

## Manual de Uso: Script de Respaldo Automático

### Descripción

Este script en Bash permite comprimir un directorio específico y guardar el respaldo en una ubicación deseada. Además, puede ser programado con cron para ejecutarse automáticamente en horarios definidos.

### Requisitos

1. Un sistema Linux con Bash instalado.
2. Permisos suficientes para acceder a los directorios de origen y destino.
3. El comando tar, que generalmente ya está disponible en la mayoría de distribuciones Linux.

### Uso del Script

#### 1. Configuración y Ejecución Manual

1. Guarda el script en un archivo llamado backup\_script.sh.
2. En el script debes incluir:
  - El directorio que deseas respaldar.

La ruta donde se guardará el respaldo.
3. Guardar y despues otórgale permisos de ejecución desde la terminal:

```
chmod +x backup_script.sh
```

4. Ejecútalo con el siguiente comando:

```
./backup_script.sh
```

5. Una vez finalizado, el script creará un archivo comprimido en la ubicación indicada.

### Configuración de cron para Ejecución Automática

#### 1. Abrir el crontab

Ejecuta el siguiente comando en la terminal:

```
crontab -e
```

Esto abrirá el editor de tareas programadas de cron.

#### 2. Agregar la tarea programada

Añade la siguiente línea al final del archivo:

```
0 8,18 * * * /ruta/del/script/backup_script.sh <<EOF
```

Explicación:

- 0 8,18 \* \* \* → Ejecuta el script todos los días a las 8:00 AM y 6:00 PM.
- <<EOF ... EOF → Simula la entrada de usuario para evitar la interacción manual.

### 3. Guardar y salir

Si usaste nano, presiona CTRL+X, luego Y y Enter.

### 4. Verificar la configuración

Para confirmar que la tarea se guardó correctamente, ejecuta:

```
crontab -l
```

Esto mostrará las tareas programadas en cron.

### Verificación de Respaldo

Para confirmar que los respaldos se están generando correctamente, puedes listar los archivos en el directorio de destino con:

```
ls -lh /ruta/del/directorio_de_respaldo
```

También puedes revisar el tamaño y contenido del archivo comprimido usando:

```
tar -tf /ruta/del/directorio_de_respaldo/backup_fecha.tar.gz
```

### Notas Adicionales

- Asegúrate de que el directorio de destino tenga suficiente espacio disponible.
- Puedes modificar el tiempo de ejecución en cron ajustando la línea 0 8,18 \* \* \* con otro horario.

**Script:**

```
# Definir variables
SOURCE_DIR="/home/paula/Digitales" # Cambia esto por el directorio a respaldar
BACKUP_DIR="/home/paula"
TIMESTAMP=$(date +"%Y-%m-%d_%H-%M-%S")
BACKUP_FILE="$BACKUP_DIR/backup_${TIMESTAMP}.tar.gz"

# Verificar si el directorio de backups existe, si no, crearlo
if [[ ! -d "$BACKUP_DIR" ]]; then
    mkdir -p "$BACKUP_DIR"
fi

# Crear el respaldo
sudo tar -czf "$BACKUP_FILE" "$SOURCE_DIR"

# Mensaje de éxito
echo "Respaldo creado: $BACKUP_FILE"
```

```
paula@paula-pc:~/Digitales$ ./backup_manager.sh
Ingrese el directorio que desea respaldar: /home/paula/Digitales
Ingrese el directorio de destino del respaldo: /home/paula/Backups
tar: Removing leading '/' from member names
Respaldo creado: /home/paula/Backups/backup_2025-03-16_16-32-59.tar.gz
paula@paula-pc:~/Digitales$ ./backup_manager.sh
tar: Removing leading '/' from member names
Respaldo creado: /home/paula/Backups/backup_2025-03-16_16-38-55.tar.gz
paula@paula-pc:~/Digitales$ crontab -e
no crontab for paula - using an empty one
crontab: installing new crontab
paula@paula-pc:~/Digitales$ crontab -l
0 8,18 * * * /home/paula/Proyecto1erCorte/backup_manager.sh<<EOF
```

```
#
```