para añadir la línea `#directorio 3` para que se ejecute el comando cada 3 horas con 5 minutos y 10 segundos, de 7p a 7pm



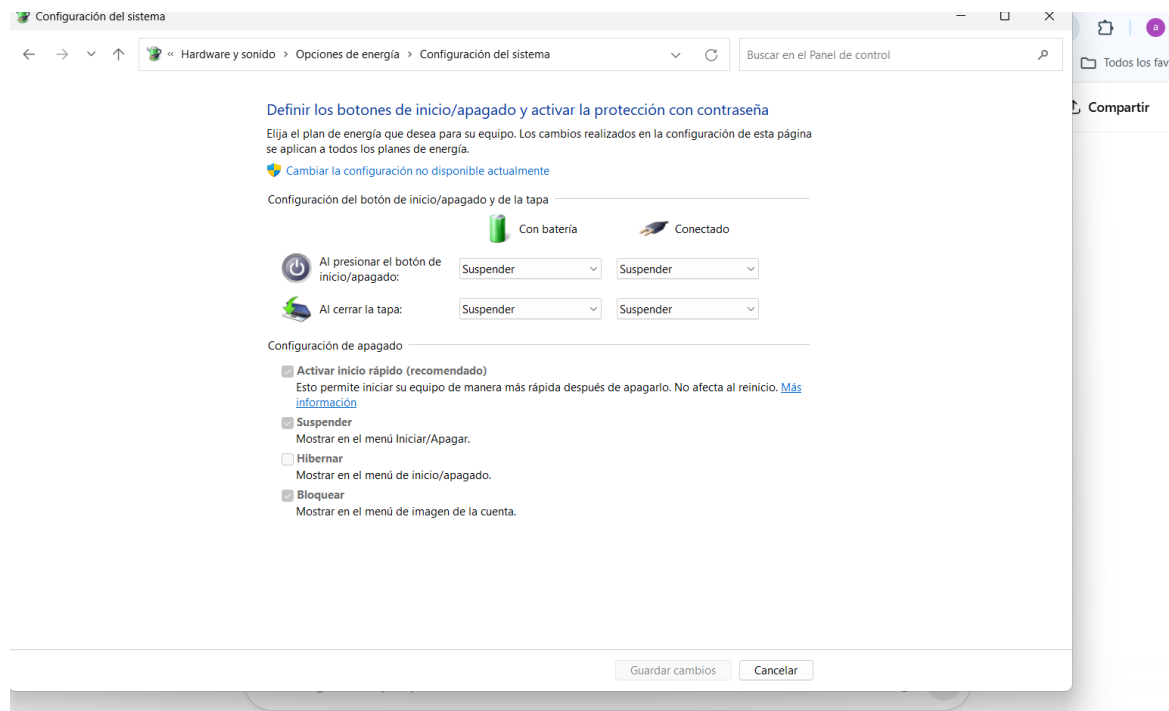
```
GNU nano 7.2 /tmp/crontab.c2yCz/crontab
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete 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
# === RESPALDO AUTOMÁTICO TAREA SISTEMAS OPERATIVOS ===
# Directorio 1: respaldo cada 8 horas (2 días consecutivos = 6 respaldos)
0 * 8 * * * bash /home/alexis/canny/Tareas/backup_tarea.sh >> /home/alexis/canny/backups/cron.log 2>&1
#
# Directorio 2: crear dos carpetas y eliminar una cada 6 horas (dos veces al día)
0 * 6 * * * bash /home/alexis/canny/Tareas/backup_tarea.sh >> /home/alexis/canny/backups/cron.log 2>&1
#
# Directorio 3: eliminar respaldos mayores a 1 día
0 0 * * * bash /home/alexis/canny/Tareas/backup_tarea.sh >> /home/alexis/canny/backups/cron.log 2>&1
#directorio cada 3 horas
5 19-23/3,1-16/3 28-29 10 * bash /home/alexis/Escritorio/RespSep24.sh >> /home/alexis/Escritorio/backupNov25/cronResp.log 2>&1
```

