



链滴

docker-compose 初体验

作者: [flowaters](#)

原文链接: <https://ld246.com/article/1513404613959>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

概述

docker-compose, 可以实现从Dockerfile来构建docker镜像。

安装

在mac下, 默认已经安装了docker-compose。其它环境下的安装见[官方文档](#)

Python web应用示例

本小节实现一个python web应用, 使用了Flask框架, 保存点击数据到redis中。

准备工作

已经安装好了Docker Engine和Docker Compose

设置

创建工程目录

```
$ mkdir composetest  
$ cd composetest
```

创建 app.py 文件

内容如下

```
from flask import Flask  
from redis import Redis  
  
app = Flask(__name_)  
redis = Redis(host='redis', port=6379)  
  
@app.route('/')  
def hello():  
    count = redis.incr('hits')  
    return 'Hello World! I have been seen {} times.\n'.format(count)  
  
if __name__ == "__main__":  
    app.run(host="0.0.0.0", debug=True)
```

其中redis是本容器的主机名; Redis服务使用默认端口6379

创建requirements.txt文件

内容如下

```
flask  
redis
```

编写 DockerFile

内容如下

```
FROM python:3.4-alpine
ADD ./code
WORKDIR /code
RUN pip install -r requirements.txt
CMD ["python", "app.py"]
```

含义

FROM python:3.4-alpine: 以Python 3.4镜像为基础来建立新镜像
ADD ./code : 把当前目录加入到镜像的/code目录下
WORKDIR /code : 设置工作目录为 /code
RUN pip install -r requirements.txt : 安装python依赖
CMD ["python", "app.py"] : 启动的默认命令 python app.py

定义服务文件 docker-compose.yml

内容如下

```
version: '3'
services:
  web:
    build: .
    ports:
      - "5000:5000"
  redis:
    image: "redis:alpine"
```

定义了两个服务，web和redis.

