



CloudOS 3.0计算可用域异常的一种排查思路

孙亚华 2021-03-31 发表

组网及说明

版本为3106H01，纳管的cas虚拟化环境

问题描述

Cloud OS前台点击【可用域管理】-【计算可用域】显示计算可用域异常，nova可用域没有记录，如下截图



过程分析

- (1) 首先确认CAS上是否有进行过前台密码的修改，经确认并没修改密码；
- (2) 后台查看计算节点的pod及其他pod，均是正常的；

```
[root@cloudos-01 ~]# pod | grep cas
[[Adefault      cas-proxy-core-rc-vkclx          1/1      Running    0      48d      10.101.4.15      192.168.65.66
default        cas-server-rc-7qitw          1/1      Running    71      1y      10.101.4.3      192.168.65.66
default        cas-server-rc-hr95t          1/1      Running    41      1y      10.101.47.3     192.168.65.65
default        cas-server-rc-p3f5v          1/1      Running    53      1y      10.101.31.2     192.168.65.67
default        cas-host02rc-0twmz           1/1      Running    0       12h     10.101.31.9     192.168.65.67
default        cas-hostrc-wfff5             1/1      Running    0       11s     10.101.31.10    192.168.65.67
[root@cloudos-01 ~]# pod | grep cas
```

- (3) 进入异常可用域对应的计算节点的pod，查看nova-compute.log，发现有连接nova-conductor超时的报错，如下截图；

```
2021-03-25 16:02:43.685 450 WARNING nova.conductor.api [req-5b28c250-d64a-469b-957d-165aac8a0b93 - - - -] Timed out waiting for nova-conductor. Is it running
? Or did this service start before nova-conductor? Reattempting establishment o
f nova-conductor connection...: MessagingTimeout: Timed out waiting for a reply
to message ID 12c7225dad2649ebalfc98370e2c0aad
2021-03-25 16:03:43.696 450 WARNING nova.conductor.api [req-5b28c250-d64a-469b-957d-165aac8a0b93 - - - -] Timed out waiting for nova-conductor. Is it running
? Or did this service start before nova-conductor? Reattempting establishment o
f nova-conductor connection...: MessagingTimeout: Timed out waiting for a reply
to message ID 2d597acc508947df81cd419d5886cfc3
2021-03-25 16:04:43.710 450 WARNING nova.conductor.api [req-5b28c250-d64a-469b-957d-165aac8a0b93 - - - -] Timed out waiting for nova-conductor. Is it running
? Or did this service start before nova-conductor? Reattempting establishment o
f nova-conductor connection...: MessagingTimeout: Timed out waiting for a reply
```

- (4) 进一步查看nova容器是否正常，发现这个节点没有nova容器，而其他如glance、cinder等容器正常。如下截图

```
[root@cloudos-01 nova]# C
[root@cloudos-01 nova]# pod | grep glance
default        glancerc-50255              1/1      Running    0
[root@cloudos-01 nova]# pod | grep cinder
default        cinderrc-tnhjn              1/1      Running    1
[root@cloudos-01 nova]# pod | grep neutron
default        neutronagentrc-nrdlp        1/1      Running    0
default        neutronserverrc-9pb74       1/1      Running    0
[root@cloudos-01 nova]# pod | grep nova
[root@cloudos-01 nova]# rc | grep glance
default        glancerc                    1        1        1        1y
[root@cloudos-01 nova]# rc | grep cinder
default        cinderrc                    1        1        1        1y
```

正常的环境是必须有nova容器的。该容器和计算可用域/nova可用域强相关。正是因为nova容器的丢失导致了前台计算可用域不正常。

解决方法

利用yaml文件，重建nova容器

- (1) 登录主节点后台，cd /opt/bin/confFile-cluster/H3CloudOS目录下，ls可以看到nova-rc的yaml文件；
- (2) 使用 kubectl create -f nova-rc.yaml 命令，重建nova容器；
- (3) pod | grep nova看，已经生成了nova容器；
- (4) 将异常的计算可用域删除重新添加之后，正常。

如下截图

```
root@cloudos-01:~# cd /opt/bin/confFile-cluster/H3CloudOS
root@cloudos-01:~# ls
barbican-rc.yaml  ceilometer-service.yaml  glance-service.yaml  keystone-rc.yaml  memcached-service.yaml  nova-service.yaml  trove-rc.yaml
barbican-service.yaml  cinder-rc.yaml  heat-rc.yaml  keystone-service.yaml  netsecrity-core-rc.yaml  sahara-rc.yaml  trove-service
bingo-rc.yaml  cinder-service.yaml  heat-service.yaml  kintion-rc.yaml  netsecrity-core-nc.yaml  sahara-service.yaml
bingo-service.yaml  compute-core-rc.yaml  image-core-rc.yaml  kintion-service.yaml  neutron-agent-rc.yaml  split-agent-gateway-rc.yaml
cas-proxy-core-rc.yaml  compute-core-service.yaml  image-core-service.yaml  manila-rc.yaml  neutron-rc.yaml  split-agent-gateway-service.yaml
cas-proxy-core-service.yaml  custom  ironic-rc.yaml  manila-service.yaml  neutron-service.yaml  storage-core-rc.yaml
ceilometer-rc.yaml  glance-rc.yaml  ironic-service.yaml  memcached-rc.yaml  nova-rc.yaml  storage-core-service.yaml

root@cloudos-01:~# kubectl create -f nova-rc.yaml
replicationcontroller "novarc" created
root@cloudos-01:~# kubectl get pods
NAME                                READY     STATUS    RESTARTS   AGE
novarc-sn4n1                        1/1       Running   0           1m
root@cloudos-01:~# kubectl get pods -o wide
NAME                                READY     STATUS    RESTARTS   AGE       IP            NODE
novarc-sn4n1                        1/1       Running   0           1m        10.101.31.10  cloudos-01
root@cloudos-01:~# kubectl get pods -o wide
NAME                                READY     STATUS    RESTARTS   AGE       IP            NODE
novarc-sn4n1                        1/1       Running   0           1m        10.101.31.10  cloudos-01
```



