第1章 k8s外部不能访问pod

1、问题描述：

在搭建好的k8s集群内创建的容器，只能在其所在的节点上curl可访问，但是在其他任何主机上无法访问容器占用的端口

1.1、解决方案

vim /etc/sysctl.conf

找到这一行，放开注释

# Uncomment the next line to enable packet forwarding for IPv4

net.ipv4.ip\_forward=1

重启主机（必须要重启才能生效）

第2章 创建私有仓库问题

2.1、问题描述，提示需要https协议问题解决

[root@docker docker]# docker push 10.0.0.10:5000/test/nginx:v1

The push refers to repository [10.0.0.10:5000/test/nginx]

Get https://10.0.0.10:5000/v2/: http: server gave HTTP response to HTTPS client

2.1.1、解决方法1：（docker 1.2以上版本解决方法）

在/etc/docker/daemon.json添加以下信息

{ "insecure-registries":["10.0.0.10:5000"] 必须要加在第一行

重启docker，重启registry

systemctl restart docker.service

2.1.2、解决方法2：（docker1.2以下版本解决方法）

报错信息2：

[root@lnmp ~]# docker pull 10.0.0.10:5000/test/nginx:v1

Error response from daemon: invalid registry endpoint https://10.0.0.10:5000/v0/: unable to ping registry endpoint https://10.0.0.10:5000/v0/

v2 ping attempt failed with error: Get https://10.0.0.10:5000/v2/: tls: oversized record received with length 20527

v1 ping attempt failed with error: Get https://10.0.0.10:5000/v1/\_ping: tls: oversized record received with length 20527. If this private registry supports only HTTP or HTTPS with an unknown CA certificate, please add `--insecure-registry 10.0.0.10:5000` to the daemon's arguments. In the case of HTTPS, if you have access to the registry's CA certificate, no need for the flag; simply place the CA certificate at /etc/docker/certs.d/10.0.0.10:5000/ca.crt

2.2、解决办法：

在/etc/sysconfig/docker中添加如下信息即可

other\_args="--insecure-registry 10.0.0.10:5000" 私有仓库地址

other\_args="--insecure-registry registry:5000" 公有仓库地址

重启docker，重启registry

/etc/init.d/docker restart

第3章 下载镜像出现问题

3.1、问题1：提示/etc/rhsm/ca/redhat-uep.pem no file or dirctory

3.1.1、解决方法：

3.1.1.1、yum安装需要的依赖包

yum -y install \*rhsm\*

1

3.1.1.2、下载python-rhsm-certificates软件并生成密钥文件

wget http://mirror.centos.org/centos/7/os/x86\_64/Packages/python-rhsm-certificates-1.19.10-1.el7\_4.x86\_64.rpm

生成密钥

rpm2cpio python-rhsm-certificates-1.19.10-1.el7\_4.x86\_64.rpm | cpio -iv --to-stdout ./etc/rhsm/ca/redhat-uep.pem | tee /etc/rhsm/ca/redhat-uep.pem

3.1.1.3、重新pull镜像

docker pull registry.access.redhat.com/rhel7/pod-infrastructure:latest

1

第4章 不能删除容器

4.1、docker报错rpc error: code = 14 desc = grpc: the connection is unavailable

4.1.1、尝试关闭容器，进入容器操作界面也报相同错误：

[root@k8s-node-1 ~]# docker exec -it 7119f8f5feef /bin/bash

rpc error: code = 14 desc = grpc: the connection is unavailable

4.1.1.2、停止容器依旧提示错误

[root@k8s-node-1 ~]# docker stop 7119f8f5feef

Error response from daemon: Cannot stop container 7119f8f5feef: Cannot kill container 7119f8f5feef4c649d9ec04734e6224e2d837fa030de271f269f0b71eea29327: rpc error: code = 14 desc = grpc: the connection is unavailable

4.1.1.3、删除容器依旧提示错误（-f强制删除）

[root@k8s-node-1 ~]# docker rm -f 7119f8f5feef

Error response from daemon: Could not kill running container 7119f8f5feef4c649d9ec04734e6224e2d837fa030de271f269f0b71eea29327, cannot remove - Cannot kill container 7119f8f5feef4c649d9ec04734e6224e2d837fa030de271f269f0b71eea29327: rpc error: code = 14 desc = grpc: the connection is unavailable

4.2、解决办法：

4.2.1、使用docker-containerd命令以debug模式调试容器

注意：那个node上的容器不能删除就在那台node上面执行以下命令

[root@k8s-node-1 ~]# docker-containerd -l unix:///var/run/docker/libcontainerd/docker-containerd.sock --metrics-interval=0 --start-timeout 2m --state-dir /var/run/docker/libcontainerd/containerd --shim docker-containerd-shim --runtime docker-runc --debug

WARN[0000] containerd: low RLIMIT\_NOFILE changing to max current=1024 max=4096

DEBU[0000] containerd: read past events count=1

low RLIMIT\_NOFILE changing to max current=1024 max=4096DEBU[0000] containerd: grpc api on /var/run/docker/libcontainerd/docker-containerd.sock

DEBU[0000] containerd: container restored id=354af53914e3f76e653a26d9e9da8d4fbef4ef18cc2176371b89871a9126a646

DEBU[0000] containerd: container restored id=3f0bf43f7ca97c439b64370cee09205b35e58ed35e49f957412f58affbe4ed4b

