

Grid Infrastructure 11.2 bietet mit Single Client Access Name (SCAN) ein sehr nützliches und effizientes Feature. Der Artikel stellt nicht einfach nur dieses Feature vor, schließlich ist es nicht mehr brandneu und wurde bei zahlreichen Oracle-Kunden bereits vor mehreren Jahren installiert.

# Grid Infrastructure 11.2: Implementierung einer „Private“-Virtual-IP parallel zu SCAN-Virtual-IPs

Yann Neuhaus, dbi services

Der wichtigste Vorteil von SCAN-Adressen und SCAN-Listnern besteht darin, dass sie Oracle-Clients-Datenbanken (RAC oder Single) ansprechen können, die unter Grid Infrastructure 11.2 „gehostet“ werden, wobei im Oracle\*Net-Alias nur eine Adresse definiert ist (siehe Listing 1).

Über diese Adresse kann die Verbindung zu mehreren Knoten hergestellt

werden, selbst wenn dem Cluster zusätzliche Server hinzugefügt wurden. Im Hintergrund wird die Verbindung auf dem DNS nach dem Round-Robin-Prinzip unter Verwendung von SCAN VIPs (VIP steht in diesem Artikel für Virtual IP) verwaltet.

SCAN kann jedoch nicht alle Probleme lösen. Eine Grenze wird beispielsweise bei dem Versuch erreicht,

die Datenbank von zwei verschiedenen Netzwerken aus anzusprechen. Die Default-Datenbank (Public) ist über die SCAN-Infrastruktur absolut problemlos erreichbar (SCAN VIPs und SCAN-Listener). Wie steht es jedoch mit einem zusätzlichen „privaten“ Netzwerk, das von bestimmten spezifischen Clients oder Geräten genutzt wird (siehe Abbildung 1)?

## Voraussetzungen und Ziele

Unsere Infrastruktur besteht aus einem Grid Infrastructure 11.2.0.3 Cluster (Knoten1/Knoten2), wobei zwei Knoten eine kundenspezifische Failover-Datenbank-Ressource namens „ERP-DB“ hosten. Diese Datenbank kann von den Clients über das Netzwerk 172.30.16.0/20 (unser „Public“-Netzwerk) und über die SCAN-Infrastruktur (SCAN VIPs/Listener) angesprochen werden. Parallel dazu müssen sich einige zusätzliche Geräte über das Netzwerk 192.168.4.0/26 (unser „private“ Client Netzwerk) in der gleichen Datenbank anmelden. Listing 2 zeigt die aktuelle SCAN-Konfiguration.

## Mögliche Lösungen

Der erste Gedanke zu dieser Problematik geht natürlich dahin, eine zweite SCAN-Infrastruktur im zweiten Netzwerk (192.168.4.0/26) anzulegen. Leider wird diese Vorgehensweise von Oracle nicht unterstützt, was durch die Anfrage SR 3-5093337641 bei My Oracle Support bestätigt wurde. Eigene Tests haben auch den Autor zum gleichen Ergebnis gebracht (siehe dazu Listing 3).

```
ERPDB.DBI.CH =
  (DESCRIPTION =
    (ADDRESS= (PROTOCOL = TCP)(HOST = cluster-scan-v.dbi.ch )
    (PORT = 1521))
    (CONNECT_DATA =
      (SERVER=DEDICATED)
      (SERVICE_NAME = ERPDB.DBI.CH)
    )
  )
```

Listing 1

```
oracle@node1:/home/oracle/ [+ASM1] srvctl config scan
SCAN name: cluster-scan-v.dbi.ch, Network:
1/172.30.16.0/255.255.240.0/bond0
SCAN VIP name: scan1, IP: /cluster-scan-v.dbi.ch/172.30.20.231
SCAN VIP name: scan2, IP: /cluster-scan-v.dbi.ch/172.30.20.232
SCAN VIP name: scan3, IP: /cluster-scan-v.dbi.ch/172.30.20.230
```

Listing 2

```
root@node1:/root/ [+ASM1] srvctl add scan -n cluster-scan-prv-v.dbi.ch -k 2 -S 192.168.4.0/255.255.255.192/bond1
PRCS-1037 : Single Client Access Name VIPs already exist
```

