EBENEZER IT · GUÍAS PASO A PASO

GUÍA 13 · EPISODIOS 30, 31 Y 32

Copias, restauración y mantenimiento

Una copia cada noche de todo lo que no se puede volver a instalar, cifrada y fuera del servidor. Una prueba de restauración de verdad, sin tocar nada de lo que funciona. Y la lista de lo que hay que revisar una vez al mes para que todo siga así.

~3 horas
la primera vez, en tres sesiones

Requisito
Guías 01 a 03; el resto según lo que tengas

Ubuntu 24.04 LTS
rclone · Google Drive · Docker

www.ebenezer.org.es · Alejandro Quevedo · Versión pública 1.0 · noviembre 2026

Índice

Antes de empezar

0 Qué vas a construir y valores de esta guía

Parte A · La copia de cada noche

Paso 01 Qué copiar y qué no

Paso 02 rclone y Google Drive, cifrado

Paso 03 El script de copia

Paso 04 La primera copia

Paso 05 Cada noche, y que te avise si falla

Parte B · Restaurar: probarlo de verdad

Paso 06 Bajar y descifrar una copia

Paso 07 Las claves y la configuración

Paso 08 La base de datos, en una MariaDB de prueba

Paso 09 Correos, chat y webmail

Paso 10 Limpiar

Paso 11 Si un día pierdes el servidor

Parte C · El mantenimiento mensual

Paso 12 Actualizaciones del sistema

Paso 13 Contenedores y CrowdSec

Paso 14 La revisión del mes

Paso 15 Comprobación final e instantánea

Anexos

A Problemas frecuentes

B El script completo

C La lista del mes

Cómo usar esta guía

Tienes aquí **todos los pasos y comandos**, en orden. La explicación de cada paso (el porqué de cada línea) está en el **vídeo del episodio**.

Para copiar los comandos sin errores, usa la versión web: **https://www.ebenezer.org.es/guia-13-copias.html**. Al copiar desde un PDF se pueden romper los saltos de línea y las comillas.

ANTES DE EMPEZAR

Qué vas a construir

Las instantáneas de IONOS te salvan de un error tuyo, pero viven en el mismo proveedor que el servidor. Una copia de verdad está **en otro sitio**, **cifrada**, se hace **sola**, te **avisa si falla** y, sobre todo, **se ha probado a restaurar**.

Pieza	Qué hace
El script	Cada noche a las 03:00: bases de datos, configuración, correos, webs, volúmenes de Docker... en `/home/TU_USUARIO/backups`, 7 días
rclone	Sube cada copia a Google Drive cifrada con tus contraseñas: Google solo ve archivos ilegibles. Guarda 14 días
El aviso	Si algún paso falla, te llega un correo con el error
La prueba	Bajar una copia, descifrarla y comprobar que la base de datos se carga y las claves están: en un contenedor aparte, sin tocar nada
El mes	Actualizaciones, contenedores, avisos y espacio: media hora al mes

Valores de esta guía

Valor	Qué es
TU_USUARIO	Tu usuario del servidor (Guía 01)
TU_BUZON	Tu correo, donde llegan los avisos

Las dos contraseñas del cifrado, fuera del servidor

rclone cifra con dos contraseñas que se guardan en el propio servidor. Si pierdes el servidor y no las tienes apuntadas en otro sitio (un gestor de contraseñas), **las copias de Drive no se pueden abrir**. Es lo primero que hay que apuntar en el paso 2.

PARTE A

La copia de cada noche

Qué merece la pena copiar, cómo sacarlo del servidor cifrado y el script que lo hace solo. Corresponde al **episodio 30**.

Paso 1 Episodio 30

Qué copiar y qué no

Por qué: copiar «todo el disco» parece lo seguro, pero sube gigas cada noche, tarda y mezcla lo importante con lo que se regenera solo.

