**2. Los respaldos deben guardarse en una carpeta compartida entre los contenedores y el host.**

Para guardar los respaldos de MongoDB en una carpeta compartida entre los contenedores y el host, puedes crear un volumen en Docker. A continuación, te muestro cómo hacerlo paso a paso.

**1. Configura un volumen de Docker**

Primero, debes crear una carpeta en tu host que se utilizará para almacenar los respaldos. Asegúrate de que la carpeta tenga los permisos adecuados.

bash

mkdir -p /ruta/al/directorio/de/respaldo

**2. Configura el script de respaldo**

Ajusta el script backup\_mongodb.sh para que guarde los respaldos en la carpeta compartida. Aquí te dejo un ejemplo del script, asegurándote de que la ruta de salida coincida con el volumen compartido:

bash

#!/bin/bash

# Variables de configuración

MONGO\_HOST="mongodb" # Usa el nombre del contenedor si se está ejecutando en Docker

MONGO\_PORT="27017" # Cambia si es diferente

MONGO\_USER="tu\_usuario" # Usuario de MongoDB si es necesario

MONGO\_PASSWORD="tu\_contraseña" # Contraseña de MongoDB si es necesario

DBNAME="nombre\_de\_tu\_base" # Nombre de la base de datos

OUTPUT\_DIR="/ruta/al/directorio/de/respaldo" # Ruta en el host donde se guardan los respaldos

TIMESTAMP=$(date +"%Y%m%d\_%H%M%S")

BACKUP\_DIR="${OUTPUT\_DIR}/${DBNAME}\_${TIMESTAMP}"

# Crear el directorio de respaldo

mkdir -p $BACKUP\_DIR

# Realizar el respaldo

if [ -z "$MONGO\_USER" ] || [ -z "$MONGO\_PASSWORD" ]; then

mongodump --host $MONGO\_HOST --port $MONGO\_PORT --db $DBNAME --out $BACKUP\_DIR

else

mongodump --host $MONGO\_HOST --port $MONGO\_PORT --db $DBNAME -u $MONGO\_USER -p $MONGO\_PASSWORD --out $BACKUP\_DIR

# Opcional: Eliminar respaldos más antiguos de X días

find $OUTPUT\_DIR -type d -name "${DBNAME}\_\*" -mtime +7 -exec rm -r {} \;

echo "Respaldo de la base de datos ${DBNAME} realizado en ${BACKUP\_DIR}"

**3. Ejecutar un contenedor de MongoDB con el volumen compartido**

Si estás utilizando Docker, puedes crear un contenedor de MongoDB y montar el volumen compartido. Por ejemplo:

bash

docker run --name mongodb -v /ruta/al/directorio/de/respaldo:/backups -d mongo

En este comando, /ruta/al/directorio/de/respaldo es el directorio en tu host. Este directorio se montará en el contenedor en la ruta /backups.

**4. Programa el script en el contenedor**

Si deseas que el script se ejecute dentro del contenedor, puedes configurarlo usando cron dentro del contenedor de MongoDB o utilizar un contenedor adicional solo para los respaldos.

Si optas por crear otro contenedor para los respaldos, puedes usar un contenedor que tenga acceso a MongoDB:

bash

docker run --name mongo-backup -v /ruta/al/directorio/de/respaldo:/backups --link mongodb:mongo -d your\_backup\_image

Asegúrate de reemplazar your\_backup\_image por tu imagen que contenga el script de respaldo.

**5. Configurar Cron en el contenedor de respaldo**

Dentro del contenedor de respaldo, asegúrate de que cron esté instalado y configura el cron para ejecutar el script backup\_mongodb.sh cada hora, como se describió anteriormente.

Notas finales

Con esto, deberías poder realizar respaldos automáticos de MongoDB y almacenarlos en un directorio compartido entre tus contenedores y el host. Asegúrate de probar el script manualmente antes de depender de

la programación con cron.