如何修改Workspace/Learningspace默认docker网段

孟小涛 2020-11-06 发表

问题描述

由于Workspace产品docker服务器会占用172.17.0.0/16网段，所以网络规划需要避开该网段，以防引起冲突，部署失败。但又是客户出已经规划该网段，因此需要进行docker占用网段调整

解决方法

按照此方法使用reloadDocker-2106tar.gz已验证场景如下：

1、 单机管理平台未初始化，单台主机操作修改docker网段，初始化管理平台成功；增加备机管理节点，增加前操作修改docker网段与主节点一致，双机搭建成功，环境网段正确修改，license和云盘功能正常（E1010P01版本验证）

2、 非双机两主机集群管理平台未初始化，两台主机均操作修改docker网段，初始化管理平台成功，环境网段正确修改，license和云盘功能正常（E1011L06第一轮版本验证）

3、 非双机两主机集群管理平台已初始化，管理节点操作修改docker网段，环境网段正确修改，license和云盘功能正常（E1011分级管理特性版本验证）

4、 双机已搭建完成，主节点修改docker网段，备节点仅添加SNAT规则，环境网段正确修改，license和云盘功能正常（E1011L01版本验证）

【请注意】

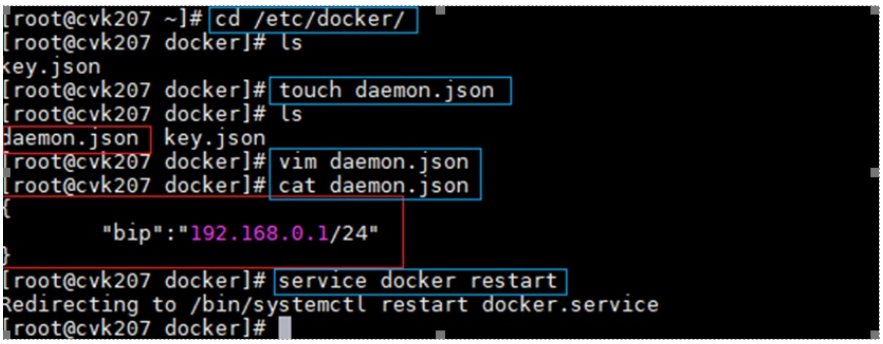
- 规划修改的docker网段需要是环境中未被使用的网段。
- 管理平台未初始化，docker网段修改需要在每台服务器进行修改操作。
- 管理平台已初始化完成，docker网段修改在管理节点进行修改操作即可。
- 管理平台已初始化完成且docker网段已修改，新增节点若作为计算节点，则无需修改docker网段，直接添加即可；新增节点若作为搭建双机热备的节点，则需要执行docker网段修改后再添加，且修改内容与主节点保持一致。
- 双机热备环境搭建已完成，docker网段修改在主节点进行修改操作，备节点仅增加与主节点一致的SNAT规则即可，即执行步骤（4）

【操作步骤:

(1) 后台进入/etc/docker目录，查看是否存在daemon.json，若不存在则新建一个。

(2) vim daemon.json编辑文件内容，修改网段为规划网段并保存（此处以192.168.0.1/24网段为例），cat daemon.json确认修改正确。

