✅ Final Checklist: Solving Airflow + dbt Permission Issues (Logs & Target)

# 📁 1. Create `logs/` and `target/` folders manually

If they don’t already exist inside your local dbt project directory:

mkdir -p dbt\_project/dbtnewdemo/dbtproject/logs  
mkdir -p dbt\_project/dbtnewdemo/dbtproject/target

✅ This prevents dbt from trying to create them at runtime, which causes permission errors inside Docker containers.

# 🧩 2. Fix Windows File Permissions using PowerShell

Grant full access to `target/` and `logs/` so Docker containers (Linux) can write into them:

icacls "dbt\_project\dbtnewdemo\dbtproject\target" /grant Everyone:F /T  
icacls "dbt\_project\dbtnewdemo\dbtproject\logs" /grant Everyone:F /T

# 📦 3. Mount Volumes in `docker-compose.yml`

Ensure these are mounted in both `airflow` and `airflow-scheduler`:

volumes:  
 - ./dbt\_project/dbtnewdemo/dbtproject:/usr/app/dbt  
 - ./dbt\_project/dbtnewdemo/dbtproject/target:/usr/app/dbt/target  
 - ./dbt\_project/dbtnewdemo/dbtproject/logs:/usr/app/dbt/logs

Also keep the default Airflow mounts:

- ./airflow/dags:/opt/airflow/dags  
 - ./airflow/logs:/opt/airflow/logs  
 - ./airflow/plugins:/opt/airflow/plugins

# 🛠 4. Update the Bash Command in the Airflow DAG

Avoid using the default log path that writes to `/usr/app/dbt/logs/dbt.log`. Instead use:

bash\_command='cd /usr/app/dbt && dbt run --profiles-dir . --log-path /tmp/dbt-logs'

# 🔁 5. Rebuild & Restart Docker Containers

Once everything is set:

docker-compose down  
docker-compose up --build -d

🧠 This ensures new permissions, volumes, and folders are respected by all services.

# 🧪 6. Test DAG Execution in Airflow

Go to Airflow UI: `http://localhost:8080`  
Trigger `dbt\_run\_dag`  
Check logs → You should see `dbt run` working without any PermissionError.