其中web服务

- 使用当前目录下的Dockerfile来生成
- 将容器中的5000端口提供给外部的5000端口，其中5000端口为Flask web服务器的默认端口

redis服务使用公开的redis镜像

生成和运行镜像

生成镜像

执行

```
docker-compose up
```

结果如下

```
[abeffect@fox composetest]$ docker-compose up
```

```
Building web
Step 1/5 : FROM python:3.4-alpine
3.4-alpine: Pulling from library/python
ab7e51e37a18: Pull complete
4a57a4e05b89: Pull complete
547e91589715: Pull complete
3a89af968cdb: Pull complete
1d882d2fb1c0: Pull complete
Digest: sha256:e716e8ca57060b881433ebcfe72b8c2f81c2d9c05b02b868c6540ff80400c9cd
Status: Downloaded newer image for python:3.4-alpine
---> ec2a8f24cd3f
Step 2/5 : ADD ./code
---> e15132d21a7f
Step 3/5 : WORKDIR /code
---> ba3d1a3f3c9d
Removing intermediate container 4af36b601af8
Step 4/5 : RUN pip install -r requirements.txt
---> Running in 15668592c1cc
Collecting flask (from -r requirements.txt (line 1))
  Downloading Flask-0.12.2-py2.py3-none-any.whl (83kB)
Collecting redis (from -r requirements.txt (line 2))
  Downloading redis-2.10.6-py2.py3-none-any.whl (64kB)
Collecting itsdangerous>=0.21 (from flask->-r requirements.txt (line 1))
  Downloading itsdangerous-0.24.tar.gz (46kB)
Collecting Jinja2>=2.4 (from flask->-r requirements.txt (line 1))
  Downloading Jinja2-2.10-py2.py3-none-any.whl (126kB)
Collecting click>=2.0 (from flask->-r requirements.txt (line 1))
  Downloading click-6.7-py2.py3-none-any.whl (71kB)
Collecting Werkzeug>=0.7 (from flask->-r requirements.txt (line 1))
  Downloading Werkzeug-0.13-py2.py3-none-any.whl (311kB)
Collecting MarkupSafe>=0.23 (from Jinja2>=2.4->flask->-r requirements.txt (line 1))
  Downloading MarkupSafe-1.0.tar.gz
Building wheels for collected packages: itsdangerous, MarkupSafe
  Running setup.py bdist_wheel for itsdangerous: started
  Running setup.py bdist_wheel for itsdangerous: finished with status 'done'
  Stored in directory: /root/.cache/pip/wheels/fc/a8/66/24d655233c757e178d45dea2de22a046d92766abfb741129a
  Running setup.py bdist_wheel for MarkupSafe: started
  Running setup.py bdist_wheel for MarkupSafe: finished with status 'done'
  Stored in directory: /root/.cache/pip/wheels/88/a7/30/e39a54a87bcbe25308fa3ca64e8ddc7d9b3e5afa21ee32d57
Successfully built itsdangerous MarkupSafe
Installing collected packages: itsdangerous, MarkupSafe, Jinja2, click, Werkzeug, flask, redis
Successfully installed Jinja2-2.10 MarkupSafe-1.0 Werkzeug-0.13 click-6.7 flask-0.12.2 itsdangerous-0.24 redis-2.10.6
---> 1257635dcda5
Removing intermediate container 15668592c1cc
Step 5/5 : CMD python app.py
---> Running in 25cad534cd57
---> 42b4b60fdd57
Removing intermediate container 25cad534cd57
Successfully built 42b4b60fdd57
Successfully tagged composetest_web:latest
WARNING: Image for service web was built because it did not already exist. To rebuild this im
```

```

ge you must use `docker-compose build` or `docker-compose up --build`.
Pulling redis (redis:alpine)...
alpine: Pulling from library/redis
1160f4abea84: Pull complete
a8c53d69ca3a: Pull complete
2dbffcaae30e: Pull complete
0a00d98b862d: Pull complete
6162f8e8bd54: Pull complete
ed931789deac: Pull complete
Digest: sha256:0153c5db97e5852b25c5c4715cf88cff9d9dfd8fef5c4c81df8b4dd1a984abc7
Status: Downloaded newer image for redis:alpine
Creating composetest_redis_1 ...
Creating composetest_web_1 ...
Creating composetest_redis_1
Creating composetest_web_1 ... done
Attaching to composetest_redis_1, composetest_web_1
redis_1 | 1:C 16 Dec 05:06:10.723 # oO0OoO0OoO0Oo Redis is starting oO0OoO0OoO0Oo
redis_1 | 1:C 16 Dec 05:06:10.723 # Redis version=4.0.6, bits=64, commit=00000000, modifie
=0, pid=1, just started
redis_1 | 1:C 16 Dec 05:06:10.723 # Warning: no config file specified, using the default config.
n order to specify a config file use redis-server /path/to/redis.conf
redis_1 | 1:M 16 Dec 05:06:10.724 * Running mode=standalone, port=6379.
redis_1 | 1:M 16 Dec 05:06:10.724 # WARNING: The TCP backlog setting of 511 cannot be en
orced because /proc/sys/net/core/somaxconn is set to the lower value of 128.
redis_1 | 1:M 16 Dec 05:06:10.724 # Server initialized
redis_1 | 1:M 16 Dec 05:06:10.724 # WARNING you have Transparent Huge Pages (THP) supp
rt enabled in your kernel. This will create latency and memory usage issues with Redis. To fix t
is issue run the command 'echo never > /sys/kernel/mm/transparent_hugepage/enabled' as r
ot, and add it to your /etc/rc.local in order to retain the setting after a reboot. Redis must be
estarted after THP is disabled.
redis_1 | 1:M 16 Dec 05:06:10.724 * Ready to accept connections
web_1 | * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
web_1 | * Restarting with stat
web_1 | * Debugger is active!
web_1 | * Debugger PIN: 196-949-087

```

查看结果

打开浏览器 <http://localhost:5000/>, 可见

Hello World! I have been seen 4 times.