Listing 3

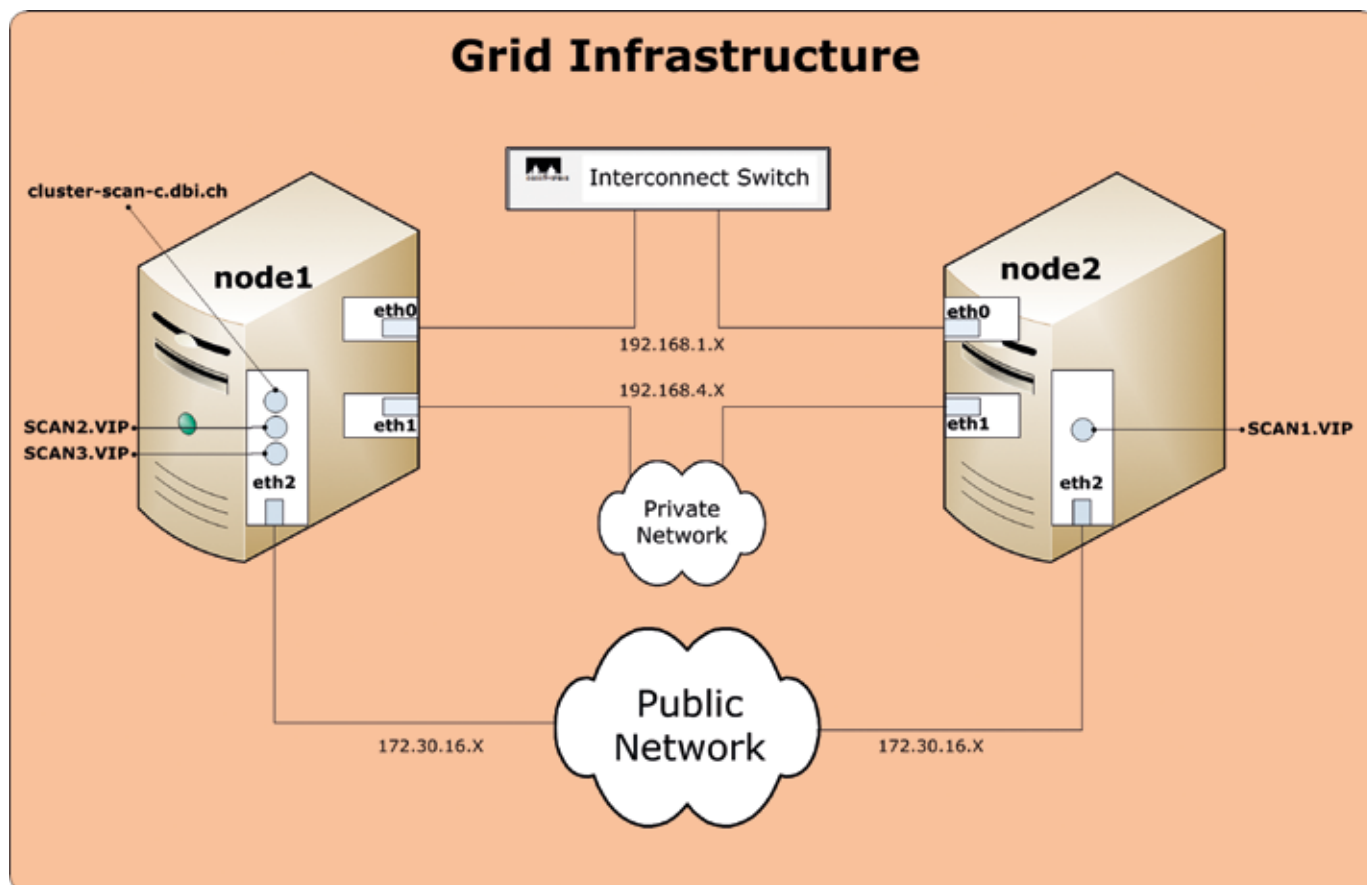


Abbildung 1: Grid-Infrastruktur

Wenn im Cluster keine zusätzliche SCAN-Infrastruktur zulässig ist, reduzieren sich die Möglichkeiten ganz erheblich, sich für spezielle Geräte aus dem 192.168.4.0/26-Netzwerk anzumelden. Die einzige Lösung besteht darin, einen Listener anzulegen, der mit einer im Netzwerk 192.168.4.0/26 gestarteten VIP verbunden ist. Zu diesem Zweck könnte man entweder ein neues Skript schreiben, das die Listener-Ressource startet/stopppt/prüft/bereinigt, oder die vom Grid Infrastructure 11.2 Framework bereitgestellten Features verwenden, wobei wir uns natürlich für die letzte Alternative entschieden haben. Warum sollten wir eine Option neu entwickeln, die es bereits gibt?

