不用公网ip验证submariner的NAT穿透

20210715

因为最开始的时候，cluster-a是global router的，所以本文档最开始是接着global router这个case写的

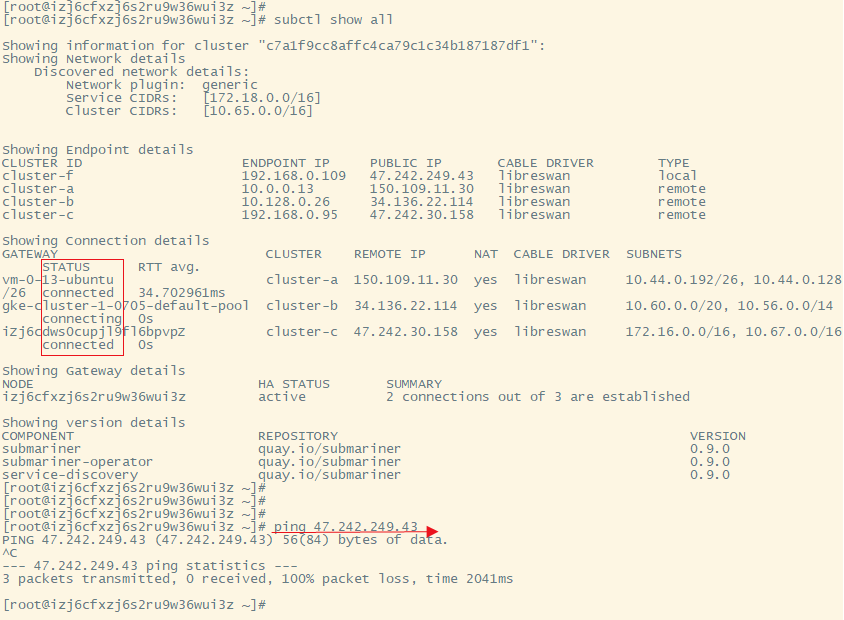
北京的同事说我们的case有问题，cluster-b在join的过程中用到了公网ip。而在我们的需求中，cluster-c不能使用公网ip。我们采用TKE和AKS验证我们的猜想，现在的情况是TKE（cluster-a）和AKS（cluster-b）已经join了subctl的borker，并且cluster-a和cluster-c都带有公网ip。

于是我们创建了一个cluster-f，cluster-f与cluster-c处于同一地区（阿里云，香港），cluster-f所包含的三个master节点都不带有公网ip。我们通过cluster-c的节点可以访问cluster-f（同处于一个内网），并且对cluster-f进行配置，实验表明，即使cluster-f的节点没有公网ip，我们的case依旧能work。

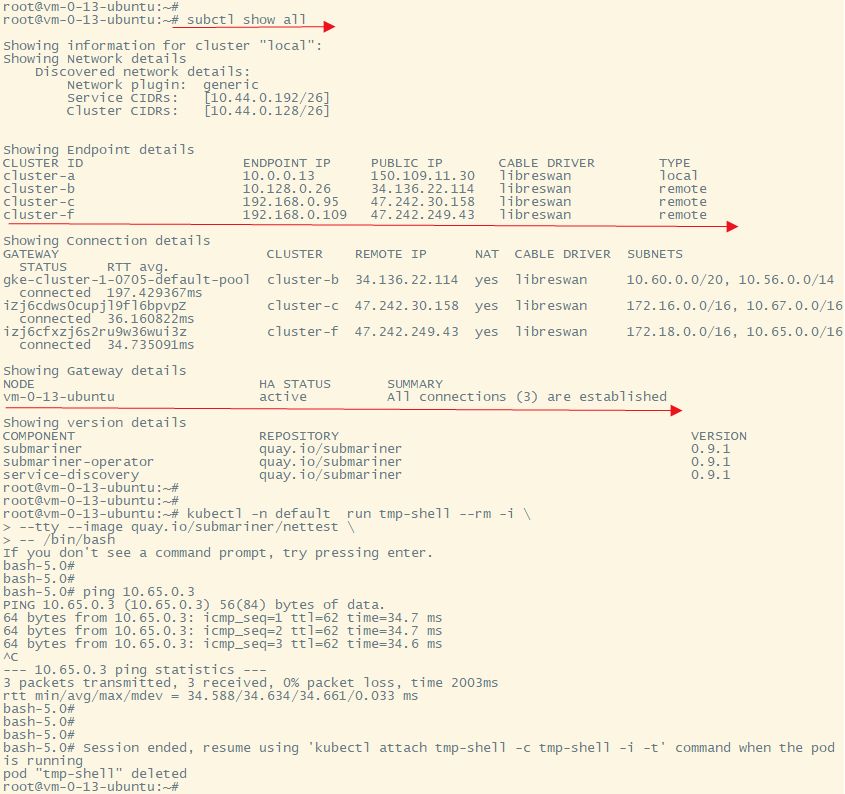
cluster-f明明没有购买公网ip，但是这里却显示有公网ip，如何解释（这个公网ip是NAT的公网ip）

还有一个问题是在这种情况下，cluster-f能连接cluster-a（broker），能连接cluster-c（同处于一个内网），但是无法连接cluster-b（GKE），估计是NAT的问题，后面新开了一个case没有这个问题