Copiar	Por qué
Bases de datos (MariaDB, Postgres de Chatwoot, SQLite de Roundcube)	Buzones, alias, contactos, WordPress, conversaciones
Correos (`/var/mail/vhosts`)	Los mensajes
`/etc` de cada servicio	Toda tu configuración, y en `/etc/opendkim` **las claves DKIM**: sin ellas habría que crear otras y cambiar el DNS
Redis	Lo que Rspamd ha aprendido de tu spam
La base de datos de ejabberd	Cuentas, contactos y mensajes del chat
Volúmenes de Docker	Datos de Grafana, Uptime Kuma, wg-easy…
`/usr/local/bin` , `/opt`	Tus scripts y tus compose, con sus `.env`

No copiar	Por qué
Las métricas de Prometheus	Se regeneran, y pueden pesar más que todo lo demás junto
La carpeta de datos de Postgres o MariaDB «en caliente»	Copiar sus archivos mientras funcionan da copias que pueden no abrir. Se copian con su herramienta (`mysqldump` , `pg_dump`)
Los programas (`/usr` , las imágenes de Docker)	Se vuelven a instalar

Comprueba antes de seguir

Sabes qué se copia, qué no y por qué.

Paso 2 Episodio 30

rclone y Google Drive, cifrado

Por qué: rclone copia carpetas a casi cualquier nube.

EN EL SERVIDOR

```
sudo apt install rclone
sudo rclone config
```

2.1 · El remoto de Google Drive

1. `n` (nuevo remoto) → nombre `gdrive` → tipo **drive** (Google Drive).
2. `client_id` y `client_secret`: en blanco. `scope`: **1** (acceso completo). Lo demás, por defecto.
3. Cuando pregunte *Use web browser to automatically authenticate?*, responde **n**: el servidor no tiene navegador.
4. Te dará una orden `rclone authorize "drive" ...`. Ejecútala en tu PC (con rclone para Windows), inicia sesión con tu cuenta de Google en el navegador que se abre y copia el resultado de vuelta en el servidor.
5. *Configure this as a Shared Drive?*: **n**. Confirma con **y**.

2.2 · El remoto cifrado

1. `n` → nombre `gdrive-cifrado` → tipo **crypt**.
2. `remote`: `gdrive:servidor-cifrado` (una carpeta de tu Drive, donde irá todo cifrado).
3. `filename_encryption`: **standard** (también cifra los nombres de archivo). `directory_name_encryption`: **true**.
4. **Dos contraseñas**: elige **y** (escribir tu propia) y pon una contraseña larga; para la segunda (*salt*), otra distinta.
5. **Apunta las dos en tu gestor de contraseñas antes de seguir.** Sin ellas, las copias no se pueden descifrar en ningún otro equipo.

EN EL SERVIDOR

```
sudo rclone lsd gdrive:
echo "prueba de cifrado" | sudo rclone rcat gdrive-cifrado:prueba.txt
sudo rclone cat gdrive-cifrado:prueba.txt
sudo rclone ls gdrive:servidor-cifrado
sudo rclone delete gdrive-cifrado:prueba.txt
```

Comprueba antes de seguir

Lees el archivo de prueba a través de `gdrive-cifrado`, en Drive aparece con un nombre ilegible, y las dos contraseñas están en tu gestor.

Paso 3 Episodio 30

El script de copia

Por qué: un solo script, con una sección por cada cosa que copia.

EN EL SERVIDOR

```
curl -fsSO https://www.ebenezer.org.es/descargas/backup_total.sh
curl -fsSO https://www.ebenezer.org.es/descargas/SHA256SUMS.txt
sha256sum -c --ignore-missing SHA256SUMS.txt
install -m 700 backup_total.sh ~/backup_total.sh
rm backup_total.sh SHA256SUMS.txt
nano ~/backup_total.sh
```

En `nano`, cambia `TU_USUARIO` en `BASE=` y revisa las listas de carpetas de las secciones 2 y 4: quita lo que no tengas y añade lo tuyo. El script completo está en el Anexo B. Lo importante de su diseño:

Comprueba antes de seguir

El script está en `~/backup_total.sh` con tu usuario y tus carpetas, y `bash -n ~/backup_total.sh` no da errores.

