Több adatbázis hálózati szeparációja Single instance és RAC környezetben *(Linux alapokon)*

Tapsonyi Tamás
Informatikai tanácsadó
KSH KÖZPONTI STATISZTIKAI HIVATAL

Probléma felvetés

Ha egy környezetben több adatbázis is fut melyeknek a szeparációját meg kell oldani, akkor az alábbi feladatokkal szembesülünk

Probléma felvetés

Ha egy környezetben több adatbázis is fut melyeknek a szeparációját meg kell oldani, akkor az alábbi feladatokkal szembesülünk

- Tárolás

Probléma felvetés

Ha egy környezetben több adatbázis is fut melyeknek a szeparációját meg kell oldani, akkor az alábbi feladatokkal szembesülünk

- Tárolás
- Process/memória

Probléma felvetés

Ha egy környezetben több adatbázis is fut melyeknek a szeparációját meg kell oldani, akkor az alábbi feladatokkal szembesülünk

- Tárolás
- Process/memória
- Hálózat

Tárolás szeparációja

Tárolás szeparációja

- Hagyományos FS jogosítvány kezeléssel
  - Csak Single Instance

Tárolás szeparációja

- Hagyományos FS jogosítvány kezeléssel
  - Csak Single Instance
- ASM diskgroup-okkal és azokon beállított jogosítvány kezeléssel
  - RAC
  - Single Instance

Process/memória szeparációja

- Különböző futtató user alatt az operációs rendszer szolgáltatta elkülönítés megoldja ez a problémánkat.
  - Továbbá erőforrás kontrollra is van lehetőségünk

Hálózati hozzáférés szeparációja

- Single Instance

Hálózati hozzáférés szeparációja

- Single Instance
  - Adatbázis trigger
    - nem hatékony
    - biztonságilag is aggályos

Hálózati hozzáférés szeparációja

- Single Instance
  - Adatbázis trigger
    - nem hatékony
    - biztonságilag is aggályos
  - Külön Listener más porton vagy IP-n
    - Hálózati módszerekkel tűzfal hatékonyan kezelhető, akár több szinten is

Hálózati hozzáférés szeparációja ..

- RAC
  - Adatbázis Trigger >>>lsd. Mint az önálló példány esetében

Hálózati hozzáférés szeparációja ..

- RAC
  - Adatbázis Trigger >>>lsd. Mint az önálló példány esetében
  - Listener szeparáció
    - Hálózati izolációval nem adatbázis, hanem hálózati (tűzfal) oldalon dől el hogy ki fér hozzá az adatbázishoz.

Listener Szeparáció a felvetődő problémák és megoldásuk

- újabb Scan Listener csak olyan OracleNetworkhöz adható ahol nincs másik Scan

Listener Szeparáció a felvetődő problémák és megoldásuk

- újabb Scan Listener csak olyan OracleNetworkhöz adható ahol nincs másik Scan
- Következmény
  - Másik hálózatban lévő NIC előfeltétel (Trunk VLAN is jó lehet)

Új szegmensben másik scan és vip listenerek lérehozásának lépéséi

Új szegmensben másik scan és vip listenerek lérehozásának lépéséi

- Előfeltétel, hogy a node-okon legyen fent a működő új network

Új szegmensben másik scan és vip listenerek lérehozásának lépéséi

- Előfeltétel, hogy a node-okon legyen fent a működő új network
- oracle Network Interface hozzáadása

Új szegmensben másik scan és vip listenerek lérehozásának lépéséi

- Előfeltétel, hogy a node-okon legyen fent a működő új network
- oracle Network Interface hozzáadása
- Új net definiálása

Új szegmensben másik scan és vip listenerek lérehozásának lépéséi

- Előfeltétel, hogy a node-okon legyen fent a működő új network
- oracle Network Interface hozzáadása
- Új net definiálása
- VIP címek definiálása és elindítása

Új szegmensben másik scan és vip listenerek lérehozásának lépéséi

- Előfeltétel, hogy a node-okon legyen fent a működő új network
- oracle Network Interface hozzáadása
- Új net definiálása
- VIP címek definiálása és elindítása
- VIP listenerek definiálása majd elindítása

Új szegmensben másik scan és vip listenerek lérehozásának lépéséi

- Előfeltétel, hogy a node-okon legyen fent a működő új network
- oracle Network Interface hozzáadása
- Új net definiálása
- VIP címek definiálása és elindítása
- VIP listenerek definiálása majd elindítása
- SCAN cím definiálása, elindítása