#### Erstellung der VIP

Zunächst einmal muss mit dem Benutzer „root“ im privaten Netzwerk eine VIP angelegt werden. „Private“ ist dabei nicht das Interconnect Netzwerk, sondern das Netz, das von einigen speziellen Clients im Netzwerk 192.168.4.0/26 genutzt wird. Mit nachfolgendem Be-

fehl wird eine derartige VIP erstellt. Beachten Sie bitte den Parameter „-k 2“, der angibt, dass wir uns hierbei auf ein anderes Netzwerk und nicht auf das eigentliche Standard-Public-Netzwerk beziehen („-k 2“ steht für ein zweites Netzwerk, siehe Listing 4).

#### Erstellung des Failover-Listeners

Nachdem die VIPs im privaten Netzwerk angelegt wurden, müssen wir den Failover-Listener definieren. Seit Oracle Clusterware 11.1 können Listener mit dem Befehl „srvctl“ angelegt werden (empfohlen für die Verwaltung der Oracle-Ressourcen). Die über diesen Befehl angelegten Listener sind RAC-Listener, also solche, die auf allen Knoten des Clusters laufen. In unserem Fall läuft der Listener jedoch nur auf dem Knoten, auf dem auch die VIP gestartet wurde. Dies führt zu einem kleineren Problem, mit dem wir uns später befassen wollen.

Der wichtigste Vorteil des Befehls „srvctl“ beim Anlegen des Failover-Listeners besteht jedoch darin, dass das

```
root@node1:/root/ [grid] srvctl
add vip -n node1 -k 2 -A \
erpdb-cluster-prv-v.dbi.
ch/255.255.255.192/bond
```

Listing 4

Framework, das den Listener aufruft (Skript zum Starten/Stoppen/Prüfen/Bereinigen) bereits verfügbar ist und von Grid Infrastructure 11.2 bereitgestellt wird. Der wichtigste Nachteil ist, dass man verstehen muss, was im Hintergrund geschieht, um die Abläufe tatsächlich so konfigurieren zu können, wie man sie sich vorstellt.

Beim Anlegen des Listeners und auch beim Starten dieser Listener greift „srvctl“ auf die Oracle\*Net-Konfiguration zu und ändert die Einstellungen. Diese Änderungen betreffen insbesondere die beiden Dateien „listener.ora“ und „endpoint\_listeners.ora“.

Während des Startprozesses können Softlinks von „\$ORACLE\_HOME/network/admin“ auf „TNS\_ADMIN“

```

oracle@node1:/u00/app/oracle/product/11.2.0/db_3/network/admin
[ERPDB] ls -lrt *.ora
lrwxrwxrwx 1 oracle oinstall 42 Dec 20 17:43 tnsnames.ora -> /u00/app/ora-
cle/network/admin/tnsnames.ora

lrwxrwxrwx 1 oracle oinstall 52 Jan  6 10:51 endpoints_listener.ora -> /
u00/app/oracle/network/admin/endpoints_listener.ora

lrwxrwxrwx 1 oracle oinstall 42 Jan  6 11:06 listener.ora -> /u00/app/ora-
cle/network/admin/listener.ora

```

Listing 5

```

grid@node1:/home/grid/ [grid] crsctl status res ERPDB.db -t
Cluster Resources
-----
ERPDB.db
   1          ONLINE  ONLINE          node1

grid@node1:/home/grid/ [grid]
crsctl status res ora.erpdb-cluster-prv-v.vip -t
Cluster Resources
-----ora.erpdb-
cluster-prv-v.vip
   1          ONLINE  ONLINE          node1