Paso 4 Episodio 30

La primera copia

Por qué: a mano, para ver que todo sale bien antes de dejarla programada.

EN EL SERVIDOR

```
sudo bash ~/backup_total.sh > /tmp/copia-prueba.txt 2>&1; echo "Código: $?"
grep -E "OK ⊣ERROR ⊣Total:|COPIA" /tmp/copia-prueba.txt
sudo rclone lsf gdrive-cifrado:servidor-backups --dirs-only
```

Comprueba antes de seguir

`Código: 0`, ningún `ERROR`, `COPIA COMPLETADA SIN ERRORES` y la carpeta de hoy en Drive.

Paso 5 Episodio 30

Cada noche, y que te avise si falla

Por qué: en la tabla de tareas de root.

EN EL SERVIDOR

```
sudo crontab -e
```

CONTENIDO DEL ARCHIVO · LÍNEA PARA LA TABLA DE ROOT

```
0 3 * * * /bin/bash /home/TU_USUARIO/backup_total.sh > /home/TU_USUARIO/backups/last_log.txt
2>&1 || (echo "Subject: [Copia] Errores en la copia de $(hostname)"; echo; grep -E
"ERROR|COPIA" /home/TU_USUARIO/backups/last_log.txt) | /usr/sbin/sendmail TU_BUZON
```

Prueba el aviso sin esperar a un fallo:

EN EL SERVIDOR

```
false || (echo "Subject: [Copia] Prueba de aviso"; echo; echo "Los avisos de la copia
funcionan.") | /usr/sbin/sendmail TU_BUZON
```

Comprueba antes de seguir

Te llega el correo «[Copia] Prueba de aviso», y mañana existe `~/backups/backup_log_FECHA.txt` con `SIN ERRORES`.

PARTE B

Restaurar: probarlo de verdad

Una copia que nunca se ha restaurado es una suposición. Bajamos la de hoy de Drive y comprobamos que de verdad se puede recuperar todo, en una carpeta y un contenedor aparte: **nada de lo que funciona se toca**. Corresponde al **episodio 31**.

Paso 6 Episodio 31

Bajar y descifrar una copia

Por qué: lo mismo que harías desde un servidor nuevo: pedirle a rclone la copia y que la descifre por el camino.

EN EL SERVIDOR

```
D=/root/prueba-restauracion
sudo mkdir -p $D
sudo rclone lsf gdrive-cifrado:servidor-backups --dirs-only | tail -n 3
sudo rclone copy gdrive-cifrado:servidor-backups/FECHA_DE_LA_COPIA $D -P
sudo ls -lh $D
```

Comprueba antes de seguir

Están todos los archivos de la copia, con el mismo tamaño que en `~/backups`.

Paso 7 Episodio 31

Las claves y la configuración

Por qué: las claves DKIM son lo más difícil de rehacer: comprobamos que las de la copia son idénticas a las que usa el servidor.

EN EL SERVIDOR

```
sudo tar -xzf $D/configs_sistema.tar.gz -C $D etc/opendkim
sudo sh -c "cd $D && find etc/opendkim -name '*.private' -exec sha256sum {} \;"
sudo sh -c "cd / && find etc/opendkim -name '*.private' -exec sha256sum {} \;"
sudo tar -tzf $D/configs_sistema.tar.gz | grep -c ""
```

Comprueba antes de seguir

Las huellas coinciden.

Paso 8 Episodio 31

La base de datos, en una MariaDB de prueba

Por qué: el único modo de saber si una copia de una base de datos sirve es cargarla.

EN EL SERVIDOR

```
sudo docker run -d --name mariadb-prueba --network none -e MARIADB_ROOT_PASSWORD=prueba-temporal mariadb:10.11
sleep 20
sudo zcat $D/db_full_dump.sql.gz | sudo docker exec -i mariadb-prueba mariadb -uroot -pprueba-temporal; echo "Importación: $?"
sudo docker exec mariadb-prueba mariadb -uroot -pprueba-temporal -e "SELECT table_schema AS base, COUNT(*) AS tablas FROM information_schema.tables WHERE table_schema NOT IN ('mysql','sys','information_schema','performance_schema') GROUP BY table_schema;"
sudo docker exec mariadb-prueba mariadb -uroot -pprueba-temporal -e "SELECT COUNT(*) AS buzones FROM TU_BASE_CORREO.mailbox;"
```

Comprueba antes de seguir

`Importación: 0`, están todas tus bases y el número de buzones coincide.

