

11gR2 RAC 新特性之 Highly Available IP (HAIP)

付思明 (Simon Fu) |Senior Support Engineer
Oracle Global Software Support

在 11g 之前的版本,为了使单块私网网卡的 NIC down 引起节点踢出的可能性最小化,通常要依赖与 OS 厂商的 bonding、trunking、teaming 等类似技术将冗余的网卡绑定在一起使用。从 11.2.0.2 版本开始, Oracle Clusterware 提供了通过 ip 故障切换来保障内部连接冗余的一体化解决方案。

多个私网网卡可以在安装阶段定义,也可以在后来使用 oifcfg 更改。ora.cluster_interconnect.haip 资源将从 “link-local” IP 范围 (169.254.*.*) 中为每个私网网卡选取一个高可用的虚 IP (HAIP)。默认地,私网流量会在所有活动的内联网卡上进行负载均衡,如果一个私网网卡损坏或者无法通信, Oracle GI 软件会透明地将相应的 HAIP 地址移到其中一个剩余的在工作的网卡上面。相比于第三方网卡绑定技术,在提供高可用性的同时又有效利用了带宽。

即使定义了更多的私网网卡, GI 软件最大能够激活的私网网卡数目为四个,而集群实际使用的 HAIP 地址数则取决于集群中最先启动的节点中激活的私网网卡数目。所以如果增加私网网卡,需要重启所有节点的 clusterware 才会生效。

以下通过示例演示:1、使用 HAIP 的好处; 2、想让私网连接继续使用固定 ip 的情况下如何配置。

示例环境:

两节点 11.2.0.4 RAC, 公网网卡 eth0 (192.168.56.0); 私网网卡 eth1 (192.168.132.0)、待加入私网网卡 eth2(192.168.188.0)

```
[grid@cluster1 ~]$ oifcfg iflist -p -n
eth0 192.168.56.0 PRIVATE 255.255.255.0<===公网
eth1 192.168.132.0 PRIVATE 255.255.255.0<===私网
eth1 169.254.0.0 UNKNOWN 255.255.0.0<===HAIP
eth2 192.168.188.0 PRIVATE 255.255.255.0<===未用

[grid@cluster1 ~]$ oifcfg getif
eth0 192.168.56.0 global public
eth1 192.168.132.0 global cluster_interconnect

=====
```

示例 1: 新增 HAIP

将 eth2 网卡加入 cluster_interconnect

```
[grid@cluster1 ~]$ oifcfg setif -global eth2/192.168.188.0:cluster_interconnect

[grid@cluster1 ~]$ oifcfg getif
eth0 192.168.56.0 global public
eth1 192.168.132.0 global cluster_interconnect
eth2 192.168.188.0 global cluster_interconnect
```

从 ifconfig 输出可以看到, 现在并未在 eth2 分配相应的 haip:

```
[root@cluster1 ~]# ifconfig -a
...
eth1 Link encap:Ethernet HWaddr 08:00:27:E0:69:0D
    inet addr:192.168.132.21 Bcast:192.168.132.255 Mask:255.255.255.0
    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
    RX packets:69340 errors:0 dropped:0 overruns:0 frame:0
```

```

TX packets:68488 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:48254901 (46.0 MiB) TX bytes:44633200 (42.5 MiB)
eth1:1 Link encap:Ethernet HWaddr 08:00:27:E0:69:0D
      inet addr:169.254.31.178 Bcast:169.254.255.255 Mask:255.255.0.0 ←HAIP
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
eth2 Link encap:Ethernet HWaddr 08:00:27:64:5D:AA
      inet addr:192.168.188.21 Bcast:192.168.188.255 Mask:255.255.255.0
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
      RX packets:219 errors:0 dropped:0 overruns:0 frame:0
      TX packets:187 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:92054 (89.8 KiB) TX bytes:70836 (69.1 KiB)
lo Link encap:Local Loopback
...

