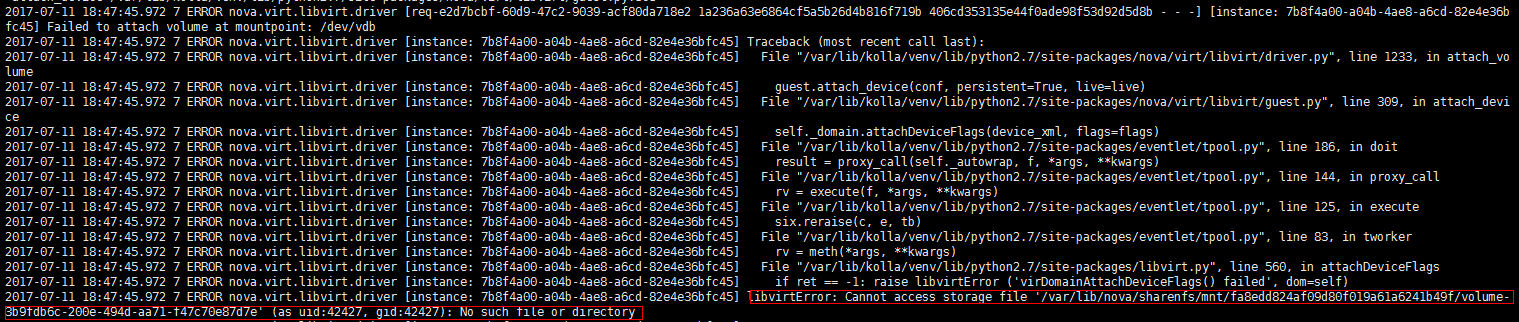
虚机挂载NFS卷失败问题分析

问题背景：

虚机挂载NFS卷失败，错误如下：

代码分析：

nova挂载NFS磁盘的过程如下：

1. nova\_compute请求Cinder获取Volume的connection\_info，对于NFS卷返回的connection\_info信息为：

{

"connection\_info": {

"driver\_volume\_type": "nfs",

"mount\_point\_base": "/var/lib/cinder/mnt",

"data": {

"name": "volume-3b9fdb6c-200e-494d-aa71-f47c70e87d7e",

"format": "raw",

"encrypted": false,

"qos\_specs": null,

"export": "172.24.3.96:/ozb\_nfs\_glance\_volume\_236",

"access\_mode": "rw",

"options": null

}

}

}

1. nova\_compute中的nova.virt.libvirt.volume.nfs.LibvirtNFSVolumeDriver获取connection\_info中的export信息，将该路径mount到CONF.nfs\_mount\_point\_base目录下（默认情形下该目录是/var/lib/nova/mnt）。

核心代码：

def \_ensure\_mounted(self, connection\_info):  
 nfs\_export = connection\_info['data']['export']  
 mount\_path = self.\_get\_mount\_path(connection\_info)  
 """  
 LibvirtNFSVolumeDriver检查/proc/mounts目录是否有已经有nfs\_export挂载的记录  
 """  
 if not libvirt\_utils.is\_mounted(mount\_path, nfs\_export):  
 """  
 nfs\_export如果还没有挂载，调用\_mount\_nfs接口进行挂载，  
 挂载命令大概为：mount -t nfs <nfs\_export> <mount\_path>。  
 """  
 options = connection\_info['data'].get('options')  
 self.\_mount\_nfs(mount\_path, nfs\_export, options, ensure=True)  
 return mount\_path

1. nova\_compute通知libvirt去挂载的NFS目录下寻找VOLUME对应的磁盘文件，并调用libvirt的接口从镜像文件初始化virio设备：

核心代码：

def attach\_device(self, conf, persistent=False, live=False):  
 """  
 libvirt.VIR\_DOMAIN\_AFFECT\_CONFIG:配置是否持久化到XML文件；  
 ibvirt.VIR\_DOMAIN\_AFFECT\_LIVE:配置是否立即生效  
 """  
 flags = persistent and libvirt.VIR\_DOMAIN\_AFFECT\_CONFIG or 0  
 flags |= live and libvirt.VIR\_DOMAIN\_AFFECT\_LIVE or 0  
 device\_xml = conf.to\_xml()  
 if six.PY3 and isinstance(device\_xml, six.binary\_type):  
 device\_xml = device\_xml.decode('utf-8')  
 LOG.debug("attach device xml: %s", device\_xml)  
 """  
 调用libvirt接口将镜像文件挂载到VM，nova\_compute等待该接口的返回值  
 """  
 self.\_domain.attachDeviceFlags(device\_xml, flags=flags)

根因分析：

由于在kolla环境中nova\_compute和nova\_libvirt部署在不同容器中。挂载流程中，nova\_compute在自己容器中执行mount命令挂载存储的NFS目录。挂载完成后，libvirt在nova\_libvirt容器中检查对应目录下的VOLUME文件。两次操作发生在不同目录！

通过手工测试发现在nova\_compute容器中执行mount命令，nova\_libvirt容器中在同样的路径下看不到NFS目录下的文件，和nova\_libvirt打印的错误日志一致：



nova\_libvirt容器中通过mount命令将NFS目录挂到指定地址，对VM执行挂载，nova\_libvirt读写该文件失败：



在容器外将NFS目录挂载到/var/lib/docker/volumes/nova\_compute/\_data/sharenfs/mnt的子目录下，重启nova\_libvirt容器，挂载成功！

结论：

在nova\_compute容器中执行挂载nfs目录后。在物理机上、或者nova\_libvirt容器中都看不到该目录，且当前容器也无法正常读写该nfs目录下的文件。

目前对接NFS后端的手工规避方案：在容器外手工挂载NFS目录到/var/lib/docker/volumes/nova\_compute/\_data/sharenfs/mnt/<xxxxx>路径下。重启 nova\_libvirt容器，该路径会被映射到容器的/var/lib/nova /sharenfs/mnt/<xxxxx>目录中，之后执行挂载即可。