```

Listing 6

```

oracle@node1:/home/oracle/ [+ASM1] echo $TNS_ADMIN
/u00/app/oracle/network/admin

oracle@node1:/home/oracle/ [+ASM1]
srvctl add listener -l LISTENER_ERPDB -p TCP:1522 -o \
/u00/app/oracle/product/11.2.0/db_3 -k 2

```

Listing 7

```

oracle@node1:~/ [+ASM1] srvctl setenv listener -l \
LISTENER_ERPDB -t TNS_ADMIN=$TNS_ADMIN

oracle@node1:~/ [+ASM1]
srvctl getenv listener -l LISTENER_ERPDB
LISTENER_ERPDB:
TNS_ADMIN=/u00/app/oracle/network/admin

```

Listing 8

```

oracle@node1:/u00/app/oracle/ [+ASM1]
srvctl start listener -l LISTENER_ERPDB

```

Listing 9

(die auf /u00/app/oracle/network/admin zeigen, siehe Listing 5) durchaus verwendet werden, wobei zu beachten ist, dass „srvctl“ diese Softlinks später ändert und durch geänderte Dateien ersetzt. Listing 5 zeigt den aktuellen Speicherort der Oracle\*Net-Konfigurationsdatei.

Deshalb müssen wir nach dem Anlegen des Listeners über „srvctl“ die Variable „TNS\_ADMIN“ setzen, um eventuelle Änderungen in „\$ORACLE\_HOME/network/admin“ zu verhindern und „srvctl“ zu zwingen, beim erstmaligen Start des Listeners mit den echten Dateien unter „\$TNS\_ADMIN“ (/u00/app/oracle/network/admin) zu arbeiten.

Der über die nachfolgenden Schritte angelegte Failover-Listener wird mit der zuvor erstellten VIP verbunden: „erpdb-cluster-prv-v.vip“. Aus diesem Grund kann dieser Listener nur auf dem Knoten gestartet werden, auf dem auch die VIP ausgeführt wird. Vergewissern Sie sich zunächst, ob die VIP und die Datenbank tatsächlich auf dem ersten Knoten laufen (siehe Listing 6).

Legen Sie auf dem ersten Knoten den Failover-Listener für die ERPDB-Datenbank im privaten Netzwerk („-k 2“) an. Außerdem wird ausdrücklich angegeben, dass der Listener auf einem separaten Port (1522) laufen soll, um eventuelle Konflikte mit den SCAN-Listenern zu vermeiden (siehe Listing 7).

Setzen Sie die Umgebungsvariable „TNS\_ADMIN“ für diesen Listener, um steuern zu können, welche Änderungen beim Starten des Listeners vorgenommen werden sollen (siehe Listing

```

oracle@node1:/u00/app/oracle/
product/11.2.0/db_3/network/ad-
min/ [+ASM1] cat listener.ora
# Fokus auf dem neuen Listener
LISTENER_
ERPDB=(DESCRIPTION=(ADDRESS_
LIST=(ADDRESS=(PROTOCOL=IPC)
(KEY=LISTENER_ERPDB))))

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_
LISTENER_ERPDB=ON #
Vom Agenten eingefügte Zeile

```

Listing 10

8) und starten Sie den Listener mit „`srvctl`“ (siehe Listing 9).

Nach dem erfolgreichen Start wurde die Datei „`listener.ora`“ unter `$TNS_ADMIN` geändert (siehe Listing 10).

Wie wir sehen können, wurde der Listener „`LISTENER_ERPDB`“ auf beiden Knoten in die Datei eingefügt. Die alte Datei „`listener.ora`“ wurde als „`.bak`“-Datei gesichert (siehe Listing 11).

Zu diesem Zeitpunkt wird empfohlen, den Listener wieder zu stoppen und die Konfiguration auf beiden Knoten den Anforderungen anzupassen. Wir wollen, dass der Listener der Virtual IP lauscht, die auf dem privaten Netzwerk läuft (`erpdb-cluster-prv-v`). Dazu passen wir den „`HOST=`“-Parameter in der Datei „`listener.ora`“ entsprechend an (siehe Listing 12).

