

docker-compose的使用

Docker-Compose卸载（二进制包方式的安装进行删除）：

```
rm /usr/local/bin/docker-compose
```

先做docker-compose的安装：

```
[root@zhx ~]# curl -L
https://get.daocloud.io/docker/compose/releases/download/1
.24.0/docker-compose-`uname -s`-`uname -m` >
/usr/local/bin/docker-compose
  % Total    % Received % Xferd  Average Speed   Time
Time      Time     Current                Dload  Upload   Total
Spent     Left  Speed
100  423  100  423    0     0   1983      0 --:--:-- -
-:--:-- --:--:-- 1985
100 15.4M  100 15.4M    0     0  23.5M      0 --:--:-- -
-:--:-- --:--:-- 23.5M
```

添加可执行权限：

```
[root@zhx ~]# chmod +x /usr/local/bin/docker-compose
```

测试安装结果：

```
[root@zhx ~]# docker-compose --version
docker-compose version 1.24.0, build 0aa59064
```

关于docker-compose模板文件：

compose 文件是一个定义服务、网络 and 卷的 **YAML** 文件。Compose 文件的默认路径是 `./docker-compose.yml` 【提示：可以用 `.yml` 或 `.yaml` 作为文件扩展名】。

文件的组成部分：

Docker-Compose标准模板文件应该包含version、services、networks 三大部分，最关键的是services和networks两个部分。

networks关键字：指定自定义网络

yml文件配置项（文件组成）：

version: 定义的yaml文件的版本只能向下兼容【例如我的compose版本是1.24那我只能使用3的版本，只管用】

services: 说明这里可以有多个服务

tomcat: 这里主要是给配置人员看的，起什么名字都行。

restart: always 如果我的服务器重启，这个容器也重启。【linux中所有的服务都随系统启动而启动】

container_name: tomcat容器的名字

image: 是指定服务的镜像名称或镜像ID。如果镜像在本地不存在，Compose将会尝试拉取镜像。

depends_on: 标签用于解决容器的依赖、启动先后的问题

ports:用于端口映射的标签

- **80:80 左边宿主机端口：右边容器端口**

```
version: '3'
services:
  tomcat:
    restart: always
    image: tomcat
    container_name: tomcat
    depends_on:
      - nginx
    ports:
      - 8080:8080
  nginx:
    container_name: nginx
    image: nginx
    ports:
      - 80:80
    restart: always
```