(3) service docker restart重启docker服务。



第(4) 执行命令iptables-save -c，在显示的信息中会发现除了默认的172.17.0.0/16 SNAT规则，会多出一条192.168.0.0/24的新增规则：[0:0] -A POSTROUTING -s 192.168.0.0/24 ! -o docker0 -j MASQUERADE。如果在该处查询时没有发现有多出的规则，则执行命令iptables -t nat -A POSTROUTING -s 192.168.0.0/24 ! -o docker0 -j MASQUERADE手动进行增加，然后再次查询。

```
[root@cvk207 docker]# iptables-save -c
# Generated by iptables-save v1.4.21 on Thu Oct 14 09:40:36 2021
*nat
:PREROUTING ACCEPT [648:58424]
:INPUT ACCEPT [3:229]
:OUTPUT ACCEPT [3:180]
:POSTROUTING ACCEPT [3:180]
[2:64] -A POSTROUTING -s 172.17.0.0/16 ! -o docker0 -j MASQUERADE
COMMIT
# Completed on Thu Oct 14 09:40:36 2021
# Generated by iptables-save v1.4.21 on Thu Oct 14 09:40:36 2021
*filter
:INPUT ACCEPT [12961165:2706399479]
:FORWARD ACCEPT [773:46380]
:OUTPUT ACCEPT [22671869:27623443285]
:DOCKER-USER - [0:0]
[554541877:1046971316806] -A INPUT -s 127.0.0.1/32 -p tcp -j ACCEPT
[0:0] -A INPUT -p tcp -m tcp --dport 8080 -m connlimit --connlimit-above 300 --connlimit-mask 32 --connli
mit-saddr -j REJECT --reject-with icmp-port-unreachable
[0:0] -A INPUT -p tcp -m tcp --dport 8443 -m connlimit --connlimit-above 300 --connlimit-mask 32 --connli
mit-saddr -j REJECT --reject-with icmp-port-unreachable
[0:0] -A INPUT -p tcp -m tcp --dport 7070 -j REJECT --reject-with icmp-port-unreachable
[0:0] -A INPUT -p tcp -m tcp --dport 7443 -j REJECT --reject-with icmp-port-unreachable
[773:46380] -A FORWARD -j DOCKER-USER
[925:55388] -A DOCKER-USER -j RETURN
COMMIT
# Completed on Thu Oct 14 09:40:36 2021
```

```
[root@cvk207 docker]# iptables -t nat -A POSTROUTING -s 192.168.0.0/24 ! -o docker0 -j MASQUERADE
[root@cvk207 docker]# iptables-save -c
# Generated by iptables-save v1.4.21 on Thu Oct 14 09:42:11 2021
*nat
:PREROUTING ACCEPT [688:56045]
:INPUT ACCEPT [1:73]
:OUTPUT ACCEPT [2:120]
:POSTROUTING ACCEPT [2:120]
[2:64] -A POSTROUTING -s 172.17.0.0/16 ! -o docker0 -j MASQUERADE
[0:0] -A POSTROUTING -s 192.168.0.0/24 ! -o docker0 -j MASQUERADE
COMMIT
# Completed on Thu Oct 14 09:42:11 2021
# Generated by iptables-save v1.4.21 on Thu Oct 14 09:42:11 2021
*filter
:INPUT ACCEPT [12976523:2710280247]
:FORWARD ACCEPT [773:46380]
:OUTPUT ACCEPT [22709084:27671586003]
:DOCKER-USER - [0:0]
[554564609:1047014584310] -A INPUT -s 127.0.0.1/32 -p tcp -j ACCEPT
[0:0] -A INPUT -p tcp -m tcp --dport 8080 -m connlimit --connlimit-above 300 --connlimit-mask 32 --connli
mit-saddr -j REJECT --reject-with icmp-port-unreachable
[0:0] -A INPUT -p tcp -m tcp --dport 8443 -m connlimit --connlimit-above 300 --connlimit-mask 32 --connli
mit-saddr -j REJECT --reject-with icmp-port-unreachable
[0:0] -A INPUT -p tcp -m tcp --dport 7070 -j REJECT --reject-with icmp-port-unreachable
[0:0] -A INPUT -p tcp -m tcp --dport 7443 -j REJECT --reject-with icmp-port-unreachable
[773:46380] -A FORWARD -j DOCKER-USER
```

(5) vim /etc/cvm/iptables.roles，将文件中默认的172.17.0.0/16的那条规则修改为步骤(4)查询看到的新增的规则（如果是双机热备环境，备机也需要检查下该文件，同步修改保存）。

```
[root@cvk207 docker]# vim /etc/cvm/iptables.roles
# Generated by iptables-save v1.4.21 on Thu Sep 16 14:13:22 2021
*nat
:PREROUTING ACCEPT [4:380]
:INPUT ACCEPT [4:380]
:OUTPUT ACCEPT [1:60]
:POSTROUTING ACCEPT [1:60]
-A POSTROUTING -s 192.168.0.0/24 ! -o docker0 -j MASQUERADE
COMMIT
# Completed on Thu Sep 16 14:13:22 2021
# Generated by iptables-save v1.4.21 on Thu Sep 16 14:13:22 2021
*filter
:INPUT ACCEPT [286:36249]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [1440:171079]
:DOCKER-USER - [0:0]
-A INPUT -s 127.0.0.1/32 -p tcp -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8080 -m connlimit --connlimit-above 300 --connlimit-mask 32 --connlimit-sa
ddr -j REJECT --reject-with icmp-port-unreachable
-A INPUT -p tcp -m tcp --dport 8443 -m connlimit --connlimit-above 300 --connlimit-mask 32 --connlimit-sa
ddr -j REJECT --reject-with icmp-port-unreachable
-A INPUT -p tcp -m tcp --dport 7070 -j REJECT --reject-with icmp-port-unreachable
-A INPUT -p tcp -m tcp --dport 7443 -j REJECT --reject-with icmp-port-unreachable
-A FORWARD -j DOCKER-USER
-A DOCKER-USER -j RETURN
COMMIT
# Completed on Thu Sep 16 14:13:22 2021
```