这里为什么cluster-f能连接cluster-c，cluster-c公网ip，优先级高于NAT

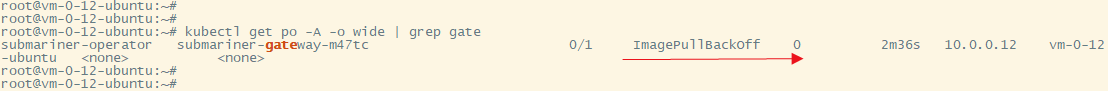


对于cluster-a来说，cluster-a能够和所有cluster正常通信



可能遇到的问题

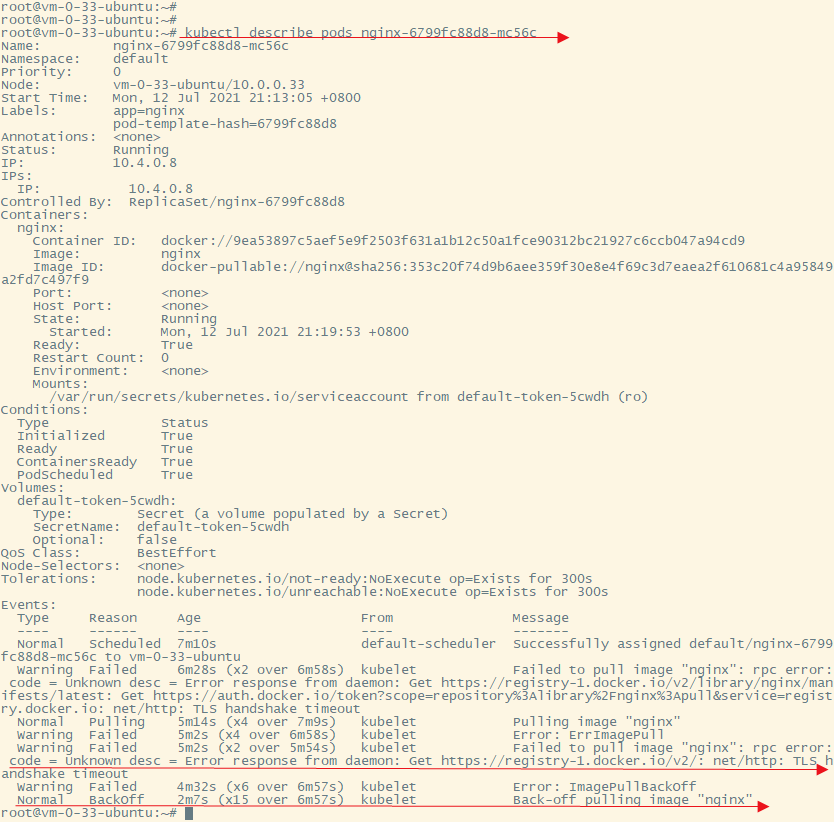
gateway起不来ImagePullBackoff问题，可能是暂时没有下载下来，我是在报错之后等一会就好了



查看describe是否正常

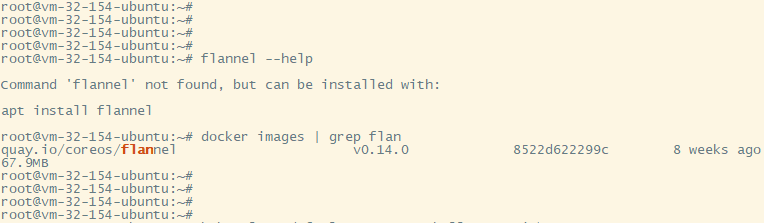
kubectl describe pods <pod name>

kubectl describe pods nginx-6799fc88d8-mc56c



查看flannel的版本

docker images | grep flannel



查看日志

kubectl logs -f <your pod name> -n <your namespace>

如果觉得日志太长，可以限制输出日志的字符数，日入输出日志的前3000 byte

kubectl logs --limit-bytes=3000 <your pod name> -n <your namespace>

在执行更复杂的NAT方案时，我们一开始遇到了困难，然后我们再GitHub提出了对应的issue  
https://github.com/submariner-io/submariner/issues/1492

现在这个问题已经解决了，下面是issue的文字版原文

I have some problems in learning submariner. Actually I fail to replay some cases in NAT traversal

I have successfully deployed submariner to my 4 kubernetes clusters (version 1.19.7) created by kubeadm, and all of them works well include pods and services. The problem is that pods in cluster-b can't access to pods in cluster-c by cluster IP across cluster, but access to pods in cluster-a, cluster-b and cluster-d by cluster IP across cluster. The four clusters are deployed as Fig. 1, referring to the column "Public Cloud vs On-Premises" in this link https://submariner.io/operations/nat-traversal