查看新镜像

```

[abeffect@fox ~]$ docker image ls
REPOSITORY          TAG          IMAGE ID          CREATED          SIZE
composetest_web    latest      42b4b60fdd57     5 minutes ago   92
9MB

```

```

[abeffect@fox ~]$ docker inspect 42b4b60fdd57
[
  {
    "Id": "sha256:42b4b60fdd574d18f27884f32edf2cc4449523eb97d1554661aec5e50fe1efd3

```

```

    "RepoTags": [
      "composetest_web:latest"
    ],
    "RepoDigests": [],
    "Parent": "sha256:1257635dcda544e585de8b2ea1307ff0b7faf6a15c8ab90fc88aa1a6086d
cf0",
    "Comment": "",
    "Created": "2017-12-16T05:04:27.8917461Z",
    "Container": "25cad534cd57b81a9c6a06f298dd7b1a5c0191dc5f22abf9504fe7275648b62
",
    "ContainerConfig": {
      "Hostname": "25cad534cd57",
      "Domainname": "",
      "User": "",
      "AttachStdin": false,
      "AttachStdout": false,
      "AttachStderr": false,
      "Tty": false,
      "OpenStdin": false,
      "StdinOnce": false,
      "Env": [
        "PATH=/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
        "LANG=C.UTF-8",
        "GPG_KEY=97FC712E4C024BBEA48A61ED3A5CA953F73C700D",
        "PYTHON_VERSION=3.4.7",
        "PYTHON_PIP_VERSION=9.0.1"
      ],
      "Cmd": [
        "/bin/sh",
        "-c",
        "#(nop) ",
        "CMD [\"python\" \"app.py\"]"
      ],
      "ArgsEscaped": true,
      "Image": "sha256:1257635dcda544e585de8b2ea1307ff0b7faf6a15c8ab90fc88aa1a608
d1cf0",
      "Volumes": null,
      "WorkingDir": "/code",
      "Entrypoint": null,
      "OnBuild": [],
      "Labels": {}
    },
    "DockerVersion": "17.09.0-ce",
    "Author": "",
    "Config": {
      "Hostname": "",
      "Domainname": "",
      "User": "",
      "AttachStdin": false,
      "AttachStdout": false,
      "AttachStderr": false,
      "Tty": false,
      "OpenStdin": false,

```

```

"StdinOnce": false,
"Env": [
  "PATH=/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
  "LANG=C.UTF-8",
  "GPG_KEY=97FC712E4C024BBEA48A61ED3A5CA953F73C700D",
  "PYTHON_VERSION=3.4.7",
  "PYTHON_PIP_VERSION=9.0.1"
],
"Cmd": [
  "python",
  "app.py"
],
"ArgsEscaped": true,
"Image": "sha256:1257635dcda544e585de8b2ea1307ff0b7faf6a15c8ab90fc88aa1a608
d1cf0",
"Volumes": null,
"WorkingDir": "/code",
"Entrypoint": null,
"OnBuild": [],
"Labels": null
},
"Architecture": "amd64",
"Os": "linux",
"Size": 92857408,
"VirtualSize": 92857408,
"GraphDriver": {
  "Data": {
    "LowerDir": "/var/lib/docker/overlay2/e83123443e921e32993dd696e7695b379a294
18cdef437eceb9357528495ea5/diff:/var/lib/docker/overlay2/9cfaa5ec27e127277ee46b31b4d
c42ead3ee226d78a18d4e73d790afdb36192/diff:/var/lib/docker/overlay2/c5322d8fc626f2227
5ca47245659fb61a8425cb19cae870313816a6f7cdd795/diff:/var/lib/docker/overlay2/a0973b7
a22766473abcba4de270ca158fe7a880edfbef9e951bd8df7e092cdc/diff:/var/lib/docker/overla
2/4fa6628eb56d3084ca7df065efcbd67da2a28830a88a21d8f4be4ce0ac329b1a/diff:/var/lib/do
ker/overlay2/fc1d51ffe7967f8159ed2d2d7008a32b2d106ddb6b87f4fccadd2e3e16fae5c/diff",
    "MergedDir": "/var/lib/docker/overlay2/ff7f62d0dea47792506e149a63e87b79b545
35be28cb176256df6d0f11a56a8/merged",
    "UpperDir": "/var/lib/docker/overlay2/ff7f62d0dea47792506e149a63e87b79b54583
be28cb176256df6d0f11a56a8/diff",
    "WorkDir": "/var/lib/docker/overlay2/ff7f62d0dea47792506e149a63e87b79b54583
be28cb176256df6d0f11a56a8/work"
  },
  "Name": "overlay2"
},
"RootFS": {
  "Type": "layers",
  "Layers": [
    "sha256:52a5560f4ca0b62c53985258faceddc20afe53fca394aec8a32083e01659d9fd"
    "sha256:7c5ea328fb338eacb5da19231dff14cfef5f4048541efd160ecd3f0351862124"
    "sha256:c30b37dd392440c1c27a25382477535de1d11d939eb34c49b23868f11b6d7
6b",
    "sha256:97291ed7e0ce205b5bc6c47668975b75dab976f07748dd7002f93c54a7b5c7
d",

