Specify the env file docker compose uses

Asked 8 years, 10 months ago   Modified 4 years, 4 months ago   Viewed 47k times

▲

37

▼

Is it possible to specify the env file that docker compose uses for variable substitution? Currently it defaults to ".env" but I have different flavours of compose files which should have different envs.

docker   docker-compose

🔖

↺   Share Improve this question Follow

asked Nov 10, 2016 at 10:25

Zuriar
11.9k ● 23 ● 64 ● 98

---

2   I found this docs.docker.com/compose/environment-variables but this appears to need env_file for each container.. Is it possible to set it globally for the whole docker-compose file thus overriding the default .env?
– Zuriar Nov 10, 2016 at 10:31

3 Answers

Sorted by: Highest score (default) ⬍

▲

25

▼

🔖

✔

↺

You can use inheritance for this. If you have one "base" service where you set up the environment, all of your other services can inherit from that.

Example:

```
version: "2"

services:
  base:
    env_file:
      - my_env.txt

  web:
    extends:
      service: base
    image: foo

  database:
    extends:
      service: base
    image: foo-db
```

The above example has everything in the same file, but you can also split this up into multiple files, where the `base` service would reside in a `base.yaml` file. You just need to add `file:`

`base.yaml` to the `extends` section. Please see [the documentation here](#).

I use this approach for setting the proxy variables for all containers. I have a `proxy.yaml` file that defines a `proxy-app` service that picks up the proxy environment variables from the shell. All of my real services extend the `proxy-app` service and thus inherit the environment settings from that service.

Share   Improve this answer   Follow

answered Nov 11, 2016 at 9:00

nwinkler
54.9k ● 23 ● 166 ● 170

3 Comments ⌄

Add a comment

Tobi Over a year ago

Attention: Will not work in docker compose v3 and up

▲ 19   ▣ Reply   •••

Luke Davis Over a year ago

This will not work on Docker Compose v3+ because each service must have a defined image or a build context. Documented [here](#) and discussed [here](#)

▲ 3   ▣ Reply   •••

Peter Kionga-Kamau Over a year ago

This does not address the question, which is how to specify a different .env file, but instead provides a workaround, which is not useful for the vast majority of viewers who are looking for a solution to the question as asked for a variety of reasons

▲ 2   ▣ Reply   •••

▲

20

▼

The `--env-file` command-line argument and the `env_file` docker-compose.yml variable specify the env file to use for the container, not for the container build. To set a different file (e.g. *alt.env*) for the build itself, use this:

```
env $(cat alt.env) docker-compose up --build
```

Share   Improve this answer   Follow

answered May 12, 2021 at 0:29

Peter Kionga-Kamau
7,186 ● 2 ● 22 ● 16

5 Comments ⌄

Add a comment

parity3 Over a year ago

I'm not seeing the distinction about --env-file only applying to service deploy and not build, mentioned anywhere in the documentation: docs.docker.com/compose/environment-variables/… This seems like really odd behavior. But after digging some more, the use of the env-file has been broken on v2 for years now: github.com/docker/compose/issues/8515

▲ 3 | 🗩 Reply | ⋯

Peter Kionga-Kamau Over a year ago

Correct, it is not a documented behavior.

▲ 2 | 🗩 Reply | ⋯

Gustav Streicher Over a year ago

@PeterKionga-Kamau This behaviour is not specific to Docker, it's specific to the shell. More specifically it's how the `env` command works. So, if you want to read up on it, then checkout the man pages for the `env` command. You can do that locally by running `man env`.

▲ 0 | 🗩 Reply | ⋯

Peter Kionga-Kamau Over a year ago ✎

@GustavStreicher That is not what I am talking about. I am indicating that there is a distinction between *build* and *runtime* environments

▲ 1 | 🗩 Reply | ⋯

Gustav Streicher Over a year ago

@PeterKionga-Kamau I see.

▲ 0 | 🗩 Reply | ⋯

▲

7

▼

🔖

↺

According to the documentation, it's now possible to load an environment file (contrary to a per-service file), docker-compose will then export the env variables defined in this env file prior to starting any service, they can then be used in the docker-compose.yml config file itself:

```
version: "3.7"
services:

  node:
    environment:
      APP_ENV: "${APP_ENV}"
      NODE_ENV: "${NODE_ENV}"
    ports:
      - "${HOST_EXPOSED_NODEJS_DEBUG_PORT}:9229"
    volumes:
      - type: bind
        source: ./project
        target: /var/www/project
        read_only: false
```

Since docker-compose 1.25 it's also possible to specify a custom .env file with the --env-file flag (unfortunately it's currently not possible to specify multiple .env files with the --env-file flag)

Share   Improve this answer   Follow   edited Nov 26, 2020 at 15:11   answered Jul 7, 2018 at 14:19

MADforFUNandHappy   Harel Moshe
567 ● 7 ● 25   **454** ● 7 ● 21

7 Comments ⌄

Add a comment

Peter Kionga-Kamau Over a year ago ✎

This is misleading. --env-file is for specifying *environment variables* for the container, *not* for specifying values to substitute in docker-compose.yml.

▲ 7   💬 Reply   ⋯

Gustav Streicher Over a year ago ✎

@PeterKionga-Kamau Nope. If you are using the `docker run` command directly, then the `--env-file` flag sets the environment for the container you are directly spinning up. If you are using the `docker compose` command, then the `--env-file` flag sets the environment for the pre-parsing step on the `compose.yml` file.

▲ 0   💬 Reply   ⋯

Peter Kionga-Kamau Over a year ago ✎

@GustavStreicher Not relevant - Docker compose specifically does NOT use `docker run`, it is a different thing altogether. Please refer to the docker compose [documentation](#).

▲ 0   💬 Reply   ⋯

Gustav Streicher Over a year ago

@PeterKionga-Kamau I'm literally referring to the use of `docker compose` and `docker run` as separate commands in my previous comment. So, I'm not sure what you are disagreeing with. When using the `docker compose` command, the `--env-file` means one thing. When using the `docker run` command, the `--env-file` means something else. That's exactly what my previous comment says.

▲ 0   💬 Reply   ⋯

Gustav Streicher Over a year ago

@PeterKionga-Kamau My comment is relevant because I'm pointing out that your expected behaviour only works if you are using the `docker run` command. If you are using the `docker compose` command then your expected behaviour no longer holds, and then I explain what actually happens.

▲ 1   💬 Reply   ⋯

|

Start asking to get answers

Find the answer to your question by asking.

Ask question

Explore related questions

docker   **docker-compose**

See similar questions with these tags.