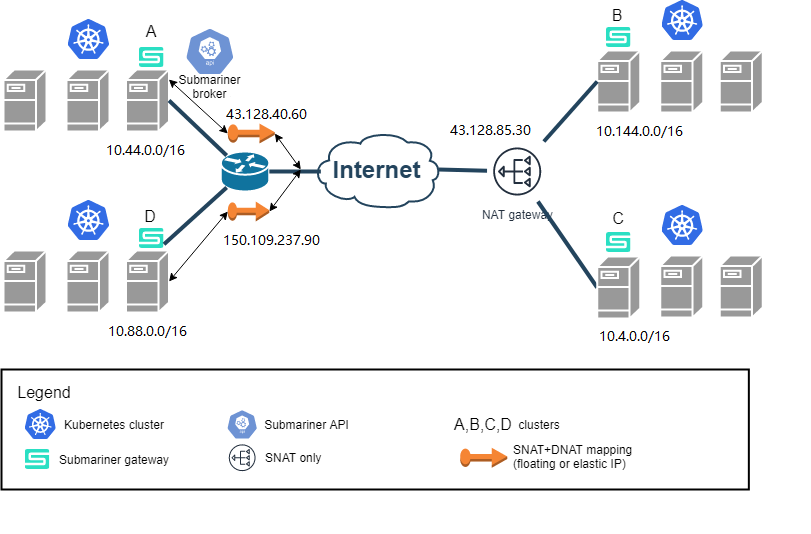


Fig. 1

reproduce steps

1. create 4 kubernetes clusters by kubeadm with no pod/service CIDR overlap
2. assign a NAT gateway for cluster-b and cluster-c
3. deploy broker on cluster-a
4. join cluster-a, cluster-b, cluster-c and cluster-d to broker cluster cluster-a
5. kubectl annotate node, restart the gateways and Router Port Mapping
6. try verify-manually case by using a nginx service like https://submariner.io/getting-started/quickstart/k3s/#verify-manually

my scripts to create above 4 clusters for reproduce this issue:

cluster-a.sh.txt

for cluster-a

root@vm-32-154-ubuntu:~# subctl show all

Showing information for cluster "kubernetes":

Showing Network details

Discovered network details:

Network plugin: generic

Service CIDRs: [10.45.0.0/16]

Cluster CIDRs: [10.44.0.0/16]

Showing Endpoint details

CLUSTER ID ENDPOINT IP PUBLIC IP CABLE DRIVER TYPE

cluster-a 10.0.32.154 43.128.40.60 libreswan local

cluster-b 10.0.0.12 43.128.85.30 libreswan remote

cluster-c 10.0.0.33 43.128.85.30 libreswan remote

cluster-d 172.29.100.152 150.109.237.90 libreswan remote

Showing Connection details

GATEWAY CLUSTER REMOTE IP NAT CABLE DRIVER SUBNETS STATUS RTT avg.

vm-0-12-ubuntu cluster-b 43.128.85.30 yes libreswan 10.145.0.0/16, 10.144.0.0/16 connected 32.225572ms

vm-0-33-ubuntu cluster-c 43.128.85.30 yes libreswan 10.5.0.0/16, 10.4.0.0/16 connected 36.770258ms

vm-100-152-ubuntu cluster-d 150.109.237.90 yes libreswan 10.89.0.0/16, 10.88.0.0/16 connected 36.819204ms

Showing Gateway details

NODE HA STATUS SUMMARY

vm-32-154-ubuntu active All connections (3) are established

Showing version details

COMPONENT REPOSITORY VERSION

submariner quay.io/submariner 0.9.0

submariner-operator quay.io/submariner 0.9.0

service-discovery quay.io/submariner 0.9.0

for cluster-b

root@vm-0-12-ubuntu:~# subctl show all

Showing information for cluster "kubernetes":

Showing Network details

Discovered network details:

Network plugin: generic

Service CIDRs: [10.145.0.0/16]

Cluster CIDRs: [10.144.0.0/16]

Showing Endpoint details

CLUSTER ID ENDPOINT IP PUBLIC IP CABLE DRIVER TYPE

cluster-b 10.0.0.12 43.128.85.30 libreswan local

cluster-a 10.0.32.154 43.128.40.60 libreswan remote

cluster-c 10.0.0.33 43.128.85.30 libreswan remote

cluster-d 172.29.100.152 150.109.237.90 libreswan remote

Showing Connection details

GATEWAY CLUSTER REMOTE IP NAT CABLE DRIVER SUBNETS STATUS RTT avg.

vm-32-154-ubuntu cluster-a 43.128.40.60 yes libreswan 10.45.0.0/16, 10.44.0.0/16 connected 34.503358ms

vm-0-33-ubuntu cluster-c 43.128.85.30 yes libreswan 10.5.0.0/16, 10.4.0.0/16 connecting 0s

vm-100-152-ubuntu cluster-d 150.109.237.90 yes libreswan 10.89.0.0/16, 10.88.0.0/16 connected 68.298751ms

Showing Gateway details

NODE HA STATUS SUMMARY

vm-0-12-ubuntu active 2 connections out of 3 are established

Showing version details

COMPONENT REPOSITORY VERSION

submariner quay.io/submariner 0.9.0

submariner-operator quay.io/submariner 0.9.0

service-discovery quay.io/submariner 0.9.0

The results are summarized in the table

| cluster-name | cluster-a | cluster-b | cluster-c | cluster-d |

| ------------- | ------------- | ------------- | ------------- | ------------- |