Új szegmensben másik scan és vip listenerek lérehozásának lépéséi

- Előfeltétel, hogy a node-okon legyen fent a működő új network
- oracle Network Interface hozzáadása
- Új net definiálása
- VIP címek definiálása és elindítása
- VIP listenerek definiálása majd elindítása
- SCAN cím definiálása, elindítása
- SCAN listener definiálása, elindítása

Új szegmensben másik scan és vip listenerek lérehozásának lépéséi

- Előfeltétel, hogy a node-okon legyen fent a működő új network
- oracle Network Interface hozzáadása
- Új net definiálása
- VIP címek definiálása és elindítása
- VIP listenerek definiálása majd elindítása
- SCAN cím definiálása, elindítása
- SCAN listener definiálása, elindítása
- Adatbázisok konfigurálása hogy csak a megfelelő Local(vip) és Remote(scan) listenerbe regisztráljanak be.

….kiadott parancsok...

oifcfg setif -global eth1/192.168.2.0:private

srvctl add network -netnum 2 -subnet 192.168.2.0/255.255.255.0/eth2

srvctl add vip -node srv1 -netnum 2 -address 192.168.2.21/255.255.255.0/eth2

srvctl add vip -node srv2 -netnum 2 -address 192.168.2.22/255.255.255.0/eth2

srvctl start vip -vip srv1_2

srvctl start vip -vip srv2_2

srvctl add listnener -listener listener_vip2 -endpoints "TCP:1531"

srvctl start listener -listener listener_vip2

srvctl add scan -scanname scan_net2 -netnum 2   ###192.168.2.23 DNS-ben

srvctl add scan_listener -netnum 2 -listner scanlsnr_net2 -endpoints "TCP:1531"

srvctl start scan_listener -netnum 2

Felmerülő kérdések

- a második networkön milyen kliensek érik el az adatbázist?

Felmerülő kérdések

- a második networkön milyen kliensek érik el az adatbázist?
  - Ha a network azonos nincs gond
    - Ez nem jó design, mert a kliens gyakorlatilag mindent elérhet, több szintű kontrollra nincs lehetőség

Felmerülő kérdések

- a második networkön milyen kliensek érik el az adatbázist?
  - Ha a network azonos nincs gond
    - Ez nem jó design, mert a kliens gyakorlatilag mindent elérhet, több szintű kontrollra nincs lehetőség
  - Ha a kliens másik Net-en (nem a Listener hálózati szegmensében) van, akkor kezdődnek a bajok, hiszen akkor route-ra is gondolni kell. Mert az alapértelemezett útvonal az elsődleges hálózati csatolón keresztül van definiálva így a második szegmens interface-én jövő kérés az alapértelmezett úton indulna vissza, az már kernel szinten eldobásra kerül.

Routing..

- Definiálhatunk külön route-ot a kliensek vagy azok LANjai felé
  - *route add -host 192.168.3.3 eth2*
    vagy
  - *route add -net 192.168.3.0 netmask 255.255.255.0 eth2*

Routing..

- Definiálhatunk külön route-ot a kliensek vagy azok LANjai felé
  - *route add -host 192.168.3.3 eth2*
    vagy
  - *route add -net 192.168.3.0 netmask 255.255.255.0 eth2*

Készen is vagyunk.

Routing..

- Definiálhatunk külön route-ot a kliensek vagy azok LANjai felé
  - *route add -host 192.168.3.3 eth2*
    vagy
  - *route add -net 192.168.3.0 netmask 255.255.255.0 eth2*

Készen is vagyunk.

Sajnos nem, mert azokkal a kliensekkel mit csinálunk, akiknek mind a két(vagy több) SCAN -en keresztül el kell érniük az adatbázisokat (pl.:Oracle Enterprise Manager Cloude Control)

Ha a static route kevés...

- Használjuk a szabály vagy policy alapú iproute2 által kínált lehetőségeket.

Ha a static route kevés...

- Használjuk a szabály vagy policy alapú iproute2 által kínált lehetőségeket.
  - Korszerű Linux-okban alapértelmezetten benne van, csak használni kell.

Ha a static route kevés...

- Használjuk a szabály vagy policy alapú iproute2 által kínált lehetőségeket.
  - Korszerű Linux-okban alapértelmezetten benne van, csak használni kell.
  - El kell készíteni egy másik route-táblát a route-okkal

Ha a static route kevés...