(6) 执行命令docker ps | grep vdi-lcp查询vdi-lcp版本号。根据版本号获取对应的运行脚本包reloadDocker-xxx tar.gz（xxx代表vdi-lcp版本号），例如：如果是2106版本，使用reloadDocker-2106.tar.gz，如果是2103版本，使用reloadDocker-2103.tar.gz。

```
[root@cvk207 docker]# docker ps | grep vdi-lcp
752d3741522        license/liccmgr-x86:2106   "/bin/sh -c 'java ${…"   3 weeks ago         Up 56 second
      0.0.0.0:18080->8080/tcp   vdi-lcp
```

(7) 将获取的reloadDocker-xxx.tar.gz文件上传到CVM主机（此处以/root路径为例，双机热备环境上传主节点即可）。

(8) 进入文件上传所在路径，执行命令chmod 777 reLoadMain.sh.x修改 reLoadMain.sh.x 文件权限。

(9) 执行脚本./ reLoadMain.sh.x。

1

```
[root@cvk207 docker]# cd /root/
[root@cvk207 ~]# ls
anaconda-ks.cfg  original-ks.cfg  reloadDocker-2106.tar.gz
[root@cvk207 ~]# tar -zxf reloadDocker-2106.tar.gz
[root@cvk207 ~]# ls
anaconda-ks.cfg  original-ks.cfg  reloadDocker-2106  reloadDocker-2106.tar.gz
[root@cvk207 ~]# cd reloadDocker-2106/
[root@cvk207 reloadDocker-2106]# ls
reLoadLcpDocker.sh.x  reLoadMain.sh.x
[root@cvk207 reloadDocker-2106]# chmod 777 reLoadMain.sh.x
[root@cvk207 reloadDocker-2106]# ./reLoadMain.sh.x
**********************************reload masterHost**********************************
---------------------------------------------reload lcp start--------------------------
Redirecting to /bin/systemctl status docker.service
   Active: active (running) since Thu 2021-10-14 10:50:19 CST; 16min ago
docker is running
Redirecting to /bin/systemctl restart docker.service
start reload lcp
4752d3741522        license/liccmgr-x86:2106   "/bin/sh -c 'java ${…"   3 weeks ago         Up Less than
a second   0.0.0.0:18080->8080/tcp   vdi-lcp
lcp docker already loaded, unload first
4752d3741522
lcp docker unloaded
lcp docker start reload
Successfully updated MySQL user's privileges.
8be1400bf13f02da6fe33d196f06fa24838a794f2e271df9124b2ceb969a0239
reload lcp docker done
docker is keep running
---------------------------------------------reload lcp done---------------------------
---------------------------------------------reload nextcloud start--------------------------
nextCloud not loaded before, no need to reload
---------------------------------------------reload nextcloud done---------------------------
**********************************reload masterHost done**********************************

there is no slaveHost
################################################reload done#############################################
##
[root@cvk207 reloadDocker-2106]#
```

(10) ifconfig docker查询docker ip，确认使用网段已更改。

```
[root@cvk207 ~]# ifconfig docker
docker0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.1  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::42:a1ff:fe56:e3f6  prefixlen 64  scopeid 0x20<link>
        ether 02:42:a1:56:e3:f6  txqueuelen 0  (Ethernet)
        RX packets 4102  bytes 538109 (525.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3365  bytes 559683 (546.5 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

(11) 登录管理平台，确认license状态正常，云盘（若启用）正常工作。

附件下载：reloadDocker-2103.tar.gz reloadDocker-2106-20220311.tar.gz