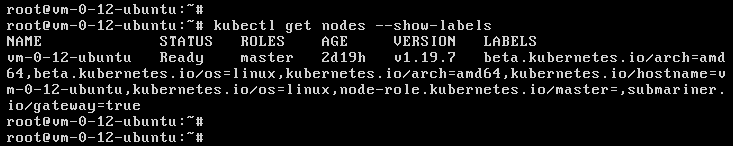
| cluster-a | Y | Y | Y | Y |

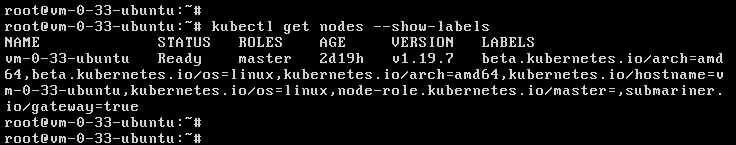
| cluster-b | Y | Y | ###N | Y |

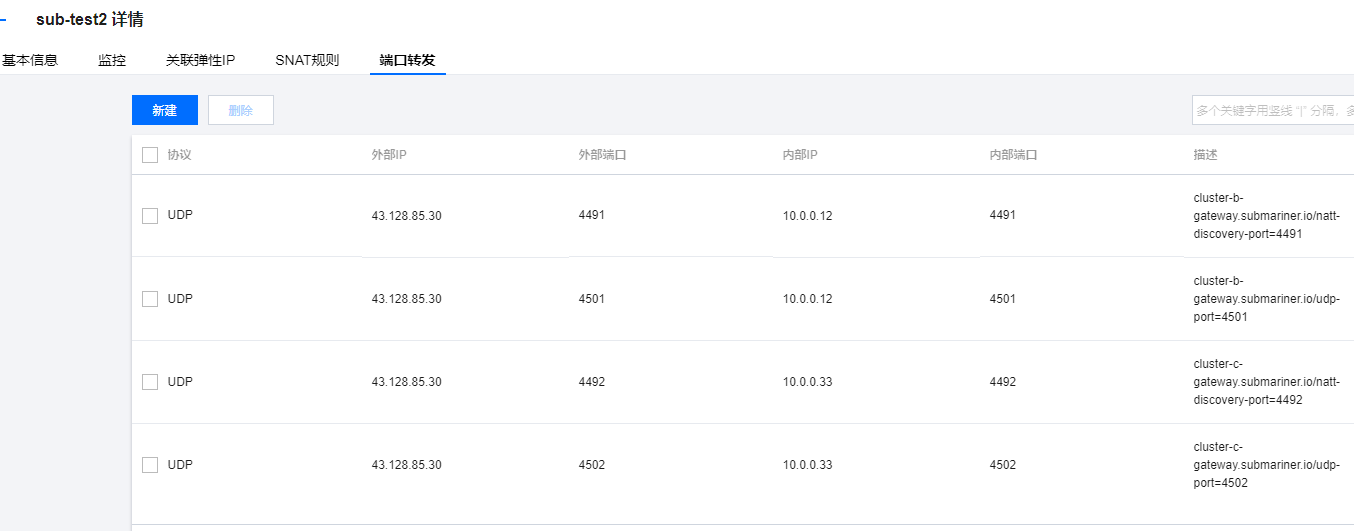
| cluster-c | Y | ###N | Y | Y |

| cluster-d | Y | Y | Y | Y |

kubectl get nodes --show-labels