```

只重启一个节点的 CRS，查看 HAIP 变化情况：

```

[root@cluster1 ~]# /grid/bin/crsctl stop cluster
[root@cluster1 ~]# /grid/bin/crsctl stat res -t -init
-----
NAME TARGET STATE SERVER STATE_DETAILS
-----
Cluster Resources
-----
ora.asm
  1 ONLINE ONLINE cluster1 Started
ora.cluster_interconnect.haip
  1 ONLINE ONLINE cluster1
...
[root@cluster1 ~]# ifconfig -a
eth1 Link encap:Ethernet HWaddr 08:00:27:E0:69:0D
      inet addr:192.168.132.21 Bcast:192.168.132.255 Mask:255.255.255.0
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
      RX packets:77845 errors:0 dropped:0 overruns:0 frame:0
      TX packets:75946 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:53536822 (51.0 MiB) TX bytes:48175928 (45.9 MiB)
eth1:1 Link encap:Ethernet HWaddr 08:00:27:E0:69:0D
      inet addr:169.254.31.178 Bcast:169.254.255.255 Mask:255.255.0.0 <==HAIP
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
eth2 Link encap:Ethernet HWaddr 08:00:27:64:5D:AA
      inet addr:192.168.188.21 Bcast:192.168.188.255 Mask:255.255.255.0
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
      RX packets:1194 errors:0 dropped:0 overruns:0 frame:0
      TX packets:989 errors:0 dropped:0 overruns:0 carrier:0

```

```

collisions:0 txqueuelen:1000
RX bytes:689695 (673.5 KiB) TX bytes:381913 (372.9 KiB)
lo Link encap:Local Loopback
...

```

如上可看出，只重启一个节点，在新增的私网上并未分配 HAIP，这是因为集群实际使用的 HAIP 地址数取决于集群中最先启动的节点在 GI 启动时激活的私网网卡数目(节点 1 重启后，集群中最先启动的节点就变成了节点 2，而节点 2 的 GI 启动时激活的私网网卡是 eth1)。

关闭两节点的 CRS，再重启节点 1 的 CRS，可以看到 eth2 网卡上分配了新的 HAIP：

```

[root@cluster1 ~]# /grid/bin/crsctl stop crs
[root@cluster2 ~]# /grid/bin/crsctl stop crs
[root@cluster1 ~]# /grid/bin/crsctl start crs
CRS-4123: Oracle High Availability Services has been started.
[root@cluster1 ~]# ifconfig -a
...
eth1 Link encap:Ethernet HWaddr 08:00:27:E0:69:0D
    inet addr:192.168.132.21 Bcast:192.168.132.255 Mask:255.255.255.0
    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
    RX packets:116266 errors:0 dropped:0 overruns:0 frame:0
    TX packets:89278 errors:0 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:1000
    RX bytes:93365354 (89.0 MiB) TX bytes:54077940 (51.5 MiB)
eth1:1 Link encap:Ethernet HWaddr 08:00:27:E0:69:0D
    inet addr:169.254.73.50 Bcast:169.254.127.255 Mask:255.255.128.0 <===HAIP 地址
    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
eth2 Link encap:Ethernet HWaddr 08:00:27:64:5D:AA
    inet addr:192.168.188.21 Bcast:192.168.188.255 Mask:255.255.255.0
    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
    RX packets:1739 errors:0 dropped:0 overruns:0 frame:0
    TX packets:1622 errors:0 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:1000
    RX bytes:897718 (876.6 KiB) TX bytes:596062 (582.0 KiB)
eth2:1 Link encap:Ethernet HWaddr 08:00:27:64:5D:AA
    inet addr:169.254.244.74 Bcast:169.254.255.255 Mask:255.255.128.0 <===HAIP 地址
    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
lo Link encap:Local Loopback
...

```

示例 2 : HAIP failover/failback 测试

1、关闭节点 2 的 eth2 网卡，观察 eth2 上的 haip 变化

```
[root@cluster2 ~]# ifdown eth2
```

```
[root@cluster2 ~]# ifconfig -a
```

```
...
```

```
eth1      Link encap:Ethernet  HWaddr 08:00:27:4A:B0:4F
          inet addr:192.168.132.22  Bcast:192.168.132.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:138719 errors:0 dropped:0 overruns:0 frame:0
          TX packets:141121 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:102191125 (97.4 MiB)  TX bytes:109113160 (104.0 MiB)
```

```
eth1:1    Link encap:Ethernet  HWaddr 08:00:27:4A:B0:4F
          inet addr:169.254.33.35  Bcast:169.254.127.255  Mask:255.255.128.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