DEBU[0000] containerd: container restored id=4b848d33a32a332635929b95eb7291abeb32f177a3c65248568b959dbfbc2712

DEBU[0000] containerd: container restored id=4ed8d1f971a0ea5035b507511d802a1445af9e771cde670814104102a7cc2d6f

ERRO[0000] containerd: notify OOM events error=open /proc/13541/cgroup: no such file or directory

DEBU[0000] containerd: container restored id=7119f8f5feef4c649d9ec04734e6224e2d837fa030de271f269f0b71eea29327

ERRO[0000] containerd: notify OOM events error=open /proc/12860/cgroup: no such file or directory

DEBU[0000] containerd: container restored id=7bdba0a1ee81997bdbb5958e31123538ac8a6730c6cc7120fe7359439b52b410

DEBU[0000] containerd: container restored id=8ba79a79836b4350335375f89fc1473a6a86593375fbac6344fb17e4dddff43f

DEBU[0000] containerd: container restored id=9692f3570460186de681476bd068d008891b24b3906f190443f24e97343c3e57

DEBU[0000] containerd: supervisor running cpus=1 memory=977 runtime=docker-runc runtimeArgs=[] stateDir=/var/run/docker/libcontainerd/containerd

DEBU[0000] containerd: process exited id=7119f8f5feef4c649d9ec04734e6224e2d837fa030de271f269f0b71eea29327 pid=init status=143 systemPid=13541

ERRO[0000] containerd: deleting container error=exit status 1: "container 7119f8f5feef4c649d9ec04734e6224e2d837fa030de271f269f0b71eea29327 does not exist\none or more of the container deletions failed\n"

DEBU[0000] containerd: process exited id=7bdba0a1ee81997bdbb5958e31123538ac8a6730c6cc7120fe7359439b52b410 pid=init status=137 systemPid=12860

ERRO[0000] containerd: deleting container error=exit status 1: "container 7bdba0a1ee81997bdbb5958e31123538ac8a6730c6cc7120fe7359439b52b410 does not exist\none or more of the container deletions failed\n"

^CINFO[0056] stopping containerd after receiving interrupt

4.2.2、调试后发现容器状态变为了未开启，尝试删除容器，成功

docker exec -it 3e22bd0b6a40 /bin/bash

Error response from daemon: Container 3e22bd0b6a40c85d2af45b5d65fb3648acab7e0ad05fa909201051a8f00a3d15 is not running

docker rm -f zen\_mclean

zen\_mclean

第5章 k8s下DNS问题

5.1、kubelet提示DNS错误信息

kubelet does not have ClusterDNS IP configured and cannot create Pod using "ClusterFirst" policy. Fail

5.2、解决办法：

在cat /etc/kubernetes/kubelet 配置文件中添加如下内容即可

KUBE\_ARGS="--cluster-dns=10.0.0.110 --cluster-domain=cluster.local"

重启 systemctl daemon-reload; systemctl restart kubelet 即可

第6章 docker run （镜像）报错，文件系统不支持

1、报错信息如下：

/usr/bin/docker-current: Error response from daemon: error creating overlay mount to /var/lib/docker/overlay2/7b4a1ef8a539785fde3fa4cabc4bb9d90967a30calid argument.

See '/usr/bin/docker-current run --help'.

2、报错原因

这个是因为用的overlay2文件系统，而系统默认只能识别overlay文件系统

所以我们就要更新文件系统了

3、解决方法：

systemctl stop docker //停掉docker服务

rm -rf /var/lib/docker //注意会清掉docker images的镜像

vi /etc/sysconfig/docker-storage //将文件里的overlay2改成overlay即可

DOCKER\_STORAGE\_OPTIONS="--storage-driver overlay2 " #修改前

DOCKER\_STORAGE\_OPTIONS="--storage-driver overlay " #修改后

vi /etc/sysconfig/docker //去掉option后面的--selinux-enabled

4、重新启动docker即可

systemctl start docker

1

第7章 docker运行apache报错

7.1、报错信息如下：

[root@k8s-node-3 ~]# docker logs 99e3fc059214

WordPress not found in /var/www/html - copying now...

Complete! WordPress has been successfully copied to /var/www/html

AH00534: apache2: Configuration error: No MPM loaded.

7.2、解决方法：

systemctl stop docker //停掉docker服务

rm -rf /var/lib/docker //注意会清掉docker images的镜像

vi /etc/sysconfig/docker-storage //将文件里的overlay2改成devicemapper即可

DOCKER\_STORAGE\_OPTIONS="--storage-driver overlay2 " #修改前

DOCKER\_STORAGE\_OPTIONS="--storage-driver devicemapper " #修改后

7.3、重启docker服务

systemctl start docker

第8章 启动pod报错信息如下：

[root@k8s\_master k8s\_yaml]# kubectl -n ingress-nginx get events #通过事件查看错误信息

Warning FailedCreate ReplicaSet Error creating: pods "nginx-ingress-controller-9fc7f4c5f-5f2k4" is forbidden: SecurityContext.RunAsUser is forbidden

7m42s Warning FailedCreate ReplicaSet Error creating: pods "nginx-ingress-controller-9fc7f4c5f-25wr7" is forbidden: SecurityContext.RunAsUser is forbidden

8.1、解决办法：

修改apiserver配置文件，将SecurityContextDeny去掉，重启kube-apiserver即可解决