Wir haben die von SCAN-ähnlichen Listenern verwendete Adresse „`ADDRESS=(PROTOCOL=IPC) (KEY=LISTENER_ERPDB)`“ durch klassische Eingänge „`ADDRESS=(PROTOCOL=TCP)(HOST=erpdb-cluster-prv-v) (PORT=1522)(IP=FIRST)`“ ersetzt, um den Listener an die VIP und an Port 1522 zu binden. Beachten Sie, dass eine Zeile absolut nicht geändert oder angepasst werden darf (siehe Listing 13).

Sollte eine dieser beiden Zeilen „`LISTENER_ERPDB=`“ oder „`ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_ERPDB=`“ in der Datei „`listener.ora`“ nicht stehen, passt der „`srvctl`“-Prozess die Datei „`listener.ora`“ bei jedem Hochfahren erneut an seine Anforderungen an. Starten Sie den Listener dann neu (siehe Listing 14).

Vergewissern Sie sich, dass „`listener.ora`“ noch immer als Softlink auf beide Knoten zeigt (siehe Listing 15), und überprüfen Sie, ob der Listener läuft (UP, siehe Listing 16).

Um Oracle-Client-Verbindungen auf diesem Listener zu ermöglichen, empfiehlt es sich, eine „`SID_LIST`“ zu definieren (siehe Listing 17). Eine dynamische Service-Anmeldung mithilfe des Parameters „`local_listener`“ würde das Funktionieren des SCAN stören, da eine private Adresse zurückgesendet würde. Einziger Nachteil dieser Lösung ist, dass ein „`SERVICE_NAME`“ permanent auf diesen Private-Listener

```
LISTENER_ERPDB=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)
(KEY=LISTENER_ERPDB)))) # Vom Agenten eingefügte Zeile
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_ERPDB=ON # Vom Agenten eingefügte Zeile
```

Listing 11

```
oracle@node1:/u00/app/oracle/ [+ASM1]
srvctl stop listener -l LISTENER_ERPDB

# LISTENER_ERPDB=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)
(KEY=LISTENER_ERPDB)))) # Von dbi services geänderte Zeile

LISTENER_ERPDB=(DESCRIPTION=(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=TCP) (HOST=erpdb-cluster-prv-v) (PORT=1522)
(IP=FIRST)))) # Von dbi services geänderte Zeile
```

Listing 12

```
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_ERPDB=ON
# Vom Agenten eingefügte Zeile
```

Listing 13

```
oracle@node1:/u00/app/oracle/ [+ASM1]
srvctl start listener -l LISTENER_ERPDB
```

Listing 14

```
oracle@node1:/u00/app/oracle/product/11.2.0/db_3/network/admin/ [+ASM1]
ls -lrt listener.ora
lrwxrwxrwx 1 oracle oinstall 42 Jan  6 13:03 listener.ora -> /u00/app/oracle/network/admin/listener.ora
```

Listing 15

```
oracle@node1:/u00/app/oracle/ [ERPDB]
lsnrctl status LISTENER_ERPDB
...

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=erpdb-cluster-prv-v)
(PORT=1522)(IP=FIRST)))
...
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.168.4.21)(PORT=1522)))
The listener supports no services
The command completed successfully
```

Listing 16

```
SID_LIST_LISTENER_ERPDB =
(SID_LIST =
(SID_DESC =
(GLOBAL_DBNAME = ERPDB.DBI.CH )
(SID_NAME      = ERPDB )
(ORACLE_HOME   = /u00/app/oracle/product/11.2.0/db_3 )
)
)
```

Listing 17

```
oracle@node1:/u00/app/oracle/ [ERPDB]
lsnrctl status LISTENER_ERPDB
...
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=erpdb-clu-
ster-prv-v)(PORT=1522)(IP=FIRST)))
...
Listening Endpoints Summary...
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.168.4.21)(PORT=1522)))
Services Summary...
Service "ERPDB.DBI.CH" has 1 instance(s).
Instance "ERPDB", status READY, has 1 handler(s) for this service...
Service "ERPDBXDB.DBI.CH" has 1 instance(s).
Instance "ERPDB", status READY, has 1 handler(s) for this service...
Service "ERPDB_SITE1.DBI.CH" has 1 instance(s).
Instance "ERPDB", status READY, has 1 handler(s) for this service...
The command completed successfully
```