```
eth2      Link encap:Ethernet  HWaddr 08:00:27:54:47:3F  <===网卡已不在 running 状态
          BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:3610 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3622 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1485795 (1.4 MiB)  TX bytes:1566270 (1.4 MiB)
```

```
eth2:1    Link encap:Ethernet  HWaddr 08:00:27:54:47:3F  <===haip 尚未 failover
          inet addr:169.254.178.66  Bcast:169.254.255.255  Mask:255.255.128.0
          BROADCAST MULTICAST  MTU:1500  Metric:1
```

```
...
```

```
[root@cluster2 ~]# ifconfig -a
```

```
...
```

```
eth1      Link encap:Ethernet  HWaddr 08:00:27:4A:B0:4F
          inet addr:192.168.132.22  Bcast:192.168.132.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:149474 errors:0 dropped:0 overruns:0 frame:0
          TX packets:154607 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:108602615 (103.5 MiB)  TX bytes:118778807 (113.2 MiB)
```

```
eth1:1    Link encap:Ethernet  HWaddr 08:00:27:4A:B0:4F
          inet addr:169.254.33.35  Bcast:169.254.127.255  Mask:255.255.128.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

```
eth1:2    Link encap:Ethernet  HWaddr 08:00:27:4A:B0:4F  <===再次查看，发现原来在 eth2 上的 haip 已经 failover
          inet addr:169.254.178.66  Bcast:169.254.255.255  Mask:255.255.128.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

```
eth2      Link encap:Ethernet  HWaddr 08:00:27:54:47:3F
          BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:3610 errors:0 dropped:0 overruns:0 frame:0
```

```

TX packets:3622 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:1485795 (1.4 MiB) TX bytes:1566270 (1.4 MiB)
lo    Link encap:Local Loopback
...

```

查看另一节点，原来在 eth2 上的 haip 同样 failover 了：

```

[root@cluster1 ~]# ifconfig -a
...
eth1    Link encap:Ethernet  HWaddr 08:00:27:E0:69:0D
        inet addr:192.168.132.21  Bcast:192.168.132.255  Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:154975 errors:0 dropped:0 overruns:0 frame:0
        TX packets:149623 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:119027703 (113.5 MiB)  TX bytes:108719355 (103.6 MiB)
eth1:1  Link encap:Ethernet  HWaddr 08:00:27:E0:69:0D
        inet addr:169.254.73.50  Bcast:169.254.127.255  Mask:255.255.128.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
eth1:2  Link encap:Ethernet  HWaddr 08:00:27:E0:69:0D <===节点 1 的 eth2 上的 haip 也一样 failover 了。
        inet addr:169.254.244.74  Bcast:169.254.255.255  Mask:255.255.128.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
eth2    Link encap:Ethernet  HWaddr 08:00:27:64:5D:AA
        inet addr:192.168.188.21  Bcast:192.168.188.255  Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:3650 errors:0 dropped:0 overruns:0 frame:0
        TX packets:3578 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:1570768 (1.4 MiB)  TX bytes:1480331 (1.4 MiB)
lo    Link encap:Local Loopback
...

```

2、再恢复 eth2 网卡的运行，观察 haip 的变化

```

[root@cluster2 ~]# ifup eth2
几秒钟后，
[root@cluster2 ~]# ifconfig -a
...
eth1    Link encap:Ethernet  HWaddr 08:00:27:4A:B0:4F
        inet addr:192.168.132.22  Bcast:192.168.132.255  Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:155315 errors:0 dropped:0 overruns:0 frame:0
        TX packets:162464 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:112143790 (106.9 MiB)  TX bytes:124908274 (119.1 MiB)
eth1:1  Link encap:Ethernet  HWaddr 08:00:27:4A:B0:4F
        inet addr:169.254.33.35  Bcast:169.254.127.255  Mask:255.255.128.0

```