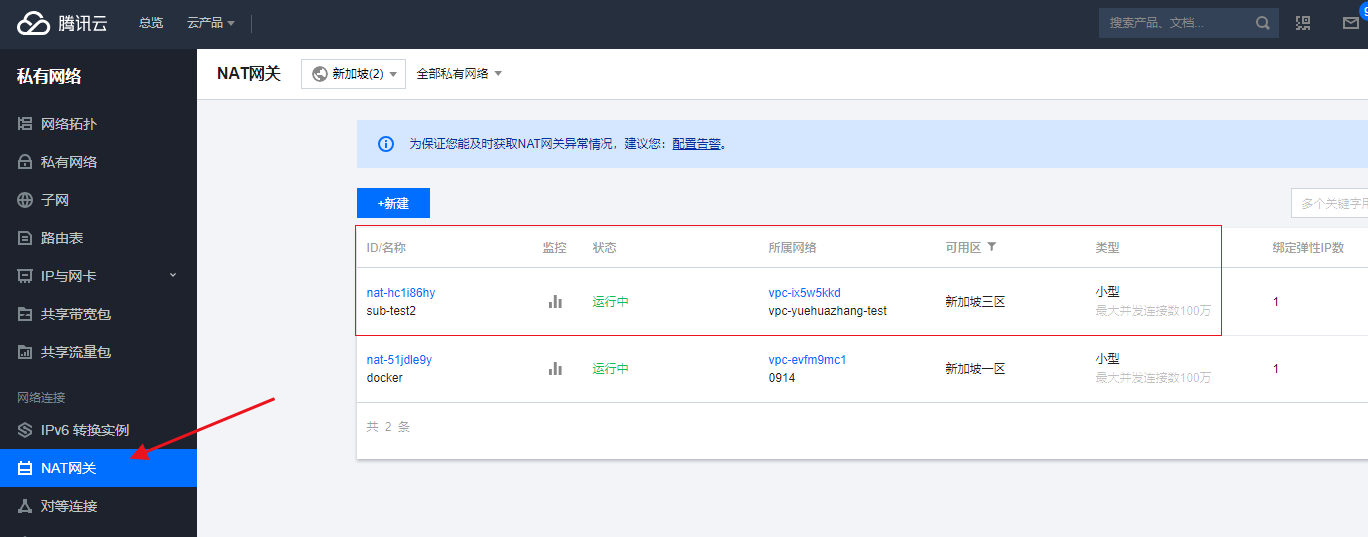




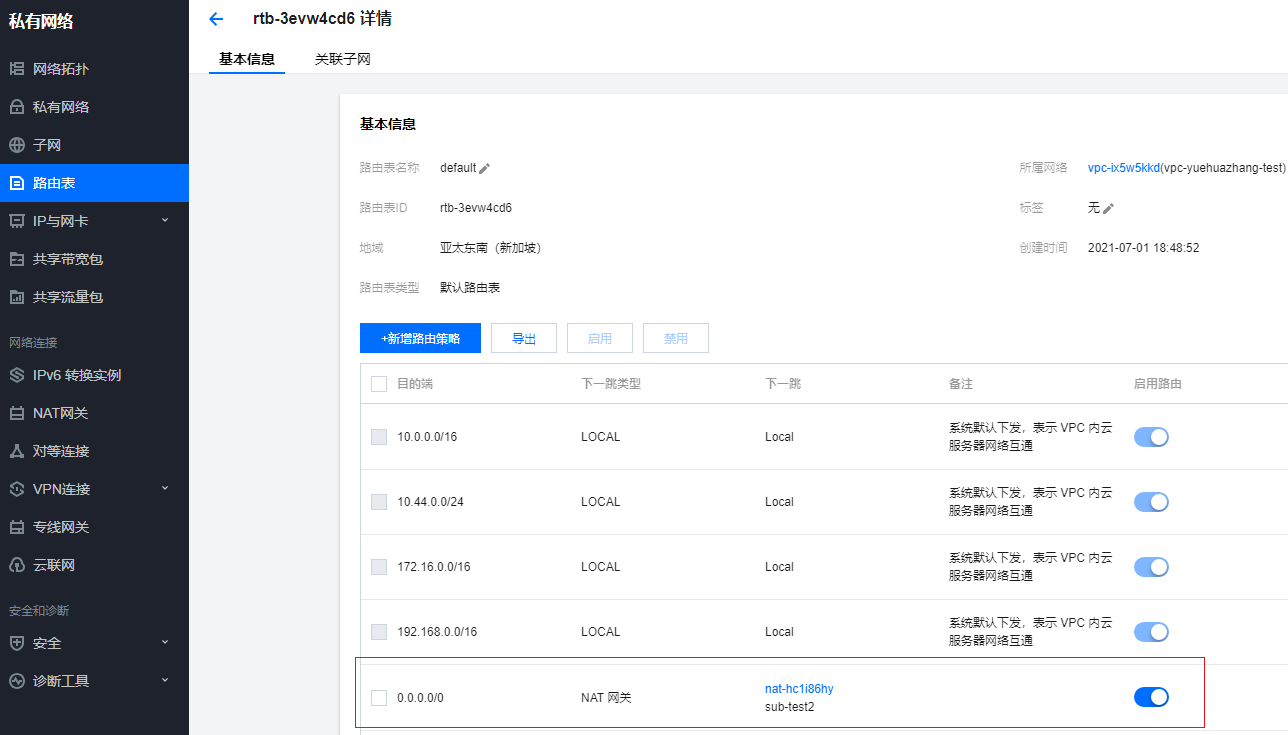


解决方案，注意以下几点即可

(1)创建一个NAT网关，红色框是我们创建的NAT网关



把这个网关添加到内网cluster所在的路由表



(2)安装subctl的时候，需要安装v0.9.1的版本，v0.9.0版本有问题

curl -Ls https://get.submariner.io | VERSION=0.9.1 bash

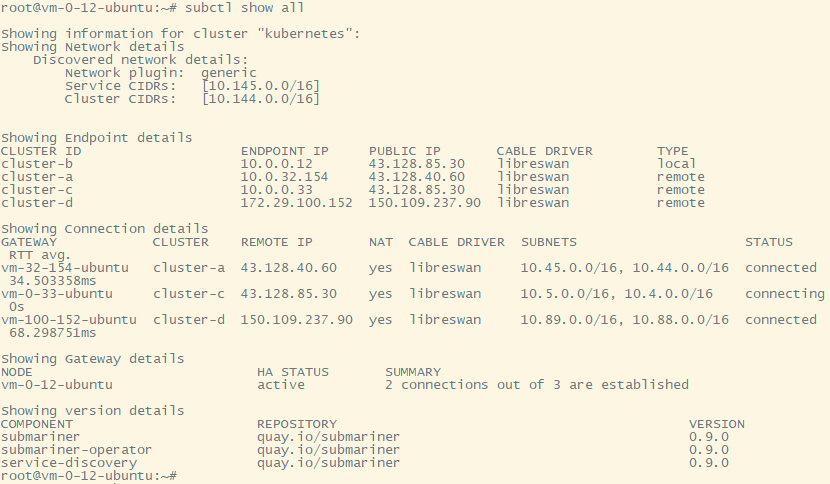
