Docker CE Installation on WSL 2... in four steps

domenica 28 aprile 2024 11:01

Yes, you can run Docker containers directly within your WSL 2 (Windows Subsystem for Linux version 2) environment on Windows 11, treating it much like an independent Linux machine. This setup allows you to use Docker natively within Linux without needing Docker Desktop for Windows. Here's how you can set it up:

Step 1: Ensure WSL 2 is Enabled and Installed

First, make sure you have WSL 2 installed and enabled. You can check your WSL version by opening a PowerShell window as administrator and running: `wsl --list --verbose`
If you haven't installed WSL or need to upgrade to WSL 2, you can follow the official Microsoft documentation to do so.

Step 2: Install Your Preferred Linux Distribution

If you haven't already, install a Linux distribution from the Microsoft Store (e.g., Ubuntu, Debian, Fedora, etc.). Once installed, launch it to complete its setup.

Step 3: Install Docker within Your WSL 2 Linux Distribution

1. ### Update and Upgrade Your Linux Distribution

   Open your WSL 2 terminal and run the following commands to update and upgrade your distribution's packages:

   ```
   sudo apt update && sudo apt upgrade -y
   ```

2. ### Install Docker's Dependencies

   Next, install the necessary packages to allow your system to use repositories over HTTPS:

   ```
   sudo apt install apt-transport-https ca-certificates curl software-properties-common -y
   ```

3. ### Add Docker's Official GPG Key and Set up the Stable Repository

   ```
   curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
   sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
   ```

4. ### Install Docker CE (Community Edition)

   Update your package index and install Docker CE:

   ```
   sudo apt update
   sudo apt install docker-ce -y
   ```

5. ### Manage Docker as a Non-root User

   To run Docker commands without sudo, add your user to the Docker group: `sudo usermod -aG docker $USER`
   You'll need to exit your WSL session and start it again or log out and log back in for this to take effect.

6. ### Start Docker on WSL 2

   You might need to start the Docker service manually: `sudo service docker start`
   To have Docker start automatically with your WSL 2 distribution, you can add the above command to your shell's profile script (e.g., .bashrc or .zshrc).

Step 4: Verify Docker Installation

Run a test Docker container to verify your installation: `docker run hello-world`
This should download a test image and run a container, outputting the "Hello from Docker!" message if everything is ok.

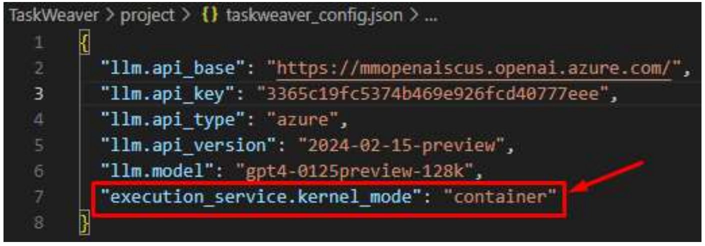
Conclusion

Following these steps, you've installed Docker directly within your WSL 2 environment, enabling you to run Docker containers as if in an independent Linux machine. This setup is particularly useful for development and testing purposes, providing a more Linux-native experience on Windows 11.

Installing TaskWeaver in Docker

Here are the three prerequisites to run TaskWeaver in a Docker container

- Docker is installed on the host machine, as we did here above ;-)
- A specific Docker image must be built and made available on the host machine for code execution.
- We must set "**execution_service.kernel_mode**": "**container**" in taskweaver_config.json



```
{
  "llm.api_base": "https://mmopenaiscus.openai.azure.com/",
  "llm.api_key": "3365c19fc5374b469e926fcd40777eee",
  "llm.api_type": "azure",
  "llm.api_version": "2024-02-15-preview",
  "llm.model": "gpt4-0125preview-128k",
  "execution_service.kernel_mode": "container"
}
```

- Il servizio Docker si può installare anche in una WSL 2 (un windows subsystem for Linux), come ho spiegato passo passo sul mio repo github di OpenAI
- È possibile costruire l'immagine Docker eseguendo il seguente comando nella cartella di root del git clonato:

```
cd scripts

# based on your OS
./build_executor.ps1 # for Windows
./build_executor.sh # for Linux or macOS
```

Note: if you the error

```
-bash: ./build_executor.sh: Permission denied
```

, please check permissions with `ls -l ./build_executor.sh`

If you get something like `-rw-r--r--`

, this means the following things:

- `-rw-`: The owner of the file (mauromi) has read (r) and write (w) permissions, but not execute (x) permission.
- `r--`: Members of the group (also mauromi in this case) have read (r) permission only.
- `r--`: Others (anyone else) have read (r) permission only.

