

Restoring a Backup

Restoring Redis backups is essential for disaster recovery, staging environment duplication, or rolling back to a known state. Elestio supports backup restoration both through its web dashboard and manually through Docker Compose and command-line methods. This guide explains how to restore Redis backups from RDB or AOF files, covering both full and partial restore scenarios, and includes solutions for common restoration issues.

Restoring from a Backup via Terminal

This method applies when you have an RDB (dump.rdb) or AOF (appendonly.aof) file from a previous backup. To restore the backup, you replace the existing Redis data file(s) inside the data directory used by the container. Redis loads this data at startup, making it essential to stop the server before replacing the files.

Stop the Redis Container

Shut down the Redis container cleanly to avoid file corruption:

```
docker-compose down
```

Replace the Backup File

Move your backup file into the appropriate location inside the Redis volume. Assuming you have a backup named backup_2025_05_19.rdb:

```
cp ./backup_2025_05_19.rdb /opt/app/data/dump.rdb
```

Make sure this file path corresponds to the volume used in your docker-compose.yml. For example:

```
volumes:
  - ./data:/data
```

If you're restoring an AOF file, replace appendonly.aof instead:

```
cp ./appendonly_2025_05_19.aof /opt/app/data/appendonly.aof
```

Restart Redis

Start Redis again so it loads the restored data file:

```
docker-compose up -d
```

Redis will automatically load `dump.rdb` or `appendonly.aof` depending on your configuration (set in `redis.conf` with `appendonly yes/no`).

Restoring via Docker Compose Terminal

If you prefer working inside the container, you can also copy the file directly into the Redis container using Docker commands.

Copy the Backup File into the Container

```
docker cp ./backup_2025_05_19.rdb $(docker-compose ps -q redis):/data/dump.rdb
```

If restoring an AOF file:

```
docker cp ./appendonly_2025_05_19.aof $(docker-compose ps -q redis):/data/appendonly.aof
```

Restart Redis Inside Docker Compose

```
docker-compose restart redis
```

Redis will detect the updated data file and load it during startup.

Partial Restores in Redis

Redis does not natively support partial restores like MySQL. However, you can achieve similar outcomes with the following strategies:

Restore Selected Keys via Redis CLI

If you exported individual key-value pairs using the `redis-cli --rdb` or similar logic, you can use a script to reinsert only those keys.

Example using `redis-cli` and a JSON/CSV conversion:

```
cat keys_to_restore.txt | while read key; do
    value=$(cat dump.json | jq -r ".$key")
    redis-cli SET "$key" "$value"
done
```

This approach assumes you have extracted individual key-value pairs into a format suitable for scripting.

Restore from A Partial AOF

If your append-only file includes only a subset of commands, Redis will replay those on startup. You can prepare a stripped-down AOF file for specific keys or operations, then follow the full AOF restore method described above.

Common Errors & How to Fix Them

Restoring Redis data can fail for a few specific reasons, especially related to permissions, missing config values, or service conflicts. Here are some frequent issues and how to solve them.

1. NOAUTH Authentication Required

```
(error) NOAUTH Authentication required.
```

Cause: You're attempting to issue commands or restore data into a Redis instance that requires authentication.

Resolution: Always provide the password with your commands:

```
redis-cli -a yourpassword
```

For automated scripts, use:

```
redis-cli -a "$REDIS_PASSWORD" < restore_script.txt
```

2. Redis Fails to Start After Restore

```
Fatal error loading the DB: Invalid RDB format
```

Cause: Corrupted or incompatible `dump.rdb` or `appendonly.aof` file.

Resolution: Ensure the backup file matches the Redis version you're using. Try restoring with a version of Redis that matches the backup environment.

3. Data Not Restored

Cause: Redis is configured to use AOF, but only an RDB file was restored or vice versa.

Resolution: Confirm your redis.conf or container command: entry defines which persistence method is enabled:

```
appendonly yes # For AOF
```

```
appendonly no # For RDB
```

Make sure the correct file (either dump.rdb or appendonly.aof) is in /data.

4. Permission Denied When Copying Files

```
cp: cannot create regular file '/opt/app/data/dump.rdb': Permission denied
```

Resolution: Ensure your terminal session or script has write access to the target directory. Use sudo if needed:

```
sudo cp ./backup.rdb /opt/app/data/dump.rdb
```

Revision #1

Created 20 May 2025 09:30:46 by kaiwalya

Updated 20 May 2025 10:08:20 by kaiwalya