export PATH=$PATH:~/.local/bin

echo export PATH=\$PATH:~/.local/bin >> ~/.profile

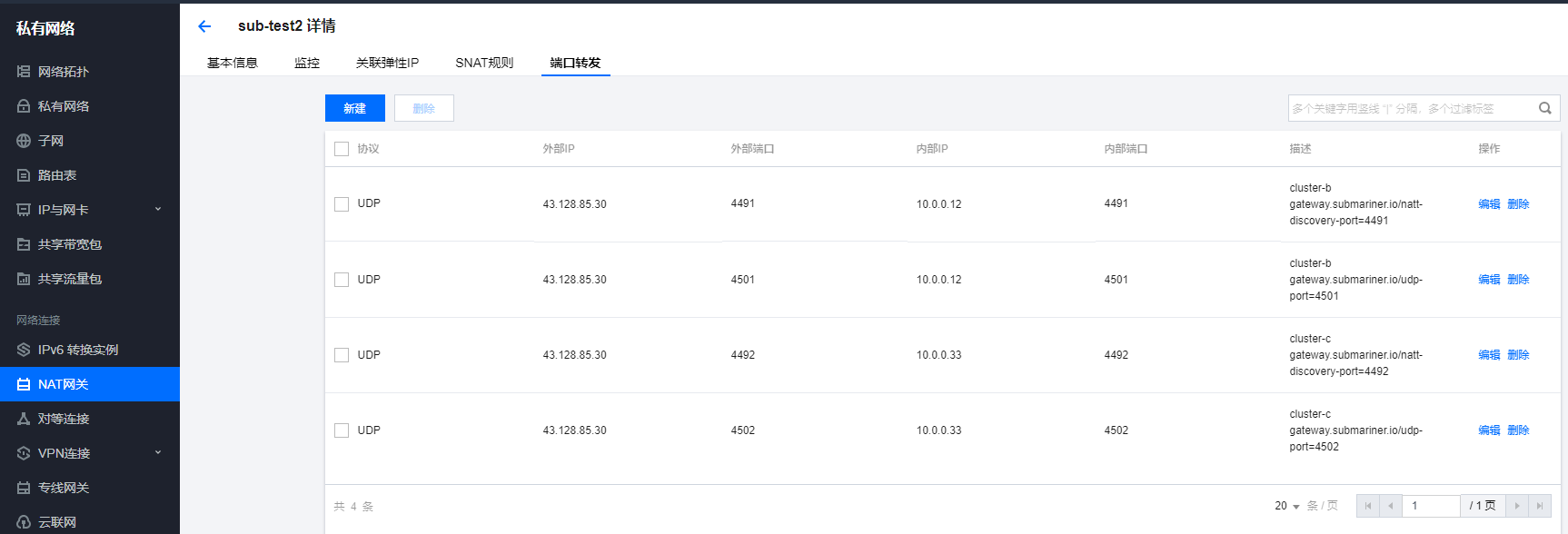
正常执行脚本cluster-a7.sh cluster-b7.sh cluster-c7.sh cluster-d7.sh

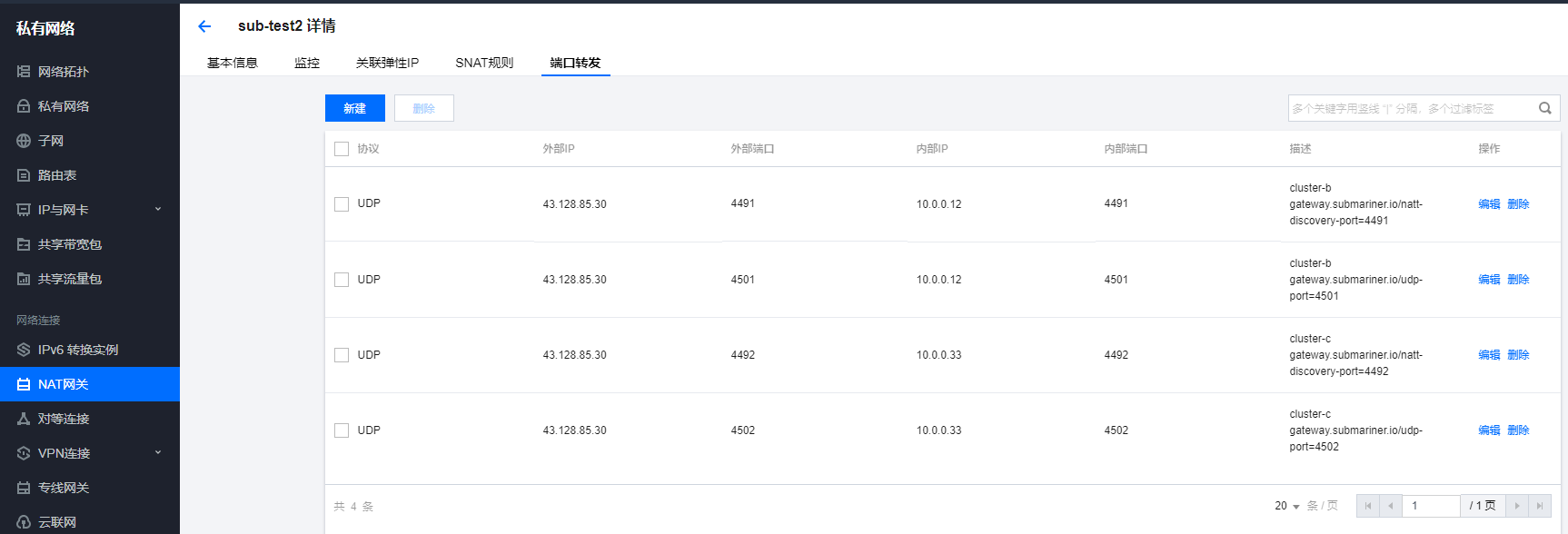
然后四个cluster都运行起来并且部署成功，使用subctl show all命令查看连接状态，发现有一些连接是不通的，他们会显示成connecting

