

AI-03953 DockerLab Backup Script

👤 Owner	© Rigel Arcayan
☰ Tags	Project
🕒 Created time	June 19, 2026 9:30 PM

CLI to backup my dockerlab, the files are residing in /home/rarcayan/dockerlab . I want a folder backup of the two folders inside name apps and infrastructure.

Here is a concise, production-ready CLI solution to back up your `apps` and `infrastructure` directories.

Since Docker containers often have active file locks or stateful data (like databases), the cleanest approach is to create a compressed archive (`.tar.gz`) while optionally pausing containers if you need absolute data consistency.

The Quick Command

Run this from your terminal to create a timestamped backup file in your home directory:

Bash

```
tar -czvf ~/dockerlab_backup_$(date +%F_%H%M%S).tar.gz \  
-C /home/rarcayan/dockerlab apps infrastructure
```

Breakdown of the Command

- `tar -czvf` : Creates (`c`) a zipped (`z`) archive, showing progress (`v`), and saves it to a specified file (`f`).
- `$(date +%F_%H%M%S)` : Automatically inserts a clean timestamp (e.g., `2026-06-19_212000`) so you never overwrite old backups.

- `C /home/rarcayan/dockerlab`: Changes the execution context to your `dockerlab` directory. This is a critical best practice—it prevents the backup from storing absolute paths (like `/home/rarcayan/...`), making it incredibly easy to extract or restore anywhere else later.

Recommended: The Safe Scripted Approach

If these folders contain live databases (like MySQL or Postgres) or stateful application data, running a backup while Docker is actively writing can occasionally corrupt the backup file.

Here is a quick script to safely pause the containers, run the backup, and resume them immediately:

Bash

```
#!/bin/bash
BACKUP_DIR="/home/rarcayan"
SOURCE_DIR="/home/rarcayan/dockerlab"
TIMESTAMP=$(date +%F_%H%M%S)

echo "Pausing Docker containers..."
docker pause $(docker ps -q) 2>/dev/null

echo "Creating backup..."
tar -czvf "$BACKUP_DIR/dockerlab_backup_${TIMESTAMP}.tar.gz" -C "$SOURCE_DIR" apps infrastr
ucture

echo "Resuming Docker containers..."
docker unpause $(docker ps -q) 2>/dev/null

echo "Backup complete: $BACKUP_DIR/dockerlab_backup_${TIMESTAMP}.tar.gz"
```

To Verify Your Backup

To quickly inspect the contents of the generated `.tar.gz` file without extracting it to make sure both folders are intact, use:

Bash

```
tar -tvf ~/dockerlab_backup_*.tar.gz
```