Listing 18

```
crsctl modify resource ora.erpdb-cluster-prv-v.vip -attr "HOSTING_
MEMBERS='node1 node2', ACTIVE_PLACEMENT=0"
```

Listing 19

```
oracle@node1:/u00/app/oracle/ [+ASM1]
crsctl stop res ora.erpdb-cluster-prv-v.vip -f
CRS-2673: Attempting to stop 'ora.LISTENER_ERPDB.lsnr' on 'node1'
CRS-2677: Stop of 'ora.LISTENER_ERPDB.lsnr' on 'node1' succeeded
CRS-2673: Attempting to stop 'ora.erpdb-cluster-prv-v.vip' on 'node1'
CRS-2677: Stop of 'ora.erpdb-cluster-prv-v.vip' on 'node1' succeeded
```

Listing 20

```
oracle@node1:/u00/app/oracle/ [+ASM1]
crsctl stat res ora.LISTENER_ERPDB.lsnr -p
...
START_DEPENDENCIES=hard(type:ora.cluster_vip_net2.type)
pullup(type:ora.cluster_vip_net2.type)
STOP_DEPENDENCIES=hard(intermediate:type:ora.cluster_vip_net2.type)
...
```

Listing 21

zugreifbar wäre, was eventuell „/ as sysdba“-Anmeldungen über das Passwort-File erlauben würde.

Überprüfen Sie anschließend, ob die Datenbank-Dienste beim Listener richtig registriert sind (siehe Listing 18). Passen Sie dann als Benutzer „root“ die VIP-Ressource an („hosting\_members“ und „active\_placement“, siehe Listing 19).

„ACTIVE\_PLACEMENT“ bedeutet, dass die Ressourcen nach einem Fail-over der Ressourcen nicht automatisch auf einen reparierten Knoten zurückfallen. Die VIP bleibt nach dem Fail-over auf dem verwendeten Knoten, unabhängig davon, welche Knoten im Cluster (wieder) auftauchen.

Wird der Listener „LISTENER\_ERPDB“ gestartet oder gestoppt, bedingt dies, dass auch die VIP im Netzwerk 192.168.4.0/26 gestartet oder gestoppt wird. Wenn wir die VIP stoppen, endet auch der Listener „LISTENER\_ERPDB“ (siehe Listing 20).

Da der Listener mit dem Netzwerktyp „2“ („-k 2“) angelegt wurde, ist für die Listener-Ressource eine (harte) Abhängigkeitsverbindung (Dependency Link) mit der VIP in diesem Netzwerk angelegt (siehe Listing 21). Deswegen endet der Listener, wenn die VIP gestoppt wird. Als wir den Listener mit „srvctl“ gestartet haben (siehe Listing 22), konnten wir direkt beobachten, dass die VIP tatsächlich auch auf dem erwarteten Knoten gestartet wurde (siehe Listing 23).

### Abhängigkeiten zwischen Datenbank und Listener

Abschließend verbinden wir die Datenbank mit dem Listener und der VIP. Listener und VIP werden beim Start der Datenbank („schwach“) vorausgesetzt:

- *Hard*  
ora.DATA\_INFRRA.dg: DATA\_INFRRA-

```
oracle@node1:/u00/app/oracle/ [ERPDB]
srvctl start listener -l LISTENER_ERPDB -n node1
```

Listing 22

Plattengruppen sind für den Start der Datenbank erforderlich

- **Weak**

Typ: ora.scan\_listener.type,ora.erpdb-cluster-prv-v.vip, ora.LISTENER\_ERPDB.lsnr. VIP, SCAN und LISTENER sollten laufen, jedoch die DB nicht daran hindern, zu starten (siehe Listing 24).

Nun können wir die Datenbank mit „crsctl relocate -f“ von einem Knoten auf einen anderen verschieben (siehe Listing 25).