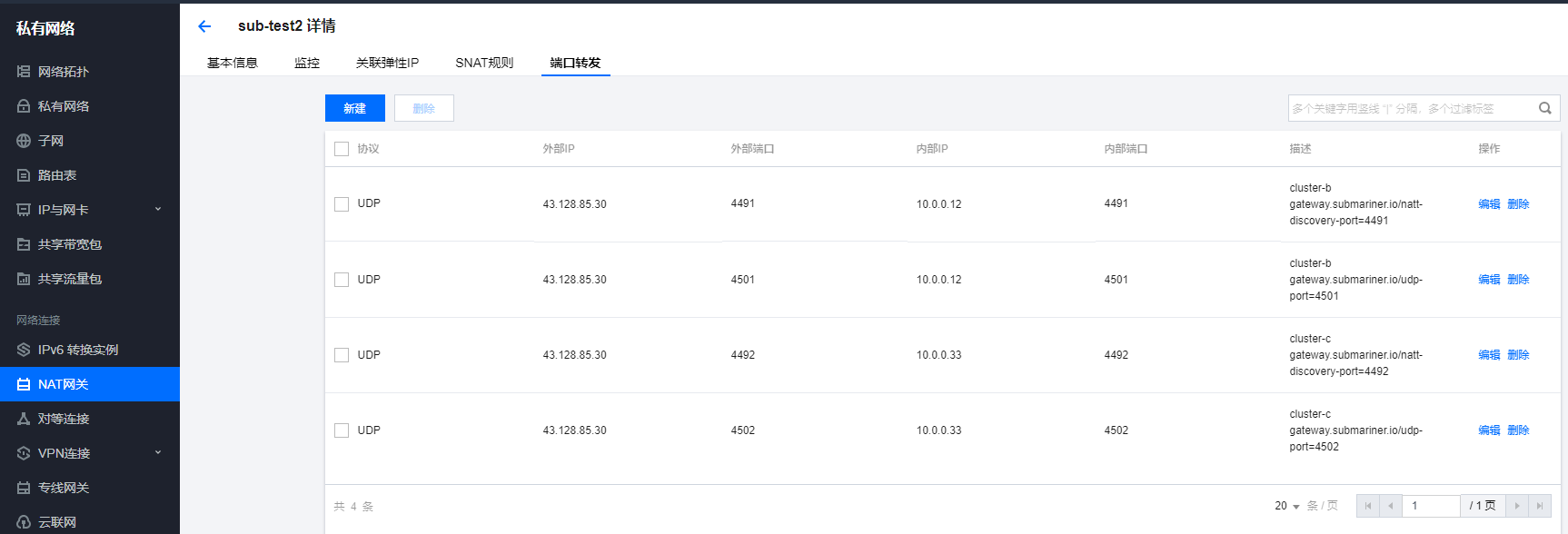
总是会有一些显示成connecting，反正总是会有一些连接是不通的，具体哪个不通确实不完全相同。我做过两个相同的case，有的case是这个连接不通，有的case是另一些连接不通



(3)按照port mapping.sh文件配置在对应的cluster和NAT网关出配置端口映射，配置完成后端口映射关系如下图所示(图片太长，我截成了两段)