Paso 9 Episodio 31

Correos, chat y webmail

Por qué: los tres que faltan, sin extraer nada: basta con mirar dentro de cada archivo.

EN EL SERVIDOR

```
sudo tar -tzf $D/emails.tar.gz | grep -c -E '/(cur|new)/[^/]+$'
sudo tar -tzf $D/ejabberd_data.tar.gz | grep -c -E 'copia-ejabberd.backup|\.DCD$'
sudo ls -lh $D/roundcube.sqlite
```

Comprueba antes de seguir

Cientos o miles de correos, los archivos del chat y la base de Roundcube.

Paso 10 Episodio 31

Limpiar

Por qué: se borra todo lo de la prueba.

EN EL SERVIDOR

```
sudo docker rm -f mariadb-prueba
sudo docker image rm mariadb:10.11
sudo rm -rf /root/prueba-restauracion
```

Cada tres meses

Repite la parte B cada tres meses, y siempre que cambies algo grande en el script. Tarda veinte minutos y es la única forma de saber que tus copias sirven.

Comprueba antes de seguir

No queda ni el contenedor ni la carpeta de la prueba.

Paso 11 Episodio 31

Si un día pierdes el servidor

Por qué: con las copias probadas, rehacer el servidor es seguir las guías en orden y, en cada servicio, poner tus datos en lugar de empezar de cero.

1. **Servidor base** (Guía 01) con el mismo dominio. Instala rclone, crea los dos remotos con **las mismas contraseñas** de tu gestor y baja la última copia (paso 6).
2. **Cada servicio**: instálalo como en su guía y, antes de arrancarlo, copia encima su configuración desde `configs_sistema.tar.gz` (solo su carpeta, con `tar -xzf ... -C / etc/SERVICIO`).
3. **Bases de datos**: `zcat db_full_dump.sql.gz | sudo mariadb` en la MariaDB nueva.
4. **Correos**: `sudo tar -xzf emails.tar.gz -C /` y `sudo chown -R vmail:vmail /var/mail/vhosts`.
5. **Chat**: con ejabberd parado, la carpeta de `ejabberd_data.tar.gz`; o, con ejabberd en marcha, `sudo ejabberdctl restore /var/lib/ejabberd/copia-ejabberd.backup`.
6. **Docker**: con los contenedores parados, los volúmenes de `docker_data.tar.gz` en `/var/lib/docker/volumes`, y después `docker compose up -d`.
7. **Certificados**: se pueden pedir de nuevo con Certbot; o restaurar `/etc/letsencrypt` entero.
8. **DNS**: si la IP del servidor nuevo cambia, actualiza los registros A y el PTR. Las claves DKIM son las mismas: el registro TXT no cambia.

Comprueba antes de seguir

Sabes en qué orden se restaura y dónde está cada cosa en la copia.

PARTE C

El mantenimiento mensual

Media hora al mes, siempre el mismo día, con una instantánea antes. Corresponde al **episodio 32**.

Paso 12 Episodio 32

Actualizaciones del sistema

Por qué: Ubuntu instala solo los parches de seguridad cada día (`unattended-upgrades`).

EN EL SERVIDOR

```
systemctl is-active unattended-upgrades
sudo apt update
apt list --upgradable 2>/dev/null | tail -n +2
sudo apt upgrade
ls /var/run/reboot-required 2>/dev/null && cat /var/run/reboot-required.pkgs
```

Cuando apt pregunte qué servicios reiniciar

`needrestart` enseña una lista de servicios que usan bibliotecas actualizadas. **No marques** `docker` : reiniciarlo para todos los contenedores a la vez, la VPN incluida, y si estás conectado por ella te quedas fuera. Docker se reinicia solo con el reinicio del servidor.