```

```
    "sha256:8348954981b10ddde61e6f520992d54f3b2847a520330ebcbd95cbb01d0d5
50",
    "sha256:bd34a5638bcc4c1b68420ee40d46e8844c0d12657eb9ab7720aae3af93a5b4
7",
    "sha256:24edd138c6481e9a541537e3740d205db7bd285f4f00ad02622a1ddb95ff69
d"
  ]
},
"Metadata": {
  "LastTagTime": "2017-12-16T05:04:28.0303435Z"
}
}
]
```

停止应用

```
docker-compose down
```

或者

```
Ctrl+C
```

```
Gracefully stopping... (press Ctrl+C again to force)
Stopping composetest_web_1 ... done
Stopping composetest_redis_1 ... done
```

增加一个bind点

修改docker-compose.yml文件，增加一个bind点。

```
version: '3'
services:
  web:
    build: .
    ports:
      - "5000:5000"
    volumes:
      - ./code
  redis:
    image: "redis:alpine"
```

volumes参数将当前目录挂载到容器的code目录下，实现可以动态修改code目录内容，而不用重新生成镜像。

重新生成镜像

```
[abeffect@fox composetest]$ docker-compose up
Recreating composetest_web_1 ...
Starting composetest_redis_1 ...
Starting composetest_redis_1
Recreating composetest_web_1 ... done
Attaching to composetest_redis_1, composetest_web_1
```



```
redis_1 | 1:C 16 Dec 05:46:49.230 # oO0OoO0OoO0Oo Redis is starting oO0OoO0OoO0Oo
redis_1 | 1:C 16 Dec 05:46:49.230 # Redis version=4.0.6, bits=64, commit=00000000, modifie
=0, pid=1, just started
redis_1 | 1:C 16 Dec 05:46:49.230 # Warning: no config file specified, using the default config.
n order to specify a config file use redis-server /path/to/redis.conf
redis_1 | 1:M 16 Dec 05:46:49.231 * Running mode=standalone, port=6379.
redis_1 | 1:M 16 Dec 05:46:49.231 # WARNING: The TCP backlog setting of 511 cannot be en
forced because /proc/sys/net/core/somaxconn is set to the lower value of 128.
redis_1 | 1:M 16 Dec 05:46:49.231 # Server initialized
redis_1 | 1:M 16 Dec 05:46:49.231 # WARNING you have Transparent Huge Pages (THP) supp
rt enabled in your kernel. This will create latency and memory usage issues with Redis. To fix t
his issue run the command 'echo never > /sys/kernel/mm/transparent_hugepage/enabled' as r
oot, and add it to your /etc/rc.local in order to retain the setting after a reboot. Redis must be
 restarted after THP is disabled.
redis_1 | 1:M 16 Dec 05:46:49.231 * DB loaded from disk: 0.000 seconds
redis_1 | 1:M 16 Dec 05:46:49.231 * Ready to accept connections
web_1 | * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
web_1 | * Restarting with stat
web_1 | * Debugger is active!
web_1 | * Debugger PIN: 196-949-087
```