### Nachteile

Einen Nachteil bringt diese Lösung jedoch mit sich. Der Listener kann natürlich nicht auf dem Knoten gestartet werden, auf dem die VIP nicht bereitsteht. Diese Situation lässt sich in der Protokolldatei „oraagent\_oracle.log“ unter „/u00/app/11.2.0/grid/log/{NODE}/agent/crsd/oraagent\_oracle“ auf dem Knoten ablesen, auf dem der Listener nicht gestartet wurde. Zunächst weist das Logfile darauf hin, dass die VIP fehlt, was in Ordnung ist (siehe Listing 26). Dann schlägt der nach jedem Start durchgeführte Überprüfungsprozess fehl (siehe Listing 27).

Insgesamt geht Grid Infrastructure mit dieser Situation recht elegant um und stoppt den Listener. Dabei wird auch der Status der Ressource von „stopping“ auf „offline“ geändert (siehe Listing 28).

Der angestrebte Status bleibt weiterhin „online“, der aktuelle Ressourcen-Status ist jedoch „offline“. Diese Situation kann einige Beeinträchtigungen bei der Überwachung nach sich ziehen, da der „Zielstatus“ nicht mit dem „aktuellen Status“ übereinstimmt. Gewöhn-

```
oracle@node1:/u00/app/oracle/ [+ASM1]
crsctl status res ora.erpdb-cluster-prv-v.vip -t
Cluster Resources
-----
ora.erpdb-cluster-prv-v.vip
1 ONLINE ONLINE
node1
```

Listing 23

```
crsctl modify resource ERPDB.db -attr \ „START_DEPENDENCIES='hard(ora.
DATA_INFRA.dg) \
weak(type:ora.scan_listener.type, \
ora.erpdb-cluster-prv-v.vip,ora.LISTENER_ERPDB.lsnr) \
pullup(ora.DATA_INFRA.dg, \
ora.erpdb-cluster-prv-v.vip,ora.LISTENER_ERPDB.lsnr)', \
STOP_DEPENDENCIES='hard(intermediate:ora.asm,shutdown:ora.DATA_INFRA.
dg)'”
```

Listing 24

```
oracle@node1:/u00/app/oracle/ [+ASM1]
crsctl relocate res ERPDB.db -f

CRS-2673: Attempting to stop 'ora.LISTENER_ERPDB.lsnr' on 'node1'
CRS-2677: Stop of 'ora.LISTENER_ERPDB.lsnr' on 'node1' succeeded
CRS-2673: Attempting to stop 'ora.erpdb-cluster-prv-v.vip' on 'node1'
CRS-2677: Stop of 'ora.erpdb-cluster-prv-v.vip' on 'node1' succeeded
CRS-2672: Attempting to start 'ora.erpdb-cluster-prv-v.vip' on 'node2'
CRS-2676: Start of 'ora.erpdb-cluster-prv-v.vip' on 'node2' succeeded
CRS-2673: Attempting to stop 'ERPDB.db' on 'node1'
CRS-2677: Stop of 'ERPDB.db' on 'node1' succeeded
CRS-2672: Attempting to start 'ERPDB.db' on 'node2'
CRS-2676: Start of 'ERPDB.db' on 'node2' succeeded

oracle@node1:/u00/app/oracle/ [+ASM1]
-----Cluster
Resources
-----
ERPDB.db
1 ONLINE ONLINE node2
```

Listing 25

```
2012-01-11 10:59:54.706: [ora.LISTENER_ERPDB.lsnr][1359505728]
{1:64912:13988} [check] Warning, could not get the dependent VIP address
2012-01-11 10:59:54.706: [ora.LISTENER_ERPDB.lsnr][1359505728]
{1:64912:13988} [check] return val (clsagfwret) = 0
```

Listing 26

```
2012-01-11 11:03:44.727: [ora.LISTENER_ERPDB.lsnr][1274325312]
{1:64912:14075} [check] (:CLSN00010:)Connecting to (DESCRIPTION=(ADDRESS
=(PROTOCOL=TCP)(HOST=erpdb-cluster-prv-v)(PORT=1522)(IP=FIRST)))
2012-01-11 11:03:44.727: [ora.LISTENER_ERPDB.lsnr][1274325312]
{1:64912:14075} [check] (:CLSN00010:)TNS-12541: TNS:no listener
2012-01-11 11:03:44.727: [ora.LISTENER_ERPDB.lsnr][1274325312]
{1:64912:14075} [check] (:CLSN00010:) TNS-12560: TNS:protocol adapter
error
2012-01-11 11:03:44.727: [ora.LISTENER_ERPDB.lsnr][1274325312]
{1:64912:14075} [check] (:CLSN00010:) TNS-00511: No listener
2012-01-11 11:03:44.727: [ora.LISTENER_ERPDB.lsnr][1274325312]
{1:64912:14075} [check] (:CLSN00010:) Linux Error: 111: Connection
refused
```

