# docker-compose自定义网络，固定容器ip地址

## 一 在docker-compose.yml自定义网络

#### V2版本

version: '2'  
services:  
 nginx:  
 image: nginx:latest  
 container\_name: nginx-ip  
 restart: always  
 networks:  
 extnetwork:  
 ipv4\_address: 172.19.0.2  
 ports:  
 - '8080:80'  
  
networks:  
 extnetwork:  
 ipam:  
 config:  
 - subnet: 172.19.0.0/16  
 gateway: 172.19.0.1

说明：gateway是网关地址subnet是网络号段extnetwork是自定义的网络名称

上面配置中我们的[nginx](https://so.csdn.net/so/search?q=nginx&spm=1001.2101.3001.7020)容器固定ip为172.19.0.2

#### V3版本

version: '3'  
services:  
 nginx:  
 image: nginx:latest  
 container\_name: nginx-ip  
 restart: always  
 networks:  
 nginx-network:  
 ipv4\_address: 172.19.0.2  
 ports:  
 - '8080:80'  
  
#networks:  
# nginx-network:  
# ipam:  
# config:  
# - subnet: 172.19.0.0/16  
  
networks:  
 nginx-network:  
 driver: bridge  
 ipam:  
 driver: default  
 config:  
 - subnet: 172.19.0.0/16

**V2以前，可以在ipam里指定gateway V3以后不许了。**

**静态指定ip，则每个容器都要静态指定，添加networks 部分**



#### V3 多实例

version: "3"  
services:  
  
 main-server:  
 restart: always  
 image: main\_server:v0.0.1  
 container\_name: "main-server"  
 ports:  
 - "8080:80"  
 networks:  
 app\_net:  
 ipv4\_address: 172.30.0.2  
  
 sub-server:  
 restart: always  
 image: sub\_server:v0.0.1  
 container\_name: "sub-server"  
 ports:  
 - "8080:80"  
 networks:  
 app\_net:  
 ipv4\_address: 172.30.0.3  
  
networks:  
 app\_net:  
 driver: bridge  
 ipam:  
 driver: default  
 config:  
 - subnet: 172.30.0.0/24

#### 参考文档

参考文档

<https://blog.csdn.net/zhuchunyan_aijia/article/details/87079744>

<http://www.cncsto.com/article/2086> <https://www.ddpool.cn/article/92756.html>

<https://www.cnblogs.com/charnet1019/p/16005194.html>

<https://www.shuzhiduo.com/A/QW5Yj3Z9dm/> docker network create --subnet=172.18.0.0/16 mynetwork

<https://www.codenong.com/3abd3d28214dea526084/> 多实例

<https://zhuanlan.zhihu.com/p/224876594>

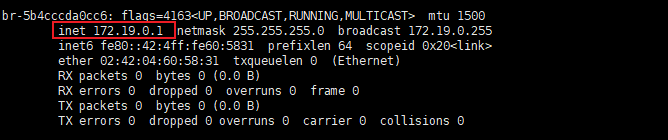
<https://blog.51cto.com/u_15685799/5392844>

## 二 事先通过docker命令创建相应网络

docker 创建网络

docker network create --subnet=172.19.0.0/24 xmmtr3-network

创建网桥之后,使用ifconfig查看 会多出一个网桥,该网桥在docker启动或者重启之后,会自动显示出来



使用docker network rm xmmtr3-network 移除网桥

docker-compose.yml

version: '2.2'  
services:  
 xmmtr3\_redis:  
 container\_name: xmmtr3\_redis  
 image: redis:5  
 restart: always  
 cpu\_shares: 50  
 mem\_limit: 200m  
 environment:  
 - TZ=Asia/Shanghai  
 command: ["redis-server","/usr/local/etc/redis/redis.conf"]  
 volumes:  
 - "./conf/redis.conf:/usr/local/etc/redis/redis.conf"  
 - "./data/redis:/data"  
 ports:  
 - '50012:6379'  
 networks:  
 xmmtr3-network:  
 ipv4\_address: 172.19.0.2  
  
 xmmtr3\_mongo:  
 image: mongo:5.0  
 container\_name: xmmtr3\_mongo  
 command: mongod --config /etc/mongo/mongo.conf --logpath /log/mongo.log  
 restart: always  
 mem\_limit: 3g  
 environment:  
 TZ: Asia/Shanghai  
 volumes:  
 - ./data/mongo:/data/db  
 - ./log:/log  
 - ./conf:/etc/mongo  
 ports:  
 - "50117:27017"  
 networks:  
 xmmtr3-network:  
 ipv4\_address: 172.19.0.3  
  
#引用自定义网络  
networks:  
 xmmtr3-network:  
 external: true

redis.conf

protected-mode no

port 6379

tcp-backlog 511

timeout 0

tcp-keepalive 300

daemonize no

supervised no

pidfile /var/run/redis\_6379.pid

loglevel notice

logfile ""

databases 16

save 900 1

save 300 10

save 60 10000

stop-writes-on-bgsave-error yes

rdbcompression yes

rdbchecksum yes

dbfilename dump.rdb

dir ./

slave-serve-stale-data yes

slave-read-only yes

repl-diskless-sync no

repl-diskless-sync-delay 5

repl-disable-tcp-nodelay no

slave-priority 100

requirepass 123.com

appendonly yes

appendfsync everysec

no-appendfsync-on-rewrite no

auto-aof-rewrite-percentage 100

auto-aof-rewrite-min-size 512mb

aof-load-truncated yes

lua-time-limit 5000

slowlog-log-slower-than 10000

slowlog-max-len 128

latency-monitor-threshold 0

notify-keyspace-events ""

hash-max-ziplist-entries 512

hash-max-ziplist-value 64

list-max-ziplist-size -2

list-compress-depth 0

set-max-intset-entries 512

zset-max-ziplist-entries 128

zset-max-ziplist-value 64

hll-sparse-max-bytes 3000

activerehashing yes

client-output-buffer-limit normal 0 0 0

client-output-buffer-limit slave 256mb 64mb 60

client-output-buffer-limit pubsub 32mb 8mb 60

hz 10

aof-rewrite-incremental-fsync yes

mongo.conf

storage:

dbPath: "/data/db"

# 关于存储引擎wiredTiger的参数设置

# 参考：<https://docs.mongodb.com/manual/reference/configuration-options/#storage-options>

wiredTiger:

engineConfig:

# cacheSizeGB: <number>

journalCompressor: "zstd"

directoryForIndexes: true

collectionConfig:

blockCompressor: "zstd"

security:

authorization: enabled

systemLog:

verbosity: 0

quiet: false

traceAllExceptions: false

destination: "file"

logAppend: true

logRotate: reopen

net:

bindIpAll: true

port: 27017

[Docker绑定固定IP/跨主机容器互访操作](http://www.kaotop.com/it/305441.html)

<https://www.jb51.net/article/200693.htm>