4-el comando `crontab -l` es para ver si el comando si se añadió



```
alexis@LAPTOP-9EVLKCK: $ crontab -e
crontab: installing new crontab
alexis@LAPTOP-9EVLKCK: $ crontab -l
# 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
# === RESPALDO AUTOMÁTICO TAREA SISTEMAS OPERATIVOS ===
# Directorio 1: respaldo cada 8 horas (2 días consecutivos = 6 respaldos)
0 * 8 * * * bash /home/alexis/canny/Tareas/backup_tarea.sh >> /home/alexis/canny/backups/cron.log 2>&1
#
# Directorio 2: crear dos carpetas y eliminar una cada 6 horas (dos veces al día)
0 * 6 * * * bash /home/alexis/canny/Tareas/backup_tarea.sh >> /home/alexis/canny/backups/cron.log 2>&1
#
# Directorio 3: eliminar respaldos mayores a 1 día
0 0 * * * bash /home/alexis/canny/Tareas/backup_tarea.sh >> /home/alexis/canny/backups/cron.log 2>&1
#directorio cada 3 horas
5 19-23/3,1-16/3 28-29 10 * bash /home/alexis/Escritorio/RespSep24.sh >> /home/alexis/Escritorio/backupNov25/cronResp.log 2>&1
alexis@LAPTOP-9EVLKCK: $
```

## 5-Desactive esto para que el programa ejecutara sin necesidad de que se apagara la laptop



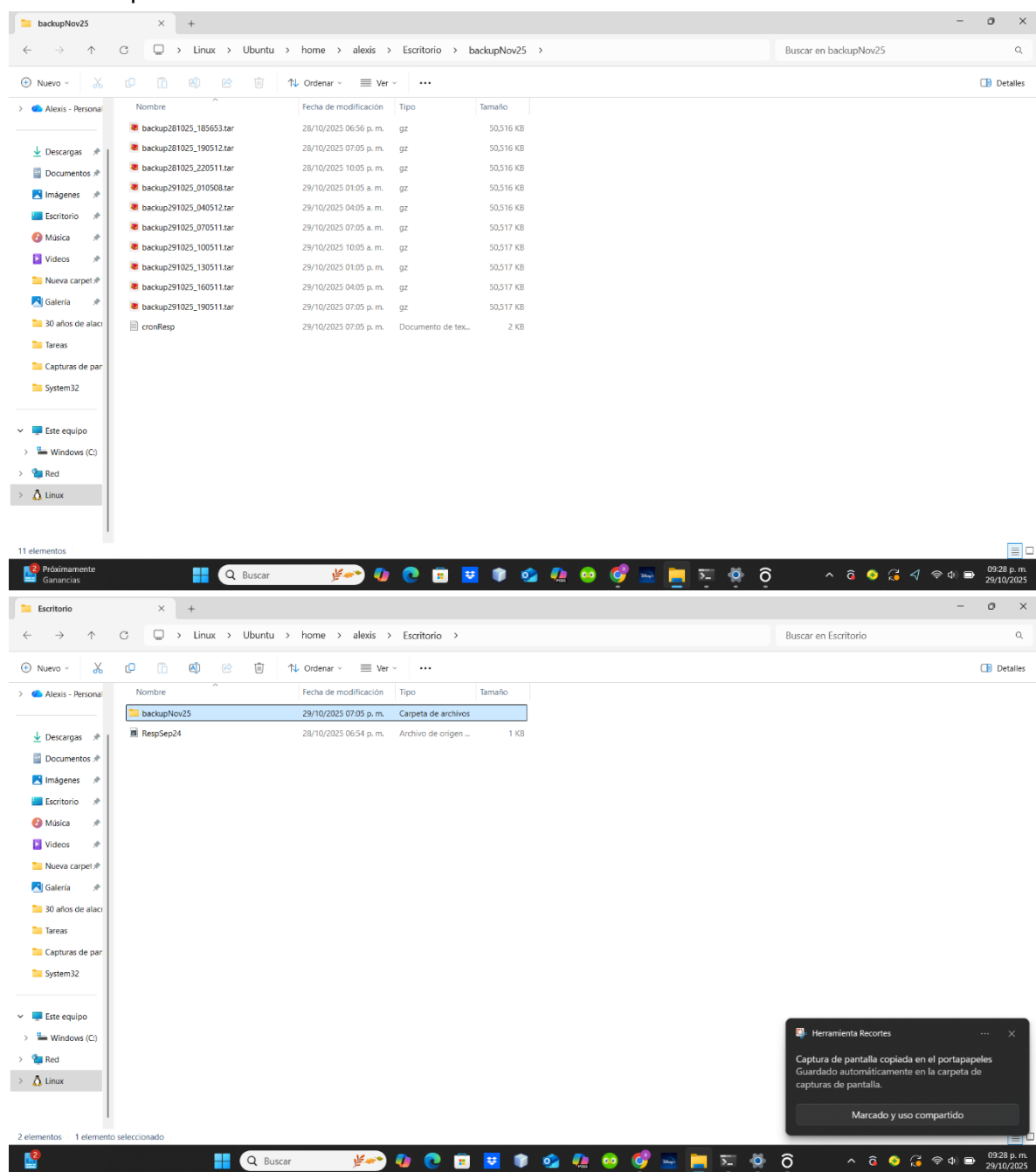
## 6-con comando ls -lh vemos que si estén creando los respaldos cada 3 horas

```
alexis@LAPTOP-9EVLNCK: $ ls -lh ~/Escritorio/backupNov25
total 297M
-rw-r--r-- 1 alexis alexis 50M Oct 28 18:56 backup281025_185653.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 28 19:05 backup281025_190512.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 28 22:05 backup281025_220511.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 29 01:05 backup291025_010508.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 29 04:05 backup291025_040512.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 29 07:05 backup291025_070511.tar.gz
-rw-r--r-- 1 alexis alexis 640 Oct 29 07:05 cronResp.log
alexis@LAPTOP-9EVLNCK: $ cat ~/Escritorio/backupNov25/cronResp.log
tar: Removing leading '/' from member names
✓ Respaldo creado: /home/alexis/Escritorio/backupNov25/backup281025_190512.tar.gz
tar: Removing leading '/' from member names
✓ Respaldo creado: /home/alexis/Escritorio/backupNov25/backup281025_220511.tar.gz
tar: Removing leading '/' from member names
✓ Respaldo creado: /home/alexis/Escritorio/backupNov25/backup291025_010508.tar.gz
tar: Removing leading '/' from member names
✓ Respaldo creado: /home/alexis/Escritorio/backupNov25/backup291025_040512.tar.gz
tar: Removing leading '/' from member names
✓ Respaldo creado: /home/alexis/Escritorio/backupNov25/backup291025_070511.tar.gz
alexis@LAPTOP-9EVLNCK: $
```