Listing 27

```
2012-01-11 11:05:20.075: [ AGFW][1404766528] {1:64912:14108} ora.
LISTENER_ERPDB.lsnr node2 1 state changed from: STOPPING to: OFFLINE
2012-01-11 11:05:20.076: [ AGFW][1404766528] {1:64912:14108} Agent
sending last reply for: RESOURCE_STOP[ora.LISTENER_ERPDB.lsnr node2
1] ID 4099:100765
```

Listing 28

```
grid@node2:/u00/app/11.2.0/ [grid]
crsctl stat res ora.LISTENER_ERPDB.lsnr -t
Local Resources
-----ora.
LISTENER_ERPDB.lsnr
      ONLINE  ONLINE      node1
      ONLINE  OFFLINE    node2
```

Listing 29

```
# srvctl add vip -n node1priv-vip -k 2 -A node1DR-vip/255.255.255.0/
bond1 -v
# srvctl add vip -n node2priv-vip -k 2 -A node1DR-vip/255.255.255.0/
bond1 -v
```

Listing 30

lich weist eine derartige Konfiguration auf ein Problem mit einer Ressource hin. In unserem Fall hingegen handelt es sich nicht um ein Problem, sondern um eine normale Erscheinung. Diese Situation muss vom Überwachungsteam differenziert in Betracht gezogen werden (siehe Listing 29).

Beachten Sie bitte, dass die Anzahl der Server, auf denen eine Ressource laufen kann, durch die Eigenschaft „Kardinalität“ eingeschränkt werden kann. Leider enthält der Ressourcen-

Typ Listener diese Eigenschaft nicht, sodass ich diese Situation nicht beheben konnte.

**Fazit**

Mit dieser Strategie konnten wir mit minimalem Aufwand eine zuverlässige und robuste Lösung implementieren. Auch wenn diese Lösung einige Nachteile birgt und nicht unbedingt perfekt ist, lässt sich auf diesem Wege dennoch ohne großen Aufwand eine mangelnde Option ausgleichen. Wir hoffen

selbstverständlich, dass die kommenden Oracle-Versionen die Möglichkeit bieten, mehrere SCANS in den Clustern anzulegen/zu verwalten.

Der Autor bedankt sich bei Martin Bach, der diesen Artikel vor der Veröffentlichung durchgesehen hat. Er ist Oracle ACE und Mitglied der Experten-Gruppe „OAK Table“. Vor allem ist er aber ein freundlicher und hilfsbereiter Mensch. Er merkte an, dass bei der Erstellung der VIP auch folgende Strategie angewendet werden kann, um benannte VIPs für die Knoten anzulegen (siehe Listing 30).

Dies hat den Vorteil, dass die VIP namentlich angesprochen werden kann, auch wenn in diesem Fall auf dem Client bei der Herstellung der Datenbankverbindung eine „ADDRESS\_LIST“ verwendet werden muss. Grid Infrastructure wird damit zu einer stabilen und kompletten Cluster-Lösung, die sich mit ASM durch optimale Host- und Sicherheits-Redundanz auszeichnet. Oracle bestärkt seine Strategie und setzt andere Clusterlösungen weiterhin stark unter Druck.

Yann Neuhaus  
yann.neuhaus@dbi-services.com



## Berliner Expertenseminare

- Wissensvertiefung für Oracle-Anwender
- Mit ausgewählten Schulungspartnern
- Von Experten für Experten
- Umfangreiches Seminarangebot

5./6. Juni 2012

Solaris 11 Zonen – deep dive  
Referent: Heiko Stein

18./19. September 2012

Performance  
Referent: Lutz Fröhlich

16./17. Oktober 2012

Oracle ADF  
Referent: Ulrich Gerkmann-Bartels

www.doag.org

DOAG

Deutsche ORACLE-Anwendergruppe e.V.