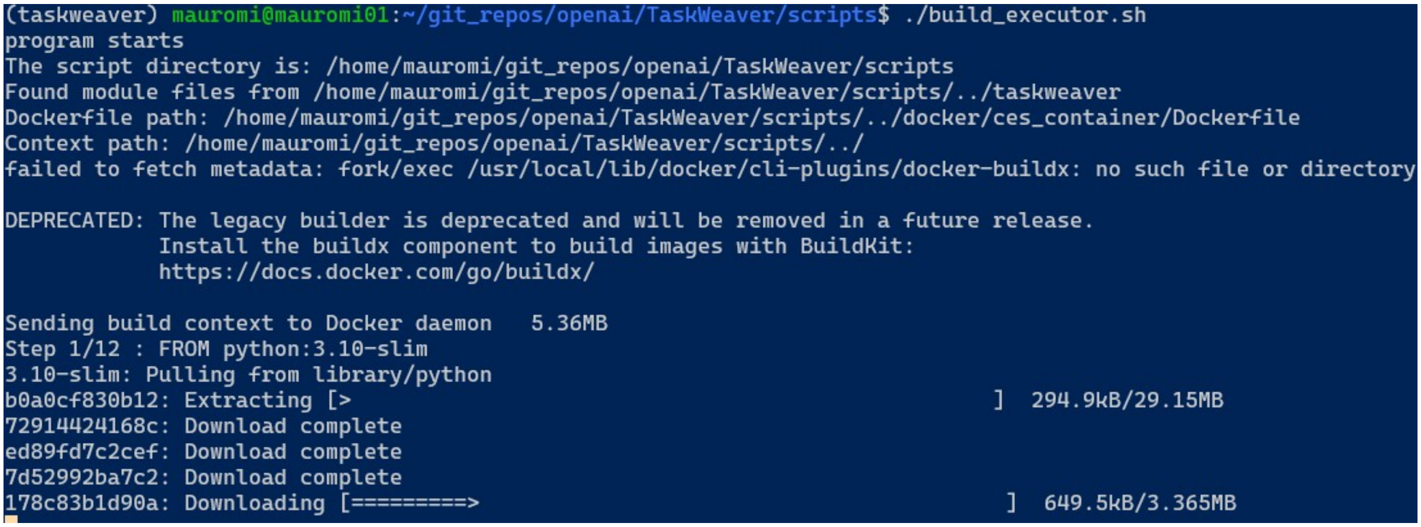
To give the right permissions, run `chmod u+x build_executor.sh`

Then check again the permissions; you should get `-rwXr--r--`

Now run again `./build_executor.sh`. If you get an error like "`...-bash: ./build_executor.sh: /bin/bash^M: bad interpreter: No such file or directory...`", please

- Install Dos2Unix: `sudo apt install dos2unix`
- Run it : `dos2unix build_executor.sh`

This time, the command `./build_executor.sh` will work. Important: recall to use the proper Conda environment, and do NOT use the elevated mode (sudo):

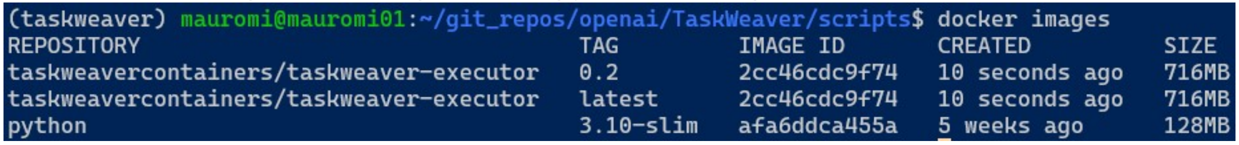


```
(taskweaver) mauromi@mauromi01:~/git_repos/openai/TaskWeaver/scripts$ ./build_executor.sh
program starts
The script directory is: /home/mauromi/git_repos/openai/TaskWeaver/scripts
Found module files from /home/mauromi/git_repos/openai/TaskWeaver/scripts/../taskweaver
Dockerfile path: /home/mauromi/git_repos/openai/TaskWeaver/scripts/../docker/ces_container/Dockerfile
Context path: /home/mauromi/git_repos/openai/TaskWeaver/scripts/../
failed to fetch metadata: fork/exec /usr/local/lib/docker/cli-plugins/docker-buildx: no such file or directory

DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/

Sending build context to Docker daemon   5.36MB
Step 1/12 : FROM python:3.10-slim
3.10-slim: Pulling from library/python
b0a0cf830b12: Extracting [>                                                  ]  294.9kB/29.15MB
72914424168c: Download complete
ed89fd7c2cef: Download complete
7d52992ba7c2: Download complete
178c83b1d90a: Downloading [=========>                                        ]  649.5kB/3.365MB
```

At the end, we should get the following (not running yet!) Docker images:



```
(taskweaver) mauromi@mauromi01:~/git_repos/openai/TaskWeaver/scripts$ docker images
REPOSITORY                              TAG         IMAGE ID       CREATED          SIZE
taskweavercontainers/taskweaver-executor 0.2        2cc46cdc9f74   10 seconds ago   716MB
taskweavercontainers/taskweaver-executor latest     2cc46cdc9f74   10 seconds ago   716MB
python                                  3.10-slim   afa6ddca455a   5 weeks ago      128MB
```