## 7-Al final ejecutan el comando nuevamente y veremos que si se ejecutaron los comandos

```
alexis@LAPTOP-9EVLNCK: $ ls -lh ~/Escritorio/backupNov25
total 494M
-rw-r--r-- 1 alexis alexis 50M Oct 28 18:56 backup281025_185653.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 28 19:05 backup281025_190512.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 28 22:05 backup281025_220511.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 29 01:05 backup291025_010508.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 29 04:05 backup291025_040512.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 29 07:05 backup291025_070511.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 29 10:05 backup291025_100511.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 29 13:05 backup291025_130511.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 29 16:05 backup291025_160511.tar.gz
-rw-r--r-- 1 alexis alexis 50M Oct 29 19:05 backup291025_190511.tar.gz
-rw-r--r-- 1 alexis alexis 1.2K Oct 29 19:05 cronResp.log
alexis@LAPTOP-9EVLNCK: $
```

## 8-Vemos que si se crearon en la ruta dada



El respaldo programado se realizó correctamente cada 3 horas con 5 minutos, cumpliendo con la programación solicitada.

El cron ejecutó el script RespSep24.sh de manera automática, generando respaldos comprimidos en la carpeta backupNov25.

Se verificó que la laptop permaneciera encendida con la tapa cerrada, permitiendo la ejecución continua durante 24 horas.

Los resultados confirman el correcto funcionamiento del proceso automatizado de respaldo en Linux utilizando **Shell Script + Crontab**.