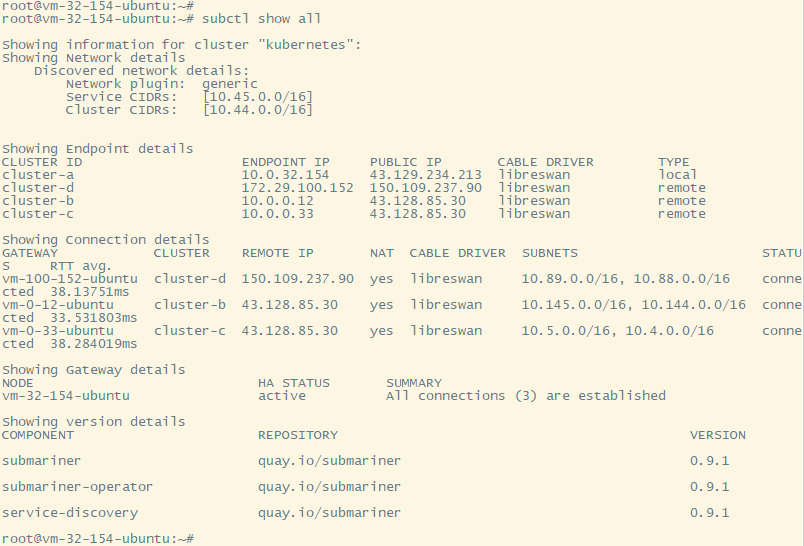






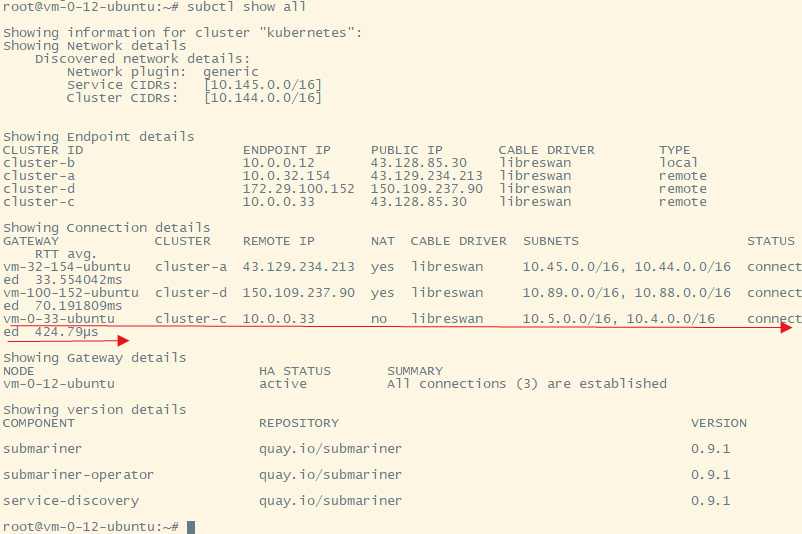
配置完成后，再次检查集群，所有的来连接都正常了

cluster-a

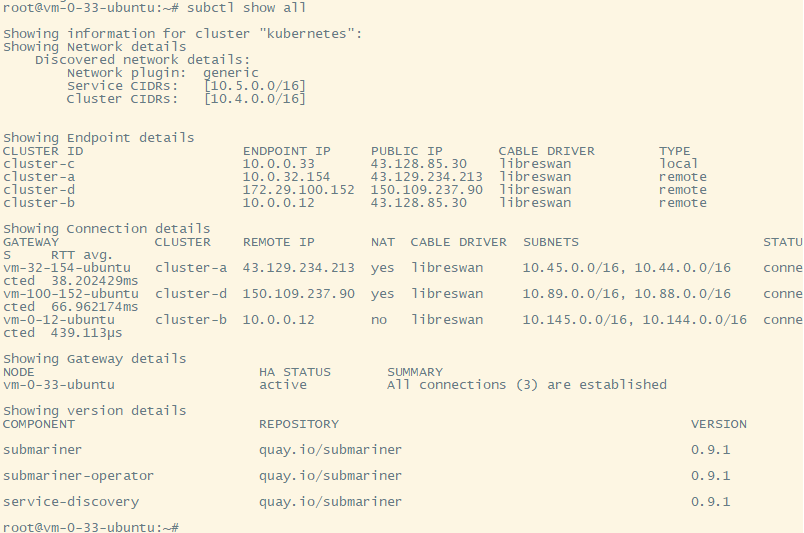


cluster-b

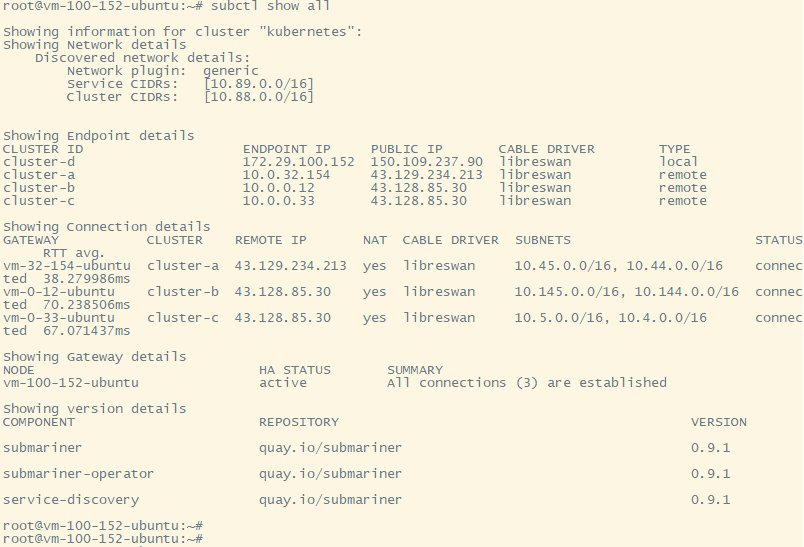
注意cluster-b与cluster-c是不需要经过NAT的，所以这里是no，另外cluster-b和cluster-c从属于同一个内网，所以他们的连接时延特别短

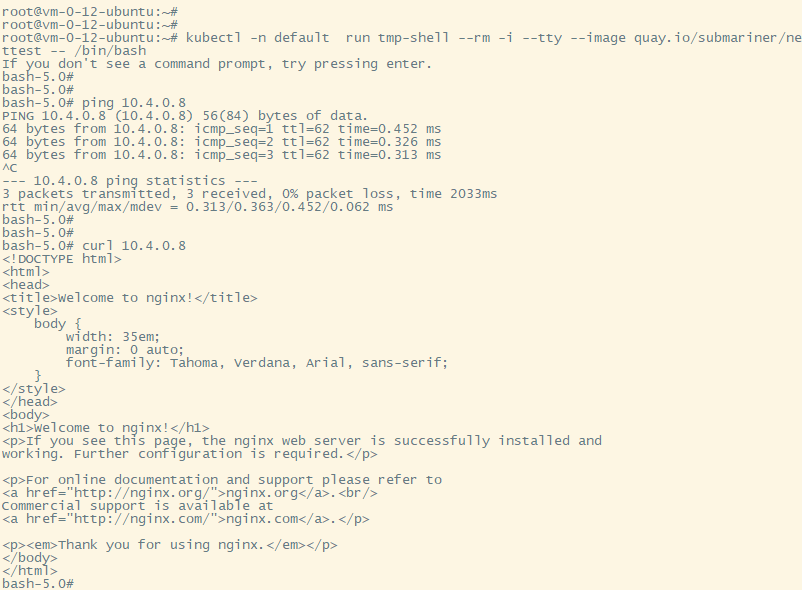


cluster-c



cluster-d





如果你用的subctl v0.9.0 + port mapping，出现有的连接失败的情况，可以不用回滚机器，执行以下命令可以在线升级

kubectl -n submariner-operator set image deployment/submariner-operator submariner-operator=quay.io/submariner/submariner-operator:0.9.1

kubectl patch Submariner submariner -n submariner-operator -p '{"spec": {"version": "0.9.1"}}' --type=merge