更新应用

1. 修改 app.py 文件，将return行的内容更改如下

```
return 'Hello from Docker! I have been seen {} times.\n'.format(count)
```

2. 刷新浏览器，可见

```
Hello from Docker! I have been seen 13 times.
```

其它命令

1. docker-compose up -d

后台运行参数 -d

```
[abeffect@fox composetest]$ docker-compose up -d
Starting composetest_web_1 ...
Starting composetest_redis_1 ...
Starting composetest_web_1
Starting composetest_web_1 ... done
```

2. docker-compose ps

```
[abeffect@fox composetest]$ docker-compose ps
      Name                       Command                                State      Ports
-----
composetest_redis_1  docker-entrypoint.sh redis ...      Up        6379/tcp
composetest_web_1   python app.py                          Up        0.0.0.0:5000->5000/tcp
```

3. docker-compose run web env

docker-compose run参数, 在容器外, 运行容器内的命令

```
[abeffect@fox composetest]$ docker-compose run web env
PATH=/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=c58f98dd0dc6
TERM=xterm
LANG=C.UTF-8
GPG_KEY=97FC712E4C024BBEA48A61ED3A5CA953F73C700D
PYTHON_VERSION=3.4.7
PYTHON_PIP_VERSION=9.0.1
HOME=/root
```

4. docker-compose --help

查看其它命令

5. docker-compose stop

停止由-d参数启动的容器

```
[abeffect@fox composetest]$ docker-compose stop
Stopping composetest_web_1 ... done
Stopping composetest_redis_1 ... done
```

6. docker-compose down --volumes

停止所有, 删除redis的数据volumes

```
[abeffect@fox composetest]$ docker-compose down --volumes
Removing composetest_web_run_6 ... done
Removing composetest_web_run_5 ... done
Removing composetest_web_run_4 ... done
Removing composetest_web_run_3 ... done
Removing composetest_web_run_2 ... done
Removing composetest_web_run_1 ... done
Removing composetest_web_1 ... done
Removing composetest_redis_1 ... done
Removing network composetest_default
```

到此, 已经了解了docker-compose的基本运行.

常用指令

见 docker-compose 帮助

	build
uild or rebuild services	
bundle	Generate a Docker bundle from the C
mpose file	
config	Validate and view the Compose file
create	Create services
down	Stop and remove containers, networks,

images, and volumes

events	Receive real time events from containers
exec	Execute a command in a running container
help	Get help on a command
images	List images
kill	Kill containers
logs	View output from containers
pause	Pause services
port	Print the public port for a port binding
ps	List containers
pull	Pull service images
push	Push service images
restart	Restart services
rm	Remove stopped containers
run	Run a one-off command
scale	Set number of containers for a service
start	Start services
stop	Stop services
top	Display the running processes
unpause	Unpause services
up	Create and start containers
version	Show the Docker-Compose version information

示例:

ps List containers

`docker-compose ps`

`docker ps -a`

config

`docker-compose -f lnmp.yaml config`

`docker-compose -f lnmp.yaml config --services`

`docker-compose -f lnmp.yaml config --volumes`

down: 停止并删除容器、网络、镜像、数据卷

exec: 在运行的容器中执行命令

images: 列出镜像

参考

- [官方文档](#)
- [官方python web应用教程](#)
- [docker-compose初试及命令基础](#)