Si hace falta reiniciar: `sudo reboot` , y al volver, el chequeo (`sudo /usr/local/bin/chequeo-servidor.sh`) y una mirada a Uptime Kuma.

Con Ubuntu Pro

Si tienes Ubuntu Pro (gratis para uso personal), `sudo pro status` debe mostrar `esm-apps` , `esm-infra` y `livepatch` activos. Livepatch parchea el núcleo en caliente: muchos meses no hará falta reiniciar.

Comprueba antes de seguir

`apt upgrade` sin nada pendiente y, si hizo falta, el servidor reiniciado y el chequeo en orden.

Paso 13 Episodio 32

Contenedores y CrowdSec

Por qué: las imágenes de Docker no se actualizan con `apt` .

EN EL SERVIDOR

```
docker ps --format "table {{.Names}}\t{{.Image}}\t{{.Status}}"
sudo cscli hub update && sudo cscli hub upgrade && sudo systemctl reload crowdsec
```

Para actualizar una imagen: lee las notas de la versión, cambia la etiqueta en su compose (o en el `.env`), y después `sudo docker compose pull NOMBRE && sudo docker compose up -d NOMBRE` . Si algo falla, vuelves a poner la etiqueta anterior y repites.

Comprueba antes de seguir

Sabes qué versión corre cada contenedor y has decidido cuáles actualizar.

Paso 14 Episodio 32

La revisión del mes

Por qué: lo que no avisa solo, pero conviene mirar.

Mirar	Cómo	Qué buscar
Las copias	`grep -h COPIA ~/backups/backup_log_*.txt`	Todas `SIN ERRORES`
El espacio	`df -h /` y `sudo du -sh /var/log /var/lib/docker`	Menos del 80 %, y nada que crezca sin control
Uptime Kuma	El panel	Caídas que no recuerdes y certificados próximos a caducar
Grafana	El panel del servidor, último mes	Memoria o disco con tendencia a subir
fail2ban y CrowdSec	`sudo fail2ban-client status` · `sudo cscli metrics` `show decisions`	Que bloqueen; un salto grande puede ser un ataque nuevo
Usuarios y claves	`sudo ejabberdctl registered_users TU_DOMINIO`, PostfixAdmin, `~/.ssh/authorized_keys`	Que no haya cuentas ni claves que ya no deberían estar
La restauración	Parte B	Cada tres meses

Comprueba antes de seguir

Has repasado la lista (Anexo C) y anotado lo que haya que hacer.

Paso 15 Episodio 32

Comprobación final e instantánea

Por qué: el estado al terminar esta guía.

EN EL SERVIDOR

```
sudo crontab -l | grep backup_total
grep -h "COPIA" ~/backups/backup_log_*.txt | tail -n 3
sudo rclone lsf gdrive-cifrado:servidor-backups --dirs-only | tail -n 3
```

Comprobación	Resultado correcto
Tarea programada	La línea del paso 5, con el aviso por correo
Últimas copias	`SIN ERRORES`
Drive	Las copias de los últimos días
Restauración	Probada (parte B), y con fecha para repetirla

Comprobación	Resultado correcto
Contraseñas del cifrado	En tu gestor, fuera del servidor

En IONOS, la instantánea `13-copias`.

Comprueba antes de seguir

Si mañana pierdes el servidor, tienes una copia de anoche, cifrada, fuera de él, y sabes que se puede restaurar porque lo has hecho.

ANEXOS

Para cuando algo no sale

Lo primero, siempre: `grep -n ERROR ~/backups/backup_log_$(date +%F).txt` y las líneas de alrededor.

