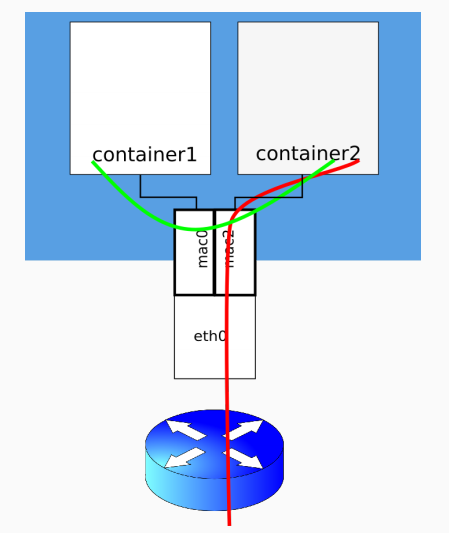
# 为Docker配置MACVLAN网络

韦体东

## MACVLAN模式

MACVLAN有四种模式，分别是Bridge（默认），Private，VEPA(Virtual Ethernet Port Aggregator)，Passthru。

这些类型的区别，网上比较多，不细说，我只配置默认的Bridge模式。



## 配置docker网络

我们在eth0口上配置macvlan。

先看看eth0的基本信息，上面之前有一个IP 11.10.4.1：

node-885896414408 ~ # ip addr show eth0

2: eth0: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc mq state UP group default

link/ether fe:fc:fe:2a:6f:05 brd ff:ff:ff:ff:ff:ff

inet 11.10.4.1/16 brd 11.10.255.255 scope global eth0

valid\_lft forever preferred\_lft forever

inet6 fe80::fcfc:feff:fe2a:6f05/64 scope link

valid\_lft forever preferred\_lft forever

为了不和这个IP冲突，我们需要在创建macvlan的时候，排除此IP，使用docker network命令创建：

node-885896414408 ~ # docker network create -d macvlan --subnet=11.10.4.0/16 --gateway=11.10.4.254 --aux-address="exclude\_host=11.10.4.1" -o parent=eth0 macvlan0

d72dd733ce3a7a26b93d7e472e2ef036f26191f9577ab673978689b0bd3f5205

查看：

node-885896414408 ~ # docker network ls

NETWORK ID NAME DRIVER SCOPE

d72dd733ce3a macvlan0 macvlan local

好了，现在已经有了macvlan0网络，属于macvlan类型，我们启动docker的时候，指定其使用macvlan0即可：

node-885896414408 ~ # docker run -it --network=macvlan0 mytest /bin/bash -l

[mytest@ ~]#ifconfig

eth0 Link encap:Ethernet HWaddr 02:42:0b:0a:00:01

inet addr:11.10.0.1 Bcast:0.0.0.0 Mask:255.255.0.0

inet6 addr: fe80::42:bff:fe0a:1/64 Scope:Link

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:0 errors:0 dropped:0 overruns:0 frame:0

TX packets:6 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:0 (0.0 B) TX bytes:508 (508.0 B)

鉴于macvlan的特性，从容器内部是ping不通宿主机的，但是，可以ping通外部网络：

[mytest@ ~]#ping 11.10.4.1 #不能PING通主机

PING 11.10.4.1 (11.10.4.1) 56(84) bytes of data.

From 11.10.0.1 icmp\_seq=1 Destination Host Unreachable

From 11.10.0.1 icmp\_seq=2 Destination Host Unreachable

[mytest@ ~]#ping 11.10.4.2 #但能PING通其他机

PING 11.10.4.2 (11.10.4.2) 56(84) bytes of data.

64 bytes from 11.10.4.2: icmp\_req=1 ttl=64 time=0.207 ms

64 bytes from 11.10.4.2: icmp\_req=2 ttl=64 time=0.756 ms

## 容器和宿主机通信

为了能和宿主机通信，我们需要在宿主机上再创建一个macvlan网络：

node-885896414408 ~ # ip link add link eth0 macvlan0-host type macvlan mode bridge

node-885896414408 ~ # ip link set dev macvlan0-host up

node-885896414408 ~ # ip addr add 11.10.4.254/16 dev macvlan0-host

查看一下配置后的虚拟macvlan网口：

node-885896414408 ~ # ifconfig macvlan0-host

macvlan0-host: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500

inet 11.10.4.254 netmask 255.255.0.0 broadcast 0.0.0.0

inet6 fe80::c026:88ff:fe53:cae prefixlen 64 scopeid 0x20<link>

ether c2:26:88:53:0c:ae txqueuelen 1000 (Ethernet)

RX packets 62 bytes 3472 (3.3 KiB)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 8 bytes 648 (648.0 B)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

此时docker内部就能PING通宿主机IP 11.10.4.254了，如果还是不通，可以加一条路由：

ip route add 11.10.0.2 dev macvlan0-host

不过，我看已经有默认路由：

11.10.0.0/16 dev macvlan0-host proto kernel scope link src 11.10.4.254

流程很简单，之所以记录，是因为网上多转载的文章，多只说了如何配置docker网络部分，后面和宿主机通信部分，都是一笔带过，很少有实际的操作命令贴出。

## 参考文档

1. Macvlan and Ipvlan Network Drivers <http://www.rendoumi.com/zui-xin-ban-ben-dockerwang-luo-bu-fen-suo-yong-de-macvlanhe-ipvlan/>

2017/7/17