- Használjuk a szabály vagy policy alapú iproute2 által kínált lehetőségeket.
  - Korszerű Linux-okban alapértelmezetten benne van, csak használni kell.
  - El kell készíteni egy másik route-táblát a route-okkal
  - El kell készíteni a szabályokat, hogy mely esetben kell a másik táblát használni

Lépésről-lépésre (1)

- Készítsünk egyedi route táblát mind a két host-on (srv1, srv2)

  echo "22 oranet2" >>/etc/iproute2/rt_tables

- Készítsük el a route-okat és a szabályokat ha

  srv1_net2: 192.168.2.11,
  srv2_net2: 192.168.2.12,
  gw:192.168.2.254,
  vip2_1: 192.168.2.21,
  vip2_2:192.168.2.22,
  scan2 192.168.2.23

Lépésről-lépésre (2)

- Srv1: /etc/sysconfig/network-scricpts/route-eth2:
  192.168.2.0/24 dev eth2 src 192.168.2.11 table oranet2
  default via 192.168.131.254 dev eth2 table oranet2

- Srv1: /etc/sysconfig/network-scripts/rule-eth2:
  from 192.168.2.11/32 table oranet2
  to 192.168.2.11/32 table oranet2
  from 192.168.2.21/32 table oranet2
  to 192.168.2.21/32 table oranet2
  from 192.168.2.22/32 table oranet2
  to 192.168.2.22/32 table oranet2
  from 192.168.2.23/32 table oranet2
  to 192.168.2.23/32 table oranet2

Lépésről-lépésre (3)

- Srv2: /etc/sysconfig/network-scricpts/route-eth2:
  192.168.2.0/24 dev eth2 src 192.168.2.12 table oranet2
  default via 192.168.131.254 dev eth2 table oranet2

- Srv2: /etc/sysconfig/network-scripts/rule-eth2:
  from 192.168.2.12/32 table oranet2
  to 192.168.2.12/32 table oranet2
  from 192.168.2.21/32 table oranet2
  to 192.168.2.21/32 table oranet2
  from 192.168.2.22/32 table oranet2
  to 192.168.2.22/32 table oranet2
  from 192.168.2.23/32 table oranet2
  to 192.168.2.23/32 table oranet2

Ráncfelvarrás

- Az így kialakított rendszerünk működik, gyakorlatilag a költöző címekhez való szabályok mind a két szerveren mindig élnek.
- Ugyan problémát nem okoz, de nem szép
- A statikus rule konfigurációs állományokban csak a node címeihez tartozó szabályok maradjanak
- Készítsük el a szabály hozzáadását és törlését vezérlő scripteket és a szervereken helyezzük el (esetleg ACFS-re)
- Készítsünk Cluster erőforrásokat amik a vip/scan címekhez kötöttek

Cluster szolgáltatás regisztrációja

- Ha például az új vip2-es szabály beállító scriptünk /usr/local/sbin/vip_2_1_ip2_rule akkor
  *crsctl add resource vip2_1_ip2_rule -type generic_application *
  -attr "START_PROGRAM='/usr/local/sbin/vip2_1_ip2_rule add',
  STOP_PROGRAM='sudo /usr/local/sbin/vip2_1_ip2_rule del',
  CLEAN_PROGRAM='sudo /usr/local/sbin/vip2_1_ip2_rule del',
  CHECK_PROGRAM='sudo /usr/local/sbin/vip2_1_ip2_rule status',
  START_DEPENDENCIES='hard(ora.srv1_2.vip)'\,
  STOP_DEPENDENCIES='hard(ora.srv1_2.vip)'"
- A másik VIP és SCAN címmel hasonlóan járunk el.

Vezérlő vip2_1_ip2_rule script tartalma

```
case $1 in
start)
sudo ip rule add from 192.168.2.21/32 table oranet2
sudo ip rule add to 192.168.2.21/32 table oranet2
;;
stop)
sudo ip rule del from 192.168.2.21/32 table oranet2
sudo ip rule del to 192.168.2.21/32 table oranet2
;;
esac
```

Néhány megjegyzés

- Az ismertetett parancsok EnterpriseLinux 6 és 7 verziókon lettek kipróbálva, de kisebb módosításokkal minden korszerű linux-on futnak
- A Clusterware parancsai 12.1.0.2-es verzión lettek tesztetlve, de ismereteim szerint ezek a parancsok nem változtak, vélhetőleg 18c-n is futnak.

Jó ha kéznél van

- Clusterware Administration and Deployment Guide
- Database Administrator's Guide
- Policy Routing With Linux (Matthew G. Marsh)
- Linux Advanced Routing Mini HOWTO

Köszönöm a megtisztelő figyelmet!