```

UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
eth2    Link encap:Ethernet  HWaddr 08:00:27:54:47:3F
        inet addr:192.168.188.22  Bcast:192.168.188.255  Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:4338 errors:0 dropped:0 overruns:0 frame:0
        TX packets:4351 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:1667591 (1.5 MiB)  TX bytes:1726237 (1.6 MiB)
eth2:1  Link encap:Ethernet  HWaddr 08:00:27:54:47:3F  <==haip failback 回来
        inet addr:169.254.178.66  Bcast:169.254.255.255  Mask:255.255.128.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
lo      Link encap:Local Loopback
...
[root@cluster1 ~]# ifconfig -a
...
eth1    Link encap:Ethernet  HWaddr 08:00:27:E0:69:0D
        inet addr:192.168.132.21  Bcast:192.168.132.255  Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:170566 errors:0 dropped:0 overruns:0 frame:0
        TX packets:161724 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:128815063 (122.8 MiB)  TX bytes:114950993 (109.6 MiB)
eth1:1  Link encap:Ethernet  HWaddr 08:00:27:E0:69:0D
        inet addr:169.254.73.50  Bcast:169.254.127.255  Mask:255.255.128.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
eth2    Link encap:Ethernet  HWaddr 08:00:27:64:5D:AA
        inet addr:192.168.188.21  Bcast:192.168.188.255  Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:4714 errors:0 dropped:0 overruns:0 frame:0
        TX packets:4638 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:1811047 (1.7 MiB)  TX bytes:1755341 (1.6 MiB)
eth2:1  Link encap:Ethernet  HWaddr 08:00:27:64:5D:AA
        inet addr:169.254.244.74  Bcast:169.254.255.255  Mask:255.255.128.0<==节点 1 同样,haip 也 failback 回 eth2
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
lo      Link encap:Local Loopback
...

```

通过以上测试可见，HAIP 在同时利用多个私网网卡带宽提升私网性能的同时，又具备了类似于 OS 层网卡绑定解决方案的高可用性。

示例 3：跳过 HAIP 启动集群数据库

在有些客户环境下，私网是通过 VLAN 划出来的，而出于网络管理要求，要求 VLAN 的 ip 地址与网卡必须是绑定的，私网 ip 也必须是固定的 ip 地址（虽然按 Oracle RAC 的安装要求，私网应该是独立隔离的网络），这时 HAIP 会无法分配，导致依赖它的 ASM 资源无法启动。这种情况下，在安装时执行 root.sh 时（或升级执行 rootupgrade.sh 时会报出 ora.cluster_interconnect.haip 无法启动的错误），查看集群状态如下所示。

```
[root@cluster1 bin]# ./crsctl start crs
CRS-4123: Oracle High Availability Services has been started.
[root@cluster1 bin]# ./crsctl stat res -t
CRS-4535: Cannot communicate with Cluster Ready Services
CRS-4000: Command Status failed, or completed with errors.
[root@cluster1 bin]# ./crsctl stat res -t -init
-----
NAME          TARGET STATE     SERVER          STATE_DETAILS
-----
Cluster Resources
-----
ora.asm
  1 ONLINE OFFLINE Instance Shutdown
ora.cluster_interconnect.haip<===OFFLINE
  1 ONLINE OFFLINE
ora.crf
  1 ONLINE ONLINE cluster1
ora.crsd
  1 ONLINE OFFLINE
ora.cssd
  1 ONLINE ONLINE cluster1
ora.cssdmonitor
  1 ONLINE ONLINE cluster1
ora.ctssd
  1 ONLINE ONLINE cluster1 ACTIVE:0
ora.diskmon
  1 OFFLINE OFFLINE
ora.drivers.acfs
  1 ONLINE OFFLINE
ora.evmd
  1 ONLINE INTERMEDIATE cluster1
ora.gipcd
  1 ONLINE ONLINE cluster1
ora.gpnpd
  1 ONLINE ONLINE cluster1
ora.mdnsd
  1 ONLINE ONLINE cluster1
```

这种情况下可以通过如下简单修改，使 RAC 的私网通信以之前版本的方式来运行。

1. 手工创建 pfile 及 spfile 启动 ASM 实例:

```
[root@cluster1 bin]# su - grid
[grid@cluster1 ~]$ export ORACLE_SID=+ASM1
[grid@cluster1 ~]$ cd /grid/dbs
[grid@cluster1 dbs]$ cat /etc/hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1        localhost.localdomain localhost
192.168.56.21    cluster1.fsm.com      cluster1
192.168.56.22    cluster2.fsm.com      cluster2
192.168.56.23    cluster3.fsm.com      cluster3
192.168.188.21   cluster1-pub.fsm.com   cluster1-pub
192.168.188.22   cluster2-pub.fsm.com   cluster2-pub
192.168.188.23   cluster3-pub.fsm.com   cluster3-pub
192.168.132.21   cluster1-priv.fsm.com  cluster1-priv
192.168.132.22   cluster2-priv.fsm.com  cluster2-priv
192.168.132.23   cluster3-priv.fsm.com  cluster3-priv
192.168.56.26    cluster1-vip.fsm.com    cluster1-vip
192.168.56.27    cluster2-vip.fsm.com    cluster2-vip
192.168.56.29    cluster3-vip.fsm.com    cluster3-vip
192.168.188.26   cluster1-vip2.fsm.com   cluster1-vip2
192.168.188.27   cluster2-vip2.fsm.com   cluster2-vip2
192.168.188.28   cluster3-vip2.fsm.com   cluster3-vip2
192.168.56.28    cluster-scan
```

1) 手工创建 pfile

```
[grid@cluster1 dbs]$ vi init+ASM1.ora
+ASM1.asm_diskgroups='DATA'#Manual Mount
+ASM2.asm_diskgroups='DATA'#Manual Mount
*.asm_diskstring='/dev/oracleasm/disks/*'
*.asm_power_limit=1
*.diagnostic_dest='/home/grid/app/grid'
*.instance_type='asm'
*.large_pool_size=12M
*.remote_login_passwordfile='EXCLUSIVE'
+ASM1.cluster_interconnects = '192.168.132.21'
+ASM2.cluster_interconnects = '192.168.132.22'
```

2) 生成 spfile , 在 11gR2 GI ASM 中 create spfile 会更新 Gnp profile :

```
[grid@cluster1 dbs]$ sqlplus / as sysasm
SQL> startup pfile='/grid/dbs/init+ASM1.ora'
ASM instance started
SQL> create spfile='+CRS' from pfile;

File created.
SQL> startup force
```

```
SQL>show parameter spfile
```

NAME	TYPE	VALUE

spfile	string	+CRS/cluster-scan/asmparameterfile/registry.253.846655653

2. 确认 cluster_interconnect 的修改已同步到 gnpn profile :

```
[grid@cluster1 dbs]$ asmcmd spget
```

```
+CRS/cluster-scan/asmparameterfile/registry.253.846655653
```

3. 修改 ASM 的启动依赖属性，去掉其对 haip 的启动依赖：

```
[grid@cluster1 bin]$ ./crsctl modify res ora.asm -attr
```

```
"START_DEPENDENCIES='hard(ora.cssd,ora.ctssd)pullup(ora.cssd,ora.ctssd)weak(ora.drivers.acfs)" -init
```

4.修改 haip 资源的自启动设置

```
[root@cluster1 bin]# ./crsctl modify res ora.cluster_interconnect.haip -attr "ENABLED=0" -init
```

5. 重启集群验证:

```
[root@cluster1 bin]# ./crsctl stop crs -f
```

```
[root@cluster1 bin]# ./crsctl start crs
```

```
CRS-4123: Oracle High Availability Services has been started.
```

```
[root@cluster1 bin]# ./crsctl stat res -t -init
```

NAME	TARGET	STATE	SERVER	STATE_DETAILS

Cluster Resources				

ora.asm				
1	ONLINE	ONLINE	cluster1	Started ◀ ASM 成功启动
ora.cluster_interconnect.haip				
1	ONLINE	OFFLINE		
...				

6. 数据库实例的 cluster_interconnects 参数修改:

```
[oracle@cluster1 ~]$ sqlplus / as sysdba
```

```
SQL> alter system set cluster_interconnects='192.168.132.21' scope=spfile sid='mydb1';
```

```
SQL> alter system set cluster_interconnects='192.168.132.22' scope=spfile sid='mydb2';
```

7.重启实例后，检查心跳网络

```
SQL> select * from v$cluster_interconnects;
```

NAME	IP_ADDRESS	IS_SOURCE

eth1	192.168.132.21	NO cluster_interconnects parameter

注：

- 1、 cluster_interconnects 可以设置为使用多个私网（ CLUSTER_INTERCONNECTS = ifn [: ifn] ），这样集群可以在多个私网之间做负载均衡，但由于没有虚拟 IP 而无法实现 failover，与使用 HAIP 的方案相比，缺少了高可用性。
- 2、 Windows 平台在目前版本尚不支持 HAIP 特性，默认就不存在 HAIP 资源，故无需上述修改。