Anexo A · Problemas frecuentes

Lo que ves	Qué pasa	Qué hacer
`ERROR - MariaDB`	`mysqldump` no puede entrar	Como root, MariaDB entra por *socket* sin contraseña; si cambiaste eso, pon el usuario en `/root/.my.cnf` con permisos 600
`ERROR - Redis`	Redis pide contraseña o no está	`redis-cli -h 127.0.0.1 ping` debe decir `PONG`
`ERROR - Subida a Google Drive`	El acceso a Drive caducó o se revocó	`sudo rclone config reconnect gdrive:` (con `rclone authorize` en tu PC, como en el paso 2)
`ERROR - Copia de Chatwoot de hoy`	No se hizo su copia de las 02:45	`systemctl status copia-chatwoot.service`
Las copias pesan cada vez más	Algo crece: un volumen, los correos, los registros	Compara los tamaños de `ls -lh ~/backups/*/` de varios días
`rclone` no descifra en otro equipo	Contraseñas distintas	Las dos (y en el mismo orden) que apuntaste en el paso 2
No llega el aviso de error	Postfix o la línea de cron	Repite la prueba del paso 5 y mira `/var/log/mail.log`

Anexo B · El script completo

El mismo que se descarga en el paso 3. Cambia `TU_USUARIO` y adapta las listas de carpetas a tu servidor.

CONTENIDO DEL ARCHIVO · ~/BACKUP_TOTAL.SH

```
#!/bin/bash
#==============================================================================
# backup_total.sh — Copia nocturna del servidor (Ebenezer IT · Guía 13)
#------------------------------------------------------------------------------
# Se ejecuta como root desde su crontab (03:00). Hace una copia local en
# /home/TU_USUARIO/backups/FECHA, guarda 7 días en el servidor y sube todo cifrado a
# Google Drive (rclone, remoto gdrive-cifrado), donde guarda 14 días.
#
# Código de salida: 0 = todo bien | 1 = algún paso falló (mira el registro)
# Versión 2.1 · noviembre de 2026
#==============================================================================
set -uo pipefail     # -u: error si se usa una variable sin definir
                     # pipefail: una tubería falla si falla cualquiera de sus partes
umask 077            # todo lo que se crea, solo legible por root

# --- CONFIGURACIÓN ---
FECHA=$(date +%Y-%m-%d_%H%M)
BASE="/home/TU_USUARIO/backups"
DIR_BACKUP="$BASE/$FECHA"
LOG_FILE="$BASE/backup_log_$(date +%F).txt"
REMOTO="gdrive-cifrado:servidor-backups"
DIAS_LOCAL=7
DIAS_REMOTO=14
ERRORES=0

mkdir -p "$DIR_BACKUP"

# --- FUNCIONES ---
# log: escribe en pantalla y en el registro
log() { echo "$*" | tee -a "$LOG_FILE"; }

# paso: ejecuta un comando; si falla, lo anota y suma un error (pero sigue)
paso() {
    local desc="$1"; shift
    if "$@" >> "$LOG_FILE" 2>&1; then
        log "      OK - $desc"
    else
        log "      ERROR - $desc"
        ERRORES=$((ERRORES + 1))
    fi
}

# empaqueta: tar.gz tolerante con archivos que cambian o no existen
#   (código 1 de tar = "algún archivo cambió mientras se leía": es normal en
#    registros y buzones vivos; código 2 = error de verdad)
empaqueta() {
    local salida="$1"; shift
    tar -czf "$DIR_BACKUP/$salida" --ignore-failed-read --warning=no-file-changed "$@"
    [ $? -le 1 ]
```

```
}

tamano() { du -h "$DIR_BACKUP/$1" 2>/dev/null | cut -f1; }

log "═══ INICIO COPIA: $FECHA ═══"

# --- 1. BASES DE DATOS (correo, WordPress, Roundcube...) ---
log "[1/10] MariaDB..."
# --single-transaction: una foto coherente de las tablas InnoDB sin bloquearlas
# --quick: lee fila a fila, sin cargar tablas enteras en memoria
paso "MariaDB" bash -c "set -o pipefail; mysqldump --all-databases --single-transaction --quick
--routines --events --triggers | gzip > '$DIR_BACKUP/db_full_dump.sql.gz'"
log "      $(tamano db_full_dump.sql.gz)"

# --- 2. CONFIGURACIONES DEL SISTEMA ---
log "[2/10] Configuraciones..."
paso "Configuraciones" empaqueta configs_sistema.tar.gz \
    /etc/nginx /etc/postfix /etc/dovecot /etc/opendkim /etc/opendkim.conf \
    /etc/rspamd /etc/redis /etc/unbound /etc/mysql /etc/php \
    /etc/ufw /etc/letsencrypt /etc/ssh /etc/fail2ban /etc/crowdsec \
    /etc/ejabberd /etc/apparmor.d/local /etc/systemd/system /etc/rsyslog.d \
    /etc/logwatch /etc/crontab /etc/cron.d /etc/cron.daily /etc/cron.hourly \
    /etc/hosts /etc/hostname /etc/resolv.conf /etc/aliases /etc/mailname \
    /usr/local/bin /opt/monitoring
log "      $(tamano configs_sistema.tar.gz)"

# --- 3. REDIS (lo que Rspamd ha aprendido del spam) ---
log "[3/10] Redis..."
# SAVE vuelca a disco lo que Redis tiene en memoria; después se copia el archivo
if command -v redis-cli > /dev/null; then
    paso "Redis" sh -c "redis-cli -h 127.0.0.1 SAVE > /dev/null && cp /var/lib/redis/dump.rdb
'$DIR_BACKUP/redis_dump.rdb'"
    log "      $(tamano redis_dump.rdb)"
else
    log "      (sin Redis)"
fi

# --- 4. ARCHIVOS WEB ---
log "[4/10] Webs..."
paso "Webs" empaqueta web_files.tar.gz \
    /var/www \
    /usr/share/roundcube/plugins /usr/share/roundcube/skins /etc/roundcube
log "      $(tamano web_files.tar.gz)"
# Roundcube guarda contactos, identidades y ajustes en SQLite (no en MariaDB).
# Con el programa sqlite3, ".backup" hace una copia coherente aunque esté en uso.
RC_DB=/var/lib/dbconfig-common/sqlite3/roundcube/roundcube
if [ ! -f "$RC_DB" ]; then
    log "      (Roundcube sin SQLite: su base, si la tiene, va en MariaDB)"
elif command -v sqlite3 > /dev/null; then
    paso "Roundcube (SQLite)" sqlite3 "$RC_DB" ".backup '$DIR_BACKUP/roundcube.sqlite'"
else
    paso "Roundcube (SQLite, copia simple)" cp "$RC_DB" "$DIR_BACKUP/roundcube.sqlite"
fi

# --- 5. CORREOS ---
log "[5/10] Correos..."
paso "Correos" empaqueta emails.tar.gz /var/mail/vhosts
```

```
log "      $(tamano emails.tar.gz)"

# --- 6. DOCKER ---
log "[6/10] Docker..."
# Fuera: las métricas de Prometheus (se regeneran y pesan mucho) y la carpeta de
# Postgres de Chatwoot en caliente (su copia buena la hace pg_dump: Guía 04)
paso "Volúmenes de Docker" empaqueta docker_data.tar.gz \
    --exclude='*/monitoring_prometheus-data' \
    --exclude='*/chatwoot_postgres_data' \
    /var/lib/docker/volumes /etc/docker
log "      $(tamano docker_data.tar.gz)"
paso "Archivos docker-compose" bash -c "find /home /opt /root \( -name 'docker-compose*.yml' -o
-name 'docker-compose*.yaml' \) -print0 2>/dev/null | tar -czf
'$DIR_BACKUP/docker_compose_files.tar.gz' --null -T -"

# --- 7. EJABBERD (chat) ---
log "[7/10] ejabberd..."
# Copia coherente de su base de datos (Mnesia) antes de empaquetar la carpeta
if [ -x /usr/sbin/ejabberdctl ]; then
    paso "ejabberd: copia de la base de datos" /usr/sbin/ejabberdctl backup
/var/lib/ejabberd/copia-ejabberd.backup
    paso "ejabberd: carpeta" empaqueta ejabberd_data.tar.gz /var/lib/ejabberd /etc/ejabberd
    log "      $(tamano ejabberd_data.tar.gz)"
else
    log "      (sin ejabberd)"
fi

# --- 8. TAREAS PROGRAMADAS DE USUARIOS ---
log "[8/10] Crontabs..."
paso "Crontabs" empaqueta cron_jobs.tar.gz /var/spool/cron/crontabs

# --- 9. CHATWOOT Y /opt/apps ---
log "[9/10] Chatwoot y /opt/apps..."
# Si tienes Chatwoot (Guía 04), su copia de las 02:45 deja bd-, config- y archivos-FECHA
if [ -d /var/backups/chatwoot ]; then
    paso "Copia de Chatwoot de hoy" sh -c "ls /var/backups/chatwoot/*-$(date +%F).* > /dev/null
2>&1 && cp /var/backups/chatwoot/*-$(date +%F).* '$DIR_BACKUP/'"
fi
[ -d /opt/apps ] && paso "/opt/apps" empaqueta opt_apps.tar.gz /opt/apps

# --- 10. ESTE SCRIPT ---
log "[10/10] El propio script..."
paso "Script" cp "$0" "$DIR_BACKUP/backup_total.sh"

log ""
log "═══ ARCHIVOS GENERADOS ═══"
ls -lh "$DIR_BACKUP/" | tee -a "$LOG_FILE"
log "Total: $(du -sh "$DIR_BACKUP" | cut -f1)"

# --- LIMPIEZA LOCAL ---
log ""
log "Borrando copias locales de más de $DIAS_LOCAL días..."
find "$BASE" -mindepth 1 -maxdepth 1 -type d -mtime +"$DIAS_LOCAL" -exec rm -rf {} + 2>>
"$LOG_FILE"
find "$BASE" -maxdepth 1 -name "backup_log_*.txt" -mtime +"$DIAS_REMOTO" -delete 2>>
"$LOG_FILE"
```

```
# --- COPIA A GOOGLE DRIVE (cifrada) ---
log ""
log "Subiendo a Google Drive..."
rclone mkdir "$REMOTO" >> "$LOG_FILE" 2>&1
paso "Subida a Google Drive" rclone copy "$BASE" "$REMOTO" --exclude "last_log.txt"

log "Borrando de Google Drive las copias de más de $DIAS_REMOTO días..."
LIMITE=$(date -d "$DIAS_REMOTO days ago" +%F)
for d in $(rclone lsf "$REMOTO" --dirs-only 2>> "$LOG_FILE"); do
    dia=${d%%_*}
    if [[ "$dia" < "$LIMITE" ]]; then
        log "  Borrando de Drive: $d"
        rclone purge "$REMOTO/$d" --drive-use-trash=false >> "$LOG_FILE" 2>&1
    fi
done
rclone delete "$REMOTO" --max-depth 1 --include "backup_log_*.txt" --min-age "${DIAS_REMOTO}d"
\
    --drive-use-trash=false --tpslimit 2 >> "$LOG_FILE" 2>&1

# --- RESULTADO ---
log ""
if [ "$ERRORES" -eq 0 ]; then
    log "═══ COPIA COMPLETADA SIN ERRORES: $FECHA ═══"
    exit 0
else
    log "═══ COPIA CON $ERRORES ERROR(ES): $FECHA — revisa las líneas ERROR ═══"
    exit 1
fi
```

Anexo C · La lista del mes

	Tarea	Paso
☐	Instantánea en IONOS antes de empezar	—
☐	`apt update` y `apt upgrade` (sin marcar docker en needrestart)	12
☐	¿Hace falta reiniciar? Reinicio, chequeo y Uptime Kuma	12
☐	Versiones de los contenedores y `cscli hub upgrade`	13
☐	Copias del mes, todas sin errores	14
☐	Espacio en disco	14
☐	Uptime Kuma y Grafana del mes	14
☐	fail2ban y CrowdSec	14
☐	Usuarios, buzones y claves SSH que ya no deberían estar	14
☐	Cada tres meses: restauración de prueba	6–10